THÈSE de DOCTORAT
Opérée au sein de :
l'Université Lille

Ecole Doctorale MADIS-631
Collaboration Cifre Worldline - Inria
Laboratoire Paul Painlevé - Modal (Inria)

Spécialité de doctorat : Mathématiques et leurs interactions

présentée par:

Etienne KRÖNERT

Anomaly detection in time series using breakpoint detection and multiple testing

Détection d'anomalies dans les séries temporelles grâce à la détection de ruptures et aux tests multiples

---

dirigée par Cristian PREDA et Alain CELISSE

Thèse soutenue le 2 octobre 2024, devant le jury composé de:

Madalina OLTEANU Professeure, Université Paris Dauphine-PSL	**Rapporteure et présidente**
Guillem RIGAILL Directeur de recherche, INRAE	**Rapporteur**
Stéphane ROBIN Professeur, Sorbonne université	**Examinateur**
Etienne ROQUAIN Maître de conférence (HDR), Sorbonne université	**Examinateur**
Cristian PREDA Professeur, Université de Lille	**Directeur de thèse**
Alain CELISSE Professeur, Panthéon-Sorbonne Université	**Co-Directeur de thèse**
Dalila HATTAB Worldline	**Membre invitée**

Résumé français

L'objectif de ce travail est de développer de nouvelles méthodes dans le domaine de la détection d'anomalies. Un détecteur d'anomalies a pour but d'identifier les points de données qui ont été générés par un processus différent de celui de référence. Généralement, les détecteurs d'anomalies sont entraînés sur un historique et ne prennent en compte que la loi de référence de l'ensemble d'entraînement. Ou alors, le détecteur est mis à jour en temps réel à partir d'une fenêtre glissante de longueur fixe. Aucune de ces approches ne tient compte de la dynamique réelle de la série temporelle, ce qui conduit à générer des faux positifs lorsque la loi de référence change.

L'approche que l'on propose consiste à détecter en amont ces changements de lois à l'aide de détecteurs de ruptures. Les anomalies sont ensuite retrouvées dans les segments homogènes qui viennent d'être identifiés. L'intérêt de cette approche pour une entreprise comme Worldline est de pouvoir détecter les incidents survenant sur son système informatique rapidement et en réduisant le nombre de fausses alertes. On prend aussi soin de contrôler théoriquement le nombre de faux positifs à travers le False Detection Rate (FDR).

Dans un premier temps, on tente de développer un nouvel estimateur de la p-valeur. Après comparaison avec l'existant, ce nouvel estimateur s'avère trop complexe sans gain réel. Par la suite, on travaille avec l'estimateur de la p-valeur empirique.

Puis, étudie théoriquement le problème de la détection d'anomalies sur des séries iid. On développe ainsi une procédure permettant le contrôle du FDR d'une série de longueur infinie en contrôlant une variante du FDR sur des sous-séries de longueur fixe. Les expériences empiriques permettent de montrer dans quelles conditions ce contrôle est atteint en pratique.

Enfin, on introduit notre nouveau détecteur d'anomalies basé sur les ruptures pour des séries iid par morceaux. On montre que la procédure de contrôle du FDR reste effective dans ce nouveau contexte. L'utilisation d'un détecteur de ruptures entraîne toutefois deux difficultés : les segments de petite longueur et les délais de détection de ruptures. Afin de traiter ces difficultés, on introduit un score de confiance. Le détecteur est évalué empiriquement dans le but de montrer la pertinence et les limites de notre approche.

English Abstract

The purpose of this thesis is to develop new methods in the field of anomaly detection. An anomaly detector should identify data points that have been generated by a process different from the reference process. In general, anomaly detectors are trained on past data and only consider the reference distribution given in the training set. Alternatively, the detector is updated in real time from a fixed length sliding window. Neither approach takes into account the true dynamics of the time series, leading to false positives when the reference distribution changes.

The proposed approach consists in detecting these distribution changes upstream, using breakpoint detectors. Anomalies are then retrieved within the identified homogeneous segments. The advantage of this approach for a company like Worldline is that it can quickly detect incidents in its system, while reducing the number of false alarms. In addition, care is taken to theoretically control the number of false positives through the False Detection Rate (FDR).

The first step is an attempt to develop a new estimator of the p-value. After comparison with existing estimators, this new estimator turns out to be too complex, with no real gain. The empirical p-value estimator is then used.

Then, the problem of anomaly detection on iid series is studied theoretically. A procedure is developed to control the FDR of a series of infinite length by controlling a variant of the FDR on subseries of fixed length. Empirical experiments show under which conditions this control is achieved in practice.

Finally, our new anomaly detector based on breakpoints is introduced for piecewise iid series. It is shown that the FDR control procedure is effective in this new context. However, the use of a breakpoint detector leads to two difficulties: small segment length and breakpoint detection delays. To overcome these difficulties, a confidence score is introduced. The detector is empirically evaluated to show the relevance and limitations of our approach.

Remerciements

En premier lieu, je remercie Madalina Olteanu et Guillem Rigaill d'avoir accepté de prendre sur leur temps pour rapporter ce manuscrit de thèse, ainsi que pour leurs remarques pertinentes qui m'ont permis d'améliorer ce travail. Je remercie également Stéphane Robin et Etienne Roquain d'avoir accepté de faire partie de mon jury.

Ensuite, je souhaite remercier Alain Celisse et Cristian Préda. Merci de m'avoir fait confiance et d'avoir accepté d'encadrer cette thèse. Merci de m'avoir fait découvrir le monde de la recherche, d'avoir pris le temps pour nos discussions scientifiques. Merci aussi pour vos observations avisées et vos précieux conseils qui m'ont aidé dans la réalisation du présent travail.

Je remercie également Dalila Hattab d'avoir accepté de m'encadrer pour ce travail à Worldline. Je te remercie d'avoir pris le temps de me challenger et de me conseiller lors de nos discussions régulières.

Je remercie Worldline et l'ANRT d'avoir cofinancé cette thèse Cifre. Je remercie les équipes administratives de l'Inria et de Worldline qui m'ont aidé dans lors des procédures administratives.

Je remercie aussi mes collègues de MODAL pour ces moments de camaraderies lors des pauses-café ou des sorties après le travail. Merci à Alain, Arthur, Axel, Camille, Christophe, Clarisse, Cristian, Eglantine, Ernesto, Filippo, Florent, François, Guillemette, Hemant, Issam, Louise, Myriam, Rachid, Rim, Vincent, Wilfried et Yaroslav.

Je remercie aussi mes collègues de CBDO Office, pour avoir apporté de la bonne humeur au travail. Merci à Céline, Christine, Dalila, Elisabeth, Faiza, Judith et Maelio.

Enfin, je tiens à remercier ma famille et mes amis qui m'ont toujours soutenu. Rien n'aurait été possible sans vous.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this introduction chapter, the different important notions such as anomaly detection, breakpoint detection, and multiple testing procedures are introduced. Then, the contributions of our thesis are positioned in relation to the literature. Finally, an overview of the different chapters is given.

1.1 A business need at the root of a machine learning problem

The motivation for this thesis originates from the industrial needs of the Worldline company. Worldline is a payment processor that processes transactions for numerous European banks. It is essential for the company to guarantee uninterrupted access to its services. An outage that prevents the delivery of payments for an extended period of time is very costly to the company in terms of reputation. To speed up incident resolution, it is important to be able to detect the occurrence of incidents as early as possible. To achieve this, various metrics are monitored in real time to detect deviations that indicate a current or future failure. Traditionally, detection thresholds are defined manually by experts and updated periodically to account for changes in the monitored metrics. However, this task is time-consuming and error-prone.

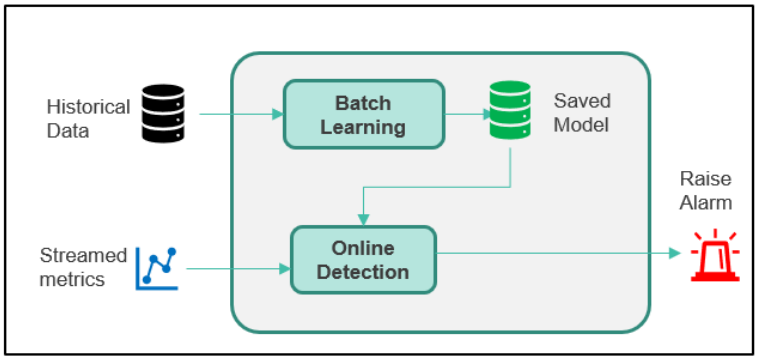


Figure 1.1: High level description of machine learning based anomaly detection.

To automate the task of setting thresholds and improve the performance of the detector, it has been proposed to use machine learning to learn the statistical behavior of the monitored metric from a history of data, as shown in Figure 1.1. The streamed metric is then compared to the normal behavior. An alert is sent if the difference between what is observed and what is predicted by the model is too large. Behind every alert that is sent, there is a team responsible for handling it. This includes investigating the causes of the potential incident and then proposing and implementing a procedure to resolve the incident. So it's not only important to detect incidents with minimal delay, but also to reduce the number of false alarms as much as possible to avoid alarm fatigue. An additional difficulty is that the incident detection system must be able to adapt in real time to changes in the monitored system. In the following sections, the problem of anomaly detection in time series is explored in more detail to understand the challenges involved.

1.2 Time series data

1.2.1 Definition and main properties

A time series is used to represent a signal measured over time. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, with Ω the set of all possible outcomes, $\mathcal{F}$ a σ-algebra on Ω and $\mathbb{P}$ a probability measure on $\mathcal{F}$. A time series of length T is a sequence of T random variables $(X_t)_{1 \leq t \leq T}$ taking values in a space noted $\mathcal{X}$. The space of observations $\mathcal{X}$ can refer to different sets. If $\mathcal{X} = \mathbb{R}$ it is a univariate time series and if $\mathcal{X} = \mathbb{R}^n$ with $n > 1$ it is a multivariate time series. One important property of time series is stationarity. A time series is said to be stationary when its statistical properties are constant over time. There are different types of stationarity [114]. Strict stationarity of order 1 is defined by all random variables in the series following the same distribution, denoted $\mathcal{P}_0$. Noting $\mathcal{P}_{X_t}$ the probability distribution of X_t:

$$\forall t \in [\![1, T]\!], \qquad \mathcal{P}_{X_t} = \mathcal{P}_{X_1} = \mathcal{P}_0 \tag{1.1}$$

This property is interesting because it reduces the number of marginal probability distributions to be estimated to one for the entire time series. Another property that facilitates the study of time series is ergodicity. Ergodicity refers to the property of a time series where the time average is equal to the average over all possible realizations [114]. This property guarantees that the statistical properties of the time series distribution, such as the mean, can be known by

observing a single realization of the time series.

$$\mathbb{E}[X_1] \overset{\text{a.s.}}{=} \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} X_t dt \tag{1.2}$$

Not all stationary time series are ergodic. For example, suppose there's a time series in which all points have the same value at each realization $X_t = X_1 \sim B(0.5)$. The series is stationary because all points follow Bernoulli's law, but if the series is averaged over one realization, you get either 0 or 1, and never the true mean of 0.5. If the random variables of the time series are independent and identically distributed (iid), the series is stationary and ergodic, but this is not a necessary property. The following shows how to deal with a time series that does not satisfy the stationarity property.

1.2.2 Trend Seasonality decomposition

A time series may contain different patterns. Seasonal patterns occur when the time series is affected by seasonal factors, such as the day of the week or the month of the year. Seasonal pattern are periodic and repeat itself overtime. The trend pattern exists when the time series increases or decreases over the long term. These phenomena are not stationary and must be understood when analyzing a time series. An additive model describes a time series as the sum of several components: seasonality (S_t), trend (T_t), and residual (R_t).

$$X_t = T_t + S_t + R_t \tag{1.3}$$

Decomposing a time series makes forecasting easier. Assuming that the seasonal component does not evolve in the short term, the seasonal component can be predicted by copying the seasonal pattern learned from historical data.

A simple method [83] for decomposing a series involves four steps. Assume a seasonality of length w.

1. Estimate the trend component using a moving average over a window of length w. $\hat{T}_t = \frac{1}{2w}\sum_{u=t-w}^{t+w} X_t$
2. Calculate the detrend series: $\tilde{X}_t = X_t - \hat{T}_t$
3. By averaging all the seasons in the detrend series, an estimate of the seasonal component over one season is obtained. This seasonal pattern is repeated identically over the entire length of the series to construct the seasonal component.
4. Removing the seasonal component from the detrend series yields the residual. $\hat{R}_t = \tilde{X}_t - \hat{S}_t$.

1.2.3 Breakpoint detection

In many situations, the time series is not stationary and its properties change over time. The breakpoint detection problem aims to find the locations of these changes, called breakpoints, and noted $(\tau_i)_{1\leq i\leq D+1}$, where D is the number of segments, subseries between two consecutive breakpoints. The convention $\tau_1 = 1$ and $\tau_{D+1} = T + 1$, which are not real breakpoints, are used to simplify the notation. An advantage of this approach is that it allows the series to be split into homogeneous sub-series that are easier to analyze. As shown in Figure 1.2, the series is

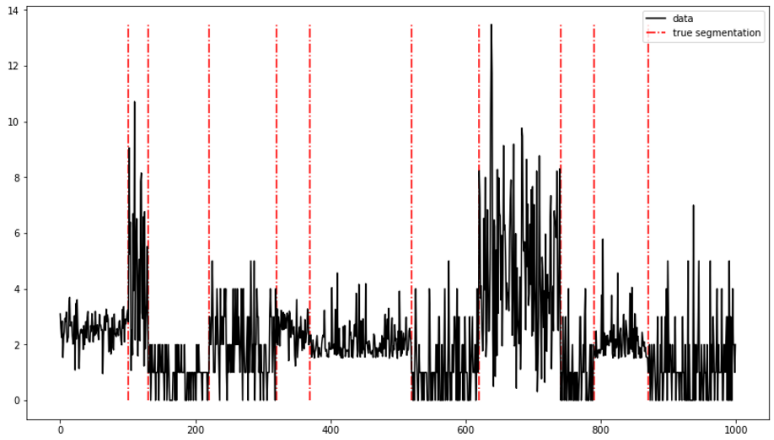


Figure 1.2: Breakpoints in time series.

stationary between two consecutive breakpoints τ_i and τ_{i+1}.

$$\mathcal{P}_{X_{\tau_i-1}} \neq \mathcal{P}_{X_{\tau_i}} = \mathcal{P}_{X_{\tau_i+1}} = \ldots = \mathcal{P}_{X_{\tau_{i+1}-1}} \neq \mathcal{P}_{X_{\tau_{i+1}}} \tag{1.4}$$

There are many ways to approach the problem, see for example these reviews [163, 4] for offline breakpoint detectors and [7] for online breakpoint detectors. It will be seen later that one of the objectives of this manuscript is to study the use of breakpoint detectors for online anomaly detection. However, online anomaly detection can be used with an offline breakpoint detector. In this case, the detected breakpoints are re-evaluated with each new observation. The disadvantage of this approach is the longer calculation time, but the advantage is the improved accuracy. The authors of [163] show that a breakpoint detector can be described as an optimization problem using three notions.

- **Cost function**. A cost function $\mathcal{C}(\cdot)$ measures the homogeneity of a given subseries $X_{t_1}^{t_2}$. With a well chosen cost function, $\mathcal{C}(X_{t_1}^{t_2})$ is high when there is at least one breakpoint between t_1 and t_2. The cost function is low when there is no breakpoint in this subseries.
- **Search method**: The search method enables to explore a set of possible segmentations, called $\mathcal{T}$, of the optimization problem. Each search method is a trade-off between accuracy and computational complexity [163].
- **Penalty function**: The penalty function is useful when the number of breakpoints is unknown. It avoids overestimating the number of breakpoints by penalizing segmentations with a large number of breakpoints. The penalty function $pen(\cdot)$ increases based on the number of segments, noted as D_τ.

The segmentation returned by the breakpoint detector is the one that minimizes the penalized cost function among the explored solutions:

$$\hat{\tau} \in \arg\min_{\tau \in \mathcal{T}} \sum_{i=1}^{D_\tau} \mathcal{C}(X_{\tau_i}^{\tau_{i+1}-1}) + pen(D_\tau) \tag{1.5}$$

Given the computational cost of solving the exact optimization problem, some anomaly detectors propose approximate solutions. Some approximate search methods are presented in Section 1.2.3.1-1.2.3.3, followed by the exact solution in Section 1.2.3.4. There are several difficulties with breakpoint detection: the number of breakpoints is usually unknown, changes can occur in any feature, not just the mean or variance, and parametric assumptions are difficult to verify. For each method, there is a discussion of how these different problems are addressed.

1.2.3.1 CUSUM detector

CUSUM is an online breakpoint detector [66, 168] that observes data points sequentially. It relies on calculating a cumulative sum of the deviations between an observed value and a predicted value. Once a change has occurred, the observed deviations accumulate until they exceed a threshold indicating that a breakpoint has been detected. To achieve this, two models f_{θ_0} and f_{θ_1} are used. Under the $\mathcal{H}_0$ hypothesis, there are no breakpoints, the data $[1, T]$ are all generated by the f_{θ_0} model. Under the $\mathcal{H}_1$ hypothesis, there is a breakpoint b such that the data $[1, b-1]$ are generated by the model f_{θ_0} and the data $[b, T]$ are generated by the model f_{θ_1}. The test statistic used is called the likelihood ratio [168], which is calculated as follows, assuming independence, where $\mathcal{C}$ is the segment cost function defined in Eq 4.21:

$$E^b_{t_1,t_2} = \mathcal{C}(X^{t_2}_{t_1}) - \mathcal{C}(X^{b-1}_{t_1}) - \mathcal{C}(X^{t_2}_{b}) \tag{1.6}$$

$$E^b_{0,t} = \log \frac{\prod_{t=1}^{b-1} f_{\theta_0}(X_t) \prod_{t=b}^{T} f_{\theta_1}(X_t)}{\prod_{t=1}^{T} f_{\theta_0}(X_t)} \tag{1.7}$$

The breakpoint location is estimated by maximizing the likelihood ratio, as follows:

$$E_{0,t} = \max_b E^b_{0,t} \tag{1.8}$$

$$= \max_b \sum_{t=b}^{T} \log f_{\theta_1}(X_t) - \log f_{\theta_0}(X_t) \tag{1.9}$$

$$= \max(E_{0,t-1} + (\log f_{\theta_1}(X_t) - \log f_{\theta_0}(X_t)), 0) \tag{1.10}$$

The last expression shows that it's possible to compute $E_{0,t}$ sequentially, which is interesting for an online computation. In its original version [126], f is a Gaussian distribution and the means θ_0 and θ_1 are fixed in advance. The Generalized Likelihood Ration (GRL) algorithm [148, 34] estimates θ_1 using maximum likelihood. This makes it possible to apply CUSUM even if the distribution after the breakpoint is unknown, but makes the sequential computation of $E_{0,t}$ more difficult. The CUSUM algorithm can be nonparametric using a kernel based statistic [55]. The thresholds are set to control the Average Run Length (ARL_0) [168], i.e. the average time for detecting a false breakpoint.

1.2.3.2 Window-based breakpoint detection.

Window-based breakpoint detection is another well-known search method aimed at quickly obtaining an approximate solution to the optimization problem. It can be used as online detector. The time series is scanned by a window of length $2w$. At each point t, the discrepancy between the first and second half of the window is computed. Points where the discrepancy is locally maximum indicate the position of breakpoints. This gives an approximate solution to Eq. 4.21, as only the local cost associated with a breakpoint is considered, without taking the global cost

into account. The discrepancy can be expressed in terms of the segment cost, as follows:

$$E^t_{t-w,t+w} = \mathcal{C}(X^{t+w}_{t-w}) - \mathcal{C}(X^{t-1}_{t-w}) - \mathcal{C}(X^{t+w}_t) \tag{1.11}$$

In practice, however, the discrepancy is usually measured by a two-sample test statistic [88]. This is a statistical test that evaluates the hypothesis that $X_{t-w}, \ldots, X_{t-1}$ comes from the same distribution as $X_t, \ldots, X_{t+w}$. Here is a selection of statistics commonly used in breakpoint detection.

t-test The t-test is a parametric test [107] that detects changes in the mean, assuming that the two samples are generated according to independent Gaussians.

$$E^t_{t-w,t+w} = \frac{1}{s_p(\sqrt{2/w})} \left| \frac{1}{w} \sum_{u=t-w}^{t-1} X_u - \frac{1}{w} \sum_{u=t}^{t+w} X_u \right| \tag{1.12}$$

With $s_p = \sqrt{\frac{(w-1)\sigma_1^2 + (w-1)\sigma_2^2}{n_1+n_2-2}}$.

Generalized Likelihood Ratio (GRL) GRL [148] computes the likelihood ratio between two models:

- under H_0, the two samples are generated from the same distribution. The likelihood of each point is calculated using $f_{\hat{t}heta}$, where $\hat{t}heta$ is the MLE estimator considering points from both samples.
- Under H_1, the samples are generated according different distributions. The true likelihood of the first sample is calculated by $f_{\hat{t}heta_1}$. Where $\hat{t}heta_1$ is calculated by the MLE estimator considering only the first sample. Similarly, the true likelihood of the second sample is calculated by $f_{\hat{\theta}_2}$.

When there is a breakpoint at position t, the likelihood under H_1 is significantly larger than the likelihood under H_0.

$$E^t_{t-w,t+w} = \frac{\prod_{u=t-w}^{t-1} f(X_i|\hat{\theta}_1) \prod_{u=t}^{t+w} f(X_i|\hat{\theta}_2)}{\prod_{u=t-w}^{t+w} f(X_i|\hat{\theta})} \tag{1.13}$$

Mean Max discrepancy GLR relies on a parametric model and the t-test can only detect shifts in the mean. Mean Max Discrepancy [68] uses the properties of Reproducing Kernel Hilbert Spaces (RKHS) to increase the detection power of the test while remaining non-parametric. Instead of computing the difference between the means of each sample, it computes the difference between the means of their projections, through a mapping function ϕ, in a high dimensional RKHS, called H_K.

$$E^t_{t-w,t+w} = \left\| \frac{1}{w} \sum_{u=t-w}^{t-1} \phi(X_u) - \frac{1}{w} \sum_{u=t}^{t+w} \phi(X_u) \right\|^2_{H_K} \tag{1.14}$$

This formulation is difficult to use in practice, therefore it is preferable to define ϕ and H_K implicitly using a positive semi-definite kernel, as allowed by Mercer's theorem. The mean max

discrepancy is expressed directly using a kernel K is [68] as follows:

$$E^t_{t-w,t+w} = \frac{1}{w^2}\sum_{u=t-w}^{t-1}\sum_{v=t-w}^{t-1} K(X_u, X_v) + \frac{1}{w^2}\sum_{u=t}^{t+w}\sum_{v=t}^{t+w} K(X_u, X_v) - \frac{2}{w^2}\sum_{u=t-w}^{t-1}\sum_{u=t+1}^{t+w} K(X_u, X_v) \tag{1.15}$$

To improve the time efficiency, [42] suggests an MMD estimator that is linear in time.

1.2.3.3 Binary segmentation

Binary segmentation is another search method for quickly obtaining an approximate solution in offline breakpoint detection. In binary segmentation, the first step is to find the point that optimally splits the complete $[0, T]$ series into two subseries.

$$\hat{b} = \arg\min_{t\in[1,T]} \mathcal{C}(X_1^{t-1}) + \mathcal{C}(X_t^T) \tag{1.16}$$

Where $\mathcal{C}$ is the segment cost function defined in Eq. 4.21.

This results in two sub-series. This method is then applied recursively to each subseries until the desired number of breakpoints is found. The search for each single breakpoint is generally done by maximizing a CUSUM statistic on the subseries under consideration, rather than explicitly minimizing the cost function. For example, the article [58] uses the formulation Eq 1.17. This formulation is similar to that of Section 1.2.3.1, as shown in [9], where the start of the subseries is $s = t - w$ and the end is $e = t + w$. Let s, e be two integers, the CUSUM statistic associated with the subseries X_s^e and with the breakpoint in $b \in [\![s, e]\!]$ is calculated as follows:

$$E^b_{s,e} = \sqrt{\frac{e-b}{(e-s+1)(b-s)}}\sum_{u=e}^{b-1} X_u - \sqrt{\frac{b-s}{(e-s+1)(e-b-1)}}\sum_{u=b}^{e} X_u \tag{1.17}$$

The CUSUM statistic presented in Eq. 1.17 can only detect changes in the mean. Other CUSUM statistics exist to detect other types of changes, for example [8] detects changes occurring in the covariance. More generally, any statistic that exists for a two-sample test can be used; for example, [124] studies the use of the Kolmogorov-Smirnov test, and [107] suggests an entropy-based test. The number of breakpoints is chosen either by a threshold η on $\max_b E^b_{s,e}$ [125] or by minimizing the penalized criterion defined in Eq. 4.21 after applying $D_{max} - 1$ times the binary segmentation to find D_{max} segments.

1.2.3.4 Optimal Solution

The previous search methods provide approximate solutions to the optimization problem formulated by Eq. 4.21 in order to limit the computational complexity. Nevertheless, once the cost associated with a segmentation can be written as the sum of the costs associated with each segment, it is possible to find the optimal D-segment segmentation in a relatively efficient way using dynamic programming. The cost of the optimal segmentation in D segments can be written as the cost of a segment adding the cost of an optimal segmentation in $D-1$ segments. Let $L_{T,D}$ be the cost of the optimal segmentation of X_1^T into the D segments. By noting b, the last

breakpoint, and $\mathcal{T}^D$ the set of all possible segmentations in D segments, it gives:

$$L_{T,D} = \min_{\tau \in \mathcal{T}^D} \sum_{i=1}^{D_\tau} \mathcal{C}(X_{\tau_i}^{\tau_{i+1}-1}) \tag{1.18}$$

$$L_{T,D} = \min_b L_{b,D-1} + \mathcal{C}(X_b^T) \tag{1.19}$$

The associated optimal segmentation is noted $\hat{\tau}_D$. The locations of the various breakpoints in $\hat{\tau}_D$ can be obtained by applying Eq. 1.19 recursively. It provides an exact solution to the optimization problem in time $O(DT^2)$.

The number of breakpoints is obtained by minimizing the penalized criterion. If the penalty is linear, the number of segment and their locations can be estimated directly using dynamic programming in a single pass.

$$L_{T,D} = \min_b L_{b,D-1} + \mathcal{C}(X_b^T) + \beta \tag{1.20}$$

[91] suggests a pruning strategy to solve this optimization problem with a linear time complexity.

Otherwise, for each possible number of segments D, the positions of the breakpoints is found with their associated cost, and then the penalty is added to estimate the number of segments.

$$\hat{D} = \arg\min_D L_{T,D} + pen(D) \tag{1.21}$$

An example of a breakpoint detector that uses dynamic programming is Kernel Change Point (KCP).

KCP KCP [6, 38] uses a kernel to define the segment cost function. If the kernel used is "characteristic", any kind of change can be detected [6]. The Gaussian kernel is characteristic. For a given segmentation τ and a kernel K, the cost is given by:

$$\hat{R}(\tau) = \frac{1}{t}\sum_{u=1}^{t} K(X_u, X_u) - \frac{1}{t}\sum_{i=1}^{D_\tau} \frac{1}{\tau_{i+1} - \tau_i} \sum_{u,v=\tau_i}^{\tau_{i+1}-1} K(X_u, X_v) \tag{1.22}$$

The penalty function is given by:

$$pen(\tau) = r_1 D_\tau + r_2 \log \binom{t-1}{D_\tau - 1} \tag{1.23}$$

where the coefficients r_1 and r_2 are estimated by fitting the penalty function on the estimated cost for over segmented segmentations [12].

1.3 Anomaly detection on time series

1.3.1 Definitions

An anomaly is an observation or group of observations that appears suspicious when compared to other observations. In general, anomalies are evidence that an unexpected event has occurred in the process that produces the observed data. For example, it could be a failure in an industrial machine monitored by sensors, a disease detected from a patient's test results, or a computer

attack detected from log files... Mathematical modeling of anomalies assumes the existence of a reference process law, called $\mathcal{P}_{0,t}$. An observation is said to be an anomaly if X_t does not follow the reference distribution. The distribution $\mathcal{P}_{0,t}$ can be constant or evolve over time. The goal of anomaly detection is to find all data points that are abnormal.

In the literature, anomalies are classified into to three types. A point anomaly is a data point that takes a value that the time series should never take. For example, a body temperature of $42°C$. A contextual anomaly is a data point that takes a value that the time series should never take according to a certain context. For example, a temperature of $30°C$ in Paris during winter. The value taken by a contextual anomaly may appear normal in other contexts, e.g. $30°C$ is normal in Paris during summer. A collective anomaly refers to a collection of points that appears suspicious. Note that each of these points may appear normal on its own. For example, a person making a payment transaction over the Internet may not appear suspicious. However, if the same transaction occurs several times in a short period of time, credit card fraud may be suspected. An anomaly detector may be good at one type of anomaly, but not another. This work focuses on punctual and contextual anomalies.

Anomaly detectors are built using machine learning. The next section discusses the different learning contexts: supervised and unsupervised.

1.3.1.1 Supervised vs Unsupervised

A supervised detection context requires the use of a labeled training set $(X_t, y_t)_{1\leq t\leq q}$, where y_t is a boolean variable indicating whether X_t is an anomaly or not. It requires a domain expert to indicate the location of anomalies in a data history. The anomaly detection problem becomes a binary classification task where a model f is learned by minimizing the classification error $\ell(f(X), y)$. Classification is a well-studied problem in machine learning, and many algorithms exist [18].

However, in most cases, labeling data is a difficult and costly process. Therefore, labeled data is rare and the majority of anomaly detectors rely on unsupervised or semi-supervised methods. Unsupervised models learn the reference behavior of the time series from historical data $(X_t)_{1\leq t\leq q}$. Anomalies are detected by comparing observations with the learned reference behavior. Models requiring to have only normal data during training are called semi-supervised. While unsupervised detectors [143] tolerate the presence of a small number of anomalies in their training set. Most unsupervised detectors learn an atypicity function $a : \mathcal{X} \to \mathbb{R}$ which assigns each data point X_t an atypicity score s_t. The value of s_t is small when X_t looks similar to the training data, and large when it is atypical. If the atypicity function is well chosen, anomalies appear with a high s_t score. However, it is difficult to directly define a threshold on the score, since the distribution of the reference scores is unknown. Therefore, an estimation is made of the p-value p_t associated with the s_t score, and then a detection threshold is defined on the p-value.

1.3.1.2 Online vs offline

A distinction is made between online and offline anomaly detection. Offline anomaly detector uses knowledge of the entire series to determine the status of normal/abnormal points [29, 30, 106]. This framework is suitable for finding anomalies on a past recording, but not for real-time anomaly detection. In online anomaly detection, data points are observed sequentially and their status is evaluated in real time [29, 100]. This is a more challenging context to work in because the future data that could help make a decision is unknown, and the constraints on computation time are greater. Online anomaly detectors are usually trained on historical data [39]. However,

the normal behavior of the series may change over time. Some anomaly detectors [71, 122] therefore update their model in real time or at regular intervals.

1.3.2 Anomaly detector error

When analyzing a time series, an anomaly detector can make two types of errors.

- Type I error involves finding an anomaly where there is none.
- Type II error is the failure to detect a true anomaly.

The evaluation results of an anomaly detector are presented in the form of a confusion matrix, as shown in Table 1.1. The rows of the matrix correspond to the true labels and the columns correspond to the estimates of the anomaly detector. TP is the number of true positives, the true anomalies successfully detected. FP is the number of false positives, the normal data points detected as anomalies. FN is the number of false negatives. TN is the number of true negatives.

True label\ Prediction	Anomaly	Normal
Anomaly	TP	FN
Normal	FP	FN

Table 1.1: Confusion matrix for Anomaly Detection.

Various performance metrics can be calculated from the confusion matrix. Anomalies are very rare in a data set. This means that the data is highly unbalanced, and balanced binary classification performance criteria such as accuracy or false positive rate are of little interest. Indeed, a detector that detects no anomalies would appear to perform well according to these criteria.

Better performance metrics are precision and recall.

$$\text{Recall} = \frac{TP}{TP + FN}; \text{Precision} = \frac{TP}{TP + FP} \tag{1.24}$$

Precision is the proportion of true anomalies among all points detected as anomalies. Recall is the proportion of true anomalies found by the detector. The F1 criterion is the harmonic mean of recall and precision. It varies between 0 and 1. The best possible value is 1. The disadvantage of the F1 criterion is that it gives equal weight to both types of errors. It is also difficult to interpret statistically.

$$F1 = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1.25}$$

All these criteria depend on the chosen threshold, so if a performance criterion is required that depends only on the atypicity scores, the PR curve describes the precision and recall for different thresholds. The PR-AUC score calculates the area under the PR curve to enable comparison between different detectors.

[1] Illustration coming from https://www.tuteurs.ens.fr/logiciels/latex/footnote.html

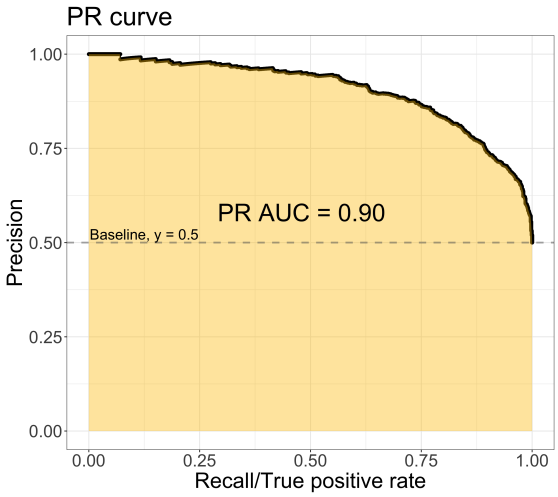


Figure 1.3: PR curve and PR-AUC.[1]

The PR-AUC score has its limitations when it comes to evaluating an online anomaly detector. Anomalies require a specific threshold to be detected. However, PR-AUC gives no indication of performance at a given threshold.

1.3.3 Short review on anomaly detection in time series

This section provides a brief review of the different methods used to build an anomaly detector. For a more detailed review, please refer to [39, 20, 143, 29].

1.3.3.1 Anomaly detection inspired from multi-dimensional data

Many anomaly detectors are not specific to time series, they detect anomalies from a list of vectors. Time series data can be transformed into a set of vectors using the windowing method: for each t, a vector of dimension w is built by collecting the w consecutive points. Noting $\tilde{T} = T - w$:

$$\forall t \in [\![1, \tilde{T}]\!], \qquad \mathbf{X}_t = (X_t, X_{t+1}, \dots, X_{t+w})^T$$

The data set $\{\mathbf{X}_t\}$ can be treated as a multidimensional data set without temporal structure. In fact, the structure is preserved within each vector. In this way, existing anomaly detection methods for multidimensional data can be used. It should be noted that anomaly detection in high-dimensional data comes with its own challenges, as described in [179].

Nearest Neighbor: Assuming that anomalies are observations that are far from the common observations, it is natural to use the notion of distance to the k-nearest neighbor as the atypicity score. Let $\mathbf{X}_1^{\tilde{T}}$ be a training set and d be a distance on $\mathcal{X}^w$, then the atypicity score function is defined as [131, 39]:

$$a_{kNN}(\mathbf{X}) = \sum_{\mathbf{O} \in kNN(\mathbf{X}, \mathbf{X}_1^{\tilde{T}})} d(\mathbf{X}, \mathbf{O})$$

The distance of each k nearest neighbor to the data point to be analyzed is summed.

Isolation-Forest: Isolation Forest [106] is an anomaly detector based on the idea that anomalies are easy to isolate from the rest of the data. And that all it takes to isolate an anomalous point is to randomly generate a few boundaries. On the other hand, to isolate a normal data point, you need to generate more random boundaries.

The algorithm works as follows: Binary trees are randomly generated. At each step, the feature to cut and the position of the cut are randomly chosen. Tree construction is completed when each data point is placed on a separate leaf. The set of trees constructed in this way forms a forest. Given a data point, the lower the average number of steps required to isolate a feature, the higher its atypicity score. Let $E(h(\mathbf{X}))$ be the average depth of $\mathbf{X}$ in the set of trees and $\kappa(n)$ be a constant depending only on the number of points in the data set n. The atypicity score with Isolation Forest is computed as follows:

$$a_{IF}(\mathbf{X}, n) = 2^{-\frac{E(h(\mathbf{X}))}{\kappa(n)}}$$

Local-Outlier Factor: Density-based anomaly detectors [39] assume that anomalous points are located in low-density regions. These approaches run into difficulties when the density varies in the data space, for example, when high-density and low-density clusters are present in the same data set. To overcome this difficulty, anomaly detectors [30] introduce the notion of relative density. A point is said to be anomalous if it is located in a region of low density relative to its nearest neighbors . Local Outlier Factor is an atypicity score that based on local density. The Local Reach Density (LRD) is defined as the inverse of the k-nearest neighbor average of the maximum reachable distance, defined as $d_k(\mathbf{X}, \mathbf{O}) = \max(d(\mathbf{X}, \mathbf{O}), d(\mathbf{X}, kNN(\mathbf{X}, \mathbf{X}_1^{\tilde{T}}))$:

$$LRD(\mathbf{X}) = \left(\frac{\sum_{\mathbf{O} \in kNN(\mathbf{X}, \mathbf{X}_1^{\tilde{T}})} d_k(\mathbf{X}, \mathbf{O})}{k} \right)^{-1}$$

The atypicity score is defined as the average ratio between the local density of the data point and the local density of its neighbors.

$$a_{LOF}(\mathbf{X}) = \frac{1}{k} \sum_{\mathbf{O} \in kNN(\mathbf{X}, \mathbf{X}_1^T)} \frac{LRD(\mathbf{O})}{LRD(\mathbf{X})}$$

Cluster based anomaly detection In some cases, normal data are grouped into several clusters corresponding to different modes of the reference distribution. Anomalies can be identified as points that do not belong to any cluster or whose distance to the nearest cluster is large. For example, DBSCAN [51, 36] makes a cluster of data points, and points that do belong to any cluster are considered anomalies. The article [121] uses k-means clustering [157] to identify the different clusters in the training data. The atypicity score is then defined as the Euclidean distance from a point to the centroid of the nearest cluster.

Online detection These different methods were originally designed as offline anomaly detectors. The analysis is performed as follows: first, the score of each point in the data set is calculated. Then, the points with the highest atypicity scores are selected. There are several ways to adapt these methods to online anomaly detection: for example, the training set can be fixed, the atypicity score function can be learned once, and the score of each new data point is

calculated as it is observed. However, if the goal is to learn normal behavior continuously, the newly observed points must be added to the training set and the learning of the atypicity score must be updated. In this way, the training set will contain all the points visited up to time t. When learning the reference behavior in real time, obsolete training points must be forgotten. This is often accomplished by using a sliding window to select the training set at time t. This can be done simply by iteratively applying the offline algorithm to the subseries available at time t. More advanced methods avoid this cost by updating only what is necessary. For example, Random Cut Forest [71] is a method inspired by Isolation Forest and adapted to real time. DiLOF [122] adapts LOF for real time.

Threshold selection The displayed scores characterize the atypicity of an observation according to different criteria. In online anomaly detection, it is not enough to define the atypicity of a point: you also need to make a binary decision on the status of the point: normal or abnormal. For this reason, a threshold on the score is used beyond which the observation is considered abnormal. To set the threshold correctly, it is necessary to know the distribution of the reference behavior scores. For example, assuming that the scores follow a Gaussian distribution, the thresholds can be set at 3 standard deviations from the mean. The LOF score distribution follows a gamma distribution under certain conditions [49], the threshold is set as a quantile of this distribution. There are also agnostic approaches that make no assumptions about the distribution of the scores, such as Conformal Anomaly Detection [100], which defines a threshold on the empirical p-values of the scores. CAD has been used in particular with kNN [33] or LOF[101].

1.3.3.2 Forecasting based method

A common approach to anomaly detection in time series is to train predictive models to learn the normal behavior of the series. Once trained, these models can then be used to compare what is observed in real time with what is forecast by the model representing the reference behavior. An anomaly is an observation that deviates too far from the predictions. There is an extensive

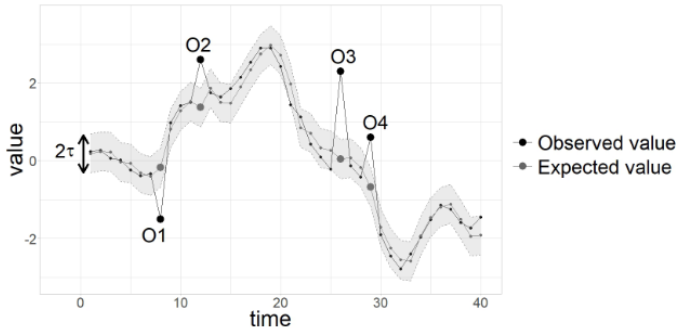


Figure 1.4: Anomaly detection based on the comparison between the predicted and observed values, illustration from [20].

literature on predictive models, some of which is presented here. A distinction is made between statistical models and models based on deep learning. For more details, please refer to [83].

ARMA An important family of statistical models for forecasting is ARMA [27, 83], which combines an Autoregressive (AR) model and a Moving Average (MA) model. An AR model predicts the value of a point from a linear combination of past values, as follows, with e_t as Gaussian white noise and $\phi_1, \ldots, \phi_p$ real coefficients:

$$X_t = c + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \ldots + \phi_p X_{t-p} + e_t \tag{1.26}$$

Moving Average (MA) models describe time series values as a weighted average of past noise values, as follows, where e_t is a Gaussian white noise and $\theta_1, \ldots, \theta_p$ are real coefficients:

$$X_t = c + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \ldots + \theta_q e_{t-p} \tag{1.27}$$

The ARMA model describes a time series as the sum of an AR model and a MA model.

$$X_t = \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{i=1}^{q} \theta_i e_{t-i} + e_t \tag{1.28}$$

In some cases, the difference series $X_t - X_{t-1}$ is easier to predict than the original series. In fact, this trick makes it possible to suppress some trend effects. The ARIMA model applies an ARMA model to the finite difference series. The weights associated with these different models are estimated by the MLE estimator. ARMA based model are used in anomaly detection by comparing the prediction and the observed value [94, 127].

EWMA Another well-known statistical forecasting model is the Exponential Weighted Moving Average (EWMA) [83]. In its simplest version, X_t is assumed to follow $\mathcal{N}(\mu_t, \sigma)$. EWMA estimates the local mean of the time series μ_t by taking the weighted sum of all observations with a decay of r for each time step in the past. $\hat{\mu}_t = \sum_{u=1}^{t} r^u (1-r) X_u$. This is easy to compute online:

$$\hat{\mu}_t = (1-r) X_t + r \hat{\mu}_{t-1} \tag{1.29}$$

r is a parameter that defines the importance of past values relative to present values. When r is close to 0, the estimate of the mean is very close to the last observation. When r is close to 1, the last observation has little influence on the estimate of the local mean. Because of its simplicity, it's one of the oldest ways to detect anomalies [173, 35, 29]. Since the model is Gaussian, observations that verify $|X_t - \hat{\mu}_t| > k\hat{\sigma}$ are considered anomalies. The parameter r is estimated by least squares on a historical data set.

The model can be made more complex by adding trend or seasonal terms, or by estimating the variance [84, 83]. Although simple, this model is still powerful. For example, it is an essential part of the winning submission to the M4 forecasting competition [109, 152].

Deep-Learning models in forecasting It is possible to view forecasting as a regression problem, where the goal is to estimate X_t using $X_{t-w}, \ldots, X_{t-1}$. In this way, many machine learning models can be trained on forecasting. In particular, deep learning methods are increasingly used as predictive models for anomaly detection. They are especially interesting for multivariate series, where building a statistical model is more difficult. In particular, recurrent neural networks have the advantage of using the entire series to predict the next observation. In contrast, other architectures, such as multi-layer perceptrons and convolutional networks, can only use a subset of the series as input. This explains their popularity, especially for LSTMs, which are better

suited for predicting long sequences [32, 82]. However, LSTMs require a lot of data to train, and other architectures are also used. [120] suggests using CNN as a prediction model.

A recent trend, encouraged by results obtained in the fields of NLP [31] and CV [92], is the research of a "foundation model" [25] for time series, which, once trained on a large and varied set of time series, should enable the prediction for other time series without the need to be trained again [177, 116].

Offline and continuous learning Predictive models trained offline become obsolete over time. It is therefore necessary to retrain them periodically. This is typically done by scheduling the training of the model on more recent data at regular intervals. Training data is selected using a sliding window containing the most recent data. More rarely, the model data can be updated in real time as new data is observed [174]. Again, choosing the length of the window used as the training set is a difficult problem. Although enough data is needed to learn normality, the training process must not be biased by outdated data.

Threshold selection In many forecasting models, the prediction error is modeled, often by a Gaussian. Thus, the quantiles of the prediction error define the thresholds of the anomaly detector. It is particularly interesting to have a model for the prediction error when the score distribution evolves over time. The GARCH model [50, 24] is interesting for heteroscedastic series [118]. If the error is not modeled, the threshold can be set using the empirical p-value, as in Section 1.3.3.1. Although less common, there are non-statistical ways to set the threshold. For example, [82] performs unsupervised threshold learning based on the assumptions that anomalies have a strong impact on the mean estimation and are rare.

1.3.3.3 Anomaly Detection as Breakpoint Detection

As seen in section 1.2.3, it is possible to detect changes in properties over a time series using breakpoint detectors. These changes are associated with changes in the behavior of the underlying process. In particular, if the properties of the series are expected to be constant over time, then a breakpoint indicates an anomaly. Breakpoint detectors are therefore also used as anomaly detectors. For example, the article [3] uses CUSUM to detect DoS attacks by applying CUSUM to the number of connections received per time interval. The detection threshold is set to control the Average Run Length, i.e. the average time to alarm assuming no change. EWMA can also be used to detect changes in the mean μ_t of the time series X_t [136] beyond fixed thresholds. These methods, known as EWMA and CUSUM control charts, were among the first to detect anomalies in the 1950s.

Another approach is to distinguish between normal and abnormal breakpoints. For example, the paper [167] first detects breakpoints using a binary segmentation algorithm 1.2.3.3. The law of the mean shift between two segments is learned. Segments with an unusually high mean shift are abnormal segments.

1.3.3.4 Anomaly Detection using Deep Learning

This section presents approaches to anomaly detection using deep learning, without mentioning the forecast based methods introduced in Section 1.3.3.2. The survey [140] identifies different uses of deep learning in anomaly detection and makes connections with classical methods.

Auto-encoder An auto-encoder is a multilayer neural network consisting of two components: the encoder and the decoder. As shown in 1.5, the number of neurons per layer in the encoder gradually decreases, as the encoder aims to build a low-dimensional representation of the input data. The number of layers in the decoder gradually increases so that the dimension of the output data is the same as the input data. Auto-encoders are trained end-to-end, minimizing reconstruction error. Once trained on normal data, an auto-encoder can be used for anomaly detection. The most common way is to use the reconstruction error as an atypicity score [165, 41, 115]. More rarely, the encoder is used to reduce the dimension of the input data, then other anomaly detection methods are used on these embedding vectors [63].

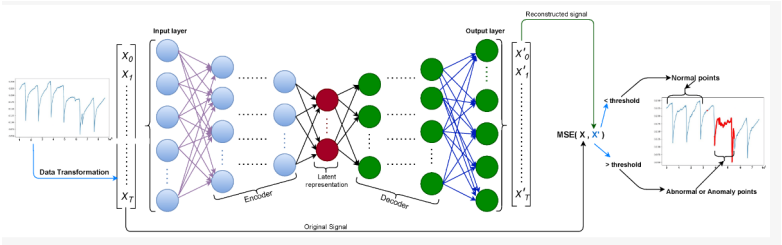


Figure 1.5: Illustration of Anomaly detection using auto encoder, illustration from [123].

Many different architectures are used to build encoders/decoders. For example [142] uses LSTM based auto-encoders to detect security breaches in IT infrastructures.

Generative Adversarial Network GANs[65, 43] simultaneously train two neural networks with adversarial goals: the generator and the discriminator. The goal of the generator is to fool the discriminator into believing that the data it generates is data from the training set. The goal of the discriminator is to detect which data is from the generator and which is from the training set.

In the context of anomaly detection, the generator can be a forecasting model or an auto-encoder [62]. The purpose of the discriminator is twofold. First, it improves the training of the forecaster/autoencoder by adding an adversarial constraint. Second, since the discriminator has been trained to detect data that does not come from the training distribution. Anomalies can be detected by the discriminator [62]. This is illustrated in Figure 1.6.

In Figure 1.6, the observed data x first passes through the encoder $\mathcal{E}$ and then through the generator/decoder $\mathcal{G}$. The $\mathcal{E}$ and $\mathcal{G}$ networks are in part trained to minimize the L_2 reconstruction error, which serves as an atypicity score on the new data. In addition, there are two discriminator networks. $\mathcal{C}_x$ is adversarially trained with $\mathcal{G}$ to give a value close to 0 for data from the training set and high for data from $\mathcal{G}$. $\mathcal{C}_z$ plays the same role, but in the data space encoded by $\mathcal{E}$. The complete atypicity score is written as (without writing the renormalization constants):

$$a_{TADGAN}(X_t) = \|X_t - \mathcal{G}(\mathcal{E}(X_t))\| + \mathcal{C}_x(X_t) + \mathcal{C}_z(\mathcal{E}(X_t)) \tag{1.30}$$

This concludes the small review on anomaly detectors. The next section looks at controlling false positives using multiple testing methods.

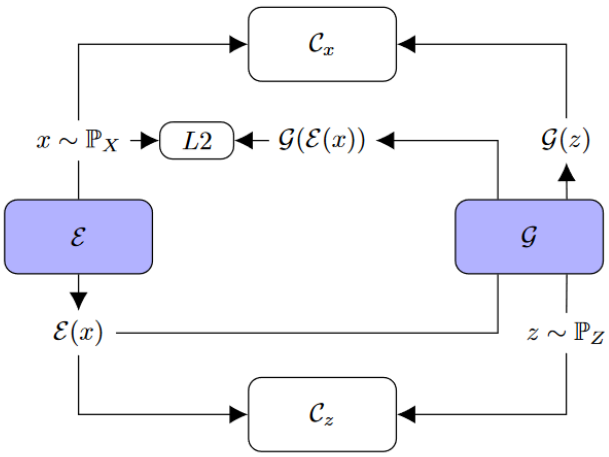


Figure 1.6: Description of the TADGAN architecture, illustration from [62].

1.4 Multiple testing

To formalize the problem of controlling the number of false positives, anomaly detection is described as a multiple testing problem. Let the hypothesis $\mathcal{H}_{0,t}$ be verified if X_t is a point from the reference process, i.e. it is not an anomaly.

$$\mathcal{H}_{0,t} \equiv X_t \sim \mathcal{P}_{0,t}$$
$$\mathcal{H}_{1,t} \equiv X_t \nsim \mathcal{P}_{0,t}$$

Type I error, measured by the False Discovery Proportion (FDP), is calculated as the number of false positives out of the number of detections. Type II error is measured by the False Negative Proportion (FNP), and is calculated as the ratio of the number of false negatives to the number of true anomalies. The False Discovery Rate is equal to the expected FDP. The False Negative Rate is equal to the expected FNP.

$$FDR_1^T = \mathbb{E}[FDP_1^T] = \mathbb{E}\left[\frac{FP}{FP+TP}\right] \tag{1.31}$$

$$FNR_1^T = \mathbb{E}[FNP_1^T] = \mathbb{E}\left[\frac{FN}{FN+TP}\right] \tag{1.32}$$

The FDP and FNP are directly related to precision and recall.

$$FDR = 1 - \mathbb{E}[\text{precision}]; FNR = 1 - \mathbb{E}[\text{recall}] \tag{1.33}$$

The aim is to control the FDR below a target value noted as α, while minimizing the value of the FNR.

1.4.1 Benjamini-Hochberg Procedure

A well-known multiple testing procedure is the Benjamini-Hochberg (BH) procedure [13]. Let m be the number of hypotheses tested, m_0 the number of true null hypotheses, and $p_1, \ldots, p_m$ the corresponding p values. The proportion of true null hypotheses is noted as $\pi_0 = \frac{m_0}{m}$. BH ensures FDR control at the $\pi_0\alpha$ level. The BH procedure starts by ordering the p-values from smallest to largest: $p_{(1)} \leq p_{(2)} \leq \ldots p_{(m)}$. The smallest p-values are the most likely to be rejected. Then the number of rejections $\hat{k}$ is selected.

$$\hat{k}_{BH} = \max\{k : p_{(k)} \leq (k/m)\alpha\} \tag{1.34}$$

As a result, the hypotheses related to the p-values $p_{(1)}, p_{(2)}, \ldots, p_{(\hat{k})}$ are rejected. In the original version of the theorem, BH requires independence of p-values. The theorem has since been extended to broader conditions, as discussed in [19].

1.4.2 Storey Procedure

To improve its detection power, it would be preferable to apply Benjamini-Hochberg with α/π_0 instead of α, so that the FDR is exactly at the desired level α. Storey's procedure [158] involves estimating π_0. The estimator used is:

$$\hat{\pi}_0 = \frac{|\{p_i > \lambda\}|}{(1-\lambda)m} \tag{1.35}$$

The result is then injected into the BH procedure:

$$\hat{k}_S = \max\{k : p_{(k)} \leq (k/m)\alpha/\hat{\pi}_0\} \tag{1.36}$$

Therefore, reject $p_{(1)}, p_{(2)}, \ldots, p_{(\hat{k}_S)}$. However, the authors of the procedure prefer to rewrite this inequality in order to show the FDR estimator:

$$\hat{k}_S = \max\{k : \underbrace{\frac{\hat{\pi}_0 p_{(k)}}{k/m}}_{\widehat{FDR}} \leq \alpha\} \tag{1.37}$$

It appears that the Storey (or BH) threshold is the largest possible threshold that ensures that some FDR estimator is smaller than α. This procedure guarantees FDR control with better detection power than BH, but the estimation of $\hat{\pi}_0$ can make the procedure more unstable. Moreover, in the context of anomaly detection, anomalies are generally very rare, $\pi_0 \approx 1$. The gain is less significant.

1.4.3 Online multiple testing

In the context of online multiple testing, a sequence of p-values (p_t) is observed one after another. The decision to reject the null hypothesis must be made before a new p-value is received. It is not possible to apply the BH or Storey procedures because they require sorting all the p-values. A sequence of thresholds (ε_t) is used and compared to the sequence of p-values (p_t). Hypotheses with a p-value below the threshold are rejected. The sequence of thresholds must be chosen carefully to control the FDR.

If all hypotheses with a p-value smaller than α are rejected, then although the type I error of

each individual test is controlled, the FDR can be arbitrarily large. If the thresholds are defined such that $\sum_{t=1} \varepsilon_t = \alpha$, then the probability of falsely rejecting at least one null hypothesis is equal to α. This is an online version of the Bonferroni correction. But then ε_t quickly converges to 0 and the power of the test becomes very poor. That's why controlling the FDR is preferred. To control the FDR, the approach used is to estimate the FDP by:

$$\widehat{FDP}_1^t = \frac{\sum_{u=1}^t \varepsilon_u}{|\mathcal{R}(t)|} \tag{1.38}$$

with $\mathcal{R}(t)$, the positions of all rejected hypotheses from 0 to t. To control the FDR, the idea is to ensure that $\widehat{FDP}_1^t \leq \alpha$ is verified at each instant. The papers [132, 87] define these ideas more rigorously, using the alpha-investing formalism, and propose procedures to explicitly compute the sequence (ε_t). As an example, the LORD procedure [87] uses two numbers w_0 and b_0 verifying $w_0 + b_0 = \alpha$ and a sequence $\gamma_t \propto \frac{\log(t \wedge 2)}{t \exp(\sqrt{t})}$ to compute a threshold sequence that ensures FDR control, as follows:

$$\varepsilon_t = \gamma_t w_0 + \sum_{u \in \mathcal{R}(t)} \gamma_{t-u} b_0 \tag{1.39}$$

It can be seen that if there is no rejection, $|\mathcal{R}(t)|$ remains constant and therefore ε_t needs to decrease to avoid breaking the condition $\frac{\sum_{u=1}^t \varepsilon_u}{|\mathcal{R}(t)|} \leq \alpha$. This makes it difficult to apply this method to anomaly detection. Indeed, the true p-values are unknown in this context and must be estimated. To verify the super-uniformity assumption, $\mathbb{P}[\hat{p}_t \leq u] \leq u$, required for FDR control, it is necessary to use the *conformal p-value*. It is calculated as follows, with a calibration set noted X_1^n and an atypicity score a:

$$\hat{p}_t = \frac{1}{n+1} \left(1 + \sum_{i=1}^{n} \mathbb{1}[a(X) > a(X_t)] \right) \tag{1.40}$$

The problem is that $\hat{p}_t \leq \frac{1}{n+1}$ and ε_t quickly become smaller than $\frac{1}{n+1}$. This means that anomalies cannot be detected unless a very large calibration set is used. As such, there are no multiple testing procedures suitable for online anomaly detection.

