On Some Unsupervised Learning Problems for Highly Dependent Time Series

Azadeh Khaleghi

Université Lille I - UFR Scientifique de Lille - INRIA Lille - Nord Europe

École Doctorale Sciences pour l'Ingénieur

Thèse de Doctorat

Spécialité : **Mathématiques**

présentée par

Azadeh KHALEGHI

SUR QUELQUES PROBLÈMES NON-SUPERVISÉS IMPLIQUANT DES SÉRIES TEMPORELLES HAUTEMENT DÉPENDANTES

dirigée par **Daniil RYABKO**

Rapporteurs:	M. Francis	**BACH**	INRIA Paris
	M. Jean-Philippe	**VERT**	Mines ParisTech

Soutenue publiquement le **18 Novembre, 2013** devant le jury composé de:

M. Francis	**BACH**	INRIA Paris	Rapporteur
M. Christophe	**BIERNACKI**	Université Lille I	Examinateur
M. Olivier	**CATONI**	École Normale Supérieur Paris	Examinateur
M. Patrick	**GALLINARI**	Université Paris 6	Examinateur
M. Daniil	**RYABKO**	INRIA Lille	Directeur
M. Jean-Philippe	**VERT**	Mines ParisTech	Rapporteur

In loving memory of my dad, whom I miss so dearly, and whose absence I feel evermore deeply.

Acknowledgements

An enormous debt of gratitude is due to many without whom the present work would not have been possible.

I am deeply grateful to my advisor, Daniil Ryabko, for his guidance, generosity, availability and kindness. I have always valued the opportunity to work with such a wonderful researcher, whose knowledge and vision have been fundamental to this thesis. Daniil, I cannot thank you enough!

It is a pleasure to recognize my dissertation committee members, Francis Bach, Jean-Philippe Vert, Christophe Biernacki, Olivier Catoni and Patrick Gallinari, for taking interest in my research and accepting to examine my work. Special thanks to Francis and Jean-Philippe for their thorough reviews and insightful comments. I also gratefully acknowledge INRIA for financing my PhD studies, and for doing a great job at facilitating research.

I would like to thank the SequeL team for making the past three years of my life so amazingly memorable. I have always enjoyed our time spent both inside and outside the lab, from exchanging research ideas to simply grabbing a drink after work, playing football, or attending concerts held by some of you! Listing all the individual permanents, students, post-docs and visiting fellows, all of whom have, in one way or another, contributed to this unique experience, certainly carries a risk I refuse to take! but you all know who you are, so, thank you all! De plus, les francophones de l'équipe, je vous remercie sincèrement pour tout ce que vous m'avez appris; merci pour votre générosité, votre patience et votre gentillesse.

Last, but never least, I would like to thank my family. I dedicate this dissertation to my lovely mom, to the loving memory of my dad, and to my brother, Khashyar. My gratitude for their endless love, support and encouragement over the years is beyond words.

Résumé

Cette thèse est consacrée à l'analyse théorique de problèmes non supervisés impliquant des séries temporelles hautement dépendantes. Plus particulièrement, nous abordons les deux problèmes fondamentaux que sont le problème d'estimation des points de rupture et le partitionnement de séries temporelles. Ces problèmes sont abordés dans un cadre extrêmement général où les données sont générées par des processus stochastiques ergodiques stationnaires. Il s'agit de l'une des hypothèses les plus faibles en statistiques, comprenant non seulement, les hypothèses de modèles et les hypothèses paramétriques habituelles dans la littérature scientifique, mais aussi des hypothèses classiques d'indépendance, de contraintes sur l'espace mémoire ou encore des hypothèses de mélange. En particulier, aucune restriction n'est faite sur la forme ou la nature des dépendances, de telles sortes que les échantillons peuvent être arbitrairement dépendants. Pour chaque problème abordé, nous proposons de nouvelles méthodes non paramétriques et nous prouvons de plus qu'elles sont, dans ce cadre, asymptotiquement consistantes.

Pour l'estimation de points de rupture, la consistance asymptotique se rapporte à la capacité de l'algorithme à produire des estimations des points de rupture qui sont asymptotiquement arbitrairement proches des vrais points de rupture. D'autre part, un algorithme de partitionnement est asymptotiquement consistant si le partitionnement qu'il produit, restreint à chaque lot de séquences, coïncides, à partir d'un certain temps et de manière consistante, avec le partitionnement cible.

Nous montrons que les algorithmes proposés sont implémentables efficacement, et nous accompagnons nos résultats théoriques par des évaluations expérimentales.

L'analyse statistique dans le cadre stationnaire ergodique est extrêmement difficile. De manière générale, il est prouvé que les vitesses de convergence sont impossibles à obtenir. Dès lors, pour deux échantillons générés indépendamment par des processus

ergodiques stationnaires, il est prouvé qu'il est impossible de distinguer le cas où les échantillons sont générés par le même processus de celui où ils sont générés par des processus différents. Ceci implique que des problèmes tels le partitionnement de séries temporelles sans la connaissance du nombre de partitions ou du nombre de points de rupture ne peut admettre de solutions consistantes.

En conséquence, une tâche difficile est de découvrir les formulations du problème qui en permettent une résolution dans ce cadre général. La principale contribution de cette thèse est de démontrer (par construction) que malgré ces résultats d'impossibilités théoriques, des formulations naturelles des problèmes considérés existent et admettent des solutions consistantes dans ce cadre général. Ceci inclut la démonstration du fait que le nombre de points de rupture corrects peut être trouvé, sans recourir à des hypothèses plus fortes sur les processus stochastiques. Il en résulte que, dans cette formulation, le problème des points de rupture peut être réduit à du partitionnement de séries temporelles.

Les résultats présentés dans ce travail formulent les fondations théoriques pour l'analyse des données séquentielles dans un espace d'applications bien plus large.

Abstract

This thesis is devoted to the theoretical analysis of unsupervised learning problems involving highly dependent time-series. Two fundamental problems are considered, namely, the problem of change point estimation as well as that of time-series clustering. The problems are considered in an extremely general framework, where the data are assumed to be generated by arbitrary, unknown stationary ergodic process distributions. This is one of the weakest assumptions in statistics, because it is more general than the parametric and model-based settings, and it subsumes most of the non-parametric frameworks considered for this class of problems. These assumptions typically have the premise that each time-series consists of independent and identically distributed observations or that it satisfies certain mixing conditions.

For each of the considered problems, novel nonparametric methods are proposed, and are further shown to be asymptotically consistent in this general framework. For change point estimation, asymptotic consistency refers to the algorithm's ability to produce change point estimates that are asymptotically arbitrarily close to the true change points. On the other hand, a clustering algorithm is asymptotically consistent, if the output clustering, restricted to each fixed batch of sequences, consistently coincides with the target clustering from some time on. The proposed algorithms are shown to be efficiently implementable, and the theoretical results are complemented with experimental evaluations.

Statistical analysis in the stationary ergodic framework is extremely challenging. In general, rates of convergence (even of frequencies to respective probabilities) are provably impossible to obtain for this class of processes. As a result, given a pair of samples generated independently by stationary ergodic process distributions, it is provably impossible to distinguish between the case where they are generated by the same process or by two different ones. This in turn, implies that such problems as time-

series clustering with unknown number of clusters, or change point detection, cannot possibly admit consistent solutions. Thus, a challenging task is to discover the problem formulations which admit consistent solutions in this general framework. The main contribution of this thesis is to constructively demonstrate that despite these theoretical impossibility results, natural formulations of the considered problems exist which admit consistent solutions in this general framework. Specifically, natural formulations of change-point estimation and time-series clustering are proposed, and efficient algorithms are provided, which are shown to be asymptotically consistent under the assumption that the process distributions are stationary ergodic. This includes the demonstration of the fact that the correct number of change points can be found, without the need to impose stronger assumptions on the process distributions. It turns out that in this formulation the change point estimation problem can be reduced to time-series clustering.

The results presented in this work lay down the theoretical foundations for the analysis of sequential data in a broad range of real-world applications.

List of author's related publications

Parts of the results presented in this thesis have appeared in the following publications.

A. Khaleghi, D. Ryabko, J. Mary, and P. Preux. Online clustering of processes. In *the international conference on Artificial Intelligence & Statistics (AI & Stats)*, La Palma, Canary Islands, 2012.

A. Khaleghi and D. Ryabko. Locating changes in highly dependent data with unknown number of change points. In *Neural Information Processing Systems (NIPS)*, Lake Tahoe, Nevada, United States, 2012.

A. Khaleghi and D. Ryabko. Nonparametric multiple change point estimation in highly dependent time series. In the *Proceedings of the 24th International Conference on Algorithmic Learning Theory (ALT)*, Singapore, 2013.
(Received the E.M. Gold award for the best student paper.)

A. Khaleghi and D. Ryabko. Asymptotically consistent estimation of the number of change points in highly dependent time series. In *proceedings of the 31st International Conference on Machine Learning (ICML)*, Beijing, China, 2014.

A. Khaleghi and D. Ryabko Nonparametric change point estimation in stationary ergodic time series. *submitted for publication in a journal.*

A. Khaleghi, D. Ryabko, J. Mari, P. Preux, Consistent algorithms for clustering time series. *submitted for publication in a journal.*.

List of main theorems

Theorem. *Algorithm 1 is an asymptotically consistent multiple change point estimator: given a sequence $\boldsymbol{x}$ with a known number κ of change points $\theta_k n,\ k = 1..\kappa$, where n denotes the length of $\boldsymbol{x}$ and the sequence between every pair of consecutive change points is generated by an (unknown) stationary ergodic process distribution, we have*

$$\lim_{n\to\infty} \sup_{k=1..\kappa} |\widehat{\theta}_k(n) - \theta_k| = 0 \ a.s.$$

Theorem. *Algorithm 2 is an asymptotically consistent list-estimator: given a sequence $\boldsymbol{x}$ with (an unknown) number κ of change points $\theta_k n,\ k = 1..\kappa$ and a parameter $\lambda \in (0, \lambda_{\min}]$ (where $\lambda_{\min} \in (0,1)$ denotes the (unknown) minimum separation of the true change points), it produces a list of at least, but possibly more than, κ estimates such that*

$$\lim_{n\to\infty} \sup_{k=1..\kappa} |\widehat{\theta}_k(n) - \theta_k| = 0 \ a.s.$$

Theorem. *Algorithm 3 is an asymptotically consistent multiple change point estimator: given a sequence $\boldsymbol{x}$ with (an unknown) number κ of change points, and the total number r of process distributions that generate $\boldsymbol{x}$, it finds the correct number of change points and estimates the changes so that w.p.1 for large enough n on we have $\widehat{\kappa} = \kappa$ and*

$$\lim_{n\to\infty} \sup_{k=1..\kappa} |\widehat{\theta}_k(n) - \theta_k| = 0 \ a.s.$$

Theorem. *Algorithm 5 is an asymptotically consistent online time-series clustering algorithm: for every fixed batch sequences, each generated by one of κ stationary ergodic process distribution, w.p.1, from some time on it groups together those and only those sequences that are generated by the same process distribution, provided that κ is known.*

Contents

Chapter 1

Introduction

The common objective in all unsupervised learning problems is to infer underlying structures in a given dataset, while no information is available beyond the raw data. This class of problems appears in a broad range of real-world scenarios. In many modern applications, vast amounts of data are produced. It is usually the practitioners' objective to put the data into effective use with no a priori knowledge as to how to achieve this purpose. In other scenarios the goal of the analysis is predefined, but there is little or no access to the correct solution. Clearly in these situations, any theoretical assumption on the nature of the data may prove infeasible. Indeed, the main purpose of the analysis is to use the data in order to extract information about the unknown underlying phenomena, rather than to confirm the model used. This calls for general unsupervised learning methods that are guaranteed to work with little assumptions.

A familiar and fundamental example of unsupervised learning problems is clustering. Here the objective is to find a few natural and meaningful groups (clusters) in a large data set, with the hope that the resulting set of clusters would help the domain experts gain more insight into the nature of the data. Another example is change point analysis where it is desired to identify the point in time at which the process distribution of a given sequence of observations has changed. Blind-source separation is yet another example, where it is required to separate a set of source signals from a given mixed signal with limited information on the processes that generate the data. Other examples include density estimation and outlier detection.

In this thesis we focus on unsupervised learning problems concerning time series where data are in the form of one or several sequences of observations. The analysis of

time series is motivated by many scientific applications from a variety of disciplines such as marketing and finance, biological and medical research, speech processing, social and environmental sciences, and many more. We consider a statistical approach where the data are viewed as samples generated by discrete-time stochastic process distributions. A large body of work exists on the statistical analysis of time series. However, the considered frameworks are usually restricted by strong assumptions. The process distributions are either assumed to come from parametric families or, in non-parametric settings it is typically assumed that the data are composed of independently and identically distributed (i.i.d) observations, or that they satisfy certain mixing conditions. Efficient algorithms have been developed that are guaranteed to work in these frameworks. However, such methods may not be useful in many practical situations, as such strong statistical assumptions do not necessarily hold in most real-world scenarios.

An important question forming the basis of our perspective in this thesis is whether it is possible to solve unsupervised learning problems without the need to make strong assumptions. That is, we investigate the possibilities and limitations of unsupervised learning methods involving time series data when as little assumptions as possible are made. To this end, we consider an extremely general framework, which subsumes most of the assumptions traditionally imposed in statistical time series analysis. The process distributions that generate the data are unknown. Our only assumption is that the data are generated by stationary ergodic process distributions. We constructively demonstrate that appropriate formulations of two classes of unsupervised learning problems, namely, change point estimation and online time series clustering admit solutions which, as we further show, are guaranteed to be asymptotically correct in the general framework considered.

Before proceeding to a more detailed exposition of our results, let us provide some motivating examples. In modern consumer markets, customer behaviour is usually monitored over time through, for example, fidelity reward cards. This type of information is further used to develop profitable market attribution strategies. Appropriate strategic insights may be obtained by grouping the consumers based on, for instance, their purchase records. In this example the collected data are in the form of time series of possibly highly dependent observations where there may even be some dependence between the different sequences. Moreover, there is no a priori knowledge as to what type of groups are to be identified. As another practical example, let us consider the

problem of author attribution in a large text written collaboratively by a fixed number of authors. For simplicity, assume that the text is composed of individual blocks written separately by each individual author. It may be desired to separate these blocks. To this end, the text may be considered as a long sequence (of letters) with change points. The change points signify the parts of the text where a block of text written by one author ends, and another block written by a different author begins. The parts of the text generated by different authors may be thought of as sequences generated by different (unknown) process distributions. The series are obviously highly dependent, and the difference between the process distributions is very likely to be only in the unknown dependence structure of the individual segments, which can possibly be in the long-range form. This may be reflected, for example, by each author's unique writing style. As a result, it is important to consider a framework that takes such dependencies into account. The differences in the marginal distributions (of letters), if present at all, are negligible and not likely to be identifiable. Such potential applications can be successfully modelled by the general statistical framework that we consider in this thesis, while the settings with stronger assumptions may prove impractical in addressing such problems.

1.1 Stationary ergodic framework

The main and only statistical assumption that we make is that each time series is generated by a stationary ergodic process distribution. Intuitively, stationarity means that the time index does not bear any information. That is, the time index at which the observations have started to be recorded is not important. Ergodicity means that asymptotically, any realization of the time series reveals complete information about its distribution. The assumption that a process distribution is stationary ergodic is one of the weakest assumptions in statistics. Indeed, by the ergodic decomposition, any stationary process can be represented as a mixture of stationary ergodic processes. This means that any realization of a stationary process can be considered a realization of a stationary ergodic process (Billingsley, 1961). By this argument, we can safely state that the setting considered in this thesis is characterized by no other assumption than stationarity, which is in turn extremely mild. Note that we make no such assumptions as independence within the samples in each sequence, or between the different sequences.

Moreover, the process distributions need not satisfy any mixing conditions: the data may have long-range dependence, and infinite memory.

Clearly this framework, however general, is still not assumption-free and cannot possibly address all real-world applications, as not all time series are expected to be modelled by stationary ergodic process distributions. For example, consider time series corresponding to video recordings of human locomotion. While the act of "running" can be thought of as an ergodic motion, a "single jump" can not. Note, however, that the stationary ergodic assumption is still rather mild as compared to the statistical assumptions typically made in the literature. As a result, this framework can be suitable for a broader range of applications. At the same time, this general setting calls for a very different kind of analysis. The methods designed for the more restrictive settings typically considered in the literature, make extensive use of rates of convergence. In contrast, rates of convergence (even of frequencies to respective probabilities) are impossible to obtain for stationary ergodic process distributions; see for example, (Shields, 1996). This entails such strong impossibility results as the non-existence of consistent tests for homogeneity. Specifically, as shown in (Ryabko, 2010b), given two samples generated by stationary ergodic process distributions, it is impossible to distinguish between the case where they are generated by the same source or by two different ones; this result holds even if the given samples are binary-valued. Due to these restrictions, for many statistical problems it is unclear whether consistent solutions even exist in this setting. Thus, one of the main challenges when considering the stationary ergodic framework consists in finding the appropriate problem formulations that, without the need for stronger assumptions, can admit consistent solutions. Moreover, the theoretical guarantees obtained for algorithms involving this class of processes, cannot be beyond asymptotic results. As a result, the algorithms developed for this framework are forced not to rely on rates of convergence. A possible downside of this scenario is that the asymptotic results obtained for these methods cannot be strengthened and in particular, rates of decrease of error cannot be obtained. On the bright side, however, this limitation may be seen as an advantage in the sense that the rate-free methods designed to work in this general framework are potentially applicable to a much wider range of real-world scenarios.

1.2 Proposed formulations and main results

In this section we give an overview of the proposed formulations along with a brief description of our results. A summary of our main contributions is also provided in Section 1.5.

In light of the theoretical impossibility results discussed above, not every statistical problem involving time series can be expected to admit a consistent solution in the stationary ergodic framework. Some examples of the problems that, as follows from the results of (Ryabko, 2010b), are provably impossible to be solved in this setting include time series clustering for unknown number of clusters, change point detection, and change point estimation when the number of change points is unknown. In this thesis we consider two classical problems namely, change point estimation and online time series clustering, introduced in Sections 1.2.1 and 1.2.2 respectively. For each problem we provide appropriate formulations that, as we demonstrate, admit asymptotically correct solutions under the assumption that the data are stationary ergodic. An overview of our formulations follows.

1.2.1 Change point estimation

Change point estimation involves an a posteriori analysis of a given heterogenous time series, as a means to locate the point in time where the process distribution that generates the samples has abruptly changed. This is an important tool in many practical applications. Indeed, solutions to many real-world problems are directly or indirectly related to the inference of changes in distribution of sequential observations. A number of application areas include bioinformatics (Picard et al., 2005, Vert and Bleakley, 2010), financial modeling (Talih and Hengartner, 2005, Lavielle and Teyssiere, 2007), network traffic data analysis (Tartakovsky et al., 2006, Lévy-Leduc and Roueff, 2009, Lung-Yut-Fong et al., 2012), fraud and anomaly detection (Bolton and Hand, 2002, Akoglu and Faloutsos, 2010) and speech segmentation (Fox et al., 2011).

The problem of change point estimation may be introduced as follows. A given sequence $\boldsymbol{x} := X_1, \ldots, X_n$ is formed as the concatenation of $\kappa + 1$ non-overlapping segments where the consecutive segments are generated by different process distributions. The index where one segment ends and another starts is called a change point. Specifically, a change point refers to the index at which the process distribution of $\boldsymbol{x}$ has

changed. Thus, $\boldsymbol{x}$ has κ change points, which are unknown and have to be estimated.

In this work we consider the problem of change point estimation in the general stationary ergodic paradigm discussed in Section 1.1. Our only assumption is that each segment is generated by some (unknown) stationary ergodic process distribution. The joint distribution over the samples can be otherwise arbitrary. The marginal distributions of any given fixed size (e.g. single-dimensional marginals) before and after a change point are allowed to be the same. For example the mean, variance etc. may remain unchanged throughout the entire sequence. This means that we consider the most general and perhaps the most natural notion of change, that is, change in the distribution of the data. We also do not make any conditions on the densities of the marginal distributions (the densities may not exist). Since little assumptions are made on how the data are generated, this setting is especially suitable for highly dependent time series, potentially accommodating a variety of new applications.

The objective is to construct algorithms that simultaneously estimate all κ change points in $\boldsymbol{x}$. The algorithms must be asymptotically consistent in the sense that their estimation error on all κ change points must be arbitrarily small if the sequence is sufficiently long. Note that the asymptotic regime is with respect to the length n of the sequence. In other words, the problem is considered offline and the sequence does not grow with time. As follows from the theoretical impossibility results discussed earlier, the number of change points cannot be estimated under these general assumptions. Usually in the literature, stronger statistical assumptions are imposed on the process distributions so that with the aid of rates of convergence it would be possible to estimate κ. However, rates of convergence are not available in this setting; thus, a natural question would be whether there exist formulations for which consistent solutions can be obtained, without the need to place any restrictions (beyond the stationarity and ergodicity) on the process distributions. To address this question, we consider three different formulations of change point problem, for each of which we provide a consistent solution; our methods are presented in Chapter 3. An overview of our formulations follows.

1. In the first formulation, we assume that the correct number κ of change points is known and provided to the algorithm. The objective is to have an algorithm that is asymptotically consistent in the general statistical framework described above. This scenario may arise in many applications. For example, the objective of a medical

study may be to investigate the causes of seizures, using electroencephalography (EEG) recordings of patients. The number of seizures that each patient has suffered may be known a priori, and the goal may be to identify the changes in the brain activity as reflected by the EEG recordings. Another example involves searching within a long text, or a long speech signal, in order to identify the points at which the subject of discussion, the writers or the speakers have changed. In many situations, while such change points are unknown, the total number of such changes may be known in advance.

We propose an efficient change point estimation algorithm, and prove it to be asymptotically consistent, under the assumption that the process distributions that generate the data are stationary ergodic, and that the correct number of change points is provided.

2. In the second formulation, we assume that κ is unknown, but we make no attempt to estimate it. Instead, our goal is to produce a list of at least κ candidate estimates, sorted in such a way that its first κ elements converge to the true change points. We call a method that achieves this goal for long enough $\boldsymbol{x}$, an asymptotically consistent "list-estimator". The idea of producing a list of potential answers as a relaxation to producing the exact solution has previously appeared in the literature. In error-control coding theory, a so-called "list-decoder" is used to produce a set of possible codewords as output (Sudan, 2000). A list model for clustering has been proposed by (M. et al., 2008) where the objective is to produce a small number of clusterings such that at least one has an arbitrarily small error with respect to the ground-truth. Note that our objective is somewhat stronger since the produced list of candidate change points is sorted in such a way that its first κ elements are consistent estimates of the true change points.

List-estimation can prove useful in practice. Indeed, in many scenarios the number of change points can be unknown. However, once provided with the sorted list produced by a consistent list-estimator, the domain expert will be able to identify the true change points much faster. A list-estimator can be especially useful when the input sequence is long. This is due to the fact that after list-estimation, a much smaller portion of the data (i.e. that specified by the output of the list-estimator) will be left for analysis. If the list-estimator is consistent, the first κ elements converge to the true change points. At this point, the task of the expert would be merely to reject the tail of redundant estimates. This in turn requires a smaller amount of time and effort as compared to the

brute-force inspection of the entire data. Let us revisit the example of inspecting a long text (or speech recording), as a means to identify the points in the data that correspond to changes in topic. Assume that, in contrast to the scenario discussed above, we have no a priori knowledge of the number of change points in this case. Thus, every point in the sequence is a potential change point. Using a list-estimator we can identify the changes by confining our inspection to the list-estimator's output, potentially saving substantial time and effort in the overall analysis. Specifically, noting that the first κ elements of the list correspond to the true change points, we can sequentially inspect the list of candidate estimates produced by the list-estimator, stopping the process as soon as an estimate does not seem to correspond to a true change of topic. In much the same way, the list-estimator can be used in almost all other such applications as video surveillance, bioinformatics, market analysis, etc., where it is required to locate an unknown number of change points within a long sequence of highly dependent observations and an expert is available to further inspect the list of candidate estimates.

We show that consistent list-estimation is possible under the assumption that the process distributions generating the time series are stationary ergodic. Specifically, we propose an efficient list-estimator that generates a (sorted) list of change point estimates. As we further show, the proposed method is asymptotically consistent: the first κ elements of its output list converge to the true change points, provided that the unknown process distributions that generate the data are stationary ergodic.

3. In the third formulation, we assume that while κ is unknown, an additional parameter, namely the total number of process distributions that generate the data is provided. For instance, there may only be two distributions that generate the time series; however, they may be alternating many times so that κ is much larger than 1. In this case, we are able to construct an asymptotically consistent algorithm that finds κ and locates the changes. This additional parameter has a natural interpretation in many real-world applications. The simple example where a pair of distributions alternate in generating a sequence with many change points, may correspond to a system whose behaviour over time alternates between normal and abnormal. This may also be a suitable model for video surveillance or fraud detection. Another application is to identify the coding vs. non-coding regions of DNA sequences. Recall the problem of author attribution mentioned earlier, where the objective is to find the points in a collaboratively written text at which the author has changed. In this case, the knowl-

edge of the number of participating authors amounts to knowing the total number of process distributions. Similarly, this model may be suitable for the problem of speech segmentation, in the case where the total number of speakers is known. In the example motivating the previous two formulations, we depicted a natural scenario where it is desired to locate the points in a given long text (or speech signal) that correspond to changes in the topic of discussion. While the number of such change points may be unknown, we may have a priori knowledge of the total number of topics covered, which may have been revisited many times over the course of discussion (in the text or speech data).

We provide an efficient change point estimation algorithm, that given the correct number of process distributions that generate the data, finds the number of change points, and locates the changes; the proposed method is shown to be asymptotically consistent under the assumption that the process distributions that generate the data are stationary ergodic. We show that in this formulation, the change point estimation problem can be reduced to time series clustering.

1.2.2 Online time series clustering

Clustering is a well-explored problem in machine learning, and is key to many applications in almost all fields of science. The goal is to partition a given dataset in a *natural* way, potentially revealing an underlying structure in the data. In this thesis we consider the problem of time series clustering, where the data to be clustered are sequences generated by discrete-time stochastic process distributions. The problem is online, meaning that the data arrive in an online fashion so that new data are revealed at every time-step, either as subsequent segments of previously observed sequences, or as new sequences. This version of the problem has many real-world applications. Indeed, in many cases the data arrive dynamically, with both new sources being added and previously available sources generating more data. It is important for a clustering algorithm to cluster recently observed data points as soon as possible, without changing its decision on the points that have already been clustered correctly. At the same time, some of the previous clustering decisions made about the sequences observed earlier may be incorrect, and may need to be updated given new observations. For instance, in the marketing attribution application discussed earlier, new customers are dynamically introduced to the market. A marketing strategy based on a batch clustering algorithm

may not be effective in this case as the offline method may be constantly confused by the continuous arrival of new information. As a result, intelligent online methods must be used that are robust with respect to the dynamic nature of the data.

We consider the following formulation of the problem. A growing body of sequences of data is given, where the number of sequences as well as the sequences themselves grow with time. We have no control over this evolution; it is only assumed that the length of each individual sequence tends to infinity. Each sequence is generated by one of κ unknown, stationary ergodic process distributions. At every time-step new samples are observed, which either extend a previously received sequence, or form a new one. Our objective is to cluster the sequences based on the process distributions that generate them. That is, we define the target clustering as the partitioning of the samples into κ clusters, where two samples are grouped together if and only if they are generated by the same process distribution. We define a clustering algorithm to be asymptotically consistent, if the clustering restricted to every fixed batch of sequences, coincides with the target clustering from some time on. As in the previously considered problems, the samples are allowed to be highly dependent; the dependence between the samples may be thought of as adversarial.

We provide an easily implementable, online clustering algorithm that as we show, is asymptotically consistent given that the marginal distribution of each sample is stationary ergodic, and that the correct number κ of clusters is provided.

1.3 Methodology

In this section we describe the common ingredients and ideas used in our approach to each of the considered problems.

1.3.1 Distance measure

In both change point estimation and time-series clustering, we need to analyze the process distributions that generate the data based on the provided samples. In change point estimation we wish to determine the point at which the process distributions generating the samples start to differ. Our clustering objective is to group the samples based on their generating distributions. Therefore, we require an appropriate measure

of distance that reflects the distance between the process distributions that generate a given pair of sequences.

It turns out that an appropriate distance function can be obtained by the consistent estimation of the so-called distributional distance (Gray, 1988). The distributional distance $d(\rho_1, \rho_2)$ between a pair of process distributions ρ_1, ρ_2 is defined as $\sum_{i\in\mathbb{N}} w_i|\rho_1(A_i) - \rho_2(A_i)|$ where, w_i are positive summable real weights, e.g. $w_i = 1/i(i+1)$, and A_i range over a countable field that generates the sigma-algebra of the underlying probability space. For a discrete alphabet A_i range over the set of all possible tuples. For example, in the case of binary alphabets, the distributional distance is the weighted sum of the differences of the probability values (measured with respect to ρ_1 and ρ_2) of all possible tuples $0, 1, 00, 01, 10, 11, 000, 001, \ldots$. In this work we consider real-valued processes so that the sets A_i range over the products of all intervals with rational endpoints (i.e. the intervals, all pairs of such intervals, triples, etc.). The formal definitions are given in Section 2. Asymptotically consistent estimates of this distance can be obtained by replacing unknown probabilities with the corresponding frequencies. As shown by (Ryabko, 2010a), these empirical approximations consistently estimate the distributional distance, provided that the corresponding process distributions are stationary ergodic. This property makes the empirical estimates of the distributional distance suitable for our purposes, allowing us to construct asymptotically consistent algorithms in the stationary ergodic framework. The distributional distance and its empirical estimates have also proved useful in other statistical problems involving stationary ergodic time series (Ryabko and Ryabko, 2010, Ryabko, 2012). From a practical perspective, although the distributional distance involves infinite summations, its empirical approximations can be easily and efficiently calculated (Ryabko, 2010a). In principal we can use other distance functions between the sequences as well, provided that the distance used reflects the distance between the underlying process distributions. In the case of change point analysis, the distance is required to satisfy convexity; a more detailed discussion on other choices for the distance function is provided in Chapter 6.

It is important to note that the distinction between the underlying process distributions is not reflected by string metrics, such as the Hamming distance, or the Levenshtein distance, etc. The Hamming distance between two sequences is defined as the minimum number of substitutions required to transform one sequence into an-

other, and the Levenshtein distance corresponds to the smallest number of deletions, insertions and substitutions needed to achieve the same objective, see e.g. (Stephen, 1994). More generally, a string distance between a pair of sequences is 0 if and only if the two sequences are exactly the same. However, what we require is for the distance to converge to 0 for long enough samples, if and only if both sequences are generated by the same process distribution. Consider a simple example where the elements of the sequences $\boldsymbol{x} := X_1, \ldots, X_n$ and $\boldsymbol{y} := Y_1, \ldots, Y_n$ are drawn independently from a Bernoulli distribution with probability $p := 1/2$. Regardless of the value of n, on average, the Hamming distance between the two sequences is $1/2$ while they are generated by the same process distribution. At the same time, the empirical estimate of the distributional distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ becomes arbitrarily small for large enough n.

1.3.2 Some common ideas behind the proposed methods

When addressing the problems considered in this thesis, it is not possible to directly evaluate the candidate solutions. This is due to the following two reasons. First, the considered problems are unsupervised, and second, the rates of convergence are provably impossible to obtain for stationary ergodic process distributions. This means that performance guarantees are not available for the potential solutions. Therefore, we make no attempt to select a single candidate solution. Our approach consists in having the algorithm produce a set of potential solutions, assigning each a performance weight. The final decision is made either as a weighted combination of the candidate solutions or as a list of solutions sorted based on their assigned performance weights. We design the weights so that they converge to zero on the solutions that are asymptotically incorrect, and to non-zero constants on the asymptotically correct solutions. This property ensures that a weighted combination of the candidate solutions stabilizes on those that are asymptotically correct, or that the top elements of a list of solutions ordered based on performance weight converge to the correct answers.

As an example, let us consider the problem of time series clustering. An asymptotically consistent solution to the offline formulation of the problem (for stationary ergodic time series) has already been proposed by (Ryabko, 2010a). The online formulation introduced in Section 1.2.2 can be carefully reduced to a series of offline clustering problems. However, since new data arrive dynamically, the entire batch of sequences observed at a given time step may potentially contain sequences for which sufficient

information has not yet been collected, and for which the estimates of the distance (with respect to any other sequence) are bound to be misleading. Such "bad" sequences can confuse any (consistent) offline algorithm (e.g. that of (Ryabko, 2010a)), rendering the clustering procedure useless in an online setting. Therefore, the naive approach of applying an offline algorithm to the entire data observed at every time step is bound to fail. Another approach would be to cluster a fixed selected set of (reference) samples, using a consistent offline method, and assign the remaining samples to the nearest cluster. However, this procedure would be asymptotically consistent only if the selected reference set contains at least one sequence sampled from each and every one of the κ process distributions. In the considered formulation, it is not possible to directly identify such appropriate reference set. Our solution is based on a weighted combination of several clusterings, each obtained by running a consistent offline algorithm on different portions data. To this end we use two sets of weights: the first is to penalize for large batches, reducing the effect of "bad" sequences in the algorithm's decision; the second is calculated as the minimum inter-cluster distances between the candidate cluster centers obtained at every iteration of running the offline algorithm. This way, if a batch does not contain samples from all κ process distributions, its corresponding output will be assigned a performance weight that from some time on converges to zero. Therefore, from some time on the online clustering algorithm will base its decisions on the offline clustering of appropriate batches. This approach is described in Chapter 4. Similarly, we reduce the multiple change point estimation problem to a series of single change point problems and weight the sets of candidate estimates obtained sequentially. See Chapter 3 for detailed descriptions of our change point estimation algorithms.

1.4 Related work

In this section, we start our review of related literature with a non-exhaustive list of other general approaches to statistical learning problems involving time series. Next, we present the related work on the concrete problems addressed in this thesis, namely, change point estimation and time series clustering.

1.4.1 Some related results on stationary ergodic time series

Even though sequential analysis has a vast literature, consistent non-parametric methods to address inference problems for highly dependent time series are rather scarce. In particular, apart from the results of (Ryabko, 2010a, Ryabko and Ryabko, 2010) discussed in the next section, unsupervised learning for time series has not been considered in this framework before. An incomplete list of some results on statistical learning problems concerning stationary ergodic time series (other than the problems addressed in this thesis) include sequence prediction (Ryabko, 1988, Morvai et al., 1996, 1997a, Algoet, 1992, Ryabko, 2010c), hypothesis testing (Csiszar and Shields, 2004, Nobel, 2006, Ryabko, 2012, Ryabko and Ryabko, 2010), and classification and regression (prediction with side information) (Algoet, 1999, Morvai et al., 1997b, Adams and Nobel, 2012). Moreover, many natural problems turn out to be impossible to solve in the stationary ergodic framework. For example, as mentioned previously (Ryabko, 2010b) proves the impossibility of homogeneity testing; (Adams and Nobel, 1998) show that no procedure can consistently estimate the one-dimensional marginal density of every stationary ergodic process for which such a density exists; in the context of sequence prediction, it has been shown that asymptotically accurate prediction at every step is impossible (Ryabko, 1988).

1.4.2 General non-stochastic approaches

A different approach to obtain general results is to make absolutely no statistical assumptions about the data. Instead, the data is viewed simply as an arbitrary deterministic sequence. The algorithm is a sequential decision maker whose performance is measured with respect to a given set of reference methods called experts. Thus, instead of making stronger assumptions on the process distributions that generate the data, assumptions are made on the comparison class of methods. In a typical formulation, the set of experts is finite or countably infinite. At time step t the algorithm makes a decision as a weighted average of the experts' decisions, and its output is compared against the true outcome to calculate a *loss*. Each expert also incurs a loss in the same way and is assigned a performance weight calculated as the exponential decay of its cumulative loss up to time t. Thus, the experts with smaller cumulative loss are weighted more. This results in keeping as small as possible the algorithm's so-called

cumulative regret with respect to each expert, measured as the difference between the its cumulative loss, and that of the expert's. To the best of our knowledge, the problems of change point estimation and time series clustering have not been studied under the general paradigm of expert advice. This approach is used in sequential prediction and bandit problems. The reader is referred to the monograph of (Cesa-Bianchi and Lugosi, 2006) for a comprehensive overview. It is worth noting that clustering is often considered as a purely combinatorial problem, and the data is not assumed to be generated by some probability distribution. Specifically, a similarity measure is usually fixed and a particular combinatorial objective, for example k-means, is optimized. Thus, the common approach to clustering is only loosely related to the problems considered in this thesis; see more discussion in Section 1.2.2.

Interestingly, our methods bear some similarity to those based on experts advice. As discussed earlier (in Section 1.3) in order to arrive at the final solution (e.g. estimation, clustering, etc.) we use decision strategies that rely on weights. That is, we maintain an exhaustive list of potential solutions, assigning each solution a performance weight. We produce a weighted combination of the candidate solutions or a list sorted in decreasing order of performance weight at the output. However, an important difference is that the quality of potential solutions cannot be evaluated directly in our formulations. Therefore, we cannot select a single candidate solution based on performance. As discussed in Section 1.3, we use appropriate weights that asymptotically reflect the performance of each potential solution. The weights are designed to converge to zero on the asymptotically incorrect decisions, and to non-zero constants on the candidate solutions that are asymptotically correct, indirectly reflecting the quality of candidate solutions in asymptotic. The similarity between the two approaches is not surprising, as solutions based on weighted combinations arise in many learning and statistical problems. The distributional distance between the process distributions discussed in Section 1.3.1 is already an example of such approach: the distance is calculated as a weighted sum of the differences in probabilities of sets in a collection that generates the Borel sigma algebra. Other examples of such weighted-based methods that concern sequential prediction literature, including probabilistic approaches of (Solomonoff, 1978, Ryabko, 1988, 2011).

1.4.3 Work on change point analysis

Change point analysis is a classical problem in statistics, with a vast literature in both parametric and non-parametric settings (Basseville and Nikiforov, 1993, Brodsky and Darkhovsky, 1993, 2000, Müller and Siegmund, 1994, Csörgö and Horváth, 1998, Chen, 2012). A typical parametric formulation of the problem involves the estimation of a single change in the mean, while the data are assumed to be independently and identically distributed in each segment, and the process distributions come from specific known families, (e.g. Gaussian). More general settings have also been considered. However, even in the case of stationary ergodic time series, change point estimation has rarely been addressed to the same extent of generality as that considered in this thesis. In this section we provide a non-exhaustive review of some related non-parametric approaches.

Most non-parametric approaches to estimating a single change point in a given sequence $\boldsymbol{x} := X_1, \ldots, X_n$ are usually based on the following idea. Initially, every possible index $i \in \{1, \ldots, n\}$ is considered a potential change point. The difference between the empirical expectation of the two segments $X_1, \ldots X_i$ and $X_{i+1}, \ldots, X_n$ on either side of every fixed $i \in \{1, \ldots, n\}$ is calculated, and the change point estimate is chosen as the index that maximizes this difference in absolute value. A more general approach is based on maximizing the difference between the empirical distributions of the two segments under a given norm. Different norms give rise to different test statistics. The commonly used distances include the Kolmogorov-Smirnov statistics, obtained when the difference is calculated under the L_∞ norm, the Cramér-von Mises statistics corresponding to the use of L_2 norm, and the generalizations thereof.

Change point estimation for independent sequences has been widely studied (see the references above for comprehensive reviews). The problem of single change point estimation has also been considered under more general non-parametric assumptions. A body of work described by (Brodsky and Darkhovsky, 1993) involves non-parametric change in the mean estimators for stationary ergodic process distributions satisfying strong mixing conditions; their single-dimensional marginals are different. More recent results on both online and offline single change point analysis for observations satisfying mixing conditions include (Kokoszka and Leipus, 2002, Hariz et al., 2007, Brodsky and Darkhovsky, 2008, Papantoni-Kazakos and Burrell, 2010).

An argument by (Brodsky and Darkhovsky, 1993) suggests that estimating changes

in the general multidimensional distribution of a random sequence can be conveniently reduced to locating changes in the mean of single dimensional distributions. The argument is as follows. Consider the case where the sequence $\boldsymbol{x} \in \mathbb{R}^n$ has a single change point. If there is a change in the general distribution, then there is some $m \in \mathbb{N}$ such that the m-dimensional marginals before and after the change are different. We can partition the space $\mathbb{R}^m$ into a set of finite non-overlapping subsets $A_1, A_2, \ldots, A_l$ for some finite integer $l \in \mathbb{N}$, and obtain the empirical estimate of each of the m-dimensional distributions. Since the distributions are different, for appropriate l and set $A_j,\ j = 1..l$, the empirical distributions are also different, and this difference can be made arbitrarily close to the difference between the distributions. We can define l^m indicator sequences reflecting all possible crossings of $\boldsymbol{x}$ with the sets $A_j,\ j = 1..l$. The probability of each set A_j can be estimated as the expected value of these indicator sequences. Since the distributions are different, at least one of the indicator sequences must have a change in the mean. Therefore, the problem of change in the distribution can be reduced to that in the mean of one of the l^m indicator sequences. However, this argument is helpful only if the distinguishing sets, or the parameters m and l are known in advance. For the case where this information is not available, they propose a recurrent selection procedure for different values of m, where the procedure is to be stopped if a priori requirements on the quality of detection are satisfied. However, such requirements, (e.g. rates of convergence) are not available for the general case of stationary ergodic process distributions considered in this thesis. Moreover, even when the process distributions satisfy stronger assumptions (e.g. mixing conditions) and can have rates of convergence, due care must be taken when combining estimates from different dimensions m, and quantization levels l, since the direct combination of estimates obtained for all values of m and l would result in uncontrollable estimation errors.

More general notions of change have also been considered. However, the assumption that the single-dimensional marginals are different is prevalent in the literature. For independent observations, (Carlstein, 1988, Dumbgen, 1991) proposed consistent methods to estimate a change in the single-dimensional distribution of a given time series, and determined their rates of convergence. In a more general setting, (Giraitis et al., 1995) proposed a Kolmogorov-Smirnov type statistic to estimate a change in the single-dimensional marginals of dependent (not necessarily mixing) observations. Their method relies on known asymptotic rates as well as the limiting distribution of

the statistic used to estimate the change point. Among the set of assumptions usually made in the literature, perhaps the framework considered by (Carlstein and Lele, 1993) is closest to our statistical setting. They proposed a consistent change point estimator for general stationary ergodic process distributions. Their method is based on a class of the so-called "mean-dominant" norms, which includes both the Kolmogorov-Smirnov and the Cramér-von Mises statistics. However, unlike the formulation considered in this thesis, they consider the problem of single change point estimation and assume that the one-dimensional marginals before and after the change are different.

The objective of estimating a change in the distribution of stationary ergodic process time series has recently been considered by (Ryabko and Ryabko, 2010). They proposed a method to estimate a single change point in the distribution of observations, and proved it to be asymptotically consistent provided that the process distributions that generate the data are stationary ergodic. In order to consistently estimate the change point, they need not rely on rates of convergence or on any such a priori requirements on performance. The change point estimate is obtained as the maximizer of the empirical estimate of the distributional distance discussed in Section 1.3.1. By this approach, the problem of selecting the appropriate distinguishing sets discussed above is solved using a weighted sum of the differences in all m-dimensional marginals, and all quantization levels l, penalizing for higher values of m and l. Thus, in order for consistency to hold, it is not required for the finite-dimensional marginals of any given fixed size before and after the change to be different. We generalize the work of (Ryabko and Ryabko, 2010) to the case of multiple change points. This extension turned out to be much more complex.

The problem of multiple change point estimation is not as widely explored as that concerning single change point analysis. An approach proposed by (Vostrikova, 1981) is based on the recursive application of a single-change point estimator to the segments formed by breaking the sequence at the change point is estimated at the previous iteration. The multiple change point problem has also been addressed from a global optimization perspective. Both parametric and non-parametric approaches have been considered, and the formulations concern the change in the single-dimensional distributions over independent samples (Yao, 1988, Lebarbier, 2005, Vert and Bleakley, 2010, Lung-Yut-Fong et al., 2011, Lévy-Leduc and Roueff, 2009) as well as over dependent observations satisfying mixing conditions (Lavielle, 1999, Lavielle and Teyssiere, 2007).

In particular the latter approaches are based on minimizing a so-called contrast function. The form of the contrast function depends on the specific family of distributions considered or on the specific form of change sought. The problem for the unknown number of change points is addressed by adding a penalization term to the function that has to be optimized. In this framework the problem of finding the change in the structure of dependence, that is not in the finite dimensional marginals, but in the time series distribution has also been addressed . However this only concerns parametric settings, specifically changes in the spectrum (Lavielle, 2005) and the so-called long memory parameter in a certain parametric family (Kokoszka and Leipus, 2002).

1.4.4 Work on clustering

Clustering is a well-known problem in learning theory, and has been extensively studied in the literature. Unlike in the case of time series clustering, as follows from the results of (Kleinberg, 2002) the mere notion of *good clustering* is difficult to formally define in the general case. The main challenge with this problem is that, its intuitive objective is hard to properly formalize. Thus, a common approach to clustering is to fix a distance (or similarity) measure, and consider a specific combinatorial objective to optimize, such as k-means, see for example, (Jain, 2010) and references therein. It is worth noting that some combinatorial objectives that are considered to be natural, turn out to be computationally difficult, see for example (Mahajan et al., 2009).

As discussed earlier, our focus in this thesis is on a sub-problem of clustering, where the data are samples generated by discrete-time stochastic processes. For this particular clustering problem, a natural notion of consistency is proposed by (Ryabko, 2010a), which is further shown to be achievable under the only assumption that the process distributions generating the data are stationary ergodic. Thus, as mentioned in Section 1.2.2, the correct ground-truth readily exists as a natural part of this version of the problem. The proposed notion of consistency requires that the sequences generated by the same process distribution be grouped together, and can be achieved in asymptotic as the length of the individual sequences tend to infinity. Since the regime of asymptotic is with respect to the sequence lengths, and not with respect to the number of sequences to be clustered, consistency can be achieved even if the number of observed sequences is finite. Hence, this particular clustering problem is not affected by the impossibility result of (Kleinberg, 2002). We extend this notion of consistency to the online setting,

where as discussed in Section 1.2.2, we call an algorithm asymptotically consistent if its output clustering confined to every fixed batch of sequences has the property that from some time on those and only those sequences that are generated by the same process distribution are in the same group. Note that as discussed in Section 1.2.2, the method of (Ryabko, 2010a) which is designed for the offline formulation cannot be directly applied as a solution to the online setting considered in this work.

The clustering problem formulation considered in this thesis bears some similarity to the work of (M. et al., 2008, Balcan and Gupta, 2010). While they consider the combinatorial version of the problem with a fixed similarity measure, in their setting as well as in ours an unknown underlying ground-truth clustering exists. Under this assumption, (M. et al., 2008, Balcan and Gupta, 2010) provide a sufficient set of properties that make a similarity function useful for clustering. One such property is the so-called "strict separation" which requires that for a fixed similarity measure, the points within the same cluster are more similar to each other, than to the points in the other clusters. They show that if a similarity measure satisfies the strict separation property, it is possible to produce a hierarchical clustering tree such that the ground-truth clustering is a pruning of the tree. They also provide stability conditions for the case where this property is satisfied for all but a subset of the data points. In our formulation, every fixed batch of sequences, from some time on satisfies the strict-separation property (when the empirical distributional distance is used as the distance between the samples). Moreover, our online algorithm must be stable with respect to the newly received samples for which sufficient information has not been collected yet, and as such may not result in consistent estimates of the distance between their generating distributions, and other distributions.

Some probabilistic formulations of the time series clustering problem may be related to our formulation. Perhaps the closest would be mixture models where observations are assumed to be samples from a finite mixture of probability distributions (Smyth, 1997, Dasgupta, 1999, Biernacki et al., 2000, Cadez et al., 2000, Li and Biswas, 2002, Panuccio et al., 2002, Kumar et al., 2002, Shi and Joydeep, 2003, Achlioptas and McSherry, 2005, Bouguila and Ziou, 2006, McCullagh and Yang, 2008). In these problems, each sequence is assumed to have been generated independently according to one of k distributions. The model of the data is well-specified a priori, that is the distributions are assumed to have known forms, e.g. Gaussians, Dirichlet or hidden Markov models (HMMs).

As such, the data may be clustered using likelihood-based distances (along with for example, the k-means algorithm), or Bayesian inference. Another typical approach is to estimate the parameters of the distributions in the mixture rather than actually clustering the data points. Clearly, the main difference from our setting is in that we do not assume any known model of the data. We do not even assume independence between the different sequences.

To the best of our knowledge we are the first to consider the online clustering of highly dependent time series from the perspective of asymptotic consistency.

1.5 Summary of main contributions

We have shown that under minimal statistical assumptions, consistent, easily implementable solutions exist for some classical unsupervised learning problems. More specifically, we summarize the contributions of this thesis as follows.

- **Change point estimation.** Three formulations of the change point estimation problem are proposed, which are constructively shown to admit asymptotically consistent solutions in the framework described. The common assumption in all these problems is that each sequence is generated by stationary ergodic process distributions.

  1. **Estimating the change points (when the number κ of change points is known).** A change point estimation algorithm is proposed that is shown to be asymptotically consistent, provided that the correct number of change points is given. This means that the difference between each change point and its estimate is arbitrarily small for a long enough sequence.
  2. **Estimating the change points when the number κ of change points is unknown and the algorithm does not attempt to find it.** A so-called "list-estimator" is proposed that generates a (sorted) list of change point estimates. The first κ elements of this list converge to the true change points, while κ is unknown.
  3. **Finding the number of change points and locating the changes.** An algorithm is proposed that given the correct number of process distributions that generate the data, finds the correct number of change points, provided that the

sequence is long enough. In addition, asymptotically consistent change point estimates are provided.

- **Online time series clustering.** An efficient algorithm is proposed, that is asymptotically consistent in the online setting provided that the marginal distribution of each sample is stationary ergodic, and the correct number of clusters is known. This means that for every batch of sequences the output of the clustering algorithm restricted to this batch, from some time on groups together those and only sequences that are generated by the same process distribution.

- **Experimental evaluations.** The main contribution of this thesis is theoretical. However, we also provide some experimental evaluations of our algorithms. In order for the experimental setup to reflect the generality of our framework, we generated the data by time series distributions that, while being stationary ergodic, do not belong to any "simpler" general class of time series, and are difficult to approximate by finite-state models. The considered processes are classical examples in the literature on ergodic time series (Billingsley, 1961) . In particular, they are used by (Shields, 1996) as an example of a class of stationary ergodic processes that are not B-processes. Such time series cannot be modelled by a hidden Markov model by a finite or countably infinite set of states. Moreover, k-order Markov or hidden Markov approximations of this process do not converge to it in $\bar{d}$ distance, a distance that is stronger than d, and whose empirical approximations are often used to study general (non-Markovian) processes. The single-dimensional marginals of the different distributions generated through this procedure are the same. To the best of our knowledge we are the first to use this general class of time series in experimental evaluations. Moreover, at least in the case of change point analysis, and time series clustering, no consistent non-parametric method exists for this experimental setup. To demonstrate the applicability of this general statistical framework to real data, we have tested the offline clustering algorithm of (Ryabko, 2010a), which was not implemented before, on motion-capture sequences of human locomotion. The clustering approach of (Ryabko, 2010a) achieved better performance as compared to the state of the art.

1.6 Organization

The remainder of this thesis is organized as follows. We provide some preliminary notations and definitions in Chapter 2. This includes definitions and properties of the distributional distance, as well as a brief overview of the impossibility result of (Ryabko, 2010b) which plays an important role in our search for solvable problem formulations in the stationary ergodic framework. In Chapter 3 we provide our main theoretical results on multiple change point estimation. In this chapter we give provide three formulations of the problem and constructively show that they admit consistent solutions in the stationary ergodic paradigm. Chapter 4 addresses the problem of time series clustering, where we provide an algorithm for the online version of the problem, and prove its asymptotic consistency in the general framework introduced above. In Chapter 5, we provide some experimental results. We conclude the thesis in Chapter 6, providing some closing remarks on open problems and future directions.

Chapter 2

Preliminaries

2.1 Notations and definitions

Let $\mathcal{X}$ be a measurable space (the domain); in this work we let $\mathcal{X} = \mathbb{R}$ but extensions to more general spaces are straightforward. For a sequence $X_1, \ldots, X_n$ we use the abbreviation $X_{1..n}$. Consider the Borel σ-algebra $\mathcal{B}$ on $\mathcal{X}^\infty$ generated by the cylinders

$$\{B \times \mathcal{X}^\infty : B \in B^{m,l}, m, l \in \mathbb{N}\}$$

where the sets $B^{m,l}, m, l \in \mathbb{N}$ are obtained via the partitioning of $\mathcal{X}^m$ into cubes of dimension m and volume 2^{-ml} (starting at the origin). Let also $B^m := \cup_{l\in\mathbb{N}} B^{m,l}$. Process distributions are probability measures on the space $(\mathcal{X}^\infty, \mathcal{B})$. For $\boldsymbol{x} = X_{1..n} \in \mathcal{X}^n$ and $B \in B^m$ let $\nu(\boldsymbol{x}, B)$ denote the *frequency* with which $\boldsymbol{x}$ falls in B, i.e.

$$\nu(\boldsymbol{x}, B) := \frac{\mathbb{I}\{n \geqslant m\}}{n - m + 1} \sum_{i=1}^{n-m+1} \mathbb{I}\{X_{i..i+m-1} \in B\} \tag{2.1}$$

A process ρ is *stationary* if for any $i, j \in 1..n$ and $B \in B^m,\ m \in \mathbb{N}$, we have

$$\rho(X_{1..j} \in B) = \rho(X_{i..i+j-1} \in B).$$

A stationary process ρ is called *stationary ergodic* if for all $B \in \mathcal{B}$ with probability 1 we have

$$\lim_{n\to\infty} \nu(X_{1..n}, B) = \rho(B).$$

By virtue of the ergodic theorem (see, e.g. (Billingsley, 1965)), this definition can be shown to be equivalent to the standard definition for the stationary ergodic processes (every shift-invariant set has measure 0 or 1; see, e.g. (Csiszar and Shields, 2004)).

We can define distributions over the space $((\mathcal{X}^\infty)^\infty, \mathcal{B}_2)$ of infinite matrices with the Borel σ-algebra $\mathcal{B}_2$ generated by the cylinders

$$\{(\mathcal{X}^\infty)^k \times (B \times \mathcal{X}^\infty) \times (\mathcal{X}^\infty)^\infty : \ B \in B^{m,l}, k, m, l \in \mathbb{N}\}.$$

More specifically, consider the matrix $\mathbf{X} \in (\mathcal{X}^\infty)^\infty$ of random variables

$$\mathbf{X} := \begin{bmatrix} X_1^{(1)} & X_2^{(1)} & X_3^{(1)} & X_4^{(1)} & \dots \\ X_1^{(2)} & X_2^{(2)} & X_3^{(2)} & \dots & \dots \\ \vdots & \vdots & \dots & \dots & \vdots \\ X_1^{(N)} & X_2^{(N)} & \dots & \ddots & \ddots \\ \vdots & \vdots & \dots & \ddots & \ddots \end{bmatrix} \in (\mathcal{X}^\infty)^\infty \tag{2.2}$$

generated by some (unknown) arbitrary probability distribution ρ on $((\mathcal{X}^\infty)^\infty, \mathcal{B}_2)$. The matrix $\mathbf{X}$ corresponds to infinitely many one-way infinite sequences, each of which is generated by an *unknown* stationary ergodic distribution. Assume that the marginal distribution of ρ on each row of $\mathbf{X}$ is an unknown stationary ergodic process. Note that the requirements are only on the marginal distributions over the rows; the distribution ρ is otherwise completely arbitrary. This means that the samples in $\mathbf{X}$ are allowed to be dependent, and the dependence can be arbitrary; one can even think of the dependence between samples as *adversarial*.

2.2 The Distributional distance and its empirical estimates

In our methods, we require a means to discriminate between sequences based on the process distributions that generate them. To this end, we use empirical estimates of the so-called distributional distance, introduced in this section.

Definition 2.2.1 (Distributional Distance)**.** *The distributional distance between a pair*

of process distributions ρ_1, ρ_2 is defined as follows

$$d(\rho_1, \rho_2) = \sum_{m,l \in \mathbb{N}} w_m w_l \sum_{B \in B^{m,l}} |\rho_1(B) - \rho_2(B)|$$

where, we let $w_i := 1/i(i+1)$, $i \in \mathbb{N}$, but any summable sequence of positive weights may be used.

For example, consider finite-alphabet processes with the binary alphabet $\mathcal{X} = \{0, 1\}$. the distributional distance in this case is the weighted sum of the differences of the probability values (calculated with respect to ρ_1 and ρ_2) of all possible tuples, $0, 1, 00, 01, 10, 11, 000, 001, \ldots$, where, smaller weights are given to longer patterns. For the more general alphabets, this involves partitioning the sets $\mathcal{X}^m$, $m \in \mathbb{N}$ into cubes of decreasing volume (indexed by l) and then taking a sum over the differences in probabilities of all the cubes in these partitions. The differences in probabilities are weighted: smaller weights are given to larger m and finer partitions.

Remark 2.2.2. *The distance is more generally defined as $d(\rho_1, \rho_2) := \sum_{i \in \mathbb{N}} w_i |\rho_1(A_i) - \rho_2(A_i)|$ where A_i range over a countable field that generates the σ-algebra of the underlying probability space (Gray, 1988). Definition 2.2.1 above allows us to have a concrete choice of the sets B. Moreover, this choice makes the algorithm easily implementable. The individual cubes in $B^{m,l}$ get the same weight rather than having decreasing weights according to their index in the sequence. While this is irrelevant for the consistency results, this means that the practical implementation can take less data to converge. The same definition was used in (Ryabko, 2010a).*

We use *empirical estimates* of this distance defined as follows.

Definition 2.2.3 (Empirical estimates of $d(\cdot, \cdot)$)**.** *The empirical estimate of the distributional distance between a sequence $\boldsymbol{x} = X_{1..n} \in \mathcal{X}^n, n \in \mathbb{N}$ and a process distribution ρ is given by*

$$\widehat{d}(\boldsymbol{x}, \rho) := \sum_{m=1}^{m_n} \sum_{l=1}^{l_n} w_m w_l \sum_{B \in B^{m,l}} |\nu(\boldsymbol{x}, B) - \rho(B)| \tag{2.3}$$

and that between a pair of sequences $\boldsymbol{x}_i \in \mathcal{X}^{n_i}$ $n_i \in \mathbb{N}$, $i = 1, 2$. is defined as

$$\widehat{d}(\boldsymbol{x}_1, \boldsymbol{x}_2) := \sum_{m=1}^{m_n} \sum_{l=1}^{l_n} w_m w_l \sum_{B \in B^{m,l}} |\nu(\boldsymbol{x}_1, B) - \nu(\boldsymbol{x}_2, B)| \tag{2.4}$$

where, m_n and l_n are any sequences of integers that go to infinity with n.

Proposition 2.2.4 ($\widehat{d}(\cdot,\cdot)$ is asymptotically consistent (Ryabko, 2010a))**.** *For every pair of sequences $\boldsymbol{x}_1 \in \mathcal{X}^{n_1}$ and $\boldsymbol{x}_2 \in \mathcal{X}^{n_2}$ with joint distribution ρ and stationary ergodic marginals $\rho_i,\ i = 1,2$ we have*

$$\lim_{n_i\to\infty} \widehat{d}(\boldsymbol{x}_i, \rho_j) = d(\rho_i, \rho_j),\ i,j \in 1,2,\ \rho - a.s.,\ \text{and} \tag{2.5}$$

$$\lim_{n_1,n_2\to\infty} \widehat{d}(\boldsymbol{x}_1, \boldsymbol{x}_2) = d(\rho_1, \rho_2),\ \rho - a.s. \tag{2.6}$$

Remark 2.2.5. *The empirical estimates of the distributional distance are symmetric. Moreover, the triangle inequality holds for the distributional distance $d(\cdot,\cdot)$ and its empirical estimates $\widehat{d}(\cdot,\cdot)$ so that for all distributions $\rho_i,\ i = 1..3$ and all sequences $\boldsymbol{x}_i \in \mathcal{X}^{n_i}\ n_i \in \mathbb{N},\ i = 1..3$ we have,*

$$\begin{aligned} d(\rho_1,\rho_2) &\leqslant d(\rho_1,\rho_3) + d(\rho_2,\rho_3) \\ \widehat{d}(\boldsymbol{x}_1,\boldsymbol{x}_2) &\leqslant \widehat{d}(\boldsymbol{x}_1,\boldsymbol{x}_3) + \widehat{d}(\boldsymbol{x}_2,\boldsymbol{x}_3) \\ \widehat{d}(\boldsymbol{x}_1,\rho_1) &\leqslant \widehat{d}(\boldsymbol{x}_1,\rho_2) + d(\rho_1,\rho_2). \end{aligned}$$

Remark 2.2.6. *The distributional distance $d(\cdot,\cdot)$ and its empirical estimates $\widehat{d}(\cdot,\cdot)$ are convex functions; that is for every $\lambda \in (0,1)$ for all distributions $\rho,\ \rho_i,\ i = 1..3$ and all sequences $\boldsymbol{x}_i \in \mathcal{X}^{n_i}$ with $n_i \in \mathbb{N}, i = 1..3$ we have*

$$\begin{aligned} d(\rho_1, \lambda\rho_2 + (1-\lambda)\rho_3) &\leqslant \lambda d(\rho_1,\rho_2) + (1-\lambda) d(\rho_1,\rho_3) \\ \widehat{d}(\boldsymbol{x}_1, \lambda\boldsymbol{x}_2 + (1-\lambda)\boldsymbol{x}_3) &\leqslant \lambda \widehat{d}(\boldsymbol{x}_1,\boldsymbol{x}_2) + (1-\lambda)\widehat{d}(\boldsymbol{x}_1,\boldsymbol{x}_3) \\ \widehat{d}(\rho, \lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2) &\leqslant \lambda \widehat{d}(\rho,\boldsymbol{x}_1) + (1-\lambda)\widehat{d}(\rho,\boldsymbol{x}_2) \end{aligned}$$

The following proposition due to (Ryabko, 2010a) shows that despite the infinite summations, $\widehat{d}$ can be calculated in polynomial time.

Proposition 2.2.7 (Calculating $\widehat{d}$ given by (2.4) (Ryabko, 2010a))**.** *Consider a pair of sequences $\boldsymbol{x}_1 = X^1_{1..n_1}$ and $\boldsymbol{x}_2 = X^2_{1..n_2}$ and let $n := \max(n_1, n_2)$. The computational complexity (time and space) of calculating the empirical distributional distance $\widehat{d}(\boldsymbol{x}_1, \boldsymbol{x}_2)$*

is $\mathcal{O}(nm_n \log m_n \log s^{-1})$, *where*

$$s := \min_{\substack{X_i^1 \neq X_j^2 \\ i=1..n_1, j=1..n_2}} |X_i^1 - X_j^2|. \tag{2.7}$$

Proof. First, observe that for fixed m and l, the sum

$$T^{m,l} := \sum_{B \in B^{m,l}} |\nu(X_{1..n_1}^1, B) - \nu(X_{1..n_2}^2, B)| \tag{2.8}$$

has not more than $n_1 + n_2 - 2m + 2$ non-zero terms (assuming $m \leq n_1, n_2$; the other case is obvious). Indeed, there are $n_i - m + 1$ tuples of size m in each sequence $\boldsymbol{x}_i$, $i = 1, 2$ namely, $X_{1..m}^i, X_{2..m+1}^i, \ldots, X_{n_1-m+1..n_1}^i$. Therefore, $T^{m,l}$ can be obtained by a finite number of calculations. Furthermore, observe that $T^{m,l} = 0$ for all $m > n$ and for each m, for all $l > \log s^{-1}$ the term $T^{m,l}$ is constant. That is for each fixed m we have,

$$\sum_{l=1}^{\infty} w_m w_l T^{m,l} = w_m w_{\log s^{-1}} T^{m,\log s^{-1}} + \sum_{l=1}^{\log s^{-1}} w_m w_l T^{m,l}$$

so that we simply double the weight of the last non-zero term. (Note also that s is bounded above by the length of the binary precision in representing the random variables X_j^i.) Thus, even with $m_n, l_n \equiv \infty$ we can calculate $\widehat{d}$ precisely.

Moreover, for a fixed $m \in 1..\log n$ and $l \in 1..\log s^{-1}$ for every sequence $\boldsymbol{x}_i$, $i = 1, 2$ the frequencies $\nu(\boldsymbol{x}_i, B)$, $B \in B^{m,l}$ may be calculated using suffix trees with $\mathcal{O}(n)$ worst case construction and search complexity (see e.g. (Ukkonen, 1995)). The construction of the suffix tree is $\mathcal{O}(n)$, and searching all $z := n - m + 1$ occurrences of patterns of length m entails $\mathcal{O}(m + z) = \mathcal{O}(n)$ complexity. This brings the overall computational complexity of (2.4) to $\mathcal{O}(nm_n \log s^{-1})$. □

The following consideration can be used to set m_n.

Remark 2.2.8. *For a fixed* l *the frequencies* $\nu(\boldsymbol{x}_i, B)$, $i = 1, 2$ *of cells in* $B \in B^{m,l}$ *corresponding to values of* $m > \log_n$ *are not consistent estimates of their probabilities (and thus only add to the error of the estimate). More specifically for a pattern* $X_{j..j+m}$ *with* $j = 1..n - m$ *of length* m *the probability* $\rho_i(X_{j..j+m} \in B)$, $i = 1, 2$ *is of order* 2^{-mh_i}, $i = 1, 2$ *where* h_i *denotes the entropy rate of* ρ_i, $i = 1, 2$. *Therefore, the*

frequencies of patterns of length $m > \log n$ *in* $\boldsymbol{x}_i$, $i = 1, 2$ *are not consistent estimates of their probabilities. By the above argument, we can set* $m_n := \log n$.

2.3 Some theoretical impossibility results in the stationary ergodic framework

As discussed in Chapter 1, the main motivation for this work is to construct consistent algorithms, in the stationary ergodic framework. Many important impossibility results exist for this general framework. In general, rates of convergence (even of frequencies to respective probabilities) are impossible to obtain for stationary ergodic process distributions (see, for example, (Shields, 1996)). As a result, it is provably impossible to distinguish between stationary ergodic process distributions. This has recently been shown by (Ryabko, 2010b). The implications of this impossibility result are key in our analysis of the possibilities and limitations of the statistical methods for the unsupervised learning problems that we consider. For completeness, we repeat the statement in this section.

A discrimination procedure is defined as a family of mappings $D_n : \mathcal{X}^n \times \mathcal{X}^n \to \{0, 1\}$, $n \in \mathbb{N}$, that given a pair of samples $\boldsymbol{x} \in \mathcal{X}^n$ and $\boldsymbol{y} \in \mathcal{X}^n$ produces a binary answer where, $D_n(\boldsymbol{x}, \boldsymbol{y}) = 0$ means that $\boldsymbol{x}$ and $\boldsymbol{y}$ are generated by the same process distribution, and $D_n(\boldsymbol{x}, \boldsymbol{y}) = 1$ means that they are generated by different distributions. A discrimination procedure is *asymptotically consistent* for a class $\mathcal{C}$ of process distributions, if for any pair of processes ρ, $\rho' \in \mathcal{C}$ that independently generate samples $\boldsymbol{x} \in \mathcal{X}^n$ and $\boldsymbol{y} \in \mathcal{X}^n$ respectively, the expected output of $D_n(\boldsymbol{x}, \boldsymbol{y})$ converges to 0 if and only if, $\rho = \rho'$, for large enough n. That is, if the limit $\lim_{n\to\infty} \mathbb{E} D_n(\boldsymbol{x}, \boldsymbol{y})$ exists, and equals $\mathbb{I}\{\rho = \rho'\}$.

Theorem 2.3.1 (Ryabko (2010b))**.** *There is no asymptotically correct discrimination procedure for the set of all stationary ergodic processes.*

In fact, the impossibility theorem is shown for the class of B-processes (or Bernoulli processes), which is a subset of the stationary ergodic process, and includes the set of all k-order Markov processes and their functions; for more on B-processes the reader is referred to (Ornstein, 1974). The proof is by contradiction, where it is assumed that

a consistent discrimination procedure D exists, but a B-process is constructed and is shown to consistently confuse D, so that $\mathbb{E}D_n$ diverges.