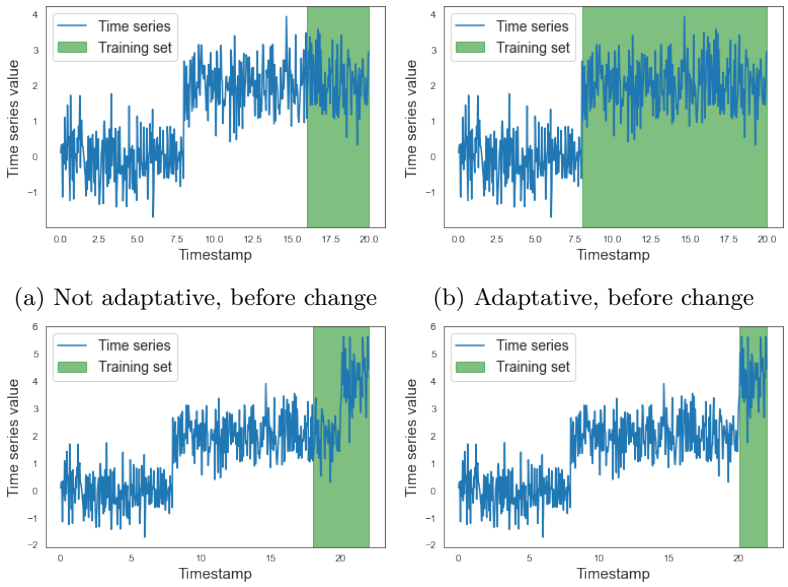
1.5 Main challenges

Following this review of anomaly detection and multiple testing, a number of issues have been identified.

1.5.0.1 Continuous learning of the reference behavior

Most anomaly detectors do not adapt to the normal behavior of the time series in real time. Rather, the normality is learned from an offline training set. The trained model is then used online to perform the anomaly detection. Furthermore, approaches that propose continuous learning typically use a fixed-size sliding window, which ignores the true dynamics of the time series and assumes that the series is locally stationary. However, as shown in Figure 1.7, the behavior of a time series may remain the same for a long period of time before abruptly changing to a different reference behavior. Using a sliding window of fixed size is suboptimal. The window

should be long during the stationary period to maximize the accuracy of the learned model, as shown in Figure 1.7a. But it should be shorter during the transition to quickly adapt to the new behavior, as shown in Figure 1.7b.



(a) Not adaptative, before change

(b) Adaptative, before change

(c) Not adaptative, after change

(d) Adaptative, after change

Figure 1.7: Comparison of the evolution of the training set at the breakpoint between a non-adaptive fixed cardinality training set and a training set adapted to the time series dynamics.

1.5.0.2 Ensuring anomaly detector performance

Anomaly detectors are evaluated on their ability to detect anomalies with few false positives. Anomaly detector benchmarks use recall/precision criteria. However, thresholds are typically estimated from quantiles on a distribution that is often assumed to be Gaussian. To guarantee the performance of a model, it is necessary to be able to estimate the p-value of the scores without knowing their true distribution and without being affected by the presence of anomalies.

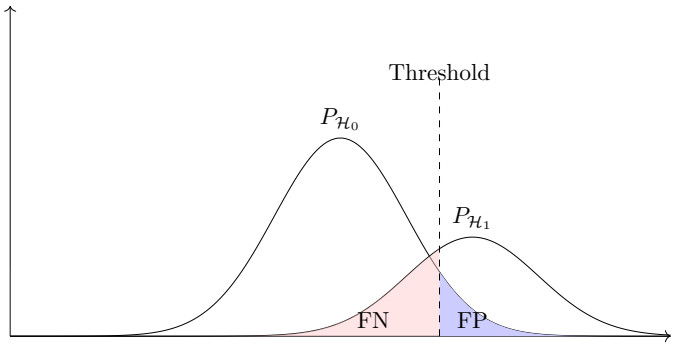


Figure 1.8: False positive and false negative depending on the threshold.

Furthermore, the use of quantiles is not directly interpretable in terms of precision. It gives an idea of the frequency of alarms, but does not allow to build a detector that is precise and alarms only when necessary. Anomaly detectors lack statistical guarantees of model performance. It would be interesting to be able to control the number of false positives. Multiple testing procedures are a source of inspiration for building anomaly detectors that guarantee performance in terms of anomaly detection. For the offline context, there are solutions that control the proportion of false discovery (FDR) [11, 110], but they are not applicable to the online context. There is no online multiple testing method that remains reliable even when the p-values have to be estimated, as in anomaly detection.

1.6 Contributions

The goal of this manuscript is to develop an anomaly detector with the ability to continuously learn the normal behavior of a time series and to provide statistical guarantees of its performance.

In order to describe the evolution of the reference distribution, breakpoints are introduced into the modeling of the time series. Let D be the number of segments and $(\tau_i) \in [\![1, T]\!]^{D+1}$ the breakpoint locations. Between two consecutive breakpoints τ_i and τ_{i+1} the series, excluding anomalies, is assumed iid and the reference distribution is noted $\mathcal{P}_{0,i}$. Breakpoints correspond to instants when the behavior of the series rapidly shifts toward a different reference distribution. Anomalies are data points that do not follow the segment reference distribution. An example of the model is shown in Figure 4.2. There are three segments, two anomalies in the second segment between breakpoints τ_2 and τ_3.

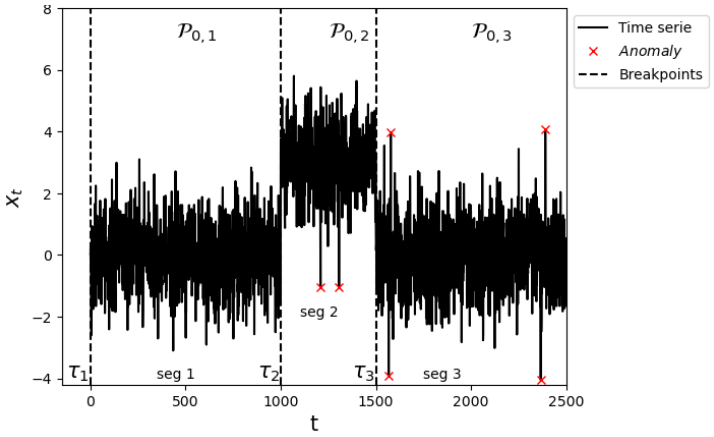


Figure 1.9: Illustration of piecewise stationary time series.

Detecting anomalies in this type of model involves several steps: First, the breakpoints in the series must be accurately identified so that the series is divided into homogeneous segments. Then, the normal behavior of the segments is learned and an atypicity score is assigned to the segment points. The next step is to estimate the p-value as accurately as possible. Finally, the threshold is chosen using a data-driven procedure, which should guarantee the statistical performance of the detector.

These different steps are developed in more detail in this manuscript as described in the following:

- Chapter 2: Estimating the p-value. The goal is to find an estimator $\hat{p}_t$ that is both robust and efficient. The approach is based on the use of the Kernel Density Estimator and the Median of Means. The difficulty lies in choosing the right bandwidth, the one which is adapted for estimating the p-value rather than the density. A new selection criterion is introduced based on minimizing the error in estimating the p-value at the tail of the distribution. An estimator based on cross-validation is built. The new selection procedure is empirically tested against existing procedures.
- Chapter 3: Proposes a data-driven threshold selection procedure that guarantees FDR control in online anomaly detection. To achieve this, two difficulties have to be solved: p-values are estimated and decisions are made online, without knowledge of the complete series. The first part focuses on understanding the relationship between the number of points in the calibration set used to estimate the p-value and the BH performance in FDR and FNR. In particular, this analysis allows to choose the optimal cardinal of the calibration set. Second, it is shown that a global control of the FDR on the whole time series can be obtained from a local control of the mFDR (modified FDR) on subseries. This makes it possible to control the global FDR without knowing the whole series. A local control of the mFDR is proposed by modifying the Benjamini-Hochberg procedure to obtain a threshold selection procedure that controls the FDR of the entire series. This multiple testing procedure is empirically evaluated on stationary time series.
- Chapter 4: This chapter develops a new anomaly detector for piecewise iid time series. It uses breakpoint detection, in particular the KCP introduced in Section 1.2.3.4. Once the segments are identified, scores can be learned for each segment. Second, the threshold selection procedure developed in Chapter 3 guarantees the FDR control of the anomaly detector. Several difficulties due to the online context had to be overcome:
  - When discovering new segments there are not enough points to learn normality. It is proposed to use information from previous segments to improve p-value estimation and enrich decision-making.
  - Online estimates of the position of breakpoints and the value of atypicity scores are a source of error and may need to be re-evaluated. Methods for learning the level of confidence that can be placed on decisions made in real time are being developed.

  The anomaly detector is empirically evaluated in depth to assess its detection capabilities and limitations.

Chapter 2

Efficient and Robust P-value Estimation using Kernel

This chapter is an attempt to improve the KDE-based p-value estimator. By redefining the selection criterion for the bandwidth parameter, the goal is to improve the accuracy of the p-value estimator. Instead of choosing the bandwidth that minimizes the density estimation error, the error in estimating the p-value at the tail of the distribution is minimized. By combining this approach with median-of-means, the goal is to build a robust and efficient estimator of the p-value tailored to anomaly detection. In this chapter, first an estimator of the new criterion is built, which is used to select the bandwidth parameter. Then, this procedure is empirically evaluated against the existing approaches.

2.1 New criterion for p-value estimation

It is assumed that a time series (X_t) is observed in an online context, to which an atypicity score s_t is assigned at each data point. This score cannot be interpreted by itself, but must be compared with the scores under the reference distribution using the p-value. The p-value is estimated using a p-value estimator, $\hat{p}$, on a set of data scores called the calibration set and denoted $\mathcal{S}^{cal}$. To simplify the notation, it is assumed throughout this chapter that $s_t = X_t$. In the same desire for simplification, the calibration set is not specified in the p-value arguments.

$$\hat{p}_t = \hat{p}(s_t, \mathcal{S}^{cal}) = \hat{p}(X_t) \tag{2.1}$$

Furthermore, the calibration set data points are assumed to be iid. This chapter focuses on finding the best p-value estimator.

The properties required for a good p-value estimator are as follows:

- Agnosticity: It should be possible to use the p-value estimator without knowing the probability distribution of the scores. The estimate of the p-value should be accurate regardless of the score distribution.

- Robustness: Since the calibration set used to estimate the p-value may contain anomalies, the p-value estimator must be robust to the presence of anomalies in the calibration set. The presence of a small fraction of extreme data should not affect the p-value estimation.

- Efficiency: The goal is to maximize the precision of the estimation, even if the calibration set contains few points.

In the literature, there are several estimators of the p-values considered. The most commonly used p-value estimators in anomaly detection are the Gaussian estimator [32] and the empirical p-value estimator [100, 33]. The Gaussian estimator is robust if the Gaussian parameters are robustly estimated. Its parametric nature makes it unsuitable for non-Gaussian distributions. The empirical estimator is distribution agnostic, but requires more data and is not robust. The p-value can also be estimated by integrating the tail distribution of a Kernel Density Estimator (KDE) [112]. The use of kernels smooths the estimation, reduces the variance and improves the efficiency of the estimator[164]. In its classical version, KDE is not robust, but it can be made robust using the Median-of-Means strategy [81]. Other estimators use extreme value theory [149]. This guarantees that in many cases the tail distribution asymptotically follows a generalized Pareto distribution [129, 130]. The limitation of these estimators in practice is the lack of guarantees outside the asymptotic behavior and the absence of robust estimators of the parameters of the generalized Pareto distribution.

The p-value estimator based on KDE is more attractive because it is agnostic to the score distribution, more efficient than the empirical p-value estimator, and can be made robust using strategies like MoM-KDE.

Definition 2.1 (Kernel density estimator [150])**.** *Let X_1^n be the calibration set generated from the reference distribution $\mathcal{P}_0$. Let K be a kernel, a non-negative function whose integral is 1. Let h be a real positive, called the bandwidth. Let K_h be the kernel dilated horizontally by h:*

$$K_h(\cdot) = \frac{1}{h} K\left(\frac{\cdot}{h}\right) \tag{2.2}$$

The Kernel Density Estimator of the value x is computed as follows:

$$\begin{aligned}\hat{f}(x) &= \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \\ &= \frac{1}{n} \sum_{i=1}^{n} K_h(x - X_i)\end{aligned}$$

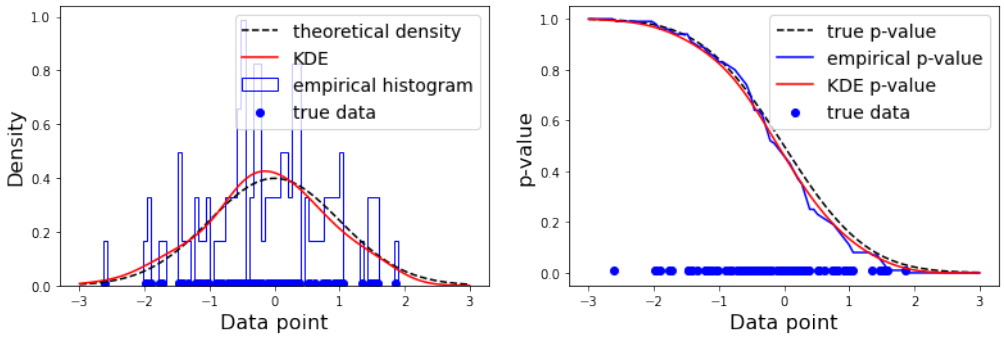
Noting $\mathcal{I}_{K_h}(x)$ the kernel K_h integrated from x to $+\infty$:

$$\mathcal{I}_{K_h}(x) = \int_x^{+\infty} K_h(z)dz \tag{2.3}$$

The p-value estimation based on KDE is computed as follows.

$$\begin{aligned}\hat{p}_h(x) &= \int_x^\infty \hat{f}(z)dz \\ &= \frac{1}{n}\sum_{i=1}^{n}\int_x^\infty K_h(z - X_i)dz \\ &= \frac{1}{n}\sum_{i=1}^{n}\mathcal{I}_{K_h}(x - X_i)\end{aligned}$$

Definition 2.1 introduces the Kernel Density Estimator (KDE) and a p-value estimator, noted $\hat{p}_h$, which integrates KDE. The KDE estimator is an efficient estimator of the density. By using kernel that smooths the predictions, the density estimation can be accurate even when the number of points in the calibration set is small. In this section, the kernel used is the Gaussian kernel. Figure 2.1 illustrates the performance differences between empirical and KDE estimators for estimating a Gaussian distribution (or p-value), using 100 data points. The x-axis represents the value of the data points and the y-axis represents the density (or p-value). The blue dots are the points generated by the Gaussian. They are used to estimate the true value, represented by the black-dashed curve. The curve of the KDE estimator, in red, is smoother than the one of the empirical estimator due to the use of kernels. This improves the performance of both the density and the p-value estimators, even with limited data.



(a) Estimation of the density

(b) Estimation of the p-value

Figure 2.1: Comparison between empirical estimator and KDE.

However, KDE is not robust, the estimation of the density and of the p-values are biased by anomalies in the calibration set. For this reason the Median-of-Means KDE is introduced.

Definition 2.2 (Median-of-Means KDE [81])**.** *Let X_1^n be the calibration set generated from the reference distribution $\mathcal{P}_0$. Let h be a real positive, called the bandwidth. Let S the number of blocs and $\mathcal{B}_1, \ldots, \mathcal{B}_S$ a partition of the calibration set.*

The Median-of-Means KDE estimates the density as follows:

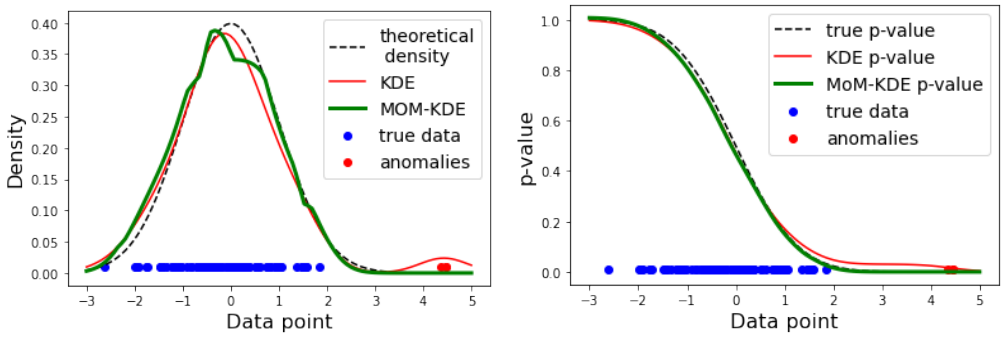
$$\hat{f}_{MoM,\ h}(x) = Median\left\{\frac{1}{h|\mathcal{B}_1|}\sum_{X\in\mathcal{B}_1} K\left(\frac{x-X}{h}\right), \ldots, \frac{1}{h|\mathcal{B}_S|}\sum_{X\in\mathcal{B}_S} K\left(\frac{x-X}{h}\right)\right\} \tag{2.4}$$

It can then be used to provide an estimation of the p-value as follows:

$$\hat{p}_h(x) = \int_x^\infty \hat{f}_{MoM,h}(z)dz$$

The median-of-means approach has been used in many areas of statistics [117, 48] and machine learning [102, 79] to improve the robustness of an estimator. The idea is to split a set into several blocks $\mathcal{B}$, apply the non-robust estimator (e.g., the mean or KDE) to each of the blocks. Then the final estimation is obtained by taking the median of these estimates. Thus, if a minority of the blocks contain anomalies, a minority of the estimators are biased and do not affect the median. The number of blocks is an important parameter in this method. It must be large enough to ensure that a sufficient number of blocks contain zero anomalies. However, if the set is divided into too many blocks, each block contains too few points and the estimation will be poor. [81] gives the convergence rate of the density estimation as a function of the number of blocks and the total number of data points. For a constant error, the required calibration set cardinality is almost proportional to the number of blocks (error constant in $\frac{Sn}{\log n}$). If the number of anomalies in the calibration set is known and denoted n_1, then the ideal number of blocks is $|\mathcal{B}| = 2n_1 + 1$. Possible alternatives to block partitioning could be to take the median of predictions obtained by bootstrapping. However, in this case, there is no guarantee of an unbiased prediction with $2n_1 + 1$ resamples.

Figure 2.2 illustrates how anomalies affect the estimation of the classical KDE and the MoM-KDE. The x-axis represents the data point value and the y-axis represents the density (or p value). The true value to be estimated is represented by the black dashed curve. Data points used by the estimators are represented by dots at the bottom of the plot. Red dots are anomalies not generated by the reference Gaussian distribution. The red line, which is the estimation from KDE, has a bump around the anomalies that biases the estimate of the density and p-value. In contrast, the MoM-KDE is not affected by the presence of anomalies and the estimate is closer to the true value.



(a) Estimation of the density

(b) Estimation of the p-value

Figure 2.2: Comparison between the classical KDE and MoM-KDE in the presence of anomalies.

KDE performance depends on the choice of kernel K_h. According to [170, 150], the choice of the symmetrical kernel family has no significant impact (e.g. Gaussian), compared to the choice of the bandwidth, denoted h. In this chapter, the focus is on the case of the Gaussian kernel

$K(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$ and on optimizing the choice of bandwidth for anomaly detection.

The classic criterion for selecting the bandwidth is to minimize the density estimation error.

Definition 2.3 (Density MISE)**.** *Let f be the true density and $\hat{f}_h$ the KDE estimation of f with bandwidth h.*

$$MISE_f(h) = \mathbb{E} \int_{-\infty}^{+\infty} (\hat{f}_h(x) - f(x))^2 dx \tag{2.5}$$

Our preliminary experiments show that this criterion gives poor results in the context of anomaly detection. This is due to KDE's difficulty in estimating tail of the distribution. Choosing bandwidths that are too small results in a large number of false positives. Similar issue was encounter by [90]. However, their solution, based on Topological Data Analysis, does not allow the p-value to be estimated directly, and therefore requires the use of another p-value estimator downstream of the KDE estimator.

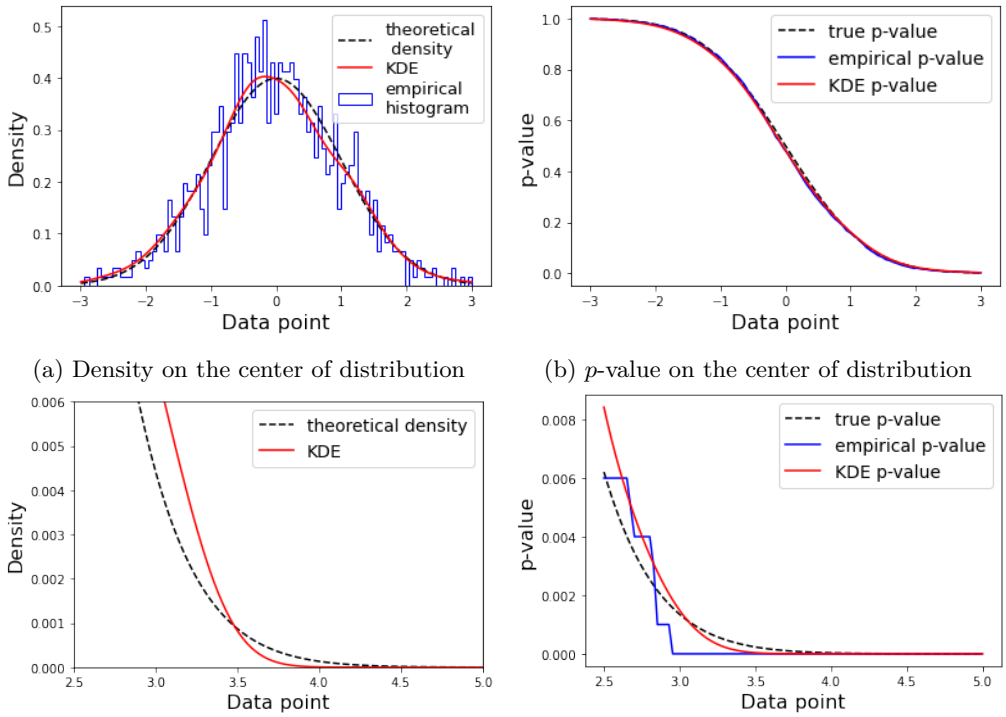
Figure 2.3d illustrates KDE's difficulty in estimating the tail of the distribution. The first two figures illustrate the estimation of the density shown in Figure 2.3a and the p-value shown in Figure 2.3b, at the center of the distribution, for data point values in $[-3, 3]$. The last two figures illustrate the estimation of the density shown in Figure 2.3c, and the p-value shown in Figure 2.3d, at the tail of the distribution, for x in $[2.5, 5]$. In these examples, 1000 points are generated using a normal distribution. Then the density and p-value are estimated using KDE, with the bandwidth selected using the "rule of thumb" method. Figure 2.3b shows that the relative error in estimating the p-value is very low at the center of the distribution. On the other hand, the relative estimation error is much higher in the tail of the distribution, as shown in Figure 2.3d where the gap between the curve associated to the KDE p-value and the theoretical curve is quite large.

To refine the criterion for selecting the bandwidth, let's recall how the p-value is used for anomaly detection. The last step of anomaly detection is to compare the estimated p-value $\hat{p}_t$ with the threshold noted as ε. A data point is detected as an anomaly if the p-value is less than ε. To improve the anomaly detector, a better estimator of the p-value is needed. An ideal estimator of the p-value should lead to the same decision as if the true p-value were known. This can be written as: $\mathbb{1}[\hat{p}_t < \varepsilon] = \mathbb{1}[p_t < \varepsilon]$. To achieve this result, the goal is to minimize the error in estimating the p-value at the tail of the distribution, for values where the p-actual value is close to ε. However, using the classic $MISE_f$, the bandwidth is optimized to correctly estimate the center of the distribution, at the expense of the tail of the distribution. The $MISE_p$ criterion, introduced by Definition 2.4, proposes to minimize the estimation error of the p-value on the tail of the distribution only. Indeed, in anomaly detection, errors of estimation on the center of the distribution are less problematic, since the data are known to be normal. On the contrary, for points whose p-value is close to ε, even a small error can change the detection result. In addition, since the density value is not used directly in anomaly detection, the estimation error on the density is replaced by an estimation error on the p-value.

Definition 2.4 (p-value MISE)**.** *Let p be the true p-value and $\hat{p}_h$ the KDE estimation of p. Let s be a real number.*

$$MISE_p(h) = \mathbb{E} \int_{s}^{+\infty} (\hat{p}_h(x) - p(x))^2 dx \tag{2.6}$$

$$= \mathbb{E}||\hat{p}_h - p||_s^2 \tag{2.7}$$



(a) Density on the center of distribution

(b) p-value on the center of distribution

(c) Density on the tail of distribution

(d) p-value on the tail of distribution

Figure 2.3: Illustration of the difficulty to estimate the tail of the distribution.

The s parameter is used to define where the tail of the distribution begins. It has to be set by the user. In this chapter, s is set to two standard deviations. In practice, the standard deviation needs to be robustly estimated. The choice of s is a trade-off between having enough data to make the estimation and being restricted to the tail of the distribution. An s that is too large implies a high variance due to the small sample size, while an s that is too large induces a form of bias because data from the centre of the distribution are included. The notation $||\cdot||_s^2 = \int_s (\cdot)^2 dx$ is not a norm, but is used to keep the notation consistent with existing literature on density estimation.

The chapter focuses on building a procedure for selecting the bandwidth h that minimizes the p-value MISE. In Section 2.2, a literature review is conducted in order to identify existing procedures for minimizing the density MISE. The following sections present research on adapting these procedures to the p-value MISE criterion. Section 2.3 presents the adaptation research on the Asymptotic MISE (AMISE) criterion. The adaptation of the Leave-One-Out (LOO) estimator is presented in Section 2.4. The Penalized Comparison to Overfitting (PCO) estimator is presented in Section 2.5. Section 2.6 empirically compares the various bandwidth selection procedure in order to determine which one minimizes the p-value criterion. Finally, the approach taken in this chapter is discussed in Section 2.7.

2.2 Review on strategies for bandwidth selection

To build a method for the selection of the bandwidth, let's take inspiration from existing methods that minimize the $MISE_f$ criterion before adapting them to $MISE_p$. This section provides an overview of the various selection methods available in the literature. In the Section 2.4-2.6, new procedures are built and evaluated. The reviews presented in [89, 77] are the starting point for the following non-exhaustive review. The various procedures are based on the minimization of a criterion that serves as a proxy for the minimization of the unknown $MISE_f$ value. Selection procedures are classified according to how the criterion attempts to estimate MISE.

- By simplifying the calculation of the MISE by its asymptotic value, which is called the AMISE.
- By using resampling methods to approximate the unknown function f.
- By using a penalized criterion.

The purpose of this classification is to guide the choice of methods to be adapted to the case of p-value MISE.

2.2.1 AMISE

Asymptotic MISE analysis is a commonly used technique to study the effect of the bandwidth h. The MISE criterion is asymptotically approximated ($n \to \infty$) by the AMISE criterion [89].

$$AMISE_f(h) = n^{-1}h^{-1}R(K) + h^4R(f'')\left(\int y^2K/2\right)^2 \tag{2.8}$$

with $\phi \in \{K, f''\}, R(\phi) = \int_{-\infty}^{+\infty} \phi(x)^2dx$. And f'' is the second derivative of the density to estimate.

The minimizer of the AMISE can be calculated by:

$$\hat{h}_{AMISE} = \left(\frac{R(K)}{nR(f'')\left(\int x^2K/2\right)^2}\right)^{1/5} \tag{2.9}$$

The AMISE criterion is often a good approximation of the $MISE_f$ criterion, the $\hat{h}_{AMISE}$ is an interesting bandwidth where the only unknown value is $R(f'')$. The following procedures suggest different ways to estimate this value.

2.2.1.1 Rule-of-thumb

A simple solution and widely used solution, called the rule-of-thumb, introduced by [46], is to replace $R(f'')$ by an estimation assuming that f belongs to a parametric family, such as the Gaussian family. When a Gaussian family is assumed, the rule-of-thumb gives $\hat{h}_{RT} = 1.05\hat{\sigma}n^{-1/5}$, where n is the cardinality of the calibration set and $\hat{\sigma}$ an estimation of the standard deviation.

2.2.1.2 Biased Cross-Validation

In the Biased Cross-Validation method, the unknown value $R(f'')$ is replaced with $R(\hat{f}''_h) - R(K'')/(mh)$. Where $R(\hat{f}''_h)$ is an estimator of $R(f'')$ using the KDE of f with bandwidth. Since

$R(\hat{f}_h'')$ introduces a bias equal to $R(K'')/(mh)$, the term $-R(K'')/(mh)$ is a bias correction term to build an unbiased estimator [144].

2.2.1.3 Plug-in method

The "plug-in" method consists in estimating $R(f'')$ by $R(\hat{f}_a'')$ where $\hat{f}_a$ is a KDE of f and a is the bandwidth, different from h. Plug-in methods provide ways to select the parameter a in order to minimize the estimation error of $R(f'')$ and plug this estimation into Eq. 2.9. There exist different plug-in methods [146, 74].

2.2.2 Resampling

2.2.2.1 Least Squared Cross Validation

Based on the expansion of the Integrated Squared Error (ISE) criterion:

$$ISE_f(h) = \int (\hat{f}_h - f)^2 = \int \hat{f}_h^2 - 2\int \hat{f}_h f + \int f^2 \tag{2.10}$$

Since the last term does not depend on h, ISE has the same minimizer as the first two terms. Only the second term cannot be computed exactly. Using the definition of expectation: $\int \hat{f}_h f = \mathbb{E}_{X\sim\mathcal{P}_0}[\hat{f}_h(X)]$, the unbiased Leave-One-Out estimator is suggested [139, 26], with $\hat{f}_{h,-i}$ being the KDE estimator for which X_i has been removed from the calibration set.

$$\mathbb{E}_{X\sim\mathcal{P}_0}[\hat{f}_h(X)] \approx \frac{1}{n}\sum_{i=1}^{n} \hat{f}_{h,-i}(X_i) \tag{2.11}$$

2.2.2.2 Smoothed Bootstrap method

The density function f is replaced by a KDE estimate $\hat{f}_a$, where a is a bandwidth that can be chosen either by the rule of thumb or equal to h. Then the MISE can be fully expressed, where $\mathbb{E}_{X^*\sim\hat{f}_a}$ is the expectation according the bootstrap distribution $\hat{f}_a$ and $\hat{f}_h^*$ is the KDE estimation using the bootstrap data $\{X_1^*, \ldots, X_n^*\}$:

$$BMISE(h) = \mathbb{E}_{X^*\sim\hat{f}_a} \int (\hat{f}_h^*(x) - \hat{f}_a(x))^2 \tag{2.12}$$

In practice, the $BMISE$ can be expressed more directly [161], there is no need to do the bootstrap explicitly.

2.2.3 Penalized criterion

Birgé and Massart suggest to treat the bandwidth selection as a model selection problem in [17, 16]. The statistical tools available in the model selection literature make it possible to develop a number of bandwidth selection procedures with solid statistical guarantees in non-asymptotic regime. In general, these approaches consist of minimizing a penalized criterion, with L_n an empirical risk and pen the penalty function:

$$Crit(h) = L_n(\hat{f}_h) + pen(h) \tag{2.13}$$

As an example the Goldenshluger-Lepski method [64] in a pairwise based procedure and minimize the following criterion:

$$Crit_{LG}(h) = \sup_{h'} \left(\int (\hat{f}_h(x) - \hat{f}_{max(h,h')}(x))^2 dx - V_1(h') \right) + V_2(h) \tag{2.14}$$

where V_1 and V_2 are two penalties developed in [64].

Another example is the Penalized Comparison to Overfitting developed in [97] where the estimation $\hat{f}$ is compared to the overfitted estimator $\hat{f}_{h_{min}}$, where h_{min} is a small bandwidth and $\frac{2}{n} \int K_h(x) K_{h_{min}}(x) dx$ is the penalty term:

$$Crit_{PCO}(h) = \int (\hat{f}_h(x) - \hat{f}_{h_{min}}(x))^2 dx + \frac{2}{n} \int K_h(x) K_{h_{min}}(x) dx \tag{2.15}$$

2.2.4 Bandwidth selection for MoM-KDE

The Median-of-Mean KDE is less studied than classical KDE so there is less work about bandwidth selection. In the article that introduces MoM-KDE [81] the authors use cross-validation method to select the parameter h.

2.3 Derivation of asymptotic criterion

In this section it is shown that it is not possible to calculate an AMISE criterion, defined in Section 2.2.1, for the p-value in the same way as for the density. This means that selection methods based on this criterion cannot be adapted. Let's follow the proof of the AMISE derivation of the density as described in [40] and apply it to the p-value. The $MISE_p$ is decomposed in bias and variance as follows:

$$MISE_p(h) = \mathbb{E} \int_s^{+\infty} (\hat{p}_h(x) - p(x))^2 dx \tag{2.16}$$

$$= \underbrace{\mathbb{E} \int_s^{+\infty} (p_h(x) - p(x))^2 dx}_{B(h)} + \underbrace{\mathbb{E} \int_s^{+\infty} (\hat{p}_h(x) - p_h(x))^2 dx}_{Va(h)} \tag{2.17}$$

where $p_h(x) = \mathbb{E}\hat{p}_h(x)$, $B(h)$ is the bias term and $Va(h)$ the variance term. In this section, to approximate the case of density, it is assumed that the cross term is 0. In the following, the asymptotic behavior of the bias is derived and it is shown that no asymptotic behavior can be derived for the variance.

2.3.1 Asymptotic bias

Proposition 2.1 gives the asymptotic behavior of the bias term.

Proposition 2.1 (Asymptotic bias)**.** *Let $\mathcal{P}_0$ be the reference distribution with f the density function and let K be a kernel. Let b be the measure of bias associated with p-value estimation using KDE. Assuming f is $\mathcal{C}^2$ and K is symmetric, then the asymptotic behavior of the bias is*

known and can be calculated as follows

$$B(h) = 0.25h^4 \left(\int_{-\infty}^{+\infty} y^2 K(y) dy \right)^2 \int_s^{+\infty} \left(\int_x^{+\infty} f''(z) dz \right)^2 dx + o(h^4) \tag{2.18}$$

The formula given in Eq. 2.1 is similar to the bias in the AMISE of density formula given in Eq. 2.8, where $\int_{-\infty}^{+\infty} f''(z)^2 dz$ is replaced by $\int_s^{+\infty} \left(\int_x^{+\infty} f''(z) dz \right)^2 dx$. In particular the bias is also $\sim h^4$ as for the $MISE_f$. Assuming that the same kind of results could be obtained for the variance, the different procedures seen in section 2.3 could be adapted for the p value.

The proof is inspired from [40] and delayed to Section 2.10.1.

2.3.2 Issue for the variance

This section shows why it is impossible to obtain asymptotic results for the variance that are similar to those obtained for the density. In the following, the same calculation steps as in the density case, described in [40], are reproduced for the case of p-values.

At first, the definition of the variance term is used

$$\begin{aligned} Va(h) &= \mathbb{E} \| \hat{p}_h - p_h \|_s^2 \\ &= \mathbb{E} \int_s^{+\infty} (\hat{p}_h(x) - p_h(x))^2 dx \end{aligned}$$

Then, the integration order is changed, assuming Fubini's theorem:

$$Va(h) = \int_s^{+\infty} \mathbb{E} (\hat{p}_h(x) - p_h(x))^2 dx \tag{2.19}$$

$$= \int_s^{+\infty} \mathrm{Var}(\hat{p}_h(x))^2 dx \tag{2.20}$$

After that, the KDE estimator is written as a sum of kernels, before using the independence property:

$$Va(h) = \int_s^{+\infty} \mathrm{Var} \left(\frac{1}{hn} \sum_{i=1}^{n} \int_x^{+\infty} K(\frac{z - X_i}{h}) dz \right) dx \tag{2.21}$$

$$= 1/(h^2 n) \int_s^{+\infty} \mathrm{Var} \left(\int_x^{+\infty} K(\frac{z - X_i}{h}) dz \right) dx \tag{2.22}$$

The solution suggest by [40] to deal with the variance term is to upper bound it by the second

moment, as follows:

$$Va(h) = 1/(h^2 n) \int_s^{+\infty} \mathrm{Var}\left(\int_x^{+\infty} K(\frac{z - X_i}{h}) dz \right) dx$$
$$\leq 1/(h^2 n) \int_s^{+\infty} \mathbb{E}\left(\int_x^{+\infty} K(\frac{z - X_i}{h}) dz \right)^2 dx$$

By variable substitution, $\frac{z - X_i}{h} \to y$, it gives:

$$Va(h) \leq 1/(n) \int_s^{+\infty} \mathbb{E}\left(\int_{\frac{x - X}{h}}^{+\infty} K(y) dy \right)^2 dx$$

Recalling the notation $\mathcal{I}_K(w) = \int_w^{\infty} K(y) dy$ is introduced, it gives:

$$\mathbb{E}(\mathcal{I}_K(\frac{x - X}{h})^2) = \int_{-\infty}^{+\infty} f(X) \mathcal{I}_K(\frac{x - X}{h})^2 dX$$

A variable substitution gives:

$$\mathbb{E}(\mathcal{I}_K(\frac{x - X}{h})^2) = \int_{-\infty}^{+\infty} f(x - hy) \mathcal{I}_K(y)^2 h dy$$

The series expansion at second order gives:

$$\mathbb{E}(\mathcal{I}_K(\frac{x - X}{h})^2) = \int_{-\infty}^{+\infty} (f(x) - hyf'(x) + o(h)) \mathcal{I}_K(y)^2 h dy$$
$$\mathbb{E}(\mathcal{I}_K(\frac{x - X}{h})^2) = hf(x) \int_{-\infty}^{+\infty} \mathcal{I}_K(y)^2 dy - h^2 f'(x) \int_{-\infty}^{+\infty} y \mathcal{I}_K(y)^2 dy + o(h^2)$$

But the term $\int_{-\infty}^{+\infty} y \mathcal{I}_K(y)^2$ is infinite since $\mathcal{I}_K(y)$ converges to 1 as y converges to $-\infty$. The only way to avoid divergence at $-\infty$ is not to separate f and $\mathcal{I}_K$, which makes it impossible to get an AMISE-type result.

2.3.3 Conclusion

It is not possible to get an exploitable variance expression for the $MISE_p$ criterion.
This prevents the use of methods developed for the $MISE_f$ criterion using AMISE. For this reason, the focus is on approaches based on cross-validation in Section 2.4 or penalized criteria 2.5 that do not use AMISE.

2.4 Least squared error cross validation Estimator

This section presents the theoretical derivation of a selection procedure inspired by Section 2.2.2.1.

2.4.1 Leave-One-Out estimator (LOO)

Let p be the p-value function of the score under $\mathcal{P}_0$ and $\hat{p}_h$ be the KDE estimation of p. Let $ISE_p(h)$ be the Integrated Squared Error when estimating the p-value using KDE with the bandwidth h. $MISE_p(h) = \mathbb{E}[ISE_p(h)]$ and $ISE_p(h)$ can be decomposed into a quadratic, a multiplier and a constant term:

$$ISE_p(h) = \int_s^{+\infty} (\hat{p}_h(x) - p(x))^2 dx \tag{2.23}$$

$$= \underbrace{\int_s^{+\infty} \hat{p}_h(x)^2 dx}_{C(h)} - 2 \underbrace{\int_s^{\infty} \hat{p}_h(x) p(x) dx}_{D(h)} + \text{cste}. \tag{2.24}$$

The aim is to find the parameter h that minimize the ISE criterion.

$$\hat{h}_{ISE} = \arg\min_h ISE_p(h) \tag{2.25}$$

The quadratic term of Eq. 2.24 can be calculated directly since all values are known. However, the multiplicative term, denoted by $D(h)$, cannot be calculated directly because p is unknown. Therefore, this term must be estimated. Proposition 2.2 allows to rewrite $D(h)$ in a more appropriate way to build an estimator.

Proposition 2.2. *Let $\mathcal{P}_0$ be the reference distribution. Let $(X_i)_{i=1}^n$ be the calibration set and $\hat{p}_h$ the KDE estimator of the p-value function p. The multiplicative term $D(h)$ introduced in Eq. 2.24 can be expressed as follows:*

$$D(h) = \mathbb{E}_{X \sim \mathcal{P}_0} \left[\mathbb{1}[X > s] \int_s^X \hat{p}_h(x) dx \right]$$

Proof of Proposition 2.2. As introduced in Eq. 2.24

$$D(h) = \int_s^{+\infty} \hat{p}_h(x) p(x) dx$$

By definition of the p-value, $p(x) = \int_x f(z)dz = \int \mathbb{1}[z > x] f(z) dz$, which gives:

$$D(h) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \mathbb{1}[x > s] \mathbb{1}[z > x] \hat{p}_h(x) f(z) dz dx$$

At this point, the aim is to change the integration order, to introduce the expression $\mathbb{E}_{X \sim \mathcal{P}_0}$. To do this, all integration limits are written as a function of z rather than x. Since $x > s$ and $z > x$ is equivalent to $z > s$ and $z > x > s$, this implies that

$$\mathbb{1}[x > s] \mathbb{1}[z > x] = \mathbb{1}[z > s] \mathbb{1}[z > x > s]$$

and then:

$$\begin{aligned} D(h) &= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \mathbb{1}[z > s]\mathbb{1}[z > x > s]\hat{p}_h(x)f(z)dzdx \\ &= \int_{-\infty}^{+\infty} \mathbb{1}[z > s] \left(\int_{-\infty}^{+\infty} \mathbb{1}[z > x > s]\hat{p}_h(x)dx \right) f(z)dz \end{aligned}$$

The expectation of $X \sim \mathcal{P}_0$ can be recognized.

$$\begin{aligned} D(h) &= \mathbb{E}_{X\sim\mathcal{P}_0} \left[\mathbb{1}[X > s] \int_{-\infty}^{+\infty} \mathbb{1}[X > x > s]\hat{p}_h(x)dx \right] \\ &= \mathbb{E}_{X\sim\mathcal{P}_0} \left[\mathbb{1}[X > s] \int_{s}^{X} \hat{p}_h(x)dx \right] \end{aligned}$$

□

The new formulation of the term $D(h)$, introduced by Proposition 2.2, allows to build an estimator by replacing the expectation by the sample mean.

Definition 2.5 (LOO estimator of $D(h)$)**.** *Let $(X_i)_1^n$ be random variable generated from $\hat{P}_0$. Let K be a kernel. The LOO estimator of $\hat{D}(h)$ can be written as:*

$$\hat{D}(h) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}[X_i > s] \int_{s}^{X_i} \hat{p}_{h,-i}(x)dx \tag{2.26}$$

$$\hat{D}(h) = \frac{1}{(n-1)n} \sum_{i=1}^{n} \sum_{j=1;j\neq i}^{n} \mathbb{1}[X_i > s] \int_{s}^{X_i} \int_{x}^{+\infty} K(\frac{z - X_j}{h})dzdx \tag{2.27}$$

In the general case, the integrals are approximated by a trapezoidal method. In the following section, a close form of this estimator is given in the case where the kernel used is the Gaussian kernel.

2.4.2 Close form for the D term

The following Proposition 2.3 gives a close form for the LOO estimator $\hat{D}(h)$ in the case of Gaussian kernel. Such close forms are interesting to improve the precision of the approximation and to reduce the computation time, since no trapezoidal computation is needed.

Proposition 2.3 (Close form of $\hat{D}(h)$, for Gaussian Kernel)**.** *Let $(X_i)_1^n$ be the calibration set generated. And K the Gaussian kernel. The LOO estimator of $\hat{D}(h)$ stated in Definition 2.5 can be written as:*

$$\begin{aligned} \hat{D}(h) = \frac{1}{h(n-1)n} \sum_{i=1}^{n} \mathbb{1}[X_i > s] \sum_{j=1,j\neq i}^{n} \Bigg[& h^2 \exp\left(-\frac{(s - X_j)^2}{2h^2}\right) - h^2 \exp\left(-\frac{(X_i - X_j)^2}{2h^2}\right) + \\ & + (X_j - s)\frac{\sqrt{\pi}}{2} h\sqrt{2}[erf(\frac{X_i - X_j}{h\sqrt{2}}) - erf(\frac{s - X_j}{h\sqrt{2}})] - (X_i - s)h\frac{\sqrt{\pi}}{\sqrt{2}} \left(1 - erf\left(\frac{X_i - X_j}{h\sqrt{2}}\right)\right) \Bigg] \end{aligned}$$

The proof is delayed to Section 2.10.2.

2.4.3 Leave-One-Out estimator for MoM-KDE

The presence of the mean in $\hat{D}$, as stated in Definition 2.5, makes the estimator sensitive to the presence of anomalies. A way to make the estimator more robust is introduced. To make the estimation of $D(h)$ more robust, the expectation in Proposition 2.2 is replaced by a Median-of-Means.

Definition 2.6 (Robust estimator LOO estimator for MoM-KDE)**.** *Let $(X_i)_1^n$ be the calibration set. Let K be a kernel. Let S the number of blocs and $\mathcal{B}_1, \dots, \mathcal{B}_S$ a partition of the calibration set. The Median-of-Means LOO estimator $\hat{D}_{MoM}(h)$ can be written as:*

$$\hat{D}_{MoM}(h) = Median\left\{ \frac{1}{|\tilde{\mathcal{B}}_1|} \sum_{\tilde{X} \in \tilde{\mathcal{B}}_1} \mathbb{1}[\tilde{X} > s] \int_s^{\tilde{X}} \int_x \hat{f}_{MoM,h,\setminus\tilde{X}}(z) dz dx, \dots, \right.$$
$$\left. \dots, \frac{1}{|\tilde{\mathcal{B}}_S|} \sum_{\tilde{X} \in \tilde{\mathcal{B}}_S} \mathbb{1}[\tilde{X} > s] \int_s^{\tilde{X}} \int_x \hat{f}_{MoM,h,\setminus\tilde{X}}(z) dz dx \right\}$$

The blocks used to calculate $\hat{D}_{MoM}$ may be different from those used to calculate $\hat{f}_{MoM}$ in Definition 2.2. In particular, if only one block is used for $\hat{f}_{MoM}$ this provides a way to calculate $\hat{D}_{MoM}$ robustly with classic KDE. As for MoM-KDE, the ideal number of blocks is $2 * n_1 + 1$, where n_1 is the number of anomalies. Unlike the mean, the median operator is not commutative with integration, so it is not possible to obtain a close form for this estimator, and each integral is approximated by the trapezoidal method.

2.4.4 Computation of $C(h)$

The second term in Eq. 2.24, labeled $C(h)$, does not need to be estimated since all quantities are known. However, there is no close form for $C(h)$, even if the kernels are Gaussian.

$$C(h) = \int_s^{+\infty} \hat{p}_h(x)^2 dx \tag{2.28}$$
$$= \sum_{i,j} \int_s^{+\infty} \mathcal{I}_{K_h}(x - X_i) \mathcal{I}_{K_h}(x - X_j) dx \tag{2.29}$$
$$\tag{2.30}$$

Thus, the value of the integral is approximated using numerical analysis methods such as the trapezoidal rule. When approximating the value of the integral on $[s, +\infty[$, two parameters need to be set: the choice of the upper bound of the integral, noted x_{max} and the choice of the integration step, noted δ_x. This approximation is written as $\tilde{C}(h, x_{max}, \delta_x)$. By reducing the integration step size, the estimation error is reduced while the computation time is increased. A procedure is needed to choose the two numerical integration parameters to respect a trade-off between accuracy and speed. Since $C(h)$ is written as a sum of Gaussians centered on X_i and with variance h^2, it is possible to get an idea of the value of these parameters. First, the tail of a Gaussian decreases rapidly toward 0. Thus, by setting the upper bound of the integration at 4 or 5 h from the maximum value of X_i, the error committed by integrating on $[s, x_{max}]$ instead

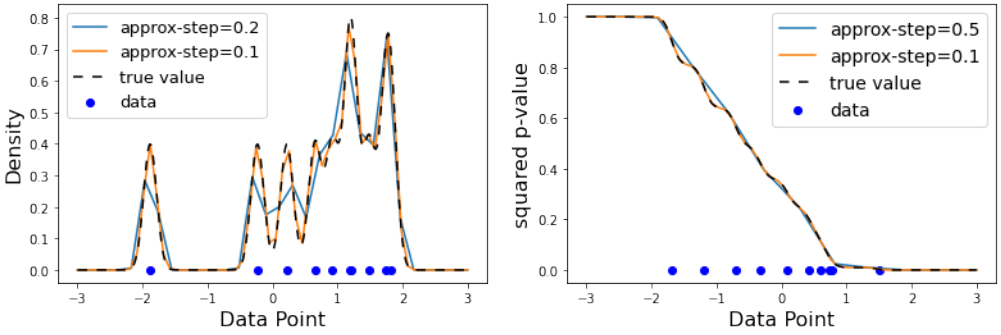
of $[s, +\infty[$ is small.

$$x_{max} = \max_i X_i + 5h$$

Moreover, h can be considered as the size of the smallest detail, on the x-axis, in the $\hat{p}_h^2$ curve. By taking a step size of the order of h, the error generated should be small.

$$\delta_x = h$$

Figure 2.4a illustrates the idea behind this heuristic. A KDE estimation is shown using 10 randomly generated points and a kernel with bandwidth $h = 0.1$. The approximations given by using different sampling steps are shown in different colors. The smaller the sampling step is, the closer the approximation is to the true value. Furthermore, with a step size of 0.1, equal to h, the approximation error is already very small. Figure 2.4b shows the same results, but on the square of the p-value, which is needed in to calculate $C(h)$. The approximation with a step of 0.1, equal to h, gives a very good approximation of the p-value.



(a) Density.

(b) Squared p-value.

Figure 2.4: Illustration of the effect of integration step size on the approximation of the density and of the squared p-value.

To support the validity of the heuristic, the approximation error as a function of the integration step is studied experimentally. A procedure is defined by repeating for a given bandwidth h:

1. Generate data from normal distribution, $\forall i \in [\![1, n]\!], \quad X_i \sim \mathcal{N}(0, 1)$
2. Compute the $C(h)$ approximation for different values of the integration step δx. $\mathcal{D}_x$ is the set of all values tested as the integration step.

$$\forall \delta_x \in \mathcal{D}_x, C_{\delta_x} = \tilde{C}(h, x_{max}, \delta_x)$$

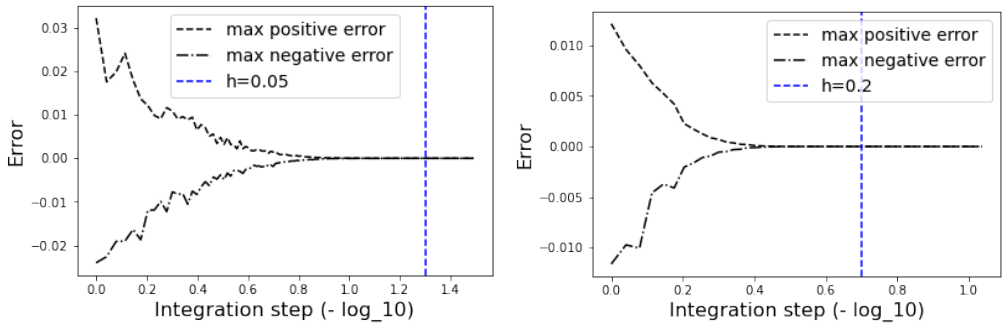
3. Calculate the approximation error by assuming that there is no approximation error when taking the smallest integration step $\delta_{x,min} = \min(\mathcal{D}_x)$, $C_{\delta_{x,min}} = C(h)$.

$$\forall \delta_x \in \mathcal{D}_x, r_{\delta_x} = C_{\delta_x} - C_{\delta_{x,min}}$$

The results are shown in Figure 2.5. The x-axis represents the size of the integration step in negative logarithmic scale. The y-axis represents the value of the approximation error. The black

curves represent the extreme error values obtained during the various tests, for each integration step tested. The vertical blue line represents the integration step at which, according to the heuristic, the integration error should be small.

The error decreases as the step size decreases (the value decreases to the right on a negative-logarithmic scale). The error quickly goes to zero before it reaches the value expected by the heuristic. The fact that the error is constant and equal to zero over a large range of data validates the idea of approaching the approximation error with $r_{delta_x} = C_{\delta_x} - C_{\delta_{x,min}} = C_{\delta_x} - C(h)$.



(a) $h = 0.05$ (b) $h = 0.2$

Figure 2.5: Error of approximation of the value of $C(h)$ as a function of the integration step size, for different bandwidths h.

2.4.5 Implementation

The complexity of the bandwidth selection algorithm, which requires calculating $\tilde{C}(h)$ and $\hat{D}(h)$ for the whole set of h, is high. Indeed, with n the number of points in the calibration set, n_x the number of step by the integration procedure and n_h the number of different tested bandwidth, the complexity is equal to $n^2 * n_x * n_h$. Given the high cost of loops in Python, it's tempting to vectorize operations using the Numpy library. But this leads to too high memory cost. Furthermore, since the median prevents the order of integration from being swapped, the computation of $\hat{D}_{MoM}$ has a higher complexity: $n^2 * n_x^2 * n_h$ (note the n_x^2 since there are two integrations). The implementation used in this chapter is based on the Numba [98] just-in-time compilation library. This optimizes loop calculation without having to store all intermediate results in vectors.

2.5 Penalized Comparison to Overfitting estimator

The goal of this section is to summarize the research efforts that have been made on adapting the PCO [97] procedure described in Section 2.2.3 to the p-value estimation problem. First, the difficulties encountered are shown when attempting to construct the penalty function by searching for an upper bound of the ideal penalty, as given in the proof of Theorem 9 of [97]. Another try have been done to define penalty using Akaike's heuristic, described in [97]. Finally, since the introduced quantities are neither computed nor upper bounded, they are replaced by estimators in order to construct a new procedure, described in Definition 2.7, for selecting the

bandwidth. The last result of the section shows that this procedure is equivalent to the LOO procedure of the Section 2.4.

2.5.1 Difficulty of extending PCO to the p-value estimation problem

The idea of PCO is to build a penalized criterion based on the comparison between $\hat{p}_h$ and the overfitting regime $\hat{p}_{h_{min}}$, where h_{min} is the smallest value of the search space. The goal is to define a criterion on the following form, where $pen(h)$ has to be chosen such that minimizing the criterion yields a good result with high probability:

$$Crit(h) = ||\hat{p}_h - \hat{p}_{h_{min}}||_s^2 + pen(h) \tag{2.31}$$

Then h can be chosen by minimizing the penalized criterion:

$$\hat{h}_{pco} = \arg\min_h Crit(h)$$

This implies, by definition:

$$||\hat{p}_{\hat{h}} - \hat{p}_{h_{min}}||_s^2 + pen(\hat{h}) \leq ||\hat{p}_h - \hat{p}_{h_{min}}||_s^2 + pen(h) \tag{2.32}$$

Then, some development gives:

$$||\hat{p}_{\hat{h}} - p||_s \leq ||\hat{p}_{\hat{h}} - p||_s + (pen(h) - 2\langle \hat{p}_h - p, \hat{p}_{\hat{h}} - p\rangle_s) - (pen(\hat{h}) - 2\langle \hat{p}_{\hat{h}} - p, \hat{p}_{\hat{h}} - p\rangle_s)$$

The term $\langle \hat{p}_h - p, \hat{p}_{\hat{h}} - p\rangle_s$ plays the role of the ideal penalty, it needs to be upper bounded to define the penalty. Expanding on this term, it gives:

$$\begin{aligned}\langle \hat{p}_h - p, \hat{p}_{\hat{h}} - p\rangle_s = &\langle \hat{p}_h - p_h, \hat{p}_{h_{min}} - p_{h_{min}}\rangle_s + \langle \hat{p}_h - p_h, p_{h_{min}} - p\rangle_s \\ &+ \langle p_h - p, \hat{p}_{h_{min}} - p_{h_{min}}\rangle_s + \langle p_h - p, \hat{p}_{h_{min}} - p\rangle_s\end{aligned}$$

The first term can be further expressed as

$$\begin{aligned}&\langle \hat{p}_h - p_h, \hat{p}_{h_{min}} - p_{h_{min}}\rangle_s = \\ &\sum_{i,j} \int_s^{+\infty} \left(\int_x^{\infty} K_h(y - X_i)dy - p_h(x)\right)\left(\int_x^{\infty} K_{h_{min}}(z - X_j)dz - p_{h_{min}}(x)\right)dx\end{aligned}$$

We have not managed to find an interesting expression for this term. The PCO criterion is studied using an heuristic evaluation.

2.5.2 Heuristic evaluation of the PCO criterion

Inspired from Akaike's heuristics in [97], given the difficulty of constructing such a penalized criterion, a heuristic approach is attempted to define the penalized criterion. Assuming that the average value of the penalized criterion should be equal to the MISE criterion, it gives:

$$\mathbb{E}[Crit(h)] = MISE_p(h) \tag{2.33}$$

The term $||\hat{p}_h - \hat{p}_{h_{min}}||_s^2$ in Eq. 2.31 can be seen as a biased estimator of the bias of the MISE criterion. Thus, it can be deduced that the ideal penalty must add the variance term as well as

the correction of the bias in order to reconstruct the total MISE criterion.

Indeed, the MISE criterion can be decomposed into bias and variance terms.

$$\mathbb{E}||\hat{p}_h - p||_s^2 = ||p_h - p||_s^2 + \mathbb{E}||\hat{p}_h - p_h||_s^2 \tag{2.34}$$

The bias of the "Comparison to overfitting" term is calculated as follows:

$$\hat{p}_h - \hat{p}_{h_{min}} = \hat{p}_h - \hat{p}_{h_{min}} - p_h + p_{h_{min}} + p_h - p_{h_{min}} \tag{2.35}$$

$$\mathbb{E}||\hat{p}_h - \hat{p}_{h_{min}}||_s^2 = \mathbb{E}||\hat{p}_h - \hat{p}_{h_{min}} - p_h + p_{h_{min}}||_s^2 + \mathbb{E}||p_h - p_{h_{min}}||_s^2 \tag{2.36}$$

$$\tag{2.37}$$

Finally the following criterion, inspired by PCO, is an unbiased estimator of MISE.

$$Crit_{PCO} = ||\hat{p}_h - \hat{p}_{h_{min}}||_s^2 - \mathbb{E}||\hat{p}_h - \hat{p}_{h_{min}} - p_h + p_{h_{min}}||_s^2 + \mathbb{E}||\hat{p}_h - p_h||_s^2 \tag{2.38}$$

The unknown of the problem is therefore the "penalty" term $-\mathbb{E}||\hat{p}_h - \hat{p}_{h_{min}} - p_h + p_{h_{min}}||_s^2 + \mathbb{E}||\hat{p}_h - p_h||_s^2$. All of our attempts to compute the variance and bias correction terms or to place a tight upper bound on them have been unsuccessful.