The implications of Theoreom 2.3.1 in our work are as follows.

1. **It is impossible to distinguish between the cases of 0 and 1 change point in a stationary ergodic sequence.** Assume that we are given two sequences $\boldsymbol{x} := X_1, \ldots, X_{n_x}$ and $\boldsymbol{y} := Y_1, \ldots, Y_{n_y}$ generated by stationary ergodic process distributions where, it is not known whether or not they are generated by the same process distribution. Let $\boldsymbol{z} := Z_1, \ldots, Z_{n_x+n_y}$ be formed by concatenating $\boldsymbol{x}$ and $\boldsymbol{y}$ so that $Z_i := X_i$ for $i = 1..n_x$ and $Z_{n_x+i} := Y_i$ for $i = 1..n_y$. Note that $\boldsymbol{z}$ has a change point (at n_x), if and if they $\boldsymbol{x}$ and $\boldsymbol{y}$ are generated by different process distributions. However, since by Theorem 2.3.1 no consistent test exists to determine the homogeneity of $\boldsymbol{x}$ and $\boldsymbol{y}$, it is impossible to decide whether or not $\boldsymbol{z}$ has a change point. It is easy to see the extension of this argument to the general case for any number of changes. That is, in general it is not possible to determine the number of change points in stationary ergodic time series.

2. **The online detection of a change in the distribution of stationary ergodic time series is impossible.** Consider the online version of the change point problem, where the data arrive sequentially and the objective is to detect whether a change in the distribution of the data has occurred. Thus, at every time-step the algorithm must determine between the cases of 0 and 1 change point in the input sequence. As discussed in Item 1 above, if the process distributions that generate the data are stationary ergodic, by Theorem 2.3.1 this objective is impossible to achieve.

3. **It is impossible to determine the number of clusters in the clustering of stationary ergodic time series.** Suppose that we are given a finite number N of sequences, each generated by one of κ stationary ergodic process distributions, where κ is unknown. We would like to cluster the sequences such that those and only those sequences generated by the same process distribution are grouped together. Hence, the number of target clusters is κ. It is easy to see that Theorem 2.3.1 corresponds to the impossibility of a special case of this problem where $N := 2$ and $\kappa \in \{1, 2\}$. More generally this result implies that it is impossible to find the number of clusters.

Chapter 3

Change point analysis

In this chapter, we consider the problem of locating changes in highly dependent time series. We assume that the data are generated by arbitrary, unknown stationary ergodic process distributions. As a result, there can be any arbitrary form of dependence between the samples. We start with the classical formulation of the problem, and study the possibilities and limitations of change point algorithms under this general framework. We introduce and motivate three different formulations of the problem, and provide non-parametric solutions that as we show, are consistent under this general framework. We make no modeling, independence, mixing or parametric assumptions. We also show that our methods are computationally efficient and can be easily implemented.

Parts of the results presented in this chapter have appeared in the proceedings of Neural Information Processing Systems (NIPS 2012), as well as in those of the 24th International Conference on Algorithmic Learning Theory (ALT 13) and the 31st International Conference on Machine Learning (ICML 14); see (Khaleghi and Ryabko, 2012), (Khaleghi and Ryabko, 2013) [1] *and Khaleghi and Ryabko (2014) respectively.*

Contents

[1] E.M. Gold best student paper

3.1 Introduction

3.1.1 The change point problem

The change point problem can be introduced as follows. A given sequence $\boldsymbol{x} := X_1, \ldots, X_n$ is formed as the concatenation of some (known or unknown) number $\kappa + 1$ of non-overlapping segments so that

$$\boldsymbol{x} = X_1..X_{\lfloor n\theta_1 \rfloor}, X_{\lfloor n\theta_1 \rfloor+1}..X_{\lfloor n\theta_2 \rfloor}, \ldots, X_{\lfloor n\theta_\kappa \rfloor+1}..X_n$$

where $\theta_k \in (0,1),\ k = 1..\kappa$. Each segment is generated by some unknown stochastic process distribution. The process distributions that generate every pair of consecutive segments are different. The index $\lfloor n\theta_k \rfloor$ where one segment ends and another starts

is called a *change point*. The parameters $\theta_k,\ k = 1..\kappa$ that specify the change points $\lfloor n\theta_k \rfloor$ are unknown and the objective is to estimate them.

In this chapter, we consider the change point problem for highly dependent data. As described in Chapter 1, the main theme of this thesis is to make as little assumptions as possible on how the data are generated. To this end, we consider an extremely general framework for the change point problem. Our only assumption is that each segment is generated by one of r (unknown) stationary ergodic process distributions. Apart from this assumption, the joint distribution over the samples can be arbitrary. This is one of the weakest assumptions in statistics. In particular, it means that we make no such assumptions as independence, finite memory or mixing. Moreover, the marginal distributions of any given fixed size (e.g. single-dimensional marginals) before and after a change point are allowed to be the same. Therefore, the means, variances, etc. may be the same throughout the entire sequence: the change refers to that in the time series distribution. The sequences before and after the change points are allowed to be arbitrarily dependent. Moreover, we do not make any conditions on the densities of the marginal distributions (the densities may not exist). Thus, our framework is especially suitable for highly dependent time series, and as such can accommodate a broad range of new applications. The reader is referred to Section 1.2.1 for an overview of other approaches to the change point problem.

Our aim is to estimate the change points consistently. An estimate $\widehat{\theta}_k$ of a change point θ_k is *asymptotically consistent* if it becomes arbitrarily close to θ_k in the limit as the length of the sequence n approaches infinity. We seek change point estimation algorithms that provide an asymptotically consistent estimate for every parameter $\theta_k,\ k = 1..\kappa$. Observe that the asymptotic regime simply means that the estimation error is arbitrarily small if the sequence is sufficiently long. In particular, the problem is offline and the sequence does not grow with time. This differs, for example, from the problem of change point detection. In the latter setting, we do not have the entire sequence at hand, but rather the samples arrive in an online fashion, and the objective is to *detect* a change as soon as possible.

In light of the theoretical impossibility results discussed in Chapter 2, it may seem impossible to consistently estimate the change points in this general framework. Indeed, as discussed in Section 2.3, such problems as the detection of change points (in both online and offline settings), and the estimation of the number of change points without

any additional assumptions cannot admit consistent solutions in this framework. The typical approach is to impose stronger assumptions on the process distributions, so that speeds of convergence can be used in the estimation of the number of change points, or in the detection of changes in an online setting. In this work we take an alternative approach, considering different kinds of additional information that place no restrictions on the time series distributions, but still allow us to construct asymptotically consistent algorithms. This results in three different formulations of the change point problem, corresponding to different kinds of the additional information available.

3.1.2 Three formulations of the change-point problem

We consider three different formulations of the change point estimation problem which depend on the nature of the additional parameters available, and provide non-parametric, asymptotically consistent algorithms for each of the problems. Formal definitions are given in Section 3.2.

1. **Known number κ of change points:** In the first formulation, the correct number κ of change points is known and provided as the only parameter to the algorithm. In this case, we seek a method that simultaneously estimates all κ parameters $\theta_k, k = 1..\kappa$ consistently, so that for large enough n, the error on each of the κ estimates is arbitrarily small. An algorithm that achieves this goal is called asymptotically consistent. With the sequence containing multiple change points, the algorithm is required to simultaneously analyze multiple segments of the input sequence, with no a priori lower bound on their lengths. In this case the main challenge is to ensure that the algorithm is robust with respect to segments of arbitrarily small lengths.

2. **List-estimation:** In the second formulation, we assume that the number κ of change points is unknown. However, in light of the impossibility results discussed above, we make no attempt to estimate κ. Instead, we aim to produce *a list* of at least but possibly more than κ estimates, sorted in such a way that its first κ elements are asymptotically consistent estimates of the unknown parameters $\theta_k, k = 1..\kappa$. We call an algorithm that achieves this goal an asymptotically consistent *"list-estimator."* In this setting, we additionally require an a priori lower bound $\lambda n, \lambda \in (0, 1)$ on the minimum length of the segments, that is, a known λ such that $\lambda \leq \theta_{i+1} - \theta_i$ for all $i = 0..\kappa$.

3. **Finding κ, assuming a known number r of different process distributions:** In the third formulation, we assume that while κ is unknown, an additional parameter is provided, namely, the correct number r of different process distributions that generate the data. We demonstrate that in this case it is possible to estimate κ and to locate the changes consistently. Under this formulation, an algorithm is called asymptotically consistent if it produces a set of $\widehat{\kappa}$ estimates of the parameters θ_k such that for large enough n,

   i. $\widehat{\kappa}$ is equal to κ, and

   ii. the estimation error on all κ estimates is arbitrarily small.

   This means that for large enough n, an asymptotically consistent algorithm not only locates indices that are arbitrarily close to the true change points, but also finds the exact number κ of change points. The latter objective is nontrivial, even under the standard i.i.d. or modeling assumptions, where it is usually addressed with penalized criteria; see, for example, (Yao, 1988, Lavielle, 1999, Lebarbier, 2005, Massart, 2005, Lavielle, 2005). Such criteria necessarily rely on additional parameters upon which the resulting number of change points depend. In a similar manner, an additional parameter, namely the number r of process distributions, is required by our method. However, we would like to emphasize that this parameter has a natural interpretation in many real-world applications. For instance in speech segmentation r may be the total number of speakers. In video surveillance as well as in fraud detection, the change may refer to the point where normal activity becomes abnormal ($r = 2$). The problem of author attribution in a given text written collaboratively by a known number r of authors is also a potential application. Genomics sequence segmentation and in particular the identification of coding versus non-coding regions in DNA sequences is yet another example. In order to find κ in this formulation, we build upon a consistent list-estimator, which in turn requires a lower bound $\lambda n,\ \lambda \in (0, 1)$ on the minimum length of the segments.

3.1.3 Some intuition behind the proposed methods

To motivate the main theme of our methods, let us first consider the case where, the input sequence $\boldsymbol{x} \in \mathcal{X}^n$ is known to have exactly one change point, that is $\kappa := 1$. By definition, in this case the change point divides the sequence into two segments

generated by two different process distributions. On the other hand, we know that the empirical estimate of the distributional distance $\widehat{d}$ between the two segments converges to the distributional distance d between the process distributions that generate them. Therefore, in this case it makes sense to estimate the change point as the index in $1..n$ that identifies two consecutive segments in $\boldsymbol{x}$ that are farthest away in $\widehat{d}$. This intuition was used in the approach by (Ryabko and Ryabko, 2010), as a means to construct a single change point estimator under a similar framework, i.e. where the data are generated by unknown, arbitrary, stationary ergodic process distributions. The case of $\kappa > 1$ turns out to be much more complex.

Ideally, in the general case with multiple change points, we would like to identify a set of subsequences of $\boldsymbol{x}$, each of which contains a single change point. This would reduce the problem to a series of single change point estimation problems and as such, allow us to estimate every change point consistently within an appropriate segment. However, under the formulations that we consider, this set of subsequences is not available in advance. Instead, we partition $\boldsymbol{x}$ in consecutive non-overlapping segments, and use a scoring system to indirectly distinguish between the subsequences of $\boldsymbol{x}$ that contain a change point and those that do not. This is the common ingredient of our methods, which is customized depending on the problem formulation addressed by the corresponding algorithm.

3.2 Problem formulation

We formalize the problem as follows. The sequence

$$\boldsymbol{x} := X_1, \ldots, X_n \in \mathcal{X}^n, \ n \in \mathbb{N}$$

generated by an unknown arbitrary process distribution, is formed as the concatenation of a number $\kappa + 1$ of sequences

$$X_{1..\lfloor n\theta_1 \rfloor}, X_{\lfloor n\theta_1 \rfloor + 1..\lfloor n\theta_2 \rfloor}, \ldots, X_{\lfloor n\theta_\kappa \rfloor + 1..n}$$

where $\theta_k \in (0, 1), \ k = 1..\kappa$. Each of the sequences

$$X_{\lfloor n\theta_{k-1} \rfloor + 1..\lfloor n\theta_k \rfloor}, \ k = 1..\kappa + 1, \ \theta_0 := 0, \ \theta_{\kappa+1} := 1$$

is generated by one out of $r \leqslant \kappa + 1$ *stationary ergodic* process distributions $\rho_1, \ldots, \rho_r$. The process distributions $\rho_1, \ldots, \rho_r$ are unknown and may be dependent. Moreover, the means, variances, or, more generally, their finite-dimensional marginal distributions of any fixed size are not required to be different. We consider the most general scenario where the *process distributions are different*. The process by which $\boldsymbol{x}$ is obtained may be formally defined as follows. Recall that the matrix $\mathbf{X} \in (\mathcal{X}^\infty)^\infty$ of random variables defined by (2.2) in Section 2, is generated by an arbitrary, unknown process distribution ρ. Consider the first $\kappa + 1$ rows of $\mathbf{X}$ and assume that

1. the marginal distribution of ρ over the k^{th} row $X_1^{(k)}, X_2^{(k)}, \ldots$ of $\mathbf{X}$ for $k = 1..\kappa + 1$, is one of r unknown stationary ergodic process distributions $\rho_1, \ldots, \rho_r$;

2. the marginal distributions over the rows of $\mathbf{X}$ indexed by consecutive indices $k,\ k+1,\ k = 1..\kappa$ are different, so that for all $k = 1..\kappa$, the rows of $\mathbf{X}$ indexed by k and $k+1$, namely, $X_1^{(k)}, X_2^{(k)}, \ldots$, and $X_1^{(k+1)}, X_2^{(k+1)}, \ldots$ are generated by two different process distributions in $\{\rho_1, \ldots, \rho_r\}$.

The sequence $\boldsymbol{x}$ is obtained by first fixing a length $n \in \mathbb{N}$ and then concatenating the segments $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{\kappa+1}$, where

$$\boldsymbol{x}_k := X_1^{(k)}, \ldots, X_{\lfloor n(\theta_k - \theta_{k-1}) \rfloor}^{(k)},\ k = 1..\kappa + 1$$

is the sequence obtained as the first $\lfloor n(\theta_k - \theta_{k-1}) \rfloor$ elements of the k^{th} row of $\mathbf{X}$ with $\theta_0 := 0,\ \theta_{\kappa+1} := 1$. For simplicity of notation, we drop the superscript $(k),\ k = 1..\kappa+1$, since its value is always clear from the context. Moreover, the floor function around $n\theta_k,\ k = 1..\kappa$ will often be dropped and assumed implicit.

Note that non-consecutive segments $\boldsymbol{x}_k$ are allowed to have the same time series distribution. Thus, there exists a ground-truth partitioning

$$\{\mathcal{G}_1, \ldots, \mathcal{G}_r\} \tag{3.1}$$

of the set $\{1..\kappa + 1\}$ into r disjoint subsets where for every $k = 1..\kappa + 1$ and $r' = 1..r$ we have $k \in \mathcal{G}_{r'}$ if and only if $\boldsymbol{x}_k$ is generated by $\rho_{r'}$. The parameters $\theta_k,\ k = 1..\kappa$ specify the *change points* $\lfloor n\theta_k \rfloor$ which separate consecutive segments $\boldsymbol{x}_k, \boldsymbol{x}_{k+1}$ generated by *different* process distributions. The change points are *unknown* and have to be estimated. Observe that by this formulation, the finite-dimensional marginals of any

fixed size before and after the change points may be the same. We consider the most general scenario, where the change is in the *process distribution*.

Let the minimum separation of the change point parameters θ_k , $k = 1..\kappa$ be defined as

$$\lambda_{\min} := \min_{k=1..\kappa+1} \theta_k - \theta_{k-1}. \tag{3.2}$$

Since the consistency properties we are after are asymptotic in n, we require that $\lambda_{\min} > 0$. Note that this linearity condition is standard in the change point literature, although it may be unnecessary when simpler formulations of the problem are considered, for example when the samples are i.i.d. However, conditions of this kind are inevitable in the general setting that we consider, where the segments and the samples within each segment are allowed to be arbitrarily dependent: if the length of one of the sequences is constant or sub-linear in n then asymptotic consistency is not possible in this setting. At the same time, we do not make any assumptions on the distance between the process distributions (e.g., the distributional distance): they may be arbitrarily close.

Our goal is to devise algorithms that provide estimates $\widehat{\theta}_k$ for the parameters θ_k, $k = 1..\kappa$. The algorithms must be *asymptotically consistent,* meaning that the difference between each parameter and its estimate goes to 0 almost surely:

$$\lim_{n\to\infty} \sup_{k=1..\kappa} |\widehat{\theta}_k(n) - \theta_k| = 0 \text{ a.s..} \tag{3.3}$$

Under this general formulation, if the correct number κ of change points is not known, it is provably impossible for an algorithm to find it. This is a direct implication of the impossibility theorem of (Ryabko, 2010b) discussed in Section 2.3. Indeed, as follows from this result, given two samples generated by stationary ergodic process distributions, it is impossible to distinguish between the case where they are generated by the same source or by two different ones. As a result, some additional information has to be provided to the algorithms. We consider the following three formulations of the change point problem which differ in the nature of the available additional information.

3.2.1 Known number κ of change points

In the first formulation, we assume that the correct number κ of change points is provided as the only additional parameter. In this case our goal is to construct an

algorithm that, given $\boldsymbol{x}$ and κ, outputs the estimates $\widehat{\theta}_k, k = 1..\kappa$ that satisfy (3.3). An algorithm that achieves this goal is provided in Section 3.3.1. Note that, even though each segment is of length at least $n\lambda_{\min}$, the segments may be arbitrarily short. On the other hand, $\lambda_{\min}$ is unknown, and no a priori knowledge on the minimum length $n\lambda_{\min}$ of the segments is available in this formulation. Therefore, the main challenge in addressing this problem is to ensure that the algorithm is robust with respect to segments of arbitrarily small length.

3.2.2 List-estimation, unknown number of change points

In the second formulation, we assume that κ is unknown. In order to get around the impossibility of estimating κ, we aim to produce a (sorted) list of candidate estimates whose first κ elements converge to the true change point parameters. More precisely, for a sequence $\boldsymbol{x} \in \mathcal{X}^n$ with κ change points at least $n\lambda_{\min}$ apart, a *list-estimator* generates an exhaustive list of possibly more than κ candidate estimates (but makes no attempt to estimate κ). The produced list must have the property that its first κ elements converge to the true parameters $\theta_k,\ k = 1..\kappa$. In order to achieve this goal, the list-estimator requires an additional parameter $\lambda \in (0, 1)$, which is a lower-bound on the minimum separation $\lambda_{\min}$. Thus, the objective in this case is to generate a list-estimator that is consistent in the sense of the following definition.

Definition 3.2.1 (List-estimator)**.** *A list-estimator Υ is a function that, given a sequence $\boldsymbol{x} \in \mathcal{X}^*$ and a parameter $\lambda \in (0, 1)$, produces a list $\Upsilon(\boldsymbol{x}, \lambda) := (\widehat{\theta}_1(n), \ldots, \widehat{\theta}_{|\Upsilon|}(n)) \in (0, 1)^{|\Upsilon|}$ of some $|\Upsilon| \geqslant \kappa$ estimates, where $n \in \mathbb{N}$ denotes the length of $\boldsymbol{x}$. Let $(\widehat{\theta}_{\mu_1}, \widehat{\theta}_{\mu_2}, \ldots, \widehat{\theta}_{\mu_\kappa}) := \boldsymbol{sort}(\widehat{\theta}_1, \ldots, \widehat{\theta}_\kappa)$ be the first κ elements of $\Upsilon(\boldsymbol{x}, \lambda)$, sorted in increasing order of value. We call* Υ asymptotically consistent *if for every $\lambda \in (0, \lambda_{\min}]$ with probability* 1 *we have*

$$\lim_{n\to\infty} \sup_{k=1..\kappa} |\widehat{\theta}_{\mu_k}(n) - \theta_k| = 0.$$

An algorithm that achieves this goal is provided in Section 3.3.2.

Remark 3.2.2 (Relation to the formulation in Section 3.2.1)**.** *Given that the first κ elements of the list produced by a consistent list-estimator converge to the true change point parameters, it may seem that a solution to the second formulation can also be used*

to solve the first problem formulation (for known κ): simply take the first κ estimates from the list produced by the list-estimator. However, there is an important difference between the two formulations. A list-estimator relies on an a priori lower bound $\lambda \in (0, \lambda_{\min}]$ on the minimum separation $\lambda_{\min}$ of the change point parameters, while this parameter is not available in the first setting described in Section 3.2.1.

3.2.3 Finding κ, assuming a known number r of different process distributions:

Finally, in the third formulation, we are able to estimate κ. To this end, we assume that while κ is unknown, the correct number r of different process distributions is provided. In addition, a lower-bound on the minimum separation $\lambda_{\min}$ of the parameters $\theta_k,\ k = 1..\kappa$ is also assumed known. It turns out, that this is sufficient for an algorithm to be able to find the correct number of change points κ. Thus, we seek a change point estimator, that, given a sequence $\boldsymbol{x}$ and the parameter r, outputs the estimated number of change points $\widehat{\kappa}$ and estimated change points $\widehat{\theta}_1, \ldots, \widehat{\theta}_{\widehat{\kappa}}$. We require that with probability 1 from some n on $\widehat{\kappa} = \kappa$, and that the estimates $\widehat{\theta}_k$ satisfy (3.3). An algorithm that achieves this goal is provided in Section 3.3.3.

Note that in the specific case where all of the process distributions are different, knowing r amounts to knowing the number of change points ($r = \kappa + 1$), and we arrive at the first formulation. More generally, the required additional parameter r can be very different from $\kappa+1$, and, as mentioned in the introduction, has a natural interpretation in many real-world applications. For instance, the sequence $\boldsymbol{x}$ may correspond to the behavior of a system over time, which may have alternated $\kappa > r - 1$ times between a known number r of states. In a simple case, the system may only take on $r = 2$ states, for example, "normal" and "abnormal." Moreover, as an example in addition to those given in the introduction, this setting may be well-suited for a speech segmentation problem in which the total number r of speakers is known, while the number κ of alternations between the speakers is not available. Thus, the number r of process distributions is provided as an intrinsic part of the problem, which can be shown to be sufficient to estimate κ.

3.3 Main results

We present three algorithms corresponding to each of the problem formulations presented in Section 3.2. The solution to the first formulation is given by a multiple change point estimator in Section 3.3.1. It is presented by Algorithm 1 which as we show in Theorem 3.3.2 is asymptotically consistent. In Section 3.3.2 we present Algorithm 2 which is a list-estimator (see Definition 3.2.1) that produces an exhaustive list of change point candidates, but makes no attempt to estimate the number κ of change points. The consistency of Algorithm 2 is established in Theorem 3.3.3. We address the third problem formulation in Section 3.3.3, where a multiple change point estimator is given by Algorithm 3. This method relies on a consistent list-estimator (such as Algorithm 2) as a sub-routine. The consistency of the algorithm is established in Theorem 3.3.4. In this section we describe our methods and state the main consistency results, informally explaining why they hold. The proofs of the theorems are deferred to Section 3.4.

The following two operators namely, the intra-subsequence distance $\Delta_{\boldsymbol{x}}$ and the single-change-point estimator $\Phi_{\boldsymbol{x}}$ are used in our methods, namely in Algorithm 1 and Algorithm 2.

Definition 3.3.1. *Let $\boldsymbol{x} = X_{1..n}$ be a sequence and consider a subsequence $X_{a..b}$ of $\boldsymbol{x}$ with $a < b \in 1..n$.*

i. Define the intra-subsequence distance of $X_{a..b}$ as

$$\Delta_{\boldsymbol{x}}(a,b) := \widehat{d}(X_{a..\lfloor \frac{a+b}{2} \rfloor}, X_{\lceil \frac{a+b}{2} \rceil..b}) \tag{3.4}$$

ii. Define the single-change-point estimator of $X_{a..b}$ as

$$\Phi_{\boldsymbol{x}}(a,b,\alpha) := \operatorname*{argmax}_{t \in a..b} \widehat{d}(X_{a-n\alpha..t}, X_{t..b+n\alpha}) \tag{3.5}$$

where $\alpha \in (0,1)$.

3.3.1 Known number κ of change points

We address the first problem formulation where, as described in Section 3.2, the number κ of change points is the only additional parameter provided. The solution is given by Algorithm 1.

Before describing the step-by-step procedure, let us start by giving an overview of what Algorithm 1 aims to do. The algorithm attempts to simultaneously estimate all κ change points using the single change point estimator $\Phi_{\boldsymbol{x}}$ given by (3.5). To this end, it partitions the input-sequence $\boldsymbol{x}$ into non-overlapping segments where, using $\Phi_{\boldsymbol{x}}$ it is possible to estimate every change point consistently. For $\Phi_{\boldsymbol{x}}$ to produce asymptotically consistent estimates, the partitioning has to have each change point isolated within a single segment of the partition, and the lengths of all of the segments in the partition have to be linear functions of n. Moreover, the segments that contain the change points should be *sufficiently far* from other change points, where "sufficiently" far means within a distance linear in n. Such partitioning may be obtained by dividing $\boldsymbol{x}$ into consecutive non-overlapping segments, each of length $n\alpha$ with $\alpha := \lambda/3$ for some $\lambda \in (0, \lambda_{\min}]$ where $\lambda_{\min}$ is given by (3.2). Since, by definition, $\lambda_{\min}$ specifies the minimum separation of the change point parameters, the resulting partitioning has the property that every three consecutive segments contain in *at most one* change point. Observe that even if $\lambda_j \leqslant \lambda_{\min}$, not all segments in the partition contain a change point. However, $\lambda_{\min}$ is not known to the algorithm. Moreover, even if $\lambda_j \leqslant \lambda_{\min}$, not all segments in the partition contain a change point. The algorithm uses the score function $\Delta_{\boldsymbol{x}}$ given by (3.4) to identify the segments that contain change points. As for $\lambda_{\min}$, instead of trying to find it, the algorithm produces many partitionings (using different guesses of $\lambda_{\min}$), and produces a set of candidate change point estimates using each of them. Finally, a weighted combination of the candidate estimates is produced. The weights are designed to converge to zero on iterations where our guess for a lower bound on $\lambda_{\min}$ is incorrect.

More specifically, Algorithm 1 works as follows. Given a sequence $\boldsymbol{x} \in \mathcal{X}^n$ the algorithm iterates over $j = 1..\log n$ and at each iteration, it produces a guess λ_j as a lower-bound on the true minimum separation $\lambda_{\min}$ of the change point parameters. For every fixed j, a total of $\kappa + 1$ grids are generated, each composed of evenly-spaced boundaries $b_i^{t,j},\ i = 0..\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor$, that are $n\alpha_j$ apart for $\alpha_j := \lambda_j/3,\ \lambda_j := 2^{-j}$. The generated grids have distinct starting positions $\frac{n\alpha_j}{t+1}$ for $t = 1..\kappa + 1$. (As shown in the proof of Theorem 3.3.2, this ensures that for a fixed j at least one of the grids for some $t \in 1..\kappa+1$ has the property that the change points are not located at the boundaries.) Among the segments of the grid, κ segments of highest intra-subsequence distance $\Delta_{\boldsymbol{x}}$, (given by (3.4)) are selected. The single-change-point estimator $\Phi_{\boldsymbol{x}}$ is used to seek a candidate change point parameter in each of the selected segments. The weighted

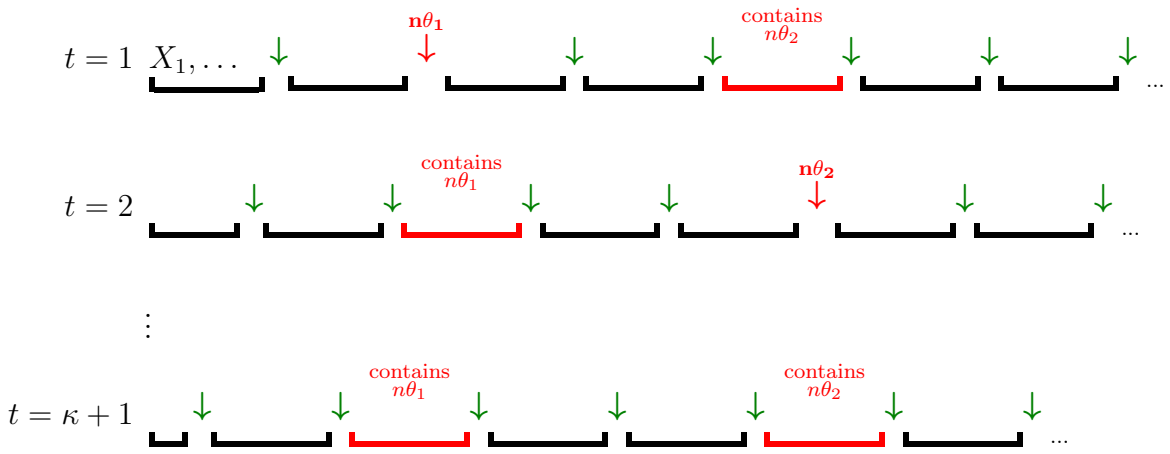

Figure 3.1: For a fixed j, Algorithm 1 generates $\kappa + 1$ grids composed of segments of length $n\alpha_j$ but with distinct starting points: $n\alpha_j/(t+1)$, $t = 1..\kappa + 1$, where α_j is the algorithm's guess of $\lambda_{\min}/3$. At the iteration shown in this figure, α_j is small enough so that every three consecutive segments contain at most one change point. Since there are κ change points, there exists at least one grid (in this example the one corresponding to $t = \kappa + 1$) with the property that none of the change points are located exactly at the boundaries.

Algorithm 1 A multiple change point estimator for known κ

input: $\boldsymbol{x} = X_{1..n}$, Number κ of Change points
initialize: $\eta \leftarrow 0$
for $j = 1..\log n$ **do**
  $\lambda_j \leftarrow 2^{-j},\ \alpha_j \leftarrow \lambda_j/3,\ w_j \leftarrow 2^{-j}$ ▷ Set the step size and iteration weight
  for $t = 1..\kappa + 1$ **do**
    $b_i^{t,j} \leftarrow n\alpha_j(i + \frac{1}{t+1}),\ i = 0..\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor$ ▷ Generate boundaries
    for $l = 0..2$ **do**
      $d_i \leftarrow \Delta_{\boldsymbol{x}}(b_{l+3(i-1)}^{t,j}, b_{l+3i}^{t,j}),\ i = 1..\frac{1}{3}(\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor - l)$ ▷ where $\Delta_{\boldsymbol{x}}$ is given by (3.4)
      $\gamma_l \leftarrow d_{[\kappa]}$ ▷ Store the κ^{th} highest value
    end for
    $\gamma(t,j) \leftarrow \min_{l=0..2} \gamma_l$ ▷ Calculate the grid's performance score
    $\widehat{\pi}_k^{t,j} := \Phi_{\boldsymbol{x}}(b_{[k]}^{t,j}, b_{[k]}^{t,j}, \alpha_j),\ k = 1..\kappa$ ▷ Estimate change points in κ segments of highest $\Delta_{\boldsymbol{x}}$
    $\eta \leftarrow \eta + w_j\gamma(t,j)$ ▷ Update the sum of weights
  end for
end for
$\widehat{\theta}_k \leftarrow \frac{1}{n\eta} \sum_{j=1}^{\log n} \sum_{t=1}^{\kappa+1} w_j\gamma(t,j)\widehat{\pi}_k^{t,j},\ k = 1..\kappa$ ▷ Calculate the final estimates
return: $\widehat{\theta}_1, \ldots, \widehat{\theta}_\kappa$

combination is given as the final estimate for every change point parameter $\theta_k,\ k = 1..\kappa$. To this end, two sets of weights are used, namely, an iteration weight $w_j := 2^{-j}$ and a performance score $\gamma(t,j)$. The former gives lower precedence to finer grids. To calculate the latter, at each iteration on j and t, for every fixed $l \in 0..2$, a partition of the grid is considered, which is composed of the boundaries $b_l + 3i,\ \frac{1}{3}(\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor - l)$. Each partition, in turn, specifies a set of non-overlapping consecutive segments of length $n\lambda_j$, for each of which a parameter γ_l is calculated as the κ^{th} highest intra-distance value Δ_x of its segments; the performance weight $\gamma(t,j)$ is obtained as $\min_{l=0..2} \gamma_l$. (As shown in the proof, $\gamma(t,j)$ converges to zero on the iterations where either $\lambda_j > \lambda_{\min}$ or there exists some change point parameter θ_k that is on the boundary of one of the segments of the partition.)

Theorem 3.3.2. *Algorithm 1 is asymptotically consistent, provided that the correct*

number κ of change points is given:

$$\lim_{n\to\infty} \sup_{k=1..\kappa} |\widehat{\theta}_k(n) - \theta_k| = 0 \ \ a.s..$$

The proof is provided in Section 3.4.2. Here, we provide an intuitive description of the consistency result.

Proof Sketch:

To see why this procedure works, first observe that the empirical estimate $\widehat{d}(\cdot,\cdot)$ of the distributional distance is consistent. Thus, the empirical distributional distance between a given pair of sequences converges to the distributional distance between their generating processes. From this we can show that the intra-subsequence distance $\Delta_{\boldsymbol{x}}$ corresponding to the segments in the grid that do not contain a change point converges to zero. This is established in Lemma iii, provided in Section 3.4. On the other hand, since the generated grid becomes finer as a function of j, from some j on, we have $\alpha_j < \lambda_{\min}/3$ so that every three consecutive segments of the grid contain *at most* one change point. In this case, for every segment that contains a change point, the single-change-point estimator $\Phi_{\boldsymbol{x}}$ produces an estimate that, for long enough segments, becomes arbitrarily close to the true change point. This is shown in Lemma 3.4.3, provided in Section 3.4. Moreover, for large enough n, the performance scores associated with these segments are bounded below by some non-zero constant. Thus, the κ segments of highest $\Delta_{\boldsymbol{x}}$ each contain a change point which can be estimated consistently using $\Phi_{\boldsymbol{x}}$. However, the estimates produced at a given iteration for which $\alpha_j > \lambda_{\min}/3$ may be arbitrarily bad. Moreover, recall that even for $\alpha_j \leqslant \lambda_{\min}/3$, an appropriate grid to provide consistent estimates must have the property that no change point is exactly at the start or at the end of a segment. However, it is not possible to directly identify such appropriate grids. The following observation is key to the indirect identification of such appropriate segments.