Therefore, these quantities are estimated to construct the full MISE estimator.

2.5.3 Estimator of the penalty for PCO criterion

The estimation of the penalty term for the PCO criterion is described in this section.

The variance term, can be written as:

$$\mathbb{E}||\hat{p}_h - p_h||_s^2 = \int_s^{+\infty} \mathbb{E}(\hat{p}_h(x) - p_h(x))^2 dx \tag{2.39}$$

$$= \int_s^{+\infty} \mathrm{Var}(\hat{p}_h(x)) dx \tag{2.40}$$

$$= \frac{1}{n} \int_s^{+\infty} \mathrm{Var}(\int_x K_h(x - X)) dx \tag{2.41}$$

$$\tag{2.42}$$

Reminding $\mathcal{I}_{K_h}(x)$ denotes the K_h kernel integrated from x to $+\infty$. The variance estimator is computed as:

$$\hat{Va}(h) = \frac{1}{n} \int_s^{+\infty} \left(\frac{1}{n-1} \sum_{i=1}^{n} (\mathcal{I}_{K_h}(x - X_i))^2 - \frac{1}{n(n-1)} (\sum_{i=1}^{n} \mathcal{I}_{K_h}(x - X_i))^2 \right) dx \tag{2.43}$$

Notice the $n-1$ term of the unbiased variance estimator. Similarly the bias correction term Δb

can be estimated by:

$$\hat{\Delta}b(h) = \frac{1}{n}\int_s^{+\infty}\left(\frac{1}{n-1}\sum_{i=1}^{n}(\mathcal{I}_{K_h}(x-X_i) - \mathcal{I}_{K_{h_{min}}}(x-X_i))^2\right. \tag{2.44}$$

$$\left. -\frac{1}{n(n-1)}(\sum_{i=1}^{n}\mathcal{I}_{K_h}(x-X_i) - \mathcal{I}_{K_{h_{min}}}(x-X_i))^2\right) dx \tag{2.45}$$

Definition 2.7 (PCO inspired bandwidth selection)**.** *Let $\hat{p}$ be the p-value estimator based on KDE. Let $\widehat{Va}(h)$ and $\hat{\Delta}b(h)$ be the estimators of the variance and bias correction terms. The PCO-based MISE estimator $\widehat{Crit}_{pco}$ is computed as follows:*

$$\widehat{Crit}_{pco}(h) = ||\hat{p}_h - \hat{p}_{h_{min}}||_s^2 - \hat{\Delta}b(h) + \widehat{Va}(h) \tag{2.46}$$

The PCO-based bandwidth $\hat{h}_{PCO}$ is defined as:

$$\hat{h}_{PCO}(h) = \arg\min_h \widehat{Crit}_{pco}(h) \tag{2.47}$$

The following proposition shows that this procedure is equivalent to the LOO procedure described in Section 2.4. This means that both procedures always select the same values for the bandwidth parameter.

Proposition 2.4 (PCO procedure is equivalent as the LOO procedure)**.** *Under the assumption that $\mathcal{I}_{K_{h_{min}}}(x)$ is equal to the step function $\mathbb{1}[x < 0]$, then the LOO procedure and the PCO procedures select the same bandwidth:*

$$\hat{h}_{PCO}(h) = \hat{h}_{LOO}(h) \tag{2.48}$$

The proof is delayed to Section 2.10.3. The assumption $\mathcal{I}_{K_{h_{min}}}(x) \approx \mathbb{1}[x < 0]$ is often verified when h_{min} is small. Therefore, the PCO procedure is not considered in the empirical experiments in this chapter to avoid duplicating the results of the LOO procedure.

2.6 Empirical results

In the previous Section 2.4, a new method for selecting the bandwidth h has been developed based on LOO. In this section, this procedure is compared with existing p-value estimation methods in terms of their ability to estimate correctly the p-value.

2.6.1 Precision of the p-value estimator

The accuracy of different estimators of the p-value is assessed by measuring the mean integral squared error $MISE = \mathbb{E}\int_s(\hat{p}(x)-p(x))^2dx$ for different estimators and for different distribution laws. In particular, different KDE-based estimators are compared by varying procedure for selecting the bandwidth value h.

2.6.1.1 Description of the experiment

The following notations are used: $\mathcal{P}_0$ is the reference distribution, n is the cardinality of the calibration set, $\hat{h}$ is the procedure selection for the bandwidth parameter and $\hat{p}$ the p-value estimator. Furthermore, the integration scheme applied to the tail is described by the following parameters: s the lower limits of the tail, x_{max} the upper limit of the tail used to approximate the integration to infinity and δ_x the step size of the integration approximation. These parameters are chosen according to the heuristic described in Section 2.4.4. The experiment is described as follows:

1. Generate the calibration set of cardinality n.

$$\forall i \in [\![1, n]\!], \qquad X_i \sim \mathcal{P}_0$$

2. Select the bandwidth parameter $h = \hat{h}(X_1^n)$

3. Estimate the p-value at the tail of the distribution $[s, x_{max}[$, with an integration step equal to δ_x:

$$\forall i \in [\![1, (x_{max} - s)/\delta_x]\!], \hat{p}_i = \hat{p}(s + i\delta_x, X_1^n, h) \tag{2.49}$$

4. Compute the squared integrated error, comparing the true p-value p with the estimation $\hat{p}$:

$$ISE = \sum_{i=1}^{(x_{max}-s)/\delta_x} \delta_x(\hat{p}_i - p(s + i\delta_x))^2 \tag{2.50}$$

The same experiment is repeated 100 times to get an idea of the data distribution. To test the agnosticity the bandwidth selection method, several reference laws $\mathcal{P}_0$ are considered: $\mathcal{N}(0, 1)$, $t(df = 5)$, $Exp(1)$, and $U(0, 1)$. To evaluate the efficiency of the p-value estimator, the cardinality of the calibration set varies with values $n \in \{20, 100, 500\}$. The different smoothing selection methods are tested: the LOO estimator introduced in Section 2.4 and the classical LOO estimator of the density error presented in Section 2.2.2.1. To have a better understanding of the results, an oracle version of each LOO estimator is considered, in which the true value of the ISE criterion is minimized. To go further in the analysis, other p-value estimators are also tested: the Gaussian estimator, the empirical estimator and the Peak over the Threshold (POT)[149] estimator.

2.6.1.2 Results and analysis

The results are shown in Figures 2.6-2.9. In each figure, the integrated p-value estimation error, ISE_p, is plotted from (a) to (c) for different p-value estimators and for different calibration set cardinalities. The different estimators shown are:

- KDE-p-LOO, the KDE estimator where the bandwidth was chosen by minimizing the LOO estimator of ISE_p,
- KDE-p-O, the KDE estimator where the bandwidth was chosen by minimizing the true error ISE_p,

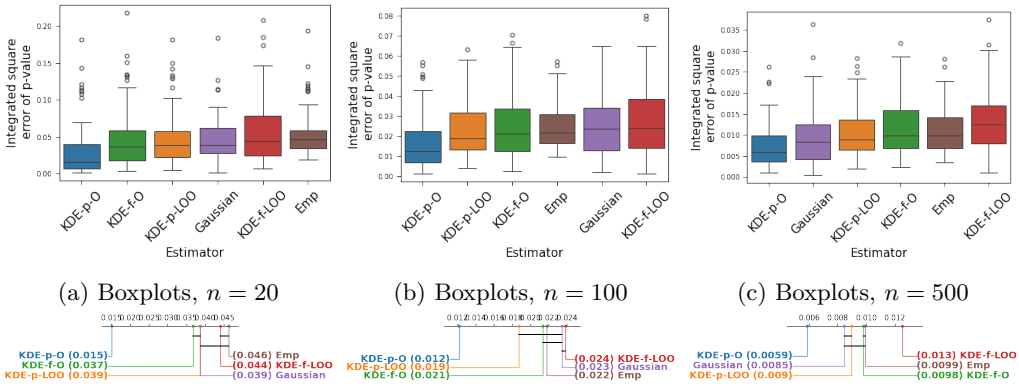
- KDE-f-LOO, the KDE estimator where the bandwidth was chosen by minimizing the LOO estimator of ISE_f,
- KDE-f-O, the KDE estimator where the bandwidth was chosen by minimizing the true value of ISE_f,
- *POT*: peak-over the threshold estimator,
- *Emp* the empirical p-value estimator and
- *Gaussian* the Gaussian estimator.

To distinguish statistically the performances of the different estimators, Critical Difference (CD) diagrams [47] are shown from (d) to (f). Each estimator of the p-value is placed on the horizontal line as a function of the $MISE_p$. Estimators whose $MISE_p$ are not significantly different are connected with horizontal lines. Therefore, paired permutation tests [54] are used to compare the performance of two estimators. For each pair of estimators, the hypothesis tested is: "The $MISE_p$ is the same using these two p-value estimators". The Bonferroni correction is applied to account for the large number of tests performed. The significance level is set to 0.05.

The Gaussian estimator performs well on Gaussian data, as shown in Figure 2.6 and performs poorly on other types of data, as illustrated in Figure 2.8. More surprisingly, the Gaussian estimator is not the best estimator on Gaussian data, as shown in Figure 2.6b. The POT and KDE-p-O estimators have a lower p-value estimation error. This is probably due to the fact that they are more tail-specific estimators. As shown in Figure 2.6c and 2.8c, the POT estimator obtains the minimum estimation error compared to the other estimators, for Gaussian and exponential data. On the contrary, the error is large for Student of degree 2, as shown in Figure 2.7c. Note that in extreme value theory, the distribution tails of the Gaussian and exponential distributions are modeled by Gumbel distributions [130], for which the shape parameter of the generalized Pareto distribution is zero. On the other hand, the Student distribution is Fréchet [130], with a positive shape parameter. The POT estimator seems to be advantageous mainly for Gumbel-type laws. The KDE-p-O estimator performs well on all data types. In particular, its performance is significantly better compared to KDE-f-O. This shows the theoretical interest of minimizing ISE_p instead of ISE_f. However, KDE-p-LOO does not significantly outperform KDE-f-LOO and KDE-f-O, as shown in Figure 2.8c. The poor performance of the LOO estimator prevents the gains of the $MISE_p$ minimization strategy from being exploited. The empirical p-value estimator performs as good as kernel estimators (except KDE-f-O) when the cardinality of the calibration set is equal to 500, as shown in Figure 2.8c. In Figure 2.8b, the kernel methods outperform Emp when the number of points is smaller.

The results show that the kernel estimators have the best performance for estimating p-values in the general cases while the bandwidth selection method has a little impact on the performance. In the case of a Gumbel-type law (light tail), POT estimator improves the performance. In the situation with a sufficient number of calibration points, the empirical p-value is able to achieve similar results compared to KDE.

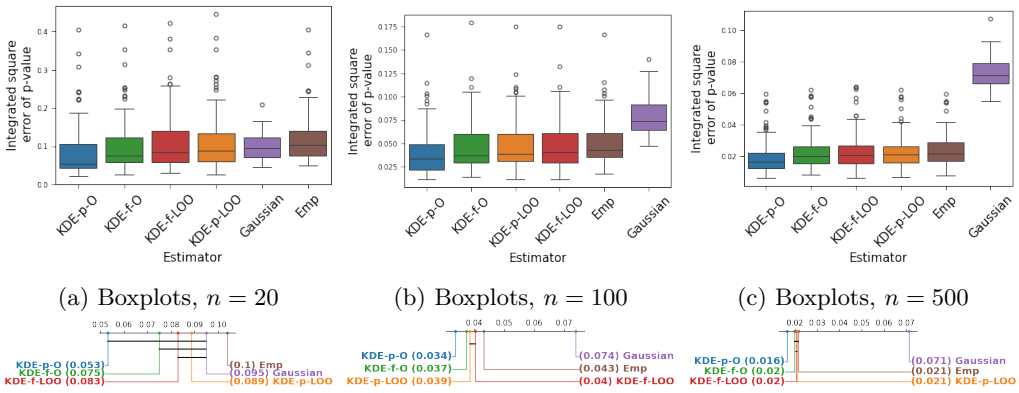
The selection strategy developed in Section 2.4 does not improve the performance of the p-value estimator because the LOO estimator of the ISE error is not precise enough. Moreover, as discussed in Section 2.4.5, this estimator has a high computational cost. Thus the strategy is of no practical interest unless the LOO estimator is replaced by a better one. This analysis is aligned with the study [166] which concludes that different bandwidth selection methods have the same performance.



(a) Boxplots, $n = 20$ (b) Boxplots, $n = 100$ (c) Boxplots, $n = 500$

(d) CD diagram, $n = 20$ (e) CD diagram, $n = 100$ (f) CD diagram, $n = 500$

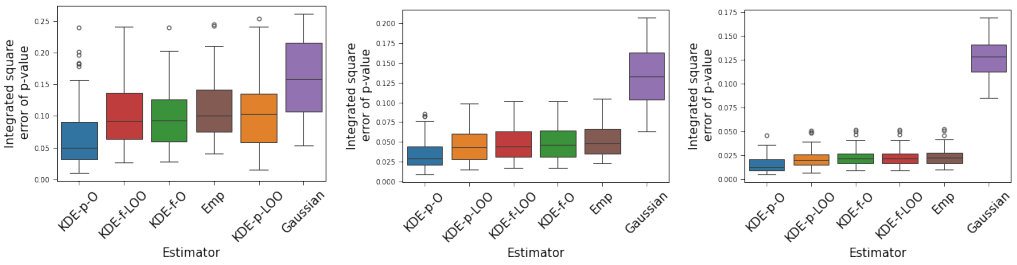
Figure 2.6: Integrated p-values estimation error according to the p-values estimator for $\mathcal{N}(0, 1)$ data, for different calibration set cardinalities (n).



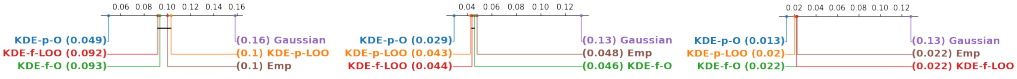
(a) Boxplots, $n = 20$ (b) Boxplots, $n = 100$ (c) Boxplots, $n = 500$

(d) CD diagram, $n = 20$ (e) CD diagram, $n = 100$ (f) CD diagram, $n = 500$

Figure 2.7: Integrated p-values estimation error according to the p-values estimator for $t(2)$ data, for different calibration set cardinalities (n).

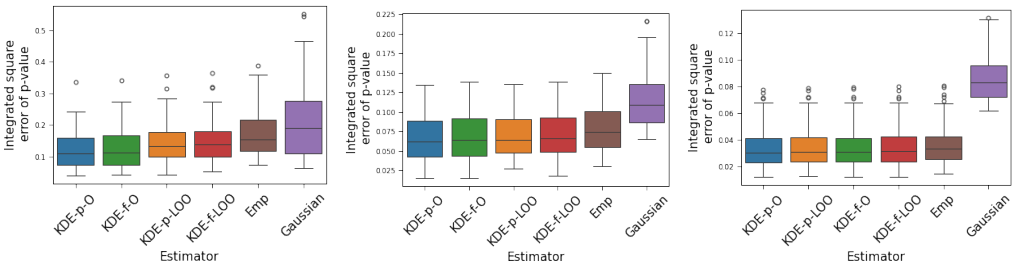


(a) Boxplots, $n = 20$ (b) Boxplots, $n = 100$ (c) Boxplots, $n = 500$

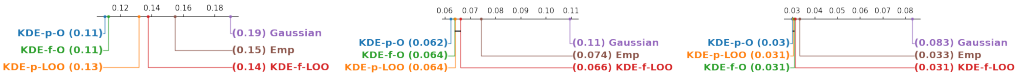


(d) CD diagram, $n = 20$ (e) CD diagram, $n = 100$ (f) CD diagram, $n = 500$

Figure 2.8: Integrated p-values estimation error according to the p-values estimator for $Exp(1)$ data, for different calibration set cardinalities (n).



(a) Boxplots, $n = 20$ (b) Boxplots, $n = 100$ (c) Boxplots, $n = 500$



(d) CD diagram, $n = 20$ (e) CD diagram, $n = 100$ (f) CD diagram, $n = 500$

Figure 2.9: Integrated p-values estimation error according to the p-values estimator for $U(0, 5)$ data, for different calibration set cardinalities (n).

2.6.2 Robustness of the estimator

This section evaluates the robustness of the p-value estimators. This is done by observing the effect on the value of the estimation error ISE by adding anomalies to the calibration set. In particular, the focus is on the MoM-KDE estimator and the relationship between the number of blocks and the performance of the estimator.

2.6.2.1 Description of the experiment

The same experimental procedure is used as in Section 2.6.1.1, but anomalies are added to the calibration set. Calibration set data are generated as follows, by noting n_1 the number of anomalies, let $\mathcal{A}$ be the anomaly subset of $[\![1, n]\!]$ of cardinality n_1 obtained by a random draw without replacement and Δ the value of abnormal data points. Using a single value for anomalies makes it possible to control in a deterministic way the level of atypicity of anomalies in a sample that contains few of them:

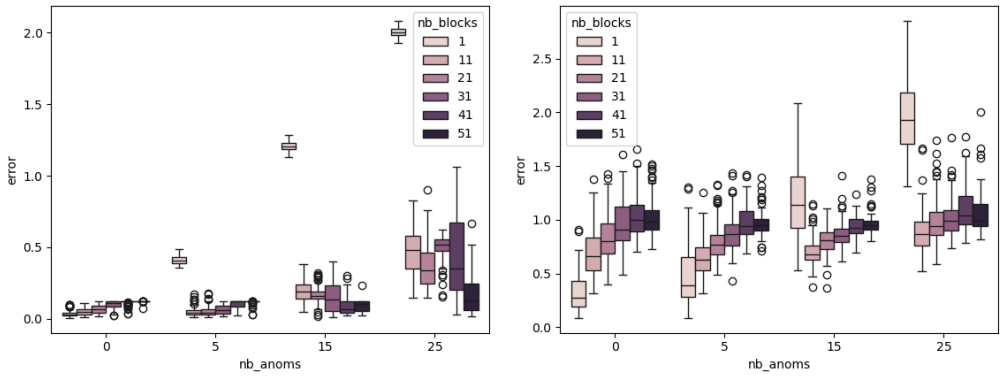
$$\begin{aligned} \forall i \in \mathcal{A}, \quad & X_i = \Delta \\ \forall i \in [\![1, n]\!] \backslash \mathcal{A}, \quad & X_i \sim \mathcal{P}_0 \end{aligned}$$

The same experiment is repeated 100 times to get an idea of the data distribution. In order to evaluate the impact of the anomalies on the estimation error of the p-value, n_1 takes values over $\{0, 5, 15, 25\}$. The number of blocks tested are $\{1, 11, 21, 31, 41, 51\}$. The parameter Δ is chosen equal to 4. These numbers are chosen so that at least one of the estimators has more than twice as many blocks as there are anomalies in the calibration set. The KDE bandwidth is selected using the LOO estimator introduced in Section 2.4. The bandwidth of MoM-KDE uses a different LOO estimator, introduced in Definition 2.6, which takes into account the different blocks. The LOO uses the same blocks as MoM-KDE. To test the agnosticity of the bandwidth selection method, several reference laws $\mathcal{P}_0$ are considered: $\mathcal{N}(0, 1)$, $t(df = 5)$. The parameter s is set to 2.

2.6.2.2 Results and analysis

The results are shown as boxplots in Figure 2.10. The x-axis represents the number of anomalies in the calibration set. The y-axis represents the integrated error of the p-value and the color is graduated according to the number of blocks used by MoM-KDE to partition the calibration set. When the number of blocks is 1, MoM-KDE reverts to the classic KDE. Figure 2.10a shows the results for Gaussian data. In the case of no anomalies, the estimation error increases with the number of blocks. However, this degradation is small compared to the one generated by the presence of anomalies in the classical KDE. The estimation error increases rapidly with the number of anomalies. For a constant number of anomalies, the estimation error decreases with the number of blocks. When the number of blocks is more than twice the number of anomalies, the estimation error of the p-value is close to that obtained without anomalies. For example, in the presence of 15 anomalies, estimators using 1, 11 or 21 blocks see their performance degraded. On the other hand, estimators using 31 or 41 blocks are not affected. This is consistent with the discussion given in Section 2.1 after Definition 2.2. Furthermore, even when the number of blocks is small and anomalies affect MoM-KDE's estimation, the impact is much smaller than on the classic KDE. This is probably due to the fact that the number of anomalies in each block remains small. For KDE, on the other hand, the single block contains all the anomalies.

For Student data the anomalies seam to have less impact on the performance of the estimator, as shown in Figure 2.10b. This could be due to the fact that the performance of the p-value estimators is worse on Student data than on Gaussian data. Indeed, even in the absence of anomalies, the error of the KDE is around 0.3, compared to 0.05 in the Gaussian case. At least 15 anomalies must be added to observe the deterioration of the classic KDE. MoM-KDEs are not affected once the number of blocks reaches 11. Therefore, although MoM-KDE performs worse than KDE when the number of anomalies is low, it becomes advantageous when the number of anomalies is higher. The difficulty in practice is that the number of anomalies is unknown.



(a) Gaussian data (b) Student data

Figure 2.10: Integrated p-values estimation error using the KDE p-value estimator as a function of the number of block (nb_blocks) and the number of anomalies (nb_anoms).

2.7 FDR control using KDE p-value estimator

The approach taken in this chapter has been to minimize the p-value estimation error by optimizing the kernel bandwidth. To this end, a new selection strategy based on a LOO estimator has been developed in Section 2.4. It was also with this objective in mind that the different p-value estimators were compared in Section 2.6.1.1.In the general case, KDE estimates the p-value with a lower estimation error compared to other competitors that have been tested. Moreover, the use of the Median-of-Means principle proved their effectiveness in reducing the impact of anomalies on the p-value estimation error. However, no significant improvement was found in the proposed bandwidth selection approach compared to more conventional ones.

Nevertheless, estimating the p-value is only one step in anomaly detection. When a data-driven threshold is used, these p-values are used to select the threshold. The final step is to compare the p-value to this threshold. It is difficult to establish how an estimation error affects the threshold and anomaly detection. In particular, it's not clear what the p-value estimation error should be to guarantee a given FDR [13] control or FNR level. Also, it is not known if the criterion studied in this chapter is the best one for this purpose. In this section different bandwidth selection procedures are experimentally compared in terms of their ability to control the FDR [13] with a low FNR. The goal is to see if a relationship can be established between bandwidth selection, FDR control and the various problem parameters: calibration set cardinality n, test set cardinality m and the desired FDR level α. In this experiment, the aim is to evaluate the ability of KDE to control the FDR in the simplest case, so no anomalies are added to the calibration

set and MoM-KDE is not tested.

2.7.1 Experiment description

The following notations are used: $\mathcal{P}_0$ is the reference distribution, n is the cardinality of the calibration set, m is the cardinality of the test set, m_1 is the number of anomalies in the test set, Δ is the shift value of the anomalies, $\hat{h}$ is the procedure selection for the bandwidth parameter and α is the desired FDR level. The experiment procedure is described as follows:

1. Generate the calibration set of cardinality n.

$$\forall i \in [\![1, n]\!], \qquad X_i \sim \mathcal{P}_0$$

2. Generate the test set of cardinality m with m_1 anomalies.

$$\begin{aligned} \forall j \in [\![1, m_1]\!], &\qquad Y_j = \Delta \\ \forall j \in [\![m_1, m]\!], &\qquad Y_j \sim \mathcal{P}_0 \end{aligned}$$

3. Select the bandwidth parameter $h = \hat{h}(X_1^n)$

4. Estimate the p-value in the test set, using the kernel estimator with bandwidth h.

$$\forall j \in [\![1, m]\!], \qquad p_j = \hat{p}_h(Y_j, X_1^n, h)$$

5. Apply the Benjamini-Hochberg procedure to select the threshold [13].

$$\hat{\varepsilon} = \hat{\varepsilon}_{BH}(\hat{p}_1, \ldots, \hat{p}_m)$$

6. Compute the FDP and FNP criteria. Note that anomalies are generated in the first m_1 observations.

$$\begin{aligned} FDP &= \frac{\sum_{j=m_1}^{m} \mathbb{1}[\hat{p}_j \leq \hat{\varepsilon}]}{\sum_{j=1}^{m} \mathbb{1}[\hat{p}_j \leq \hat{\varepsilon}]} \\ FNP &= \frac{\sum_{j=1}^{m_1} \mathbb{1}[\hat{p}_j > \hat{\varepsilon}]}{m_1} \end{aligned}$$

This procedure is repeated 1000 times for each scenario with: $n \in \{50j, j \in [\![1, 20]\!]\}$ and the following selection procedures:

- "minimise density-error on support": Select the bandwidth parameter that minimizes the density estimation error on the entire support. This is the classical LOO criterion.
- "minimise sf-error on tail": Select the bandwidth parameter that minimizes the p-value estimation error at the tail of the distribution. This is the LOO criterion introduced in Section 2.4.
- "minimise density-error on tail": Select the bandwidth parameter that minimizes the density estimation error at the tail of the distribution.

- "minimise sf-error on support": Select the bandwidth parameter that minimizes the p-value estimation error on the entire support.

The classical procedure and the one introduced in Section 2.4 differ in two criteria: the error, which is related to the p-value instead of the density, and the integration support, which is limited to the tail of the distribution. To understand how each modification affects the performance, two other procedures are used: "minimise density-error on tail" and "minimise sf-error on support".

The parameter m takes value in $\{52, 100\}$, m_1 takes value 2 when $m = 52$ and 1 when $m = 100$, the value of anomalies is $\Delta = 4$, and α takes values in $\{0.02, 0.05, 0.1\}$. The reference distributions tested are $\mathcal{N}(0, 1)$ and $t(5)$.

2.7.2 Results and analysis

The results are shown in Figures 2.11-2.13 as a set of charts with the FDR or FNR in the y-axis and the cardinality of the calibration set n in the x-axis. The different color lines correspond to the different bandwidth procedures. The subfigures (a), (c) and (e) present the FDR and the subfigures (b), (d) and (f) present the FNR. Furthermore, the subfigures (a) and (b) show the results corresponding to the case where $\alpha = 0.02$, the subfigures (c) and (d) show the results corresponding to the case where $\alpha = 0.05$ and the subfigures (e) and (f) show the results corresponding to the case where $\alpha = 0.1$. Finally, the different figures correspond to different scenarios: Figure 2.11 where the calibration set is $m = 52$ and the reference distribution is Gaussian. Figure 2.12 where the calibration set is $m = 100$. Figure 2.13 where the calibration set is $m = 52$ and the reference distribution is Student.

The FDR decreases with the cardinality of the calibration set; if the calibration set contains enough data points, the FDR converges to the desired FDR. For example, in Figure 2.11a, starting with $n = 800$ data points, the FDR is close to the desired FDR $\alpha = 0.02$. The FNR appears to decrease as the cardinality of the calibration set increases. However, some unexpected behavior can be noticed, for example in Figure 14.a. the FNR increases until $n = 1200$, where it reaches $FNR = 0.04$, before dropping abruptly to $FNR = 0$.

When the desired FDR α is large, the FDR decreases more rapidly to α. For example, in Figure 2.12c, where $\alpha = 0.02$, the desired FDR is reached around 1600, while in Figure 2.12e, where $\alpha = 0.1$, it is reached at $n = 300$. More points in the calibration set are needed to control the FDR at a lower level. A comparison of Figure 2.11 and Figure 2.12 shows the effect of the test set cardinality on the FDR and FNR.

In Figure 2.11a, where the test set cardinality is equal to 52, $n = 800$ points in the calibration set are required to control the FDR. Also, the FNR is 0.03 before $n = 1250$. After that, the FNR is 0. In Figure 2.12a, where the test set cardinality is 100, the calibration set needs to contain 1600 points to enable FDR control. Also, the FNR is about 0.1 for all values of n. Thus, FDR and FNR control becomes more difficult as the test cardinality increases.

The reference distribution generated by the calibration set affects the FDR control. For example, the FDR control stops at $n = 250$ in Figure 2.11c, where the reference distribution is Gaussian, whereas 800 points are needed in Figure 2.13c, where the reference distribution is Student. Although the KDE estimator is nonparametric, the performance of the p value estimation or bandwidth depends on the reference distribution. The performance of the various bandwidth selection procedures is similar and difficult to distinguish from data noise. The performance of the various bandwidth selection procedures is similar and difficult to distinguish from the noise in the data. However, it seems that the FDR associated with the "minimise

sf-error on support" method has a lower FDR than the other methods. As shown for example in Figure 2.12c, the consequence is that FDR control can be achieved for smaller n values, but at the same time the FNR is also higher, as shown in Figure 2.12d.

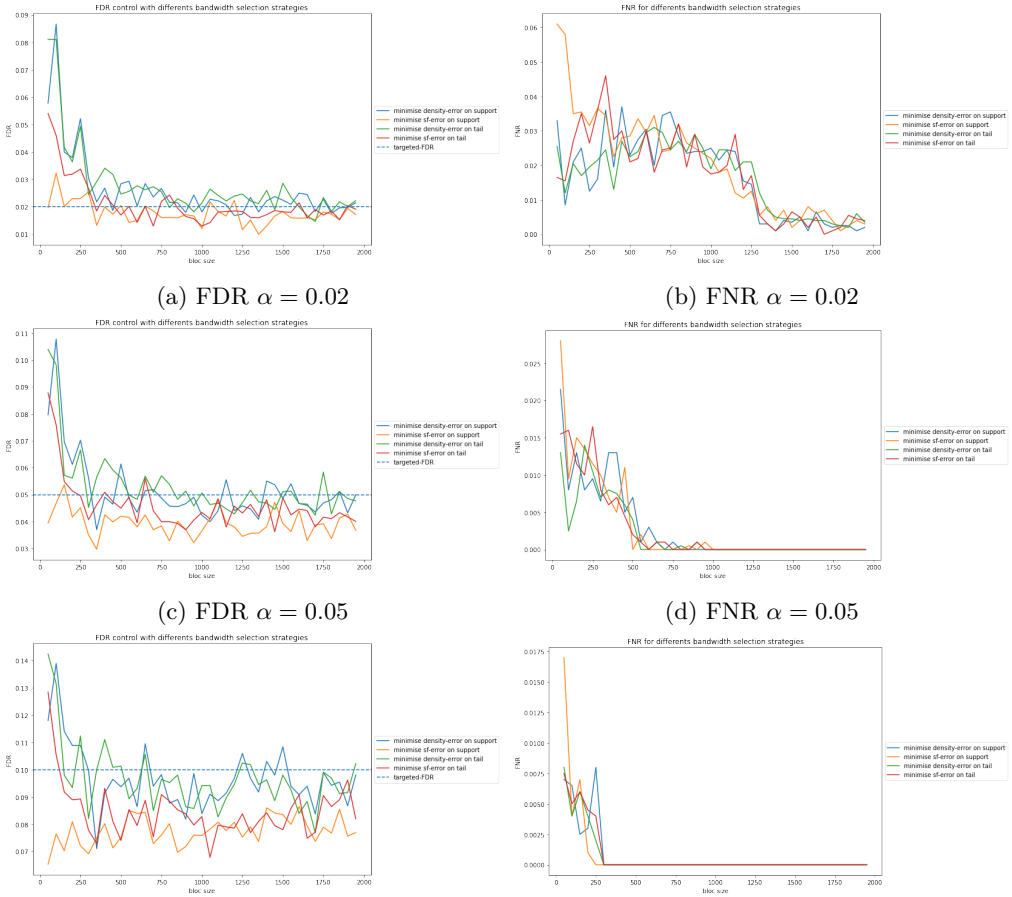
2.7.3 Conclusion of the experiment

Once again, there is little difference in performance between the different bandwidth selection methods. FDR control requires more calibration points if the test set cardinality m is large or if the control level α is small. Beyond qualitative judgments, it's difficult to provide guarantees on the control that KDE allows over the FDR. For these reasons, besides the high computational cost of finding the bandwidth, we prefer to use the empirical p-value estimator, which is easier to study.

2.8 Conclusion

The goal of this chapter was to build a p-value estimator which is robust and efficient. To achieve these objectives, the idea was to develop a new procedure for selecting the bandwidth for KDE that minimizes the prediction error of the p-value at the tail of the distribution. For this purpose, a new leave-one-out estimator was developed. However, empirical studies show that this new procedure does not lead to an improvement of the p-value estimation in practice due to the high variance of the estimator. Furthermore, KDE-based p-value estimators do not provide theoretical guarantees about the anomaly detector measured by FDR and FNR. For these reasons, the empirical p-value estimator is preferred, even though it is less efficient and robust, because it is easier to study theoretically. In the next chapter, the control of the FDR with the empirical p-value is studied.

2.9 Supplement figures for Section 2.7

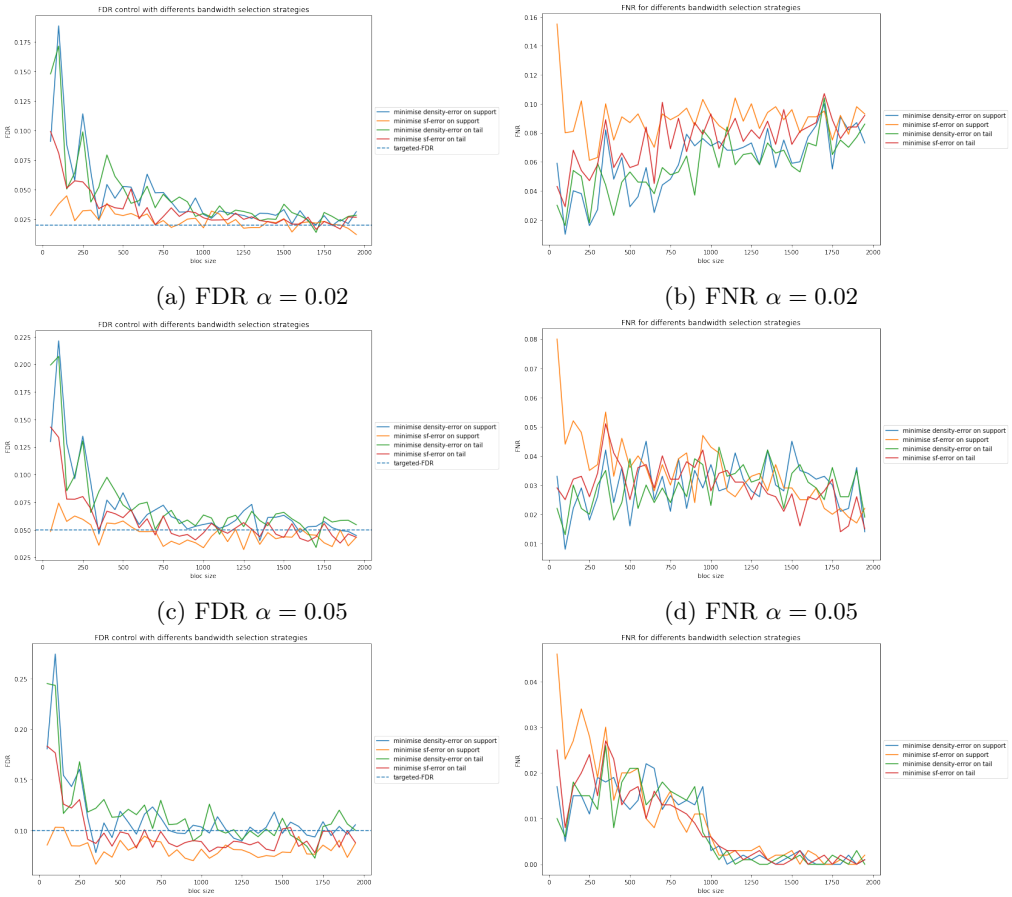


(a) FDR $\alpha = 0.02$ (b) FNR $\alpha = 0.02$

(c) FDR $\alpha = 0.05$ (d) FNR $\alpha = 0.05$

(e) FDR $\alpha = 0.1$ (f) FNR $\alpha = 0.1$

Figure 2.11: FDR and FNR on Gaussian data with $m = 52$.

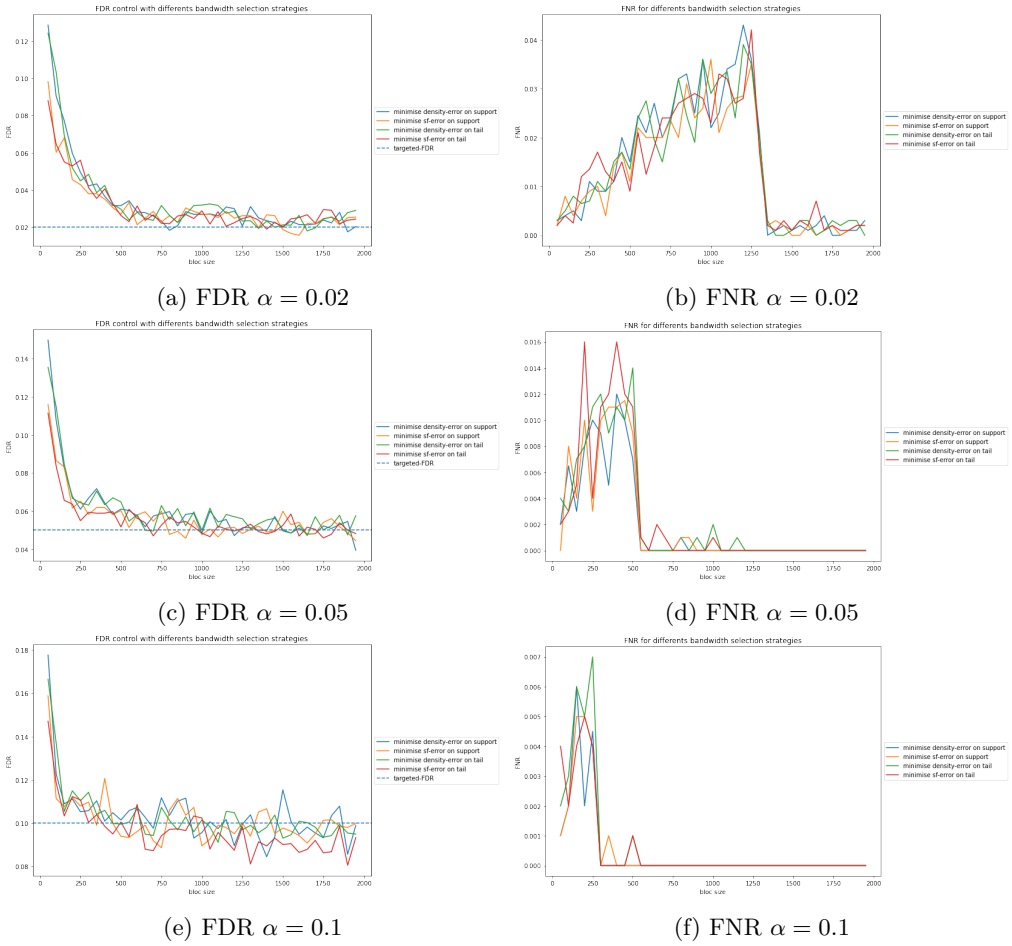


(a) FDR $\alpha = 0.02$ (b) FNR $\alpha = 0.02$

(c) FDR $\alpha = 0.05$ (d) FNR $\alpha = 0.05$

(e) FDR $\alpha = 0.1$ (f) FNR $\alpha = 0.1$

Figure 2.12: FDR and FNR on Gaussian data with $m = 100$.



(a) FDR $\alpha = 0.02$ (b) FNR $\alpha = 0.02$

(c) FDR $\alpha = 0.05$ (d) FNR $\alpha = 0.05$

(e) FDR $\alpha = 0.1$ (f) FNR $\alpha = 0.1$

Figure 2.13: FDR and FNR on student data with $m = 52$.

2.10 Proofs

2.10.1 Proof of Proposition 2.1

Proof of asymptotic bias in Section 2.1. By definition the bias is equal to:

$$B(h) = ||p_h - p||_s^2 \tag{2.51}$$

$$= \int_s^{+\infty} (\mathbb{E}(\hat{p}_h(x) - p(x))^2 dx \tag{2.52}$$

Let's have a look at the asymptotic behavior of the term within the integral, which allows the final result to be obtained by integration. Let x_0 be a real belonging to $[s, +\infty[$, let f be the density associated with the reference distribution $\mathcal{P}_0$.

$$\mathbb{E}(\hat{p}_h(x_0) - p(x_0) = \int \hat{p}_h(x_0) d\mathbb{P}(X_1^n) - p(x_0) \tag{2.53}$$

$$\mathbb{E}(\hat{p}_h(x_0) - p(x_0) = \int \int_{x_0}^{+\infty} \frac{1}{hn} \sum_{i=1}^{n} K\left(\frac{z - X_i}{h}\right) dz d\mathbb{P}(X_1^n) - \int_{x_0}^{+\infty} f(z) dz \tag{2.54}$$

$$\mathbb{E}(\hat{p}_h(x_0) - p(x_0) = \int_{-\infty}^{+\infty} \int_{x_0}^{+\infty} \frac{1}{h} K\left(\frac{z - X}{h}\right) f(X) dz dX - \int_{x_0}^{+\infty} f(z) dz \tag{2.55}$$

Since $\int_{x_0}^{+\infty} K\left(\frac{z-X}{h}\right) dz < z$ and then $\int_{-\infty}^{+\infty} \int_{x_0}^{+\infty} K\left(\frac{z-X}{h}\right) f(X) dz dX < 1$, the Fubini's theorem allows to switch the order of integration:

$$\mathbb{E}(\hat{p}_h(x_0) - p(x_0) = \int_{x_0}^{+\infty} \frac{1}{h} \int_{-\infty}^{+\infty} K\left(\frac{z - X}{h}\right) f(X) dX dz - \int_{x_0}^{+\infty} f(z) dz \tag{2.56}$$

The following substitution is used: $X \to z - hy$. Then $dX \to -h dy$ and:

$$\mathbb{E}(\hat{p}_h(x_0) - p(x_0) = \int_{x_0}^{+\infty} (\int_{-\infty}^{+\infty} K(y) f(z - hy) dy - f(z)) dz \tag{2.57}$$

The series expansion at second order gives:

$$\mathbb{E}(\hat{p}_h(x_0) - p(x_0) = \int_{x_0}^{+\infty} (\int_{-\infty}^{+\infty} K(y)(f(z) + hyf'(z) + 0.5h^2y^2f''(z) + o(h^2)) dy - f(z)) dz \tag{2.58}$$

Since $\int K(y) = 1$, the terms $f(z)$ cancel each other and by symmetry of $K(y)$ $\int K(y) hyf'(z) dy = 0$. Then it gives

$$\mathbb{E}(\hat{p}_h(x_0) - p(x_0) = \int_{x_0}^{+\infty} 0.5h^2 f''(z) \int_{-\infty}^{+\infty} y^2 K(y) + o(h^2) dz \tag{2.59}$$

$$= 0.5h^2 \int y^2 K(y) dy \int_{x_0}^{+\infty} f''(z) dz + o(h^2) \tag{2.60}$$

The bias is finally obtained by integrating the square of this value:

$$B(h) = \int_s^{+\infty} (\mathbb{E}(\hat{p}_h(x) - p(x))^2 dx \tag{2.61}$$

$$= \int_s (1/2h^2 \int_{-\infty}^{+\infty} y^2 K(y) dy \int_x^{+\infty} f''(z) dz + o(h^2))^2 dx \tag{2.62}$$

$$= 0.25h^4 \left(\int_{-\infty}^{+\infty} y^2 K(y) dy \right)^2 \int_s^{+\infty} \left(\int_x^{+\infty} f''(z) dz \right)^2 dx + o(h^4) \tag{2.63}$$

□

2.10.2 Proof of Proposition 2.3

Proof of Proposition 2.3. As stated in Definition 2.5, the LOO estimator $\hat{D}$ is expressed as:

$$\hat{D}(h) = \frac{1}{(n+1)n} \sum_{i=1}^{n+1} \sum_{j=1; j\neq i}^{n+1} \mathbb{1}[X_i > s] \underbrace{\int_s^{X_i} \int_x^{+\infty} K(\frac{z - X_j}{h}) dz dx}_{D_{i,j}(h)} \tag{2.64}$$

In order to invert the two integrals, the integration limits are replaced by conditions in the indicator function $\mathbb{1}$.

$$D_{i,j}(h) = \int_s^{+\infty} \int_s^{+\infty} \mathbb{1}[x \leq X_i] \mathbb{1}[z \geq x] K(\frac{z - X_j}{h}) dz dx$$

After inverting the two integrals, the indicator function can be isolated from the kernel.

$$D_{i,j}(h) = \int_s^{+\infty} K(\frac{z - X_j}{h}) \left(\int_s^{+\infty} \mathbb{1}[x \leq X_i] \mathbb{1}[z \geq x] dx \right) dz$$

Using, that $x \leq X_i$ and $x \leq z$ implies $x \leq \min(X_i, z)$ it gives:

$$\int_s^{+\infty} \mathbb{1}[x \leq X_i] \mathbb{1}[z \geq x] dx = \max(\min(X_i, z) - s, 0)$$

This result will be injected into the expression of $D(h)$ given by Eq. 2.64, in this expression it is shown that: $\min(z - s, X_i - s) \geq 0$. Then the integral is split in two, first $z < X_i$ then $z > X_i$.

$$D_{i,j}(h) = \int_s^{+\infty} K(\frac{z - X_j}{h}) (\min(X_i, z) - s) dz \tag{2.65}$$

$$= \underbrace{\int_s^{X_i} K(\frac{z - X_j}{h}) (z - s) dz}_{I_1} + \underbrace{\int_{X_i}^{\infty} K(\frac{z - X_j}{h}) (X_i - s) dz}_{I_2} \tag{2.66}$$

In the following, each integral noted I_1 and I_2 is calculated in the case of the Gaussian kernel.

It begins with a substitution: $(z - X_i \to z)$

$$I_1 = \int_{s-X_j}^{X_i-X_j} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2h^2}\right)(z - s + X_j)dz \tag{2.67}$$

$$= \int_{s-X_j}^{X_i-X_j} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2h^2}\right) z dz + \frac{X_j - s}{\sqrt{2\pi}} \int_{s-X_j}^{X_i-X_j} \exp\left(-\frac{z^2}{2h^2}\right) dz \tag{2.68}$$

The first term of Eq. 2.68, can be integrated by recognize $\int u' \exp u = \exp u$, when $u = \exp(-\frac{z^2}{2h^2})$.

For the second term, with substitution $z/\sqrt{2}h \to z$ it gives:

$$\int_{s-X_j}^{X_i-X_j} \exp\left(\frac{-z^2}{2h^2}\right) dz = \int_{\frac{s-X_j}{\sqrt{2}h}}^{\frac{X_i-X_j}{\sqrt{2}h}} \exp\left(-z^2\right) dz \tag{2.69}$$

Then using the error function, noted erf and defined as $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp -z^2 dz$, it gives:

$$\int_{s-X_j}^{X_i-X_j} \exp\left(\frac{-z^2}{2h^2}\right) dz = (X_j - s)\frac{\sqrt{\pi}}{2} h\sqrt{2}[\text{erf}(\frac{X_i - X_j}{h\sqrt{2}}) - \text{erf}(\frac{s - X_j}{h\sqrt{2}})] \tag{2.70}$$

Thus, the integral I_1 can be expressed as:

$$I_1 = h^2 \left[\exp(-\frac{-t^2}{2h^2})\right]_{X_i-X_j}^{s-X_j} - (X_j - s)\frac{\sqrt{\pi}}{2} h\sqrt{2}[\text{erf}(\frac{X_i - X_j}{h\sqrt{2}}) - \text{erf}(\frac{s - X_j}{h\sqrt{2}})] \tag{2.71}$$

The integral I_2 is calculated using the same variable substitution $((z - X_i)/(h\sqrt{2}) \to z)$:

$$I_2 = (X_i - s)h\sqrt{2}\frac{\sqrt{\pi}2}{2\sqrt{\pi}} \int_{\frac{X_i-X_j}{h\sqrt{2}}}^{+\infty} \exp\left(-z^2\right) dz \tag{2.72}$$

$$= (X_i - s)h\frac{\sqrt{2}}{\sqrt{\pi}}\left(1 - \text{erf}\left(\frac{X_i - X_j}{h\sqrt{2}}\right)\right) \tag{2.73}$$

By combining Eq. 2.71 and Eq. 2.73, the value of $D_{i,j}(h)$ is calculated as follows:

$$D_{i,j}(h) = h^2 \exp\left(-\frac{(s - X_j)^2}{2h^2}\right) - h^2 \exp\left(-\frac{(X_i - X_j)^2}{2h^2}\right)$$
$$+ (X_j - s)\frac{\sqrt{\pi}}{2} h\sqrt{2}[\text{erf}(\frac{X_i - X_j}{h\sqrt{2}}) - \text{erf}(\frac{s - X_j}{h\sqrt{2}})] - (X_i - s)h\frac{\sqrt{\pi}}{\sqrt{2}}\left(1 - \text{erf}\left(\frac{X_i - X_j}{h\sqrt{2}}\right)\right)$$

Finally, the value of $D(h)$ is obtained by summing all the $D_{i,j}$

□

2.10.3 Proof of Proposition 2.4

Proof of Proposition 2.4. First, the PCO criterion is simplified:

- $||\hat{p}_h - \hat{p}_{h_{min}}||_s^2$, decomposed as quadratics and multiplicative terms:

$$||\hat{p}_h - \hat{p}_{h_{min}}||_s^2 = ||\hat{p}_h||_s^2 - 2\langle \hat{p}_h, \hat{p}_{h_{min}} \rangle + ||\hat{p}_{h_{min}}||_s^2 \tag{2.74}$$
$$\tag{2.75}$$

Since the last term is a constant and $\hat{p}_h(x) = \frac{1}{n}\sum_{i=1}^n \mathcal{I}_{K_h}(x - X)$, then:

$$||\hat{p}_h - \hat{p}_{h_{min}}||_s^2 = \frac{1}{n^2}\sum_{i,j}\int_s^{+\infty} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_h}(x - X_j)dx \tag{2.76}$$
$$- \frac{2}{n^2}\sum_{i,j}\int_s^{+\infty} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_j)dx + c \tag{2.77}$$

- $-\hat{\Delta}b(h) + \hat{v}(h)$: the following notations are introduced:

1. $v_1(x) = \frac{1}{n-1}\sum_{i=1}^n (\mathcal{I}_{K_h}(x - X_i))^2$
2. $v_2(x) = \frac{1}{n(n-1)}(\sum_{i=1}^n \mathcal{I}_{K_h}(x - X_i))^2$
3. $\delta_1(x) = \frac{1}{n-1}\sum_{i=1}^n (\mathcal{I}_{K_h}(x - X_i) - \mathcal{I}_{K_{h_{min}}}(x - X_i))^2$
4. $\delta_2(x) = \frac{1}{n(n-1)}(\sum_{i=1}^n \mathcal{I}_{K_h}(x - X_i) - \mathcal{I}_{K_{h_{min}}}(x - X_i))^2$

Since $\widehat{Va}(h) = \frac{1}{n}\int_s^{+\infty} v_1(x) - v_2(x)dx$ and $\hat{\Delta}b(h) = \frac{1}{n}\int_s^{+\infty} \delta_1(x) - \delta_2(x)dx$, then $-\hat{\Delta}b(h) + \hat{v}(h) = -\frac{1}{n}\int_s^{+\infty} \delta_1(x) - v_1(x) - (\delta_2(x) - v_2(x))dx$.

Let a be a function that depends on h and b a constant, the following result is always satisfied:

$$(a(h) - b)^2 - a(h)^2 = -2a(h)b + c' \tag{2.78}$$

The value of $\delta_1(x) - v_1(x)$ is computed by applying Eq. 2.78 with $a(h) = \mathcal{I}_{K_h}(t - X_i)$ and $b = \mathcal{I}_{K_{h_{min}}}(x - X_i)$

$$\delta_1(x) - v_1(x) = -\frac{2}{n-1}\sum_{i=1}^n \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_i) \tag{2.79}$$

Similarly, with $a(h) = \sum_{i=1}^n \mathcal{I}_{K_h}(x - X_i)$ and $b = \mathcal{I}_{K_{h_{min}}}(x - X_i)$ it gives

$$\delta_2(x) - v_2(x) = -\frac{2}{n(n-1)}\left(\sum_{i=1}^n \mathcal{I}_{K_h}(x - X_i)\right)\left(\sum_{j=1}^n \mathcal{I}_{K_{h_{min}}}(x - X_j)\right) \tag{2.80}$$
$$= -\frac{2}{n(n-1)}\sum_{i,j} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_j) \tag{2.81}$$

By combining the results from Eq. 2.79 and Eq. 2.81, it gives

$$-\hat{\Delta}b(h) + \hat{v}(h) = \int_s^{+\infty} \left(\frac{2}{n(n-1)} \sum_{i=1}^{n} I_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_i) \right. \tag{2.82}$$

$$\left. - \frac{2}{n^2(n-1)} \sum_{i,j} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_j) \right) dx \tag{2.83}$$

- $\widehat{Crit}_{pco}$: as a reminder in Eq. 2.77 the term $||\hat{p}_h - \hat{p}_{h_{min}}||^2$ is decomposed into a sum of a quadratic term $\frac{1}{n^2}\sum_{i,j}\int_s^{+\infty} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_h}(x - X_j)dt = \mathcal{Q}(h)$ and a multiplicative term $-\frac{2}{n^2}\sum_{i,j}\int_s^{+\infty} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_j)dx = \mathcal{M}(h, h_{min})$. In $\mathcal{M}(h, h_{min})$, the same terms are found as in $\delta_2(x) - v_2(x)$. By adding the multiplicative term to the bias correction and the variance $(-\hat{\Delta}b(h) + \hat{v}(h))$, the result is as follows, after factoring similar terms:

$$\mathcal{M}(h, h_{min}) - \hat{\Delta}b(h) + \hat{v}(h) = \left(\frac{-2}{n^2} + \frac{-2}{n^2(n-1)} \right) \int_s^{+\infty} \sum_{i,j} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_j)dx$$
$$+ \left(\frac{2}{n(n-1)} \right) \int_s^{+\infty} \sum_{i} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_i)dx$$

Since $\left(\frac{-2}{n^2} + \frac{-2}{n^2(n-1)} \right) = \frac{-2(n-1)-2}{n^2(n-1)} == \frac{-2}{n(n-1)}$, then it gives that:

$$\mathcal{M}(h, h_{min}) - \hat{\Delta}b(h) + \hat{v}(h) = \frac{-2}{n(n-1)} \left(\int_s^{+\infty} \sum_{i,j} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_j)dx \right.$$
$$\left. - \int_s^{+\infty} \sum_{i} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_i)dx \right)$$
$$\mathcal{M}(h, h_{min}) - \hat{\Delta}b(h) + \hat{v}(h) = \frac{-2}{n(n-1)} \int_s^{+\infty} \sum_{i \neq j} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_j)dx$$

Finally, $\widehat{Crit}_{pco}$ is written as the sum of the quadratic term $\mathcal{Q}(h)$ and a multiplicative term $\mathcal{M}_2(h, h_{min})$, where the multiplicative term $\mathcal{M}_2(h, h_{min})$ being the term $\mathcal{M}(h, h_{min})$ without the diagonal $(i = j)$.

$$\widehat{Crit}_{pco}(h) = \mathcal{Q}(h) + \mathcal{M}_2(h, h_{min}) \tag{2.84}$$

$$= \frac{1}{n^2} \sum_{i,j} \int_s \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_h}(x - X_j)dx \tag{2.85}$$

$$+ \frac{2}{n(n-1)} \int_s \sum_{i \neq j} \mathcal{I}_{K_h}(x - X_i)\mathcal{I}_{K_{h_{min}}}(x - X_j)dx \tag{2.86}$$

In the last part of the proof, it is shown that the LOO criterion, introduced in Definition 2.5, can be written as Eq. 2.86.

As a reminder, the LOO criterion is written as:

$$ISE_{LOO} = \underbrace{\frac{1}{n^2}\sum_{i,j}\int_s \mathcal{I}_{K_h}(x-X_i)\mathcal{I}_{K_h}(x-X_j)dt}_{C(h)} \tag{2.87}$$

$$+ \underbrace{\frac{1}{(n+1)n}\sum_{i=1}^{n+1}\sum_{j=1;j\neq i}^{n+1}\mathbb{1}[X_i>s]\int_s^{X_i}\int_x^{+\infty}\mathcal{I}_{K_h}(z-X_j)dzdx}_{\hat{D}(h)} \tag{2.88}$$

Recognizing that $C(h)$ is equal to the quadratic term of $\widehat{Crit}_{pco}$, all what remains to complete the proof, is to prove the following equality:

$$\mathbb{1}[X_i>s]\int_s^{X_i}\mathcal{I}_{K_h}(t-X_j)dt = \int_s \mathcal{I}_{K_h}(t-X_j)\mathcal{I}_{K_{h_{min}}}(t-X_i)dt \tag{2.89}$$

For this purpose, the integration limits $[s, X_i]$ are replaced by an indicator function $\mathbb{1}[X_i > t]$.

$$\mathbb{1}[X_i>s]\int_s^{X_i}\mathcal{I}_{K_h}(t-X_j)dt = \mathbb{1}[X_i>s]\int_s \mathbb{1}[X_i>t]\mathcal{I}_{K_h}(t-X_j)dt \tag{2.90}$$

$$= \int_s \mathbb{1}[X_i>t]\mathcal{I}_{K_h}(t-X_j)dt \tag{2.91}$$

This gives the desired result, since by hypothesis, the equality $\mathbb{1}[X_i > t] = \mathcal{I}_{K_{h_{min}}}(t-X_i)$ is assumed true.

□

Chapter 3

FDR Control For Online Anomaly Detection

In this chapter, the results published in "FDR Control for Online Anomaly Detection" [95] are presented. A new data-driven threshold procedure ensuring FDR control at a desired level α is built. Our strategy relies on a local control of the "modified FDR" (mFDR) on subseries. An important ingredient in this control is the cardinality of the calibration set used to compute the empirical p-values, which turns out to be an influential parameter. A new strategy for tuning this parameter is developed that yields the desired FDR control over the entire time series. The statistical performance of this strategy is analyzed by theoretical guarantees and its practical behavior is assessed by simulation experiments, which support our conclusions.

3.1 Introduction

3.1.1 Alarm fatigue

By observing indicators along the time to check the system health, anomaly detection aims at raising an alarm if abnormal patterns are detected [2, 105]. A motivation for automatic anomaly detection is to reduce the workload of operations teams by allowing them to prioritize their efforts where necessary. This is usually made possible by using statistical and machine learning models [39, 29, 20]. However when badly calibrated an anomaly detector leads to alarm fatigue. An overwhelming number of alarms desensitizes the people tasked responding to them, leading to missed or ignored alarms or delayed responses [45, 21]. One of the reasons for alarm fatigue is the high number of false positives which take time to be managed [153, 103]. The main goal of the present work is to design a new (theoretically grounded) strategy allowing to control the number of false positives when performing automatic anomaly detection in sequential context. Literature is presented first to illustrate the challenges of reducing the number of false positives.

3.1.2 Related work

As seen in Section 1.3, a high diversity of atypicity score functions can be used to detect different abnormality patterns [56]. Abnormality scores are often not easily interpretable if the score distribution is unknown. Therefore, it is impossible to make a judicious choice of the detection threshold. The Conformal Anomaly Detection was introduced to alleviate this issue.

Conformal Anomaly Detection Conformal Anomaly Detection (CAD) introduced in [100] is a method derived from Conformal Prediction [5]. The goal of CAD is to give a probabilistic interpretation of the score using conformal p-values. Inductive Conformal Anomaly Detection (ICAD) introduced in [100] improves the CAD linear complexity in time and adapts it for Online Anomaly Detection by introducing the concept of *calibration set*. CAD can be used with a wide variety of anomaly score functions. For instance [141] presents an anomaly detector based kernels combined with CAD. The paper [33] combined distance and density based scoring function with CAD. CAD gives the opportunity to control the expected number of false positives within a time period. But its main limitation is that it yields no control over the false alarm rate on the whole time series that is, the proportion of false positive among all detections. By contrast the present work aims at having a control over it, more precisely on the False Discovery Rate (FDR).

FDR Control Benjamini-Hochberg (BH) procedure [13, 14] is a multiple testing procedure that controls the proportions of false positives among rejections that is the False Discovery Rate. The BH procedure can be improved by estimating the proportion of anomalies in the dataset [37, 70]. Most procedure based on Benjamini-Hochberg assume that the true p-values are known. When the distribution of the scores under the null hypothesis is unknown it is generally not possible to ensure the FDR control with BH. For instance the Monte-Carlo Multiple-Testing has been suggested by [73, 60, 175] to overcome this difficulty. In offline context, FDR can be controlled using conformal p-values with BH as shown in [11, 171, 111]. Moreover the FDR control can be achieved simultaneously with upper and lower bound as suggested in [110]. But as illustrated in Section 3.8.1 the power can be really low with small calibration set. An alternative method for controlling FDR is based on the so-called "local FDR" [160, 169]. Unfortunately this approach relies on a Gaussian assumption.

Online FDR Control In online multiple-testing, the decision of new observed value as an anomaly has to be done instantaneously. If the BH procedure is applied on the current time series, the time complexity will increase with the length of the time series. To tackle this problem, recent papers advocate different methods for the online control of the FDR [169, 87, 133, 172]. In [169] the author suggests using the principle of local FDR. At each observation, a decision is taken depending on the estimation of the local FDR. The [87, 133, 172] introduce a method based on alpha-investing. The p-value is compared to an adaptive threshold depending on the previous decisions. But this method is not applicable for conformal p-values because of its low detection power.

Controlling false positives for online anomaly detection remains a difficult task. In particular two challenges arise with online anomaly detection:

- The true p-values are unknown and need to be estimated.
- The decisions are made in an online context, whereas most of the multiple testing methods are done in the offline context.

The main contributions are to tackle these challenges. More precisely it is established that it is possible to design online anomaly detectors controlling the FDR of the time series.

- This chapter study the relationship between the FDR and the cardinality of the calibration set used to estimate p-values. To guarantee FDR control, a calibration set cardinality tuning method is proposed.
- This chapter describes an online calibration strategy for anomaly detection based on multiple testing ideas to control the False Discovery Rate (FDR).
  - It explains how control of the whole time series FDR can be obtained from control of a modified version of subseries mFDR. This makes it possible to control the FDR within an online context.
  - A modified version of the Benjamini-Hochberg procedure is suggested to achieve local control of the modified FDR.

3.1.3 Description of the chapter

First, the problem is explained and important objects are introduced in Section 3.2. Second Section 3.3 deals with conditions on p-values estimations to ensure local control of FDR is controlled at a desired level. Third this chapter develops algorithms that allows global control the FDR time series and studies them in Section 3.4. Finally our solution is evaluated against one competitor from the literature in Section 3.5.

3.2 Statistical framework

3.2.1 The Anomaly Detector

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, with Ω the set of all possible outcomes, $\mathcal{F}$ a σ-algebra on Ω and $\mathbb{P}$ a probability measure on $\mathcal{F}$. Assume a realization of the random variables $(X_t)_{t\geq 1}$, with X_t taking values in a set $\mathcal{X}$ for all t. $T \in \mathbb{N} \cup \{\infty\}$ is the length of the time series. Let $\mathcal{P}_0$ be a probability distribution, called reference distribution, on the space $\mathcal{X}$. For each instant t, the observation X_t is called "normal" if $X_t \sim \mathcal{P}_0$. Otherwise, X_t is an "anomaly". The aim of an online anomaly detector is to find all anomalies among the new observations along the time series $(X_t)_{t\geq 1}$: for each instant $t > 1$, a decision is taken about the status of X_t based on past observations: $(X_s)_{1\leq s\leq t}$.

Like many other anomaly detectors, our detector is based on the notions of atypicity score, p-value, and threshold. The novelty of our approach lies in the use of a data-driven threshold, which allows to control FDR at a desired α level. This threshold is calculated by applying the Benjamini-Hochberg procedure to a subseries of length m with a carefully chosen α' level. At the same time, the size of the calibration set must be correctly specified. More precisely, the new anomaly detector described in Algorithm 1 relies on:

1. **Atypicity score**: A score $a : \mathcal{X} \to \mathbb{R}$ is a function reflecting the atypicity of an observation X_t. To be more specific, the further $\mathcal{P}_{X_t}$ is from $\mathcal{P}_0$, the larger $a(X_t)$ is expected to be. It is often implemented using a non-conformity measure (NCM) [100], which measures how different a point X_t is from a training set $\mathcal{X}^{train}$, $a(X_t) = \overline{a}(X_t, \mathcal{X}^{train})$.
2. **p-value**: It is the probability of observing $a(X)$ higher than $a(X_t)$ if $X \sim \mathcal{P}_0$. It is

estimated using the empirical p-value, by the following equation:

$$\hat{p}_e(s_t, \mathcal{S}_t^{cal}) = \frac{1}{|\mathcal{S}_t^{cal}|} \sum_{s \in \mathcal{S}_t^{cal}} \mathbb{1}[s_t > s] \tag{3.1}$$

Where $S_t^{cal} = \{a(X_{u_1}), \ldots, a(X_{u_n})\}$ is the calibration set with n data points. The p-value enables an interpretable criterion measuring how unlikely $X_t \sim \mathcal{P}_0$ is. The empirical p-value is chosen because it is agnostic to the true and unknown distribution, which is not the case for the Gaussian p-value estimator. The performance of the BH procedure applied to empirical p-values depends strongly on the cardinality of the calibration set ($|\mathcal{S}_t^{cal}| = n$). Section 3.3.2 investigates how to optimally choose this number.

3. **Detection threshold**: $\varepsilon_t \in [0, 1]$, it discriminates observations considered as abnormal from others. The observations considered as anomalies are X_t whose (estimated) p-value is smaller than the threshold ε_t. To control FDR of the entire time series, a data-driven threshold is computed using the m most recent p-values. This detection threshold is computed using a multiple testing procedure inspired by Benjamini-Hochberg (BH) and described in Section 3.4.3. This procedure requires the calculation of an α' value estimated from a training set or using heuristics. See Section 3.4.4.1 for more details.

Algorithm 1 FDR Control Online Anomaly Detection

Require: T length of the time series, $(X_t)_{1 \leq t \leq T}$ time series, α desired FDR, m subseries length, ν integer to tune the calibration set cardinality

Require: Either (Z_t) an historical dataset or π the proportion of anomalies

if historical dataset **then**
  $\alpha' \leftarrow \arg\max_{\tilde{\alpha}} \left(\frac{\hat{\mu}_{R^{**}_{\tilde{\alpha}}}}{\hat{\mu}_{R_{\tilde{\alpha}}}} \tilde{\alpha} \leq \alpha \right)$ ▷ Estimate α' with training set
else
  $\alpha' \leftarrow \frac{\alpha}{1 + \frac{1-\alpha}{m\pi}}$ ▷ Estimate α' with heuristics
end if
$n \leftarrow \nu \cdot \frac{m}{\alpha'} - 1$ ▷ Get the calibration set cardinality
for t in $[1, T]$ **do**
  $s_t \leftarrow a(X_t)$
  $\hat{p}_t \leftarrow \hat{p}_e(s_t, \mathcal{S}_t^{cal})$ ▷ Compute empirical p-value
  $\hat{\varepsilon}_t \leftarrow \hat{\varepsilon}_{BH_{\alpha'}}(\hat{p}_{t-m}, \ldots, \hat{p}_t)$ ▷ Get the threshold using Benjamini-Hochberg
  if $\hat{p}_t < \hat{\varepsilon}_t$ **then** ▷ Retrieve anomalies
    $d_t = 1$
  else
    $d_t = 0$
  end if
end for
Output: $(d_t)_{t=1}^T$ boolean list that represent the detected anomalies.

In the next sections, the design choices of Algorithm 1 are specified and justified by a theoretical study of the detector. Section 3.3 studies the local control of FDR when BH is applied to subseries of m empirical p-values. This allows to specify the choice of the calibration set cardinality used in Algorithm 1. Then, in Section 3.4, the FDR control on the complete time series is obtained from the local control of some modified FDR. This allows to specify the procedure for selecting α' in Algorithm 1.

In classical anomaly detector a constant threshold ε allows to control the "detection frequency": a smaller threshold will generate fewer detections. This is equivalent to defining anomalies as points above a quantile in the tail of the score distribution. Nevertheless, in practice, the calibration of the threshold is difficult. Since ε affects directly the number of false detections (false positives), it is not possible to know in advance the number of false positives due to the choice of ε.

One of the main contributions of this work consists in developing a data-driven rule allowing to choose a threshold $\hat{\varepsilon}_t$ at each time step t. This rule has the advantage of ensuring a global control of the false discovery rate on the complete set of observations.

3.2.2 Control of false positives and multiple testing

Since the present goal is to use FDR, a natural strategy is to rephrase the online anomaly detection problem as a multiple testing problem: At each step $1 \leq t \leq T$, a statistical test is performed on the hypotheses:

$$\mathcal{H}_{0t}, \text{“}X_t \text{ is not an anomaly ”} \qquad \text{against} \qquad \mathcal{H}_{1t} \text{“}X_t \text{ is an anomaly”}.$$

A natural criterion controlling the proportion of type I errors (False Positives) of the whole time series is FDR [13]. For a given data-driven threshold $\hat{\varepsilon}$ and a set of *estimated* p-values $\hat{p} = (\hat{p}_t)_{t\geq 1}$, the FDR criterion of the sequence from 1 to T is given by

$$\begin{aligned} &FDR_1^T(\hat{\varepsilon}, \hat{p}) = \mathbb{E}[FDP_1^T(\hat{\varepsilon}, \hat{p})], \\ &\text{with} \quad FDP_1^T(\hat{\varepsilon}, \hat{p}) = \frac{\sum_{t\in\mathcal{H}_0} \mathbb{1}[\hat{p}_t \leq \hat{\varepsilon}_t]}{\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t \leq \hat{\varepsilon}_t]}, \end{aligned}$$

with the convention that $0/0 = 0$. In the above expression, FDP_1^T denotes the *False Discovery Proportion* (FDP) of the time series from 1 to T. Also $\mathcal{H}_0 = \{t \in \mathbb{N}^* | \mathcal{H}_{0t} \text{ is true}\}$ is called the set of null hypotheses. Let us emphasize that the anomalies (according to Algorithm 1) satisfy $\mathbb{1}[\hat{p}_t \leq \hat{\varepsilon}_t] = 1$. The notation $FDR_1^T(\hat{\varepsilon}, \hat{p})$, used in this chapter, highlights the impact of $\hat{\varepsilon}$ and $\hat{p}$ on the FDR value. The main objective of the present work is to define a data-driven sequence $\hat{\varepsilon} : t \mapsto \hat{\varepsilon}_t$ such that, for a given control level $\alpha \in [0, 1]$, under weak assumptions on the sequence $\hat{p} : t \mapsto \hat{p}_t$,

$$FDR_1^T(\hat{\varepsilon}, \hat{p}) \leq \alpha \tag{3.2}$$

The control is said exact when "$\leq$" is replaced with "$=$". Such a control would imply that for a level $\alpha = 0.1$, at most 10% of the detected anomalies along the whole time series are false positives.

The detection power of the anomaly detector is measured by means of the *False Negative Rate* defined, for the sequence from 1 to T, by

$$FNR_1^T(\hat{\varepsilon}, \hat{p}) = \mathbb{E}[FNP_1^T(\hat{\varepsilon}, \hat{p})], \tag{3.3}$$

$$\text{with} \quad FNP_1^T(\hat{\varepsilon}, \hat{p}) = \frac{\sum_{t\in\mathcal{H}_1} \mathbb{1}[\hat{p}_t \leq \hat{\varepsilon}_t]}{|\mathcal{H}_1|}, \tag{3.4}$$

where FNP_1^T denotes the *False Negative Proportion* (FNP) of the sequence from 1 to T and $\mathcal{H}_1 = \{t \in \mathbb{N}^* | \mathcal{H}_{1t} \text{ is true}\}$ is the set of alternative hypotheses.

However a crucial remark at this stage is that controlling FDR on the complete time series

is a highly challenging task in the present online context for at least two reasons:

- The main existing approaches for controlling FDR are described in an "offline" framework where the whole series is observed first, and decisions are taken afterwards [13, 110]. This makes these approaches useless in the present context.
- The already existing approaches designed in the online context [87, 172, 137] are difficult to parameterize and hard to apply with *estimated p-values*. Let us emphasize that realistic scenarios usually exclude the knowledge of the true probability distribution of the test statistics, leading to approximating or estimating the related p-values in practice. For example, [135] present results only in Gaussian data. Furthermore, multiple test procedures are generally not established with anomaly detection in mind, and perform poorly when the proportion of anomalies is close to 0.

3.2.3 FDR control with Empirical p-value

A classical (offline) strategy for controlling FDR is the so-called Benjamini-Hochberg (BH) multiple testing procedure [13]. Exact control relies on the knowledge of true p-values, which is usually not realistic. Actually since the true reference distribution is unknown in practical anomaly detection scenarios, there is no true p-values available.

3.2.3.1 Empirical p-value

The atypicity level of an observation is quantified by an atypicity score. The underlying scoring function assigns each observation with a real value such that *the more atypical the observation, the higher the score value*. The interpretation is that the higher the score (value) at an observation, the more unlikely the corresponding observation has been generated from a reference distribution implicitly encoded in the scoring function.

Examples (Examples of scoring functions). Let $x \in \mathcal{X}$ and $z_1^\ell = \{z_1, \ldots, z_\ell\}$ be a training set generated from $\mathcal{P}_0$.

1. Z-score [178, 32]: Let μ and σ be estimators of mean and standard deviation of z_1^ℓ,

$$a(x) = a_Z(x, z_1^\ell) = |(x - \hat{\mu})/\hat{\sigma}|$$

2. kNN score [151, 33]: Let d be a metric on $\mathcal{X}$ and $k > 0$ and $kNN(x, z_1^\ell)$ is the set k-th nearest neighbors of x in z_1^ℓ.

$$a(x) = a_{kNN}(x; z_1^\ell, k) = \frac{1}{k} \sum_{z \in kNN(x, z_1^\ell)} d(x, z)$$

The choice of the abnormality score depends on the structure of the time series and the type of anomalies one is looking for. Intuitively a desirable scoring function should assign a high abnormality score to any true anomaly. For example, Z-score is only able to detect anomalies that are in the tail of the distribution. By contrast it is not effective to detect abnormal point between two modes of data with a bimodal distribution [178, 32]. kNN score is more suited for multi-modal data because they raise a high score for points far from the observations of the training set. The intuition behind such a scoring function is that normal data should have a low distance from the training set. To the best of our knowledge, there does not exist any scoring function suitable for detecting all types of anomalies.

Defining a meaningful threshold from a score is the classical strategy for deciding that an observation is anomalous or not. This requires to know the true distribution of these scores, which is not realistic in general. The induced estimation step is usually made by two means. On the one hand, one can assume a parametric Gaussian distribution for the scores [154, 32, 78]. On the other hand, one can estimate the score distribution by use of sampling techniques [141, 33]. Since the Gaussian assumption can cause some troubles when it is violated, the present work rather focuses on the second strategy by considering anomaly detection relying on empirical p-values. By contrast to the Gaussian assumption, a strong asset of empirical p-value is that they can be used no matter the true score distribution or the scoring function.

Definition 3.1 (Empirical p-value)**.** *Let a be a scoring function. Let $\{x_1, ..., x_n\} \subset \mathcal{X}$ be a set of data called the calibration set and their value of the atypicity score is noted $s_i = a(x_i)$. The empirical p-value is a function defined by*

$$\forall s \in \mathbb{R}, \quad \hat{p}_e(s; \{s_1, \dots, s_n\}) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(s_i \geq s). \tag{3.5}$$

Let us emphasize that Definition 3.1 describes an estimator of the p-value under $\mathcal{P}_0$ provided the calibration set is composed of points generated from the reference distribution $\mathcal{P}_0$. However it is well known that the main difficulty with this p-value estimator is that it is not itself a p-value [111] since the so-called *super-uniformity property* is violated. More precisely, super-uniformity means that, for all $u \in [0, 1]$,

$$\mathbb{P}_{X, X_1, \dots, X_n \sim \mathcal{P}_0}(\hat{p}_e(X; \{X_1, \dots, X_n\}) \leq u) \leq u.$$

Therefore empirical p-values are usually replaced by an other p-value estimator called the conformal p-value[110, 100], given by

$$\hat{p}_c(s; \{s_1, \dots, s_n\}) = \frac{1}{n+1} \left(1 + \sum_{i=1}^{n} \mathbb{1}(s_i \geq s) \right). \tag{3.6}$$

This definition implies the p-value property for all u in $[0, 1]$. But this estimator is less powerful, as illustrated by Figure 3.11 in Section 3.8.1 where the FNR resulting from the use of conformal p-values is always larger than that of empirical p-values.

As a consequence, an important remark is that the present work focuses on empirical p-values (and not on conformal ones). However another motivation for this choice is provided in Section 3.3.2.2 where it is proved that the *super-uniformity* property also holds true with empirical p-values under some specific conditions that will be detailed later.

3.2.3.2 BH-procedure does not control FDR with empirical p-values

The present section starts by describing the behavior of the BH-procedure as well as establishing the resulting FDR control. An illustration is provided that the BH-procedure does not control FDR are the prescribed level when empirical p-values are used. This illustration is then theoretically justified, which shows that straightforwardly using the BH-procedure in our online context is prohibited.

Definition 3.2 (Benjamini-Hochberg ([13, 171]))**.** *Let m be an integer and $\alpha \in [0, 1]$. Let $(p_i)_{1 \leq i \leq m} \in [0, 1]^m$ be a family of p-values. The Benjamini-Hochberg (BH) procedure, denoted*

by BH_α, is given by

- *a data-driven threshold:*

$$\hat{\varepsilon}_{BH_\alpha}(p_1,\dots,p_m) = max\{\frac{\alpha k}{m}; p_{(k)} \leq \frac{\alpha k}{m}, k \in [\![1,m]\!]\},$$

- *a set of rejected hypotheses:*

$$BH_\alpha\left(p_1,\dots,p_m\right) = \{i; p_i \leq \hat{\varepsilon}_{BH_\alpha}, i \in [\![1,m]\!]\}\,.$$

The intuition behind this procedure consists in drawing the ordered statistics $i \mapsto p_{(i)}$ (Figure 4.15) with $p_{(1)} \leq \dots \leq p_{(n)}$ and the straight line $i \mapsto \frac{\alpha i}{m}$. Then the BH-procedure amounts to rejecting all hypotheses corresponding to p-values smaller than the last crossing point between the straight line and the ordered p-values curve.

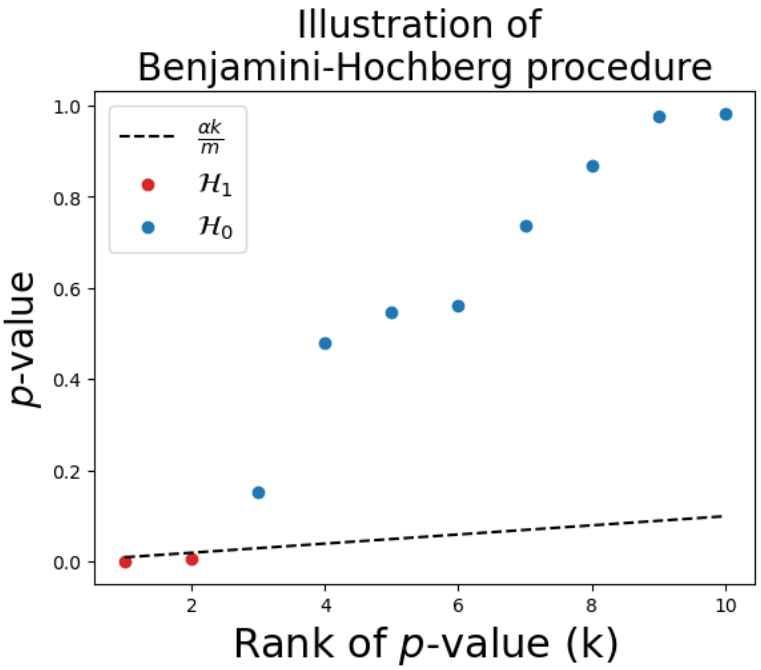


Figure 3.1: Illustration of the Benjamini-Hochberg procedure. p-values are sorted by increasing order. The threshold is the greatest p-value that is lower than $\alpha k/m$, when k is the rank of the p-value.

The striking property of this procedure is to yield the desired control of the FDR at the prescribed level α as stated by the next result.

Theorem 3.1 (FDR control with BH [13])**.** *Let m be a positive integer and $(X_i)_1^m$ be* independent *random variables such that $X_i \sim \mathcal{P}_0$, $1 \leq i \leq m_0$, and $X_i \sim \mathcal{P}_1$, $m_0+1 \leq i \leq m$. Let us also define the set of true p-values, for all $1 \leq i \leq m$ by $p_i = \mathbb{P}_{X\sim P_0}(a(X) \geq a(X_i)) \in [0,1]$, and assume that each $p_i \sim U([0,1])$. Then for every $\alpha \in]0,1]$, BH_α applied to $p = (p_i)_{1\leq i\leq m}$ yields*

the exact FDR control at the prescribed level α that is,

$$FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, p) = \frac{m_0\alpha}{m}.$$

The proof of the theorem is deferred to Section 3.7.1. In particular, the FDR control results from the fact that under $\mathcal{H}_0$, the true p-values follow a uniform distribution. The equality could be replaced by an upper bound if the uniform distribution assumption were weakened by the super-uniform property.

By contrast with the previous framework, when performing anomaly detection, the abnormality score is computed using a scoring function, and the true p-value is given by

$$p_t = \mathbb{P}_{X\sim\mathcal{P}_0}\left(a(X) \geq a(X_t)\right),$$

where the notation clearly emphasizes the dependence with respect to the *unknown* reference distribution. This justifies why empirical p-values are now substituted to true ones as earlier explained (see Eq. 3.5).

A difficulty resulting from using empirical p-values in the BH-procedure is that the FDR control does no longer hold true as illustrated by Figure 3.2. This figure displays the actual FDR value (plain blue curve) versus the cardinality of the calibration set used to computed the empirical p-values (see Definition 3.1) in the specific situation of Gaussian data. Except for some a few values of the calibration set cardinality, the FDR control is no longer achieved (red horizontal line). Furthermore the actual FDR value is higher than the desired $m_0/m\alpha$. This results from the fact that the super uniform property is violated when using empirical p-values as established by Proposition 3.1 below.

Proposition 3.1 (Distribution of empirical p-value under H_0)**.** *Let $X \sim \mathcal{P}_0$ where $\mathcal{P}_0$ is the probability distribution under $\mathcal{H}_0$, the calibration set cardinality is denoted by n, and $\{X_1, \ldots, X_n\} \sim \mathcal{P}_0^n$ is the calibration set. If one further assumes that there are no ties among $a(X_1), \ldots, a(X_n)$, then the empirical p-value at X is denoted by $\hat{p}_e(a(X); \{a(X_1), \ldots, a(X_n)\})$ and follows the discrete uniform distribution*

$$U(0, \frac{1}{n}, \frac{2}{n}, \ldots, 1).$$

Let us mention that under $\mathcal{H}_0$ the empirical p-value has a different distribution from that of the conformal p-value [110] which follows $U(1/(n+1), \ldots, 1)$. The conformal p-value is never smaller than $1/(n+1)$, which raises issues in terms of the detection power with lots of false negatives (see the right panel of Figure 3.11 and the discussion in Section 3.8.1). With empirical p-values, it can be easily checked that

$$\mathbb{P}\left(\hat{p}_e(a(X); \{a(X_1), \ldots, a(X_n)\}) \leq 0\right) = \frac{1}{n+1} > 0,$$

which violates the super uniformity property. As a consequence, FDR is no longer controlled by the BH-procedure [14] applied to empirical p-values. Other consequences owing to the use of empirical p-values violating super uniformity are illustrated in Section 3.3.1.

The assumption of no ties are allowed among the scores $a(X_i)$s is quite mild and fulfilled most of the time as supported by Example 3.2.3.1 as long as the reference distribution is continuous (admits a density).

Proof of Proposition 3.1. Since $X \sim \mathcal{P}_0$ and $X_1, \ldots, X_n \sim \mathcal{P}_0^n$ are independent, the empirical p-value $\hat{p}$ is a random variable computed from X and X_1^n and satisfies for any $0 \leq \ell \leq n$ that

$$\begin{aligned}
&\mathbb{P}(\hat{p} = \ell/n) \\
&= \mathbb{P}(a(X_{(\ell+1)}) < a(X) \leq a(X_{(\ell)})) \\
&= \mathbb{P}(\{\text{rank of } a(X) \text{ among } \{a(X), a(X_1), \ldots, a(X_n)\} \text{ is } \ell + 1\}),
\end{aligned}$$

where $a(X_{(n)}) < a(X_{(n-1)}) < \cdots < a(X_{(1)})$. The conclusion comes from noticing that the probability distribution of $\{X, X_1, \ldots, X_n\}$ is exchangeable, and assumption of no ties in scores $a(X_i)$ □

Let us also notice that Figure 3.2 shows that there exist particular values of the calibration set cardinality for which FDR is still controlled at the prescribed level. This perspective is further explored in Section 3.3.2.2, where a new multiple testing procedure yielding the desired FDR control for the whole time series is devised.

3.3 FDR control with Empirical p-values

The goal here is to describe a strategy achieving the desired FDR control for a time series of length m when using empirical p-values. A motivating example is first introduced for emphasizing the issue in Section 3.3.1. Then a theoretical understanding is provided along Section 3.3.2 which results in a new solution which applies to independent empirical p-values. An extension is then discussed to the non-independent setup in Section 3.3.3.1. Finally experimental results are reported in Section 3.3.4 to (empirically) assess the validity of our previous theoretical conclusions.

3.3.1 Motivating example

The purpose here is to further explore the effect of the calibration set cardinality on the actual FDR control when using empirical p-values. This gives us more insight on how to find mathematical solutions.

Let us start by generating observations using two distributions. The reference distribution is $\mathcal{P}_0 = \mathcal{N}(0, 1)$ and the alternative distribution is $\mathcal{P}_1 = \mathcal{N}(4, 10^{-4})$. The anomalies are located in the right tail of the reference distribution. The length m of the signal is $m = 100$. The number of observations under $\mathcal{P}_0$ is $m_0 = 99$. The experiments have been repeated $B = 10^4$ times.

Figure 3.2 displays the actual value of FDR as a function of the cardinality n of the calibration set $\{x_1, ..., x_n\}$ used to compute the empirical p-values (see Definition 3.1). One clearly see that FDR is not uniformly controlled at level $m_0/m\alpha$. However there exist particular values of n for which this level of control is nevertheless achieved. As long as n has become large enough ($n \geq 500$), repeated picks can be observed with a decreasing height as n grows.

3.3.2 FDR control: main results for *i.i.d.* p-values

The present section aims at first explaining the shape of the curve displayed in Figure 3.2. This will help getting some intuition about how to design an online procedure achieving the desired FDR control for the full time series.

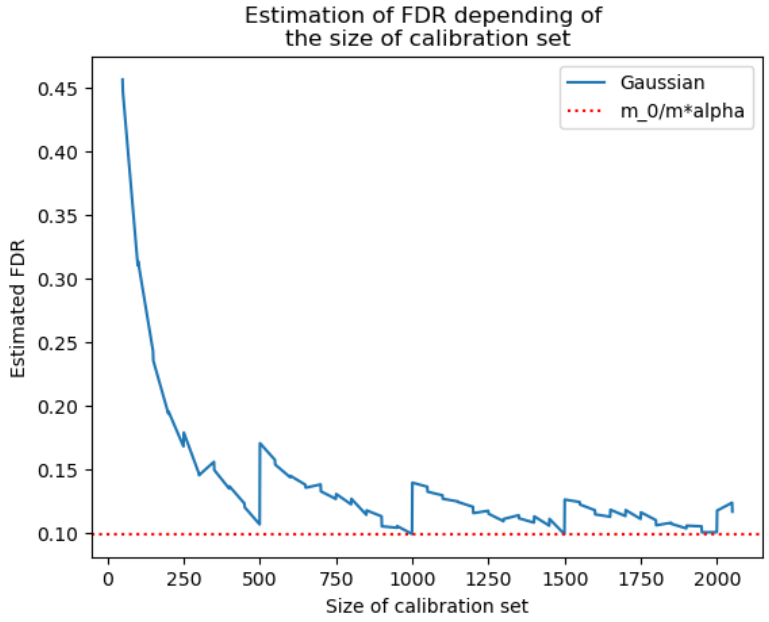


Figure 3.2

3.3.2.1 Proof of FDR control by BH revisited

The main focus is first given to independent p-values. In what follows, the classical proof (Proof 3.7.1) of the FDR control by the BH-procedure is revisited then leading to the next result. Its main merit is to provide the mathematical expression of the plain blue curve observed in Figure 3.2.

Theorem 3.2. *Let n be the cardinality of the calibration sets and m be that of the set of tested hypotheses where $\{X_1, \dots, X_m\}$ denotes a set of random variables. Let $m_0 \leq m$ be the cardinality of the random variables from the reference distribution $\mathcal{P}_0$. Let the empirical p-value be denoted, for any $i \in [\![1, m]\!]$, by $\hat{p}_i = \hat{p}_e(a(X_i), \{a(Z_{i,1}), \dots, a(Z_{i,n})\})$, where the calibration set is $\{Z_{i,1}, \dots, Z_{i,n}\}$ and each $Z_{i,j} \sim \mathcal{P}_0$. Each p-value is calculated using calibration sets that are independent of each other. Let the random variables $R(i)$ be the number of detections raised by BH_α when replacing X_i with 0, as defined along the proof detailed in Section 3.7.1. Then for every $\alpha \in]0, 1]$, the FDR value over the sequence from 1 to m is given by*

$$FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) = m_0 \sum_{k=1}^{m} \frac{\frac{\lfloor \frac{\alpha k n}{m} \rfloor + 1}{n+1}}{k} \mathbb{P}(R(i) = k),$$

where $\hat{\varepsilon}_{BH_\alpha}$ denotes the BH_α threshold from Definition 3.2 when the BH-procedure is applied to the empirical p-values $\hat{p} = (\hat{p}_i; 1 \leq i \leq m)$.

In general it is not possible to compute the exact value of the FDR without knowing the distribution of the random variables $R(i)$. This is in contrast with the case of true p-values where $\mathbb{P}(p_i \leq \frac{\alpha k}{m}) = \frac{\alpha k}{m}$, where k are simplified, whereas with empirical p-values $\mathbb{P}(\hat{p}_i \leq \frac{\alpha k}{m}) = \frac{\lfloor \frac{\alpha k n}{m} \rfloor + 1}{n+1}$, which prevents from any simplification of the final bound. Nevertheless this value still suggests a

solution to circumvent this difficulty: requiring conditions on α, m, and n such that $\frac{\lfloor \frac{\alpha k n}{m} \rfloor + 1}{n+1} = \frac{\alpha k}{m}$, for all k. This is precisely the purpose of next Corollary 3.1.

Proof of Theorem 3.2. When applying Proof 3.7.1, the only modification is that $\mathbb{P}(\hat{p}_i \leq \frac{\alpha k}{m})$ is not equal to $\frac{\alpha k}{m}$ since $\hat{p}_i$ now follows the discrete uniform distribution

$$\mathbb{P}(\hat{p}_i \leq \frac{\alpha k}{m}) = \sum_{\ell=0}^{\lfloor n\alpha k/m \rfloor} \mathbb{P}(n\hat{p}_i = \ell) = \frac{\lfloor \frac{\alpha k n}{m} \rfloor + 1}{n+1}.$$

Plugging this in the FDR expression, it gives

$$FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) = m_0 \sum_{k=1}^{m} \frac{\frac{\lfloor \frac{\alpha k n}{m} \rfloor + 1}{n+1}}{k} \mathbb{P}(R(i) = k).$$

Recall that $\hat{p}_i$ follows $U(0, 1/n, 2/n, \ldots, 1)$ entails that $n\hat{p}_i$ follows $U(0, 1, 2, \ldots, n)$. □

3.3.2.2 Tuning of the calibration set cardinality

The previous result is used to suggest a tuning method for the calibration set cardinality in order to control the FDR.

Corollary 3.1. *Under the same notations and assumptions as Theorem 3.2, the next two results hold true.*

1. *Assume that there exists an integer $1 \leq \nu$ such that $\frac{\nu m}{\alpha}$ is an integer. If the cardinality n of the calibration set satisfies $n = n_\nu - 1 = \nu m/\alpha - 1$, then*

$$FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) = \frac{m_0 \alpha}{m}.$$

2. *For every $\alpha \in]0, 1]$, assume that the cardinality of the calibration set satisfies $n = n_\nu - 1 = \lceil \frac{\nu m}{\alpha} \rceil - 1$, for any integer $\nu \geq 1$. Then,*

$$\frac{n}{(n+1)} \frac{m_0 \alpha}{m} \leq FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) \leq \frac{m_0 \alpha}{m}.$$

The proof is postponed to Section 3.7.2. The first statement in Corollary 3.1 establishes that recovering the desired control of FDR at the exact prescribed level α is possible on condition that the calibration set cardinality is large enough and more precisely that $n = \nu m/\alpha - 1$. This (mild) restriction on the values of α reflects that the empirical p-values do not satisfy the super-uniformity property. By contrast, the second statement yields the desired control at the level $\alpha m_0/m$ by means of lower and upper bounds. In particular, the lower bound tells us that the FDR value can be not lower than the desired level $\alpha m_0/m$ up to a multiplicative factor equal to $1 - 1/n$, which goes 1 as n grows. For instance with $\alpha = 0.1$ and $m = 100$, $n_\nu = 1000$ would yield that $FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) \cdot m/(m_0\alpha) \in [0.999, 1]$. This small lack of control is the price to pay for allowing any value of $\alpha \in]0, 1]$. It is also important to recall that in the anomaly detection field, abnormal events are expected to be rare. As a consequence $\frac{m_0}{m}$ is close to 1 and the actual FDR level is close to the desired α. However in situations where m_0/m could depart from 1 too strongly, then incorporating an estimator of m_0/m would be helpful.

3.3.3 Extension to dependent p-values

In Section 3.3.2, Corollary 3.1 states that FDR is controlled at a prescribed level with empirical p-values for which the super-uniformity property is not fulfilled. A key ingredient in the proof was the independence property across empirical p-values. One purpose of the present section is to extend these results to non-independent p-values. Such an extension is interesting because in practice not all p-values can be calculated from independent calibration sets. In practice, BH is applied to families of m p-values, $\hat{p}_{t-m+1}, \dots, \hat{p}_t$, derived from a time series. There are essentially two ways of estimating these p-values from the (X_t) series. Either a single calibration set is used to calculate the p-values. $\forall i \in [\![1, m]\!], p_{t-m+i} = \hat{p}_e(X_{t-m+i}, \{X_{t-m-n}, \dots, \{X_{t-m})$. Or (more common in practice), the n preceding points are used as the calibration set. $\forall i \in [\![1, m]\!], p_{t-m+i} = \hat{p}_e(X_{t-m+i}, \{X_{t-m-n-i}, \dots, \{X_{t-m-i})$. In this case, there is no single calibration set, but overlapping calibration sets. The control of the FDR is studied in these two cases

Towards this extension, the concept of positive regression dependency (referred to as PRDS) [14] turns out to be useful. The PRDS property is a form of positive dependence between p-values where all pairwise p-value correlations are positive. It results that a small p-value for a given observation makes other p-values for all considered observations simultaneously small as well, and vice-versa [11].

3.3.3.1 Theoretical results to dependent p-values

A classical result established in [14] proves that FDR is upper bounded by $\alpha m_0/m$ provided the p-value family satisfies the PRDS and super-uniformity properties. It turns out that this result can be extended to our estimator with the same choice of calibration set cardinality as the one discussed in Corollary 3.1. Another important achievement is the fact that FDR can be also lower bounded in the case where the calibration set is the same for all (empirical) p-values (see Definition 3.1). This results originally proved by [110] is extended here to empirical p-values computed with a calibration set cardinality tuned as suggested in Corollary 3.1.

In Section 3.3.2, considering the cardinality of the calibration set is correctly chosen, it has been proved that the control of the FDR can be achieved with estimated p-values for which the super-uniformity property is not fulfilled. The results obtained for i.I.d. p-values will be extended in this section for non i.i.d. p-values.

For this extension, the concept of positive regression dependency on each one from a subset called PRDS [14] is introduced. The PRDS property is a form of positive dependence of p-values where all pairwise p-value correlations are positive. Larger scores in the calibration set make the p-values for all test points simultaneously smaller, and vice-versa [11].

Definition 3.3 (PRDS property)**.** *A family of p-values $\hat{p}_1^m$ is PRDS on a set $I_0 \subset \{1, \dots, m\}$ if for any $i \in I_0$ and any increasing set A, the probability $\mathbb{P}[\hat{p}_1^m \in A | \hat{p}_i = u]$ is increasing in u.*

A classical result in [14] asserts that the FDR is upper bounded by $\frac{m_0}{m}\alpha$ in the case where the p-value family is PRDS and super-uniform. This result can be extended our estimator with the same choice of calibration set cardinality than in Theorem 3.1.

Corollary 3.2 (Corollary of Theorem 1.2 in [14])**.** *Suppose the family of p-values $\hat{p}_1^m$ is PRDS on the set $\mathcal{H}_0$ of true null hypotheses and suppose that $\hat{p}_1^m$ respects super-uniformity an all thresholds*

that can may resulting from BH

$$\forall k \in [\![1, m]\!] \qquad \mathbb{P}(\hat{p}_i < \frac{\alpha k}{m}) \leq \frac{\alpha k}{m},$$

Then, the FDR is upper-bounded by α

$$FDR(\hat{\varepsilon}_B H, \hat{p}) \leq \frac{m_0 \alpha}{m}$$

Unique calibration set More over, in the case where the calibration set is the same for all p-values, the FDR can also be lower bounded as shown in [110]. This result can be extended to empirical p-values given in Definition 3.1 with a calibration set cardinality tuned as proposed in Theorem 3.1.

Corollary 3.3 (Corollary of Theorem 3.4 in [110])**.** *Assuming the following conditions: Let n be the cardinality of the calibration set, n be the cardinality of the active set and m_0 the number of normal observations. Let $\mathcal{P}_0$ be the reference distribution. Let Z_i for i in $[\![1, m]\!]$ independents random variables, following $\mathcal{P}_0$. Let X_i for i in $[\![1, m]\!]$ be random independents variables and independents from (Z_j). There are exactly m_0 random variables following the $\mathcal{P}_0$ distribution. Let a be a scoring function. For all i in $[\![1, m]\!]$, let $\hat{p}_i$ be the empirical p-values associated with the random variables X_i and computed as follows, $\hat{p}_i = \hat{p}_e(a(X_i), \{a(Z_1), ..., a(Z_n)\})$.*

If the cardinality of the calibration set is a multiple of $n = n_\nu = \nu m/\alpha - 1$, then the FDR using $\hat{BH}_\alpha$ on $(\hat{p}_i)_{1 \leq i \leq m}$ is equal to $\frac{m_0 \alpha}{m}$:

$$FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) = \frac{m_0 \alpha}{m}$$

The result of this corollary is close to that of Corollary 3.1, but here the p-values are all calculated from the same calibration set whereas they were calculated on independent calibration sets.

Overlapping calibration set In the context of online anomaly detection, moving windows are classically used to capture and process the incoming data. This is why the calibration sets of the p-value family will partially overlap. To have a perfect control of the FDR, an upper and lower bounds is needed. For simplicity's sake, the score function a is not displayed in this paragraph's equations.

According Proposition 3.2, p-values with overlapping calibration sets are PRDS.

Proposition 3.2 (PRDS property for overlapping calibration sets)**.** *Let X_i for i in $[\![1, m]\!]$ be random independents variables. There are exactly m_0 random variables following the $\mathcal{P}_0$ distribution, with belong to $\mathcal{H}_0$. Let $\boldsymbol{Z}$ be the random vector that combine all calibration set, all elements of $\boldsymbol{Z}$ are generated from $\mathcal{P}_0$. The set of n indices defining the elements of calibration set related to $\hat{p}_i$ in $\boldsymbol{Z}$ is noted $\mathcal{D}_i$. The calibration set related to X_1 is noted $\boldsymbol{Z}_{\mathcal{D}_1} = (\boldsymbol{Z}_{i_1}, ..., \boldsymbol{Z}_{i_n})$. For all i in $[\![1, m]\!]$: $\hat{p}_i = p\text{-}value(X_i, \boldsymbol{Z}_{\mathcal{D}_i})$.*

Under these conditions, the set of p-values is PRDS on $\mathcal{H}_0$

The proof of the proposition is in delayed to Section 3.7.3. Since such p-values are are PRDS, it gives an upper bound control of the FDR using Corollary 3.4.

Corollary 3.4 (PRDS property for overlapping calibration sets)**.** *Under the same conditions as Proposition 3.2 and the condition on calibration set cardinality satisfy* $\exists \nu \geq 1, n = \nu \frac{m}{\alpha} - 1$*:*

$$FDR(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) \leq \frac{m_0 \alpha}{m}$$

Theoretically, the upper control is obtained for overlapping calibration sets. The question of strict control is then studied experimentally, in the next section.

3.3.3.2 Calibration set and impact of the overlap

In the context of online anomaly detection, moving windows are usually used to capture and process the incoming observations. In this context, the calibration set coincides with the data points within this window. Since successive windows are overlapping each other depending on the size of the shift, the resulting calibration sets used for computing the successive empirical p-values are also overlapping. To have a perfect control of the FDR, an upper and lower bounds are needed. Since Section 3.7.3 proves that such p-values are PRDS, it gives an upper bound control of the FDR. No theoretical results exists to compute the lower bound, indeed the existing proof in [110] did not extend to overlapping calibration sets. Therefore the next discussion suggests to establish this lower bound empirically.

The following experiments aims at drawing a comparison between the FDR values in three scenarios: independent calibration sets, partially overlapping calibration sets with an overlap size driven by the value of sn (size of the shift), and the same calibration set for all empirical p-values. To be more specific, the calibration sets (and corresponding empirical p-values) were generated according to the following scheme. Each calibration set is of cardinality n. When moving from one calibration set to the next one, the shift size is equal to sn, where s in $[0, 1]$ is the proportion of independent data between calibration sets, resulting in an overlap of cardinality $(1 - s)n$. Therefore an overlap occurs as long as $s < 1$. All these ways to build the calibration sets are called "calibration sets strategies".

1. The independent p-values (iid Cal.) are generated according to

$$\forall i \in [\![1, m]\!], \quad \mathbf{Z}_i \sim \mathcal{P}_0^n, \quad \hat{p}_{1,i} = \hat{p}_e(X_i, \mathbf{Z}_i). \tag{3.7}$$

2. The p-values with the same calibration set (Same Cal.) are generated by

$$\forall i \in [\![1, m]\!], \quad \mathbf{Z} \sim \mathcal{P}_0^n, \quad \hat{p}_{2,i} = \hat{p}_e(X_i, \mathbf{Z}). \tag{3.8}$$

3. The p-values with overlapping calibration sets (Over. Cal.) are generated given, for $0 < s < n$, by

$$\begin{aligned} &\forall i \in [\![1, m]\!], \quad \mathbf{Z}_i = \{Z_{\lfloor isn \rfloor + 1}, \ldots, Z_{\lfloor isn \rfloor + n}\}, \quad \hat{p}_{3,i} = \hat{p}_e(X_i, \mathbf{Z}_i), \\ &\qquad \text{and} \qquad \{Z_{s+1}, \ldots, Z_{\lfloor 2sn \rfloor + 1}, \ldots, Z_{\lfloor msn \rfloor + 1}, \ldots, Z_{\lfloor msn \rfloor + n}\} \sim \mathcal{P}_0^{ms+n}. \end{aligned} \tag{3.9}$$

According to these three scenarios, as s increases, the overlap cardinality decreases, which results in more and more (almost) independent calibration sets. This is illustrated by the empirical results collected in Table 3.1. For each calibration set strategy, presented in row, and for each calibration set cardinality in column, the estimated FDR is shown. In this experiment, the reference distribution $\mathcal{P}_0$ is the Gaussian $\mathcal{N}(0, 1)$ and the anomalies are equal to $\Delta = 4$. The

number of tested p-values, noted m, is equal to 100 and m_1, the number of anomalies, is equal to 1. On each sample, BH-procedure is applied with $\alpha = 0.1$ and the FDP is computed. Each FDR is estimated over 10^3 repetitions.

n	249	250	499	500	749	750	999	1000
Same Cal.	0.164	0.175	0.112	0.183	0.137	0.131	0.097	0.154
Over Cal. (s=0.1%)	0.167	0.174	0.100	0.156	0.138	0.125	0.093	0.140
Over Cal. (s=0.2%)	0.162	0.176	0.095	0.170	0.124	0.127	0.109	0.143
Over Cal. (s=0.5%)	0.163	0.166	0.110	0.170	0.116	0.132	0.111	0.149
Over Cal. (s=1%)	0.151	0.180	0.094	0.177	0.127	0.128	0.099	0.143
Over Cal. (s=2%)	0.164	0.180	0.108	0.179	0.133	0.140	0.097	0.143
Over Cal. (s=5%)	0.168	0.172	0.108	0.169	0.125	0.130	0.096	0.144
Over Cal. (s=10%)	0.165	0.181	0.104	0.185	0.122	0.140	0.105	0.146
Over Cal. (s=20%)	0.173	0.207	0.109	0.171	0.136	0.149	0.101	0.140
Over Cal. (s=50%)	0.180	0.187	0.103	0.183	0.121	0.128	0.094	0.143
iid Cal.	0.171	0.188	0.115	0.174	0.138	0.143	0.104	0.132

Table 3.1: FDR results with overlapping calibration sets

The values of n in the columns of Table 3.1 are chosen such that, for each pair of columns, the FDR value is smaller for the left column and larger for the right column (see Figure 3.2 for a visual illustration of this phenomenon). Table 3.1 illustrates that, in the context of the present numerical experiments, the FDR estimation is not too strongly impacted by the value of s (proportion of the overlap). To assert that the observed differences between FDR estimations in each column are not significant, permutation tests [53, 128] are performed. Under $\mathcal{H}_{0n}$ hypothesis, the FDR are the same across all calibration set strategies for the calibration set cardinality n. Under $\mathcal{H}_{1n}$ there are at least two calibration strategies leading to different FDR. The FDP samples that have been used to estimate the FDR are reused. The maximal gap between sample means is used as statistic. The test is performed using the function "permutation_test" from the Python library called Scipy. The significance level is fixed at 0.05. Since multiple tests are performed over the different cardinalities, the threshold for rejecting a hypothesis is 0.00625, according Bonferroni correction. The results are display in Table 3.2. All tested hypotheses have a p-values greater than the threshold 0.00625. There are no significant difference in the resulting FDR between the different proportions of overlapping in calibration sets. This would suggest that considering overlapping calibration sets should not worsen too much the control of false positives and negatives.

n	249	250	499	500	749	750	999	1000
p-value of the test	0.300	0.0326	0.572	0.313	0.588	0.435	0.735	0.690

Table 3.2: p-values resulting from permutations test

3.3.4 Empirical Results: Assessing the FDR control

The purpose of the present section is to compute the actual FDR value when empirical p-value are used instead of true ones. The question raised here is to check whether the FDR of the full

time series is truly controlled at a prescribed level α. The empirical results must be compared with the theoretical FDR expression that has been established in Theorem 3.2.

In what follows, Section 3.3.4.1 describes the simulation design that has been considered, Section 3.3.4.2 details the criteria used for the assessment, and Section 3.3.4.3 discusses the experimental results.

3.3.4.1 Simulation design

Two scenarios have been considered to explore how much the thickness of the distribution tails can influence the results.

1. **Thin tails:**

   The reference probability distribution is $\mathcal{P}_0 = \mathcal{N}(0,1)$ for normal observations and $\mathcal{P}_1 = \delta_{\Delta_\mathcal{N}}$ for anomalies, where $\Delta_\mathcal{N} \in \mathbb{R}$ is a parameter encoding the strength of the shift. Here $\delta_{\Delta_\mathcal{N}}$ denotes the Dirac measure such that $\delta_{\Delta_\mathcal{N}}(z) = 1$ if $z = \Delta_\mathcal{N}$ and 0 otherwise. A Gaussian reference distribution and anomalies generated from a Dirac distribution in the right tail. $\Delta_\mathcal{N}$ is the size of the abnormal spike in the Gaussian distribution.

2. **Thick tails:**

   $\mathcal{P}_0 = \mathcal{T}(5)$ is a Student probability distribution with 5 degrees of freedom and $\mathcal{P}_1 = \delta_{\Delta_\mathcal{T}}$ denotes the alternative distribution of anomalies, where $\Delta_\mathcal{N} \in \mathbb{R}$ is a parameter encoding the strength of the shift.

Regarding the value of the shift strength in Scenarios 1 and 2, two values of $\Delta_\mathcal{N}$ have been considered 3.5 and 4. The values of $\Delta_\mathcal{T}$ have been chosen such that

$$\mathbb{P}_{X\sim\mathcal{N}(0,1)}(X > \Delta_\mathcal{N}) = \mathbb{P}_{X\sim\mathcal{T}(5)}(X > \Delta_t)$$

for each choice of $\Delta_\mathcal{N}$. This avoids any bias in the comparison of the detection power of the considered strategy depending on the ongoing scenario.

Different cardinalities have been considered for the calibration set following the mathematical expression

$$n \in \{\nu \cdot 10, \nu \in [\![1, 200]\!]\} \cup \{\nu' \cdot 10 - 1, \nu' \in [\![1, 200]\!]\}.$$

In particular all integers between 10 and 2 000 are explored with a step size equal to 10 as well as all integers between 9 and 1 999 with a step size of 10. This choice is justified by the particular expression of the FDR value provided by Theorem 3.2.

All the n elements of the calibration set are generated from the reference distribution that is, $\{Z_1, \ldots, Z_n\} \sim \mathcal{P}_0$. All the m observations corresponding to the tested hypotheses $\{X_1, \ldots, X_m\}$ are generated according to a mixture of $m_1 = 1$ anomalies from $\mathcal{P}_1$ and $m_0 = m - m_1$ normal observations from $\mathcal{P}_0$. Here $m = 100$ and $m_0 = 99$.

Each simulation condition has been repeated $B = 10^4$ times. For each repetition $1 \leq b \leq B$, the observations are indexed by b such that $X_{b,j} \sim \mathcal{P}_1$ for each $j \in [\![1, m_1]\!]$, and $X_{b,j} \sim \mathcal{P}_0$ for $j \in [\![m_1 + 1, m]\!]$.

3.3.4.2 Criteria for the performance assessment

In the present scenarios, anomalies are all located in the right tail of the reference probability distribution. Therefore the empirical p-value are computed according to Definition 3.1 with the scoring function $a(x) = x$. For each repetition $1 \leq b \leq B$,

$$\forall 1 \leq j \leq m, \quad \hat{p}_{b,j} = p\text{-value}(X_{b,j}, \{Z_{b,1}, \ldots, Z_{b,n}\}).$$

After computing the empirical p-values, the BH_α procedure (see Definition 3.2) is applied in such a way that, for any $1 \leq j \leq m$,

$$d_{b,j} = \mathbb{1}_{BH_\alpha(\hat{p}_{b,1},\ldots,\hat{p}_{b,m})}(j),$$

where $\mathbb{1}_I$ denotes the indicator function of the index set I. The FDP value of the sequence from 1 to m is computed from the knowledge of the true label of the observations as "normal" or "anomaly". For each repetition $1 \leq b \leq B$,

$$(FDP_1^m)_b = \frac{\sum_{j=m_1+1}^{m} d_{b,j}}{\sum_{j=1}^{m} d_{b,j}}.$$

The results obtained after the B repetitions are averaged within the FDR estimate of the sequence from 1 to m as

$$FDR_1^m = \frac{1}{B}\sum_{b=1}^{B}(FDP_1^m)_b.$$

The FNR value of the sequence from 1 to m (Equation 3.2.2) is estimated by

$$(FNP_1^m)_b = \frac{1}{m - m_0}\sum_{j=m_0+1}^{m} d_{b,j}, \quad \text{and} \quad FNR_1^m = \frac{1}{B}\sum_{b=1}^{B}(FNP_1^m)_b.$$

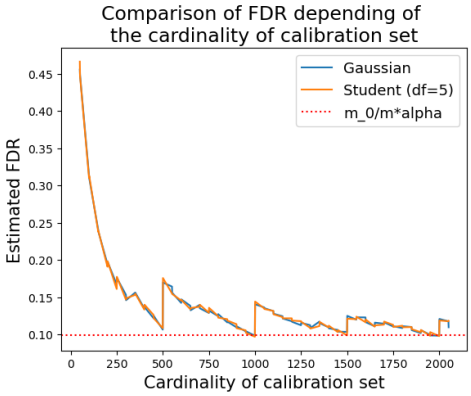
3.3.4.3 Results and analysis

Figure 3.3 displays the FDR value (left panel) and the FNR value (right panel) as a function of the calibration set cardinality for the two scenarios (Gaussian and Student) described in Section 3.3.4.1. The blue (respectively orange) curve corresponds to the Gaussian (resp. Student) reference distribution. The horizontal line is the prescribed level $\alpha = 0.1$ at which FDR should be controlled with true p-values (Theorem 3.1). Figures 3.3a and 3.3b are obtained with $\Delta_{\mathcal{N}} = 4$, while Figures 3.3c and 3.3d result from $\Delta_{\mathcal{N}} = 3.5$.

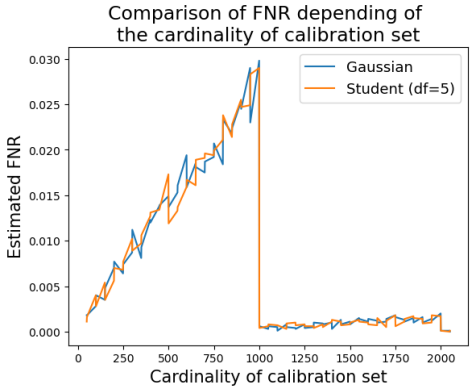
According to these plots, the behavior of both FDR and FNR does not exhibit any strong dependence with respect to the reference probability distribution. The results are very close for both Gaussian and Student distributions.