Consider the partitioning of $\boldsymbol{x}$ into κ consecutive segments where there exists at least one segment with more than one change point. Since there are exactly κ change points, there must exist at least one segment in this partitioning that does not contain any change points at all. As follows from Lemma iii, the segment that contains no change points has an intra-subsequence distance $\Delta_{\boldsymbol{x}}$ that converges to 0. On the iterations for which $\alpha_j > \lambda_{\min}/3$, at least one of the three partitions has the property that among

every set of κ segments in the partition, there is *at least* one segment that contains no change points. In this case, $\Delta_{\boldsymbol{x}}$ corresponding to the segment without a change point converges to 0. The same argument holds for the case where $\alpha_j \leqslant \lambda_{\min}$, while at the same time a change point happens to be located exactly at the boundary of a segment in the grid. Observe that for a fixed j, the algorithm forms a total of $\kappa + 1$ different grids, with the same segment size, but distinct starting points $\frac{n\alpha_j}{t+1}$ $t = 1..\kappa + 1$. Since there are κ change points, for all j such that $\alpha_j \leqslant \lambda_{\min}/3$ there exists at least one appropriate grid (for some $\tau \in 1..\kappa + 1$), that simultaneously contains all the change points within its segments. In this case, $\gamma(\tau, j)$ converges to a non-zero constant. The final estimate $\widehat{\theta}_k$ for each change point parameter θ_k is obtained as a weighted sum of the candidate estimates produced at each iteration. Two sets of weights are used in this step, namely $\gamma(t, j)$ and w_j, whose roles can be described as follows.

1. $\gamma(t, j)$ is used to penalize for the (arbitrary) results produced on iterations on $j \in 1..\log n$ and $t \in 1..\kappa + 1$ where , either $\alpha_j > \lambda_{\min}/3$, or while we have $\alpha_j \leqslant \lambda_{\min}/3$ there exists some θ_k for some $k \in 1..\kappa$ such that $\lfloor n\theta_k \rfloor \in \{b_i^{t,j} : i = 0..\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor\}$. As discussed, $\gamma(t, j)$ converges to zero only on these iterations, while it is bounded below by a non-zero constant on the rest.

2. w_j is used to give precedence to estimates sought in longer segments. Since the grids are finer for larger j, at some higher iterations the segments may not be long enough to produce consistent estimates.

Therefore, if n is large enough, the final estimates $\widehat{\theta}_k,\ k = 1..\kappa$ produced by Algorithm 1 converge to the true change point parameters, $\theta_k,\ k = 1..\kappa$. □

3.3.2 List-estimation

In this section we consider the second problem formulation, where the number κ of change points is unknown. However, due to the theoretical impossibility results discussed in Section 2.3, it is impossible to estimate κ. Instead, we present a list-estimator, namely Algorithm 2, to produce a list of estimated change point parameters, sorted in such a way that its first κ elements estimate θ_k, $k = 1..\kappa$. As shown in Theorem 3.3.3, the proposed method is asymptotically consistent (in the sense of Definition 3.2.1).

The main idea of the approach is similar to that of Algorithm 1. In particular, we partition $\boldsymbol{x}$ into segments, in each of which a change point estimate is sought using

$\Phi_{\boldsymbol{x}}$, and is given a performance score using $\Delta_{\boldsymbol{x}}$. However, in the present setting κ is unknown. Therefore, unlike the previous method, Algorithm 2 cannot rely on κ to *choose* appropriate segments that contain the true change points. Instead, it estimates a change point in every segment of the grid and sorts the estimates in decreasing order of their performance scores. More specifically, Algorithm 2 works as follows. Given $\lambda \in (0, 1)$ (which is provided to the algorithm), a sequence of evenly-spaced indices b_i^t is formed. The index-sequence is used to partition $\boldsymbol{x} = X_{1..n}$ into consecutive segments of length $n\alpha$, where $\alpha := \lambda/3$. The single-change-point estimator $\Phi_{\boldsymbol{x}}$ is used to generate a candidate change point within every segment. Moreover, the intra-subsequence-distance $\Delta_{\boldsymbol{x}}$ of each segment is used as its performance score. The change point candidates are ordered according to the performance scores of their corresponding segments. Recall that, the algorithm assumes the input parameter λ to be a lower-bound on the true minimum separation $\lambda_{\min}$ of the actual change point parameters. The sorted list of estimated change points is filtered in such a way that its elements are at least λ apart. This is done using a greedy procedure: a set of available estimates is maintained, and at each step, an available estimate of highest score is selected, and added to the final list. The available change point candidates within $n\lambda/2$ of the selected change point estimate are made unavailable. The algorithm proceeds until the set of available estimates becomes empty. There may be more than κ candidate estimates produced; however, as shown in Theorem 3.3.3, for long enough $\boldsymbol{x}$ the first κ estimates $\widehat{\theta}_k,\ k = 1..\kappa$ of the sorted list converge to the true change points $\theta_1, \cdots, \theta_\kappa$.

Theorem 3.3.3. *Algorithm 2 is an asymptotically consistent list-estimator in the sense of Definition 3.2.1, provided that $\lambda \leq \lambda_{\min}$, where $\lambda_{\min}$ is the minimum distance between the true change point parameters $\theta_k, k = 1..\kappa$.*

The proof is provided in Section 3.4.3. Here we give an intuitive explanation.

Proof Sketch: When $\lambda \leqslant \lambda_{\min}$, each of the index-sequences generated with $\alpha := \frac{\lambda}{3}$ partitions $\boldsymbol{x}$ in such a way that every three consecutive segments of the partition contain *at most* one change point. Also, the segments are of lengths αn. Therefore, if n is large enough, the single-change-point estimator $\Phi_{\boldsymbol{x}}$ produces correct candidates within each of the segments that contains a true change point. Moreover, the performance scores of the segments without change points converge to 0, while each of those corresponding to the segments that contain a change point converges to a non-zero constant. Thus,

Algorithm 2 A list-estimator Υ

input: Sequence $\boldsymbol{x} = X_{1..n}$, lower bound λ
initialize: $\alpha \leftarrow \lambda/3$, $\widehat{\boldsymbol{\pi}} \leftarrow ()$ ▷ where $\widehat{\boldsymbol{\pi}}$ is an empty sequence
$b_i^t \leftarrow n\alpha(i + \frac{1}{t+1}),\ i = 0..\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor,\ t = 1, 2$ ▷ Generate two sets of boundaries
$s(t, i) \leftarrow \Delta_{\boldsymbol{x}}(b_i^t, b_{i+1}^t),\ i = 0..\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor,\ t = 1, 2$ ▷ Calculate a score for $X_{b_i^t..b_{i+1}^t}$
$\widehat{\pi}_i^t := \Phi_{\boldsymbol{x}}(b_i^t, b_{i+1}^t, \alpha),\ i = 0..\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor - 1,\ t = 1, 2$ ▷ Estimate a change point in $X_{b_i^t..b_{i+1}^t}$
Remove duplicate estimates and sort based on scores:
$\mathcal{U} \leftarrow \{(t, i) : i \in 0..\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor, t = 1, 2\}$
while $\mathcal{U} \neq \varnothing$ **do**
    $(\tau, l) \leftarrow \operatorname{argmax}_{(t,i)\in\mathcal{U}}\ s(t, i)$ ▷ break ties arbitrarily
    $\widehat{\boldsymbol{\pi}} \leftarrow \widehat{\boldsymbol{\pi}} \oplus \widehat{\pi}_l^\tau$ ▷ Append an available candidate of highest score $\widehat{\pi}_l^\tau$ to $\widehat{\boldsymbol{\pi}}$
    $\mathcal{U} \leftarrow \mathcal{U} \setminus \{(t, i) : \widehat{\pi}_i^t \in (\widehat{\pi}_l^\tau - \lambda n/2, \widehat{\pi}_l^\tau + \lambda n/2)\}$ ▷ Remove estimates within $\lambda n/2$
end while
$\widehat{\boldsymbol{\theta}} := (\widehat{\theta}_1, \ldots, \widehat{\theta}_{|\mathcal{U}|}) \leftarrow \frac{1}{n}\widehat{\pi}_1, \ldots, \frac{1}{n}\widehat{\pi}_{|\mathcal{U}|}$ ▷ Generate a list of change point estimates
return: A list $\widehat{\boldsymbol{\theta}}$ of estimated change point parameters.

the κ candidate estimates of highest performance score that are at least at a distance λn from one another, each converge to a unique change point parameter.

Observe that depending on the grid boundaries b_i^t, it may happen that some of the true change points are at the start or at the end of a segment of the grid where candidate estimates are sought. For these change points, even if their corresponding parameters are estimated consistently, they will be associated with arbitrarily small performance scores. To get around this problem, we generate two grids composed of segments of length αn but with distinct starting points: $n\alpha/(t+1),\ t = 1, 2$. This way, every change point is fully contained within *at least* one segment from either of the two partitions. This scenario is depicted in Figure 3.2. Note that this strategy is similar to that in Algorithm 1, with the difference that here, unlike in Algorithm 1, we do not require the grid to simultaneously contain all κ change points within its segments. It suffices to ensure that each change point is contained within at least one segment among the union of those in both grids. From the above argument, if $\boldsymbol{x}$ is long enough, the segments with change points will have higher scores, and their corresponding change point parameters will be estimated correctly, confirming the consistency of Algorithm 2. □

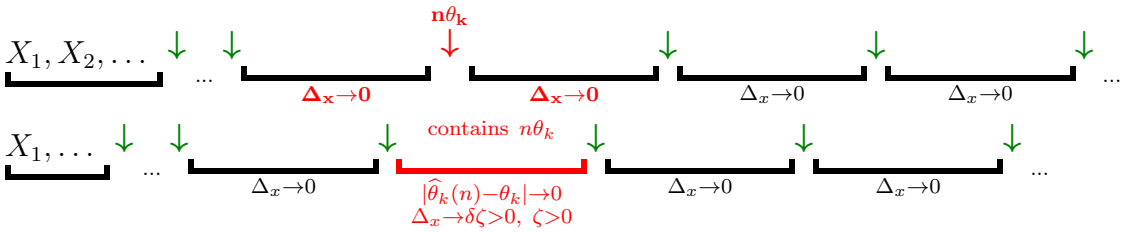

Figure 3.2: Algorithm 2 generates two grids composed of segments of length αn but with distinct starting points: $n\alpha/(t+1)$, $t = 1, 2$. This way, every change point is fully contained within *at least* one segment from either of the two partitions. Since $\alpha < \lambda_{\min}/3$, every three consecutive segments contain *at most* one change point.

3.3.3 Known number r of process distributions, unknown number of change points

In the third problem formulation, while κ is unknown an additional parameter namely, the total number r of process distributions is available. As discussed in Section 3.2 this additional parameter has a natural interpretation in many applications. Our goal in this setting is to produce a consistent multiple change point estimation algorithm that not only locates the changes but also finds the correct number κ of change points. We reduce the problem of finding κ to time series clustering via list-estimation introduced in the previous section. This is realized in Algorithm 3 which, as we show in Theorem 3.3.4, is asymptotically consistent.

The key idea is to have a consistent list-estimator (such as Algorithm 2) produce a list of (possibly more than κ) change-point estimates, and then use a consistent time series clustering method to identify the redundant estimates in the list. As discussed in Section 3.2, the segments $\boldsymbol{x}_k$, $k = 1..\kappa$ separated by the change points are each generated by one of r unknown process distributions. Hence there exists a natural partitioning of the indices $1..\kappa+1$ into r groups defined by (3.1), where the indices $k, k' \in 1..\kappa+1$ are grouped together if and only if the segments $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k'}$ are generated by the same process distribution. A time series clustering procedure takes several sequences and groups them into clusters. It is called *asymptotically consistent* (Ryabko, 2010a) if, for large enough n, it puts two sequences into the same cluster if and only if the time series distribution that generated them is the same. However, an important

difference, which makes the consistency result of Ryabko (2010a) not directly applicable here, is that we have to deal with concatenations of sequences generated by different distributions, rather than with individual sequences each generated by a single distribution.

Algorithm 3 A multiple change point estimator for known r

input: $\boldsymbol{x} \in \mathcal{X}^n$, $\lambda \in (0, \lambda_{\min}]$, Number r of process distributions
$\Upsilon \leftarrow \Upsilon(\boldsymbol{x}, \lambda)$ ▷ Obtain a list of candidate estimates via a consistent list-estimator
$\{\psi_i : i = 1..|\Upsilon|\} \leftarrow \mathbf{sort}(\{n\widehat{\theta} : \widehat{\theta} \in \psi\})$ ▷ so that $i < j \Leftrightarrow \psi_i < \psi_j,\ i, j \in 1..|\Upsilon|$.
$\psi_0 \leftarrow 0,\ \psi_{|\Upsilon|+1} \leftarrow n$
$\mathcal{S} \leftarrow \{\widetilde{\boldsymbol{x}}_i := X_{\psi_{i-1}+1..\psi_i} : i = 1..|\Upsilon| + 1\}$ ▷ Generate a set $\mathcal{S}$ of consecutive segments
Partition $\mathcal{S}$ into r clusters:
$c_1 \leftarrow 1$ ▷ Initialize r farthest segments as cluster centers
$c_j \leftarrow \operatorname{argmax}_{i=1..|\Upsilon|} \min_{i'=1}^{j-1} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{x}}_{c_{i'}}),\ j = 2..r$
$T(\widetilde{\boldsymbol{x}}_i) \leftarrow \operatorname{argmin}_{j=1..r} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{x}}_{c_j}),\ i = 1..|\Upsilon|$ ▷ Assign every segment to a cluster
Eliminate redundant estimates:
$\mathcal{C} \leftarrow \{1..|\Upsilon|\}$
for $i = 1..|\Upsilon|$ **do**
  if $T(\widetilde{\boldsymbol{x}}_i) = T(\widetilde{\boldsymbol{x}}_{i+1})$ **then**
    $\mathcal{C} \leftarrow \mathcal{C} \setminus \{i\}$
  end if
end for
$\widehat{\kappa} \leftarrow |\mathcal{C}|$ ▷ Estimate the number κ of change points
$\widehat{\theta}_i := \frac{1}{n}\psi_i,\ i \in \mathcal{C}$ ▷ Generate the final estimates for $\theta_k,\ k = 1..\kappa$
return: $\widehat{\kappa}$, $\widehat{\theta}_i,\ i \in \mathcal{C}$

More specifically, Algorithm 3 works as follows. First, a consistent list-estimator is used to obtain an initial set of change point candidates. The estimates are sorted in *increasing order* to produce a set $\mathcal{S}$ of consecutive non-overlapping segments of $\boldsymbol{x}$. The set $\mathcal{S}$ is then partitioned into r clusters. We use the following clustering procedure. First, a total of r cluster centers are obtained as follows. The first segment $\boldsymbol{x}_1$ is chosen as the first cluster center. Iterating over $j = 2..r$ a segment is chosen as a cluster center if it has the highest minimum distance from the previously chosen cluster centers. Once the r cluster centers are specified, the remaining segments are assigned to the closest cluster. In each cluster, the change point candidate that joins a pair of consecutive segments of $\boldsymbol{x}$ is identified as *redundant* and is removed from the list. Once all of the redundant candidates are removed, the algorithm outputs the remaining candidate

estimates.

Theorem 3.3.4. *Algorithm 3 is an asymptotically consistent, which means that, with probability one from some n on*

$$\widehat{\kappa} = \kappa$$

and

$$\lim_{n\to\infty} \sup_{k=1..\kappa} |\widehat{\theta}_k(n) - \theta_k| = 0,$$

provided $\lambda \in (0, \lambda_{\min}]$ and the correct number r of process distributions are given.

The proof is provided in Section 3.4.4. Here, we give an intuitive explanation as to why the algorithm works.

Proof Sketch: Since a consistent list-estimator Υ (such as Algorithm 2) is used in the first step, for long enough $\boldsymbol{x}$ an initial set of possibly more than κ estimated parameters is generated, that contains κ elements which are arbitrarily close to the true change point parameters. (Since κ is unknown, the fact that the correct estimates are located first in the list is irrelevant; all we can use here is that the correct change point estimates are somewhere in the list.) Therefore, if $\boldsymbol{x}$ is long enough, the largest portion of each segment in $\mathcal{S}$ is generated by a single process distribution. Since the initial change point candidates are at least $n\lambda$ apart, the lengths of the segments in $\mathcal{S}$ are linear functions of n. Thus, we can show that for large enough n, the empirical estimate of the distributional distance between a pair of segments in $\mathcal{S}$ converges to 0 if and only if the same process distribution generates most of the two segments. Given the total number r of process distributions, for long enough $\boldsymbol{x}$ the clustering algorithm groups together those and only those segments in $\mathcal{S}$ that are generated by the same process distribution. This lets the algorithm identify and remove the redundant candidates. By the consistency of Υ, the remaining estimates converge to the true change point parameters. □

3.3.4 Computational complexity

It is easy to see that the proposed methods are computationally efficient and can be easily implemented. Indeed, the most computationally exhaustive part of all three algorithms is the calculation of the distributional distance $\widehat{d}$, which, as follows from

Remark 2.2.8 has complexity bounded above by $\mathcal{O}(n \text{ polylog } n)$ (with $m_n := \log n$) where n is the length of the segment. In Algorithm 1, for a fixed j, a total of $1/\alpha_j$ distance calculations are done on segments of length $3\alpha_j$, and a total of $\kappa\alpha_j n$ distance calculations are done to estimate each change point; the procedure is repeated $\kappa + 1$ times. Summing over $j \in 1..\log n$ iterations, the overall complexity is of order at most $\mathcal{O}(\kappa^2 n^2 \text{ polylog } n)$. The rest of the computations are of negligible order. In Algorithm 2 and Algorithm 3, at most n distance calculations are made. Thus, by a similar argument, the computational complexity of these algorithms is of order at most $\mathcal{O}(n^2 \text{ polylog } n)$.

3.4 Proofs

In this section we prove the consistency of the methods proposed in Section 3.3, namely Theorems 3.3.2, 3.3.3 and 3.3.4. The main proofs rely on some technical lemmas stated in Section 3.4.1. The following additional notation will also be used in the proofs. Let $\boldsymbol{x} \in \mathcal{X}^n,\ n \in \mathbb{N}$ be a sequence with κ change points. consider a sequence $\boldsymbol{b} := b_1, \dots b_{|\boldsymbol{b}|} \in \cup_{i=1}^n \{1..n\}^i$. For every change point parameter $\theta_k,\ k = 1..\kappa$, denote by

$$L(k) := \max_{b \in \boldsymbol{b}} \{b : b \leqslant n\theta_k\} \text{ and } R(k) := \min_{b \in \boldsymbol{b}} \{b : b > n\theta_k\} \tag{3.6}$$

the elements of $\boldsymbol{b}$ that appear immediately to the left and to the right of $\lfloor n\theta_k \rfloor$ respectively.

3.4.1 Technical lemmas

For simplicity of exposition, we let $m_n := n$ and $l_n := \infty$ throughout this section, where m_n and l_n are the parameters of the empirical estimates of the distributional distance given by (2.3) and (2.4) in Definition 2.2.3. It is easy to check that the same arguments hold for the general case.

Lemma 3.4.1. *Let $\boldsymbol{x} = X_{1..n}$ be generated by a stationary ergodic process ρ. For all $\alpha \in (0, 1)$ the following statements hold with ρ-probability 1:*

(i) $\lim_{n\to\infty} \sup_{\substack{b_1, b_2 \in 1..n \\ |b_2 - b_1| \geqslant \alpha n}} \sum_{\substack{B \in B^{m,l} \\ m,l \in 1..T}} |\nu(X_{b_1..b_2}, B) - \rho(B)| = 0$ *for every* $T \in \mathbb{N}$.

(ii) $\lim_{n\to\infty} \sup_{\substack{b_1,b_2\in 1..n \\ |b_2-b_1|\geqslant \alpha n}} \widehat{d}(X_{b_1..b_2},\rho)=0.$

(iii) $\lim_{n\to\infty} \sup_{|b_2-b_1|\geqslant \alpha n} \Delta_{\boldsymbol{x}}(b_1,b_2)=0$

Proof. To prove part (i) we proceed as follows. Assume by way of contradiction that the statement is not true. Therefore, there exists some $\lambda>0$, $T\in\mathbb{N}$ and sequences $b_1^{(i)}\in 1..n_i$ and $b_2^{(i)}\in 1..n_i,\ n_i,i\in\mathbb{N}$ with $|b_2^{(i)}-b_1^{(i)}|\geqslant \alpha n$, such that with probability $\Delta>0$ we have

$$\sup_{i\in\mathbb{N}} \sum_{\substack{B\in B^{m,l} \\ m,l\in 1..T}} |\nu(X_{b_1^{(i)}..b_2^{(i)}},B)-\rho(B)|>\lambda. \tag{3.7}$$

Using the definition of $\nu(\cdot,\cdot)$ it is easy to see that the following inequalities hold

$$\begin{aligned} |\nu(X_{b_1..b_2},B)-\rho(B)| &\leqslant |(1-\frac{m-1}{b_2-b_1})\nu(X_{b_1..b_2},B)-\rho(B)|+\frac{m-1}{b_2-b_1} \\ &\leqslant \sum_{i=1}^{2}\frac{b_i}{b_2-b_1}|\nu(X_{1..b_i},B)-\rho(B)|+\frac{4(m-1)}{b_2-b_1} \end{aligned} \tag{3.8}$$

for every $B\in B^{m,l},\ m,l\in\mathbb{N}$ and all $b_1<b_2\in\mathbb{N}$.

Fix $\varepsilon>0$. For each $m,l\in 1..T$ we can find a finite subset $S^{m,l}$ of $B^{m,l}$ such that

$$\rho(S^{m,l})\geqslant 1-\frac{\varepsilon}{T^2 w_m w_l}. \tag{3.9}$$

For every $B\in S^{m,l},\ m,l\in 1..T$, there exists some $N(B)$ such that for all $n\geqslant N(B)$ with probability one we have

$$\sup_{b\geqslant n}|\nu(X_{1..b},B)-\rho(B)|\leqslant \frac{\varepsilon\rho(B)}{T^2 w_m w_l}. \tag{3.10}$$

Define $\zeta_0:=\min_{m,l\in 1..T}\frac{\varepsilon}{T^2 w_m w_l}$ and let $\zeta:=\min\{\alpha,\zeta_0\}$; observe that $\zeta>0$. Let

$$N:=\max_{m,l=1..T,B\in S^{m,l}} N(B)/\zeta. \tag{3.11}$$

Consider the sequence $b_1^{(i)},\ i\in\mathbb{N}$.

1. For every $m, l \in 1..T$ we have

$$\sup_{\substack{i\in\mathbb{N}\\ b_1^{(i)}\leqslant\zeta n}} \frac{b_1^{(i)}}{b_2^{(i)}-b_1^{(i)}} \leqslant \frac{\zeta}{\alpha} \leqslant \frac{\varepsilon}{\alpha T^2 w_m w_l} \tag{3.12}$$

2. On the other hand, by (3.10) and (3.11) for all $n \geqslant N$ we have

$$\sup_{\substack{i\in\mathbb{N}\\ b_1^{(i)}>\zeta n}} |\nu(X_{1..b_1^{(i)}}, B) - \rho(B)| \leqslant \frac{\varepsilon\rho(B)}{T^2 w_m w_l}. \tag{3.13}$$

Increase N if necessary to have

$$\sum_{m,l=1}^{T} w_m w_l \frac{m}{\alpha N} \leqslant \varepsilon. \tag{3.14}$$

For all $n \geqslant N$ and $m \in 1..T$. For all $n \geqslant N$ we obtain

$$\begin{aligned}
&\sup_{i\in\mathbb{N}} \sum_{m,l=1}^{T} w_m w_l \sum_{B\in B^{m,l}} |\nu(X_{b_1^{(i)}..b_2^{(i)}}, B) - \rho(B)| \\
&\leqslant \sup_{i\in\mathbb{N}} \Big(\sum_{m,l=1}^{T} w_m w_l \sum_{B\in S^{m,l}} |\nu(X_{b_1^{(i)}..b_2^{(i)}}, B) - \rho(B)|\Big) + \varepsilon \qquad (3.15)\\
&\leqslant \sup_{i\in\mathbb{N}} \Big(\sum_{m,l=1}^{T} w_m w_l \sum_{B\in S^{m,l}} \frac{b_2^{(i)}}{b_2^{(i)}-b_1^{(i)}} |\nu(X_{1..b_2^{(i)}}, B) - \rho(B)|\Big) \\
&+ \sup_{\substack{i\in\mathbb{N}\\ b_1^{(i)}>\zeta n}} \Big(\sum_{m,l=1}^{T} w_m w_l \sum_{B\in S^{m,l}} \frac{b_1^{(i)}}{b_2^{(i)}-b_1^{(i)}} |\nu(X_{1..b_1^{(i)}}, B) - \rho(B)|\Big) \\
&+ \sup_{\substack{i\in\mathbb{N}\\ b_1^{(i)}\leqslant\zeta n}} \Big(\sum_{m,l=1}^{T} w_m w_l \sum_{B\in S^{m,l}} \frac{b_1^{(i)}}{b_2^{(i)}-b_1^{(i)}} |\nu(X_{1..b_1^{(i)}}, B) - \rho(B)|\Big) + 5\varepsilon \qquad (3.16)\\
&\leqslant \varepsilon(3/\alpha + 5) \qquad (3.17)
\end{aligned}$$

where (3.15) follows from (3.9); (3.16) follows from (3.8) and (3.14); and (3.17) follows from (3.10), (3.13), (3.12), summing over the probabilities, and noting that $\frac{b_2^{(i)}}{b_1^{(i)}-b_2^{(i)}} \leqslant \frac{1}{\alpha}$

for all $b_2^{(i)} - b_1^{(i)} \geqslant \alpha n$. Observe that (3.17) holds for any $\varepsilon > 0$, and it particular it holds for $\varepsilon \in (0, \frac{\lambda}{3/\alpha+5})$. Therefore, we have

$$\sup_{i\in\mathbb{N}} \sum_{\substack{B\in B^{m,l}\\ m,l\in 1..T}} |\nu(X_{b_1^{(i)}..b_2^{(i)}}, B) - \rho(B)| < \lambda$$

contradicting (3.7). Part (i) follows.

(ii) Fix $\varepsilon > 0$, $\alpha \in (0,1)$ and $\zeta \in (0,1)$. We can find some $T \in \mathbb{N}$ such that

$$\sum_{m,l=T}^{\infty} w_m w_l \leqslant \varepsilon. \tag{3.18}$$

By part (i) of Lemma 3.4.1, there exists some N such that for all $n \geqslant N$ we have

$$\sup_{\substack{b_1,b_2\in 1..n\\ |b_2-b_1|\geqslant\alpha n}} \sum_{m,l=1}^{T} \sum_{B\in B^{m,l}} |\nu(X_{b_1..b_2}, B) - \rho(B)| \leqslant \varepsilon. \tag{3.19}$$

From (3.18) and (3.19), for all $n \geqslant N$ we have

$$\begin{aligned}\sup_{\substack{b_1,b_2\in 1..n\\ |b_2-b_1|\geqslant\alpha n}} \widehat{d}(X_{b_1..b_2}, \rho) &\leqslant \sup_{\substack{b_1,b_2\in 1..n\\ |b_2-b_1|\geqslant\alpha n}} \sum_{m,l=1}^{T} w_m w_l \sum_{B\in B^{m,l}} |\nu(X_{b_1..b_2}, B) - \rho(B)| + \varepsilon \\ &\leqslant 2\varepsilon\end{aligned}$$

and part (ii) of the lemma follows.

(iii) Fix $\varepsilon > 0$, $\alpha \in (0,1)$. Without loss of generality assume that $b_2 > b_1$. Observe that for every $b_1 + \alpha n \leqslant b_2 \leqslant n$ we have $\frac{b_1+b_2}{2} - b_1 = b_2 - \frac{b_1+b_2}{2} \geqslant \alpha n/2$. Therefore, by (ii) there exists some N such that for all $n \geqslant N_1$ we have

$$\begin{aligned}\sup_{b_2-b_1\geqslant\alpha n} \widehat{d}(X_{b_1..\frac{b_1+b_2}{2}}, \rho) &\leqslant \varepsilon, \\ \sup_{b_2-b_1\geqslant\alpha n} \widehat{d}(X_{\frac{b_1+b_2}{2}..b_2}, \rho) &\leqslant \varepsilon.\end{aligned}$$

It remains to use the definition of $\Delta_{\boldsymbol{x}}$ (3.4) and the triangle inequality to observe that

$$\begin{aligned}\sup_{b_2-b_1\geqslant\alpha n}\Delta_{\boldsymbol{x}}(b_1,b_2) &= \sup_{b_2-b_1\geqslant\alpha n}\widehat{d}(X_{b_1..\frac{b_1+b_2}{2}}, X_{\frac{b_1+b_2}{2}..b_2})\\ &\leqslant \sup_{b_2-b_1\geqslant\alpha n}\widehat{d}(X_{b_1..\frac{b_1+b_2}{2}},\rho)+\widehat{d}(X_{\frac{b_1+b_2}{2}..b_2},\rho)\leqslant 2\varepsilon\end{aligned}$$

for all $n\geqslant N$, and (iii) follows. □

Lemma 3.4.2. *Assume that a sequence $\boldsymbol{x}=X_{1..n}$ has a change point $\pi=\theta n$ for some $\theta\in(0,1)$ so that the segments $X_{1..\pi}$, $X_{\pi..n}$ are generated by two different process distributions ρ, ρ' respectively. If ρ, ρ' are both stationary ergodic then with probability one, for every $\zeta\in(0,\min\{\theta,1-\theta\})$ we have*

(i) $$\lim_{n\to\infty}\sup_{\substack{b\in 1..(\theta-\zeta)n\\ t\in\pi..(1-\zeta)n}}\widehat{d}(X_{b..t},\frac{\pi-b}{t-b}\rho+\frac{t-\pi}{t-b}\rho')=0$$

(ii) $$\lim_{n\to\infty}\sup_{\substack{b\in\zeta n..\pi\\ t\in(\theta+\zeta)n..n}}\widehat{d}(X_{b..t},\frac{\pi-b}{t-b}\rho+\frac{t-\pi}{t-b}\rho')=0$$

Proof. Fix $\varepsilon>0$, $\theta\in(0,1)$, $\zeta\in(0,\min\{\theta,1-\theta\})$. There exists some $T\in\mathbb{N}$ such that

$$\sum_{m,l=T}^{\infty}w_m w_l\leqslant\varepsilon. \tag{3.20}$$

To prove part (i) of the lemma we proceed as follows. First, by the definition of $\nu(\cdot,\cdot)$ given by (2.1), for all $B\in B^{m,l}$ $m,l\in 1..T$ and all $b\in 1..(\theta-\zeta)n,\ t\in\pi..(1-\zeta)n$ such that $t-\pi>m-1$ we have

$$\begin{aligned}|\nu(X_{\pi..t},B)-\rho'(B)|\leqslant{}&\frac{n-\pi}{t-\pi-m+1}|\nu(X_{\pi..n},B)-\rho'(B)|\\ &+\frac{n-t}{t-\pi-m+1}|\nu(X_{t..n},B)-\rho'(B)|+\frac{3(m-1)}{t-\pi-m+1}\end{aligned} \tag{3.21}$$

Therefore, for all $B\in B^{m,l}$ $m,l\in 1..T$ and all $b\in 1..(\theta-\zeta)n,\ t\in\pi..(1-\zeta)n$ such that

$t-\pi > m-1$ we obtain

$$\begin{aligned}
&|\nu(X_{b..t},B) - \frac{\pi-b}{t-b}\rho(B) - \frac{t-\pi}{t-b}\rho'(B)| \\
&\leqslant |(1-\frac{m-1}{t-b})\nu(X_{b..t},B) - \frac{\pi-b}{t-b}\rho(B) - \frac{t-\pi}{t-b}\rho'(B)| + \frac{m-1}{t-b} \\
&\leqslant \frac{\pi-b}{t-b}|\nu(X_{b..\pi},B)-\rho(B)| + \frac{t-\pi-m+1}{t-b}|\nu(X_{\pi..t},B)-\rho'(B)| + \frac{3(m-1)}{t-b} \\
&\leqslant \frac{\pi-b}{t-b}|\nu(X_{b..\pi},B)-\rho(B)| + \frac{n-\pi}{t-b}|\nu(X_{\pi..n},B)-\rho'(B)| \\
&\qquad\qquad + \frac{n-t}{t-b}|\nu(X_{t..n},B)-\rho'(B)| + \frac{6(m-1)}{t-b} \qquad (3.22)
\end{aligned}$$

where the first inequality follows from the fact that $\nu(\cdot,\cdot) \leqslant 1$, the second inequality follows from the definition of $\nu(\cdot,\cdot)$ given by (2.1) and the third inequality follows from (3.21). Note that (3.22) is easy to verify directly for $t-\pi \leq m-1$. Observe that $\pi - b \geqslant \zeta n$ for all $b \in 1..(\theta-\zeta)n$. Therefore, by part (i) of Lemma 3.4.1, there exists some N' such that for all $n \geqslant N'$ we have

$$\sup_{b\in 1..(\theta-\zeta)n} \sum_{m,l=1}^{T} w_m w_l \sum_{B\in B^{m,l}} |\nu(X_{b..\pi},B)-\rho(B)| \leqslant \varepsilon. \qquad (3.23)$$

Similarly, $n - t \geqslant \zeta n$ for all $t \in \pi..(1-\zeta)n$. Therefore, by part (ii) of Lemma 3.4.1, there exists some N'' such that for all $n \geqslant N''$ we have

$$\sup_{t\in \pi..(1-\zeta)n} \sum_{m,l=1}^{T} w_m w_l \sum_{B\in B^{m,l}} |\nu(X_{t..n},B)-\rho'(B)| \leqslant \varepsilon. \qquad (3.24)$$

Note that $t - b \geqslant \zeta n$ for all $b \in 1..(\theta-\zeta)n,\ t \in \pi..(1-\zeta)n$. Therefore, we have

$$\frac{n}{t-b} \leqslant \frac{1}{\zeta}. \qquad (3.25)$$

For all $n \geqslant \frac{T}{\varepsilon\zeta},\ m \in 1..T$, $b \in 1..(\theta-\zeta)n$ and $t \in \pi..(1-\zeta)n$ we have

$$\frac{m-1}{t-b} \leqslant \frac{m}{\zeta n} \leqslant \varepsilon. \qquad (3.26)$$

Let $N := \max\{N', N'', \frac{T}{\varepsilon\zeta}\}$. By (3.22), (3.23), (3.24), (3.25) and (3.26), for all $n \geqslant N$ we have

$$\sup_{\substack{b\in 1..(\theta-\zeta)n \\ t\in \pi..(1-\zeta)n}} \sum_{m,l=1}^{T} w_m w_l \sum_{B\in B^{m,l}} |\nu(X_{b..t}, B) - \frac{\pi-b}{t-b}\rho(B) - \frac{t-\pi}{t-b}\rho'(B)| \leqslant 3\varepsilon(2+\frac{1}{\zeta}) \quad (3.27)$$

Finally, by (3.20) and (3.27) for all $n \geqslant N$ we obtain

$$\sup_{\substack{b\in 1..(\theta-\zeta)n \\ t\in \pi..(1-\zeta)n}} \widehat{d}(X_{b..t}, \frac{\pi-b}{t-b}\rho + \frac{t-\pi}{t-b}\rho') \leqslant \varepsilon(7+\frac{3}{\zeta})$$

and part (i) of Lemma 3.4.2 follows. The proof of the second part is analogous. □

Lemma 3.4.3. *Consider a sequence $\boldsymbol{x} \in \mathcal{X}^n$, $n \in \mathbb{N}$ with κ change points. Let $\boldsymbol{b} := b_1, \ldots, b_{|\boldsymbol{b}|} \in \cup_{i=1}^{n}\{1..n\}^i$, be a sequence of indices with $\min_{i\in 1..|\boldsymbol{b}|-1} b_{i+1} - b_i \geqslant \alpha n$ for some $\alpha \in (0,1)$, such that*

$$\inf_{\substack{k=1..\kappa \\ b\in\boldsymbol{b}}} |\frac{1}{n}b - \theta_k| \geqslant \zeta \quad (3.28)$$

for some $\zeta \in (0,1)$.