As illustrated by Figures 3.3a and 3.3c, the FDR control at the prescribed level is achieved for particular values of the calibration set cardinality. These values coincide with the ones exhibited by Theorem 3.1, which are multiples of $\alpha/m = 10^3$ (up to a downward shift by 1).

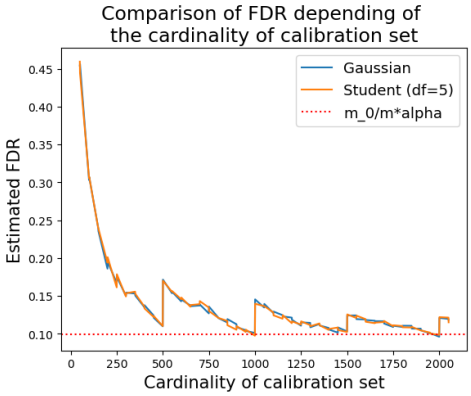
A striking remark is that the FNR curve sharply increases from 1 to $n = 999$. This reflects that although the FDR value becomes (close to) optimal as n increases from 1 to $n = 999$, the proportion of false negatives simultaneously increases leading to a suboptimal statistical



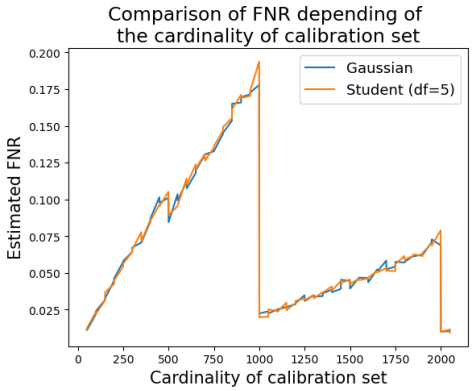
(a) FDR depending on n with 4 sigmas anomalies



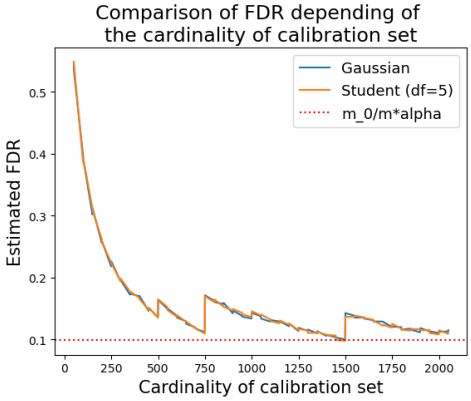
(b) FNR depending on n with 4 sigmas anomalies



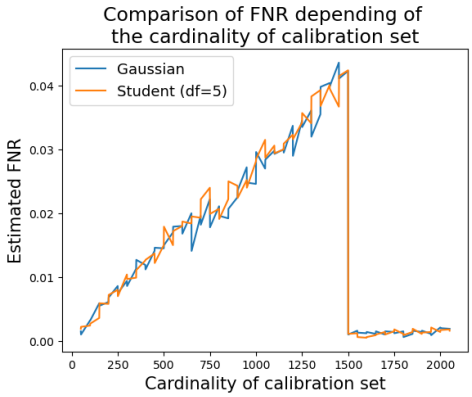
(c) FDR depending on n with 3.5 sigmas anomalies



(d) FNR depending on n with 3.5 sigmas anomalies



(e) FDR depending on n with 4 sigmas anomalies and $m = 150$



(f) FNR depending on n with 4 sigmas anomalies and $m = 150$

Figure 3.3: Effect of calibration set cardinality and abnormality score on the FDR control and the FNR

performance (because of too many false negatives). Fortunately a larger cardinality n of the calibration set, for instance $n = 1999$, would greatly improve the results at the price of a larger calibration set, which also increases the computational cost.

Consistently with what is established in Theorem 3.1, the FDR value does not depend on the strength of the distribution shift Δ as illustrated by Figures 3.3a and 3.3c. As long as FDR is concerned (which is an expectation), the shift plays no role. Let us mention that focusing of the expectation does not say anything about the probability distribution of FDP, which can be influenced by the shift strength. By contrast, the comparison of Figures 3.3b and 3.3d clearly shows the impact of the shift strength on the FNR value. As the shift strength becomes lower, anomalies are more difficult to be detected which inflates the FNR value.

The best cardinality n of the calibration set depends on the number m of tested hypotheses according to Theorem 3.1. For instance, Figure 3.3a shows the value $n = 999 = 100/0.1 - 1$ as a good candidate since it achieves the desired FDR control while reducing both the number of false negatives and the computation cost. By contrast, Figure 3.3e rather exhibits the value $n = 1499 = 150/0.1 - 1$ as the smallest n allowing a perfect FDR control and a small number of false negatives.

Figure 3.3a shows other intermediate values calibration set cardinalities yielding the FDR control. For instance $n = 1499$ (between $n = 999$ and $n = 1999$) is predicted by Theorem 3.1. However complementary experiments (summarized by Figure 3.12 in Section 3.8.2) illustrate that these intermediate values of n allowing the FDR control actually depend on the number of anomalies m_1. Their existence can be explained by the distribution of the number of detections. For example, Figure 3.12d shows a high probability of detecting 3 anomalies. Assuming there exists $k^* \in [\![1, m]\!]$ such that $\mathbb{P}(R(i) = k^*) \approx 1$, Theorem 3.2 justifies that

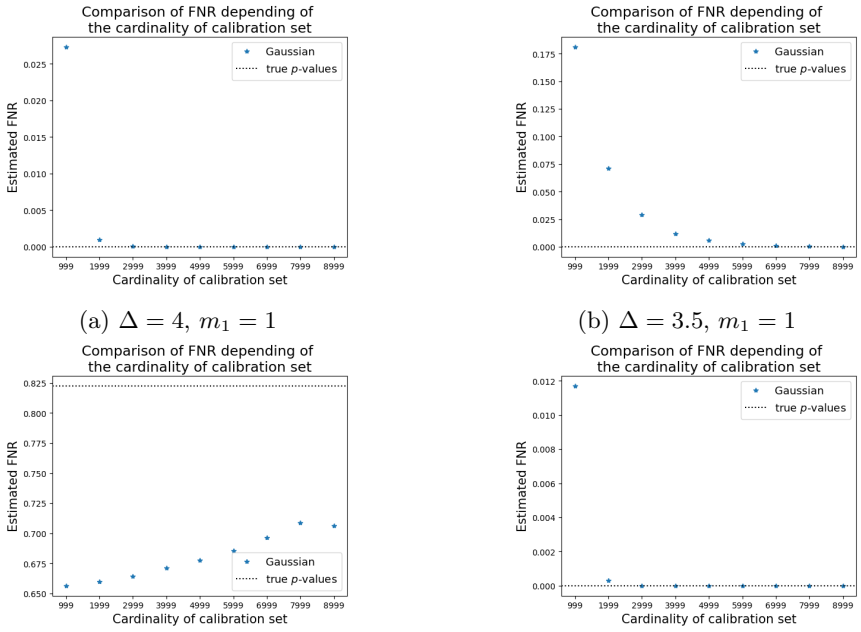
$$\begin{aligned} FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) &= \frac{n}{n+1} \cdot \alpha \frac{m_0}{m} + \frac{m_0}{n+1} \sum_{k=1}^{m} \frac{(1 - q_{n,k})}{k} \mathbb{P}(R(i) = k) \\ &\approx \frac{n}{n+1} \cdot \alpha \frac{m_0}{m} + \frac{m_0}{n+1} \frac{(1 - q_{n,k^*})}{k^*}. \end{aligned}$$

Then the proof detailed in Section 3.7.2 yields that $1 - q_{n,k^*} = \frac{\alpha}{m(n+1)}$ can be reached for all $\nu \geq 1$, such that $n = \nu \frac{m}{\alpha k^*} - 1$. This allows to conclude that $FDR_1^m(\hat{\varepsilon}_{BH_\alpha}, \hat{p}) \approx \frac{m_0 \alpha}{m}$.

3.3.4.4 How to choose the right cardinality of the calibration set?

Intuitively an optimal choice of the cardinality n of the calibration set should enable the FDR control while minimizing the number of false negatives and avoiding any excessive computation time. To achieve this objective, the first part of Corollary 3.1 explains that n must be chosen from the set $\mathcal{N} = \{\nu m/\alpha - 1, \quad \nu \geq 1\}$. Using the simulation scenarios described in Section 3.3.4.1, the aim is to visualize the relationship between the calibration set cardinality and FNR when $n \in \mathcal{N}$. The results are summarized by Figure 3.4 where the FNR value is displayed versus n. For all the considered scenarios (Fig. 3.4a, 3.4b, 3.4c, 3.4d), the FNR value converges to the value reached with true p-values (horizontal dashed line) as n grows. From Figures 3.4a and 3.4b, the convergence speed depends on the "difficulty" of the problem. Ideally, the the smallest n that ensure any desired FNR level would be chosen.

In practice, the lack of labeled observations prevents us from computing the actual FNR value, making the choice of the optimal value of n highly challenging. To tackle this challenge our suggestion is to choose the largest possible value of n that does not exceed the computation



(a) $\Delta = 4$, $m_1 = 1$

(b) $\Delta = 3.5$, $m_1 = 1$

(c) $\Delta = 3$, $m_1 = 1$

(d) $\Delta = 3$, $m_1 = 5$

Figure 3.4: FNR as a function of the calibration set cardinality constrained to belong to $\mathcal{N}$.

time limit. Doing that would output a value of n minimizing the FNR criterion while meeting the computational constraints. However following this suggestion does not prevent us from computational drawbacks as illustrated by Figure 3.4a where the FNR optimal value is reached for $n = 3999$ while choosing a larger n does not bring any gain (but still increases the computational costs).

3.4 Global FDR control over the whole time series

While working with streaming time series data, the anomaly detection problem requires to control the FDR value of the full time series to make sure that the global false alarm rate (FDR) remains under control at the end of the iterative process. The final criterion that is to be controlled is then the global FDR criterion given by

$$FDR_1^{\infty}(\hat{\varepsilon}, \hat{p}),$$

where $\hat{\varepsilon} = (\hat{\varepsilon}_t)_{t \geq 1}$ denotes a sequence of data-driven thresholds, and $\hat{p}$ stands for a sequence of empirical p-values (see Section 3.3 for further details). By contrast with this global objective, anomaly detection nevertheless requires making decision at each time step that is, for each new observation, without knowing what the next ones look like. This justifies the need for another (local) criterion that will be used to make a decision at each iteration, leading to the sequence of data-driven thresholds $\hat{\varepsilon} = (\hat{\varepsilon}_t)_{t \geq 1}$. One additional difficulty results from the connection one needs to create between this local criterion and the (global) FDR of the full time series.

To this end, Section 3.4.1 starts by showing that controlling the FDR criterion for subseries of the full time series does not provide the desired *global* FDR control. Here "global" means "on

the full time series" by contrast with the *local* FDR control, corresponding to controlling FDR for a strict subseries of the full one. Then Section 3.4.2 explores the connection between FDR for the full time series and the so-called modified-FDR (mFDR) for subseries. In particular, it turns out that controlling the mFDR value for all subseries of a given length m yields the desired FDR control for the full time series. Section 3.4.3 then explains how the classical BH-procedure can be modified to get the mFDR control for subseries of length m, while Section 3.4.5 illustrates the practical behavior of the considered strategies on simulation experiments.

3.4.1 Local and global FDR controls are not equivalent

Let us consider a time series partitioned into 4 subseries as illustrated in Figure 3.5. The normal points are displayed in black and the anomalies in white. The surrounded points are those that have been detected as anomalies by the procedure.

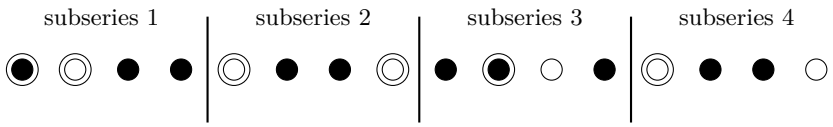


Figure 3.5: Illustration of anomaly detection in subseries.

When computing the number of rejections, false positives and the False positive rate for each subseries, it comes

- Subseries 1 : 2 rejections, 1 false positive. $FDP_1^4 = 0.5$
- Subseries 2 : 2 rejections, 0 false positive. $FDP_5^9 = 0$
- Subseries 3 : 1 rejections, 1 false positive. $FDP_{10}^{14} = 1$
- Subseries 4 : 1 rejections, 0 false positive. $FDP_{15}^{19} = 0$

If one assumes that the same probability distribution has generated the observations within each subseries, the estimated (local) FDR can be defined as average of the successive FDP values for each subseries that is, $FDR_1^4 = 0.375$. Let us notice that the notation emphasizes that this FDR value is the average over subseries of respective length $m = 4$. If one reproduces the same reasoning for the full time series, it comes: 6 rejections, two false positives, so that $FDP_1^{16} = 1/3 = 0.333$. This example highlights that the FDP of the full time series is not equal to that of smaller subseries. This phenomenon gives some intuition on possible reasons why applying the classical BH-procedure on local windows of length m (subseries) does control the FDR criterion for the individual subseries, but does not yield the desired global FDR control for the full time series. This intuition is confirmed by the boxplots of Figure 3.6, where BH_α has been applied on subseries of length $m = 100$. The left boxplot shows that BH_α provides the desired control at level $\alpha = 10\%$ for each individual subseries of length m. However the right boxplot clearly departs from α, meaning that the actual FDR value for the full times series of length 1 000 is strongly larger than α (more than 20% on average) leading to more false positives at the level of the full time series. The boxplots represent the quantile of FDP over 100 repetitions.

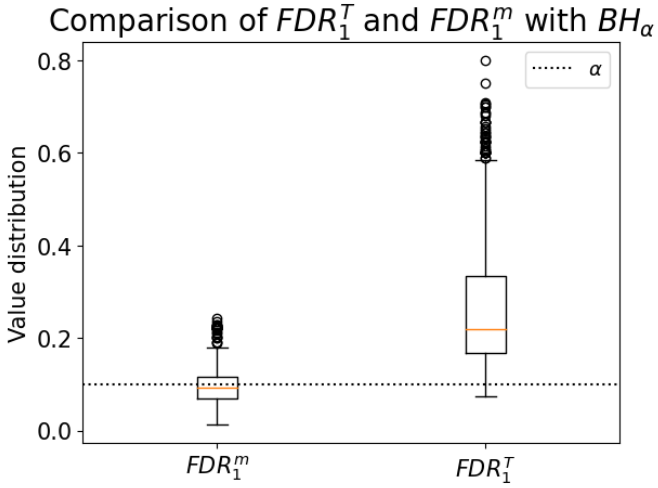


Figure 3.6: Comparison of the calculation of the FDR computed locally on a subseries and the FDR computed globally on the whole time series with Benjamini-Hochberg procedure applied on a subseries. This result is obtained by cutting a series of cardinality 1000 into 10 subseries of cardinality 100. Then the Benjamini-Hochberg procedure is applied on each subseries.

3.4.2 mFDR can help in controlling the FDR of the full time series

3.4.2.1 Mixture model and time series

At Section 3.2.2 anomalies was described using a fix set of true null hypothesis like in classical multiple testing framework. In order to establish our global control from the local control, the classical assumption is made that the true labels are derived from a Bernoulli distribution. The previous model can therefore be considered as a realization of the mixture model. In this section one assumes that the time series is generated from a mixture process between a reference distribution $\mathcal{P}_0$ and an alternative distribution $\mathcal{P}_1$. The anomaly positions are supposed to be independent and generated by a Bernoulli distribution. This is a common assumption usually in the literature [80, 155] for simplification purposes.

Definition 3.4 (Time series process with anomalies)**.** *Let $\pi \in [0,1]$ be the anomaly proportion and $\mathcal{P}_0$ and $\mathcal{P}_1$ be two probability distributions on the observation domain $\mathcal{X}$. $\mathcal{P}_0$ is the reference distribution and $\mathcal{P}_1$ denotes the alternative distribution. The generation process of a time series containing anomalies $(A_t)_{t\geq 0}$ is given, for every $t \geq 0$, by*

- *$A_t \sim B(\pi)$ (Bernoulli distribution)*
- *if $A_t = 0$, then $X_t \sim \mathcal{P}_0$.*
- *if $A_t = 1$, then $X_t \sim \mathcal{P}_1$.*

Moreover given the above scheme, $(X_t)_{t\geq 0}$ is a random process with independent and identically distributed random variables $X_t \sim (1-\pi)\mathcal{P}_0 + \pi\mathcal{P}_1$.

This definition details the way anomalies are generated. In particular it assumes that anomalies are independent from each other. Let us mention that this does not prevent us from observing a sequence of successive anomalies along the time series. However this scheme substantially dif-

fers from the case analyzed by [68] where specific patterns with successive anomalies are looked for.

3.4.2.2 Preliminary discussion: Disjoint and Overlapping subseries

In the context of online anomaly detection, the main focus in what follows is put on two situations where the data-driven thresholds $(\hat{\varepsilon}_t)_{t\geq 0}$ can be defined from a set of m empirical p-values: (i) the *disjoint* case where disjoint subseries of length m are successively considered, and (ii) the *overlapping* case where the subseries (of length m) successively considered share $m-1$ common observations at each step. Note that the notion of overlapping sub-series, used in the threshold calculation, is distinct from that of overlapping calibration sets, used in the calculation of p-values.

Let us start with a subseries of length m where each observation is summarized by its corresponding empirical p-value, and let us assume that there exists a function $f_m : [0,1]^m \to [0,1]$ that is mapping a set of m empirical p-values onto a real-valued random variable. This random variable corresponds to the data-driven threshold that is applied to the subseries of length m to detect potential anomalies. This function f_m is called the *local threshold* function since it outputs a threshold which applies to a subseries of length m.

Given the above notations, the threshold sequences $\hat{\varepsilon}_d = (\hat{\varepsilon}_{d,t})_t$ and $\hat{\varepsilon}_o = (\hat{\varepsilon}_{o,t})_t$ can be defined as follows.

- **Disjoint subseries:** $\hat{\varepsilon}_d : t \mapsto \hat{\varepsilon}_{d,t}$ is given by

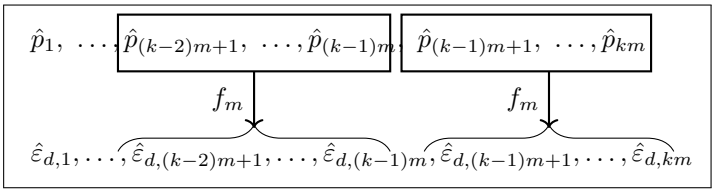
$$\forall k \geq 0, \quad \forall t \in [\![km+1, (k+1)m]\!], \quad \hat{\varepsilon}_{d,t} = f_m(\hat{p}_{km+1}, \ldots, \hat{p}_{(k+1)m}) \tag{3.10}$$

- **Overlapping subseries:** $\hat{\varepsilon}_o : t \mapsto \hat{\varepsilon}_{o,t}$ is given by

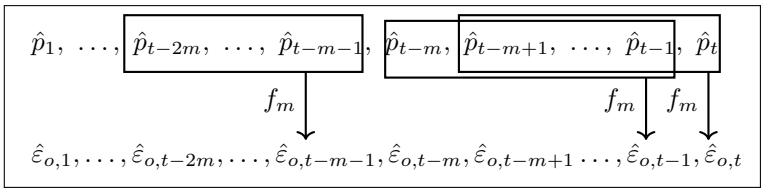
$$\forall t \geq m, \qquad \hat{\varepsilon}_{o,t} = f_m(\hat{p}_{t-m+1}, \ldots, \hat{p}_t). \tag{3.11}$$

Figure 3.7 illustrates these two situations. In Figure 3.7a, the full time series is split into small disjoint subseries of length m. f_m is applied to each such subseries and the threshold is the same for all observations within a given subseries. Figure 3.7b displays the situation where overlapping subseries are successively considered. Because two successive subseries differ from each other by two observations, the thresholds are different at each time step unlike the disjoint case. Furthermore the sequences $\hat{\varepsilon}_d$ and $\hat{\varepsilon}_o$ do not enjoy the same dependence properties. Figure 3.7a illustrates that all thresholds $\hat{\varepsilon}_{d,(k-1)m+1}, \ldots, \hat{\varepsilon}_{d,km}$ are computed by applying f_m to the same subseries $\hat{p}_{(k-1)m+1}, \ldots, \hat{p}_{km}$. Therefore only thresholds computed from different subseries are independent, while all thresholds from the same subseries are equal. In other words, $\hat{\varepsilon}_{d,t_1}$ and $\hat{\varepsilon}_{d,t_2}$ are independent if and only if t_1 and t_2 belong to different subseries that is, $\lfloor t_1/m \rfloor \neq \lfloor t_2/m \rfloor$, where $\lfloor \cdot \rfloor$ denotes the integer part. By contrast Figure 3.7b shows that the variables $\hat{\varepsilon}_{o,t}$ and $\hat{\varepsilon}_{o,t-1}$ are dependent because they share $m-1$ common observations. But all of them are still different and, for each t, $\hat{\varepsilon}_{o,t}$ is independent from $\hat{\varepsilon}_{o,t-m-1}$. This can be reformulated as $\hat{\varepsilon}_{o,t_1}$, $\hat{\varepsilon}_{o,t_2}$ are independent if and only if $|t_1 - t_2| > m$.

In the present online anomaly detection context, considering the overlapping case sounds more convenient since the detection threshold can be updated at each time step (as soon as a new observation has been given), which makes the anomaly detector more versatile. However for technical reasons, next Theorem 3.3 still focuses disjoint subseries as a means to introduce



(a) Illustration of disjoint windows.



(b) Illustration of Overlapping sliding windows.

Figure 3.7: Comparison of disjoint window and overlapping window for the threshold function.

important notions without introducing too many technicalities, while Theorem 3.4 extends the previous results to the more realistic case of overlapping subseries.

3.4.2.3 FDR control with disjoint subseries

As illustrated in Section 3.4.1, controlling FDR on each subseries of length m (locally) is not equivalent to controlling FDR (globally) on the full time series. However in online anomaly detection, a decision has to be made at each time step regarding the potential anomalous status of each new observation. (This is a typical instance of a local decision since at step t, the decision making process ignores what will be observed at the next step.) This requires a criterion to be controlled locally (on subseries) in such a way that the resulting global FDR value (the one of the full time series) can be proved to be controlled at the desired level α.

This requirement for a local criterion justifies the introduction of the modified FDR criterion, denoted by mFDR [172, 57], which is defined as follows.

Definition 3.5 (mFDR)**.** *With the previous notations, the mFDR expression of the subseries from $t-m+1$ to t is given by*

$$mFDR^t_{t-m+1}(\hat{\varepsilon}, \hat{p}) = \frac{\mathbb{E}\left[\sum_{u \in \mathcal{H}_0, t-m+1 \le u \le t} \mathbb{1}[\hat{p}_u \le \hat{\varepsilon}_u]\right]}{\mathbb{E}\left[\sum_{u=t-m+1}^{t} \mathbb{1}[\hat{p}_u \le \hat{\varepsilon}_u]\right]},$$

where $\hat{\varepsilon} = (\hat{\varepsilon}_u)_{t-m+1 \le u \le t}$ denotes a sequence of thresholds, $\hat{p}$ is a sequence of empirical p-values evaluated at each observation of the subseries from $t-m+1$ to t.

Mathematically the difference between the mFDR and the FDR is that the expectation is no longer on the ratio but independently on the numerator and the denominator. The main interest for mFDR is clarified by Theorem 3.3, which establishes its connection to FDR. To be more specific, the control of the latter at the α level provides a global control of the FDR at the same level under simple conditions.

Theorem 3.3 (Global FDR control with disjoint subseries)**.** *Assume that $\hat{\varepsilon}_d : t \mapsto \hat{\varepsilon}_{d,t}$ is given by $\hat{\varepsilon}_{d,t} = f_m(\hat{p}_{km+1}, \ldots, \hat{p}_{(k+1)m})$, for any $t \in [\![km+1, (k+1)m]\!]$ $(k \ge 0)$ and any integer $m \ge 1$*

(see Eq. (3.10)). Let us also assume that the p-value random process $\hat{p} = (\hat{p}_t)_{t\geq 1}$ *follows the scheme detailed in Definition 3.4. Then, the global FDR value of the full (infinite) time series is equal to the local mFDR value of the any subseries of length* m *from* $t = km + 1, k \in \mathbb{N}^*$*. More precisely,*

$$FDR_1^{\infty}(\hat{\varepsilon}_d, \hat{p}) = mFDR_{km+1}^{(k+1)m}(\hat{\varepsilon}_d, \hat{p}) = mFDR_{km+1}^{(k+1)m}(\hat{\varepsilon}_d, \hat{p}).$$

Since the full time series is assumed to be infinite, Theorem 3.3 is an asymptotic result. It gives rise to a strategy for controlling the (asymptotic) FDR criterion at level α by means of successive local controls of mFDR on small subseries of length m. According to the asymptotic nature of Theorem 3.3, there is no particular constraint on the integer m. However when dealing within time series of a finite length T, the Theorem 3.3 proof suggests that choosing an m "not too large" would be better since then, $k = T/m$ would take large values making the LLN applicable (see for instance Eq. (3.12)). Actually in the online anomaly detection context, practitioners only have a limited freedom regarding the choice of m. Therefore, for a given fixed m, the control of the FDR value of the full time series given by Theorem 3.3 will be all the more accurate as T will be large. Fortunately this is not a limitation in the online anomaly detection context. The main limitation of Theorem 3.3 lies in the use of disjoint subseries, which sounds somewhat restrictive (at least from a practical perspective). This limitation will be overcome by next Theorem 3.4.

Proof of Theorem 3.3. Let $k \geq 1$ denote an integer and $T = mk$. Then, the FDP definition and the A_t variables introduced in Definition 3.4 justify that

$$FDP_1^T(\hat{\varepsilon}_d, \hat{p}) = \frac{FP_1^T(\hat{\varepsilon}_d, \hat{p})}{R_1^T(\hat{\varepsilon}_d, \hat{p})} = \frac{\sum_{t=1}^T \mathbb{1}[\hat{p}_t < \hat{\varepsilon}_{d,t}](1 - A_t)}{\sum_{t=1}^T \mathbb{1}[\hat{p}_t < \hat{\varepsilon}_{d,t}]},$$

where $R_1^T(\hat{\varepsilon}_d, \hat{p})$ and $FP_1^T(\hat{\varepsilon}_d, \hat{p})$ respectively denote the number of rejections (resp. false positives) at the threshold $\hat{\varepsilon}_d$ for the subseries $\hat{p}$.

Using the partitioning into k subseries of length m, its first comes that $FP_1^T(\hat{\varepsilon}_d, \hat{p}) = \sum_{i=1}^k FP_{im-1}^{(i+1)m}(\hat{\varepsilon}_d, \hat{p})$. It is also noticeable that the k random variables $\{FP_{im-1}^{(i+1)m}(\hat{\varepsilon}_d, \hat{p})\}_{1\leq i\leq k}$ are independent and identically distributed since the thresholds $\hat{\varepsilon}_{d,_i}$ remain unchanged within each subseries, they are identically distributed from one block to another, and the empirical p-values from different blocks are independent and identically distributed as well. Therefore the random variables $(FP_{im-1}^{(i+1)m}(\hat{\varepsilon}_d, \hat{p}))_{1\leq i\leq k}$ are independent and identically distributed, which implies (Law of Large Numbers theorem) that, almost surely,

$$\lim_k \frac{1}{k} \sum_{i=1}^k FP_{im-1}^{(i+1)m}(\hat{\varepsilon}_d, \hat{p}) = \mathbb{E}[FP_1^m(\hat{\varepsilon}_d, \hat{p})], \tag{3.12}$$

where the expectation is taken over all sources of randomness. (Here it is implicitly assumed that T can go to $+\infty$.) Repeating the argument for$R_1^T(\hat{\varepsilon}_d, \hat{p})$, it also comes that

$$\mathbb{E}[R_1^m(\hat{\varepsilon}_d, \hat{p})] = \lim_k \frac{1}{k} \sum_{i=1}^k R_{im-1}^{(i+1)m}(\hat{\varepsilon}_d, \hat{p}), \qquad a.s..$$

The conclusion then results from noticing that

$$mFDR_1^m(\hat{\varepsilon}_d, \hat{p}) = \frac{E[FP_1^m(\hat{\varepsilon}_d, \hat{p})]}{E[R_1^m(\hat{\varepsilon}_d, \hat{p})]} = \lim_k \frac{\sum_{i=1}^k FP_{im-1}^{(i+1)m}(\hat{\varepsilon}_d, \hat{p})}{\sum_{i=1}^k R_{im-1}^{(i+1)m}(\hat{\varepsilon}_d, \hat{p})} = \lim_T FDP_1^T(\hat{\varepsilon}_d, \hat{p}).$$

The proof is completed by calculating the expectation on each side of the sign equals. □

In fact, this proves stronger than just the control of the FDR_1^∞ at the $mFDR_1^m$ level, since it is shown that it is actually each FDP_1^∞ that is controlled at the $mFDR_1^m$ level. In the rest of this chapter, only the FDR control will be discussed, as it will be seen that this is what can be obtained for time series of finite length.

3.4.2.4 FDR control with overlapping windows

Theorem 3.4 is a generalization of Theorem 3.3. Unlike previous Theorem 3.3, following Theorem 3.4 establishes a similar control of the global FDR criterion on the full time series by means of successive local controls of the mFDR criterion on subseries that are allowed to overlap each other. This is closer to the practical situation arising in online anomaly detection where one new observations is collected at each time step, inducing a shift by one of the set of observations for which a decision has to be made.

Theorem 3.4 (Global FDP control using local threshold)**.** *Let $\hat{p} = (\hat{p}_t)_{t\geq 1}$ be the p-value random process, for a time series that follows the scheme detailed in Definition 3.4. Let $\hat{\mathbf{P}} = (\hat{\mathbf{P}}_{t,k})_{t\geq 1, 1\leq k\leq m}$ a process of p-values vectors, such that $\hat{\mathbf{P}}_{t,m} = \hat{p}_t$. Assume that $\hat{\varepsilon}_o : t \mapsto \varepsilon_{o,t}$ is given by $\hat{\varepsilon}_{o,t} = f_m(\hat{\mathbf{P}}_t)$, for any $t \geq 1$, with $f_m : [0,1]^m \to [0,1]$ permutation invariant. Let us also assume that there exists n such that $|t_1 - t_2| > n$ implies that $\hat{p}_{t_1}$ and $\hat{p}_{t_2}$ are independent, and $|t_1 - t_2| > n+m$ implies that $\hat{P}_{t_1}$ and $\hat{P}_{t_2}$ are independent. For all t, $\hat{\mathbf{P}}_t = (\hat{\mathbf{P}}_{t,1}, \ldots, \hat{\mathbf{P}}_{t,m})$ is exchangeable. Then, the global FDR (and FDP) value of the full (infinite) time series is equal to the local mFDR value of the any subseries of length m computed at time $t \in \mathbb{N}^*$. More specifically*

$$FDP_1^\infty(\hat{\varepsilon}_o, \hat{p}) = FDR_1^\infty(\hat{\varepsilon}_o, \hat{p}) = mFDR_{t-m+1}^t(\hat{\varepsilon}_o, \hat{\mathbf{P}}_t) = mFDR_1^m(\hat{\varepsilon}_o, \hat{\mathbf{P}}_m)$$

Theorem 3.4 gives a similar result to the one of Theorem 3.3 but in a more realistic framework corresponding to the real time anomaly detection context. In particular the main improvement lies in that a threshold can be recomputed at each time step from a (shifted) subseries of length m. An important consequence is that the desired control for the FDR of the full (infinite) time series at level α can be achieved provided one can control the successive mFDR of all (shifted) subseries of length m at level α. This point is not obvious at all and constitutes the main concern of Section 3.4.3 where a new multiple testing procedure is designed to yield the desired control of the mFDR criterion. The main limitation of Theorem 3.4 is the requirement that f_m has to be permutation invariant. Let us emphasize that this property holds true with the BH-procedure for instance.

This general result encapsulates several cases of threshold computation, including:

- Threshold computed on disjoint subseries as Eq. 3.10 with iid p-values:

  Suppose the a sequence of p-values $(\hat{p}_t)_{t\geq 1}$, which is computed from iid calibration set $\hat{p}_t = \hat{p}_e(X_t, \{, Z_{1,t}, \ldots, Z_{n,t})$ and that the thresholds are computed from disjoint p-values

sets:

$$\forall k \geq 0, \quad \forall t \in [\![km+1, (k+1)m]\!], \quad \hat{\mathbf{P}}_t = (\hat{p}_{km+1}, \ldots, \hat{p}_{(k+1)m}) \tag{3.13}$$

- Threshold computed on Overlapping subseries as Eq.3.11 with p-values computed using single calibration set at each t: Suppose that are each t the m p-values are computed using the same calibration set:

$$\forall t \geq m, \forall i \in [\![1, m]\!] \qquad \hat{p}_{t-i+1,t} = \hat{p}_e(X_{t-i+1}, \{X_{t-n-m}, \ldots, X_{t-m}\}) \tag{3.14}$$

  Then using $\hat{p}_t = \hat{p}_{t,t}$ and

$$\forall t \geq m, \qquad \hat{\mathbf{P}}_t = (\hat{p}_{t-m+1,t}, \ldots, \hat{p}_{t,t}). \tag{3.15}$$

- Threshold computed on Overlapping subseries as Eq.3.11 with p-values computed using overlapping calibration sets: Suppose that are each t the p-value is computed using the n previous calibration set:

$$\forall t \geq n, \qquad \hat{p}_t = \hat{p}_e(X_t, \{X_{t-n-1}, \ldots, X_{t-1}\}) \tag{3.16}$$

  And the threshold is computed using the m last p-values

$$\forall t \geq m, \qquad \hat{\mathbf{P}}_t = (\hat{p}_{t-m+1}, \ldots, \hat{p}_t). \tag{3.17}$$

Let us also mention that the empirical p-values for instance computed as $\hat{p}_t = \hat{p}_e(X_t, \{X_{t-n}, \ldots, X_{t-1}\})$ actually satisfy the requirements of Theorem 3.4 regarding the independence and the stationarity.

Proof of Theorem 3.4. Let us start with the FDP expression for a time series of length T.

$$\begin{aligned} FDP_{t=1}^{T} &= \frac{\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < \hat{\varepsilon}_{o,t}](1 - A_t)}{\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < \hat{\varepsilon}_{o,t}]} \\ &= \frac{\frac{1}{T}\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < f_m(\hat{\mathbf{P}}_t)](1 - A_t)}{\frac{1}{T}\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < f_m(\hat{\mathbf{P}}_t)]}, \end{aligned}$$

The decision process $(\mathbb{1}[\hat{p}_t < \hat{\varepsilon}_{o,t}])_t$ and the false positives process $(\mathbb{1}[\hat{p}_t < \hat{\varepsilon}_{o,t}]A_t)_t$ are not independent, therefore it is not possible to use the Law of Large Numbers directly. The alternative strategy consists first in splitting the numerator and denominator into several disjoint subseries corresponding to independent and identically distributed processes. Then partitioning the times series of length $T = T'(n+m)$ into T' subseries, each of length $n+m$, it results that

$$\begin{aligned} &\frac{1}{T}\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < f_m(\hat{\mathbf{P}}_t)](1 - A_t) \\ &= \frac{1}{n+m}\sum_{k=1}^{n+m} \left(\frac{1}{T'}\sum_{t=0}^{T'-1} \mathbb{1}[\hat{p}_{t(n+m)+k} < f_m(\hat{\mathbf{P}}_{t(n+m)+k})](1 - A_{t(n+m)+k}) \right). \end{aligned} \tag{3.18}$$

Interestingly for each k from 1 to $m+n$, the summands within the brackets do all belong to different subseries, which makes the sum over t a sum of independent and identically distributed

random variables.

It results that, for each $1 \leq k \leq n+m$, the average within the brackets is converging to its expectation by the LLN theorem.

Since the limit of a (finite) sum is equal to the sum of the limits, the average in Eq. (3.18) is converging and

$$\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < f_m(\hat{\mathbf{P}}_t)](1-A_t) = \sum_{k=1}^{m} \mathbb{E}\left[\mathbb{1}[\hat{p}_k < f_m(\hat{\mathbf{P}}_k)](1-A_k)\right] \text{ a.s.} \tag{3.19}$$

$$= m\mathbb{E}\left[\mathbb{1}[\hat{p}_m < f_m(\hat{\mathbf{P}}_m)](1-A_m)\right] \text{ a.s..} \tag{3.20}$$

Furthermore,

$$FP_1^m(\hat{\varepsilon}_o, \hat{\mathbf{P}}_m) = \sum_{k=1}^{m} \mathbb{1}[\hat{\mathbf{P}}_{m,k} < f_m(\hat{\mathbf{P}}_m)](1-A_k)$$

Using the exchangeability of $\hat{\mathbf{P}}_m$ and the permutation invariance of f_m is comes:

$$\mathbb{E}FP_1^m(\hat{\varepsilon}_o, \hat{\mathbf{P}}_1) = m \times \mathbb{E}[\mathbb{1}[\hat{\mathbf{P}}_{m,m} < f_m(\hat{\mathbf{P}}_m)](1-A_m)]$$

Finally with $\hat{\mathbf{P}}_{m,m} = \hat{p}_m$ and Eq. 3.20 it gives:

$$\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < f_m(\hat{\mathbf{P}}_m)](1-A_t) = \mathbb{E}FP_1^m(\hat{\varepsilon}_o, \hat{\mathbf{P}}_m)$$

Then after applying the same reasoning on the denominator, it gives:

$$FDP_1^\infty(\hat{\varepsilon}_o, \hat{p}) = \frac{\mathbb{E}FP_1^m(\hat{\varepsilon}_o, \hat{\mathbf{P}}_m)}{\mathbb{E}R_1^m(\hat{\varepsilon}_o, \hat{\mathbf{P}}_m)}$$

□

3.4.3 Modified BH-procedure and mFDR control

As shown in Section 3.4.2, controlling the FDR value of the full time series is possible. The strategy then consists in first controlling the mFDR criterion of all successive subseries of length m along the full time series at level α. The main challenge addressed in the present section is to design a new multiple testing procedure that controls the local mFDR criterion at a prescribed level α.

In Section 3.4.3.1, it is proved that applying the classical BH-procedure on a time series of length m does not yield the control of mFDR at level α. However the proof of this result gives rise to a strategy for modifying the classical BH-procedure (Section 3.4.4.3) in a such a way that applying the so-called modified BH-procedure provides the desired mFDR control at level α, under some conditions.

3.4.3.1 mFDR control with the BH-procedure

Next Proposition 3.3 establishes the actual mFDR level achieved by the BH-procedure.

Proposition 3.3. *Let $(X_i)_1^m$ satisfy the requirements detailed by Definition 3.4, with an abnormality proportion π. m_0 is the random variable representing the number of data points generated by $\mathcal{P}_0$. Let $p = (p_1, \dots, p_m)$ be the associated p-values. Let α belong to $[0, 1]$. Suppose there are integers ν and n such that $n = \nu\frac{m}{\alpha} - 1$. Let $R^m_{\alpha,1}$ be the random variable representing the number of rejections when BH_α is applied to p. Suppose one of the following three statements is true:*

1. *$(p_1, \dots, p_m)$ are true p-values that is, for any $1 \leq i \leq m$, $p_i = \mathbb{P}_{X\sim\mathcal{P}_0}(a(X) \geq a(X_i))$.*
2. *$(p_1, \dots, p_m)$ are empirical p-values with independent calibration sets of cardinality n, $p_i = \hat{p}_e(a(X_i), \{a(Z_{i,1}), \dots, a(Z_{i,n}))$. With $Z_{i,j}$ are iid random variables generated by $\mathcal{P}_0$.*

   In these two cases, let choose i in $\mathcal{H}_0$, the sequence $(p'_j)_{1\leq j\leq m}$ is defined by $p'_i = 0$ and $p'_j = p_j$. Let $R^{*,m}_{\alpha,1}$ define the number of rejections when applying BH_α on p'
3. *$(p_1, \dots, p_m)$ are empirical p-values with a unique calibration set of cardinality n, $p_i = \hat{p}_e(a(X_i), \{a(Z_1), \dots, a(Z_n))$. With Z_j are iid random variables generated by $\mathcal{P}_0$.*

   Let choose i in $\mathcal{H}_0$, the sequence $(p'_j)_{1\leq j\leq m}$ is defined by $p'_i = 0$ and $p'_j = p_j - \frac{1}{n}\mathbb{1}[p_j < p_i]$. Let $R^{*,m}_{\alpha,1}$ define the number of rejections when applying BH_α on p'.

Then, applying BH_α to the p-values $(p_i)_{1\leq i\leq m}$ leads to

$$mFDR^m_1(p) = \alpha\frac{\mathbb{E}\left[\frac{|\mathcal{H}_0|}{m}R^{*,m}_{1,\alpha}\right]}{\mathbb{E}R^m_{1,\alpha}} \leq \alpha\frac{\mathbb{E}R^{*,m}_{1,\alpha}}{\mathbb{E}R^m_{1,\alpha}},$$

Furthermore, if $\mathbb{E}[R^{*,m}_{\alpha,1}|m_0]$ is decreasing:

$$mFDR^m_1(p) \leq \alpha(1-\pi)\frac{\mathbb{E}R^{*,m}_{1,\alpha}}{\mathbb{E}R^m_{1,\alpha}},$$

The proof is moved to Section 3.7.4. If the ratio $\mathbb{E}R^{*,m}_{1,\alpha}/\mathbb{E}R^m_{1,\alpha}$ were known, it could be possible to control of the $mFDR$ criterion at level α by simply applying the BH-procedure with a preliminary level $\alpha' = \frac{m}{m_0}\frac{\mathbb{E}R^*_\alpha}{\mathbb{E}R_\alpha}\alpha$. Unfortunately at this stage, this ratio is not known and the latter strategy cannot be straightforwardly applied. Deriving such a modified BH-procedure is the purpose of the next sections. Let us also recall that in the anomaly detection context, m_0 is unknown but expected to be close to m since only a few anomalies are usually expected. Therefore the main challenge remains to compute $\mathbb{E}R^{*,m}_{1,\alpha}/\mathbb{E}R^m_{1,\alpha}$.

3.4.4 Evaluating the ratio of rejection numbers

3.4.4.1 Using heuristic arguments

The Section 3.4.3.1 raises the importance of the ratio $\mathbb{E}R^{*,m}_{1,\alpha}/\mathbb{E}R^m_{1,\alpha}$ of rejection numbers. The present section aims at deriving a numeric approximation to this ratio. In a first step, a first result details the value of the denominator. In a second step, an approximation to the numerator is derived based on a heuristic argument and also empirically justified on simulation experiments.

When $mFDR$ is assumed to equal α, the expected number of rejections can be made explicit.

Proposition 3.4. *With the previous notation, let* $(X_1, \dots, X_m)$ *be given by Definition 3.4, where* π *denotes the unknown proportion of anomalies, and assume that* $mFDR_1^m = \alpha$ *and* $FNR_1^m = \beta \in [0, 1]$. *Then*

$$\mathbb{E}[R_{1,\alpha}^m] = \frac{m\pi(1-\beta)}{1-\alpha}. \tag{3.21}$$

The proof is postponed to Section 3.7.5. For instance, Eq. (3.21) establishes that the expected number of rejection output by BH_α increases with π, the unknown proportion of anomalies along the signal. This makes sens since the more anomalies, the more expected rejections. The expected number of rejection is also increasing with α: the larger α, the less restrictive the threshold, and the more rejections should be made. However the number of rejection decreases with the FNR value β. As β increases, the proportion of false negatives grows meaning that fewer alarms are raised, which results in a smaller number of rejections.

In what follows, the assumption is made that anomalies are easy to detect, meaning that the FNR_1^m value β is negligible compared to 1. In this context, Proposition 3.4 would yield that

$$\mathbb{E}[R_{1,\alpha}^m] \approx \frac{m\pi}{1-\alpha}. \tag{\textbf{Power}}$$

An another assumption is also made about the relationship between $\mathbb{E}[R_{1,\alpha}^m]$ and $\mathbb{E}[R_{1,\alpha}^{*,m}]$. This assumption is based on a heuristic argument supported by the results of numerical experiments as reported in Table 3.3. In what follows, it is assumed that

$$\mathbb{E}[R_{1,\alpha}^{*,m}] = \mathbb{E}[R_{1,\alpha}^m] + 1. \tag{\textbf{Heuristic}}$$

No mathematical proof of this statement is given in the present paper. However, Table 3.3 displays numerical values which empirically support this approximation, whereas further analyzing the connection between these quantities should be necessary.

BH_α	0.05	0.1	0.2
$\mathbb{E}[R_{1,\alpha}^m]$	2.14	2.32	2.78
$\mathbb{E}[R_{1,\alpha}^m(i)]$	3.18	3.44	3.99

Table 3.3: Numerical evaluations for different values of α (10^3 repetitions)

Let us emphasize that Table 3.3 has been obtained with Gaussian data (generated similarly to those detailed in Section 3.3.4). For all the three considered values of α, one observes that $\mathbb{E}[R^*]$ remains close to (but also slightly larger than) $\mathbb{E}[R] + 1$.

These two assumptions give rise to a strategy for computing the ratio $\frac{\mathbb{E}[R^*]}{\mathbb{E}[R]}$. So all ingredient to build a procedure that control mFDR are given.

$$\frac{\mathbb{E}[R^*]}{\mathbb{E}[R]} = 1 + \frac{1-\alpha}{m\pi} \tag{3.22}$$

$$\alpha' = \alpha \left(1 + \frac{1-\alpha}{m\pi}\right)^{-1} \tag{3.23}$$

$$\tag{3.24}$$

The value α' can be used with the mBH procedure. Under (**Heuristic**) and (**Power**), according to Corollary 3.5, mBH allows global control of FDR at the level α.

3.4.4.2 Estimation on a training set

The true proportion of anomalies is usually not known, and estimating the proportion of anomalies is error-prone. Assumptions are also difficult to ensure. The number of detections is known by the user. To estimate $\mathbb{E}[R_{\tilde{\alpha}}]$, the procedure $BH_{\tilde{\alpha}}$ is applied to each of the subseries of length m from the training set. The average number of detections is computed and noted as $\hat{\mu}_{R_{\tilde{\alpha}}}$. However, it is not possible to compute $R^*_{\tilde{\alpha}'}$, the number of detections after one p-value associated with normal data is set to 0 since the true labels are unknown. But since the proportion of true anomalies π is close to 0, $\mathbb{E}[R^*_{\tilde{\alpha}'}]$ can be approximate by $\mathbb{E}[R^{**}_{\tilde{\alpha}'}]$, the number of detections after one p-values (possible abnormal) is set to 0. Then, to estimate $\mathbb{E}[R^{**}_{\tilde{\alpha}'}]$, the procedure $BH_{\tilde{\alpha}}$ is applied again to the same subseries but after a randomly chosen p-value is replaced by 0. Here again, the average number of detections is computed and noted $\hat{\mu}_{R^{**}_{\tilde{\alpha}}}$. By varying $\tilde{\alpha}$ it is possible to estimate:

$$\alpha' = \arg\max_{\tilde{\alpha}} \left\{ \frac{\hat{\mu}_{R^{**}_{\tilde{\alpha}}}}{\hat{\mu}_{R_{\tilde{\alpha}}}} \tilde{\alpha} \leq \alpha \right\} \tag{3.25}$$

This estimator of α' can be used for the modified BH procedure. According to Corollary 3.5, the global control of FDR is ensured by $\hat{\varepsilon}_{BH_{\alpha'}}$ when the size of the training set goes to infinity.

3.4.4.3 Modified BH

From previous Sections 3.4.3.1 and 3.4.4.1, it is now possible to suggest and analyze the new modified BH-procedure (mBH in the sequel).

Definition 3.6 (Modified BH-procedure (mBH))**.** *Let m be an integer and $\alpha \in [0,1]$. Let us introduce the level $\alpha' = \alpha'(\alpha)$ an estimate of $\arg\max_{\tilde{\alpha}} \left\{ \frac{\mathbb{E}[R^{*,m}_{1,\tilde{\alpha}}]}{\mathbb{E}[R^m_{1,\tilde{\alpha}}]} \tilde{\alpha} \leq \alpha \right\}$, which can be given by some procedure. Then the modified BH-procedure, denoted by mBH_α, is given for all true p-values $(p_1, \ldots, p_m) \in [0,1]^m$ by,*

$$mBH_\alpha(p_1, \ldots, p_m) = BH_{\alpha'}(p_1, \ldots, p_m).$$

The related mBH_α threshold at level α is defined as

$$\varepsilon_{mBH_\alpha} = \varepsilon_{BH_{\alpha'}},$$

when computed with true p-values, and $\hat{\varepsilon}_{mBH_\alpha} = \hat{\varepsilon}_{BH_{\alpha'}}$ when used with empirical p-values.

The above definition defines the mBH_α in terms of the BH-procedure by simply changing the level of control α'. This new level value depends on the unknown proposition π of anomalies.

Since in realistic anomaly detection scenarios observations are not labeled, [159] provides guidelines on how π could be estimated. Combining the results of Theorem 3.4 and Proposition 3.3, the procedure mBH allows to get the control of FDR of the complete series at the desired level α. This property is described in Corollary 3.5.

Corollary 3.5 (Control of FDR using mBH)**.** *Under the same notations and assumptions as Theorem 3.4. Let m and n be two integers. Suppose one of the following statements if true:*

1. *For all t p-values of $\hat{\boldsymbol{P}}_t$ are true p-values.*
2. *For all t p-values of $\hat{\boldsymbol{P}}_t$ are are empirical p-values with independent calibration sets of cardinality n.*

   Let i be a true null hypothesis in $\hat{\boldsymbol{P}}_t$, the sequence $\hat{\boldsymbol{P}}'_t$ is defined by $\hat{\boldsymbol{P}}'_{t,i} = 0$ and $\hat{\boldsymbol{P}}'_{t,j} = \hat{\boldsymbol{P}}_{t,j}$ for $j \neq i$. Let $R^{*,m}_{\alpha,1}$ be the number of rejections when applying BH_α to $\hat{\boldsymbol{P}}'_t$.
3. *For all t, p-values of $\hat{\boldsymbol{P}}_t$ are are empirical p-values with a unique calibration set of cardinality n.*

   Let i be a true null hypothesis in $\hat{\boldsymbol{P}}_t$, the sequence $\hat{\boldsymbol{P}}'_t$ is defined by $\hat{\boldsymbol{P}}'_{t,i} = 0$ and $\hat{\boldsymbol{P}}'_{t,j} = \hat{\boldsymbol{P}}_{t,j} - \frac{1}{n}\mathbb{1}[\hat{\boldsymbol{P}}_{t,j} < \hat{\boldsymbol{P}}_{t,i}]$ for $j \neq i$. Let $R^{*,m}_{\alpha,1}$ be the number of rejections when applying BH_α to $\hat{\boldsymbol{P}}'_t$.

Suppose there are ν and α' such that:

$$\frac{\mathbb{E}R^{*,m}_{1,\alpha'}}{\mathbb{E}R^{m}_{1,\alpha'}}\alpha' = \alpha$$

Then, FDR of the entire time series can be controlled at level α by using $\hat{\varepsilon}_{BH_{\alpha'},t} = f_m(\hat{\mathbf{P}}_t)$

$$FDR^{\infty}_{1}(\hat{\varepsilon}_{BH_{\alpha'}}, \hat{p}) \leq (1-\pi)\alpha \tag{3.26}$$

The proof is moved to Section 3.7.6. Corollary 3.5 describes the properties that the sequence of p-values $\hat{p}$ and the sequence of vector p-values $\hat{\mathbf{P}}$ should satisfy in order to get the control on FDR. Theorem 3.5 gives practical ways to compute $\hat{p}$ and $\hat{\mathbf{P}}$ that satisfy these requirements. It gives the ingredient to build an anomaly detector controlling FDR at a desired level α. Our Algorithm 1 implements this result.

Theorem 3.5 (Global FDR control using mBH_α)**.** *Let (X_t) be a mixture process introduced in Definition 3.4. Let $\alpha \in [0,1]$ be the desired FDR level for the full time series. Let m and n be integers.*

If one of the three statements is true:

1. $\forall t, \hat{p}_t = 1 - \mathbb{P}_{X\sim\mathcal{P}_0}(a(X) > a(X_t))$
2. $\forall t, \hat{p}_t = \hat{p}_e(a(X_t), \{a(Z_{t,1}), \ldots, a(Z_{t,n})\})$ *with* $Z_{t,i} \sim \mathcal{P}_0$

   In this two cases, let $\hat{\mathbf{P}}_t = (\hat{p}_{t-m+1}, \ldots, \hat{p}_t)$. Let choose i in $\mathcal{H}_0 \cap [\![t-m+1, t]\!]$, the sequence $\hat{\boldsymbol{P}}'_t$ is defined by $\hat{\boldsymbol{P}}'_{t,i} = 0$ and $\hat{\boldsymbol{P}}'_{t,j} = \hat{\boldsymbol{P}}_{t,j}$ for $j \neq i$. Let $R^{*,m}_{\alpha,1}$ be the number of rejections when applying BH_α to $\hat{\boldsymbol{P}}'_t$.

3. $\forall k \in [\![1, m]\!], (\hat{\boldsymbol{P}}_t)_{t,k} = \hat{p}_e(a(X_{t-m+k}), \mathcal{S}^{cal})$*, with the calibration set*

$$\mathcal{S}^{cal} = \{(1 - A_{t-n+1-m})a(X_{t-n+1-m}) + A_{t-n+1-m}a(Z_{t,1}), \dots, \\ (1 - A_{t-m})a(X_{t-m}) + A_{t-m}a(Z_{t,n})\}$$

with $Z_{t,i} \sim \mathcal{P}_0$ *and* $\hat{p}_t = \hat{\boldsymbol{P}}_{t,m}$*.*

Let choose i *in* $\mathcal{H}_0 \cap [\![t - m + 1, t]\!]$*, the sequence* $\hat{\boldsymbol{P}}'_t$ *is defined by* $\hat{\boldsymbol{P}}'_{t,i} = 0$ *and* $\hat{\boldsymbol{P}}'_{t,j} = \hat{\boldsymbol{P}}_{t,j} - \frac{1}{n}\mathbb{1}[\hat{\boldsymbol{P}}_{t,j} < \hat{\boldsymbol{P}}_{t,i}]$ *for* $j \neq i$*. Let* $R^{*,m}_{\alpha,1}$ *define the number of rejections when applying* BH_α *to* $\hat{\boldsymbol{P}}'_t$*.*

Suppose there are α' *and* ν *such that:*

$$\frac{\mathbb{E}R^{*,m}_{1,\alpha'}}{\mathbb{E}R^{m}_{1,\alpha'}}\alpha' \leq \alpha \quad \text{and} \quad n = \nu m/\alpha' - 1.$$

Then, with $\hat{\varepsilon}_{BH_{\alpha'},t} = \hat{\varepsilon}_{BH_{\alpha'}}(\hat{\boldsymbol{P}})_t$*:*

$$FDR^{\infty}_1(\hat{\varepsilon}_{BH_{\alpha'}}, \hat{p}) \leq (1 - \pi)\alpha.$$

The main merit of Theorem 3.5 is to establish the actual level of control for the global FDR of the full time series depending on the type of empirical p-value used in the anomaly detection process. The last type of empirical p-values is (almost) the one used in practice in the present work. More specifically, Section 3.5.2.3 describes empirical p-values based on a "Sliding Calibration Set".

Proof of Theorem 3.5. The Corollary 3.5 gives the two properties that the p-values families has to verify to control FDR of the time series:

- The $\hat{\mathbf{P}}_t$ are identically distributed and independent when time distance is larger than $n+m$.
- For each t, $\hat{\mathbf{P}}_t$ are either true p-values, or empirical p-values with independent or unique calibration set.

In the following, these properties are verified for the different p-values.

1. The sequence of true p-value $\mathbb{P}_{X\sim\mathcal{P}_0}(a(X) > a(X_t))$ is i.i.d., because the time series mixture is i.i.d. Then $\hat{\mathbf{P}}_t = (\hat{p}_{t-m+1}, \dots, p_t)$ are independent for a time distance larger than m. Using the first statement of Corollary 3.5 FDR of the whole time series is controlled at level $(1 - \pi)\alpha$.
2. The sequence of empirical p-value is i.i.d., because the time series mixture i and the calibration sets are i.i.d. Then $\hat{\mathbf{P}}_t = (\hat{p}_{t-m+1}, \dots, p_t)$ are independent for a time distance larger than m. Using the second statement of Corollary 3.5 FDR of the whole time series is controlled at level $(1 - \pi)\alpha$.
3. This p-value family is not i.i.d. However, because the calibration are build using a sliding window of size n, two p-values subseries of length m, $\hat{\mathbf{P}}_{t_1}$ and $\hat{\mathbf{P}}_{t_2}$, are independent when $|t_1 - t_2| > m + n$. Then the third statement of Corollary 3.5 ensure that FDR of the whole time series is controlled at level $(1 - \pi)\alpha$.

□

3.4.5 Empirical results

In this section, the abilities to get local control of the mFDR and the global control of the FDR, using mBH are assessed empirically. Corollary 3.5 and Theorem 3.5 give theoretical results about the control of the mFDR for subseries under the assumptions **Power** and **Heuristic**. However, these assumptions are hard to ensure in practice. In Section 3.4.5.1, the assessment is done on simulated data where the level of atypicity of the anomalies varies from one sample to another. Different scenarios are tested to verify if the mFDR control hold. Theorem 3.3 and Theorem 3.4 give FDR control over the full time series. In Section 3.4.5.2, the abilities of thresholds computed on disjoint and overlapping subseries to control the mFDR using are compared. Theorem 3.3 and Theorem 3.4 give asymptotic FDR control over the full time series. But there is no result about the speed of convergence, which is necessary when used on finite time series. In Section 3.4.5.3, the FDR for the full time series is calculated across different situations, as a function of time series size. It is possible to figure out when the entire series reaches control of the FDR.

3.4.5.1 Control of the mFDR on disjoint subseries

Experiment description From Corollary 3.5 the mBH_α controls the $mFDR_1^m$ only if the **Power** assumption is satisfied. Since the power of the anomaly detector depends on how easy it is to detect anomalies, the level of atypicity δ is introduced. To quantifies the atypicity of a data point X_t, the true p-value is computed as $p_t = \mathbb{P}_{X\sim\mathcal{P}_0}(X > X_t)$, and the atypicity level is defined as the inverse of the p-value: $\delta_t = 1/p_t$. The atypicity level is preferred over the p-values because it is easier to show on the x-axis of the chart, when the p-value is small. To evaluate the effect of power, for each sample all anomalies have their level of atypicity lower bounded a given parameter δ. Therefore, it is possible to observe the effect of a variation in the level of atypicity on the $mFDR_1^m$, FDR_1^m and FNR_1^m.

For a given scenario—meaning a proportion of anomalies π, a level of atypicity δ, and a desired level of mFDR noted α—the actual mFDR, FDR, and FNR are estimated. These quantities are estimated using $J = 50$ samples of m data points. To control the estimation error made when estimating on a finite number of samples, each estimation is repeated $B = 100$ times. The estimation proceeds as follows:

1. With $1 \leq b \leq B$, and $1 \leq j \leq J$, m data point are generated.
   - m_0 normal data $p_{b,j,1}, \ldots, p_{b,j,m_0}$ are generated according the reference law $U([0,1])$.
   - m_1 abnormal data $p_{b,j,m_0+1}, \ldots, p_{b,j,m}$ are generated using the alternative law $U([0, 1/\delta])$, with δ the level of atypicity of the anomalies.
2. Then, for each sample, the thresholds are estimated with BH and mBH procedures:
   - $\hat{\varepsilon}_{b,j,BH} = BH_\alpha(p_{b,j,1}, \ldots, p_{b,j,m})$,
   - $\hat{\varepsilon}_{b,j,mBH} = mBH_\alpha(p_{b,j,1}, \ldots, p_{b,j,m})$.
3. The number of rejections, false positives and false negatives are computed on each sample and according each threshold. Using $M \in \{mBH, BH\}$:
   - $R_{b,j,M} = \sum_{i=1}^{m} \mathbb{1}[p_{b,j,i} \leq \hat{\varepsilon}_{b,j,M}]$,
   - $FP_{b,j,M} = \sum_{i=1}^{m_0} \mathbb{1}[p_{b,j,i} \leq \hat{\varepsilon}_{b,j,M}]$,
   - $FN_{b,j,M} = \sum_{i=m_0+1}^{m} \mathbb{1}[p_{b,j,i} > \hat{\varepsilon}_{b,j,M}]$.
4. The FDR, mFDR and FNR are estimated by averaging results over the J samples:

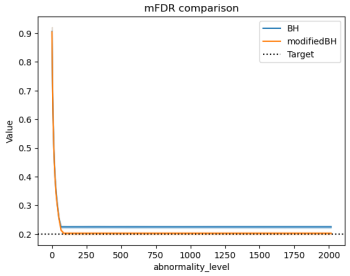
- $FDR_{b,M} = \frac{1}{J}\sum_{j=1}^{J} \frac{FP_{b,j,m}}{R_{b,j,m}}$,
- $mFDR_{b,M} = \frac{\sum_{j=1}^{J} FP_{b,j,m}}{\sum_{j=1}^{J} R_{b,j,m}}$,
- $FNR_{b,M} = \frac{1}{J}\sum_{j=1}^{J} \frac{FN_{b,j,m}}{m_1}$.

These steps are then repeated over the different scenarios.

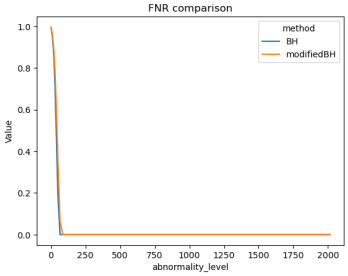
Results and Analysis The results are shown in Figure 3.8 by varying δ, α and m_1. In Figure 3.8, the level of atypicity δ in represented in the abscissa. The ordinate represents the estimated mFDR (in Figure 3.8a or 3.8c) or FNR (in Figure 3.8b or 3.8d). Different colors are used to distinguish between BH and mBH procedures. More results are displayed in Section 3.8.3.

For a low level of atypicity δ, the FNR and the mFDR are high because the anomalies are difficult to detect. By increasing δ, the FNR and the mFDR decrease. As shown in Figure 3.8b, with values of δ around 100, the FNR is equal to 0 which can also generate a constant mFDR as shown in Figure 3.8a. For the mBH-procedure, the mFDR is constant and equal to α. This is consistent with Theorem 3.4, which guarantees the control at level α when all anomalies are detected.

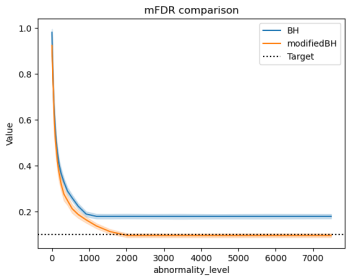
Figure 3.8d shows the totality of the anomalies detected for $\delta = 2000$. The same result in figure 3.8b with $\delta = 100$. This is explained by the different parameters of the experiment. The easier the anomalies are detected, faster the $FNR = 0$ is reached for a small δ and therefore the easier it is to guarantee $mFDR = \alpha$.



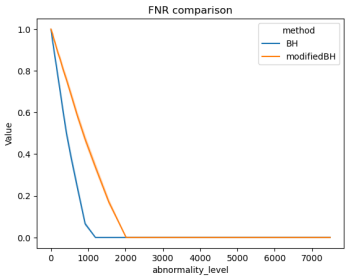
(a) mFDR, $\alpha = 0.2$, $\pi = 0.07$



(b) FNR, $\alpha = 0.2$, $\pi = 0.07$



(c) mFDR, $\alpha = 0.1$, $\pi = 0.01$



(d) FNR, $\alpha = 0.1$, $\pi = 0.01$

Figure 3.8: mFDR and FNR as a function of level of atypicity across different scenarios

Conclusion In order to control the mFDR at the desired level α using mBH, the FNR has to be equal to 0. The capacity of mBH to control the mFDR depends of the difficulty of the problem. When abnormality proportion and level of atypicity are lower, the power of mBH decreases and the mFDR is harder to control. The results of this experiment gives an idea of the atypicity level that the detector can find. For example, for $\pi = 0.01$ and $\alpha = 0.1$ the abnormality level must be at least 2000. To give an idea, this corresponds to a threshold of 3.5σ for Gaussian data ($2\Phi(3.5) = 1/2149$). The use of such a threshold seems realistic in relation to the literature [32].

3.4.5.2 Disjoint subseries vs overlapping subseries

Experiment Description Theorems 3.3 and 3.4 theoretically prove the control of the FDR over the full time series throw control of the mFDR over disjoint subseries or overlapping subseries. According Corollary 3.5, the procedure mBH_α allows the control of the mFDR over subseries under assumption **Heuristic** and **Power** that are hard to verify. Empirical results from Section 3.4.5.1 show that control of mFDR for the disjoint subseries can be obtained for scenarios where the level of atypicity δ is high enough. It still unknown whether these results hold true in cases where the subseries overlap In this section FDR control throw disjoint and overlapping subseries are compared.

For each scenario, the quantities $mFDR_1^m$ and FNR_1^m are estimated two times, using disjoint subseries and using overlapping subseries. All subseries are extracted from the same time series of length $T = 10^4$. The distribution of these estimations is obtained by repeating the experiment across $B = 100$ time series. Thus, the two estimations of $mFDR_1^m$ and FNR_1^m quantities can be compared. The experimental design is described as follows:

1. With b in $[\![1, B]\!]$ and t in $[\![1, T]\!]$, the time series is generated from a mixture model:
   - $A_{b,t} \sim Ber(\pi)$
   - If $A_{b,t} = 0$, $p_{b,t} \sim U([0, 1])$
   - Otherwise: $p_{b,t} \sim U([0, 1/\delta])$
2. The thresholds of mBH are estimated on each subseries $p_{b,t+1}, \dots, p_{b,t+m_0+m_1}$:
   - $\hat{\varepsilon}_{b,t} = mBH_\alpha(p_{b,t+1}, \dots, p_{b,t+m_0+m_1})$.
3. The numbers of rejections, false positives and false negatives are calculated, according the different cases.
   (a) In the disjoint subseries case, the quantities are computed using only thresholds on the form $\hat{\varepsilon}_{b,km}$ over disjoint subseries
   For $1 \le b \le B$ and $1 \le j \le J = T/m$:
   - $R_{b,j,d} = \sum_{t=jm+1}^{(j+1)m} \mathbb{1}[p_{b,j,t} \le \hat{\varepsilon}_{b,jm}]$,
   - $FP_{b,j,d} = \sum_{t=jm+1}^{(j+1)m} \mathbb{1}[p_{b,j,t} \le \hat{\varepsilon}_{b,jm}](1 - A_t)$,
   - $FN_{b,j,d} = \sum_{t=jm+1}^{(j+1)m} \mathbb{1}[p_{b,j,t} > \hat{\varepsilon}_{b,jm}]A_t$.

   The mFDR and FNR are estimated:
   - $mFDR_{b,d} = \frac{1}{J}\sum_{j=1}^{J} FP_{b,j,m,d} \frac{1}{J}\sum_{j=1}^{J} R_{b,j,m,d}$,
   - $FNR_{b,d} = \frac{1}{J}\sum_{j=1}^{J} \frac{FN_{b,j,m,d}}{m_1}$.

(b) In the overlapping subseries case, the quantities are computed using the thresholds from all overlapping subseries $\hat{\varepsilon}_{b,t}$:
For $1 \leq b \leq B$ and $1 \leq j \leq J = T/m$:

- $R_{b,j,o} = \sum_{t=jm+1}^{(j+1)m} \mathbb{1}[p_{b,t-m+1,t,o} \leq \hat{\varepsilon}_{b,t}]$,
- $FP_{b,j,o} = \sum_{t=jm+1}^{(j+1)m} \mathbb{1}[p_{b,t-m+1,t} \leq \hat{\varepsilon}_{b,t}](1 - A_t)$,
- $FN_{b,j,o} = \sum_{t=jm+1}^{(j+1)m} \mathbb{1}[p_{b,t-m+1,t} > \hat{\varepsilon}_{b,t}]A_t$.

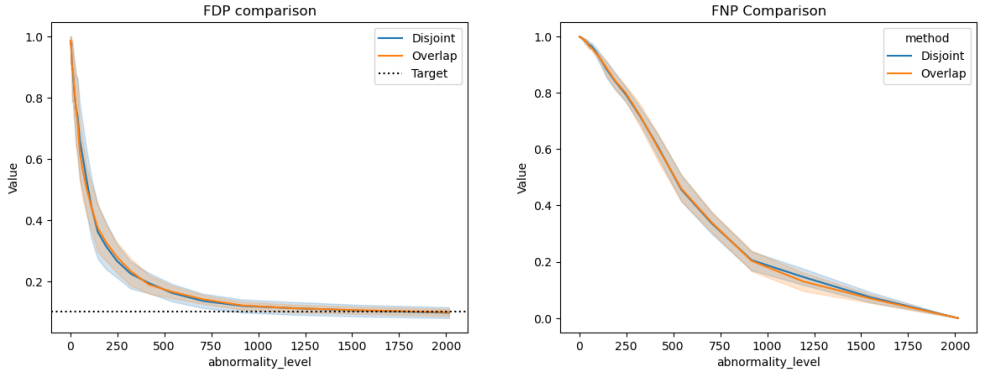
Notice the difference with disjoint windows case, all p-values of a subseries are compared to different thresholds and not to the same $\hat{\varepsilon}_{b,jm}$.

The mFDR and FNR are estimated:

- $mFDR_{b,o} = \frac{1}{J}\sum_{j=1}^{J} FP_{b,j,m,o} \frac{1}{J}\sum_{j=1}^{J} R_{b,j,m,o}$,
- $FNR_{b,o} = \frac{1}{J}\sum_{j=1}^{J} \frac{FN_{b,j,m,o}}{m_1}$.

Different scenarios are generated by varying the proportion of anomalies π and the atypicity level δ.

Results and analysis As shown in Figure 3.9, disjoint and overlapping subseries control give similar results in mFDR and FNR for considered cases. Indeed, the curves are indistinguishable and decrease at the same rate.



(a) mFDR, $\alpha = 0.1$, $\pi = 0.01$ (b) FNR, $\alpha = 0.1$, $\pi = 0.01$

Figure 3.9: Comparison of mFDR and FNR control with disjoint and overlapping windows method.

Conclusion The FDR control quality are similar for both strategies, overlapping windows and disjoint windows. This imply that performances of the anomaly detector to not decrease by using overlapping windows instead of disjoint windows. This is a practical result that allows to do real time detection without having to wait to complete disjoint windows.

3.4.5.3 Convergence of false discovery rate control

This section studies the convergence rate of the FDR over the full time series using mBH_α.

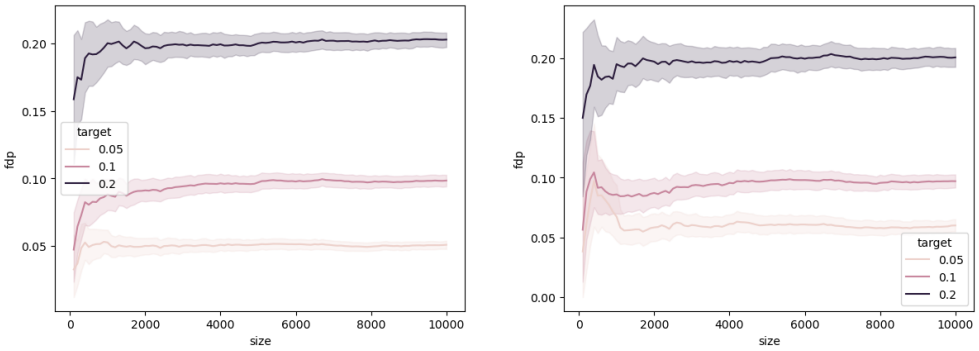
Experiment Description The theoretical results obtained in Theorem 3.4 only guarantee an asymptotic control of the FDR on the whole time series. In practice, it is more useful to have a control of the FDR at any time, i.e. on subseries of finite size. The question is empirically studied by observing the speed of convergence of the false discovery rate towards the level α. The FDR of the full time series is calculated across different scenarios, as a function of time series size. In order to get the distribution of the FDR, the experiment is repeated on $B = 100$ time series. The maximal time series size explored is $T = 10^4$.

1. For $1 \leq b \leq B$ and for $1 \leq t \leq T$:
   - $A_{b,t} \sim Ber(\pi)$
   - If $A_{b,t} = 0$, $p_{b,t} \sim U([0, 1])$
   - Otherwise: $p_{b,t} \sim U([0, 1/\delta])$
2. The thresholds are estimated with mBH_α:
$$\hat{\varepsilon}_{b,t,\alpha} = \hat{\varepsilon}_{mBH_\alpha}(p_{b,t-m+1}, \ldots, p_{b,t})$$
3. The proportion of false discovery (FDP) on the partial time series are calculated:
$$FDP_{b,t,\alpha} = \frac{\sum_{u=1}^{t}(1 - A_{b,t})\mathbb{1}[p_{b,t} \leq \hat{\varepsilon}_{b,t,\alpha}]}{\sum_{u=1}^{t}\mathbb{1}[p_{b,t} \leq \hat{\varepsilon}_{b,t,\alpha}]}$$

Different scenarios are generated by varying the proportion of anomalies π and the atypicity level δ.

Results and analysis In Figure 3.10, the false discovery proportion is represented in the ordinate according to the size of the time series given in the abscissa. The different levels of α used to compute mBH threshold are experimented with the results of the median FDP and its 95% band is shown in different colors. Different scenarios are represented by varying the proportion of anomalies between the sub figures.

It can be observed that the convergence is quite fast from a size of 2000 data points, since for a α of 0.05, is has 95% chance to have a false positive rate between 0.04 and 0.06, on Figure 3.10. Thus, the control of the false positive rate, can be ensured with a high probability, for a series of one data point per minute recorded over a few days, .



(a) $\delta = 1000$, $\pi = 0.02$ (b) $\delta = 1000$, $\pi = 0.01$

Figure 3.10: FDR over the full time series as a function of the time series size.

Conclusion This ensures that the control at level is reached not only for infinite time series but also for finite time series which allows our model to be used in practice.

3.5 Empirical simulation against competitor

The control of the FDR with p-values estimated empirically has been studied at Section 3.3. Theorem 3.2 ensure the control of the FDR_1^m when the p-values are estimated on calibration set having particular cardinality value. Theorems 3.3 and 3.4 ensure the control of the FDR of the full time series throw control of the $mFDR_1^m$ of the subseries. Corollary 3.5 enables to deduce that the mBH_α procedure can be apply to control the FDR of the full time series under the **Heuristic** and **Power** assumptions. Even though these assumptions are hard to ensure theoretically, the experiment at Section 3.4.5.1 shows that the $mFDR_1^m$ is controlled for tested scenario, provided that anomalies are sufficiently atypical. Experiment from Section 3.4.5.3 shows that the control of the FDR is possible even the time series is not infinite as required by Theorem 3.4.

These different results provide the conditions for building an anomaly detector that controls the FDR of the time series through control of the mFDR on the subseries and the p-empirical value. Our anomaly detector is evaluated under different scenarios by varying the generated anomalies and the targeted FDR. To understand the source of the difficulties that the anomaly detector may encounter, different sequences of p-values with oracle information are introduced. Our anomaly detector is compared against Levels based On Recent Discovery (LORD) which is a online multiple testing procedure, introduced in [87] to control the FDR.

3.5.1 Data

The synthetic data are generated from Gaussian distribution. With the use of the empirical p-value estimator there are no need to evaluate on other data distribution. Only anomaly proportion and the distribution shift associated to anomalies impact the performances of the anomaly detector. Data are generated accordingly with Definition 3.4 with Gaussian reference distribution and anomaly spike like in Section 3.3.4.1. The strength of the distribution shift noted by Δ takes value in $\{3\sigma, 3.5\sigma, 4\sigma\}$ and the abnormality proportion noted π is equal to 0.01. Each

generated time series contain $T = 10^4$ data points, according to Section 3.4.5.3 it is enough data points to observe FDR convergence. Each experiment is repeated over 100 time series.

For t in $[\![1, T]\!]$:

- $A_t \sim B(\pi)$
- $X_t = \begin{cases} \mathcal{N}(0, \sigma^2) & \text{if } A_t = 0 \\ \Delta\sigma & \text{else} \end{cases}$

The value of Δ represents the atypicity score of the anomalies. Anomalies with higher Δ are easier to detect. In this experiment, the standard deviation σ is set to 1.

3.5.2 Threshold and p-value estimators description

3.5.2.1 Our proposal mBH on overlapping subseries

Using the p-value with the empirical estimator, the anomalies are detected by using mBH as the threshold estimator on overlapping subseries in the Algorithm 1. For each time t, the threshold is computed as: $\hat{\varepsilon}_{mBH_\alpha,t} = f_m(\hat{p}_{t-m}, \dots, \hat{p}_t)$, where f_m is the mBH-procedure. To ensure FDR control according to Theorem 3.1, the cardinality of the calibration set to be equal to $n = \frac{m}{\alpha} - 1$. In this experiment m is equal to 100 and α takes values 0.1 and 0.2 depending the tested scenario. So the calibration set takes values 999 or 1999.

3.5.2.2 LORD

LORD introduced in [87] is based on alpha-investing rules to define a threshold on p-values. For each time t the threshold is computed from according to the alpha-investing rules, depending on previous decision made by the algorithm. For more precision refer to the original article [87]. The empirical p-value specified in Definition 3.5 does not respect this property while the conformal p-value, defined in Equation 3.2.3.1 respects this property. Using conformal p-values to apply LORD algorithm leads to a weak power detecting anomalies. The issue is that $\check{p} \geq \frac{1}{n+1}$ is always verified and the threshold sequence $\hat{\varepsilon}_t$ decreases quickly when no rejection are made. No anomaly can be detected. For these reasons, the empirical p-value introduced in Equation 3.5 is used while applying LORD and mBH. In this experiment LORD3 from [87] is used with the same parameters as in the original paper.

3.5.2.3 p-value estimation

Different sequences of p-values are used to understand the limitations of our anomaly detector. The true p-values are used to evaluate the case where the only limitation comes from the multiple testing procedure. One can thus understand how the estimation of the p-values affects the detection of anomalies. One way to estimate p-values in practice is to use the same calibration set for all p-values. This is referred as the fixed calibration set. However, the p-values may be biased in that particular calibration set. In practice, the usual way to implement the estimated p-values is to use a sliding calibration set. To evaluate the p-value of a data point X_t, n preceding data points are used as a calibration set. To a bias in the estimation, the points detected as abnormal cannot be part of the calibration set. However, the calibration set can be biased by undetected anomalies. To evaluate this impact, the sliding calibration set-$\star$ is introduced, where the knowledge oracle of the labels is used to construct the calibration set from the previous data points.

The different p-value sequences are computed as follows:

- **Oracle**: The true p-value is used instead of the estimated one.

$$\forall t \in [\![1, T]\!], \hat{p}_t = \Phi(X_t)$$

- **Fixed calibration set (Fixed Cal.)**: The p-value is estimated using the same calibration set $\{Z_i, i \in [1, n]\}$ for all observations.

$$\forall t \in [\![1, T]\!], \hat{p}_t = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}[Z_i > X_t]$$

- **Sliding Calibration set-$\star$ (Sliding Cal.-$\star$)**: The p-value is estimated using a calibration that is a sliding windows containing the n previous true normal data.

$$\forall t \in [\![1, T]\!], \hat{p}_t = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}[X_{h(t,i)} > X_t]$$

  With h the function that select observation that respect $\mathcal{H}_0$. For each t and i, $h(t, i)$ gives the i-th observation lower than t and that respect $\mathcal{H}_0$ hypothesis.

- **Sliding Calibration set (Sliding Cal.)**: The calibration set is a sliding windows containing the n previous estimated normal data.

$$\forall t \in [\![1, T]\!], \hat{p}_t = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}[X_{\hat{h}(t,i)} > X_t]$$

  With $\hat{h}$ the function that estimates the function h. For each t and i, $\hat{h}(t, i)$ give the i-th observation lower than t and $d_{\hat{h}(t,i)} = 0$.

3.5.3 Performance metrics

The anomaly detector are evaluated using their ability to control the FDR and minimize the FNR of the full time series. Therefore, the two applied metrics are the FDP and the FNP computed as:

$$FDP = \frac{\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < \hat{\varepsilon}_t](1 - A_t)}{\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < \hat{\varepsilon}_t]}$$

and

$$FNP = \frac{\sum_{t=1}^{T} \mathbb{1}[\hat{p}_t < \hat{\varepsilon}_t](1 - A_t)}{\sum_{t=1}^{T} A_t}$$

where $\hat{\varepsilon}_t$ is estimated using mBH or LORD and $\hat{p}_t$ is estimated using one of the estimator defined in Section 3.5.2.3.

3.5.4 Results

The box plots shown in Figures 3.15-3.18 represent the FDP and FNP distribution for 1000 repetitions. Inside each sub figure (a, b, c, d, e,..), the box plot distributions are displayed according to:

1. the multiple testing method mBH or LORD,
2. the p-value estimation model set to Oracle PV, Fixed Cal., Sliding Cal.-$\star$ or Sliding Cal.
3. and the distribution shift between the normal data and anomalies, noted Δ, varying from 4σ to 3σ.

Table 3.4 gives a summary using FDR and FNR estimations. It enables easily the comparison between these values coming from the different strategies combining:

1. the multiple testing method mBH or LORD,
2. the choice of the level α varying from 0.1 to 0.2,
3. the p-value estimation model set to Oracle PV, Fixed Cal., Sliding Cal.-$\star$ or Sliding Cal.
4. and the distribution shift between the normal data and anomalies, varying from 4σ to 3σ.

3.5.5 Analysis

3.5.5.1 Effect of the strength of the distribution shift Δ

According the assumption **Power** from Theorem 3.5, mBH_α enables control of the FDR at level α if all anomalies are detected.
To test this assertion, the different columns of the Table 3.4a, are compared. In the row "mBH with Oracle PV", with $\Delta = 4\sigma$ the FDR is estimated at 0.101 which is close to the desired level $\alpha = 0.1$. While, when $\Delta = 3\sigma$ the FDR level is estimated at 0.281 which is almost three times the desired level α. The FNR results in Table 3.4b needs to be taken into consideration. When $\Delta = 4\sigma$, the FNR is close to 0, while when $\Delta = 3\sigma$ the FNR is equal to 0.793. Similar results are obtained with other test configurations in Table 3.4c and Table 3.4d. The FDR control at the desired level need the FNR to be close to 0, which in this context is obtained for a Δ of at least 3.5σ.

3.5.5.2 Effect of p-value estimation

To understand how the p-value estimation can prevent the control of the FDR, the first four rows in Table 3.4a are compared. In the column "4σ", the FDR values for the configurations "Oracle PV", "Fixed Cal." and "Sliding Cal.-$\star$" are very close to the desired level $\alpha = 0.1$. This control is enabled by Theorem 3.5, since the p-values verify all hypotheses, in particular all data in the calibration sets are generated according to the reference distribution. However, in the case of "Sliding Cal.", the FDR increases at a value of 0.335. For the same configurations, the FDR remains low, between 0.019 and 0.040 as shown at Table 3.4b. The increase of FDR when using "Sliding Cal." instead of "Sliding Cal.-$\star$" is a consequence of calibration set contamination. Indeed, according to the procedure used to build the calibration sets, described in Section 3.5.2.3, all detected anomalies are removed from calibration sets used in the estimation of next p-values. When an observation is wrongly detected as an anomaly, this data point cannot be part of the calibration set at future steps of the online detection. Instead, it is replaced by an other data point having statistically a lower atypicity score. Indeed false positives have high atypicity score to be (wrongly) detected as anomalies. As a result, the calibration set contains data points with lower scores than if it had been generated under $\mathcal{P}_0$. It leads to underestimate the p-values and to increase the number of false positives. This illustrates the major drawback of mBH: it is highly sensitive to the non robustness of the p-value estimator. Figure 3.15a shows that using fixed calibration instead of sliding calibration-$\star$ gives a larger variance on the FDP while the FDR is

FDR, $\alpha = 0.1$	$\Delta = 4\sigma$	$\Delta = 3.5\sigma$	$\Delta = 3\sigma$
mBH with Oracle PV	0.101	0.113	0.281
mBH with Fixed Cal.	0.100	0.109	0.348
mBH with Sliding Cal.-$\star$	0.100	0.113	0.256
mBH with Sliding Cal.	0.335	0.222	0.346
LORD with Oracle PV	0.106	0.115	0.367
LORD with Fixed Cal.	0.111	0.277	0.736
LORD with Sliding Cal.-$\star$	0.070	0.190	0.841
LORD with Sliding Cal.	0.075	0.098	0.627

(a) FDR, $\alpha = 0.1$

FNR, $\alpha = 0.1$	$\Delta = 4\sigma$	$\Delta = 3.5\sigma$	$\Delta = 3\sigma$
mBH with Oracle PV	0.020	0.151	0.793
mBH with Fixed Cal.	0.026	0.135	0.669
mBH with Sliding Cal.-$\star$	0.019	0.140	0.669
mBH with Sliding Cal.	0.040	0.217	0.694
LORD with Oracle PV	0.033	0.260	0.905
LORD with Fixed Cal.	0.070	0.340	0.896
LORD with Sliding Cal.-$\star$	0.781	0.845	0.978
LORD with Sliding Cal.	0.052	0.327	0.907

(b) FNR, $\alpha = 0.1$

FDR, $\alpha = 0.2$	$\Delta = 4\sigma$	$\Delta = 3.5\sigma$	$\Delta = 3\sigma$
mBH with Oracle PV	0.200	0.208	0.277
mBH with Fixed Cal.	0.206	0.211	0.301
mBH with Sliding Cal.-$\star$	0.210	0.219	0.283
mBH with Sliding Cal.	0.833	0.815	0.761
LORD with Oracle PV	0.211	0.216	0.290
LORD with Fixed Cal.	0.210	0.263	0.665
LORD with Sliding Cal.-$\star$	0.061	0.149	0.625
LORD with Sliding Cal.	0.117	0.133	0.321

(c) FDR, $\alpha = 0.2$

FNR, $\alpha = 0.2$	$\Delta = 4\sigma$	$\Delta = 3.5\sigma$	$\Delta = 3\sigma$
mBH with Oracle PV	0.009	0.062	0.395
mBH with Fixed Cal.	0.014	0.045	0.355
mBH with Sliding Cal.-$\star$	0.008	0.059	0.339
mBH with Sliding Cal.	0.003	0.018	0.101
LORD with Oracle PV	0.016	0.117	0.610
LORD with Fixed Cal.	0.04	0.144	0.689
LORD with Sliding Cal.-$\star$	0.805	0.835	0.941
LORD with Sliding Cal.	0.026	0.168	0.692

(d) FNR, $\alpha = 0.2$

Table 3.4: Comparison of mBH versus LORD for online anomaly detection in Gaussian white noise with different abnormality levels.

the same. Using a single calibration set for the entire time series means that the FDP is highly dependent on the start of the time series. By modifying the calibration set at each time step, the statistical fluctuations in the FDP are smoothed over the course of the time series analysis.

3.5.5.3 Comparison with LORD

In this section, the results found using mBH and the ones using LORD are compared. As known from the literature, LORD controls the FDR of super-uniform p-values. In this experiment, the question is in the capacity of LORD method to control the FDR of empirical p-values that have no theoretical guaranties. It can be noticed in Figure 3.4a that LORD is able to ensure the control of the FDR for all calibration set definitions when anomalies are easier to detect as for $\Delta = 4\sigma$ or $\Delta = 3.5\sigma$. In particular, unlike mBH, LORD is able to control the FDR in the case of the sliding calibration set. However, mBH method has a lower FNR compared to the LORD method, as shown in 3.4a and 3.4b. For example, Table 3.4b shows that the FNR is equal to 0.019 with mBH while it is equal to 0.781 with LORD, in the case using Sliding Calibration set-$\star$ on data having $\Delta = 3\sigma$. Nevertheless, with the Sliding Calibration set case, the LORD method has quite the same FNR but with lower FDR (0.335 against 0.075). The contamination issue of mBH offsets the superior performance observed in the Sliding Calibration set-$\star$.

3.6 Conclusion

In this chapter, an online anomaly detector that aims to have a better control of the FDR at a given level α has been proposed. The research has been developed to tackle two issues:

- the empirical p-values: it ensures conditions on the calibration cardinality to ensure FDR control when using Benjamini-Hochberg.
- and the online detection: it ensures a global control of the FDR through local control of the mFDR of subseries, using a modified version of the BH-procedure.

The results of our research is the assessment of our proposal from the theoretical point of view and from empirical experiments. Our method has been compared with a method from the state of the art. It shows the strong capability for ensuring control of the FDR even in the case of empirical p-values. The major drawback and improvement path of our method is it relies on non-robust p-value estimation. In this chapter, only the simplified case of an iid time series has been studied. In the next chapter, time series with changing reference behavior are studied.

3.7 Proofs

3.7.1 Proof of Theorem 3.1

Proof of Theorem 3.1. Let R be a random variable describing the number of rejections made by BH_α that is, $R = \sum_{i=1}^{m} D_i$, where $D_i = 1$ if hypothesis $\mathcal{H}_{0,i}$ is rejected. Let also FP be the number of false positives made by BH_α. Then, $FP = \sum_{i=1}^{m} A_i D_i = \sum_{i=1}^{m_0} D_i$, where A_i is a

random variable equal to 1 if hypothesis $\mathcal{H}_{0,i}$ is true and 0 otherwise. Furthermore

$$\begin{aligned} FDP = \frac{FP}{R} &= \sum_{i=1}^{m_0} \frac{\mathbb{1}[p_i \leq \frac{\alpha R}{m}]}{R} \qquad (\text{since } D_i = \mathbb{1}[p_i \leq \tfrac{\alpha R}{m}]) \\ &= \sum_{k=1}^{m} \sum_{i=1}^{m_0} \frac{\mathbb{1}[p_i \leq \frac{\alpha k}{m}]\mathbb{1}[R=k]}{k}. \end{aligned} \tag{3.27}$$

Let us now introduce the random variables $R(i)$ that are the number of rejections generated by BH when p_i is replaced by the value 0 that is, $R(i) = BH_\alpha(p_1, .., p_{i-1}, 0, p_{i+1}, \ldots, p_m)$. It results that

$$\mathbb{1}[p_i \leq \frac{\alpha k}{m}]\mathbb{1}[R=k] = \mathbb{1}[p_i \leq \frac{\alpha k}{m}]\mathbb{1}[R(i)=k],$$

since, on the event $\{p_i \leq \frac{\alpha k}{m}\}$, p_i is rejected and therefore $R = R(i)$. Let us also notice that the independence between the p-values is already used at this stage since modifying the value of p_i does not affect that of the others.

By combining the previous argument and the independence between $R(i)$ and the other p-values, the expectation on both sides yields

$$\begin{aligned} FDP &= \sum_{k=1}^{m} \sum_{i=1}^{m_0} \frac{\mathbb{1}[p_i \leq \frac{\alpha k}{m}]\mathbb{1}[R(i)=k]}{k} \\ \Rightarrow \quad FDR = \mathbb{E}[FDP] &= \sum_{k=1}^{m} \sum_{i=1}^{m_0} \frac{\mathbb{P}[p_i \leq \frac{\alpha R}{m}]\mathbb{P}[R(i)=k]}{k} \\ &= \sum_{k=1}^{m} \sum_{i=1}^{m_0} \frac{\frac{\alpha k}{m}\mathbb{P}[R(i)=k]}{k} \\ &= \frac{m_0 \alpha}{m}, \end{aligned}$$

where the last equality results from the fact that the true p-values follow a uniform distribution on $[0,1]$. The result finally follows from noticing that for each $1 \leq i \leq m_0$, $\sum_{k=1}^{m} \mathbb{P}[R(i)=k]$, since $R(i) \geq 1$ by definition. □

3.7.2 Proof of Corollary 3.1

Proof of Corollary 3.1. To get a deeper understanding of the FDR expression obtained in Theorem 3.2, $q_{n,k}$ the fractional part of $\frac{\alpha k n}{m}$ is introduced:

$$q_{n,k} = \frac{\alpha k n}{m} - \left\lfloor \frac{\alpha k n}{m} \right\rfloor$$

When plugged into the FDR expression, it gives:

$$FDR = m_0 \sum_{k=1}^{m} \frac{\frac{\alpha k n}{m} + 1 - q_{n,k}}{n+1} \frac{}{k} \mathbb{P}(R(1) = k)$$

$$FDR = \frac{m_0 \alpha}{m} \frac{n}{n+1} + \frac{m_0}{n+1} \sum_{k=1}^{m} \frac{1 - q_{n,k}}{k} \mathbb{P}(R^* = k) \tag{3.28}$$

In order to get lower and upper bounds of the FDR, the value of $q_{n,k}$ should be expressed as a function of α, k, n and m.

For the next part of the proof, it is useful to express the relation between $q_{n,k}$ and $q_{n+1,k}$. It gives the effect of increasing the cardinality of the calibration by one. Using the definition of the fractional part:

$$q_{n+1,k} - q_{n,k} = \frac{\alpha k(n+1)}{m} - \left\lfloor \frac{\alpha k(n+1)}{m} \right\rfloor - \frac{\alpha k n}{m} + \left\lfloor \frac{\alpha k n}{m} \right\rfloor$$

$$q_{n+1,k} - q_{n,k} = \frac{\alpha k}{m} - \left\lfloor \frac{\alpha k(n+1)}{m} \right\rfloor + \left\lfloor \frac{\alpha k n}{m} \right\rfloor$$

Which can be expressed as a congruence relation:

$$q_{n+1,k} - q_{n,k} \equiv \frac{\alpha k}{m} \pmod 1 \tag{3.29}$$

Two cases are studied:

1. Particular case: there exists an integer $1 \leq \nu$ such that $\frac{\nu m}{\alpha}$ is an integer. the notation $n_\nu = \frac{\nu m}{\alpha}$ is introduced. Since: $\frac{\alpha k n_\nu}{m} = \frac{\alpha k \nu m/\alpha}{m} = k\nu$ is an integer, then the fractional part is null:

   $$q_{n_\nu,k} = 0$$

   If the calibration set cardinality n is equal to $n = n_\nu - 1 = \frac{\nu m}{\alpha} - 1$. Then, the congruence relation in Eq. 3.29 gives:

   $$q_{n_\nu - 1,k} \equiv q_{n_l,k} - \alpha k/m \pmod 1$$
   $$q_{n_\nu - 1,k} \equiv 0 - \alpha k/m \pmod 1$$

   Using the fact that fractional part of a number belongs to $[0, 1[$, the only possible value to $q_{n_\nu - 1,k}$ is:

   $$q_{n_\nu - 1,k} = 1 - \alpha k/m$$

   Plugging the value of $q_{n_\nu - 1,k}$ into Eq. 3.28, it gives:

   $$FDR = \frac{m_0 \alpha n}{m(n+1)} + \frac{m_0}{n+1} \sum_{k=1}^{m} \frac{\alpha k}{km} \mathbb{P}(R^* = k)$$

Simplifying by k and using that $\sum_{k=1}^{m} \mathbb{P}(R(i) = k) = 1$, the result is obtained:

$$FDR = \frac{m_0 \alpha n}{m(n+1)} + \frac{m_0 \alpha}{(n+1)m}$$
$$FDR = \frac{m_0 \alpha}{m}$$

2. General case: With $\alpha \in]0, 1]$, for each ν the notation $n_\nu = \lceil \frac{\nu m}{\alpha} \rceil$ is introduced. Notice that this definition is consistent with the particular case. The ceiling function definition gives:

$$\left\lceil \frac{\nu m}{\alpha} \right\rceil - 1 < \frac{\nu m}{\alpha} \leq \left\lceil \frac{\nu m}{\alpha} \right\rceil$$

Multiplying by αk on each side and the n_ν notation:

$$\frac{\alpha k (n_\nu - 1)}{m} < k\nu \leq \frac{\alpha k (n_\nu)}{m}$$

It implies that $\lfloor \frac{\alpha k (n_\nu - 1)}{m} \rfloor < \lfloor \frac{\alpha k (n_\nu)}{m} \rfloor$. Also, Eq. 3.29 is expressed as $q_{n_\nu, k} - q_{n_\nu - 1, k} \equiv \frac{\alpha k}{m} \pmod 1$:

$$1 - \frac{\alpha k}{m} \leq q_{n_\nu - 1, k} < 1 \tag{3.30}$$

Indeed, the fractional part of a number as to be larger than $1 - \alpha k/m$ so that adding $\alpha k/m$ increase the integer part.

By plugin the bounds of $q_{n_\nu - 1, k}$ into Eq. 3.28, it can gives the bounds of the FDR. At first, to compute the upper bound of the FDR the lower bound of $q_{n_\nu - 1, k}$ is used:

$$FDR \leq \frac{m_0 (n_\nu - 1)\alpha}{m n_\nu} + \frac{m_0}{n+1} \sum_{k=1}^{m} \frac{\alpha k}{km} \mathbb{P}(R^* = k)$$

With the same calculations as for the "Particular case", it gives:

$$FDR \leq \frac{m_0 \alpha}{m}$$

Similarly, the lower bound of the FDR can be obtained using the $q_{n,k}$ upper bound from Eq. 3.30 plugged into Eq. 3.28:

$$\frac{m_0 (n_\nu - 1)\alpha}{m n_\nu} + \frac{m_0}{n+1} \sum_{k=1}^{m} \frac{(1-1)}{k} \mathbb{P}(R_1 = k) < FDR$$
$$\frac{m_0 (n_\nu - 1)\alpha}{m n_\nu} < FDR$$

□

3.7.3 PRDS property for p-values having overlapping calibration sets

The following construction is used to describe a family of p-values with overlapping calibration sets. Let Z the vector that combine all calibration set, the Z_i are i.i.d. with marginal probability $\mathcal{P}_0$. The set of the n indices defining the elements of the calibration set related to $\hat{p}_i$ in Z is noted $\mathcal{D}_i$. The calibration related to X_1 is noted $Z_{\mathcal{D}_1} = (Z_{i_1}, \dots, Z_{i_n})$. For all i in $[\![1, m]\!]$: $\hat{p}_i = p\text{-value}(X_i, Z_{\mathcal{D}_i})$.

To proof that p-values with overlapping calibration set are PRDS as described in Definition 3.3, the methodology used in [11] to be extended in the case of overlapping calibration set. For i in $[\![1, m]\!]$ the calibration set associated to X_i is noted $Z_{\mathcal{D}_i}$. The law of total probabilities gives:

$$\begin{aligned}\mathbb{P}\left[\hat{p}_1^m \in A | \hat{p}_i = u\right] &= \int \mathbb{P}\left[\hat{p}_1^m \in A | \hat{p}_i = u | Z_{\mathcal{D}_i} = z\right] \mathbb{P}\left[Z_{\mathcal{D}_i} = z|\right] dz \\ &= \mathbb{E}_{Z_{\mathcal{D}_i}|\hat{p}_i=u} \mathbb{P}\left[\hat{p}_1^m \in A | \hat{p}_i = u | Z_{\mathcal{D}_i} = z\right]\end{aligned}$$

If these two lemma are suppose to be true, the PRDS property is verified.

Lemma 3.5.1. *For non-decreasing set A and vectors z, z' such that $z \succeq z'$, then*

$$\mathbb{P}\left[\hat{p}_1^m \in A | Z_{\mathcal{D}_i} = z\right] \geq \mathbb{P}\left[\hat{p}_1^m \in A | Z_{\mathcal{D}_i} = z'\right] \tag{3.31}$$

Lemma 3.5.2. *For $u \geq u'$, if i belongs to the set of inliers, the exists $Z_{\mathcal{D}_i,1} \sim Z_{\mathcal{D}_i}|\hat{p}_i = u$ and $Z_{\mathcal{D}_i,2} \sim Z_{\mathcal{D}_i}|\hat{p}_i = u'$ such that $\mathbb{P}[Z_{\mathcal{D}_i,1}] \succeq \mathbb{P}[Z_{\mathcal{D}_i,2}]$*

Indeed, take $i \in [\![1, m]\!]$ and $u \geq u'$ and define $Z_{\mathcal{D}_i,1}$ and $Z_{\mathcal{D}_i,2}$ as in the statement of Lemma 3.5.2.

$$\begin{aligned}\mathbb{P}[\hat{p}_1^m \in A | p_i = u] &= \mathbb{E}_{Z_{\mathcal{D}_i,1}}[\mathbb{P}[\hat{p}_1^m \in A | Z_{\mathcal{D}_i} = Z_{\mathcal{D}_i,1}]] \text{ (Lemma 3.5.2)} \\ &\geq \mathbb{E}_{Z_{\mathcal{D}_i,2}}[\mathbb{P}[\hat{p}_1^m \in A | Z_{\mathcal{D}_i} = Z_{\mathcal{D}_i,2}]] \text{ (Lemma 3.5.1)} \\ &\geq \mathbb{P}[\hat{p}_1^m \in A | p_i = u'] \text{ (Lemma 3.5.2)}\end{aligned}$$

It shows that, when $u \geq u'$ then $\mathbb{P}[\hat{p}_1^m \in A | p_i = u] \geq \mathbb{P}[\hat{p}_1^m \in A | p_i = u']$, which means $\mathbb{P}[\hat{p}_1^m \in A | p_i = u]$ is increasing in u. The PRDS property is satisfied. To complete the proof, the introduced lemmas are proven.

Proof of Lemma 3.5.1. Let be i in $[\![1, m]\!]$ and vectors z, z' and $\overline{z}$ vectors such that $z \succeq z'$. The vectors z, z' are used to define the calibration set related to the p-values $\hat{p}_i$ and $\overline{z}$ is used to define elements of calibrations sets that are not in the calibration set of $\hat{p}_i$. By conditioning on the calibration sets defined by $(z, \overline{z})$ and $(z', \overline{z})$ it gives:

$$\mathbb{P}\left[\hat{p}_1^m \in A | Z_{\mathcal{D}_i} = z, Z_{\overline{\mathcal{D}_i}} = \overline{z}\right] \geq \mathbb{P}\left[\hat{p}_1^m \in A | Z_{\mathcal{D}_i} = z', Z_{\overline{\mathcal{D}_i}} = \overline{z}\right] \tag{3.32}$$

This result comes from the decomposition the following decomposition, for all j in $[\![1, m]\!]$

$$\begin{aligned}\hat{p}_j &= \frac{1}{n}\sum_{k\in\mathcal{D}_j} \mathbb{1}[a(Z_k) \geq a(X_j)] \\ &= \frac{1}{n}\left(\sum_{k\in\mathcal{D}_j\cap\mathcal{D}_i} \mathbb{1}[a(Z_k) \geq a(X_j)] + \sum_{k\in\mathcal{D}_j\setminus\mathcal{D}_i} \mathbb{1}[a(Z_k) \geq a(X_j)]\right)\end{aligned}$$

The conclusion comes from $Z_{\mathcal{D}_i} \succeq Z'_{\mathcal{D}_i}$ which implies $Z_{\mathcal{D}_i\cap\mathcal{D}_j} \succeq Z'_{\mathcal{D}_i\cap\mathcal{D}_j}$.

Since $Z_{\mathcal{D}_j\setminus\mathcal{D}_i} \perp Z_{\mathcal{D}_i}$, Eq. 3.32 can be integrated over $Z_{\overline{\mathcal{D}_i}}$ to give:

$$\mathbb{P}\left[\hat{p}_1^m \in A | Z_{\mathcal{D}_i} = z\right] \geq \mathbb{P}\left[\hat{p}_1^m \in A | Z_{\mathcal{D}_i} = z'\right] \tag{3.33}$$

□

Proof of Lemma 3.5.2. Let $S'_{i,(1)} \leq S_{i,(2)} \leq \ldots \leq S_{i,(n)}$ the order statistics of $(a(Z_{\mathcal{D}_i,1}), \ldots, a(Z_{\mathcal{D}_i,n}))$. Let $S'_{i,(1)} \leq S'_{i,(2)} \leq \ldots \leq S'_{i,(n+1)}$ the order statistics of $(a(Z_{\mathcal{D}_i,1}), \ldots, a(Z_{\mathcal{D}_i,n}), a(X_i))$. And R_i the rank of $a(X_i)$ among these.

$$\left\{(S_{(1)}, \ldots, S_{(n)}) | R_i = k, S'_{i,(1)}, \ldots, S'_{i,(n+1)}\right\} = (S'_{(1)}, \ldots, S'_{(k-1)}, S'_{(k+1)}, \ldots, S'_{(n+1)}) \tag{3.34}$$

Using that R_i is independent of $S'_{i,(1)}, \ldots, S'_{i,(n+1)}$:

$$\left\{(S_{(1)}, \ldots, S_{(n)}) | R_i = k\right\} = (S'_{(1)}, \ldots, S'_{(k-1)}, S'_{(k+1)}, \ldots, S'_{(n+1)}) \tag{3.35}$$

The right-hand side is not increasing with k and $\hat{p}_i = \frac{R_i - 1}{n}$ □

3.7.4 Proof of Proposition 3.3

Lemma 3.5.3. *Let $(p_i)_{1\leq i\leq m}$ be a sequence of m p-values with m_0 true null hypothesis. Suppose i belong to the set of true negative $\mathcal{H}_0$, and D_i the random variable equal to 1 if i is detected by the BH_α procedure.*

- *If the p-values are independent and verify that: $\forall k \in [\![1, m]\!], \mathbb{P}(p_i \leq \frac{\alpha k}{m})$, then:*

$$\mathbb{P}(D_i) = \frac{\alpha\mathbb{E}[R^*]}{m} \tag{3.36}$$

  Where R is the number of hypotheses rejected by BH_α and R^* is the number of hypotheses rejected after p_i is set to 0.

- *If the p-values are empirical p-values using a unique calibration set with cardinality $\nu\frac{m}{\alpha} - 1$, then:*

$$\mathbb{P}(D_i) = \frac{\alpha\mathbb{E}[\tilde{R}^*]}{m} \tag{3.37}$$

  Where $\tilde{R}^*$ is the number of rejected hypothesis by applying BH_α after on $(p'_j)_{1\leq j\leq m}$, where $p'_i = 0$, and for $j \neq i, p'_j = p_j - \frac{1}{n}\mathbb{1}[p_j < p_i]$.

Proof of Lemma 3.5.3. **Proof of the first statement:**
Using, the random variable R representing the number of rejection of BH_α, i is rejected if p_i is below the threshold $\frac{\alpha R}{m}$.

$$\mathbb{P}[D_i] = \mathbb{E}[\mathbb{1}[p_i \leq \frac{\alpha R}{m}]] \tag{3.38}$$

Let $(p'_j)_{1\leq j\leq m}$ be defined by $p'_i = 0$ and $p'_j = p_j$. The conditions of Lemma D6 from [110] are satisfied, it follows:

$$\mathbb{P}[D_i] = \mathbb{E}[\mathbb{1}[p_i \leq \frac{\alpha R(i)}{m}]] \tag{3.39}$$

Where $R(i)$ is the number detection when applying BH_α on $(p'_j)_{1\leq j\leq m}$.

According to the law of total expectation:

$$\mathbb{P}[D_i] = \mathbb{E}[\mathbb{E}[\mathbb{1}[p_i \leq \frac{\alpha R(i)}{m}]|p\backslash p_i]]. \tag{3.40}$$

Since $R(i)$ is measurable is $p\backslash p_i$ and p_i is independent from $p\backslash p_i$, it gives:

$$\mathbb{P}[D_i] = \mathbb{E}[\mathbb{P}_{p_i}(p_i \leq \frac{\alpha R(i)}{m})] \tag{3.41}$$

By hypothesis $\mathbb{P}_{p_i}(p_i \leq \frac{\alpha R(i)}{m}) = \frac{\alpha R(i)}{m}$, then:

$$\mathbb{P}[D_i] = \frac{\alpha}{m}\mathbb{E}[R(i)] \tag{3.42}$$

which conclude the proof of the first statement.

Proof of the second statement:
Let W_i, $C_{i,j}$ defined as:

$$W_i = (\{s_1, ..., s_n, s_{n+i}\}, (s_i, i \in \mathcal{H}_0, i \neq j), (s_i, i \in \mathcal{H}_1)) \tag{3.43}$$

$$C_{i,j} = \frac{1}{n}\left(\sum_{s\in\{s_1,\ldots,s_n,s_{n+i}} \mathbb{1}[s > s_{n+j}] - 1\right) \tag{3.44}$$

1. $p_j = C_{i,j} + \frac{1}{n}\mathbb{1}[s_{n+j} > s_{n+i}]$
2. p_i independent of W_i
3. p_i follow uniform distribution in $\{0, \frac{1}{n}, \ldots, 1\}$

Lemma D.6 from [110] is applied. Let $(p'_j)_{1\leq j\leq m}$ be defined by $p'_i = 0$ and for $j \neq i$, $p'_j = C_{i,j}$. For all j, $p'_j \leq p_j$ and if $p_j > p_i$ then $p'_j = p_j$. Thus, the conditions of the Lemma D.6 from [110]

are verified, which gives:

$$\mathbb{1}[p_i \leq \frac{\alpha R}{m}] = \mathbb{1}[p_i \leq \frac{\alpha \tilde{R}(i)}{m}] \tag{3.45}$$

$$\mathbb{P}[D_i] = \mathbb{E}[\mathbb{E}[p_i \leq \frac{\alpha \tilde{R}(i)}{m} | W_i]] \tag{3.46}$$

Since, $\tilde{R}(i)$ in measurable in W_i and p_i is independent of W_i.

$$\mathbb{P}[D_i] = \mathbb{E}[\mathbb{P}_{p_i}(p_i \leq \frac{\alpha \tilde{R}(i)}{m})] \tag{3.47}$$

By hypothesis, p_i is a empirical p-value with calibration set verified that there exist an integer ν such that $n = \nu \frac{m}{\alpha} - 1$. So according to Corollary 3.1, $\mathbb{P}_{p_i}(p_i \leq \frac{\alpha \tilde{R}(i)}{m}) = \frac{\alpha \tilde{R}(i)}{m}$.

Finally, the second statement is verified with:

$$\mathbb{P}[D_i] = \frac{\alpha}{m} \mathbb{E}[\tilde{R}(i)] \tag{3.48}$$

□

Lemma 3.5.4. *Let X and Y be two random variables. Suppose that $k \mapsto \mathbb{E}[X|Y = k]$ is decreasing, then:*

$$\mathbb{E}[XY] \leq \mathbb{E}[X]\mathbb{E}[Y]. \tag{3.49}$$

Proof of Lemma 3.5.4. Let Z be a random variable that follows the same law than Y but is independent. Since $k \mapsto \mathbb{E}[X|Y = k]$ is decreasing:

$$(Y - Z)(\mathbb{E}[X|Y] - \mathbb{E}[X|Z]) \leq 0 \tag{3.50}$$

$$\mathbb{E}\left[(Y - Z)(\mathbb{E}[X|Y] - \mathbb{E}[X|Z])\right] \leq 0 \tag{3.51}$$

By distributing the product and using that Y and Z follow the same law, this gives:

$$2\mathbb{E}[Y\mathbb{E}[X|Y]] - 2\mathbb{E}[Y\mathbb{E}[X|Z]] \leq 0 \tag{3.52}$$

Finally using $\mathbb{E}[Y\mathbb{E}[X|Y]] = \mathbb{E}[XY]$ and independence of Y and Z:

$$\mathbb{E}[XY] \leq \mathbb{E}[X]\mathbb{E}[Y] \tag{3.53}$$

□

Proof of Proposition 3.3. The mFDR formula is given by

$$mFDR_1^m(p) = \frac{\mathbb{E}[FP_{1,\alpha}^m(p)]}{\mathbb{E}[R_{1,\alpha}^m(p)]}.$$

Let us compute the numerator $\mathbb{E}[FP_1^m(p)]$ value after applying the BH_α. Keep in mind that here the family of true null hypotheses is random, generated by $(A_i)_{1\leq i\leq m}$. In order to meet the conditions of Lemma 3.5.3, it is possible to condition with respect to $\mathcal{H}_0$. Using $D_i = \mathbb{1}[p_i \leq \frac{R}{m}\alpha]$, it appears that

$$FP_1^m(p) = \sum_{i\in\mathcal{H}_0} D_i.$$

Lemma 3.5.3 allows to calculate its conditional expectation:

$$\begin{aligned}\mathbb{E}[FP_{1,\alpha}^m|\mathcal{H}_0] &= \sum_{i\in\mathcal{H}_0}\frac{\alpha}{m}\mathbb{E}[R(i)|\mathcal{H}_0]\\ &= \frac{\alpha m_0}{m}\mathbb{E}[R^*|\mathcal{H}_0].\end{aligned}$$

With R^* the number of rejection where one p-values is set to 0. Integrating with respect to $\mathcal{H}_0$. Then using the fact that $m_0 = |\mathcal{H}_0|$ is measurable with respect to $\mathcal{H}_0$.

$$\begin{aligned}\mathbb{E}[FP_{1,\alpha}^m] &= \mathbb{E}\left[\frac{\alpha m_0}{m}\mathbb{E}[R^*|\mathcal{H}_0]\right]\\ &= \alpha\mathbb{E}\left[\frac{m_0}{m}R^*\right]\end{aligned}$$

Finally, if $\mathbb{E}[R^*|m_0]$ is decreasing, Lemma 3.5.4 gives,

$$\mathbb{E}[FP_{1,\alpha}^m] \leq \alpha(1-\pi)\mathbb{E}[R^*]$$

with $1-\pi = \mathbb{E}\frac{m_0}{m}$, the proportion of data generated by the reference distribution.