(i) With probability one we have

$$\lim_{n\to\infty} \inf_{k\in 1..\kappa} \Delta_{\boldsymbol{x}}(L(k), R(k)) \geqslant \delta\zeta$$

where, δ denotes the (unknown) minimum distance between the distinct distributions that generate $\boldsymbol{x}$.

(ii) Assume that we additionally have

$$[\frac{1}{n}L(k) - \alpha, \frac{1}{n}R(k) + \alpha] \subseteq [\theta_{k-1}, \theta_{k+1}] \quad (3.29)$$

where $L(k)$ and $R(k)$ are given by (3.6). With probability one we obtain

$$\lim_{n\to\infty} \sup_{k\in 1..\kappa} |\frac{1}{n}\Phi_{\boldsymbol{x}}(L(k), R(k), \alpha) - \theta_k| = 0.$$

Proof. (i). Fix some $k \in 1..\kappa$. Define $c_k := \frac{L(k)+R(k)}{2}$. Following the definition of $\Delta_{\boldsymbol{x}}(\cdot,\cdot)$ given by (3.4) we have

$$\Delta_{\boldsymbol{x}}(L(k), R(k)) := \widehat{d}(X_{L(k)..c_k}, X_{c_k..R(k)}).$$

To prove part (i) of Lemma 3.4.3, we show that for large enough n, with probability 1 we have

$$\widehat{d}(X_{L(k)..c_k}, X_{c_k..R(k)}) \geqslant \delta\zeta. \tag{3.30}$$

Let $\pi_k := \lfloor n\theta_k \rfloor$, $k = 1..\kappa$. To prove (3.30) for the case where $\pi_k \leqslant c_k$ we proceed as follows. By assumption of the lemma, we have $R(k) - L(k) \geqslant n\alpha$, so that

$$R(k) - c_k \geqslant \frac{\alpha}{2} n. \tag{3.31}$$

Fix $\varepsilon > 0$. Observe that as follows from the definition of $L(k)$ and $R(k)$, and our assumption that $\pi_k \leqslant c_k$, the segment $X_{c_k..R(k)}$ is fully generated by ρ_{k+1}. By (3.31) the condition of part (ii) of Lemma 3.4.1 hold for $X_{c_k..R(k)}$. Therefore, there exists some N_1 such that for all $n \geqslant N_1$ we have

$$\widehat{d}(X_{c_k..R(k)}, \rho_{k+1}) \leqslant \varepsilon. \tag{3.32}$$

Similarly, from (3.28) and (3.31) we have

$$\pi_{k+1} - c_k \geqslant (\zeta + \frac{\alpha}{2})n. \tag{3.33}$$

By (3.28) and (3.33), the conditions of part (i) of Lemma 3.4.2 hold for $X_{L(k)..c_k}$. Therefore, there exists some N_2 such that for all $n \geqslant N_2$ we have

$$\widehat{d}(X_{L(k)..c_k}, \frac{\pi_k - L(k)}{c_k - L(k)}\rho_k + \frac{c_k - \pi_k}{c_k - L(k)}\rho_{k+1}) \leqslant \varepsilon. \tag{3.34}$$

By (3.28) we have

$$\frac{\pi_k - L(k)}{c_k - L(k)} \geqslant \frac{\pi_k - L(k)}{n} \geqslant \zeta. \tag{3.35}$$

Moreover, we obtain

$$d(\rho_{k+1}, \frac{\pi_k - L(k)}{c_k - L(k)}\rho_k + \frac{c_k - \pi_k}{c_k - L(k)}\rho_{k+1}) = \frac{\pi_k - L(k)}{c_k - L(k)} d(\rho_{k+1}, \rho_k) \geqslant \delta\zeta \tag{3.36}$$

where the inequality follows from (3.35) and the definition of δ as the minimum distance between the distributions. Let $N := \max_{i=1,2} N_i$. For all $n \geqslant N$ we obtain

$$\begin{aligned}
\Delta_{\boldsymbol{x}}(L(k), R(k)) &= \widehat{d}(X_{L(k)..c_k}, X_{c_k..R(k)}) \\
&\geqslant \widehat{d}(X_{L(k)..c_k}, \rho_{k+1}) - \widehat{d}(X_{c_k..R(k)}, \rho_{k+1}) && (3.37) \\
&\geqslant d(\rho_{k+1}, \frac{\pi_k - L(k)}{c_k - L(k)}\rho_k + \frac{c_k - \pi_k}{c_k - L(k)}\rho_{k+1}) \\
&\quad - \widehat{d}(X_{L(k)..c_k}, \frac{\pi_k - L(k)}{c_k - L(k)}\rho_k + \frac{c_k - \pi_k}{c_k - L(k)}\rho_{k+1}) - \widehat{d}(X_{c_k..R(k)}, \rho_{k+1}) && (3.38) \\
&\geqslant d(\rho_{k+1}, \frac{\pi_k - L(k)}{c_k - L(k)}\rho_k + \frac{c_k - \pi_k}{c_k - L(k)}\rho_{k+1}) - 2\varepsilon && (3.39) \\
&\geqslant \delta\zeta - 2\varepsilon && (3.40)
\end{aligned}$$

where (3.37) and (3.38) follow from applying the triangle inequality on $\widehat{d}(\cdot,\cdot)$, (3.39) follows from (3.32) and (3.34), and (3.40) follows from (3.36). Since (3.40) holds for every $\varepsilon > 0$, this proves (3.30) in the case where $\pi_k \leqslant c_k$. The proof for the case where $\pi_k > c_k$ is analogous. Since (3.30) holds for every $k \in 1..\kappa$, part (i) of Lemma 3.4.3 follows.

(ii). Fix some $k \in 1..\kappa$. Following the definition of $\Phi_{\boldsymbol{x}}$ given by (3.5) we have

$$\Phi(L(k) - n\alpha, R(k) + n\alpha, \alpha) := \operatorname*{argmax}_{l' \in L(k)..R(k)} \widehat{d}(X_{L(k)-n\alpha..l'}, X_{l'..R(k)+n\alpha}).$$

To prove part (ii) of the lemma, it suffices to show that for every $\beta \in (0,1)$ with probability 1, for large enough n, we have

$$\widehat{d}(X_{L(k)-n\alpha..l'}, X_{l'..R(k)+n\alpha}) < \widehat{d}(X_{L(k)-n\alpha..\pi_k}, X_{\pi_k..R(k)+n\alpha}) \tag{3.41}$$

for all $l' \in L(k)..(1-\beta)\pi_k \cup \pi_k(1+\beta)..R(k)$. To prove (3.41) for $l' \in L(k)..(1-\beta)\pi_k$ we proceed as follows. Fix some $\beta \in (0,1)$ and $\varepsilon > 0$. First note that for all $l' \in$

$L(k)..(1-\beta)\pi_k$ we have

$$\frac{\pi_k - l'}{R(k) + n\alpha - l'} \geqslant \beta. \tag{3.42}$$

Note that by (3.29) the sequence $X_{L(k)-n\alpha..R(k)}$ is a subsequence of $X_{\pi_{k-1}..\pi_{k+1}}$. Consider the segment $X_{L(k)-n\alpha..R(k)}$. Observe that by (3.29) the conditions of part (ii) of Lemma 3.4.1 are satisfied by all $l' \in L(k)..R(k)$. Therefore, there exists some N_1 such that for all $n \geqslant N_1$ we have

$$\sup_{l' \in L(k)..\pi_k} \widehat{d}(X_{L(k)-n\alpha..l'}, \rho_k) \leqslant \varepsilon. \tag{3.43}$$

Similarly, consider $X_{\pi_k..R(k)+n\alpha}$. Observe that by definition of $R(k)$ we have $R(k)+n\alpha-\pi_k \geqslant n\alpha$; moreover, by (3.29) the segment is a subsequence of $X_{\pi_k..\pi_{k+1}}$. Therefore, by part (ii) of Lemma 3.4.1, there exists some N_2 such that for all $n \geqslant N_2$ we have

$$\widehat{d}(X_{\pi_k..R(k)+n\alpha}, \rho_{k+1}) \leqslant \varepsilon. \tag{3.44}$$

By (3.29), there is a single change point π_k within $X_{L(k)-n\alpha..R(k)+\alpha}$. Therefore, every $l' \in L(k)..\pi_k$ has a linear distance from π_{k-1}, i.e. $l' - \pi_{k-1} \geqslant \alpha n$ for all $l' \in L(k)..\pi_k$. On the other hand, $R(k) + n\alpha \in \pi_k + n\alpha..\pi_{k+1}$. Therefore by part (ii) of Lemma 3.4.2 there exists some N_3 such that

$$\sup_{l' \in L(k)..\pi_k} \widehat{d}(X_{l'..R(k)+n\alpha}, \frac{\pi_k - l'}{R(k) + n\alpha - l'}\rho_k + \frac{R(k) + n\alpha - \pi_k}{R(k) + n\alpha - l'}\rho_{k+1}) \leqslant \varepsilon. \tag{3.45}$$

Let $N := \max\limits_{i=1..3} N_i$. By (3.43), (3.44) and the subsequent application of the triangle inequality on $\widehat{d}(\cdot,\cdot)$ for all $n \geqslant N$ we obtain

$$\begin{aligned}
\widehat{d}(X_{L(k)-n\alpha..\pi_k}, X_{\pi_k..R(k)+n\alpha}) &\geqslant \widehat{d}(X_{L(k)-n\alpha..\pi_k}, \rho_{k+1}) - \widehat{d}(X_{\pi_k..R(k)+n\alpha}, \rho_{k+1}) \\
&\geqslant \widehat{d}(X_{L(k)-n\alpha..\pi_k}, \rho_{k+1}) - \varepsilon \\
&\geqslant d(\rho_k, \rho_{k+1}) - \widehat{d}(\rho_k, X_{L(k)-n\alpha..\pi_k}) - \varepsilon \\
&\geqslant d(\rho_k, \rho_{k+1}) - 2\varepsilon.
\end{aligned} \tag{3.46}$$

By applying the triangle inequality on $\widehat{d}(\cdot,\cdot)$, for all $n \geqslant N$ we obtain

$$\begin{aligned}
&\sup_{l' \in L(k)..(1-\beta)\pi_k} \widehat{d}(X_{L(k)-n\alpha..l'}, X_{l'..R(k)+n\alpha}) \\
\leqslant &\sup_{l' \in L(k)..(1-\beta)\pi_k} \widehat{d}(X_{L(k)-n\alpha..l'}, \rho_k) + \widehat{d}(\rho_k, X_{l'..R(k)+n\alpha}) \\
\leqslant &\sup_{l' \in L(k)..(1-\beta)\pi_k} \widehat{d}(\rho_k, X_{l'..R(k)+n\alpha}) + \varepsilon \qquad (3.47) \\
\leqslant &\sup_{l' \in L(k)..(1-\beta)\pi_k} d(\rho_k, \frac{\pi_k - l'}{R(k)+n\alpha-l'}\rho_k + \frac{R(k)+n\alpha-\pi_k}{R(k)+n\alpha-l'}\rho_{k+1}) \\
&\qquad + d(X_{l'..R(k)+n\alpha}, \frac{\pi_k - l'}{R(k)+n\alpha-l'}\rho_k + \frac{R(k)+n\alpha-\pi_k}{R(k)+n\alpha-l'}\rho_{k+1}) + \varepsilon \\
\leqslant &\sup_{l' \in L(k)..(1-\beta)\pi_k} d(\rho_k, \frac{\pi_k - l'}{R(k)+n\alpha-l'}\rho_k + \frac{R(k)+n\alpha-\pi_k}{R(k)+n\alpha-l'}\rho_{k+1}) + 2\varepsilon \qquad (3.48)
\end{aligned}$$

where (3.47) follows from (3.43), and (3.48) follows from (3.45). We also have

$$\begin{aligned}
&d(\rho_k, \rho_{k+1}) - d(\rho_k, \frac{\pi_k - l'}{R(k)+n\alpha-l'}\rho_k + \frac{R(k)+n\alpha-\pi_k}{R(k)+n\alpha-l'}\rho_{k+1}) \\
&\qquad = \frac{\pi_k - l'}{R(k)+n\alpha-l'} d(\rho_k, \rho_{k+1}) \qquad (3.49) \\
&\qquad \geqslant \beta\delta. \qquad (3.50)
\end{aligned}$$

where the inequality follows from (3.42) and the definition of δ as the minimum distance between the distributions that generate the data. Finally, from (3.46), (3.48) and (3.49) for all $n \geqslant N$ we obtain,

$$\inf_{l' \in L(k)..(1-\beta)\pi_k} \widehat{d}(X_{L(k)-n\alpha..\pi_k}, X_{\pi_k..R(k)+n\alpha}) - \widehat{d}(X_{L(k)-n\alpha..l'}, X_{l'..R(k)+n\alpha}) \geqslant \beta\delta - 4\varepsilon. \qquad (3.51)$$

Since (3.51) holds for every $\varepsilon > 0$, this proves (3.41) for $l' \in L(k)..(1-\beta)\pi_k$. The proof for the case where $l' \in (1+\beta)\pi_k..R(k)$ is analogous. Since (3.41) holds for every $k \in 1..\kappa$, part (ii) follows. □

For the next lemma and Theorem 3.3.4 we need some extra notation. Consider the set $\mathcal{S}$ of segments specified by Line (5) in Algorithm 3. For every segment $\widetilde{\boldsymbol{x}}_i :=$

$X_{\psi_{i-1}..\psi_i} \in \mathcal{S}$ where $i = 1..|\Upsilon| + 1$ define $\widetilde{\rho}_i$ as the process distribution that generates the largest portion of $\widetilde{x}_i$. That is, let $\widetilde{\rho}_i := \rho_j$ where j is such that $K \in \mathcal{G}_j$ with

$$K := \underset{k \in \mathcal{G}_{r'}}{\operatorname{argmax}} \ |\{\psi_{i-1} + 1, \ldots, \psi_i\} \cap \{n\theta_{k-1} + 1, \ldots, n\theta_k\}|$$

and $\mathcal{G}_j,\ j = 1..r$ are the ground-truth partitions defined by (3.1).

Lemma 3.4.4. *Let $\boldsymbol{x} \in \mathcal{X}^n,\ n \in \mathbb{N}$ be a sequence with κ change points at least $\lambda_{\min}$ apart for some $\lambda_{\min} \in (0, 1)$. Assume that the distributions that generate $\boldsymbol{x}$ are stationary and ergodic. Let $\mathcal{S}$ be the set of segments specified by (5) in Algorithm 3. For all $\lambda \in (0, \lambda_{\min})$ with probability one we have*

$$\lim_{n \to \infty} \sup_{\boldsymbol{x}_i \in \mathcal{S}} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\rho}_i) = 0.$$

Proof. Fix an $\varepsilon \in (0, \lambda/2)$. For simplicity of notation define $\pi_k := \lfloor n\theta_k \rfloor,\ k = 1..\kappa$. Since the initial list Υ of change point candidates are produced by a *consistent* list-estimator $\Upsilon(\boldsymbol{x}, \lambda)$, (see Definition 3.2.1), there exists an index-set

$$\mathcal{I} := \{\mu_1, \ldots, \mu_\kappa\} \in \{1..|\Upsilon|\}^\kappa$$

and some N_0 such that for all $n \geqslant N_0$ we have

$$\sup_{k=1..\kappa} \frac{1}{n} |\psi_{\mu_k} - \pi_k| \leqslant \varepsilon. \tag{3.52}$$

Moreover, the initial candidates are at least $n\lambda$ apart so that

$$\inf_{i \in 1..|\Upsilon|+1} \psi_i - \psi_{i-1} \geqslant n\lambda \tag{3.53}$$

where $\psi_0 := 0$ and $\psi_{|\Upsilon|+1} := n$. Let $\mathcal{I}' := \{1..|\Upsilon|\} \setminus \mathcal{I}$. Denote by $\mathcal{S}_1 := \{\widetilde{x}_i := X_{\psi_{i-1}+1..\psi_i} \in \mathcal{S} : \{i, i-1\} \cap \mathcal{I} = \varnothing\}$ the subset of the segments in $\mathcal{S}$ whose elements are formed by joining pairs of *consecutive elements* of $\mathcal{I}'$ and let $\mathcal{S}_2 := \mathcal{S} \setminus \mathcal{S}_1$ be its complement. Let the true change points that appear immediately to the left and to the

right of an index $j \in 1..n-1$ be given by

$$\mathcal{L}(j) := \max_{k \in 0..\kappa+1} \pi_k \leqslant j \text{ and } \mathcal{R}(j) := \min_{k \in 0..\kappa+1} \pi_k > j$$

respectively, with $\pi_0 := 0$, $\pi_{\kappa+1} := n$ where equality occurs when j is itself a change point. We have two cases.

1. Consider $\widetilde{x}_i := X_{\psi_{i-1}+1..\psi_i} \in \mathcal{S}_1$. Observe that, by definition $\widetilde{x}_i$ cannot contain a true change point for $n \geqslant N_0$ since otherwise, either $i-1$ or i would belong to $\mathcal{I}$ contradicting the assumption that $\widetilde{x}_i \in \mathcal{S}_1$. Therefore, for all $n \geqslant N_0$ we have $\widetilde{\rho}_i = \rho$ where $\rho \in \{\rho_1, \ldots, \rho_r\}$ is the process distribution that generates $X_{\mathcal{L}(\psi_{i-1})..\mathcal{R}(\psi_{i-1})}$. By (3.53), the conditions of part (ii) of Lemma 3.4.1 are satisfied by ψ_{i-1} and ψ_i. Therefore, there exists some $N_i \geqslant N_0$ such that for all $n \geqslant N_i$ we have

$$\widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\rho}_i) := \widehat{d}(X_{\psi_{i-1}..\psi_i}, \rho) \leqslant \varepsilon. \tag{3.54}$$

Let $N' := \max_{i \text{ s.t. } \widetilde{\boldsymbol{x}}_i \in \mathcal{S}_1} N_i$. For all $n \geqslant N'$ we have

$$\sup_{\widetilde{x}_i \in \mathcal{S}_1} \widehat{d}(\widetilde{x}_i, \widetilde{\rho}_i) \leqslant \varepsilon. \tag{3.55}$$

2. Take $\widetilde{x}_i := X_{\psi_{i-1}..\psi_i} \in \mathcal{S}_2$. Observe that, by definition $\mathcal{I} \cap \{i, i-1\} \neq \varnothing$ so that either $i-1$ or i belong to $\mathcal{I}$. We prove the statement for the case where $i-1 \in \mathcal{I}$. The case where $i \in \mathcal{I}$ is analogous. We start by showing that $[\psi_{i-1}, \psi_i] \subseteq [\pi - \varepsilon n, \pi' + \varepsilon n]$ for all $n \geqslant N_0$ where

$$\pi := \operatorname*{argmin}_{\pi_k, k=1..\kappa} \frac{1}{n} |\pi_k - \psi_{i-1}| \text{ and } \pi' := \mathcal{R}(\pi).$$

Since $i-1 \in \mathcal{I}$, by (3.52) for all $n \geqslant N_0$ we have $\frac{1}{n}|\pi - \psi_{i-1}| \leqslant \varepsilon$. We have two cases. Either $i \in \mathcal{I}$ so that by (3.52) for all $n \geqslant N_0$ we have $\frac{1}{n}|\psi_i - \pi'| \leqslant \varepsilon$, or $i \in \mathcal{I}'$ in which case $\psi_i < \pi'$. To see the latter statement assume by way of contradiction that $\psi_i > \pi'$ where $\pi' \neq n$; (the statement trivially holds for $\pi' = n$). By the consistency of $\Upsilon(\boldsymbol{x}, \lambda)$ there exists some $j > i-1 \in \mathcal{I}$ such that $\frac{1}{n}|\psi_j - \pi'| \leqslant \varepsilon$ for all $n \geqslant N_0$. Moreover, by (3.52) and (3.53) for all $n \geqslant N_0$ the candidates indexed by $\mathcal{I}'$ have linear distances

from the true change points:

$$\begin{aligned}\inf_{\substack{k\in 1..\kappa\\ i\in\mathcal{I}'}}|\pi_k-\psi_i| &\geqslant \inf_{\substack{k\in 1..\kappa\\ i\in\mathcal{I}',j\in\mathcal{I}}}|\psi_i-\psi_j|-|\pi_k-\psi_j|\\ &\geqslant n(\lambda-\varepsilon)\end{aligned}\tag{3.56}$$

Thus, from (3.52) and (3.56) we obtain that $\psi_i-\psi_j\geqslant\lambda-2\varepsilon>0$. Since the initial estimates are sorted in increasing order, this implies $j\leqslant i$ leading to a contradiction. Thus we have

$$[\psi_{i-1},\psi_i]\subseteq[\pi-\varepsilon n,\pi'+\varepsilon n]\tag{3.57}$$

Therefore, $\widetilde{\rho}_i=\rho$ where ρ is the process distribution $\rho\in\{\rho_1,\ldots,\rho_r\}$ that generates $X_{\pi..\pi'}$. To show that $\widehat{d}(\widetilde{x}_i,\rho)\leqslant\varepsilon$ we proceed as follows. There exists some T such that

$$\sum_{m,l=T}^{\infty}w_mw_l\leqslant\varepsilon.\tag{3.58}$$

It is easy to see that by (3.56), and the assumption that $\lambda\in(0,\lambda_{\min}]$, (where $\lambda_{\min}$ is given by (3.2)), the segment $X_{\pi..\min\{\psi_i,\pi'\}}$ has length at least $n\lambda$, i.e.

$$\min\{\psi_i,\pi'\}-\pi\geqslant n\lambda.\tag{3.59}$$

By (3.57) and (3.59) the conditions of part (i) of Lemma 3.4.1 are satisfied by π and $\min\{\psi_i,\pi'\}$. Therefore, there exists some $N_i\geqslant N_0$ such that for all $n\geqslant N_i$ we have

$$\sum_{m,l=1}^{T}w_mw_l\sum_{B\in B^{m,l}}|\nu(X_{\pi..\min\{\psi_i,\pi'\}},B)-\rho(B)|\leqslant\varepsilon.\tag{3.60}$$

Using the definition of $\nu(\cdot,\cdot)$ given by (2.1), for every $B\in B^{m,l},\ m,l\in 1..T$ we have

$$\begin{aligned}(1-\frac{m-1}{\psi_i-\psi_{i-1}})|\nu(\widetilde{x}_i,B)-\rho(B)|\leqslant&\frac{\min\{\psi_i,\pi'\}-\pi-m+1}{\psi_i-\psi_{i-1}}|\nu(X_{\pi..\min\{\psi_i,\pi'\}},B)-\rho(B)|\\ &+\frac{\mathbb{I}\{\psi_i\geqslant\pi'\}(\psi_i-\pi')}{\psi_i-\psi_{i-1}}+\frac{|\psi_i-\pi|}{\psi_i-\psi_{i-1}}\end{aligned}\tag{3.61}$$

Take $n \geqslant N_i$. Increase N_i if necessary to have

$$\frac{T}{N_i \lambda} \leqslant \varepsilon. \tag{3.62}$$

Recall that $\sum_{m,l=1}^{n} w_m w_l \leqslant 1$. For all $n \geqslant N_i$ we have

$$\widehat{d}(\widetilde{x}_i, \widetilde{\rho}_i) \leqslant \sum_{m,l=1}^{T} w_{m,l} \sum_{B \in B^{m,l}} (1 - \frac{m-1}{\psi_i - \psi_{i-1}}) |\nu(\widetilde{\boldsymbol{x}}_i, B) - \rho(B)| + \frac{m-1}{\psi_i - \psi_{i-1}} + \varepsilon \tag{3.63}$$

$$\leqslant \sum_{m,l=1}^{T} w_{m,l} \sum_{B \in B^{m,l}} (1 - \frac{m-1}{\psi_i - \psi_{i-1}}) |\nu(\widetilde{\boldsymbol{x}}_i, B) - \rho(B)| + 2\varepsilon \tag{3.64}$$

$$\leqslant \sum_{m,l=1}^{T} w_{m,l} \sum_{B \in B^{m,l}} \frac{1}{\lambda} |\nu(X_{\pi .. \min\{\psi_i, \pi'\}}, B) - \rho(B)| + 2\varepsilon(1 + 1/\lambda) \tag{3.65}$$

$$\leqslant 3\varepsilon(1 + 1/\lambda) \tag{3.66}$$

where, (3.63) follows form (3.58) and the fact that $|\nu(\cdot,\cdot) - \rho(\cdot)| \leqslant 1$, (3.64) follows from (3.62) and (3.53), (3.65) follows from (3.52), (3.53), and (3.61), and (3.66) follows from (3.60). Let $N'' := \max_{i \text{ s.t. } \widetilde{\boldsymbol{x}}_i \in \mathcal{S}_2}$. By (3.66) for all $n \geqslant N''$ we have

$$\sup_{\widetilde{\boldsymbol{x}}_i \in \mathcal{S}_2} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\rho}_i) \leqslant 3\varepsilon(1 + 1/\lambda). \tag{3.67}$$

Finally, by (3.55) and (3.67) for all $n \geqslant \max\{N', N''\}$ we have

$$\sup_{\widetilde{\boldsymbol{x}}_i \in \mathcal{S}} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\rho}_i) \leqslant 3\varepsilon(1 + 1/\lambda).$$

Since the choice of ε is arbitrary, this proves the statement. □

3.4.2 Proof of Theorem 3.3.2

Proof. On each iteration $j \in 1..\log n$ the algorithm produces a set of estimated change points. We show that on some iterations these estimates are consistent, and that estimates produced on the rest of the iterations are negligible. To this end, we partition the set of iterations into three sets as described below.

First recall that for every $j \in 1..\log n$ and $t \in 1..\kappa + 1$ the algorithm generates a

grid of boundaries $b_i^{t,j}$ that are $n\alpha_j$ apart i.e. for all $j \in 1..\log n$ and $t \in 1..\kappa+1$ we have

$$b_i^{t,j} - b_{i-1}^{t,j} = n\alpha_j,\ i = 0..\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor \tag{3.68}$$

Therefore, the segments $X_{b_i^{t,j} - b_{i-1}^{t,j}}$ have lengths that are linear functions of n. More specifically, For $j = 1..\log n$ and $t \in 1..\kappa+1$ define

$$\zeta(t,j) := \min_{\substack{k \in 1..\kappa \\ i \in 0..\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor}} |\alpha_j(i + \frac{1}{t+1}) - \theta_k| \tag{3.69}$$

(Note that $\zeta(t,j)$ can also be zero.) For all $i = 0..\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor$ we have

$$|b_i^{t,j} - \pi_k| \geqslant n\zeta(t,j) \tag{3.70}$$

(This linearity condition is a requirement by some of the lemmas which will be applied to appropriate segments of the grid as part of the arguments in the proof.)

Fix $\varepsilon > 0$. We partition the set of iterations on $j \in 1..\log n$, and consider each cases separately.

Step 1. There exists some J_ε such that

$$\sum_{j=J_\varepsilon}^{\infty} w_j \leqslant \varepsilon \tag{3.71}$$

This first subset of the set of iterations $j = 1..\log n$ corresponds to the higher iterations where λ_j is too small. In this case the resulting grids are too fine, and the segments may not be long enough for the estimates to be consistent. These iterations are penalized by small weights w_j, so that the corresponding candidate estimates become negligible.

Step 2. The second subset corresponds to the iterations where **a.** $\lambda_j \in (0, \lambda_{\min}]$ *and* **b.** the segments are long enough for the candidate change point parameter estimates to be consistent.