□

3.7.5 Proof of Proposition 3.4

Proof of Proposition 3.4. By definition $mFDR_1^m = \frac{\mathbb{E}[FP_1^m]}{\mathbb{E}R_1^m}$, and $R_1^m = FP_1^m + TP_1^m$. With hypothesis the $mFDR$ is equal to α, this gives:

$$\begin{aligned}\alpha &= \frac{\mathbb{E}[FP_1^m]}{\mathbb{E}R_1^m}\\ \alpha &= \frac{\mathbb{E}[FP_1^m]}{\mathbb{E}[FP_1^m + TP_1^m]}\\ \alpha(\mathbb{E}[FP_1^m] + \mathbb{E}[TP_1^m]) &= \mathbb{E}[FP_1^m]\\ (\alpha-1)\mathbb{E}[FP_1^m] &= -\alpha\mathbb{E}[TP_1^m]\\ \mathbb{E}[FP_1^m] &= \frac{\alpha}{1-\alpha}\mathbb{E}[TP_1^m]\end{aligned}$$

Then, the expectation of true positives is expressed using the proportion of false negatives β, the proportion of anomaly π in the m observations, A_i the random variable equal to 1 if the

observation X_i is an anomaly and d_i the random variable equal to 1 if the observation X_i is detected as anomaly :

$$\begin{aligned}\mathbb{E}[TP_1^m] &= \sum_{i=1}^{m} \mathbb{P}[A_i = 1 \text{ and } d_i = 1] \\ &= \sum_{i=1}^{m} \mathbb{P}[A_i = 1]\mathbb{P}[d_i = 1 | A_i = 1] \\ &= m\pi(1-\beta)\end{aligned}$$

Therefore, the $\mathbb{E}[FP_1^m]$ can be expressed as:

$$\mathbb{E}[FP_1^m] = \frac{\alpha m\pi(1-\beta)}{1-\alpha}$$

So the $\mathbb{E}[R_1^m]$ is expressed as follows:

$$\begin{aligned}\mathbb{E}[R_1^m] &= \frac{\alpha m\pi(1-\beta)}{1-\alpha} + m\pi(1-\beta) \\ &= \frac{m\pi(1-\beta)}{1-\alpha}\end{aligned}$$

□

3.7.6 Proof of Corollary 3.5

Proof of Corollary 3.5. All conditions being satisfied Theorem 3.4 gives that:

$$FDR_1^{\infty}(\hat{\varepsilon}_{BH_{\alpha'}}, \hat{p}) = mFDR_1^m(\hat{\varepsilon}_{BH_{\alpha'}}, \hat{\mathbf{P}}_m) \tag{3.54}$$

According to Proposition 3.3, if one of the 3 statements is true:

$$mFDR_1^m(\hat{\varepsilon}_{BH_{\alpha'}}, \hat{p}) \leq (1-\pi)\alpha' \frac{\mathbb{E}[R^*_{\alpha'}]}{\mathbb{E}[R_{\alpha'}]}$$

By hypothesis $\alpha' \frac{\mathbb{E}[R^*_{\alpha'}]}{\mathbb{E}[R_{\alpha'}]} = \alpha$ which allows to conclude.

$$mFDR_1^m(\hat{\varepsilon}_{BH_{\alpha'}}, \hat{p}) \leq (1-\pi)\alpha$$

□

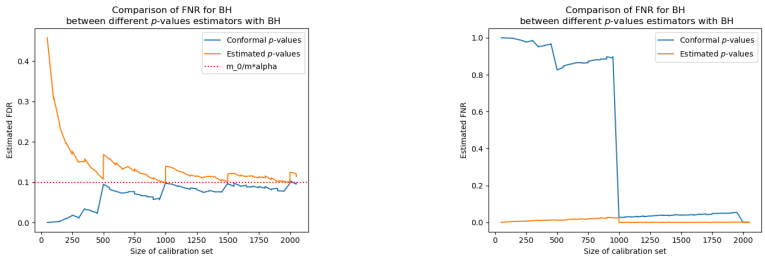
3.8 Figures

3.8.1 Comparison of p-values estimators

The control of the FDR is not achievable using classical multiple testing [14, 133] since the empirical p-value, shown in Definition 3.1, is not super-uniform. Conformal p-value estimator $\check{p}$, shown in Equation 3.2.3.1, verifies the super-uniform property. However, this estimator $\check{p} \geq \frac{1}{n+1}$ has lower power because zero anomalies are detected with thresholds below $\frac{1}{n+1}$.

Figure 3.11 displays the comparison between empirical p-values and conformal p-values using the BH-procedure. As shown in Figure 3.11a, the conformal p-values ensure an upper bound on the FDR at level $\frac{m_0}{m}\alpha$, while the empirical p-values ensure only a lower bound at the same level. Moreover, perfect control are reached for $n = 1000$ and $n = 2000$ with conformal p-values while the control is reached for $n = 999$ and $n = 1999$ with estimated p-values. As shown in Figure 3.11b, the FNR for conformal p-values estimator is always larger than the one for empirical p-values. However for the n points that control the FDR, the FNR values are close.

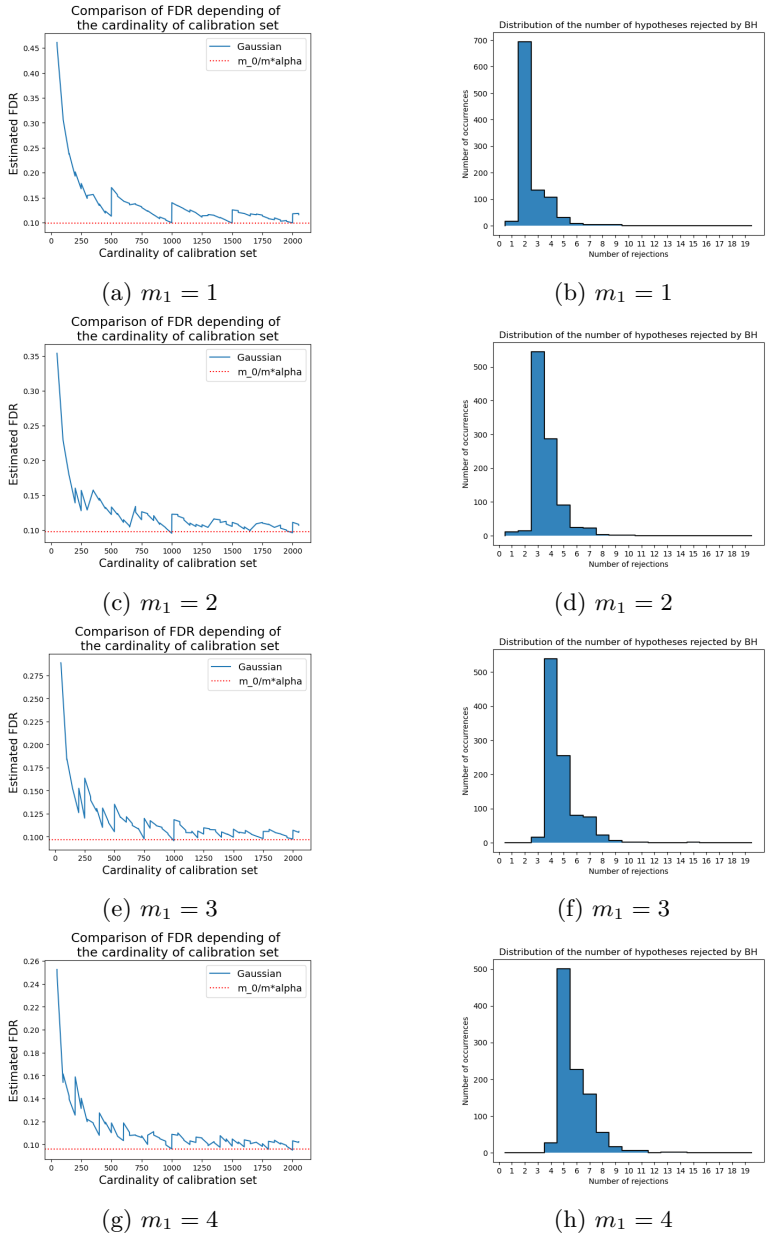
To conclude, the choice between conformal p-values and empirical p-values depends on the calibration set cardinality. Indeed, for calibration set $n = 1000$ the performances are similar. But for other calibration set cardinalities as $n = 1499$ the FDR control are similar but the FNR is better for empirical p-values.



(a) FDR (b) FNR

Figure 3.11: Comparison between p-value estimators using Benjamini-Hochberg

3.8.2 Effect of the number detections by BH on the intermediate drops for the FDR control in Section 3.3.4



(a) $m_1 = 1$ (b) $m_1 = 1$

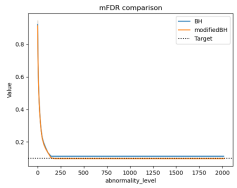
(c) $m_1 = 2$ (d) $m_1 = 2$

(e) $m_1 = 3$ (f) $m_1 = 3$

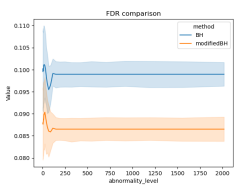
(g) $m_1 = 4$ (h) $m_1 = 4$

Figure 3.12: Effect of the number detections by BH on the intermediate drops for the FDR control

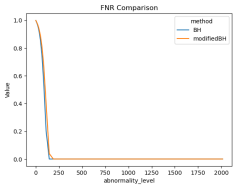
3.8.3 Figures related to experiment of Section 3.4.5.1



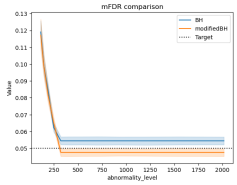
(a) mFDR, $\alpha = 0.1$, $\pi = 0.07$



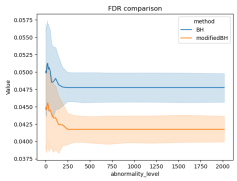
(b) FDR, $\alpha = 0.1$, $\pi = 0.07$



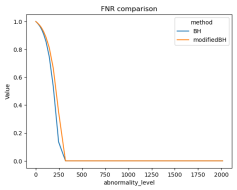
(c) FNR, $\alpha = 0.1$, $\pi = 0.07$



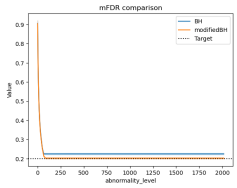
(d) mFDR, $\alpha = 0.05$, $\pi = 0.07$



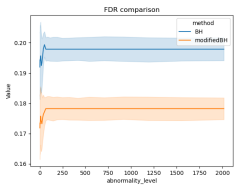
(e) FDR, $\alpha = 0.05$, $\pi = 0.07$



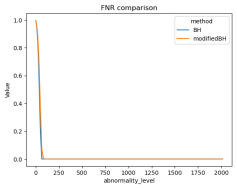
(f) FNR, $\alpha = 0.05$, $\pi = 0.07$



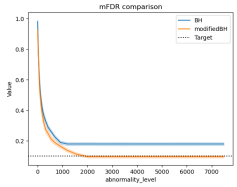
(g) mFDR, $\alpha = 0.2$, $\pi = 0.07$



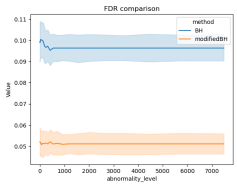
(h) FDR, $\alpha = 0.2$, $\pi = 0.07$



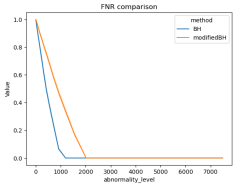
(i) FNR, $\alpha = 0.2$, $\pi = 0.07$



(j) mFDR, $\alpha = 0.1$, $\pi = 0.01$



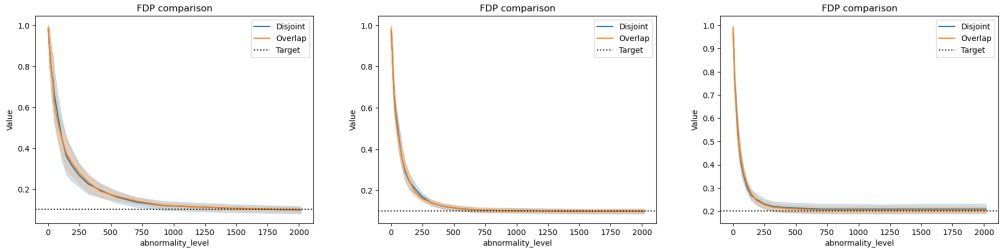
(k) FDR, $\alpha = 0.1$, $\pi = 0.01$



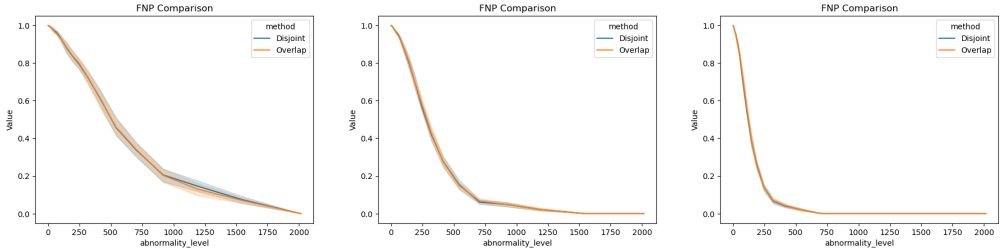
(l) FNR, $\alpha = 0.1$, $\pi = 0.01$

Figure 3.13: Effect of the atypicity level on the mFDR, FDR and FNR, according to different multiple testing procedures.

3.8.4 Figures related to experiment of Section 3.4.5.1



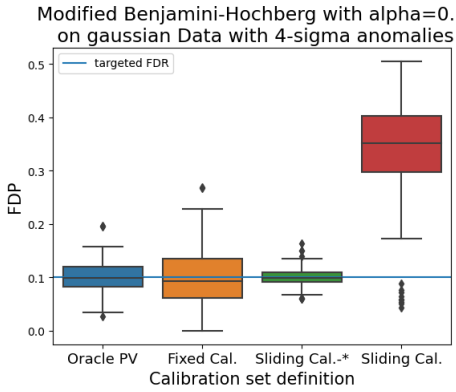
(a) mFDR, $\alpha = 0.1$, $\pi = 0.01$ (b) mFDR, $\alpha = 0.1$, $\pi = 0.02$ (c) mFDR, $\alpha = 0.2$, $\pi = 0.02$



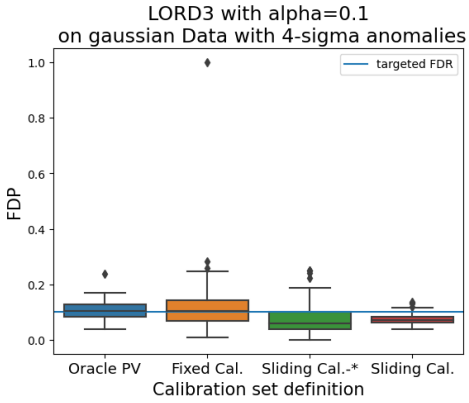
(d) FNR, $\alpha = 0.1$, $\pi = 0.01$ (e) FNR, $\alpha = 0.1$, $\pi = 0.02$ (f) FNR, $\alpha = 0.2$, $\pi = 0.02$

Figure 3.14: Effect of atypicity level on mFDR and FNR, depending on whether detection is on disjoint or overlapping subseries

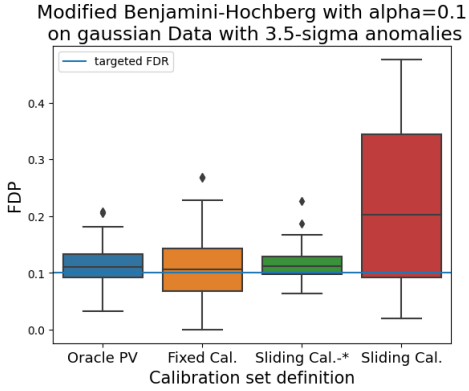
3.8.5 Figures related to the experiment of Section 3.5.4



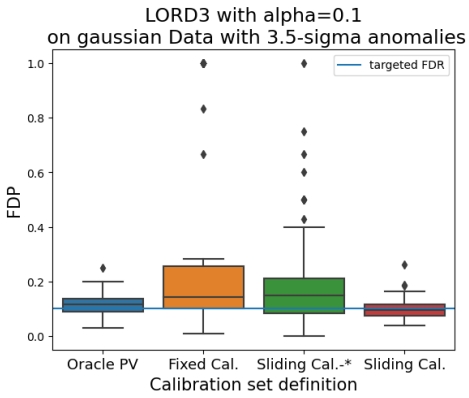
(a) mBH, 4−sigma



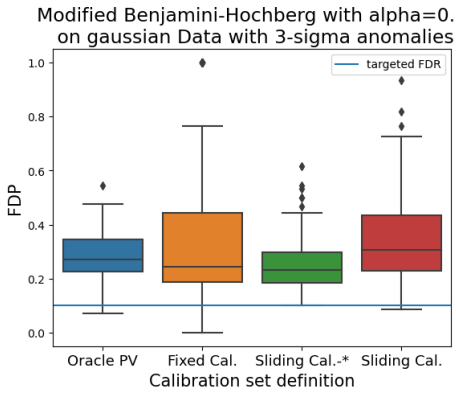
(b) LORD, 4−sigma



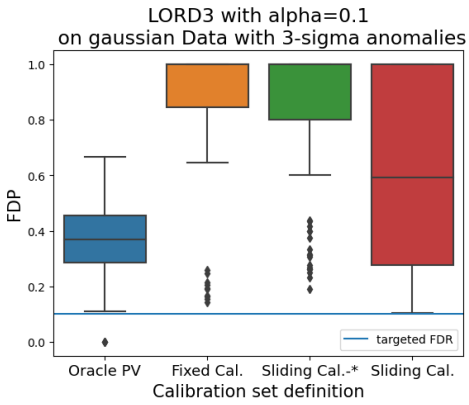
(c) mBH, 3.5−sigma



(d) LORD, 3.5−sigma

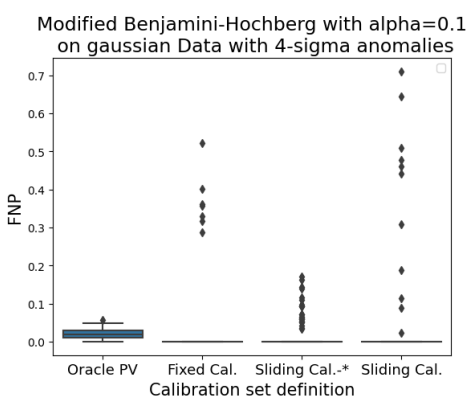


(e) mBH, 3−sigma

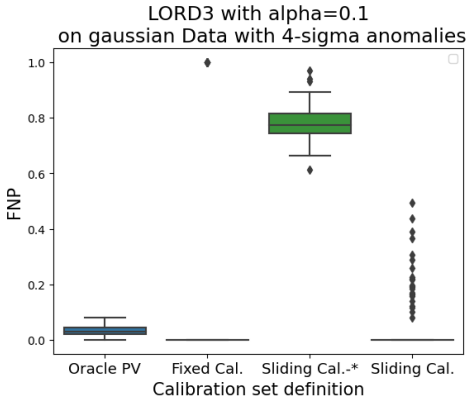


(f) LORD, 3−sigma

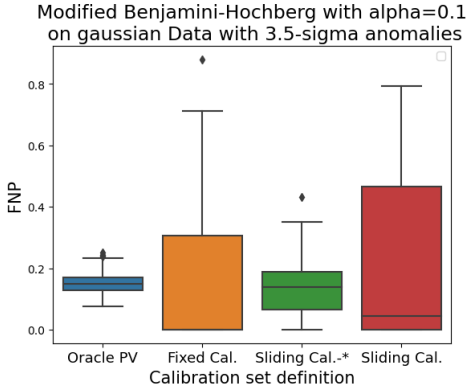
Figure 3.15: Comparison of the FDPs acquired from using different multiple testing procedures, mBH or LORD, and from the way the p-values are calculated, in the case $\alpha = 0.1$.



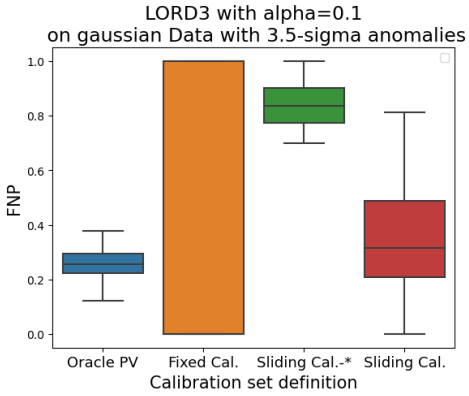
(a) mBH, 4−sigma



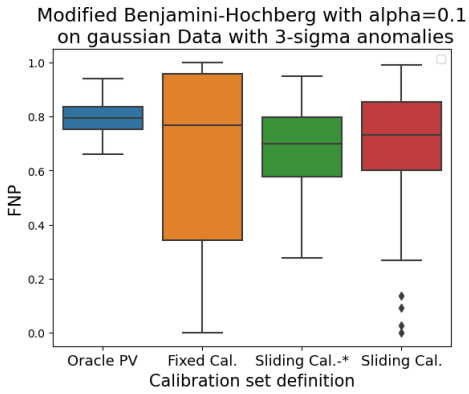
(b) LORD, 4−sigma



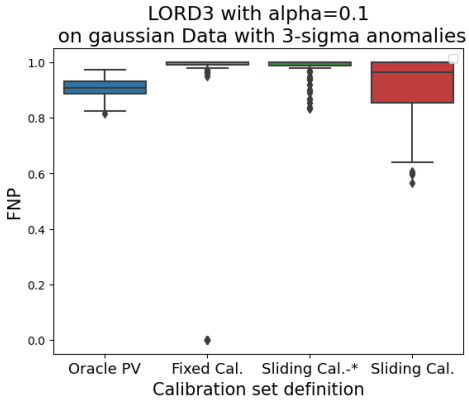
(c) mBH, 3.5−sigma



(d) LORD, 3.5−sigma

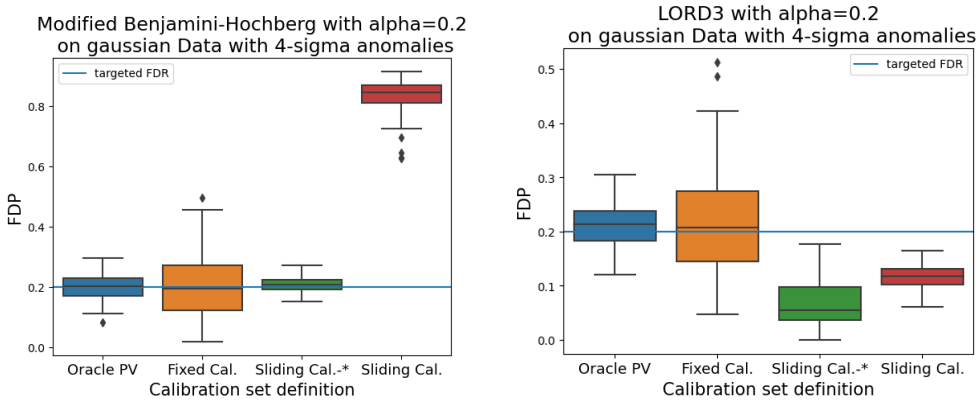


(e) mBH, 3−sigma



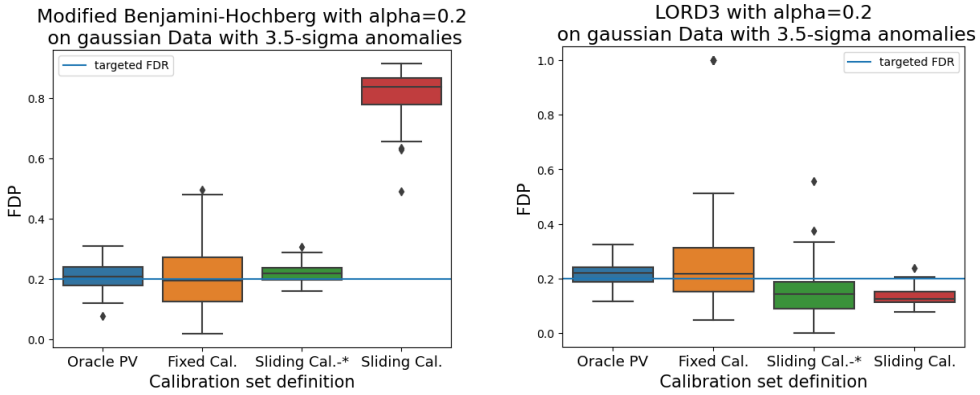
(f) LORD, 3−sigma

Figure 3.16: Comparison of the FNPs acquired from using different multiple testing procedures, mBH or LORD, and from the way the p-values are calculated, in the case $\alpha = 0.1$.



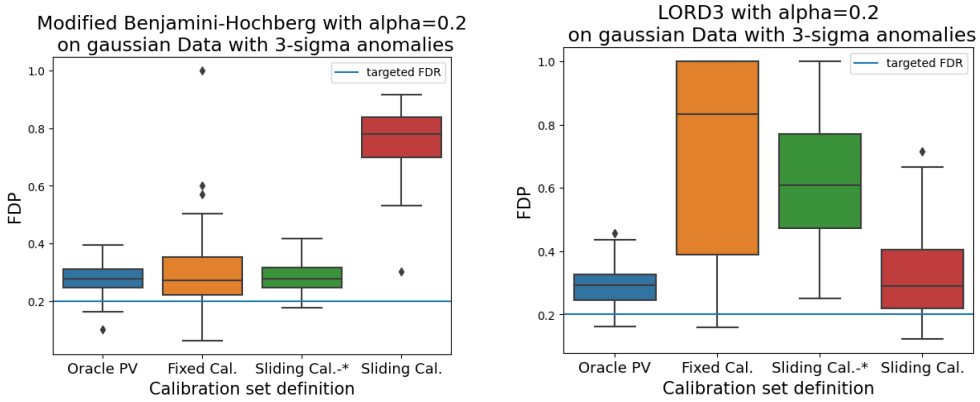
(a) mBH, 4−sigma

(b) LORD, 4−sigma



(c) mBH, 3.5−sigma

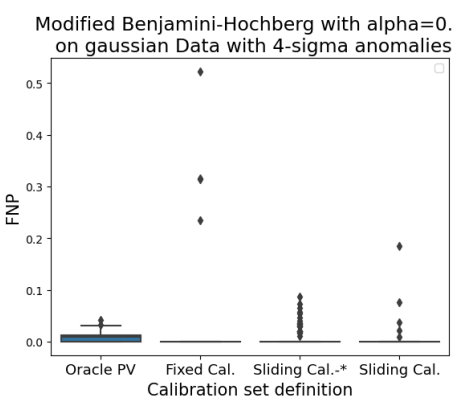
(d) LORD, 3.5−sigma



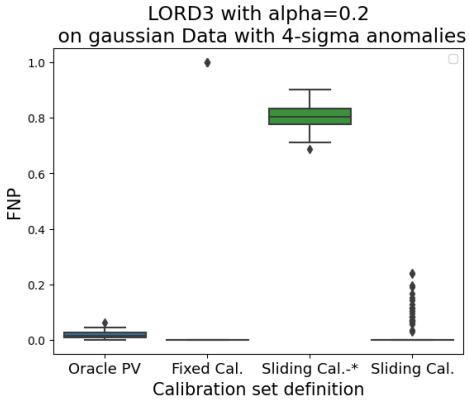
(e) mBH, 3−sigma

(f) LORD, 3−sigma

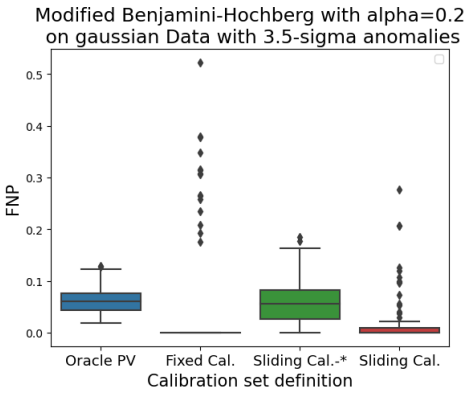
Figure 3.17: Comparison of the FDPs acquired from using different multiple testing procedures, mBH or LORD, and from the way the p-values are calculated, in the case $\alpha = 0.2$



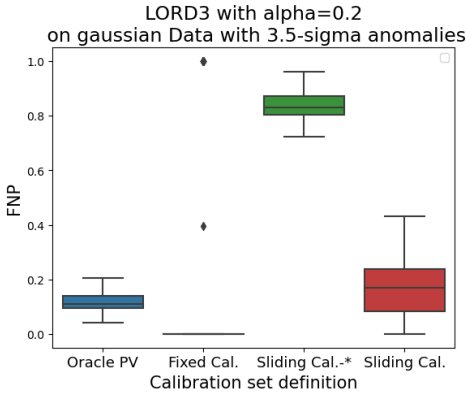
(a) mBH, 4−sigma



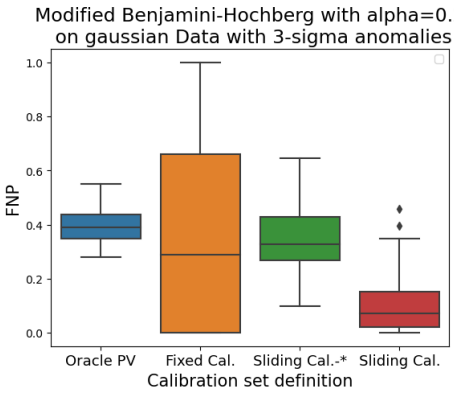
(b) LORD, 4−sigma



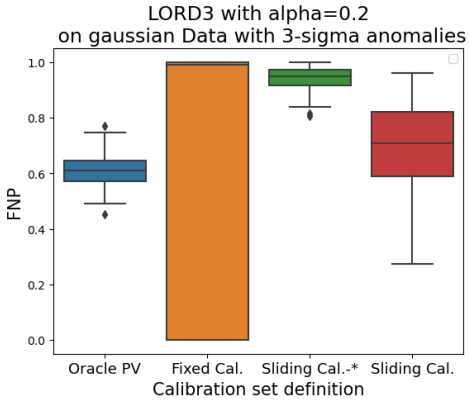
(c) mBH, 3.5−sigma



(d) LORD, 3.5−sigma



(e) mBH, 3−sigma



(f) LORD, 3−sigma

Figure 3.18: Comparison of the FNPs acquired from using different multiple testing procedures, mBH or LORD, and from the way the p-values are calculated, in the case $\alpha = 0.2$.

Chapter 4

Breakpoint based Anomaly Detection

This final chapter introduces a new anomaly detector that relies on breakpoint detection to adapt to a change in reference behavior. This chapter incorporates the results published in "Breakpoint based online anomaly detection" [96]. It begins with a presentation of the new anomaly detector. It is shown that the FDR control results presented in the previous paper are extended with this new detector. Each detector component is studied separately to optimize performance. The anomaly detector is empirically evaluated in depth to assess its capabilities and limitations.

4.1 Introduction

As seen in Chapter 1, the limitation of main Machine Learning based anomaly detector is that the reference model is learned only once on the historical dataset, which assumes that the reference of the time series is the same over time. However, there are data drifts where the reference behavior of the time series changes. If the model is not updated, the data points observed after the drift are detected as false positives. To overcome this problem, most popular strategies consist of periodically retraining the model on a fixed-length window of data. Others use a sliding window of fixed length to continuously learn the reference. For example, Random Cut Forest [71] is a method inspired by Isolation Forest and adapted to real time. DiLOF [122] adapts LOF for real time. Periodic retraining and fixed length sliding windows do not account for the true dynamics of the time series.

This chapter introduces a new anomaly detector that can update the learned reference behavior in an online context. As shown in Figure 4.1, the main idea is to use a breakpoint detector to detect changes in the reference behavior of the time series. Breakpoints are the points at which a property of the time series changes. Between two breakpoints, the data form a homogeneous segment whose characteristics are easy to learn. After detecting the breakpoints in the time series, an atypicity score can be constructed by measuring the conformity of each point to its segment. The final step is to classify as anomalies the points with an atypicity score that is too high. Note that unlike the proposals cited in [3, 52], the breakpoints do not correspond to anomalies, but to changes in the reference distribution. The use of a breakpoint detector introduces new difficulties, which are addressed in this chapter. First, the detection of a breakpoint may be

delayed, leading to temporary errors in segment assignment. Second, when a segment contains few points, it is difficult to estimate its behavior, generating anomaly detection errors. In an online context, this is particularly the case when points are observed just after a new breakpoint. This chapter responds to these difficulties by assigning a confidence score to the estimation made by the detector. This score is used to judiciously select the estimates to be updated when their assigned confidence is too low. This confidence score is learned from a historical data set.

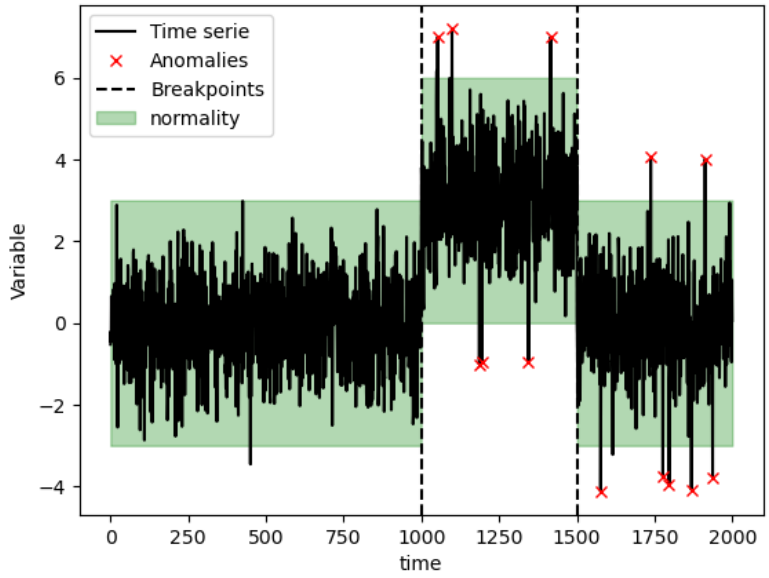


Figure 4.1: Anomaly detection based on breakpoints.

The anomaly detector presented in this chapter comes with theoretical guarantees. In Chapter 3, a new strategy has been designed in the online context to control the FDR for stationary series using a modified version of Benjamini-Hochberg applied to subseries. In this chapter, this work is extended to the nonstationary case.

The main contributions of this chapter are summarized as follows:

- A versatile online anomaly detector based on breakpoint detection is built to adapt to changes in the reference behavior of the time series. Each component of the detector is studied in depth to provide the best possible parameters and improve the performance of the anomaly detector.
- The detector is theoretically studied to demonstrate its ability to control the FDR of the entire series at a level α, under ideal hypotheses.
- The notions of active set and calibration set are introduced to deal with the difficulties of the online nature of the anomaly detector.
- The anomaly detector is empirically evaluated in numerous scenarios to determine its capabilities and limitations.

In Section 4.2, the problem of anomaly detection on piecewise iid time series is introduced, and some challenges related to non-stationarity and uncertainty in estimating breakpoint positions

in the online context are raised. In Section 4.3, the anomaly detector is described and the main theorems are presented. The following sections present the detector components in more detail. The breakpoint detector is described in Section 4.4. While detecting breakpoints, a good scoring function is needed to filter the anomalies. This question is discussed in detail and illustrated with experiments in Section 4.5. In addition, the online nature of anomaly detection makes the decision of an abnormal status much more difficult. Solutions to deal with the uncertainty of an abnormal state are discussed in Section 4.6. Thanks to results presented in Chapter 3 on how to better control the FDR, Section 4.7 integrates these results to have an optimal p-value and threshold selection used in the anomaly detector. Finally, multiple experiments and numerical results are elaborated in section 4.9.

4.2 Problem Setting

This section introduces the problem of anomaly detection in time series containing breakpoints, it explains why it differs from the iid anomaly detection problem and why it cannot be solved with an anomaly detector that does not consider the breakpoints.

4.2.1 Modeling of the problem

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, with Ω the set of all possible outcomes, $\mathcal{F}$ a σ-algebra on Ω and $\mathbb{P}$ a probability measure on $\mathcal{F}$. Assume a realization of the independent random variables $(X_t)_{t\geq 1}$, with X_t taking values in a set $\mathcal{X}$ for all t. $T \in \mathbb{N} \cup \{\infty\}$ is the length of the time series. Normality is a concept that is dependent on a context that changes over time. The instants at which the reference distribution changes are called breakpoints. Supposing there are $D-1$ breakpoints where $D \in \mathbb{N} \cup \infty$, the position of the breakpoints is noted $\tau = (\tau_1, \ldots, \tau_{D+1}) \in [1, T]^{D+1}$. The conventions $\tau_1 = 1$ and $\tau_{D+1} = T+1$, which are not real breakpoints, are used to simplify the notation. To model these different reference behaviors, several reference probability distributions are introduced and noted $\mathcal{P}_{0,i}$. For each segment i in $[\![1, D]\!]$, for each point t in this segment $[\![\tau_i, \tau_{i+1} - 1]\!]$, the observation X_t is called "normal" if $X_t \sim \mathcal{P}_{0,i}$. Otherwise X_t is an "anomaly". Between two consecutive breakpoints, all "normal" observations are generated by the same law defining a homogeneous segment. The time series (X_t) is piecewise stationary.

As illustrated in Figure 4.2, an observation X_t is an anomaly if it is not generated from the reference distribution corresponding to the current segment. Figure 4.2 shows two anomalies detected in the second segment between breakpoints τ_2 and τ_3. Four anomalies have been detected in the last segment 3.

The aim of an online anomaly detector is to find all anomalies among the new observations along the time series $(X_t)_{t\geq 1}$: for each instant $t > 1$, a decision is taken about the status of X_t based on past observations: $(X_u)_{1\leq u\leq t}$. The control of the FDR at a targeted level α can be expressed by $FDR_1^T \leq \alpha$. In the following, the construction of an anomaly detector that controls the FDR at a desired level while minimizing the FNR is studied, in the case of piecewise stationary time series.

4.2.2 Online anomaly detection in piecewise stationary time series

The aim of this section is to highlight the challenge of developing a suitable anomaly detector for the nonstationary series described in Section 4.2.1. First, a generic anomaly detector tailored to the stationary case is described. Then, it is modified to be adapted to the presence of breakpoints in the time series.

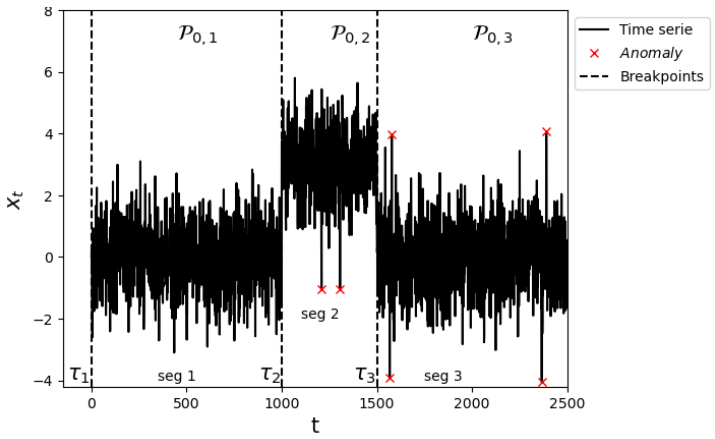


Figure 4.2: Illustration of piecewise stationary time series.

Starting point: anomaly detection in stationary time series Usually, to retrieve anomalies, a unique probability distribution $\mathcal{P}_0$ is considered as the reference distribution assuming no breakpoint in the time series data. Anomalies are defined by observations not generated under the reference distribution: $X_t \nsim \mathcal{P}_0$. In Section 3.2.1, the following general online anomaly detector description was suggested. It uses multiple testing ideas from [111] and the online context from [100]. Unlike the previous chapter, the sets involved at each step are specified. This online detector relies on the following notions:

- An atypicity score a to compare the observation X_t from a *training set* $\mathcal{X}^{train} = \{X_1, ..., X_q\}$ generated by $\mathcal{P}_0$. The more X_t deviates from the points in the training set, the more the abnormality score $s_t = a(X_t, \mathcal{X}^{train})$ is high.

- A p-value estimator $\hat{p}$, based on a *calibration set* of scores $\mathcal{S}^{cal} = \{s_{t-m-n}, ..., s_{t-m}\}$ containing scores of data points generated from $\mathcal{P}_0$, to estimate the p-value, $\hat{p}_t = \hat{p}(s_t, \mathcal{S}^{cal})$. In the online context, the calibration set can change over time.

- The value of the threshold ε can be chosen either as a fixed value for all p-values or to be data driven for subseries of p-values, called the test set. Data driven threshold allows better control of the number of false positives through the False Discovery Rate (FDR). $\hat{\varepsilon}_t = \hat{\varepsilon}(\{p_{t-m+1}, \ldots, p_t\})$

Usually, the training set and calibration set are either chosen from the start of the time series labeled with anomalies or evolve over time using sliding windows. When the training set cannot be labeled, a robust atypicity score is required. An example of a training set, calibration set and test set, in the context of online anomaly detection is shown in the following:

$$\underbrace{X_1, \ldots, X_q}_{\text{Training set}}, \ldots \underbrace{X_{t-n-m}, \ldots, X_{t-n}}_{\text{Calibration set}}, \underbrace{X_{t-m}, \ldots, X_t}_{\text{Test set}}$$

For each new observation X_t, the function a is used to get the atypicity score, trained on the training set. The value of the score cannot be interpreted directly because the distribution of the scores under $\mathcal{H}_0$ is unknown. So its p-value is estimated using the calibration set. The more the data point is atypical, the closer the p-value is to 0.

The next section discusses the reason why this anomaly detector cannot be applied in case of time series containing breakpoints. Indeed, the definitions of training and calibration used have to be reconsidered.

Training, calibration and test sets for piecewise stationarity time series Suppose the strategy used for stationary data is applied to a time series where a shift in the mean of the reference distribution occurs. Before the first shift, there are no differences with the stationary case. After the shift, all data points appear as anomalies when using the scoring function trained on the initial training set based on data before the shift. To adapt to the shift, the training and the calibration sets have to be rebuilt on the new segment of data in order to reapply the anomaly detector.

$$\underbrace{X_1, \ldots, X_{\tau_1}}_{\text{Segment 1}}, \underbrace{\underbrace{X_{\tau_1+1}, \ldots, X_{\tau_1+q}}_{\text{train}}, \underbrace{X_{\tau_1+q+1}, \ldots, X_{\tau_1+q+n}}_{\text{calibration}}, \underbrace{X_{\tau_1+q+n+1}, \ldots, X_t}_{\text{test}}}_{\text{Segment 2}}$$

However, it would take a lot of time to gather enough data for the training and calibration sets. This is the reason why two improvements are suggested. The first improvement in the case where the score is stationary across different segments, data for the calibration set can be taken from previous segments. For example, suppose the shift occurs in the mean and the score is the z-score: $(x - \mu)/\sigma$.

$$\underbrace{\underbrace{X_1, \ldots, X_q}_{\text{train}}, \underbrace{X_{q+1}, \ldots, X_{q+m}}_{\text{calibration}}}_{\text{Segment 1}}, \ldots, \underbrace{\underbrace{x_{\tau_1+1}, \ldots, X_{\tau_1+q}}_{\text{train}}, \underbrace{X_{\tau_1+q+1}, \ldots, X_t}_{\text{test}}}_{\text{Segment 2}}$$

If the scoring function is robust to the presence of anomalies inside the training set, the training can have anomalies. The whole segment can be used as training set. The test set can be part of the training set, using a leave-one-out strategy. The segment length required for anomaly detection can thus be further reduced, this constitute the second improvement.

$$\underbrace{\underbrace{X_1, ..., X_n}_{\text{calibration}}, \ldots}_{\text{Segment 1 and train}} \quad , \quad \underbrace{\ldots, \underbrace{X_{t-m} \ldots\ldots X_t}_{\text{test}}}_{\text{Segment 2 and train}}$$

4.2.3 The uncertainty of estimations

The setup of the training and calibration sets described in the previous section relies on the knowledge of the breakpoint positions. In practice, neither the number of segments D, nor the positions of the breakpoints τ_i nor the laws of the segments $\mathcal{P}_{0,i}$ are known. All these quantities must be learned using the breakpoint detector and the scoring function to perform anomaly detection.

Moreover, in an online context, the lack of knowledge of the whole series influences a good

estimation of these quantities and has a negative impact on the quality of the detection. With each new observation, different situations may occur: the position of a previous breakpoint may be adjusted or removed, or a new breakpoint may appear. As a result, the segment assigned to a data point changes. These new observations influence the composition of each segment and therefore modify the score value and status assigned to each point, especially is the segment is small. Consequently, the values of quantities associated with a data point X_u change over the time t. To reflect this evolution, a subscript t is added. For example, $\hat{p}_{u,t}$ is the p-value estimated for X_u at time t. Similarly $d_{u,t}$ is the status of the point X_u at time t. Furthermore, the concept of the active set is introduced to collect the last points observed in an online context and whose "abnormal" or "normal" status is uncertain since it may evolve due to the introduction of new data points. This uncertainty arises from the possibility that the segment assigned to a segment may change over time, or from the estimation of scores on small segments.

In the next section, a new anomaly detector based on breakpoint detection is introduced. This detector uses a breakpoint detector and proposes solutions to the difficulties introduced in this section regarding the uncertainties in estimation and FDR control.

4.3 Description of the method

This section introduces the new anomaly detector. First, a high-level description is given. Then its properties are studied in an ideal setting. Finally, the validity of the ideal hypotheses and the procedures to approach them are discussed in Section 4.3.4.

4.3.1 High level description for Breakpoint detection Based Anomaly Detector

Using the various concepts introduced in Sections 4.2.1, 4.2.2 and 4.2.3, the BreaKpoint detection based Anomaly Detector (BKAD) is introduced in Algorithm 2 through the following steps.

1. **Breakpoint detection**: A breakpoint detector estimates the number of segments, $\hat{D}_t$, and the locations of breakpoints, noted $\hat{\tau}(t)_1, \ldots, \hat{\tau}(t)_{\hat{D}_t} + 1$, in the current time series $X_1^t = (X_1, \ldots, X_t)$. The conventions$\hat{\tau}(t)_1 = 1$ and $\hat{\tau}(t)_{\hat{D}_t+1} = t + 1$, which are not real breakpoints, are used to simplify the notation. For more precision, see Section 4.4. Consequently, the segments formed by two consecutive breakpoints are expected to be homogeneous. In particular, the segment formed between the last breakpoint noted $\hat{b}_t$ and the last observed point t is called the current segment. With each new observation, the position of all the breakpoints is estimated again. In this way, a breakpoint estimated at one instant t may disappear the next instant. Thanks to dynamic programming, the computational cost of estimating all breakpoints is limited. For more precision, see Section 4.4.

2. **Active set selection**: At this stage, the points whose status is to be reevaluated are selected. This set of points is called the active set. In the current segment, the points whose confidence in the previously evaluated status is too low (lower than a selected η value) are selected. The status of the other points remains the same as in the previous step. Two types of uncertainty are considered. First, uncertainty about the value of the atypicity score on short-length segments, if the current segment is shorter than the minimal requirement ℓ_η, the active set contains the entire current segment. Second, uncertainty about the location of the breakpoints for observations that are too recent. Otherwise it contains only the last λ_η data points whose segment assignment is uncertain. The values

of ℓ_η and λ_η are derived by $\hat{f}_d$ and $\hat{f}_\tau$. Methods for estimating $\hat{f}_\tau$ and $\hat{f}_d$ are described in Section 4.6.1 and Section 4.6.2.

3. **Calibration set selection**: The calibration set is used to calculate the p-values. Therefore, the calibration set should contain points that are representative of the reference behavior. Ideally, only points from the current segment should be used. But when the current segment doesn't contain enough points, points from other segments are used. To limit the bias caused by the introduction of points from another distribution, segments most similar to the current one are selected. The similarity between segments is measured using the similarity function sim. See Section 4.7 for more details.

4. **Atypicity Score**: As described in Section 4.5, a score $a : \mathcal{X} \to \mathbb{R}$ is a function reflecting the atypicity of an observation X_t, it aims to give a high value to anomalies. It is defined as a non conformity measure to the segment. The Nonconformity Measure $\overline{a}$, is a real valued function $\overline{a}(z, B)$ that measures how different z is from the set B. A nonconformity measure can be used to compare a data point to the rest of the segment.

$$s_{u,t} = a(X_u) = \overline{a}(X_u, Seg_t(u)) \in \mathbb{R}$$

where $Seg_t(u)$ is the unique homogeneous segment that contains X_u, at time t. The NCM must be carefully chosen to be robust to the presence of anomalies in the current segment and to distinguish anomalies even with few points in this segment.

5. **p-value estimator**: The value of the atypicity score cannot be interpreted directly. The atypicity score assigned to a data point is compared with those assigned to the points in the calibration set. The probability of observing a normal data point with an atypicity score $a(X)$ greater than $a(X_t)$ is estimated. This is done using the empirical p-value estimator and the calibration set. See Section 4.8 for more details.

$$\hat{p}_e(s_{u,t}, \mathcal{S}_t^{cal}) = \frac{1}{|\mathcal{S}_t^{cal}|} \sum_{s \in \mathcal{S}_t^{cal}} \mathbb{1}[s > s_{u,t}]$$

6. **Threshold Choice**: In order to control the FDR of the complete time series, the data-driven threshold is calculated from the empirical p-values of the active set. A multiple testing procedure, inspired from Benjamini-Hochberg, is applied to determine this detection threshold. See Section 4.8 for more details. This procedure was introduced in Chapter 3. Abnormal status ($d_{u,t} = 1$) is assigned to data points with a p-value below the threshold.

Algorithm 2 Breakpoints based anomaly detection

Require: Let $T > 0$ be the time series length, $(X_t)_1^T$ be the time series, *breakpointDetection* implements breakpoint detector, η the level of uncertainty, $\hat{f}_\tau$ estimate the probability of segment assignment change and $\hat{f}_d$ are estimate the probability of status change when the breakpoint do not change, sim a similarity function between segments, $\overline{a}$ is a non conformity measure, $\hat{p}_e$ implements the empirical p-value estimator and $\hat{\varepsilon}$ selects the best threshold to be applied.

$\hat{\ell}_\eta \leftarrow \arg\min \left\{ \ell, \hat{f}_\tau(\ell) < \eta \right\}$
$\hat{\lambda}_\eta \leftarrow \arg\min \left\{ \lambda, \hat{f}_d(\lambda) < \eta \right\}$
for $t = 1$ to T **do**
  $\hat{\tau}(t) \leftarrow \text{breakpointDetection}(X_1^t)$ ▷ Detection of the breakpoints
  $b_t \leftarrow \hat{\tau}(t)_D$
  if $t - \hat{b}_t \leq \hat{\ell}_\eta$ **then** ▷ Definition of the active set
    $m_t = t - \hat{b}_t$
  else
    $m_t = min(t - \hat{b}_t, \hat{\lambda}_\eta)$
  end if
  $\mathcal{I}^{active} = \{X_{m_t}, X_{m_t+1}, \ldots, X_t\}$
  for $i = 1$ ito $\hat{D}_t$ **do** ▷ Definition of the calibration set
    for $u = \hat{\tau}_i(t)$ to $\hat{\tau}_{i+1}(t)$ **do**
      $sim_u \leftarrow sim(X_{\tau_i(t)}^{\tau_{i+1}(t)-1}, X_{\hat{b}_t}^t)$
    end for
  end for
  $sortedU = sort([\![1, t]\!], sim)$
  $filteredU = filter(u \in sortedU, d_{u,t-1} = 0)$
  $\mathcal{I}^{cal} \leftarrow \{filteredU_i, i \in [\![1, n]\!]\}$
  $\mathcal{S}^{cal} \leftarrow \{\overline{a}(X_u, Seg(u)), u \in \mathcal{I}^{cal}\}$
  for u in $\mathcal{I}^{active}$ **do** ▷ Computation of the scores
    $s_{u,t} \leftarrow \overline{a}(X_u, Seg(u))$
  end for
  for u in $\mathcal{I}^{active}$ **do** ▷ Estimation of the p-values
    $\hat{p}_{u,t} = \hat{p}_e(s_u, \mathcal{S}^{cal})$
  end for
  $\hat{\varepsilon}_t = \hat{\varepsilon}(\{\hat{p}_{u,t}, u \in \mathcal{I}^{active}\})$ ▷ Estimation of the threshold
  for u in $\mathcal{I}^{active}$ **do**
    if $\hat{p}_{u,t} < \hat{\varepsilon}_t$ **then** ▷ Computation of the status
      $d_{u,t} = 1$
    else
      $d_{u,t} = 0$
    end if
  end for
  for u in $[1, t] \backslash \mathcal{I}^{active}$ **do**
    if $t - \hat{b}_t < m$ and $u \geq \hat{b}_t - m$ **then** ▷ Segment closed
      $d_{u,t} = d_{u,\hat{b}_t}$
    else
      $d_{u,t} = d_{u,t-1}$
    end if
  end for
end for
Output: $(d_{t,T})_{t=1}^T$ boolean list that represent the detected anomalies.

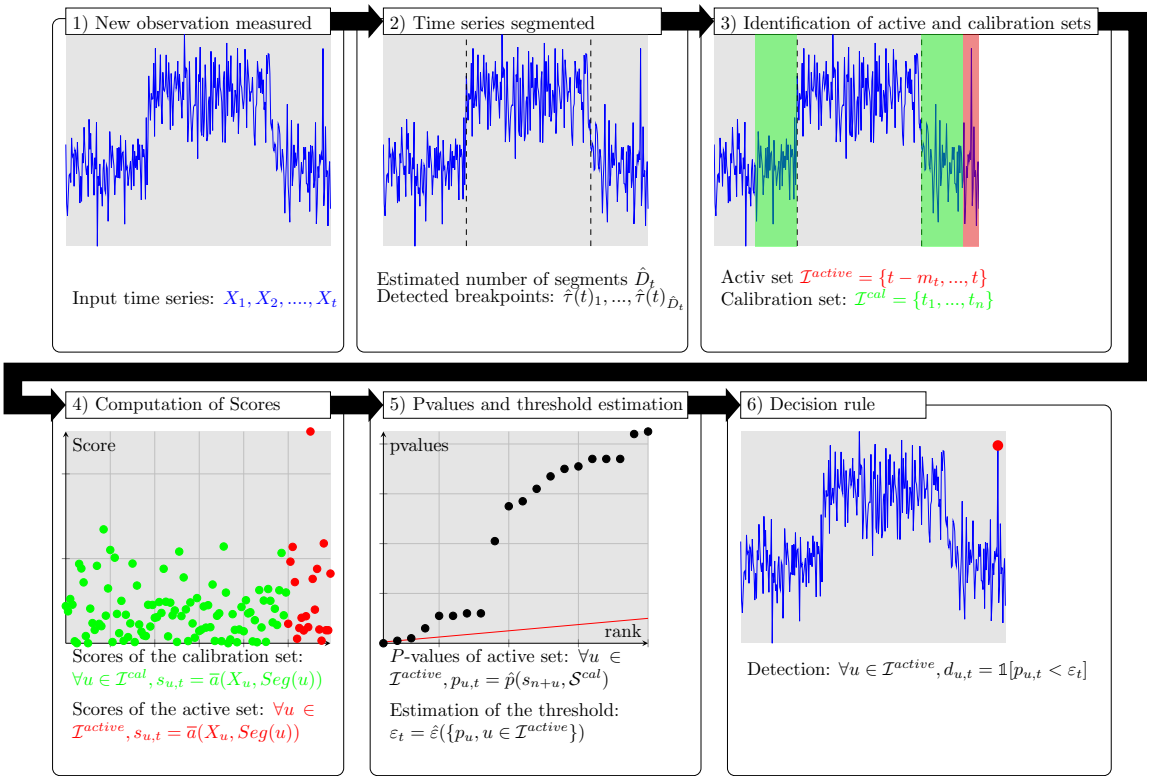


Figure 4.3: Description flow of Algorithm 2.

Algorithm 2 is illustrated in Figure 4.3, the description of the flow is given as the following:

Step 0 (not illustrated): the minimum number of points ℓ_η that a segment must contain to ensure that the atypicity score is estimated with sufficient accuracy is estimated. Similarly, the minimum delay λ_η to ensure with high probability that the assignment of a point to a segment does not change is estimated.

Step 1 : for each time step t, a new data point X_t is observed.

Step 2 : the current time series is segmented $\hat{\tau}(t)$. Each segment is homogeneous.

Step 3 : the data points having a status with low confidence are identified to build the active set. If the current segment is shorter than the minimal requirement ℓ_η, the active set contains the entire current segment. Otherwise it contains only the last λ_η data points whose segment assignment is uncertain. In the case where $\lambda_\eta > \ell_\eta$ it is possible that by going back λ_η points, the current segment is exited. These points belonging to the previous segment do not belong to the active set, and they are reassigned the historical status they had just before observing the current segment. A similarity score is assigned to each point by measuring the similarity between its segment and the current segment, then the n data points with the highest similarity score are forming the calibration set.

Step 4 : The calibration set and active set data points are scored, using the non conformity measure $\overline{a}$.

Step 5 : The p-values of the active set are estimated using the calibration set. The multiple testing

procedure is applied to the active set to obtain the data-driven threshold, in the figure the threshold is chosen using the Benjamini-Hochberg procedure.

Step 6 : A decision is made to give the abnormal status to the data point with a p-value lower than the threshold. For points outside the active set, their status remains the same as in the previous step. If a current segment has less than m points, it is considered a new segment. In this case, the previous segment has just been closed by a new breakpoint. The status of the data point preceding the new breakpoint is updated using the most relevant historical status. This status is the last one before observing the data of the current segment and biasing the status estimation ($d_{u,t} = d_{u,\hat{b}_t}$).

The modularity of our method allows a better adaptation to the diversity of time series. In the following sections, two properties of an ideal version of BKAD are investigated theoretically: its ability to control the FDR at a desired level α, in Section 4.3.2. Then, its ability to deal with uncertainties in the estimation of breakpoints and the value of scores is studied in Section 4.3.3. Finally, the validity of the hypotheses introduced is discussed.

4.3.2 Control of the FDR

In this section it is shown that under ideal conditions, the BKAD algorithm controls the FDR to a desired level α. The various assumptions involved in the control of the FDR are introduced, followed by the presentation of the theorem.

The first hypothesis concerns the generation of the true anomalies. To be able to control the FDR of the whole time series from a control on subseries, it is necessary that the distribution of the subseries is the same as to the rest of the series. The classical assumption is that the data points are generated by a mixture of a reference distribution and an alternative distribution.