Let $J(\lambda_{\min}) := -\log(\lambda_{\min}/3)$ where $\lambda_{\min}$ defined by (3.2) specifies the minimum separation of the change points. For all $j \geqslant J(\lambda_{\min})$ we have $\alpha_j \leqslant \lambda_j/3$. Therefore, at every iteration on $j \geqslant J(\lambda_{\min})$ and $t \in 1..\kappa+1$, for every change point $\theta_k,\ k \in 1..\kappa$ we have

$$[\frac{1}{n}L(k) - \alpha_j, \frac{1}{n}R(k) + \alpha_j] \subseteq [\theta_{k-1}, \theta_{k+1}] \tag{3.72}$$

where $L(\cdot)$ and $R(\cdot)$ are defined by (3.6). We further partition the set of iterations on $t \in 1..\kappa+1$ into two subsets as follows. For every fixed $j \in J(\lambda_{\min})..J_\varepsilon$ we identify a subset $\mathcal{T}(j)$ of the iterations on $t = 1..\kappa+1$ at which the change point parameters $\theta_k,\ k = 1..\kappa$ are estimated consistently and the performance scores $\gamma(t,j)$ are bounded below by a nonzero constant. Moreover, we show that if the set $\mathcal{T}'(j) := \{1..\kappa+1\}\backslash\mathcal{T}(j)$ is nonempty, the performance scores $\gamma(t,j)$ for all $j \in J(\lambda_{\min})..J_\varepsilon$ and $t \in \mathcal{T}'(j)$ are arbitrarily small.

i. To define $\mathcal{T}(j)$ we proceed as follows. For every $\theta_k,\ k = 1..\kappa$ we can uniquely define $q_k \in \mathbb{N}$ and $p_k \in [0, \alpha_j)$ so that $\theta_k = q_k\alpha_j + p_k$. Therefore, for any $p \in [0,\alpha_j)$ with $p \neq p_k,\ k = 1..\kappa$, we have $\inf_{\substack{k=1..\kappa \\ i\in\mathbb{N}\cup\{0\}}} |i\alpha_j + p - \theta_k| > 0$. Observe that we can only have κ distinct residues $p_k,\ k = 1..\kappa$. Therefore, any subset of $[0,\alpha_j)$ with $\kappa+1$ elements, contains at least one element p' such that $p' \neq p_k,\ k = 1..\kappa$. It follows that for every $j \in J(\lambda_{\min})..J_\varepsilon$ there exists at least one $t \in 1..\kappa+1$ such that $\zeta(t,j) > 0$. For every $j \in J(\lambda_{\min})..J_\varepsilon$, define

$$\mathcal{T}(j) := \{t \in 1..\kappa+1 : \zeta(t,j) > 0\}$$

Let $\bar{\zeta}(j) := \min_{t\in\mathcal{T}(j)} \zeta(t,j)$ and define $\zeta_{\min} := \inf_{j\in J(\lambda_{\min})..J_\varepsilon} \bar{\zeta}(j)$. Note that $\zeta_{\min} > 0$. By (3.70), (3.72) and hence part (i) of Lemma 3.4.3, for every $j \in J(\lambda_{\min})..J_\varepsilon$ there exists some $N_1(j)$ such that for all $n \geqslant N_1(j)$ we have

$$\inf_{t\in\mathcal{T}(j)} \gamma(t,j) \geqslant \delta\bar{\zeta}(j) \tag{3.73}$$

where δ denotes the (unknown) minimum distance between the distinct distributions that generate the data. Recall that, as specified by Algorithm 1 we have

$$\eta := \sum_{j=1}^{\log n}\sum_{t=1}^{\kappa+1} w_j\gamma(t,j).$$

Hence by (3.73) for all $n \geqslant N$ we have

$$\eta \geqslant w_{J(\lambda_{\min})}\delta\bar{\zeta}(J_{\lambda_{\min}}) \tag{3.74}$$

Moreover, by part (ii) of Lemma 3.4.3, there exists some $N_2(j)$ such that for all

$n \geqslant N_2(j)$ we have

$$\sup_{\substack{k\in 1..\kappa \\ t\in 1..\mathcal{T}(j)}} \frac{1}{n}|\widehat{\pi}_k^{t,j} - \pi_k| \leqslant \varepsilon \tag{3.75}$$

ii. Define $\mathcal{T}'(j) := \{1..\kappa+1\} \setminus \mathcal{T}(j)$ for $j \in J(\lambda_{\min})..J_\varepsilon$. It may be possible for the set $\mathcal{T}'(j)$ to be nonempty on some iterations on $j \in J(\lambda_{\min})..J_\varepsilon$. Without loss of generality, define $\gamma(t,j) := 0$ for all $j \in J(\lambda_{\min})..J_\varepsilon$ with $\mathcal{T}'(j) = \varnothing$. Observe that by definition, for all $j \in J(\lambda_{\min})..J_\varepsilon$ such that $\mathcal{T}'(j) \neq \varnothing$, we have $\max_{t\in\mathcal{T}'(j)} \zeta(t,j) = 0$ where $\zeta(t,j)$ is given by (3.69). This means that on each of these iterations, there exists some π_k for some $k \in 1..\kappa$ such that $\pi_k = b_i^{t,j}$ for some $i \in \lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor$. Since $\lambda_j \leqslant \lambda_{\min}$ for all $j \in J(\lambda_{\min})..J_\varepsilon$, we have $\pi_k..\pi_k + n\lambda_j \subseteq \pi_k..\pi_{k+1}$ and $\pi_k - n\lambda_j \subseteq \pi_{k-1}..\pi_k$. Therefore, by part (iii) of Lemma 3.4.1 respectively, there exists some $N_3(j)$ such that for all $n \geqslant N_3(j)$ we have, $\max\{\Delta_{\boldsymbol{x}}(\pi_k - n\lambda_j, \pi_k), \Delta_{\boldsymbol{x}}(\pi_k, \pi_k + n\lambda_j)\} \leqslant \varepsilon$. Thus, for every $j \in J(\lambda_{\min})..J_\varepsilon$ and all $n \geqslant N_3(j)$ we have

$$\sup_{t\in\mathcal{T}'(j)} \gamma(t,j) \leqslant \varepsilon. \tag{3.76}$$

Step 3. Consider the set of iterations, $j = 1..J(\lambda_{\min}) - 1$. Recall that it is desired for a grid to be such that every three consecutive segments contain at most one change point. This property is not satisfied for $j = 1..J(\lambda_{\min}) - 1$ since by definition on these iterations we have $\alpha_j > \lambda_j/3$. We show that for all these iterations, the performance score $\gamma(t,j)$, $1..\kappa+1$ becomes arbitrarily small. For all $j = 1..J(\lambda_{\min})-1$ and $t = 1..\kappa+1$, define the set of intervals

$$\mathcal{S}^{t,j} := \{(b_i^{t,j}, b_{i+3}^{t,j}) : i = 0..\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor - 3\}$$

and consider its partitioning into $\mathcal{S}_l^{t,j} := \{(b_{l+3i}^{t,j}, b_{l+3(i+1)}^{t,j}) : i = 0..\frac{1}{3}(\lfloor \frac{1}{\alpha_j} - \frac{1}{t+1} \rfloor - l)\}$, $l = 0..2$. Observe that, by construction for every fixed $l = 0..2$, every pair of indices $(b, b') \in \mathcal{S}_l^{t,j}$ specifies a segment $X_{b..b'}$ of length $3n\alpha_j$ and the elements of $\mathcal{S}_l^{t,j}$ index non-overlapping segments of $\boldsymbol{x}$. Since for all $j = 1..J(\lambda_{\min}) - 1$ we have $\alpha_j > \lambda_j/3$, at every iteration on $j \in 1..J(\lambda_{\min})-1$ and $t \in 1..\kappa+1$, there exists some $(b, b') \in \mathcal{S}^{t,j}$ such that the segment $X_{b..b'}$ contains more than one change point. Since there are exactly κ change points, in at least one of the partitions $\mathcal{S}_l^{t,j}$ for some $l \in 0..2$ we have that within any set of κ segments indexed by a subset of κ elements of $\mathcal{S}_l^{t,j}$, there exists at least one

segment that contains no change points. Therefore, by (3.68), (3.70) and hence part (iii) of Lemma 3.4.1, for every $j \in 1..J(\lambda_{\min}) - 1$ there exists some $N(j)$ such that for all $n \geqslant N(j)$ we have

$$\sup_{t \in 1..\kappa+1} \gamma(t, j) \leqslant \varepsilon. \tag{3.77}$$

Let $N' := \max_{j=1..J(\lambda_{\min})-1} N(j)$ and $N'' := \max_{\substack{i=1..3 \\ j=J(\lambda_{\min})..J_\varepsilon}} N_i(j)$. Define $N := \max\{N', N''\}$. By (3.71), (3.74) and that $\gamma(\cdot, \cdot) \leqslant 1$ for all $n \geqslant N$ we have

$$\frac{1}{n\eta} \sum_{j=J_\varepsilon}^{\log n} \sum_{t=1}^{\kappa+1} w_j \gamma(t, j) |\pi_k - \widehat{\pi}_k^{t,j}| \leqslant \frac{\varepsilon(\kappa + 1)}{w_{J(\lambda_{\min})} \delta \bar{\zeta}(J(\lambda_{\min}))} \tag{3.78}$$

Recall that by definition we have $\eta := \sum_{j=1}^{\log n} \sum_{t=1}^{\kappa+1} w_j \gamma(t, j)$ as follows from (3.74) is nonzero. Therefore we have

$$\frac{1}{\eta} \sum_{j=J(\lambda_{\min})}^{J_\varepsilon} \sum_{t \in \mathcal{T}(j)} w_j \gamma(t, j) \leqslant 1. \tag{3.79}$$

By (3.75) and (3.79) for all $n \geqslant N$ we have

$$\frac{1}{n\eta} \sum_{j=J(\lambda_{\min})}^{J_\varepsilon} \sum_{t \in \mathcal{T}(j)} w_j \gamma(t, j) |\pi_k - \widehat{\pi}_k^{t,j}| \leqslant \varepsilon. \tag{3.80}$$

Note that $|\pi_k^{t,j} - \widehat{\pi}_k^{t,j}| \leqslant n$ and that $\sum_{j=1}^{J(\lambda_{\min})} w_j \leqslant 1$. Therefore, by (3.74) and (3.76) for all $n \geqslant N$ we obtain

$$\frac{1}{n\eta} \sum_{j=J_\varepsilon}^{\log n} \sum_{t \in \mathcal{T}'(j)} w_j \gamma(t, j) |\pi_k - \widehat{\pi}_k^{t,j}| \leqslant \frac{\varepsilon(\kappa + 1)}{w_{J(\lambda_{\min})} \delta \bar{\zeta}(J(\lambda_{\min}))}. \tag{3.81}$$

Similarly, from (3.74) and (3.77) we obtain

$$\frac{1}{n\eta} \sum_{j=1}^{J(\lambda_{\min})-1} \sum_{t=1}^{\kappa+1} w_j \gamma(t, j) |\pi_k - \widehat{\pi}_k^{t,j}| \leqslant \frac{\varepsilon(\kappa + 1)}{w_{J(\lambda_{\min})} \delta \bar{\zeta}(J(\lambda_{\min}))} \tag{3.82}$$

Let $\widehat{\theta}_k(n) := \frac{\widehat{\pi}_k}{n}, k = 1..\kappa$. By (3.78), (3.80), (3.82) and (3.81) we have

$$\begin{aligned}
|\widehat{\theta}_k(n) - \theta_k| \leqslant \frac{1}{n\eta} & \sum_{j=1}^{J(\lambda_{\min})-1} \sum_{t=1}^{\kappa+1} w_j \gamma(t,j) |\pi_k - \widehat{\pi}_k^{t,j}| \\
& + \frac{1}{n\eta} \sum_{j=J(\lambda_{\min})}^{J_\varepsilon} \sum_{t \in \mathcal{T}(j)} w_j \gamma(t,j) |\pi_k - \widehat{\pi}_k^{t,j}| \\
& + \frac{1}{n\eta} \sum_{j=J(\lambda_{\min})}^{J_\varepsilon} \sum_{t \in \mathcal{T}'(j)} w_j \gamma(t,j) |\pi_k - \widehat{\pi}_k^{t,j}| \\
& + \frac{1}{n\eta} \sum_{j=J_\varepsilon}^{\log n} \sum_{t=1}^{\kappa+1} w_j \gamma(t,j) |\pi_k - \widehat{\pi}_k^{t,j}| \\
\leqslant \varepsilon & (1 + \frac{3(\kappa+1)}{w_{J(\lambda_{\min})} \delta \bar{\zeta}(J(\lambda_{\min}))}).
\end{aligned}$$

Since the choice of ε is arbitrary, the statement of the theorem follows. □

3.4.3 Proof of Theorem 3.3.3

Proof. The outline of the proof is as follows. **1.** we identify a subset $\mathcal{I} \subseteq \{(t,i) : t \in 1..2,\ i \in 1..\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor\}$ of index pairs corresponding to the boundaries of b_i^t that in turn, specify segments in $\boldsymbol{x}$, each of which contains a single change point. We show that for large enough n, the change points within the segments specified by $\mathcal{I}$ are estimated consistently. Moreover, for large enough n, the scores $s(\cdot,\cdot)$, assigned to the consistent estimates are bounded below by a non-zero constant. **2.** The performance scores assigned to the segments specified by the rest of the segments converge to zero. Therefore, once sorted in decreasing order of score, the top elements of the list, correspond to the consistent estimates. **3.** Finally, we show that the corresponding estimate of every change point parameter appears exactly once in the list. That is, we show that every change point parameter is estimated at least once, and the potential duplicate estimates are filtered. Therefore, for large enough n, the first κ elements of the list are estimates of the true change point parameters.

Fix an $\varepsilon > 0$. Observe that for every fixed $t = 1, 2$ we have

$$b_i^t - b_{i-1}^t = n\alpha,\ i = 1..\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor \tag{3.83}$$

where $b_i^t,\ i = 1...\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor,\ t = 1, 2$ denote the boundaries specified by Line (3) in Algorithm 2. Define

$$\zeta(t, i) := \min_{k \in 1..\kappa} |\alpha(i + \frac{1}{t+1}) - \theta_k|,\ i \in 0..\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor,\ t \in 1..2$$

and note that $\zeta(t, i)$ can also be zero. For all $i \in 0..1/\alpha,\ t = 1, 2$ and $k \in 1..\kappa$ we have

$$|\frac{1}{n} b_i^t - \theta_k| \geqslant \zeta(t, i). \tag{3.84}$$

Step 1. Define $\mathcal{I} := \cup_{k \in 1..\kappa} \mathcal{I}_k$ where

$$\mathcal{I}_k := \{(t, i) : t \in 1, 2,\ i \in 1...\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor \text{ s.t. } \theta_k \in (\frac{1}{n} b_i^t, \frac{1}{n} b_{i+1}^t)\}$$

for $k \in 1..\kappa$. Since $\lambda \leqslant \lambda_{\min}$, we have $\alpha \in (0, \lambda_{\min}/3]$. Therefore, for every $t = 1, 2$ and $k \in 1..\kappa$ we have

$$[\frac{1}{n} L(k) - \alpha, \frac{1}{n} R(k) + \alpha] \subseteq [\theta_{k-1}, \theta_{k+1}] \tag{3.85}$$

where $L(\cdot)$ and $R(\cdot)$ are given by (3.6). Define $\zeta_{\min} := \min_{(t,i) \in \mathcal{I}} \zeta(t, i)$. From the definition of $\mathcal{I}$ it follows that

$$\zeta_{\min} > 0. \tag{3.86}$$

Let δ denote the minimum distance between the distributions that generate $\boldsymbol{x}$, i.e.

$$\delta := \min_{i \neq j \in 1..r} d(\rho_i, \rho_j).$$

As follows from (3.83), (3.84), and (3.86), the conditions of the first part of Lemma 3.4.3 are satisfied by b_i^t and $b_{i+1}^t,\ i \in \mathcal{I}$. Therefore, by part (i) of Lemma 3.4.3, there exists some N_1 such that for all $n \geqslant N_1$ we have

$$\inf_{(t,i) \in \mathcal{I}} s(t, i) \geqslant \delta \zeta_{\min} \tag{3.87}$$

where as specified by Line (4) in Algorithm 2, $s(t, i)$ is the performance score calculated as the intra-subsequence distance $\Delta_{\boldsymbol{x}}(b_i^t, b_{i+1}^t)$ of the segment $X_{b_i^t..b_{i+1}^t}$. Define

$$\theta(t, i) := \theta_k,\ k \in 1..\kappa \text{ s.t. } \theta_k \in (\frac{1}{n} b_i^t, \frac{1}{n} b_{i+1}^t)$$

and let $\widehat{\theta}_i^t(n) := \frac{1}{n}\widehat{\pi}_i^t$. By (3.83), (3.84), (3.85), and (3.86), the conditions of the second part of Lemma 3.4.3 are satisfied for b_i^t and b_{i+1}^t for all $i \in \mathcal{I}$. Therefore, by part (ii) of Lemma 3.4.3, there exists some N_2 such that for all $n \geqslant N_2$ we have

$$\sup_{(t,i)\in\mathcal{I}} |\widehat{\theta}_i^t(n) - \theta(t,i)| \leqslant \varepsilon. \tag{3.88}$$

Step 2. Define $\mathcal{I}' := \{1,2\} \times \{1..\lfloor \frac{1}{\alpha} - \frac{1}{t+1} \rfloor\} \setminus \mathcal{I}$. Observe that for all $(t,i) \in \mathcal{I}'$, the segment $X_{b_i^t..b_{i+1}^t}$ has no change points, so that for some $k \in 1..\kappa$ we have

$$b_i^t..b_{i+1}^t \subseteq \lfloor n\theta_k \rfloor..\lfloor n\theta_{k+1} \rfloor. \tag{3.89}$$

By (3.83), (3.84) and (3.89), the conditions of part (iii) of Lemma 3.4.1 are satisfied by b_i^t and b_{i+1}^t, $i \in \mathcal{I}'$. Therefore, by part (iii) of Lemma 3.4.1, there exists some N_3 such that for all $n \geqslant N_3$ we have

$$\sup_{(t,i)\in\mathcal{I}'} s(t,i) \leqslant \varepsilon \tag{3.90}$$

Step 3. It remains to see that the corresponding estimate of every change point appears exactly once in the output. Let $N := \max_{i=1..3} N_i$. Recall that the algorithm uses a standard greedy approach to produce the sorted list $\widehat{\boldsymbol{\theta}}$ of estimates. Starting from the set of all candidate estimates, an available estimate of highest score is added to the list, and the candidate estimates within a radius of $\lambda n/2$ from the selected estimate, are made unavailable. The process continues until no candidate change points are available. By (3.86), (3.87) and (3.90), for all $n \geqslant N$ the segments $X_{b_i^t..b_{i+1}^t}$, $(t,i) \in \mathcal{I}$ are assigned higher scores than $X_{b_i^t..b_{i+1}^t}$, $(t,i) \in \mathcal{I}'$. Moreover, by construction for every change point θ_k, $k = 1..\kappa$ there exists some $(t,i) \in \mathcal{I}$ such that $\theta_k = \theta(t,i)$ which, by (3.88) is estimated consistently for all $n \geqslant N$. Therefore, for all $n \geqslant N$ a consistent estimate for every change point appears at least once in $\widehat{\boldsymbol{\theta}}$. Next we show that a consistent estimate for every change point appears *at most* once in $\widehat{\boldsymbol{\theta}}$. By (3.88) for all (t,i), $(t',i') \in \mathcal{I}$ such that $\theta(t,i) = \theta(t',i')$ and all $n \geqslant N$ we have

$$\begin{aligned} |\widehat{\theta}_i^t(n) - \widehat{\theta}_{i'}^{t'}(n)| &\leqslant |\widehat{\theta}_i^t(n) - \theta(t,i)| + |\widehat{\theta}_{i'}^{t'}(n) - \theta(t',i')| \\ &\leqslant 2\varepsilon. \end{aligned} \tag{3.91}$$

On the other hand, for all (t,i), $(t',i') \in \mathcal{I}$ such that $\theta(t,i) \neq \theta(t',i')$ and all $n \geqslant N$ we

have

$$\begin{aligned}\frac{1}{n}|\widehat{\pi}_i^t - \widehat{\pi}_{i'}^{t'}| &\geqslant |\theta(t,i) - \theta(t',i')| - |\widehat{\theta}_i^t(n) - \theta(t,i)| - |\widehat{\theta}_{i'}^{t'}(n) - \theta(t',i')| \\ &\geqslant |\theta(t,i) - \theta(t',i')| - 2\varepsilon \\ &\geqslant \lambda_{\min} - 2\varepsilon \end{aligned} \tag{3.92}$$

where the last inequality follows from (3.88) and the fact that the true change points are at least $\lambda_{\min}$ apart. By (3.91) and (3.92), for all $n \geqslant N$, the indices (t,i) corresponding to the potential duplicate estimates of every change point are made unavailable at every iteration. Therefore, for all $n \geqslant N$ the final list of estimates obtained through this procedure, has the property that its first κ elements are consistent estimates of the change point parameters, and the statement of the theorem follows. □

3.4.4 Proof of Theorem 3.3.4

Recall the following notation used in Lemma 3.4.4. Consider the set $\mathcal{S}$ of segments specified by Line (5) in Algorithm 3. For every segment $\widetilde{\boldsymbol{x}}_i := X_{\psi_{i-1}..\psi_i} \in \mathcal{S}$ where $i = 1..|\Upsilon| + 1$ define $\widetilde{\rho}_i$ as the process distribution that generates the largest portion of $\widetilde{x}_i$. That is, let $\widetilde{\rho}_i := \rho_j$ where j is such that $K \in \mathcal{G}_j$ with

$$K := \operatorname*{argmax}_{k \in \mathcal{G}_{r'}} |\{\psi_{i-1}+1, \dots, \psi_i\} \cap \{n\theta_{k-1}+1, \dots, n\theta_k\}|$$

and $\mathcal{G}_j,\ j = 1..r$ are the ground-truth partitions defined by (3.1).

Proof. Let $\delta := \min_{i \neq j \in 1..r} d(\rho_i, \rho_j)$ denote the minimum distance between the distinct distributions that generate $\boldsymbol{x}$. Fix an $\varepsilon \in (0, \delta/4)$. Recall that the list-estimator $\Upsilon(\boldsymbol{x}, \lambda)$ is consistent for all $\lambda \in (0, \lambda_{\min}]$ (see Definition 3.2.1). Therefore, there exists some N_1 such that for all $n \geqslant N_1$ the first κ elements of the list of candidate estimates that it produces, converge to the true change point parameters. Here, the only important message is that, for all $n \geqslant N_1$ the consistent estimates are somewhere within the list Υ. That is for all $n \geqslant N_1$ there exists a set of indices $\{\mu_k :\ k = 1..\kappa\} \subseteq 1..|\Upsilon|$ such that

$$\sup_{k \in 1..\kappa} |\widehat{\theta}_{\mu_k} - \theta_k| \leqslant \varepsilon. \tag{3.93}$$

Since there are a finite number of segments in the set $\mathcal{S}$ (specified by (5) in Algorithm 3), by Lemma 3.4.4, there exists some N_2 such that for all $n \geqslant N_2$ we have

$$\sup_{\widetilde{\boldsymbol{x}}_i \in \mathcal{S}} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\rho}_i) \leqslant \varepsilon. \tag{3.94}$$

By (3.94), and applying the triangle inequality, for all $n \geqslant N_2$ we have

$$\sup_{\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{x}}_j \in \mathcal{S},\ \widetilde{\rho}_i = \widetilde{\rho}_j} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{x}}_j) \leqslant 2\varepsilon. \tag{3.95}$$

Similarly, for all $n \geqslant N_2$ we have

$$\inf_{\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{x}}_j \in \mathcal{S},\ \widetilde{\rho}_i \neq \widetilde{\rho}_j} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{x}}_j) \geqslant \delta - 2\varepsilon. \tag{3.96}$$

By (3.95) and (3.96), for all $n \geqslant N_2$, the segments $\widetilde{x}_i,\ \widetilde{x}_{i+1} \in \mathcal{S}$ with $\widetilde{\rho}_i = \widetilde{\rho}_{i+1}$ are closer to each other (in the empirical estimate of the distributional distance) than to the rest of the segments. By (3.96), for all $n \geqslant N_2$ and every $j \in 2..r$ we have

$$\max_{i \in 1..|\mathcal{S}|} \min_{j' \in 2..j-1} \widehat{d}(\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{x}}_{c_j}) \geqslant \delta - 2\varepsilon \tag{3.97}$$

where as specified by Algorithm 3, $c_1 := 1$ and $c_j,\ j = 2..r$ are given by (7). Hence, for all $n \geqslant N_2$, the cluster centers $\widetilde{x}_{c_j}, j = 1..r$ are each generated by a different process distribution. That is, $\widetilde{\rho}_{c_j} \neq \widetilde{\rho}_{c_{j'}}$ for $j \neq j' \in 1..r$. On the other hand, the rest of the segments are each assigned to the closest cluster, so that by (3.95) for all $n \geqslant N$ we have

$$T(\widetilde{\boldsymbol{x}}_i) = T(\widetilde{\boldsymbol{x}}_{i'}) \Leftrightarrow \widetilde{\rho}_i = \widetilde{\rho}_{i'}. \tag{3.98}$$

Let $N := \max N_i,\ i = 1, 2$. It remains to show that for all $n \geqslant N$, all of the redundant estimates namely, $\widehat{\theta}_i,\ i \neq \mu_k,\ k = 1..\kappa$ are removed in the last step of the algorithm, so that for all $n \geqslant N$ there exists an index $i \in 1..|\Upsilon|$ in $\mathcal{C}$, if and only if it corresponds to a consistent estimate in Υ. To this end, we note that by (3.93) and (3.98) for all $n \geqslant N$ and all $i \in \mathcal{C}$ we have $\widetilde{\rho}_i \neq \widetilde{\rho}_{i+1}$ so that

$$\mathcal{C} = \{\mu_k :\ k = 1..\kappa\} \tag{3.99}$$

for all $n \geqslant N$. Since as specified by Algorithm 3 we have $\widehat{\kappa} := |\mathcal{C}|$, by (3.93) and (3.99)

the algorithm is consistent. □

Chapter 4

Time-series clustering

In this chapter we consider the problem of online time-series clustering. Each data point is a sequence generated by a stationary ergodic process distribution. Data arrive in an online fashion so that the sample received at every time-step is either a new sequence, or extends one that has been previously observed. As in the previous chapter, we consider a general non-parametric statistical framework for the data, assuming that the marginal distribution of each sequence is stationary and ergodic. We propose an online clustering algorithm, which we further demonstrate to be asymptotically consistent (under a natural notion of consistency). Our main purpose, as in the rest of the thesis, is to theoretically demonstrate that it is possible to have consistent methods (in this case an online clustering algorithm) in this general statistical framework. However, it turns out that this constructive theoretical solution is also computationally efficient, can be easily implemented and, as such, suits many practical applications.

This work has been published in the proceedings of the 15*th international conference on Artificial Intelligence and Statistics (AI&Stats 2012), see (Khaleghi et al., 2012).*

Contents

4.1 Introduction

As indispensable tools in almost all fields of science, clustering algorithms have been extensively studied in the literature. Informally, the goal of clustering is to partition a given dataset in a *natural* way, thus potentially revealing an underlying structure in the data. In this chapter, we focus on a subset of the clustering problem, where the data to be clustered are observations obtained sequentially over time. Specifically, we consider the problem of time-series clustering in an online setting, where new data arrive dynamically, and the objective is to construct an algorithm that correctly clusters recently observed data points as soon as possible, without changing its decision about those that have already been clustered correctly. Indeed, in many real-world scenarios it is desired to infer underlying structures in the data, while new sources are continuously being added and previously available sources are generating more data. This may arise, for example, in modern consumer markets, where customer behaviour monitored over time is used in the development of profitable marketing attribution strategies. An appropriate partitioning of the set of consumers may help product managers obtain useful strategic insights. The collected data are naturally in the form of time series, where each sequence may be highly dependent, and there may even be some dependence between the different sequences. Moreover, the cluster analysis must certainly be robust with respect to the constant introduction of new customers, as well as to the dynamic nature of consumers' interests.

4.1.1 Problem formulation

We consider the *online* version of the time series clustering problem. This means that we have a growing body of sequences of data. Both the number of sequences, as well as the sequences themselves grow with time. The manner of this evolution can be arbitrary;

we only require that the length of each individual sequence tends to infinity. As in the previous chapter, we consider a general nonparametric framework, where the joint distribution over the sequences is unknown and can be arbitrary; our only assumption is that the marginal distribution of each sample is one of κ unknown stationary ergodic process distributions. We impose no statistical assumptions (beyond stationarity and ergodicity) on the κ marginal distributions. That is, the samples are not required to be independent, and the process distributions are not assumed to have finite-memory or to satisfy mixing conditions. There can be any form of dependence between the samples, and the dependence can even be thought of as adversarial.

At time-step 1, initial segments of some of the first sequences are available to the learner. At the subsequent time steps, new data are revealed, either as subsequent segments of previously observed sequences, or as new sequences. Thus, at each time step t a total of $N(t)$ sequences $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N(t)}$ are to be clustered, where each sequence $\boldsymbol{x}_i$ is of length $n_i(t) \in \mathbb{N}$ for $i = 1..N(t)$. The total number of observed sequences $N(t)$ as well as the individual sequence-lengths $n_i(t)$ grow with time. The target clustering is to partition the samples into κ clusters based on the process distributions that generate them. In this setting, we define a clustering algorithm to be *asymptotically consistent*, if the clustering restricted to each fixed batch of sequences $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, coincides with the target clustering from some time on.

4.1.2 Main results

We present an easily implementable online clustering algorithm that, as we demonstrate, is asymptotically consistent provided that the unknown marginal distribution of each sequence is stationary ergodic, and that the correct number κ of clusters is known. We further show that our algorithm is computationally efficient: its computational complexity is at most quadratic in each argument. Note that, the assumption that κ is known and provided to the algorithm is inevitable. Indeed, as discussed in Section 2.3 the impossibility result of (Ryabko, 2010b) implies that if κ is unknown, the time series clustering problem considered in this chapter can not possibly admit a solution in this general setting. Moreover, the asymptotic results obtained in this chapter cannot be strengthened, as rates of convergence are provably impossible to obtain in this setting.

4.1.3 Relation to the offline setting

As in the previous chapter, our methods are based on the empirical estimates of the distributional distance.

Let us start by analyzing the offline version of the problem, where we are to group a *batch* of samples $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$ of length at least n into κ clusters. In this case, an asymptotically consistent algorithm produces a set of clusters, that for large enough n, coincides with the ground-truth. By the consistency of $\widehat{d}$, we can state that the samples generated by the same process distribution are closer to each other, than to those generated by different process distributions. Therefore, the batch of sequences $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$ has the so-called *strict separation property (M. et al., 2008)* in the limit as n approaches infinity. This makes many simple algorithms such as single or average linkage, or k-means (with appropriate initializations) provably consistent (in the asymptotic sense). In fact, an asymptotically consistent batch algorithm has already been proposed by (Ryabko, 2010a).

At first glance, it may seem that the online version of the problem can be solved by simply applying a consistent batch algorithm to the entire dataset observed at each time step. However, this naive approach does not result in a consistent solution. The main challenge in this problem can be identified with what we regard as "bad" sequences: recently-observed sequences, for which sufficient information has not yet been collected, and as such are not possible to be distinguished based on the process distributions that generate them. In this setting, using a batch algorithm at every time step, will result in not only mis-clustering such "bad" sequences, but also in clustering incorrectly the samples for which sufficient data is already available. That is, such "bad" sequences could render the entire batch clustering useless, leading the algorithm to even make incorrect decisions on the "good" sequences. Since new sequences may arrive arbitrarily (even in a data-dependent, or adversarial fashion), any batch algorithm will fail in this scenario. We illustrate this phenomenon in Chapter 5, where we show that the clustering error rate of our method converges to zero in an online setting, while the batch algorithm of (Ryabko, 2010a) is consistently confused by the dynamic nature of the data.

Our algorithm is based on a *weighted combination* of several clustering decisions, each obtained by running the offline algorithm on different portions of data. The partitions are combined with weights that depend on the batch size and on an appropriate

performance measure for each individual partition. The performance measure of each clustering is the minimum inter-cluster distance.

4.2 Preliminaries

4.2.1 Problem formulation

The problem may be formulated as follows. Consider the infinite matrix $\mathbf{X}$ of random variables (given by (2.2)) which is generated by an arbitrary, unknown process distribution ρ. Assume that the marginal distribution of ρ over each row of $\mathbf{X}$ is one of κ unknown, stationary ergodic process distributions $\rho_1, \ldots, \rho_\kappa$. Thus, the matrix $\mathbf{X}$ corresponds to infinitely many one-way infinite sequences, each of which is generated by a stationary ergodic distribution. Aside from this assumption, we do not make any further assumptions on the distribution ρ that generates $\mathbf{X}$. This means that the samples in $\mathbf{X}$ are allowed to be (arbitrarily) dependent; we can even think of the dependence between the samples as *adversarial*. For convenience of notation we assume that the distributions $\rho_k, k = 1..\kappa$ are ordered in the order of appearance of their first samples in $\mathbf{X}$.

At every time step $t \in \mathbb{N}$, a set $S(t)$ of samples is observed corresponding to the first $N(t) \in \mathbb{N}$ rows of $\mathbf{X}$, each of length $n_i(t), i \in 1..N(t)$, i.e.

$$S(t) = \{\boldsymbol{x}_1^t, \cdots \boldsymbol{x}_{N(t)}^t\} \text{ where } \boldsymbol{x}_i^t := X_{1..n_i(t)}^i.$$

More precisely, $S(t)$ is obtained from $\mathbf{X}$ as follows. At every time-step, a number $N(t)$ is fixed, corresponding to the total number of sequences observed at time t, i.e. the cardinality of $S(t)$. Next, some (arbitrary) lengths $n_i(t) \in \mathbb{N},\ i \in 1..N(t)$ are fixed, and for each $i \in 1..N(t)$ the sequence $\boldsymbol{x}_i(t) := X_1^{(i)}, \ldots X_{n_i(t)}^{(i)}$ is obtained as the first $n_i(t)$ elements of the i^{th} row of $\mathbf{X}$. We assume that the number $N(t)$ of samples, as well as the individual sample-lengths $n_i(t),\ i = 1..N(t)$ grow with time. That is, we assume that the length $n_i(t)$ of each sequence $\boldsymbol{x}_i$ is non-decreasing and grows to infinity (as a function of time t). The number of sequences $N(t)$ also grows to infinity. Apart from these assumptions, the functions $N(t)$ and $n_i(t)$ are completely arbitrary. The protocol is depicted in Figure 4.2.1.

Of the many ways a set of κ disjoint subsets of the rows of $\mathbf{X}$ may be produced, the

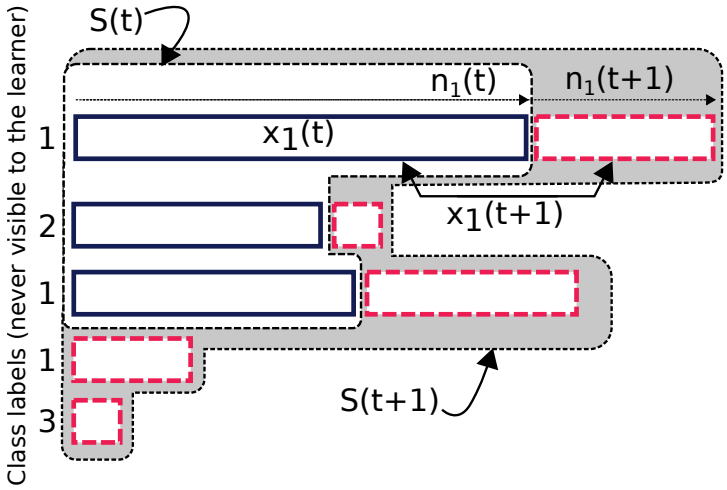

Figure 4.1: Online Protocol: solid rectangles correspond to sequences observed at time t, dashed rectangles correspond to segments arrived at time $t+1$.

most natural partitioning in this formulation is to group together those and only those rows of $\mathbf{X}$ which have the same marginal distributions. More precisely, we define the ground-truth partitioning of $\mathbf{X}$ as follows.

Definition 4.2.1 (Ground-truth $\mathcal{G}$)**.** *Let*

$$\mathcal{G} = \{\mathcal{G}_1, \ldots, \mathcal{G}_\kappa\}$$

be a partitioning of $\mathbb{N}$ *into* κ *disjoint subsets* $\mathcal{G}_k$, $k = 1..\kappa$, *such that* $\boldsymbol{x}_i$, $i \in \mathbb{N}$ *is generated by* ρ_k *for some* $k \in 1..\kappa$ *if and only if* $i \in \mathcal{G}_k$. *Call* $\mathcal{G}$ *the ground-truth clustering. Introduce also the notation* $\mathcal{G}|_N$ *for the restriction of* $\mathcal{G}$ *to the first* N *sequences:*

$$\mathcal{G}|_N := \{\mathcal{G}_k \cap \{1..N\} : k = 1..\kappa\}.$$

Remark 4.2.2. *Note that even though the* eventual *number* κ *of different time-series distributions producing the sequences is assumed known, the number of observed distributions at each individual time-step is unknown. In particular, it may be possible that at a given time-step* t *we have* $\{1..N(t)\} \cap \mathcal{G}_k = \varnothing$ *for some* $k \in 1..\kappa$.

A clustering function f takes a finite set $S := \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ of samples and a parameter κ (the number of target clusters) to produce a partition $f(S, \kappa) := \{C_1, \ldots, C_\kappa\}$ of the index-set $\{1..N\}$. The goal is to partition $\{1..N\}$ in such a way as to recover

the ground-truth clustering $\mathcal{G}$. A clustering algorithm asymptotically consistent if it achieves this goal for long enough sequences $\boldsymbol{x}_i$, $i = 1..N$ in S. That is, we call a clustering function asymptotically consistent, if with probability 1, for each $N \in \mathbb{N}$ from some time on the first N sequences are clustered correctly (with respect to the ground-truth given by Definition 4.2.1). Specifically, we have the following definition.

Definition 4.2.3 (Asymptotic consistency)**.** *A clustering function, is asymptotically consistent in the online sense, if for every $N \in \mathbb{N}$ we have*

$$\lim_{n\to\infty} f(S(t),\kappa)|_N = \mathcal{G}|_N,$$

where, $S(t) := \{\boldsymbol{x}_1^t, \dots, \boldsymbol{x}_{N(t)}^t\}$, $n := \min\{n_1(t), \dots, n_N(t)\}$, and $f(S(t),\kappa)|_N$ is the clustering result of $f(S(t),\kappa)$, restricted to the first N samples i.e.,

$$f(S(t),\kappa)|_N := \{f(S(t),\kappa) \cap \{1..N\}, k = 1..\kappa\}.$$

An example of this notion is depicted in Figure 4.2.1, for $N = 4$.

Number κ of clusters. Under the general framework described above, consistent clustering with unknown number of clusters is impossible. This follows from the impossibility result of (Ryabko, 2010b), stated in Section 2.3. Recall that by this theorem, when we have only two *binary-valued* samples, generated *independently* by two stationary ergodic distributions, it is impossible to decide whether they have been generated by the same or by different distributions, even in the weak asymptotic sense. Therefore, if the number of clusters is unknown, we are bound to make stronger assumptions on the process distributions that generate the data, (e.g. assume that the process distributions satisfy certain mixing conditions). However, since our main interest in this chapter is to develop consistent clustering algorithms under the general assumptions described above, we assume that the correct number κ of clusters is known, and proceed to work in our general statistical framework.

4.2.2 An offline clustering algorithm due to (Ryabko, 2010a)

Our method relies on a consistent batch clustering algorithm as a subroutine. To this end, we use the algorithm proposed by (Ryabko, 2010a), which we present in this section for completeness.

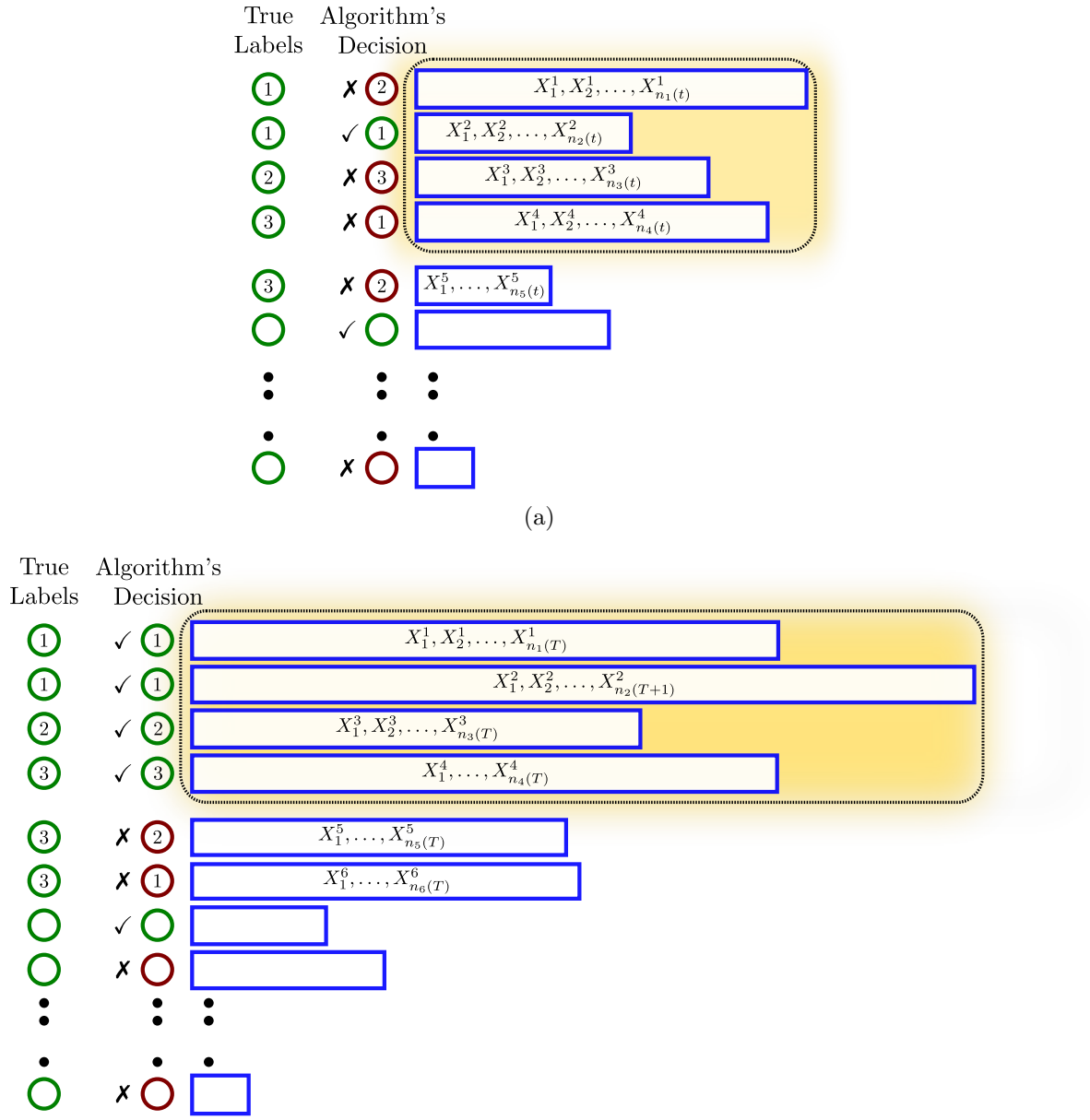

Figure 4.2: (a). The first $N = 4$ samples have not been clustered correctly yet. (b) From some time T on, they continue to be clustered correctly, regardless of the algorithm's decision on the other samples. Note that the true labels are **never observed** by the algorithm.

The procedure is outlined in Algorithm 4. Given a batch of sequences, the algorithm initializes the clusters using farthest-point initialization, and assigns each of the remaining points to the nearest cluster. More precisely, the sample $\boldsymbol{x}_1$ is assigned as the first cluster centre. Then a sample is found that is farthest away from $\boldsymbol{x}_1$ in the empirical distributional distance $\widehat{d}$ and is assigned as the second cluster centre. For each $k = 2..\kappa$ the k^{th} cluster centre is sought as the sequence with the largest minimum distance from the already assigned cluster centres for $1..k-1$. By the last iteration we have κ cluster centres. Next the remaining samples are each assigned to the closest cluster.

Algorithm 4 Offline clustering method of (Ryabko, 2010a)

INPUT: sequences $S := \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N\}$**, Number** κ **of clusters**

Initialize κ***-farthest points as cluster-centres:***

$c_1 \leftarrow 1$

$C_1 \leftarrow \{c_1\}$

for $k = 2..\kappa$ **do**

  $c_k \leftarrow \operatorname{argmax}_{i=1..N} \min_{j=1..k-1} \widehat{d}(\boldsymbol{x}_i, \boldsymbol{x}_{c_j})$**,** ***where ties are broken arbitrarily***

  $C_k \leftarrow \{c_k\}$

end for

Assign the remaining points to closest centres:

for $i = 1..N$ **do**

  $k \leftarrow \operatorname{argmin}_{j \in \bigcup_{k=1}^{\kappa} C_k} \widehat{d}(\boldsymbol{x}_i, \boldsymbol{x}_j)$

  $C_k \leftarrow C_k \cup \{i\}$

end for

OUTPUT: clusters $C_1, C_2, \cdots, C_\kappa$

As shown in Theorem 4.2.4, this method is asymptotically consistent.

Theorem 4.2.4 (Algorithm 4 is consistent. (Ryabko, 2010a))**.** *Given a set of* $N \in \mathbb{N}$ *samples* $S := \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ *of length at least* $n := \min\{n_1, \ldots, n_N\}$ *we have,*

$$\lim_{n\to\infty} Alg4(S, \kappa) = \mathcal{G}|_N,$$

provided the correct number κ *of clusters of clusters is known, and the marginal distribution of each sequence* $\boldsymbol{x}_i, i = 1..N$ *is stationary ergodic.*

4.3 Main result: a consistent online time series clustering algorithm

The online version of the problem turns out to be more complex than the offline one. Since new sequences arrive (potentially) at every time step, we can never rely on all distance estimates to be correct. Thus, as mentioned in the introduction, the main challenge can be identified with what we regard as "bad" sequences: recently-observed sequences, for which sufficient information has not yet been collected, and for which the estimates of the distance (with respect to any other sequence) are bound to be misleading. In particular, farthest-point initialization would not work. More generally, using any offline algorithm on all available data at every time step results in not only mis-clustering the "bad" sequences, but also in clustering incorrectly those for which sufficient data is already available. The solution, realized in Algorithm 5, is based on a *weighted combination* of several clusterings, each obtained by running the offline algorithm (Algorithm 4) on different portions of data. The partitions are combined with weights that depend on the batch size and on the minimum inter-cluster distance. As mentioned in Section 1.4.2, this last step of combining multiple clusterings with weights may be reminiscent of prediction with expert advice (e.g., Cesa-Bianchi and Lugosi, 2006), where experts are combined based on their past performance. However, the difference here is that the performance of each clustering cannot be measured directly in our setting.

More precisely, Algorithm 5 works as follows. Given a set $S(t)$ of $N(t)$ samples, the algorithm iterates over $j := \kappa, \dots, N(t)$ where at each iteration Algorithm 4 is used to cluster the first j sequences $\{\boldsymbol{x}_1^t, \dots, \boldsymbol{x}_j^t\}$ into κ clusters. In each cluster the sequence with the smallest index is assigned as the candidate cluster centre. A performance score γ_j is calculated as the minimum distance $\widehat{d}$ between the κ candidate cluster centers obtained at iteration j. Thus, γ_j is an estimate of the minimum inter-cluster distance. At this point we have $N(t)-\kappa+1$ sets of κ cluster centers $c_1^j, \dots, c_\kappa^j,\ j = 1..N(t)-\kappa+1$. Next, every sample $\boldsymbol{x}_i^t,\ i = 1..N(t)$ is assigned to the *closest* cluster. This is determined by minimizing the weighted combination of the distances between $\boldsymbol{x}_i^t$ and the candidate cluster centers obtained at each iteration on j. More specifically, for each $i = 1..N(t)$

Algorithm 5 Online Clustering

INPUT: Number κ of target clusters
for $t = 1..\infty$ **do**
  Obtain new sequences $S(t) = \{\boldsymbol{x}_1^t, \cdots, \boldsymbol{x}_{N(t)}^t\}$
  Initialize the normalization factor: $\eta \leftarrow 0$
  Initialize the final clusters: $C_k(t) \leftarrow \varnothing,\ k = 1..\kappa$
  Generate $N(t) - \kappa + 1$ **candidate cluster centers:**
  for $j = \kappa..N(t)$ **do**
    $\{C_1^j, \ldots, C_\kappa^j\} \leftarrow \text{Alg4}(\{\boldsymbol{x}_1^t, \cdots, \boldsymbol{x}_j^t\}, \kappa)$
    $\mu_k \leftarrow \min\{i \in C_k^j\}, k = 1..\kappa$ ▷ Set the smallest index as cluster center.
    $(c_1^j, \ldots, c_\kappa^j) \leftarrow \text{sort}(\mu_1, \ldots, \mu_\kappa)$ ▷ Sort the cluster centers increasingly.
    $\gamma_j \leftarrow \min_{k \neq k' \in 1..\kappa} \widehat{d}(\boldsymbol{x}_{c_k^j}^t, \boldsymbol{x}_{c_{k'}^j}^t)$ ▷ Calculate performance score.
    $w_j \leftarrow j^{-2}$
    $\eta \leftarrow \eta + w_j \gamma_j$
  end for
  Assign points to clusters:
  for $i = 1..N(t)$ **do**
    $k \leftarrow \text{argmin}_{k' \in 1..\kappa} \frac{1}{\eta} \sum_{j=1}^{N(t)} w_j \gamma_j \widehat{d}(\boldsymbol{x}_i^t, \boldsymbol{x}_{c_{k'}^j}^t)$
    $C_k(t) \leftarrow C_k(t) \cup \{i\}$
  end for
  OUTPUT: $\{C_1(t), \cdots, C_k(t)\}$
end for

the sequence $\boldsymbol{x}_i^t$ is assigned to the cluster k where k is defined as

$$k := \text{argmin}_{k=1..\kappa} \sum_{j=\kappa}^{N(t)} w_j \gamma_j \widehat{d}(\boldsymbol{x}_i^t, \boldsymbol{x}_{c_k^j}^t).$$

Theorem 4.3.1. *Algorithm 5 is asymptotically consistent (in the sense of Definition 4.2.3), provided that the correct number of clusters κ of clusters is known, and the marginal distribution of each sequence $\boldsymbol{x}_i, i \in \mathbb{N}$ is stationary ergodic.*

Proof Sketch: Before proceeding to the proof of Theorem 4.3.1, we provide an intuitive explanation. First, consider the following simple solution. Select a fixed (reference) portion of the samples, and provide it as input to a consistent batch clustering algorithm (such as Algorithm 4); next, assign every remaining sequence to the nearest cluster. Since the offline algorithm is asymptotically consistent, this procedure would be

asymptotically consistent as well. However, this would only be true provided that the initial selection contains at least one representative from each cluster (i.e. a sequence sampled from each and every one of the κ process distributions.) However, there is no way to find a fixed (not growing with time) portion of data that would be guaranteed to have this property. Another option would be to allow the reference set to grow with time, so that eventually it contains representatives from all clusters. However, the offline algorithm applied to a growing set of sequences will not produce consistent results, and this would bring us back to the original online problem.

A key observation we make to tackle this problem is the following. If, for some $j \in \{\kappa, \dots, N(t)\}$, each sample in the batch $\{\boldsymbol{x}_1^t, \dots, \boldsymbol{x}_j^t\}$ is generated by *at most* $\kappa - 1$ process distributions, then any partitioning of this batch into κ clusters results in a minimum inter-cluster distance γ_j, which as follows from the asymptotic consistency of $\widehat{d}$, converges to 0. On the other hand, if the set of samples $\{\boldsymbol{x}_1^t, \dots, \boldsymbol{x}_j^t\}$ contains sequences generated by all κ process distributions, γ_j converges to a non-zero constant, namely, the minimum distance between the distinct process distributions $\rho_1, \dots, \rho_\kappa$. In this case, as follows from the consistency of Algorithm 4, from some time on, the batch $\{\boldsymbol{x}_1^t, \dots, \boldsymbol{x}_j^t\}$ will be clustered correctly. Thus, instead of selecting one reference batch of sequences and constructing a set of clusters, we consider all possible batches of sequences for $j = \kappa..N(t)$, and combine them with weights. Two sets of weights are involved in this step: γ_j and w_j, where

1. γ_j is used to penalize for small inter-cluster distance, canceling the clustering results produced based on sets of sequences generated by less than κ distributions;

2. w_j is used to give precedence to chronologically earlier clusterings, protecting the clustering decisions from the presence of the (potentially "bad") newly revealed data, whose corresponding distance estimates may still be far from accurate.

It remains to use the consistency of $\widehat{d}$ and that of Algorithm 4 to see that every finite number $N \in \mathbb{N}$ of points are from some time on assigned to the correct target cluster. □

Computational Complexity. We calculate the per symbol computational complexity of Algorithm 5. Assume that the pairwise distance values are stored in a database D, and that for every sequence $\boldsymbol{x}_i^{t-1}$, $i \in \mathbb{N}$ we have already constructed a suffix tree, using

for example, the online algorithm of (Ukkonen, 1995). At time-step t, a new symbol X is received. Let us first calculate the required computations to update D. We have two cases, either X forms a new sequence, so that $N(t) = N(t-1) + 1$, or it is the subsequent element of a previously received segment, say, $\boldsymbol{x}_j^t$ for some $j \in 1..N(t)$, so that $n_j(t) = n_j(t-1) + 1$. In either case, let $\boldsymbol{x}_j^t$ denote the updated sequence. Note that for all $i \neq j \in 1..N(t)$ we have $n_i(t) = n_i(t-1)$. Recall the notation $\boldsymbol{x}_i^t := X_1^{(i)}, \dots X_{n_i(t)}^{(i)}$ for $i \in 1..N(t)$. In order to update D we need to update the distance between $\boldsymbol{x}_j^t$ and $\boldsymbol{x}_i^t$ for all $i \neq j \in N(t)$. Thus, we need to search for all m_n new patterns induced by the received symbol X, resulting in complexity at most $\mathcal{O}(N(t)m_n^2 l_n)$. Let $n(t) := \max\{n_1(t), \dots n_{N(t)}(t)\},\ t \in \mathbb{N}$. As discussed previously, we let $m_n := \log n(t)$; we also define $l_n := \log s(t)^{-1}$ where

$$s(t) := \min_{\substack{i,j \in 1..N(t) \\ u=1..n_i(t), v=1..n_j(t), X_u^{(i)} \neq X_v^{(j)}}} |X_u^{(i)} - X_v^{(j)}|,\ t \in \mathbb{N}.$$

Thus, the per symbol complexity of updating D is at most $\mathcal{O}(N(t)\log^3 n(t))$. However, note that if $s(t)$ decreases from one time-step to the next, updating D will have a complexity of order equivalent to its complete construction, resulting in a computational complexity at most $\mathcal{O}(N(t)n(t)\log^2 n(t))$. Therefore, we avoid calculating $s(t)$ at every time-step; instead, we update $s(t)$ at pre specified time-steps so that for every $n(t)$ symbols received, D is reconstructed at most $\log n(t)$ times. (This can be done, for example, by recalculating $s(t)$ at time-steps where $n(t)$ is a power of 2.) It is easy to see that with the database D of distance values at hand, the rest of the computations are of order at most $\mathcal{O}(N(t)^2)$. Thus, the per symbol computational complexity of Algorithm 5 is at most $\mathcal{O}(N(t)^2 + N(t)\log^3 n(t))$.

4.4 Proof of Theorem 4.3.1

Proof. We will show that for every $k \in 1..\kappa$ we have

$$\frac{1}{\eta}\sum_{j=1}^{N(t)} w_j \gamma_j^t \widehat{d}(\boldsymbol{x}_{c_k^j}^t, \rho_k) \to 0 \text{ a.s.} \tag{4.1}$$

Denote by δ the minimum non-zero distance between the process distributions i.e.,

$$\delta := \min_{k \neq k' \in 1..\kappa} \widehat{d}(\rho_k, \rho_{k'}). \tag{4.2}$$

Fix $\varepsilon \in (0, \delta/4)$. We can find an index J such that $\sum_{j=J}^{\infty} w_j \leqslant \varepsilon$. Let $S(t)|_j = \{\boldsymbol{x}_1^t, \cdots, \boldsymbol{x}_j^t\}$ denote the subset of $S(t)$ consisting of the first j sequences for $j \in 1..N(t)$. For $k = 1..\kappa$ let

$$s_k := \min\{i \in \mathcal{G}_k \cap 1..N(t)\} \tag{4.3}$$

index the first sequence in $S(t)$ that is generated by ρ_k where $\mathcal{G}_k,\ k = 1..\kappa$ are the ground-truth partitions given by Definition 4.2.1. Define

$$m := \max_{k \in 1..\kappa} s_k. \tag{4.4}$$

Recall that the sequence lengths $n_i(t)$ grow with time. Therefore, by the consistency of $\widehat{d}$, i.e. Lemma 2.2.4 for every fixed $j \in 1..J$ there exists some $T_1(j)$ such that for all $t \geqslant T_1(j)$ we have

$$\sup_{\substack{k \in 1..\kappa \\ i \in \mathcal{G}_k \cap \{1..j\}}} \widehat{d}(\boldsymbol{x}_i^t, \rho_k) \leqslant \varepsilon. \tag{4.5}$$

Moreover, by Theorem 4.2.4 for every $j \in m..J$ there exists some $T_2(j)$ such that Alg4$(S(t)|_j, \kappa)$ is consistent for all $t \geqslant T_2(j)$. Let

$$T := \max_{\substack{i=1,2 \\ j \in 1..J}} T_i(j)$$

Recall that by definition (i.e. 4.4) $S(t)|_m$ contains samples from all κ distributions. Therefore, for all $t \geqslant T$

$$\begin{aligned} \inf_{k \neq k' \in 1..\kappa} \widehat{d}(\boldsymbol{x}_{c_k^m}^t, \boldsymbol{x}_{c_{k'}^m}^t) &\geqslant \inf_{k \neq k' \in 1..\kappa} d(\rho_k, \rho_{k'}) - \sup_{k \neq k' \in 1..\kappa} (\widehat{d}(\boldsymbol{x}_{c_k^m}^t, \rho_k) + \widehat{d}(\boldsymbol{x}_{c_{k'}^m}^t, \rho_{k'})) \\ &\geqslant \delta - 2\varepsilon \geqslant \delta/2. \end{aligned} \tag{4.6}$$

where the first inequality follows from the triangle inequality and the second inequality follows from the consistency of Alg4$(S(t)|_m, \kappa)$ for $t \geqslant T$, the definition of δ given by

(4.2) and the assumption that $\varepsilon \in (0, \delta/4)$. Recall that (as specified in Algorithm 5) we have $\eta = \sum_{j=1}^{N(t)} w_j \gamma_j^t$. Hence, by (4.6) for all $t \geqslant T$ we have

$$\eta \geqslant w_m \delta/2. \tag{4.7}$$

By (4.7) and noting that by definition $\widehat{d}(\cdot,\cdot) \leqslant 1$ for every $k \in 1..\kappa$ we obtain,

$$\frac{1}{\eta}\sum_{j=1}^{N(t)} w_j \gamma_j^t \widehat{d}(\boldsymbol{x}^t_{c_k^j}, \rho_k) \leqslant \frac{1}{\eta}\sum_{j=1}^{J} w_j \gamma_j^t \widehat{d}(\boldsymbol{x}^t_{c_k^j}, \rho_k) + \frac{2\varepsilon}{w_m \delta}. \tag{4.8}$$

On the other hand by definition (i.e. (4.4)) the sequences in $S(t)|_j$ for $j = 1..m-1$ are generated by *at most* $\kappa - 1$ out of the κ process distributions. Therefore, at every iteration on $j \in 1..m-1$ there exists at least one pair of distinct cluster centres that are generated by *the same* process distribution. Therefore, by (4.5) and (4.7), for all $t \geqslant T$ and every $k \in 1..\kappa$ we have,

$$\frac{1}{\eta}\sum_{j=1}^{m-1} w_j \gamma_j^t \widehat{d}(\boldsymbol{x}^t_{c_k^j}, \rho_i) \leqslant \frac{1}{\eta}\sum_{j=1}^{m-1} w_j \gamma_j^t \leqslant \frac{2\varepsilon}{w_m \delta}. \tag{4.9}$$

Noting that the clusters are ordered in the order of appearance of the distributions, we have $\boldsymbol{x}^t_{c_k^j} = \boldsymbol{x}^t_{s_k}$ for all $j = m..J$ and $k = 1..\kappa$, where the index s_k is defined by (4.3). Therefore, by (4.5) for all $t \geqslant T$ and every $k = 1..\kappa$ we have

$$\frac{1}{\eta}\sum_{j=m}^{J} w_j \gamma_j^t \widehat{d}(\boldsymbol{x}^t_{c_k^j}, \rho_k) = \frac{1}{\eta}\widehat{d}(\boldsymbol{x}^t_{s_k}, \rho_k)\sum_{j=m}^{J} w_j \gamma_j^t \leqslant \varepsilon. \tag{4.10}$$

Combining (4.8), (4.9), and (4.10) we obtain

$$\frac{1}{\eta}\sum_{j=1}^{N(t)} w_j \gamma_j^t \widehat{d}(\boldsymbol{x}^t_{c_k^j}, \rho_k) \leqslant \varepsilon(1 + \frac{4}{w_m \delta}). \tag{4.11}$$

for all $k = 1..\kappa$ and all $t \geqslant T$ (establishing 4.1). To conclude the proof of the consistency consider an index $i \in \mathcal{G}_r$ for some $r \in 1..\kappa$. By Lemma 2.2.4, increasing T if necessary,

for all $t \geqslant T$ we have

$$\sup_{\substack{k\in 1..\kappa \\ i\in\mathcal{G}_k\cap 1..N}} \widehat{d}(\boldsymbol{x}_i^t, \rho_k) \leqslant \varepsilon. \tag{4.12}$$

For all $t \geqslant T$ and all $k \neq r \in 1..\kappa$ we have,

$$\begin{aligned}
\frac{1}{\eta}\sum_{j=1}^{N(t)} w_j\gamma_j^t \widehat{d}(\boldsymbol{x}_i^t, \boldsymbol{x}_{c_k^j}^t) &\geqslant \frac{1}{\eta}\sum_{j=1}^{N(t)} w_j\gamma_j^t \widehat{d}(\boldsymbol{x}_i^t, \rho_k) - \frac{1}{\eta}\sum_{j=1}^{N(t)} w_j\gamma_j^t \widehat{d}(\boldsymbol{x}_{c_k^j}^t, \rho_k) \\
&\geqslant \frac{1}{\eta}\sum_{j=1}^{N(t)} w_j\gamma_j^t (\widehat{d}(\rho_k, \rho_r) - \widehat{d}(\boldsymbol{x}_i^t, \rho_r)) - \frac{1}{\eta}\sum_{j=1}^{N(t)} w_j\gamma_j^t \widehat{d}(\boldsymbol{x}_{c_k^j}^t, \rho_k) \\
&\geqslant \delta - 2\varepsilon(1 + \frac{2}{w_m\delta}),
\end{aligned} \tag{4.13}$$

where the first and second inequalities follow from subsequent application of the triangle inequality, and the last inequality follows from (4.12), (4.11) and the definition of δ. Since the choice of ε is arbitrary, from (4.12) and (4.13) we obtain

$$\operatorname*{argmin}_{k\in 1..\kappa} \frac{1}{\eta}\sum_{j=1}^{N(t)} w_j\gamma_j^t \widehat{d}(\boldsymbol{x}_i^t, \boldsymbol{x}_{c_k^j}^t) = r. \tag{4.14}$$

Finally, note that for any fixed $N \in \mathbb{N}$ from some t on (4.14) holds for all $i = 1..N$, and the consistency statement follows. □

Chapter 5

Experimental evaluations

In this chapter we evaluate our method using synthetically generated data. Our experiments concern stationary ergodic time series with the property that the single-dimensional marginals of the sequences generated by different process distributions are the same and the samples need not satisfy any mixing conditions. Our approaches to both change point estimation and clustering are completely non-parametric. To the best of our knowledge the existing non-parametric methods cannot work in such a general framework. In order to generate the data we use stationary ergodic process distributions that do not belong to any *simpler* general class of time series. The process by which the data are generated, is outlined in Section 5.1. The empirical evaluations of our change point methods, namely Algorithms 1, 2, and 3 are provided in Section 5.2. In Section 5.3 we evaluate the performance of our online clustering algorithm (i.e. Algorithm 5); the offline method of (Ryabko, 2010a) (i.e. Algorithm 4) has been used as a means of comparison. Some experiments with real datasets (motion clustering) are also provided.

Contents

5.1 Synthetic time series generation

In order for the experimental setup to reflect the generality of our framework, we generate the synthetic data by stationary ergodic time-series distributions that do not belong to any "simpler" general class of time-series. The considered processes are classical examples in the literature on ergodic time series (Billingsley, 1961) . In particular, they are used by (Shields, 1996) as an example of a class of stationary ergodic processes that are not B-processes. More specifically, they cannot be modelled by a hidden Markov model with a finite or countably infinite set of states. To the best of our knowledge we are the first to use these process distributions, in an experimental setup, and outside a mere theoretical context.

The general process by which to generate a sequence $\boldsymbol{y} := Y_1, \ldots, Y_m \in \mathbb{R}^m,\ m \in \mathbb{N}$ is outlined below.

1. Fix a parameter $\alpha \in (0, 1)$ and two uniform distributions $\mathcal{U}_1$ and $\mathcal{U}_2$.
2. Let r_0 be drawn randomly from $[0, 1]$.
3. For each $i = 1..m$ obtain $r_i := r_{i-1} + \alpha[1 \bmod]$; draw $y_i^{(j)}$ from $\mathcal{U}_j,\ j = 1, 2$.
4. Set $Y_i := \mathbb{I}\{r_i \leqslant 0.5\} y_i^{(1)} + \mathbb{I}\{r_i > 0.5\} y_i^{(2)}$.

Note that the same procedure can be used to generate a binary-valued sequence, by setting $y_i^{(1)} := 0$ and $y_i^{(2)} := 1$ for all $i \in 1..m$, instead of sampling them from uniform distributions $\mathcal{U}_1$ and $\mathcal{U}_2$. If α is irrational this produces a real-valued stationary ergodic time series. We simulate α by a long double with a long mantissa.

5.2 Change point estimation

In this section we examine the performance of the change point estimation algorithms provided in Chapter 3. We start with the demonstration of the convergence of the error-rate with sequence length. Next we demonstrate that the dependence of Alg 2

on the lower bound λ on the minimum separation $\lambda_{\min}$ (given by Equation 3.2) of the change point parameters is rather mild.

5.2.1 Convergence with sequence length

The first experiment examines the convergence of Algorithm 1's error-rate with sequence length. To this end we fixed four parameters $\alpha_1 := 0.12..$, $\alpha_2 := 0.14..$, $\alpha_3 := 0.16..$ and $\alpha_4 := 0.18..$ (with long mantissae) to correspond to 4 different process distributions; we used uniform distributions $\mathcal{U}_1$ and $\mathcal{U}_2$ over $[0, 0.7]$ and $[0.3, 1]$ respectively, The uniform distributions were deliberately chosen to overlap. To produce $\boldsymbol{x} \in \mathbb{R}^n$ we generated $\kappa := 3$ change point parameters $\theta_k,\ k = 1..\kappa$ and set $\lambda_{\min} := 0.1$. Recall that, as specified by Equation (3.2), the parameter $\lambda_{\min}$ specifies the minimum separation of the change point parameters θ_k. Every segment of length $n_k := n(\theta_k - \theta_{k-1}),\ k = 1..\kappa + 1$ with $\theta_0 := 0,\ \theta_{\kappa+1} := 1$ was generated with $\alpha_k,\ k = 0..\kappa + 1$, and using $\mathcal{U}_1$ and $\mathcal{U}_2$. Note that by this method of data generation, the single-dimensional marginals are the same throughout the sequence, and the process distributions are not mixing. Figure To the best of our knowledge, all of the existing non-parametric methods are bound to fail in this generality. Thus, we cannot compare our methods against any other algorithm. To evaluate the performance of Algorithm 1, we provided the correct number $\kappa = 5$ of change points, however, the parameter $\lambda_{\min}$ was unknown to the algorithm. We calculated the error rate as

$$\sum_{k=1}^{\kappa} |\widehat{\theta}_k - \theta_k|. \tag{5.1}$$

Figure 5.1 demonstrates the average estimation error-rate of Algorithm 1 as a function of the sequence length n.

As for our experiments with Algorithms 2 and 3, we fixed three parameters $\alpha_1 := 0.12..$, $\alpha_2 := 0.13..$ and $\alpha_3 := 0.14..$ (with long mantissae) to correspond to $r = 3$ different process distributions. To produce $\boldsymbol{x} \in \mathbb{R}^n$ we randomly generated $\kappa := 5$ change points $\theta_k,\ k = 1..\kappa$ with the property that their minimum separation defined by (3.2) was at least 0.1, i.e. $\lambda_{\min} := 0.1$. Every segment of length $n_k := n(\theta_k - \theta_{k-1}),\ k = 1..\kappa + 1$ with $\theta_0 := 0,\ \theta_{\kappa+1} := 1$ was generated with $\alpha_{k'}$ and n_k where $k' := k[r \bmod ,]\ k = 0..\kappa + 1$. Note that by this approach and for this choice of κ the first and the last segments are generated by the same process distribution. We set $\lambda := 0.6\lambda_{\min}$, and as in the previous experiment, we used uniform distributions $\mathcal{U}_1$

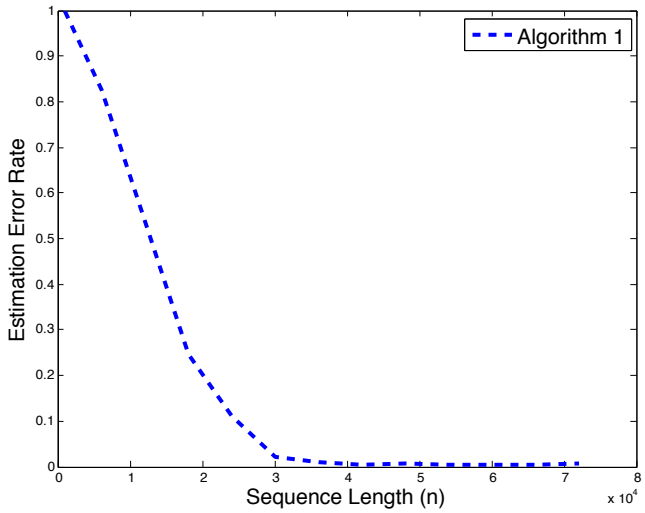

Figure 5.1: Average (over 50 runs) error of Alg1$(\boldsymbol{x}, \kappa)$, $\boldsymbol{x} \in \mathbb{R}^n$, as a function of n, where $\kappa := 3$, $\lambda_{\min} := 0.1$ and $\boldsymbol{x}$ is generated by 4 process distributions corresponding to $\alpha_1 := 0.12..$, $\alpha_2 := 0.14..$, $\alpha_3 := 0.16..$, $\alpha_4 := 0.18$, with $\mathcal{U}_1$ and $\mathcal{U}_2$ over $[0, 0.7]$ and $[0.3, 1]$ respectively.

and $\mathcal{U}_2$ over $[0, 0.7]$ and $[0.3, 1]$. Since Algorithm 2 is a list-estimator in the sense of Definition 3.2.1, it makes no attempt to estimate κ. It simply generates a sorted list of estimates, whose first κ elements converge to the true change points. Therefore, we calculate the error rate of the list-estimator on the first κ elements of its output (using 5.1), ignoring the rest of the candidate estimates produced by the algorithm. On the other hand, since Algorithm 3 is expected to also estimate κ we calculate its error as

$$\mathbb{I}\{|\mathcal{C}| \neq \kappa\} + \mathbb{I}\{|\mathcal{C}| = \kappa\} \sum_{k=1}^{\kappa} |\widehat{\theta}_k - \theta_k|.$$

Figure 5.2 demonstrates the average estimation error-rates of Algorithms 2 and 3, as a function of the sequence length n. As shown in the figure, the error rate of both algorithms tend to zero with increasing sequence length.

The proposed list-estimator (i.e. Algorithm 2) takes a parameter $\lambda \in (0, 1)$ as a lower-bound on $\lambda_{\min}$. In this experiment we show that this lower bound need not be tight. In particular, there is a rather large range of $\lambda \leqslant \lambda_{\min}$ for which the estimation error is low. To demonstrate this, we fixed the sequence length $n = 20000$ and generated a sequence $\boldsymbol{x} \in \{0, 1\}^n$ with $\kappa := 3$ change points, generated by $r := 4$ different

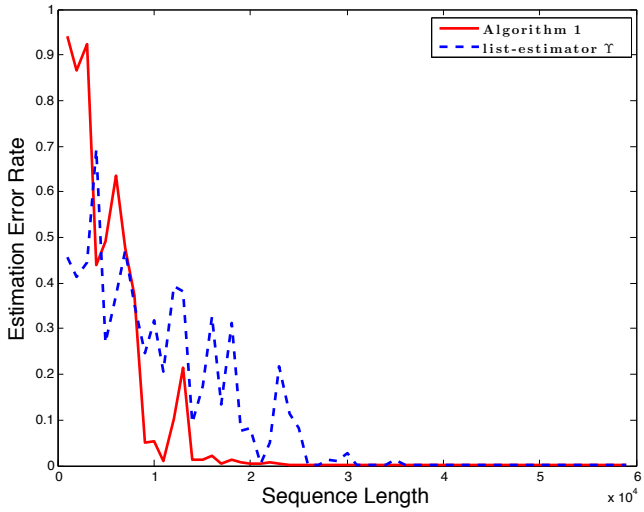

Figure 5.2: Average (over 40 runs) error rates of our algorithms, namely, Alg2$(\boldsymbol{x}, \lambda)$ and Alg3$(\boldsymbol{x}, \lambda, r)$ as a function of the length n of the input sequence $\boldsymbol{x} \in \mathbb{R}^n$, where $\boldsymbol{x}$ has $\kappa = 4$ change points and is generated by $r = 3$ distributions. The change point parameters have minimum separation $\lambda_{\min} := 0.1$. We provide. $\lambda := 0.6\lambda_{\min}$.

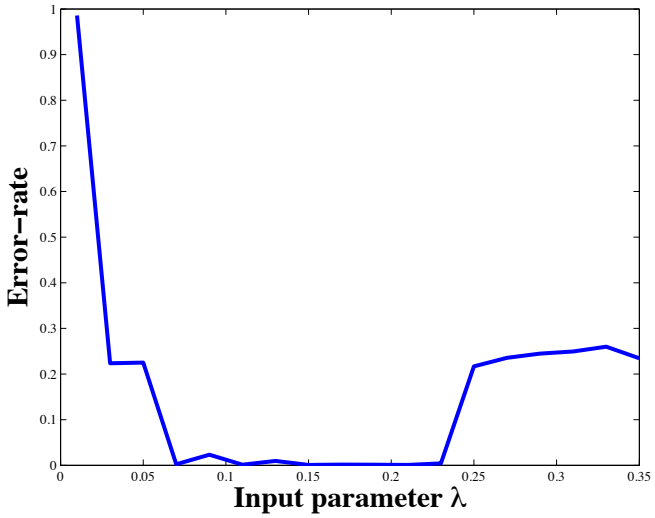

Figure 5.3: Average error-rate of Alg 2$(\boldsymbol{x}, \lambda)$ as a function the input parameter λ varied between 0.01..0.35, where the sequence length n is fixed to 20000. The input sequence has $\kappa = 3$ change points and is generated by $r = 4$ different distributions. The change point parameters have minimum separation $\lambda_{\min} := 0.23$.

process distributions, corresponding to $\alpha_1 := 0.30...$, $\alpha_2 := 0.35...$, $\alpha_3 := 0.40...$ and $\alpha_4 := 0.45....$ The minimum separation of the change point parameters was set to $\lambda_{\min} := 0.23$. We observed the error-rate of the algorithm as the input parameter λ varied between 0.01..0.35. Figure 5.3 shows the average error-rate as a function of λ.

5.3 Time series clustering

5.3.1 Synthetic data

For the purpose of our experiments, first we fix $\kappa := 5$ different (binary-valued) process distributions specified by $\alpha_1 = 0.31...$, $\alpha_2 = 0.33...$, $\alpha_3 = 0.35...$, $\alpha_4 = 0.37...$, $\alpha_5 = 0.39$. The parameters α_i are intentionally selected to be close, in order to make the process distributions harder to distinguish. Next we generate an $N \times M$ data matrix $\mathbf{X}$, each row of which is a sequence generated by one of the process distributions. Our task in both the online and the batch setting is to cluster the rows of $\mathbf{X}$ into $\kappa = 5$ clusters.

5.3.1.1 Batch Setting

In this experiment we demonstrate that in the batch setting, the clustering errors corresponding to both the online and the offline algorithms converge to 0 as the sequence-lengths grow. To this end, at every time-step t we take an $N \times n(t)$ sub-matrix $\mathbf{X}|_{\mathbf{n(t)}}$ of $\mathbf{X}$ composed of the rows of $\mathbf{X}$ terminated at length $n(t)$, where $n(t) = 5t$. Then at each iteration we let each of the algorithms, (online and offline) cluster the rows of $\mathbf{X}|_{\mathbf{n(t)}}$ into five clusters, and calculate the clustering error-rate of each algorithm. As shown in Figure 5.4 the error-rate of both algorithms decrease with sequence-length.

5.3.1.2 Online Setting

In this experiment we demonstrate that, unlike the online algorithm, the offline algorithm is consistently confused by the new sequences arriving at each time step in an online setting. To simulate an online setting, we proceed as follows: At every time-step t, a triangular window is used to reveal the first $1..n_i(t)$, $i = 1..t$ elements of the first t rows of the data-matrix $\mathbf{X}$, with $n_i(t) := 5(t - i) + 1$, $i = 1..t$. This gives a total of t sequences, each of length $n_i(t)$, for $i = 1..t$, where the i^{th} sequence for $i = 1..t$

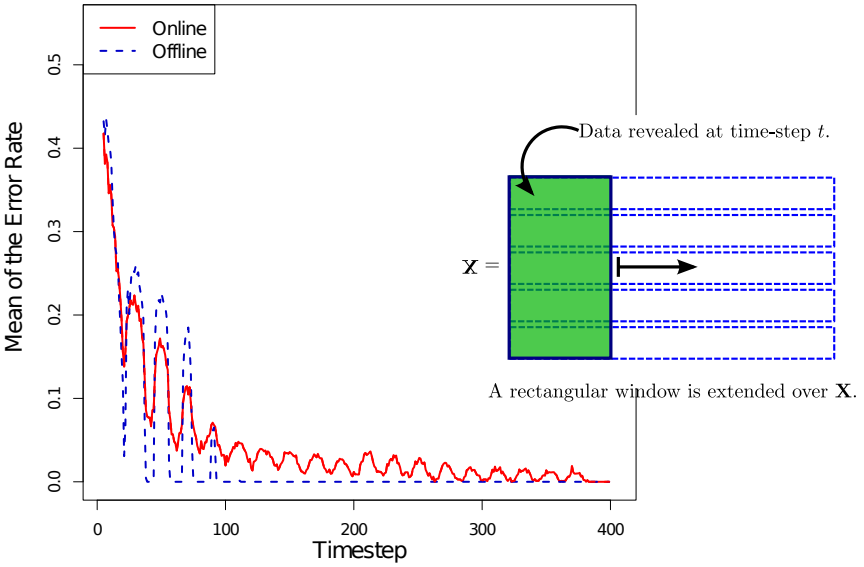

Figure 5.4: Error-rate (averaged over 100 runs) vs. sequence length in offline setting.

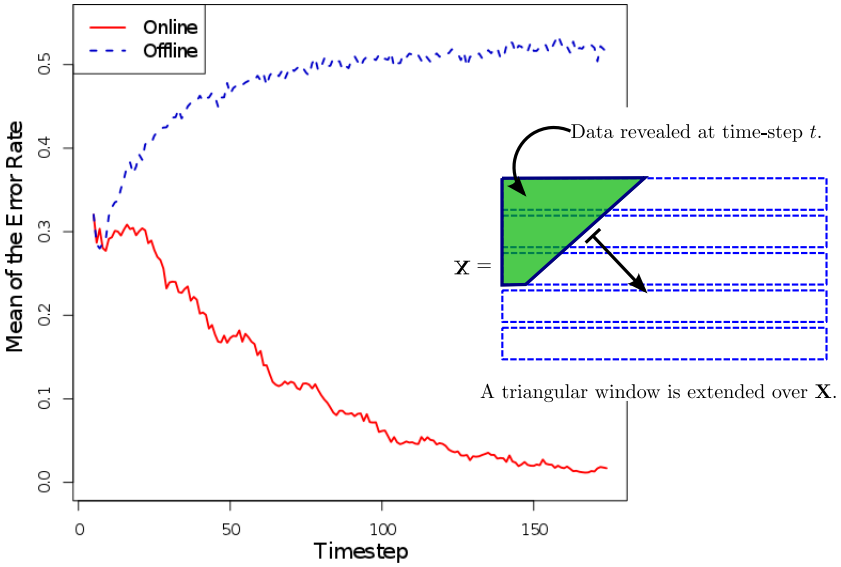

Figure 5.5: Error-rate (averaged over 100 runs) vs. number of observed samples in online setting.

corresponds to the i^{th} row of $\mathbf{X}$ terminated at length $n_i(t)$. At every time-step t the online and offline algorithms are each used to in turn cluster the observed t sequences into five clusters. As shown in Figure 5.5, in this setting the clustering error-rate of the offline algorithm remains consistently high, whereas that of the online algorithm converges to zero.

5.3.2 Real data: motion clustering

As a real application we consider the problem of clustering motion capture sequences, where groups of sequences with similar dynamics are to be identified. Data is taken from the Motion Capture database (MOCAP) (cmu) which consists of time series data representing human locomotion. The sequences are composed of marker positions on human body which are tracked spatially through time for various activities.

We compare the results against two other methods, namely those of (Li and Prakash, 2011) and (Jebara et al., 2007), which (to the best of our knowledge) constitute the state-of-the-art performance on these datasets. Note that we have not implemented these reference methods, rather we have taken their numerical results directly from their corresponding articles. In order to have common grounds for each comparison we use the same sets of sequences,[1] and the same means of evaluation as those used by (Li and Prakash, 2011, Jebara et al., 2007).

In the paper by (Li and Prakash, 2011) two MOCAP datasets[2] are used, where the sequences in each dataset are labeled with either running or walking as annotated in the database. Performance is evaluated via the conditional entropy S of the true labeling with respect to the prediction i.e., $S = -\sum_{i,j} \frac{\mathcal{M}_{ij}}{\sum_{i',j'} \mathcal{M}_{i'j'}} \log \frac{\mathcal{M}_{ij}}{\sum_{j'} \mathcal{M}_{ij'}}$ where $\mathcal{M}$ denotes the clustering confusion matrix. The motion sequences used by (Li and Prakash, 2011) are reportedly trimmed to equal duration. However, we use the original sequences as our method is not limited by variation in sequence lengths. Table 5.1 lists performance of Algorithm 4 as well as that reported for the method of (Li and Prakash, 2011); Algorithm 4 performs consistently better.

In the paper by (Jebara et al., 2007) four MOCAP datasets[3] are used, corresponding to four motions: run, walk, jump and forward jump. Table 5.2 lists performance in terms of accuracy. The datasets in Table 5.2 constitute two types of motions.

[1] marker position: the subject's right foot.
[2] subjects #16 and #35.
[3] subjects #7, #9, #13, #16 and #35.

1. Motions that can be considered ergodic (walk, run, run/jog; displayed above the double line), and

2. Non-ergodic motions (single jumps; displayed below the double line).

As shown in Table 5.2, Algorithm 4 achieves consistently better performance on the first group of datasets, while being competitive (better on one and worse on another) on the non-ergodic motions. The time taken to complete each task is in the order of few minutes on a standard laptop computer.

	Dataset	**(Li and Prakash, 2011)**	Algorithm 4
1.	Walk vs. Run (#35)	0.1015	**0**
2.	Walk vs.Run (#16)	0.3786	**0.2109**

Table 5.1: Comparison with (Li and Prakash, 2011): Performance in terms of entropy; data-sets concern ergodic motion captures.

	Dataset	**(Jebara et al., 2007)**	Algorithm 4
1.	Run(#9) vs. Run/Jog(#35)	**100%**	**100%**
2.	Walk(#7) vs. Run/Jog(#35)	95%	**100%**
3.	Jump vs. Jump fwd.(#13)	87%	**100%**
4.	Jump vs. Jump fwd.(#13, 16)	**66%**	60%

Table 5.2: Comparison with (Jebara et al., 2007): Performance in terms of accuracy; Rows 1 & 2 concern ergodic, Rows 3 & 4 concern non-ergodic motion captures.

Chapter 6

Discussion

This thesis demonstrates the existence of consistent, non-parametric, sequential learning methods for highly dependent time series. We have considered two classical unsupervised learning problems, namely, change point estimation, and clustering. For each problem we have proposed natural formulations which we have further shown to admit consistent solutions under the assumption that the data are generated by stationary ergodic process distributions. The considered framework is extremely general as it does not impose any assumptions on the data beyond stationarity and ergodicity, allowing the data to be otherwise arbitrarily generated by unknown process distributions, with no parametric assumptions or prescribed conditions on the dependence between the samples. Specifically, no independence, finite-memory or mixing conditions are required. As a result, it is well-suited for unsupervised learning problems, where the learner's objective is to infer the underlying structure in the data, while the nature of the data are completely unknown. This work, is a first step towards a new type of theoretical approach to the analysis of sequential learning methods, leaving open many interesting questions, and laying grounds for interesting future research directions to be explored. In this chapter we discuss some open problems and potential future directions.

6.1 Change point analysis

In Chapter 3 we provided three formulations for the estimation of multiple change points in stationary ergodic time series. To the best of our knowledge, we are the first to consider the change point problem to this extent of generality. Indeed, many

interesting new problems are left as future research for multiple change point analysis.

6.1.1 Change point detection (online formulation)

The algorithms proposed in this work address the retrospective detection of change points. In many applications it may be interesting to consider the change point detection problem in the case where the samples arrive in an online fashion. While a large body of work exists on online change point detection in time series that satisfy independence or strong mixing conditions, in the general stationary ergodic framework considered in this thesis, it is impossible to obtain consistent change point detectors. In fact, in light of the impossibility results that exist for the stationary ergodic framework, even the change point estimation problem considered in this thesis seemed impossible to solve at first glance. Our main challenge was to devise formulations that admit consistent solutions without the need to impose any statistical assumptions beyond stationarity and ergodicity. A natural and interesting direction would be discover formulations which allow for the online detection of change points in stationary ergodic time series.

6.1.2 Allowing some segments to have sub-linear lengths

In our formulations we assume that the change point parameters have an unknown minimum separation $\lambda_{\min} \in (0, 1)$. As a result, the segments are assumed to be of length at least θn, where n is the length of the entire sequence. As discussed earlier, this is a standard assumption even in the case where the samples are independently and identically distributed within each segment. While in general this assumption is inevitable in the case of stationary ergodic time series, it may be interesting to discover the formulations under which this linearity condition can be relaxed, hence allowing for some segments to have sub-linear lengths. Such finite-length segments would be analogous to the so-called "bad" sequences in the clustering problem considered in Chapter 4.

6.2 Online clustering

In Chapter 4, we presented an asymptotically consistent, online time series clustering algorithm for data generated by stationary ergodic process distributions. This approach lays grounds for an interesting new area of research, namely the online analysis of stationary ergodic time series. Many open directions are left to be explored. In this section we provide some interesting extensions of the clustering problem formulation considered in this work.

6.2.1 Online hierarchical clustering

As discussed earlier, in the general framework considered in this thesis it is provably impossible to estimate the number of clusters, even in the offline setting. An alternative result to obtain would be to produce a hierarchical clustering tree such that the ground-truth is some pruning of this tree, and call a hierarchical clustering algorithm asymptotically consistent if it achieves this goal in asymptotic. Observe that every batch of sequences from some time on possesses the so-called strict separation property: sequences in the same target cluster (in this case generated by the same process distribution) are closer to each other than to the sequences in other target clusters. Thus, from the results of (M. et al., 2008) it readily follows that linkage-based clustering algorithms are asymptotically consistent in the offline formulation of the problem. An interesting new direction would be the extension to the online setting. That is, similarly to our definition for asymptotic consistency in this thesis, an online hierarchical clustering algorithm can be called asymptotically consistent, if for every fixed batch of sequences from some time on, some pruning of the target clustering tree for that batch coincides with the produced hierarchy confined to the batch. Unlike in the batch setting, this extension is not immediate for the online formulation.

6.2.2 An even more relaxed setting

Our formulation of the online time-series clustering problem corresponds to the case where the length of each sequence grows to infinity with time. This assumption may not be very practical in reality. It would be interesting to consider the situation where a portion of the sequences stop growing. Recall that even under the assumption that all of the sequences grow indefinitely, at a given time-step some sequences may not be

long enough to be appropriate representatives of the process distributions that generate them. In Chapter 4 we called these sequences "bad" points, and made sure that our online algorithm is robust with respect to their potential existence at every time step. However, the current algorithm relies on the fact that every batch of sequences, from some time on, can be consistently clustered. A modified version of the algorithm could be developed to address the case where only a portion of every given batch of sequences grows with time.

6.2.3 A bandit-like formulation of clustering

Another interesting formulation to consider would be one where the algorithm has a bandit-like control over the evolution of the data. That is, at each time step t the algorithm can request a sample from one of the $N(t-1)$ sequences observed, or a new sample. The objective remains the same: to cluster the sequences based on their process distributions; however, the algorithm is required to achieve this goal while requesting as few samples as possible. This formulation can be suitable for both online and offline settings. In the offline setting, the number N of samples would be fixed, and the algorithm would have control over the growth of the individual sequences.

6.3 Extensions of the general framework

6.3.1 Other distance metrics

The methods presented in this thesis are based on the empirical estimates of the distributional distance, d. An interesting direction would be to see how our methods could be generalized so that other distances or classes of distances could be used instead of the distributional distance. A necessary property of a distance to be used in our framework is that it can be consistently estimated for stationary ergodic process distributions. Although not many such distance functions are known, some examples exist in the literature. For instance, the telescope distance, recently proposed by (Ryabko and Mary, 2012) has this property. It is based on a generalization of the Kolmogorov-Smirnov distance. As discussed earlier, Kolmogorov-Smirnov type distance functions have been used in the literature for change point analysis. Thus, we conjecture that the telescope distance or other generalizations of the Kolmogorov-Smirnov distance may

also prove useful in our formulations. Note however, that replacing the distributional distance with other distances in our methods is not straight forward. This is due to the fact that unlike for the problems considered by (Ryabko and Mary, 2012), asymptotic consistency of the distance is necessary but not sufficient for some of our algorithms to admit consistent solutions. This in particular concerns our methods for change point analysis.

6.3.2 Rates of convergence

The main focus of this work is on the most general case of stationary ergodic time series. In this generality, rates of convergence are provably impossible to obtain. Therefore, the algorithms developed for this framework are forced not to rely on rates of convergence, and as such are applicable to a wide range of situations. Perhaps, the only drawback of this general framework is that finite-sample performance guarantees are impossible to obtain and the optimality of the algorithms cannot be argued. An interesting direction would be to combine the asymptotic results of this thesis with rates of convergence in the settings where they are possible to obtain. Specifically, it would be interesting to obtain error rates and finite-time performance guarantees for our algorithms in more restrictive settings, for example, for the case where the time series satisfy mixing conditions. We conjecture that our methods are optimal (up to some constant factors) in such settings as well, even though they are clearly suboptimal under parametric assumptions. Moreover, the general question of what optimal performance guarantees can be obtained for different classes of time series remains open. For instance, it would be interesting to discover the minimal achievable probability of error for clustering algorithms addressing finite sets of samples of even independent and identically distributed observations, and to develop some algorithms to attain optimal performance.

Bibliography

Carnegie Mellon University (CMU) graphics lab motion capture database. *cited on page(s):* 118

Dimitris Achlioptas and Frank McSherry. On spectral learning of mixtures of distributions. In *Learning Theory*, pages 458–469. Springer, 2005. *cited on page(s):* 36

T.M. Adams and A.B. Nobel. On density estimation from ergodic processes. *The Annals of Probability*, 26(2):pp. 794–804, 1998. ISSN 00911798. *cited on page(s):* 30

T.M. Adams and A.B. Nobel. Uniform approximation of vapnik–chervonenkis classes. *Bernoulli*, 18(4):1310–1319, 2012. *cited on page(s):* 30

Leman Akoglu and Christos Faloutsos. Event detection in time series of mobile communication graphs. In *Proceedings of the of Army Science Conference*, pages 1–8, 2010. *cited on page(s):* 21

P. Algoet. Universal schemes for prediction, gambling and portfolio selection. *The Annals of Probability*, 20(2):901–941, 1992. *cited on page(s):* 30

Paul Algoet. Universal schemes for learning the best nonlinear predictor given the infinite past and side information. *Information Theory, IEEE Transactions on*, 45 (4):1165–1185, 1999. *cited on page(s):* 30

M.F. Balcan and P. Gupta. Robust hierarchical clustering. In *The 23rd Annual Conference on Learning Theory (COLT)*, 2010. *cited on page(s):* 36

M. Basseville and I.V. Nikiforov. *Detection of abrupt changes: theory and application*. Prentice Hall information and system sciences series. Prentice Hall, 1993. *cited on page(s):* 32

C. Biernacki, G. Celeux, and G. Govaert. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):719–725, 2000. *cited on page(s):* 36

P. Billingsley. Statistical inference about Markov chains. *Annals of Mathematical Statisics*, 32(1):12–40, 1961. *cited on page(s):* 19, 38, 112

P. Billingsley. *Ergodic theory and information*. Wiley, New York, 1965. *cited on page(s):* 42

R. Bolton and D. Hand. Statistical fraud detection: A review. *Statistical Science*, 17: 2002, 2002. *cited on page(s):* 21

N. Bouguila and D. Ziou. Online clustering via finite mixtures of dirichlet and minimum message length. *Engineering Applications of Artificial Intelligence*, 19(4):371–379, 2006. *cited on page(s):* 36

B. Brodsky and B. Darkhovsky. *Non-parametric methods in change-point problems*. Mathematics and its applications. Kluwer Academic Publishers, 1993. *cited on page(s):* 32

B. Brodsky and B. Darkhovsky. *Non-parametric statistical diagnosis: problems and methods*, volume 509. Kluwer Academic Pub, 2000. *cited on page(s):* 32

B. Brodsky and B. Darkhovsky. Sequential change-point detection for mixing random sequences under composite hypotheses. *Statistical Inference for Stochastic Processes*, 11(1):35–54, 2008. *cited on page(s):* 32

I. Cadez, S. Gaffney, and P. Smyth. A general probabilistic framework for clustering individuals and objects. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 140–149. ACM, 2000. *cited on page(s):* 36

E. Carlstein. Non-parametric Change-Point Estimation. *The Annals of Statistics*, 16 (1):188–197, 1988. *cited on page(s):* 33

E. Carlstein and S. Lele. Non-parametric change-point estimation for data from an ergodic sequence. *Teorya Veroyatnostei i ee Primeneniya*, 38:910–917, 1993. *cited on page(s):* 34

N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006. ISBN 0521841089. *cited on page(s):* 31, 104

J. Chen. *Parametric statistical change point analysis*. Birkhauser Boston, 2012. *cited on page(s):* 32

I. Csiszar and P.C. Shields. Notes on information theory and statistics. In *Foundations and Trends in Communications and Information Theory*, 2004. *cited on page(s):* 30, 42

M. Csörgö and L. Horváth. *Limit Theorems in Change-Point Analysis (Wiley Series in Probability & Statistics)*. January 1998. *cited on page(s):* 32

S. Dasgupta. Learning mixtures of gaussians. In *40th Annual Symposium on the Foundations of Computer Science*, pages 634–644, 1999. *cited on page(s):* 36

L Dumbgen. The asymptotic behavior of some non-parametric change-point estimators. *The Annals of Statistics*, pages 1471–1495, 1991. *cited on page(s):* 33

E.B. Fox, E.B. Sudderth, M.I. Jordan, and A.S. Willsky. A Sticky HDP-HMM with Application to Speaker Diarization. *Annals of Applied Statistics*, 5(2A):1020–1056, 2011. *cited on page(s):* 21

L. Giraitis, R. Leipus, and D. Surgailis. The change-point problem for dependent observations. *Journal of Statistical Planning and Inference*, pages 1–15, 1995. *cited on page(s):* 33

R. Gray. *Probability, Random Processes, and Ergodic Properties*. Springer Verlag, 1988. *cited on page(s):* 27, 43

S. Hariz, J. Wylie, and Q. Zhang. Optimal rate of convergence for non-parametric change-point estimators for nonstationary sequences. *Annals of Statistics*, 35(4): 1802–1826, 2007. *cited on page(s):* 32

A. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8): 651–666, 2010. *cited on page(s):* 35

T. Jebara, Y. Song, and K. Thadani. Spectral clustering and embedding with hidden markov models. *European Conference on Machine Learning (ECML) 2007*, pages 164–175, 2007. *cited on page(s):* 118, 119

A. Khaleghi and D. Ryabko. Locating changes in highly-dependent data with unknown number of change points. In *Neural Information Processing Systems (NIPS)*, Lake Tahoe, Nevada, United States, 2012. *cited on page(s):* 49

A. Khaleghi and D. Ryabko. Non-parametric multiple change point estimation in highly dependent time series. In *Proceedings of the 24th International Conference on Algorithmic Learning Theory (ALT'13)*, Singapore, 2013. *cited on page(s):* 49

A. Khaleghi and D. Ryabko. Asymptotically consistent estimation of the number of change points in highly dependent time series. In *Proceedings of the 31st International Conference on Machine Learning (ICML 2014)*, Beijing, China, 2014. *cited on page(s):* 49

A. Khaleghi, D. Ryabko, J. Mary, and P. Preux. Online clustering of processes. In *the international conference on Artificial Intelligence & Statistics (AI & Stats)*, pages 601–609, La Palma, Canary Islands, 2012. *cited on page(s):* 95

J. Kleinberg. An impossibility theorem for clustering. In *15th Conference Neiral Information Processing Systems (NIPS'02)*, pages 446–453, Montreal, Canada, 2002. *cited on page(s):* 35

P. Kokoszka and R. Leipus. Detection and estimation of changes in regime. *Long-Range Dependence: Theory and Applications*, pages 325–337, 2002. *cited on page(s):* 32, 35

M. Kumar, N.R. Patel, and J. Woo. Clustering seasonality patterns in the presence of errors. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 557–563. ACM, 2002. *cited on page(s):* 36

M. Lavielle. Detection of multiple changes in a sequence of dependent variables. *Stochastic Processes and their Applications*, 83(1):79–102, 1999. *cited on page(s):* 34, 53

M. Lavielle. Using penalized contrasts for the change-point problem. *Signal Processing*, 85(8):1501 – 1510, 2005. *cited on page(s):* 35, 53

M. Lavielle and G. Teyssiere. Adaptive detection of multiple change-points in asset price volatility. In *Long memory in economics*, pages 129–156. Springer, 2007. *cited on page(s):* 21, 34

E. Lebarbier. Detecting multiple change-points in the mean of gaussian process by model selection. *Signal Processing*, 85(4):717 – 736, 2005. *cited on page(s):* 34, 53

C. Lévy-Leduc and F. Roueff. Detection and localization of change-points in high-dimensional network traffic data. *The Annals of Applied Statistics*, pages 637–662, 2009. *cited on page(s):* 21, 34

C. Li and G. Biswas. Applying the hidden markov model methodology for unsupervised learning of temporal data. *International Journal of Knowledge Based Intelligent Engineering Systems*, 6(3):152–160, 2002. *cited on page(s):* 36

L. Li and B.A. Prakash. Time series clustering: Complex is simpler! In *the 28th International Conference on Machine Learning (ICML'11)*, 2011. *cited on page(s):* 118, 119

A. Lung-Yut-Fong, C. Lévy-Leduc, and O. Cappé. Robust retrospective multiple change-point estimation for multivariate data. In *IEEE Statistical Signal Processing Workshop (SSP)*, pages 405–408, 2011. *cited on page(s):* 34

A. Lung-Yut-Fong, C. Lévy-Leduc, and O. Cappé. Distributed detection/localization of change-points in high-dimensional network traffic data. *Statistics and Computing*, 22(2):485–496, March 2012. *cited on page(s):* 21

Balcan M., Blum A., and Vempala S. A discriminative framework for clustering via similarity functions. In *the proceedings of the ACM Symposium on Theory of Computing (STOC)*, 2008. *cited on page(s):* 23, 36, 98, 123

M. Mahajan, P. Nimbhorkar, and K. Varadarajan. The planar k-means problem is np-hard. In *WALCOM '09: Proceedings of the 3rd International Workshop on Algorithms and Computation*, pages 274–285, Berlin, Heidelberg, 2009. Springer-Verlag. *cited on page(s):* 35

P. Massart. A non asymptotic theory for model selection. In *European Congress of Mathematics*, pages 309–323, 2005. *cited on page(s):* 53

P. McCullagh and J. Yang. How many clusters? *Bayesian Analysis*, 3(1):101–120, 2008. *cited on page(s):* 36

G. Morvai, S. Yakowitz, and L. Gyorfi. Non-parametric inference for ergodic, stationary time series. *Annals of Statistics*, 24(1):370–379, 1996. *cited on page(s):* 30

G. Morvai, S. Yakowitz, and P. Algoet. Weakly convergent non-parametric forecasting of stationary time series. *Information Theory, IEEE Transactions on*, 43(2):483 –498, March 1997a. ISSN 0018-9448. doi: 10.1109/18.556107. *cited on page(s):* 30

G. Morvai, S. Yakowitz, and P. Algoet. Weakly convergent non-parametric forecasting of stationary time series. *Information Theory, IEEE Transactions on*, 43(2):483–498, 1997b. *cited on page(s):* 30

H. Müller and D. Siegmund. Change-point problems. Ims, 1994. *cited on page(s):* 32

A. Nobel. Hypothesis testing for families of ergodic processes. *Bernoulli*, 12(2):251–269, 2006. *cited on page(s):* 30

D. Ornstein. *Ergodic Theory, Randomness, and Dynamical Systems*. Yale Mathematical Monographs. Yale University Press, 1974. *cited on page(s):* 46

Antonello Panuccio, Manuele Bicego, and Vittorio Murino. A hidden markov model-based approach to sequential data clustering. pages 734–742. Springer, 2002. *cited on page(s):* 36

P Papantoni-Kazakos and Anthony Burrell. Robust sequential algorithms for the detection of changes in data generating processes. *Journal of Intelligent & Robotic Systems*, 60(1):3–17, 2010. *cited on page(s):* 32

F. Picard, S. Robin, M. Lavielle, C. Vaisse, and J. Daudin. A statistical approach for array cgh data analysis. *BMC bioinformatics*, 6(1):27, 2005. *cited on page(s):* 21

B. Ryabko. Prediction of random sequences and universal coding. *Problems of Information Transmission*, 24:87–96, 1988. *cited on page(s):* 30, 31

D. Ryabko. Clustering processes. In *Proceedings of the the 27th International Conference on Machine Learning (ICML 2010)*, pages 919–926, Haifa, Israel, 2010a. *cited on page(s):* 27, 28, 29, 30, 35, 36, 38, 43, 44, 67, 68, 96, 98, 101, 103, 111

D. Ryabko. Discrimination between B-processes is impossible. *Journal of Theoretical Probability*, 23(2):565–575, 2010b. *cited on page(s):* 20, 21, 30, 39, 46, 56, 97, 101

D. Ryabko. Sequence prediction in realizable and non-realizable cases. In *Proceedings of the the 23rd Conference on Learning Theory (COLT 2010)*, pages 119–131, Haifa, Israel, 2010c. *cited on page(s):* 30

D. Ryabko. On the relation between realizable and non-realizable cases of the sequence prediction problem. *Journal of Machine Learning Research (JMLR)*, 12:2161–2180, 2011. *cited on page(s):* 31

D. Ryabko. Testing composite hypotheses about discrete ergodic processes. *Test*, 21 (2):317–329, 2012. *cited on page(s):* 27, 30

D Ryabko and J. Mary. Reducing statistical time-series problems to binary classification. In *Neural Information Processing Systems (NIPS)*, pages 2069–2077, Lake Tahoe, Nevada, United States, 2012. *cited on page(s):* 124, 125

D. Ryabko and B. Ryabko. Non-parametric statistical inference for ergodic processes. *IEEE Transactions on Information Theory*, 56(3):1430–1435, 2010. *cited on page(s):* 27, 30, 34, 54

Z. Shi and G. Joydeep. A unified framework for model-based clustering. *Journal of Machine Learning Research*, 4:1001–1037, 2003. *cited on page(s):* 36

P. Shields. *The Ergodic Theory of Discrete Sample Paths*. AMS Bookstore, 1996. *cited on page(s):* 20, 38, 46, 112

P. Smyth. Clustering sequences with hidden Markov models. In *Advances in Neural Information Processing Systems*, pages 648–654. MIT Press, 1997. *cited on page(s):* 36

R. Solomonoff. Complexity-based induction systems: comparisons and convergence theorems. *IEEE Transactions on Information Theory*, IT-24:422–432, 1978. *cited on page(s):* 31

G. Stephen. *String searching algorithms*. World Scientific publishing company, 1994. *cited on page(s):* 28

M. Sudan. List decoding: Algorithms and applications. In *Theoretical Computer Science: Exploring New Frontiers of Theoretical Informatics*, pages 25–41. Springer, 2000. *cited on page(s):* 23

M. Talih and N. Hengartner. Structural learning with time-varying components: tracking the cross-section of financial time series. *Journal of the Royal Statistical Society Series B*, 67(3):321–341, 2005. *cited on page(s):* 21

A. Tartakovsky, B. Rozovskii, R. Blazek, and H. Kim. A novel approach to detection of intrusions in computer networks via adaptive sequential and batch-sequential change-point detection methods. *IEEE Transactions on Signal Processing*, 54(9):3372–3382, 2006. *cited on page(s):* 21

E. Ukkonen. On-line construction of suffix trees. *Algorithmica*, 14(3):249–260, 1995. *cited on page(s):* 45, 107

J. Vert and K. Bleakley. Fast detection of multiple change-points shared by many signals using group lars. In *NIPS*, pages 2343–2351, 2010. *cited on page(s):* 21, 34

L. Vostrikova. Detecting disorder in multidimensional random processes. *Soviet Mathematics Doklady*, 24:55–59, 1981. *cited on page(s):* 34

Y. Yao. Estimating the number of change-points via schwarz'criterion. *Statistics & Probability Letters*, 6(3):181–189, 1988. *cited on page(s):* 34, 53