Definition 4.1. *[Time series with uniform proportion of anomalies] Let D be the number of segments. Let $\tau_1, \dots, \tau_{D+1}$ be the breakpoint locations. Let $\mathcal{P}_{0,1}, \dots, \mathcal{P}_{0,D}$ be the reference distributions and $\mathcal{P}_{1,1}, \dots, \mathcal{P}_{1,D}$ be the alternative distributions. Let π be the proportion of anomalies. A time series is said to have a uniform proportion of anomalies if (A_u) the series describing anomaly locations and (X_u) the series of observations are generated as follows:*

$$\forall i \in [\![1, D]\!], \forall u \in [\![\tau_i, \tau_{i+1} - 1]\!], \qquad A_u \sim Ber(\pi) \text{ and } X_u \sim \begin{cases} \mathcal{P}_{0,i}, & \text{if } A_u = 0 \\ \mathcal{P}_{1,i}, & \text{if } A_u = 1 \end{cases} \tag{4.1}$$

To correctly detect anomalies, it is necessary to identify breakpoints without error. However, in an online context, breakpoint detection is subject to a certain time delay. To account for these conditions, it is assumed that there may be errors in the most recent observations, but that beyond λ^* data points, all breakpoints are correctly detected.

Definition 4.2. *[Ideal breakpoint detector with delay λ^*] Let τ be the true segmentation. Let λ^* be an integer. Let $\hat{\tau}$ be the breakpoint detector and $\hat{\tau}(t)$ be the estimated segmentation of $X_1, \dots, X_t$. $\hat{\tau}$ is called an ideal breakpoint detector with delay λ^* if the true segmentation is found with delay λ^*.*

$$[\![1, t - \lambda^*]\!] \cap \hat{\tau}(t) = [\![1, t - \lambda^*]\!] \cap \tau \tag{\textbf{Segmentation}}$$

It is desirable that the computed scores be iid to correctly estimate the p-value. For example, if each segment i follows a reference distribution $\mathcal{N}(\mu_i, 1)$, then only the mean changes at the

breakpoints and the oracle score $\tilde{a}(X_u, i) = |X_u - \mu_i|$ is iid. An oracle score is an atypicity score that requires the knowledge of unknown quantities to be calculated. In practice, however, the mean μ_i is not known, so the empirical mean of the segments is used. In doing so, the independence property between the scores is lost. But since $\hat{\mu}_i$ converges to μ_i, it can be assumed that for a segment of sufficient length, the scores can be considered iid. This idea is generalized in the following property.

Definition 4.3. *[Score idd under minimal segment length] Let (X_u) be a time series satisfying Definition 4.1. Assumption* **Score** *assumes that there exist an oracle score $\tilde{a}$, such that $\tilde{a}(X_u, i_u) = s_u$ is iid, where i_u is the number of the segment to which u belongs. Furthermore* **Score** *assumes there is a non conformal measure $\overline{a}$ and an integer ℓ^* such that:*

$$\forall i \in [\![1, D]\!], \forall u_1, u \in [\![\tau_i, \tau_{i+1} - 1]\!], \quad |\tau_i - u_1| \geq \ell^*, \quad \overline{a}(X_u, \{X_{\tau_i}, \ldots, X_{u_1}\}) = \tilde{a}(X_u, i) \tag{\textbf{Score}}$$

To facilitate the theoretical study, an ideal version of BKAD is introduced. The ideal BKAD algorithm is described in the following Definition 4.4. This is an ideal version of the algorithm presented in Algorithm 2, assuming no computational constraints and that the true labels are known when building the calibration set. At each time step t, the scores and p-values are updated with information from the new observed data point, then the $d_{u,t}$ status is changed in two cases:

- for the most recent observations,
- When a new segment is detected, the status of the last points of the closed segment is updated.

In other cases, $d_{u,t}$ keeps its value computed at the previous instant.

Definition 4.4. *[Ideal BKAD] Let λ' and ℓ' be two parameters. Noting $m = max(\lambda', \ell')$, for each t in $[\![1, T]\!]$, the series of scores $(s_{u,t})$, p-values $(p_{u,t})$ and decision $(d_{u,t})$ of the ideal BKAD are calculated as the following.*

First, the sequence of scores is computed as follows. The calculation is presented separately for the segments identified between two breakpoints and for the current segment (identified with his last detected breakpoint $\hat{b}_t$):

$$\forall i \in [\![1, \hat{D}_t - 1]\!], \forall u \in [\![\hat{\tau}_i(t), \hat{\tau}_{i+1}(t) - 1]\!], \quad s_{u,t} = \overline{a}(X_u, \{X_{\hat{\tau}_i(t)}, \ldots, X_{min(\hat{\tau}_{i+1}(t)-1, t-m)}\}) \tag{4.2}$$

$$\forall u \in [\![\hat{b}_t, t]\!], \quad s_{u,t} = \overline{a}(X_u, \{X_{\hat{b}_t}, \ldots, X_{t-m}\}) \tag{4.3}$$

Then, the sequence of p-values is computed as follows: $\hat{p}_{u,t} = \hat{p}_e(s_{u,t}, \mathcal{S}_t)$. With $\mathcal{S}_t$ be the calibration at time step t and computed as follows:

$$\mathcal{S}_t = \{s_{h(t-m,1),t}, \ldots, s_{h(t-m,n),t}\} \tag{4.4}$$

The calibration set is a sliding window containing the n previous scores generated according to the reference distribution. For each t and i, $h(t, i)$ gives the i-th observation lower than t that satisfies the $\mathcal{H}_0$ hypothesis.

Finally, $(d_{u,t})$ the series of decisions, is computed as follows:

- *The status of the most recent observations is updated:*

$$\forall u \in [\![max(t-m, \hat{b}_t), t]\!], \qquad d_{u,t} = \mathbb{1}[p_{u,t} < \hat{\varepsilon}(p_{t-m,t}, \dots, p_{t,t})] \tag{4.5}$$

- *If needed, the status of the last points of the previous segment is updated:*

$$\forall u \in [\![\hat{b}_t - m, \hat{b}_t - 1]\!], \qquad d_{u,t} = \mathbb{1}[p_{u,t} < \hat{\varepsilon}(p_{\hat{b}_t-m,t}, \dots, p_{\hat{b}_t-1,t})] \tag{4.6}$$

- *The status of other data points remains unchanged.*

$$d_{u,t} = d_{u,t-1} \tag{4.7}$$

The detector associates each observed data point X_u with a status $d_{u,t}$ that can evolve according to the number of observed points t. The goal of the ideal detector is that the $d_{u,t}$ value converges to a final $d_{u,T}$ value in a small number of steps and that the final decision series controls the FDR at a desired level α with a minimum of false negatives. Under assumptions **Segmentation** and **Score**, the ideal version of BKAD, described in Definition 4.4, controls the FDP and the FDR of the complete series at the level of the mFDR of the subseries of length m.

Theorem 4.1 (False Discovery Rate convergence)**.** *Let $(X)_{t\geq 1}$ be a time series of infinite size with uniform proportion of anomalies π as stated in Definition 4.1. It is assumed that assumptions* **Segmentation** *and* **Score** *are verified. Applying the ideal BKAD with $\lambda' = \lambda^*$ and $\ell' = \ell^*$ on (X_t), and noting R_a^b the number of rejections on a subset $[a, b]$ and FP_a^b the number of false positives on the same subset.*

$$R_a^b = \lim_{t\to\infty} \sum_{u=a}^{b} d_{u,t}$$
$$FP_a^b = \lim_{t\to\infty} \sum_{u=a}^{b} (1 - A_u) d_{u,t}$$

Then, the FDP, computed as $FDP_1^t = \frac{FP_1^t}{R_1^t}$, and the FDR, computed as $\mathbb{E}[\frac{FP_1^t}{R_1^t}]$, converge and their limit can be calculated as follows:

$$\lim_{t\to\infty} FDR_1^t = \lim_{t\to\infty} FDP_1^t = mFDR_1^m \tag{4.8}$$

The proof of this theorem is given in Section 4.12.1. It follows from this theorem that to control the FDP at a desired α level, it is sufficient to control the mFDR on a subseries of length m at the same level. According to Section 3.4.3, the modified BH procedure allows to control the mFDR if the p-values in the subseries are calculated with a unique calibration set, as stated in [110].

Corollary 4.1. *Under the same notations and assumptions as Theorem 4.1, let m and ν be two integers, let n and α' defined by:*

$$\alpha' = \frac{\alpha}{1 + \frac{1-\alpha}{m\pi}} \text{ and } n = \nu m/\alpha' - 1$$

the threshold procedure is the Benjamini-Hochberg procedure with level α', also called the modified Benjamini-Hochberg procedure introduced in Definition 3.6.

The number of data points detected as anomaly by BH in $[\![1, m]\!]$ is noted R_1^m. Similarly, $R_1^m(u)$ represents the number of data points detected as anomaly, when $\hat{p}_{u,t}$ is replaced with 0. Assuming that the **Power** *and* **Heuristic** *assumptions hold (for more precision refer to Section 3.4.3):*

$$\mathbb{E}[R_1^m] \approx \frac{m\pi}{1-\alpha}. \text{ and } \mathbb{E}[R_1^m(u)] = \mathbb{E}[R_1^m] + 1 \tag{4.9}$$

then the FDP of the complete time series is controlled almost surely at the level α:

$$\lim_{t\to\infty} FDR_1^t = \lim_{t\to\infty} FDP_1^t = (1-\pi)\alpha \tag{4.10}$$

From Corollary 4.1, the modified Benjamini-Hochberg procedure introduced in Definition 3.6 allows to control the FDR at the desired level α. To maximize the performance of the anomaly detector, it is important to carefully choose the cardinality of the calibration set. Indeed, n must be of the form $\nu m/\alpha - 1$ to ensure FDR control, as shown in Section 3.3.2. Section 3.4.3 conducts experiments to test the validity of the assumptions of Eq. 4.9. The FDP control is stricter and more interesting in practice than the FDR control because it is obtained for each individual time series. However, all these controls are obtained for time series of infinite length and are therefore not strictly applicable in practice. In this chapter, the focus will be on the FDR, which will be shown experimentally to be controllable even for series of finite length.

FDR control by the anomaly detector is an important property. This result guides the choice of the threshold selection procedure and the tuning of the calibration set cardinality in Algorithm 2. However, **Segmentation** and **Score** are strong assumptions, it is not possible to get perfect estimations, the following Section 4.3.3 studies the uncertainty of the estimations.

4.3.3 Manage uncertainty of estimations

This section shows how BKAD minimizes the impact of uncertainty on estimations in an online context. First, the uncertainty caused by the online context of the estimations is described mathematically. It is then shown that the ideal version of BKAD can control errors due to estimation uncertainty.

4.3.3.1 Definition of confidence score

In the previous section, the λ^* parameter fully characterizes the uncertainty associated with breakpoint estimation. All points within λ^* last observed points are potentially assigned to the wrong segment. Also, ℓ^* characterizes the uncertainty of the score estimation. All points having their score estimated with less than ℓ^* points may be assigned to a wrong status due to an error in score estimation. If a status is evaluated under uncertain breakpoint location or score estimation, it may need to be re-evaluated as additional data points are observed. In practice, there is no λ^* delay that guarantees that all breakpoints are detected with perfect accuracy. Observation of a new data point may cause the location of a breakpoint to be updated, a new breakpoint to be detected, or a breakpoint to be deleted. This can cause a point X_u to change its assigned segment over time. However, the probability of a point changing its assigned segment decreases with t. Similarly, there is no ℓ^* length that guarantees that the value of the oracle's atypicity score is known with perfect accuracy. Each time a point is added to a segment, the value of the

score is updated, which may change the status of the point X_u. These uncertainties in breakpoint locations and score values lead to uncertainty in the "abnormal" or "normal" status.

Starting from the observation that in an online context, where quantities are estimated knowing only part of the time series, it is impossible to estimate the quantity more accurately than knowing the whole time series. For each data point X_u the oracle status is introduced, noted $\tilde{d}_u$. This is the status that X_u would have been given by BKAD, assuming that the breakpoint locations and score values were estimated with knowledge of the entire time series.

Definition 4.5 (Oracle status)**.** *The oracle status, noted $\tilde{d}_u$, is the status of X_u under the hypothesis that the entire time series is known. Therefore, the breakpoint locations are estimated using the entire time series, and the atypicity score values are estimated using the entire segments. With T, the length of the full time series is potentially infinite.*

$$s_{u,T} = a(X_u, \{X_{\hat{\tau}_i(T)}, \dots, X_{\hat{\tau}i+1(T)}\}) \tag{4.11}$$

$$\tilde{d}_u = \mathbb{1}\left[\hat{p}_e(s_{u,T}, S_u) < \varepsilon(\hat{p}_e(s_{u,T}), \dots, \hat{p}_e(s_{u+m,T}))\right] \tag{4.12}$$

The oracle status defines the confidence score associated with an estimated status. It is the probability that the estimated status is the same as the oracle status.

Definition 4.6 (Confidence Score)**.** *The confidence score $c_{u,t}$ assigns to the decision made for the data point X_u, at time t, the probability that it remains the same under the oracle status*

$$c_{u,t} = \mathbb{P}\left[d_{u,t} = \tilde{d}_u\right]$$

Now that the confidence score associated with a status has been defined, the next step is to ensure a control on it.

4.3.3.2 Control the confidence in the final decision

As described in Definition 4.4, the status of each data point in the current segment is calculated as follows in three steps: Let $\hat{b}_t$ be the last breakpoint, :

1. For all u in $[\![\hat{b}_t, t]\!]$ compute the atypicity score $s_u = \overline{a}(X_u, \{X_{\hat{b}_t}, \dots, X_{t-m})$.
2. For all u in $[\![\hat{b}_t, t]\!]$ compute the p-value $p_u = \hat{p}_e(s_u, \mathcal{S}_t)$
3. For all u in $[\![\hat{b}_t, t]\!]$ compute the status $d_{u,t} = \mathbb{1}[p_u < \hat{\varepsilon}(p_{u,t}, \dots, p_{u+m,t})]$.

Various situations can lead to a change in the status of the point X_u. Before describing these situations, it is useful to introduce the following events:

- "The status of data point X_u at step t is different than the oracle status"

$$V_{u,t} = \left\{d_{u,t} \neq \tilde{d}_u\right\}$$

- "The segment to which the data point X_u is assigned at time t changes over time"

$$W_{u,t} = \left\{\exists t' > t, \hat{\tau}(t') \cap [\![\hat{b}_t, u]\!] \neq \emptyset\right\} \tag{4.13}$$

First, if a breakpoint is detected between $\hat{b}_t$ and u, as described by the event $W_{u,t}$, this means

that the score associated with X_u has to be computed from a different training set. Similarly, if a breakpoint is detected between u and $u+m$, it means that the data-driven threshold has to be computed from a different subseries. For these reasons, the probability of a point changing its assigned segment $\mathbb{P}\left[W_{u+m,t}\right]$ is of interest. If no breakpoint is detected between $\hat{b}_t$ and $u+m$, the assignment of points u to $u+m$ remains unchanged. This event is recorded in $\overline{W}_{\overline{u},t}$ with $\overline{u} = u+m$. Under the condition of the $\overline{W}_{\overline{u},t}$ event, it is possible for the status to be different from the oracle status if the addition of a data point in the current segment modifies the score value: $s_{u,T} = \overline{a}(X_u, \{X_{\hat{\tau}_i(T)}, \ldots, X_{\hat{\tau}_{i+1}(T)}\})$, with $[\![\hat{b}_t, t-m]\!] \subset [\![\hat{\tau}_i(T), \hat{\tau}_{i+1}(T)]\!]$

To bound the probabilities $\mathbb{P}\left[W_{u,t}\right]$ or $\mathbb{P}\left[V_{u,t}|\overline{W_{\overline{u},t}}\right]$ and to build the active set, some assumptions are made:

Proposition 4.1 (Stationarity)**.** *Let $\eta > 0$.*

- *Assuming $f_\tau : \lambda \mapsto \mathbb{P}\left[W_{t-\lambda,t}\right]$ is decreasing to 0 and does not depend on t.*

  Then, there exists λ_η such that:

$$\forall t \in [\![1,T]\!], \forall u \in [\![1,t]\!], \quad |u-t| \geq \lambda_\eta, \quad \mathbb{P}\left[W_{u,t}\right] \leq \eta. \tag{4.14}$$

  The smallest value respecting this property is noted $\lambda_\eta^\star$.

- *Assuming $f_d : \ell \mapsto \mathbb{P}\left[V_{u,t}|\overline{W_{\overline{u},t}}, \ell_t = \ell\right]$ is decreasing to 0 and does not depend on t. Then, there exist a segment length ℓ_η such that:*

$$\forall t \in [\![1,T]\!], \forall u \in [\![1,t]\!], \quad \ell \geq \ell_\eta, \quad \mathbb{P}\left[V_{u,t}|\overline{W_{\overline{u},t}}, \ell_t = \ell\right] \leq \eta. \tag{4.15}$$

  The smallest value of ℓ_η is noted $\ell_\eta^\star$.

The conclusions of Proposition 4.1 follow directly from the definition of convergence to 0. Before considering the consequences of this proposition in Theorem 4.2, the validity of the assumptions is discussed. The function $f_\tau : \lambda \mapsto \mathbb{P}\left[W_{t-\lambda,t}\right]$ gives the probability that the segment assigned to $X_{t-\lambda}$ changes as a function of the distance λ from the last observation. It is assumed to be decreasing because the probability of missing a breakpoint decreases with the number of points. The function $f_d : \ell \mapsto \mathbb{P}\left[V_{u,t}|\overline{W_{\overline{u},t}}, \ell_t = \ell\right]$ gives the probability of changing the status of a point conditional on the assigned segment remaining unchanged, as a function of the length ℓ of the segment. It is assumed to decrease with the number of points inside the segment. It is assumed that $\mathbb{P}\left[W_{t-\lambda,t}\right]$ and $\mathbb{P}\left[V_{u,t}|\overline{W_{\overline{u},t}}, \ell_t = \ell\right]$ do not depend on t. Since the probability of detecting a breakpoint depends on the position of the actual breakpoint, this assumption can only be verified by assuming that the position of the breakpoints is determined by a stationary process. Furthermore, the probability of detecting a breakpoint depends on the of the shift that occurs in the time series. Therefore, another necessary condition is to assume that at each breakpoint the segment law changes according to transition rules that are the same throughout the series. Assuming that the probabilities $\mathbb{P}\left[W_{t-\lambda,t}\right]$ and $\mathbb{P}\left[V_{u,t}|\overline{W_{\overline{u},t}}, \ell_t = \ell\right]$ do not depend on t, it is possible to use the same model for the entire series. Thus, there is no need to recalculate these probabilities for each observation time.

From this result, the definition of the active set as applied at the start of Algorithm 2 can be deduced. Let $\hat{\lambda}_\eta$ and $\hat{\ell}_\eta$ be estimators of λ_η and ℓ_η. The active set contains the data points whose status needs to be reassessed because the uncertainty associated with estimating the position of breakpoints or the value of scores is too large. As shown in Algorithm 3, the procedure starts by comparing the length ℓ_t of the current segment with the threshold length $\hat{\ell}_\eta$. If the length ℓ_t is lower than this threshold, the whole segment is considered as the active set since the segment

does not contain enough points to estimate the atypicity score with high precision. Otherwise, the segment contains enough points and the source of the status change is segment reassignment. Considering the data points whose distance to the end of the time series is less than $\hat{\lambda}_\eta$, the risk of being reassigned to another segment is high. Consequently, the active set will contain all points that are after the position $t - \hat{\lambda}_\eta$. In the case $\hat{\lambda}_\eta$ is larger than the length of the current segment, the active set will include the current segment. Given m_t the active set cardinality, the active set is equal to:

$$\mathcal{I}^{active} = \{t - m_t + 1, \dots, t\}$$

Algorithm 3 Computation of active set cardinality.

```
if ℓ_t < ℓ̂_η then
    m_t ← ℓ_t
else
    m_t ← min(λ̂_η, ℓ_t)
end if
return m_t
```

The goal is that by re-evaluating only the points in the active set, defined by Algorithm 3 using λ_η and ℓ_η, the final status of a data point will be the same as the oracle status with a high probability. Also, ideally, the active set should be as small as possible so that the status of a data point converges quickly to its final status. The following Theorem 4.2 addresses this issue.

Theorem 4.2 shows that under the conditions of Proposition 4.1, the ideal version of BKAD, applied with λ_η and ℓ_η parameters, controls the proportion of differences between the final status $d_{u,T}$ and the oracle status $\tilde{d}_u$.

Theorem 4.2. *Let η be the confidence threshold, λ_η and ℓ_η are defined as in Proposition 4.1. Let m be the integer defined by $m = max(\lambda_\eta, \ell_\eta)$.*

It is assumed that:

- *the probability to move the latest breakpoint beyond m is lower than η.*

$$\mathbb{P}(\exists t' \hat{b}_{t'} < \hat{b}_t \text{ and } |\hat{b}_{t'} - t'| > m || b_t - t| < m) \leq \eta \tag{4.16}$$

- *The probability of changing the segment assignment depends only on $\lambda_{u,t} = t - u$, the distance between X_u and the end of the time series t. This assumption can be used to calculate the probability of changing the segment assignment within the previous segments (segments that are not the current segment):*

$$\mathbb{P}[\exists t' > t, \hat{\tau}(t') \cap]\hat{\tau}_i(t), u] \neq \emptyset | t - \hat{\tau}(t) = \lambda] = \mathbb{P}[\exists t' > t, \hat{\tau}(t') \cap]\hat{b}_t, u] \neq \emptyset | t - \hat{b}_t = \lambda] \tag{4.17}$$

Then, applying the ideal BKAD as stated in Definition 4.4 with the parameters $\lambda' = \lambda_\eta$ and $\ell' = \ell_\eta$, for each u, the probability that the final status is different than the oracle status is lower than η:

$$\mathbb{P}(d_{u,T} \neq \tilde{d}_u) \leq 3\eta \tag{4.18}$$

Furthermore, introducing the following notation: For all t, for all u, let $\hat{\tau}^{:u}(t) = \{\hat{\tau}_i(t), \hat{\tau}_i(t) < u\}$ and $\hat{\tau}^{u+q_1:}(t) = \{\hat{\tau}_i(t), \hat{\tau}_i(t) > u + q_1\}$.

Assuming that there is a number q_1 such that

$$\forall u, t, \quad \hat{\tau}(t)^{:u} \perp \hat{\tau}^{u+q_1:}(t) \tag{4.19}$$

Then, the proportion of status that are different between the final status and the oracle status is lower than η:

$$\lim_{t \to T} \frac{1}{t} \sum_{u=1}^{t} \mathbb{1}[d_{u,t} \neq \tilde{d}_u] \leq 3\eta \tag{4.20}$$

The proof of Theorem 4.2 can be found in Section 4.13.2.

The results of Theorem 4.2 support the way the active set is built at the beginning of Algorithm 2. However, the true values of λ_η and ℓ_η are not known, so they need to be estimated. This problem is addressed in Section 4.6. Furthermore, this result is for an ideal version of BKAD, which cannot be used in practice. The following section discusses the validity of the various hypotheses.

4.3.4 Discussion of theoretical hypotheses

The previous theoretical results prove that under ideal conditions BKAD allows to detect anomalies with a control on the FDR. Also the strategy consisting in updating only the active set ensures that the final status are the same as knowing the complete time series, with a low proportion of errors. These ideal conditions cannot be verified in practice. The approach in this chapter is as follows: Each component of the BKAD detector is studied to find the best parameters. Then, the detector is empirically tested to see under which conditions it succeeds in detecting anomalies with a control on the FDR. Now, the different assumptions are examined, their validity is discussed, and the properties that the components must verify are deduced.

First, the assumption **Segmentation** cannot be verified. It is impossible to ensure that a breakpoint detector estimates the location of all breakpoints with perfect accuracy, even with a delay of $\lambda > 0$. To get closer to these working hypotheses, a powerful breakpoint detector is needed. BKAD uses KCP [6] because it has several interesting properties: the number of breakpoints is estimated by model selection, it can detect changes in any feature thanks to kernels and it does not require parametric assumptions that are difficult to verify. For more details, see the dedicated Section 4.4.

The assumption that there is an iid oracle score is always verified. Indeed, for each segment i in $[\![1, D]\!]$, the reference distribution is noted $\mathcal{P}_{0,i}$ and for each normal data point of the indices u in this segment: $\tilde{a}(X_u, i) = \mathbb{P}_{X \sim \mathcal{P}_{0,i}}(X \leq X_u)$ follows a uniform distribution $U(0, 1)$. However, there is not always uniqueness of such an oracle score. For example, if the changes occur only in the mean, then by noting μ_i the mean of the ith segment, $|X_u - \mu_i|$ is also an oracle atypicity score that verifies the iid property. In practice, the oracle score has to be estimated correctly, which can be difficult depending on the oracle score. However, it is not possible to verify the property **Score**. Indeed, it is not possible to know the value of the oracle atypicity score with perfect accuracy. To approach this property, one needs a measure of nonconformity $\overline{a}$ that converges as quickly as possible to the value of the oracle atypicity score. As described in Section 4.5, the

non conformity measure must be robust and efficient. As a consequence of the fact that the estimated atypicity scores are not iid, the scores from different segments cannot be rigorously used to construct the calibration set. To limit this issue, the calibration set is built from the segments with the most similar distribution to the one of the current segment. This mechanism is described in Section 4.7.

Another assumption of the ideal BKAD in Definition 4.4 is that the calibration set is built using only normal data points, which requires knowledge of the true labels. In the Algorithm 2, this is obtained by using the labels previously estimated by the anomaly detector. This exposes the calibration set to contamination from undetected anomalies. It can also bias the calibration set by incorrectly removing false positives. This may limit the ability of BKAD to control the FDR. From a theoretical point of view, this leads to a dependency between the calculation of the p-value at time t and the state of the data points at the previous time $t-1$, which complicates the analysis.

Finally, in the previous section, the values of λ_η^* and ℓ_η^* were obtained from the functions f_τ and f_d. These quantities are needed to reduce the uncertainty of the BKAD estimates. In Section 4.6, estimators of f_τ and f_d are introduced.

In the following Sections 4.4- 4.8 the different components of the algorithm are discussed and described in more detail.

4.4 Breakpoint estimation

As described at Section 4.2.1 the time series $(X_t)_{1\leq t\leq T}$ has D segments with breakpoints denoted $\tau_1, ..., \tau_{D+1}$. The segment $X_{\tau_i}^{\tau_{i+1}-1}$ is said homogeneous. Informative features for anomaly detection, such as the mean or the variance, can be extracted if the breakpoints are correctly identified. A good breakpoint detector is important to increase the accuracy of anomaly detection. If a shift is not well detected, the analyzed segment will be heterogeneous and the estimation of the law under $\mathcal{H}_0$ will be biased. If too many breakpoints are detected in a segment while it is homogeneous, the analyzed segments will contain fewer points and the variance of the predictions will be too high. To maximize the performance of the anomaly detection, the number and the locations of breakpoints have to be accurately estimated. The article [163] is a review of existing offline breakpoint detectors. As described with more details in Section 1.2.3, the authors show that a breakpoint detector can be described as an optimization problem, using three notions: a cost function $\mathcal{C}$, a search function on $\mathcal{T}$ and a penalty function pen. The segmentation returned by the breakpoint detector is the one that minimizes the penalized cost function among the explored solutions:

$$\hat{\tau} \in \arg\min_{\tau\in\mathcal{T}} \sum_{i=1}^{D_\tau} \mathcal{C}(X_{\tau_i}^{\tau_{i+1}-1}) + pen(D_\tau) \tag{4.21}$$

In this chapter, the Kernel Change Point (KCP) introduced in [6] is used for its advantages. The kernel-based cost function could be used for any kind of time series, univariate or multivariate, without changing the breakpoint detector. The diversity of time series and breakpoints types are handle throw the choice of the kernel and its hyperparameters. This accuracy is guaranteed by the oracle inequality given in [6]. For a given segmentation $\tau = (\tau_1, \ldots, \tau_{D+1})$ and a kernel

K, the cost is given by:

$$\hat{R}(\tau) = \frac{1}{t}\sum_{u=1}^{t} K(X_u, X_u) - \frac{1}{t}\sum_{i=1}^{D_\tau} \frac{1}{\tau_{i+1} - \tau_i} \sum_{u,v=\tau_i}^{\tau_{i+1}-1} K(X_u, X_v) \tag{4.22}$$

First, the candidate segmentations that minimize the criterion are identified for each possible number of D segments. $\mathcal{T}^D$ is the space of all candidate segmentations with D segments, $\hat{\tau}_{D,t}$ is the best candidate segmentation with D segments and $L_{D,t}$ is the cost associated with this segmentation.

$$L_{D,t} = \min_{\tau \in \mathcal{T}^D} \hat{R}(\tau)$$
$$\hat{\tau}_{D,t} = \arg\min_{\tau \in \mathcal{T}^D} \hat{R}(\tau)$$

To estimate the number of segments and thus the best segmentation, one searches for the segmentation $\hat{\tau}_{D,t}$ that minimizes the penalized criterion described in Eq. 4.21. The penalty function is given by:

$$pen(D_\tau) = r_1 D_\tau + r_2 \log \binom{t-1}{D_\tau - 1} \tag{4.23}$$

where the coefficients r_1 and r_2 are estimated by fitting the penalty function on the estimated cost for over-segmented segmentations [12].

KCP is designed as an offline breakpoint estimator. By using Dynamic Programming, the segmentation costs can be estimated without performing the same computation between time t and $t+1$ as described in [38]. This feature is necessary to be applied in an online anomaly detection. The data driven penalty function enables good accuracy in estimating the number of breakpoints. The breakpoints are detected by solving the optimization problem with the algorithm: D_{max} depends on T, according to [6], it can be chosen equal around: $n/\sqrt{\log n}$. The

Algorithm 4 Dynamic Programming for breakpoint detection.

Require: $T > 0$, (X_t) time series, $\mathcal{C}$ Kernel based cost function, D_{max} maximum segment number explored and *SlopeHeuristic* implement the slope heuristic.
  for $t \in [\![1, T]\!]$ **do**
    for $D \in [\![1, D_{max}]\!]$ **do**
      $L_{D,t} \leftarrow \min_{t' \leq t} L_{D-1,t'} + \mathcal{C}_{t',t}$
      $\hat{\tau}_D(t) \leftarrow \arg\min_{t' \leq t} L_{D-1,t'} + \mathcal{C}_{t',t}$
    end for
    $r_1, r_2 \leftarrow SlopeHeuristic(L)$
    $\hat{D} \in \arg\min_D L_{D,t} + r_1 D + r_2 \log \binom{t-1}{D-1}$
    $\hat{\tau}(t) \leftarrow \hat{\tau}_{\hat{D}}(t)$
  end for
  Output: $\forall t \in [\![1, T]\!], (\tau(t))_{1 \leq t \leq T}$ estimated segmentation at each time step.

main degree of freedom in KCP is the choice of the kernel. Characteristic kernels [59], like the Gaussian kernel, are able to detect any kind of change: shift in the mean, shift in the variance,

shift in the third moment,...

$$K(x,y) = \exp(-\frac{||x-y||^2}{2h^2}) \tag{4.24}$$

However, due to the fact that the number of points within a segment is finite, the performance of a characteristic kernel depends on the choice of hyperparameters. In the case of the Gaussian kernel, the only parameter is the bandwidth h. For changes that occur in the mean, the *median heuristic*, shown in Eq. 4.25, gives good results [61]. Defining a method to select the most relevant kernel for any kind of breakpoint is still an open question.

$$h = \text{median}_{i \neq j}(||X_i - X_j||) \tag{4.25}$$

Breakpoint detection is used to define homogeneous segments. In the next section, the characterization of atypicity in each segment is studied.

4.5 Atypicity score

In this chapter, an anomaly is a data point that does not follow the reference distribution of the segment to which it belongs. To construct an atypicity score that is higher for abnormal points, a point must be compared to the rest of the segment.The Nonconformity Measure (NCM) from [145] is introduced. The Nonconformity Measure $\overline{a}$, is a real valued function $\overline{a}(z, B)$ that measure how different z is from the set B. A nonconformity measure can be used to compare a data point with the rest of the segment. If all points within a segment are generated by the reference distribution, then the Nonconformity Measure provides an atypicity score for this segment.

$$\forall i \in [\![1, D]\!], \quad \forall [\![\tau_i, \tau_{i+1}[\![, \quad a(X_t) = \overline{a}(X_t, \{X_{\tau_i}, \ldots, X_{\tau_{i+1}-1}\} \backslash \{X_t\}) \tag{4.26}$$

The following properties are required to enable the usage of the nonconformity measure to build a good atypicity score:

- anomalies should have higher atypicity score than normal data points.
- the NCM must be robust [156, 138, 176] to the presence of anomalies. The anomalies present in the segment do not affect the value of the returned measure.
- the values returned between different segments must be comparable, so that a p-value can be estimated, with a calibration set containing values from different segments. The iid property of scoring is introduced in Definition 4.3, to formalize this idea.

The property of score stationarity depends on the time series. For example, the z-score with true known mean and standard deviation satisfies the stationarity property only if the changes generated by the breakpoints are shifts in the mean or in the standard deviation. If the change occurs in higher moments, the property is not satisfied. Furthermore, the property is not satisfied for the z-score if the mean and standard deviation need to be estimated. Since the stationarity of the score is difficult to obtain, it is approached with the following strategies:

- Ensure that the segment contains enough points to ensure the convergence of the nonconformity measures. For example, since the mean and variance must be estimated, the z-score is not stationary. However, if these parameters converge to the true mean and standard deviation, then the z-score can be considered stationary once the segments have enough points. The faster convergence is achieved, the easier it is to ensure the stationarity

property. An NCM is said to be efficient when convergence is achieved for a low number of points.

- Instead of using the entire segment, the training set can be built by resampling a specified number of points. It can be used on NCM that are highly dependent on the training set cardinality, like kNN.

- Rather than trying to ensure that the score distribution is identical in each segment, identify the segments with the most similar distribution, as described in Section 4.7.

Many NCMs depend on segment parameters to be estimated, e.g. the z-score requires knowledge of the mean and variance. To satisfy the properties of a good atypicity function, the estimators need to satisfy the following requirements:

1. the estimator should be robust to anomalies in the training set: the estimation should not be affected by the presence of anomalies in the training set.

2. The estimator should be efficient [93]. High precision estimation of the parameter should be obtained with a minimal number of samples.

4.5.1 Experiments

It has been seen that to build a good score function, the estimators used must verify the robustness and efficiency properties. To assess the robustness and the efficiency of the atypicity score, synthetic data are used for experimentation and analysis. The robustness of an estimator is its ability to be unbiased in the presence of anomalies. An estimator is said efficient when it is close to the parameter value with a limited number of data points. In this analysis, three categories of estimators are tested: one "efficient and not robust", a second "not efficient and robust" and a third "robust and efficient". These three estimators are analyzed considering the absence or presence of anomalies. The assessment is based on the parameter estimation error and on the anomaly detection performances using FDR and FNR.

4.5.1.1 Description

In this experiment, the focus is on the z-score. The atypicity of a data point x is calculated from the mean μ and standard deviation σ as follows $a_z(x, \mu, \sigma) = (x - \mu)/\sigma$. In an anomaly detection context, the mean and standard deviation are unknown and need to be estimated. There are many estimators of the mean and standard deviation. These estimators have different properties in terms of robustness and efficiency. In order to study the relationship between these properties and the performance of the anomaly detector, three estimators are chosen for each of these two values.
For the mean value the three estimators are defined as the following:

- Maximum Likelihood Estimator: $\mu_{mle} = \frac{1}{n}\sum_i x_i$. This estimator is efficient but not robust against anomaly contamination.

- Median: $\mu_r = \text{median}(x_1, ..., x_n)$. This estimator is robust but less efficient than the MLE estimator.

- Biweight location, introduced in [69]. This estimator is robust and efficient.

$$\mu_{bw} = \frac{\sum_{i=1}^{\ell}(1-u_i^2)x_i\mathbb{1}[|u_i|<1]}{\sum_{i=1}^{\ell}(1-u_i^2)}$$
$$u_i = \frac{x_i - \overline{x}}{9MAD}$$

  Where $\overline{x}$ is median of the x_i and MAD is the median absolute deviation.

For the standard deviation, the three estimators are defined as the following:

- Maximum Likelihood Estimator: $\sigma_{mle} = \sqrt{\frac{1}{n}\sum_i (x_i - \mu)^2}$. This estimator is efficient but not robust against anomaly contamination.
- Median: $\sigma_{mad} = \text{median}(||x_i - \mu|)$. This estimator is robust but less efficient than the MLE estimator.
- Biweight Midvariance estimator: introduced in [147]. This estimator is robust and efficient.

$$\sigma_{bw}^2 = \frac{\ell\sum_{i=1}^{\ell}(x_i-\overline{x})^2(1-u_i^2)^4\mathbb{1}[|u_i|<1]}{(\sum_{i=1}^{\ell}(1-u_i^2)(1-5u_i^2)\mathbb{1}[|u_i|<1])^2}$$
$$u_i = \frac{x_i - \overline{x}}{9MAD}$$

  Where $\overline{x}$ is the median of the x_i and MAD is the median absolute deviation.

All the six estimators are evaluated according to two measures:

1. First, the precision and the robustness of the estimator is evaluated using the Mean Squared Error (MSE), applying the following procedure: Let θ be either the mean or the standard deviation parameter, and $\hat{\theta}$ be an one estimator of the parameter θ. Let ℓ be the cardinality of the segment used to estimate θ. Let B be the number of repetitions for the experiments.
   (a) Generate the segment data: For b in $[1, B]$ and for i in $[1, \ell]$, $X_{b,i} \sim \mathcal{N}(0, 1)$, if the segment contains only normal data. For b in $[1, B]$ and for i in $[1, \ell_0]$, $X_{b,i} \sim \mathcal{N}(0, 1)$ and for i in $[\ell_1, \ell]$, $X_{b,i} = 4$, if the segment is contaminated by anomalies.
   (b) Estimate the parameter using the estimator: For b in $[1, B]$, $\hat{\theta}_b = \hat{\theta}(X_{b,1}, ..., X_{b,\ell})$.
   (c) Compute the MSE, $MSE = \frac{1}{B}\sum_{b=1}^{B}(\hat{\theta}_b - \theta)^2$

   Different values of the segment length ℓ are tested, from 10 to 1000. For each value of ℓ, two values of ℓ_1 are tested. One with $\ell_1 = 0$, for the case where there are no anomaly in the training set. The other with $\ell_1 = \lfloor 0.02\ell \rfloor$ for the case of contamination with anomalies. For each set of parameter values, the experiment is repeated $B = 1000$ times. The true mean is $\mu = 0$ and the true standard deviation $\sigma = 1$.
2. Then, the Anomaly Detection capacity is evaluated using the FDR and FNR criteria. This is done by simulating multiple detections inside a segment applying the following procedure: using n the calibration set cardinality, ℓ the length of the segment, ℓ_1 the number of anomalies in the training set, m the test set cardinality and m_1 the number of anomalies in the test set:

(a) Generate training segment data with ℓ_1 anomalies.

$$\forall i \in [\![1, \ell_1]\!], \quad X_i \sim \mathcal{N}(4, 0.1), \text{ and } \forall i \in [\![\ell_1, \ell]\!], \quad X_i \sim \mathcal{N}(0, 1)$$

And estimate the segment mean and standard deviation

$$\hat{\mu} = \hat{\mu}(X_1^\ell), \quad \hat{\sigma} = \hat{\sigma}(X_1^\ell)$$

(b) Generate the calibration set

$$\forall j \in [\![1, n]\!], \quad Y_j \sim \mathcal{N}(0, 1)$$

(c) Generate the test segment data

$$\forall i \in [\![1, m_1]\!], \quad Z_i \sim \mathcal{N}(4, 0.1), \text{ and } \forall i \in [\![m_1, m]\!], \quad Z_i \sim \mathcal{N}(0, 1)$$

(d) Compute the p-values of the test set, using calibration set and affected by the parameter estimations

$$\forall i \in [\![1, m]\!], \quad \hat{p}_i = \frac{1}{n} \sum_{j=1}^{n} \mathbb{1}[Y_j > (Z_i - \hat{\mu})/\hat{\sigma}]$$

(e) Anomalies are detected using the Benjamini-Hochberg procedure on the p-values. The threshold of the BH procedure is noted $\hat{\varepsilon}_{BH_\alpha}$ as stated in Definition 3.2:

$$\hat{\varepsilon} = \hat{\varepsilon}_{BH_\alpha}(\hat{p}_1, \ldots, \hat{p}_m)$$

(f) Compute FDP and FNP. Remembering that anomalies are generated in the first m_1 values of the test set:

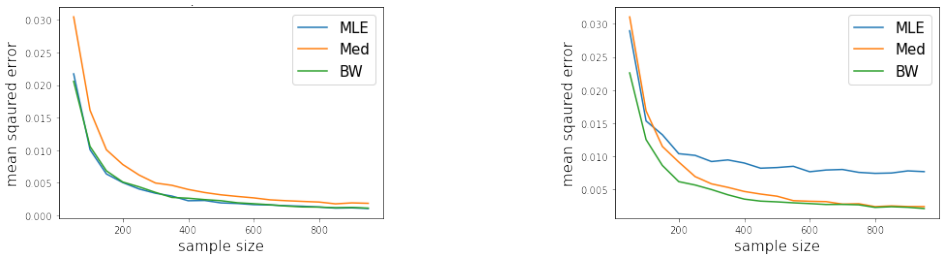
$$FDP = \frac{\sum_{j=m_1}^{m} \mathbb{1}[\hat{p}_j \leq \hat{\varepsilon}]}{\sum_{j=1}^{m} \mathbb{1}[\hat{p}_j \leq \hat{\varepsilon}]}$$
$$FNP = \frac{\sum_{j=1}^{m_1} \mathbb{1}[\hat{p}_j > \hat{\varepsilon}]}{m_1}$$

To simplify matters and make the effect of anomalies deterministic, in these experiments the number of anomalies is fixed rather than randomly drawn (unlike the mixture model). Different values of segment length ℓ are tested, from 10 to 500. For each value of ℓ, two values of ℓ_1 are tested. One with $\ell_1 = 0$, for the case where there are no anomaly in the training set. The other with $\ell_1 = \lfloor 0.02\ell \rfloor$ for the case of contamination with anomalies. The test set contain $m = 100$ data points with $m_1 = 1$ anomaly and the calibration set contains $n = 999$ data points. For each set of parameter values, the experiment is repeated $B = 10^4$ times.

4.5.1.2 Results

Figures 4.4 and 4.5 illustrate the estimators performances of the mean estimators. Figure 4.4 compares different mean estimators according to the segment length. The MSE decreases rapidly

with the sample size for all estimators in Figure 4.4a. However the MLE and BW estimators have very close and slightly better performances compared to the median estimator. This illustrates the efficiency of the MLE and BW estimators. But in the presence of anomalies, the performance of the MLE estimator is severely degraded as shown in Figure 4.4b compared to the median and BW estimators showing more robustness in the presence of outliers.



(a) Without anomaly contamination.

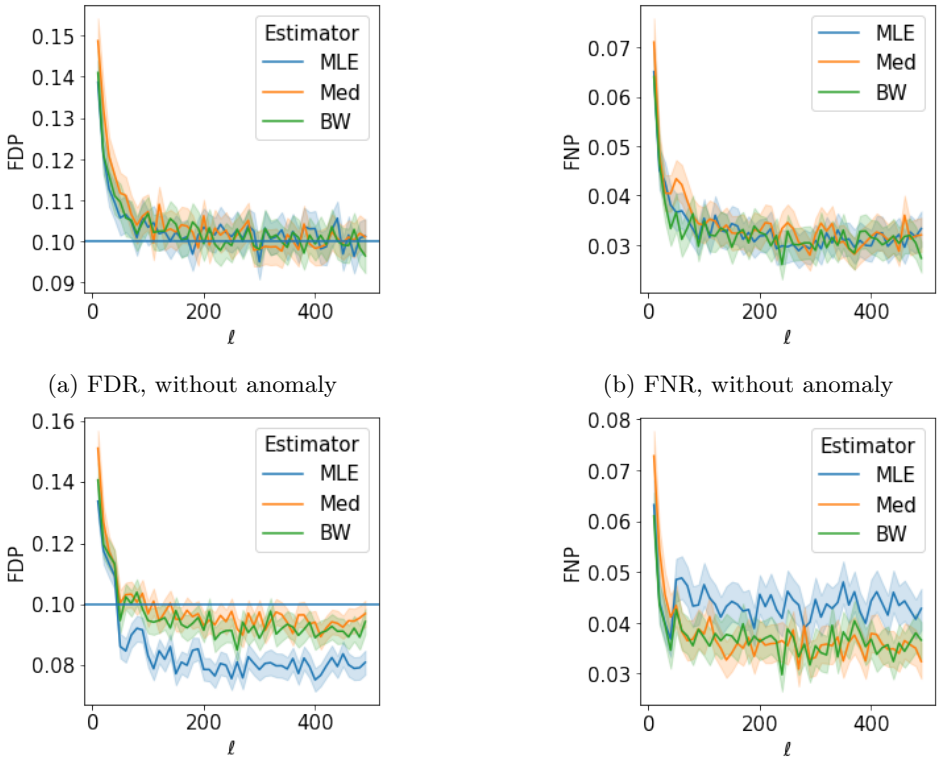
(b) With anomaly contamination.

Figure 4.4: Estimation error of the mean as a function of segment length and mean estimator used.

Figure 4.5 illustrates the FDR and FNR of the anomaly detector according to the mean estimator used. As shown in Figure 4.5a and 4.5b, in the case of non contamination by anomalies, the FDR and FNR results are very close to the target for all the estimators. However, in presence of anomalies, the MLE performance is degraded. In Figure 4.5c, the FDR is below the targeted level and in Figure 4.5d, the FNR is higher than other estimators. Either Med or BW can be used to do anomaly detection.

Figures 4.6 and 4.7 illustrate the performances of the standard deviation estimators. Figure 4.6 compares the precision using the MSE of the different standard deviation estimators according to the segment length. The MSE decreases rapidly with the sample size for all estimators in Figure 4.6a. However the MLE and BW estimators have very close and better performances when compared with the MAD estimator. This illustrates the efficiency of the MLE and BW estimators. But in the presence of anomalies, the performance of the MLE estimator is severely degraded as shown in Figure 4.6b. On the contrary, the MAD and BW estimators are less degraded and show more robustness in presence of outliers.

Figure 4.7 shows the performances measured by FDR and FNR once the anomaly detection is applied. As illustrated in Figures 4.7a and 4.7b, FDR and FNR for MAD are higher compared to MLE and BW. But in presence of anomalies, the MLE performance is degraded. The FDR is below the targeted level, as shown Figure 4.7c, and the FNR is higher than other estimators, as shown in Figure 4.7d. The strange behavior of the MLE curve in Figure 4.7d with spikes in the FNR is due to the number of anomalies increasing with every 50 data points because $\ell_1 = \lceil 0.02\ell \rceil$. The best estimator for standard deviation in case of anomaly detection is BW.



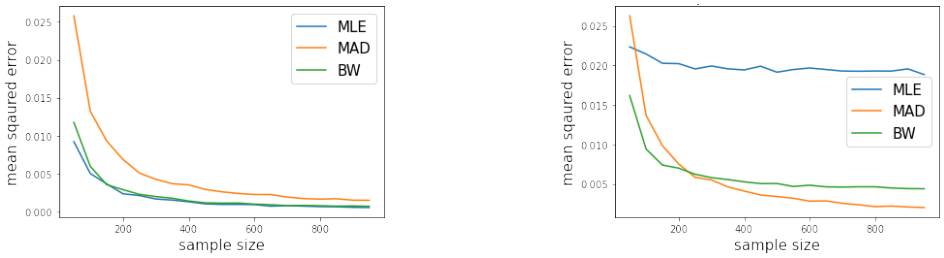
(a) FDR, without anomaly

(b) FNR, without anomaly

(c) FDR, with anomalies

(d) FNR, with anomalies

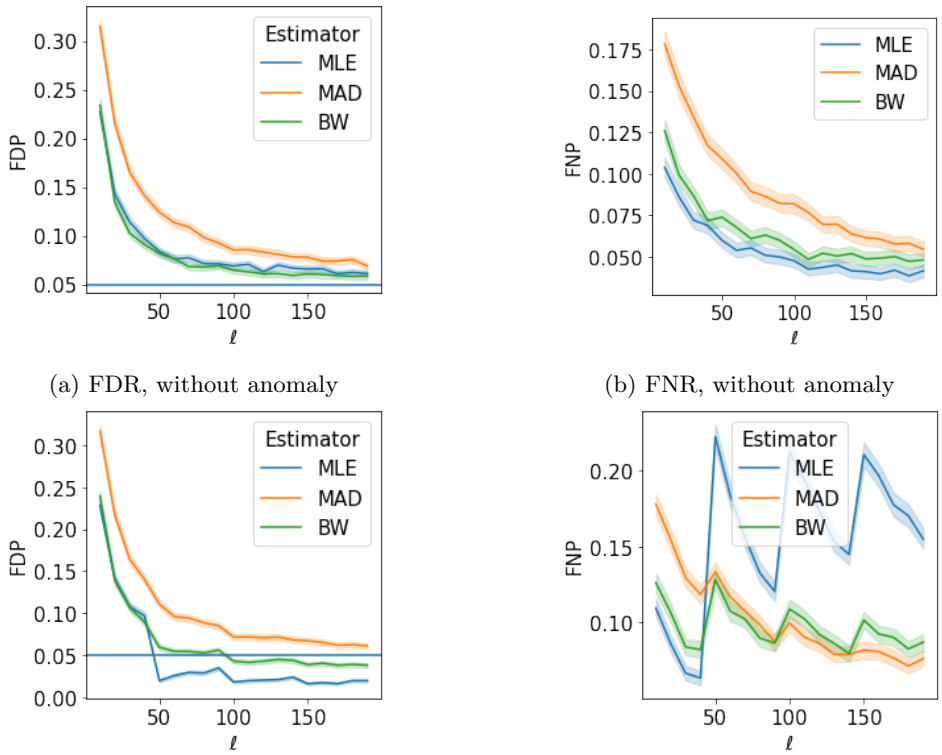
Figure 4.5: Anomaly detector performances as a function of the segment length and the mean estimator used. ($\alpha = 0.1$)



(a) Without anomaly contamination.

(b) With anomaly contamination.

Figure 4.6: Estimation error of standard deviation as a function of the segment length and the standard deviation estimator used.



(a) FDR, without anomaly

(b) FNR, without anomaly

(c) FDR, with anomalies

(d) FNR, with anomalies

Figure 4.7: Anomaly detector performances as a function of segment length and standard deviation estimator used. ($\alpha = 0.05$)

4.5.1.3 Conclusion

The experiments show the importance of the robustness and efficiency to build a good atypicity score. High MSE implies lower performance in terms of FDR and FNR control. The classical standard deviation estimators, MLE and MAD, are underperforming. For the following sections of this chapter, the BW (Biweight midvariance) estimator is used to implement the scoring function.

4.6 Confidence score estimation

Section 4.3.3 treats the uncertainty problem introduced in Section 4.2.3 from a theoretical point of view. An active set is constructed to collect points with too low a confidence score. The uncertainty in assigning a point to a segment with delay λ is modeled by $f_\tau(\lambda)$. The uncertainty in estimating the atypicity score in segments of length ℓ is described by the probability $f_d(\ell)$. Therefore, to construct the active set, the functions f_τ and f_d must be known or estimated. This question is addressed in this section, the estimation of f_τ in Section 4.6.1 and the estimation of f_d in Section 4.6.2.

4.6.1 Estimate the probability of segment assignment change

As introduced in Section 4.3.3.2, $f_\tau(\lambda)$ is the probability that the segment assignment changes when a data point is at distance λ from the end of the time series. This probability $f_\tau(\lambda)$ is needed to build the active set containing data points with uncertain status, as described in Algorithm 3. In the following, a procedure is proposed to estimate $f_\tau(\lambda)$.

4.6.1.1 Description of the method

As a reminder, the existence of $f_\tau(\lambda)$ is ensured by the stationarity assumption described in Proposition 4.1. However, stationarity is not sufficient to calculate these probabilities directly from historical data in the same time series and thus to estimate $\hat{f}_\tau$. It must also be assumed that the series $\mathbb{1}\left[W_{t-\lambda,t}\right]$ is ergodic.

Proposition 4.2 (Ergodicity)**.** *Assume* $\mathbb{1}\left[W_{t-\lambda,t}\right]$ *is stationary and ergodic. Then*

$$\mathbb{P}[W_{t-\lambda,t}] = \lim_{T\to\infty} \frac{1}{T} \sum_{\tilde{t}=1}^{T} \mathbb{1}[W_{\tilde{t}-\lambda,\tilde{t}}] \tag{4.27}$$

The conclusion of the Proposition 4.2 follows directly from the definition of ergodic process [67]. A sufficient condition to verify the ergodicity is the weak dependence or mixing [23]. There are several ways to characterize this property. The general idea is that the dependence between two points $\mathbb{1}[W_{t_1-\lambda,t_1}]$ and $\mathbb{1}[W_{t_2-\lambda,t_2}]$ must go to 0 as $t_1 - t_2$ goes to infinity. This property is assumed to be verified when these criteria are satisfied:

- the locations of the breakpoints are taken from an iid probability distribution,
- the transition in distribution between two segments (for example a shift in the mean) is generated by an iid model,
- Breakpoint detection performance is uniform over the entire time series. This is assumed to be the case for KCP.

An example of model achieving these criteria is given in Section 4.9.1. The breakpoint position iid property assumes the existence of random variables that generate the positions of these breakpoints. These random variables are assumed to be iid and uniform over $[\![1, T]\!]$. The iid property for the transition of the reference distribution implies the existence of random variables that generate the reference law of segment i from that of segment $i-1$. These random variables must be iid to guarantee uniformity in the difficulty of detecting breakpoints. As an example the time series generated in Section 4.9.1 satisfy these criteria: the segment length follows an exponential law, the positions of breakpoints follow a stationary Poisson process. The transition law is described as a homogeneous Markov chain on the parameters of the reference distribution.

The ergodicity property can be used to derive a learning algorithm. The time series is split into two parts: historical and recent data. The historical data set is built using the first $\tilde{T}$ data points of the time series. The estimation of f_τ is based on the previous segment assignment changes that occurred while detecting breakpoints on historical data. To estimate this probability, the list of all previous segmentations $\mathcal{D} = (\hat{\tau}_1, \ldots, \hat{\tau}_{\tilde{T}})$ is used. Assuming stationarity and ergodicity of $\mathbb{1}\left[W_{t-\lambda,t}\right]$, where the event $W_{t-\lambda,t}$ is described in Eq. 4.13, these historical data are used to estimate $f_\tau(\lambda)$ using Eq. 4.27.

$$\mathbb{P}\left[W_{t-\lambda,t}\right] \approx \frac{1}{\tilde{T}} \sum_{t=1}^{\tilde{T}} \mathbb{1}\left[W_{\tilde{t},\tilde{t}-\lambda}^{\tilde{T}}\right] = \hat{f}_\tau(\lambda) \tag{4.28}$$

where $W_{\tilde{t},\tilde{t}-\lambda}^{\tilde{T}} = \left\{\exists t' \in [\![\tilde{t}, \tilde{T}]\!], \quad \hat{\tau}_{t'} \cap [\![\hat{b}_{\tilde{t}}, \tilde{t}-\lambda]\!] \neq \emptyset\right\}$.

However, to improve computation time, the following expression of $W_{\tilde{t},\tilde{t}-\lambda}^{\tilde{T}}$ is preferred:

$$W_{\tilde{t}-\lambda,\tilde{t}}^{\tilde{T}} = \left\{ \left(\bigcup_{\tilde{T} \geq t' > \tilde{t}} \hat{\tau}_{t'} \right) \cap [\![\hat{b}_{\tilde{t}}, \tilde{t}-\lambda]\!] \neq \emptyset \right\} \tag{4.29}$$

With this formulation, each breakpoint is checked only once to see if it belongs to $[\![\hat{b}_{\tilde{t}}, \tilde{t}-\lambda]\!]$. Indeed, many breakpoints remain at the same position from one step to the next step while applying the breakpoint detection procedure.

Algorithm 5 Exact computation of probability of segment assignment change.

Require: $(\tau(\tilde{t}))_1^{\tilde{T}}$ list of successive segmentations
  $I, S \leftarrow 0$
  $\tau_{global} \leftarrow \emptyset$
  for $\tilde{t} \in [\tilde{T}, 1]$ **do**
    $\tau_{global} \leftarrow \tau_{global} \cup \hat{\tau}(\tilde{t})$
    for $u \in [\hat{b}_{\tilde{t}}, \tilde{t}]$ **do**
      for $b' \in \tau_{global}$ **do** ▷ b' is a breakpoint
        if $\hat{b}_{\tilde{t}} < b' \leq u$ **then**
          $I_{\tilde{t},u} \leftarrow 1$
        end if
      end for
    end for
  end for
  for $\lambda \in [0, \tilde{T}]$ **do**
    for $\tilde{t} \in [\lambda, \tilde{T}]$ **do**
      $S_\lambda \leftarrow S_\lambda + I_{\tilde{t},\tilde{t}-\lambda}$
    end for
    $\hat{f}_{\tau,\lambda} \leftarrow S_\lambda/(\tilde{T} - \lambda)$
  end for
  Output: $\hat{f}_{\tau,\lambda}$ list of $\hat{f}_\tau(\lambda)$ values for different λ
  return $\hat{f}_{\tau,\lambda}$

Algorithm 5 implements Eq. 4.29 to give an estimation of f_τ. Where $I_{\tilde{t},u} = \mathbb{1}[W_{\tilde{t},u}^{\tilde{T}}]$ and $S_\lambda = \sum_{\tilde{t}=\lambda}^{\tilde{T}} I_{\tilde{t},t-\lambda}$ so $\hat{f}_\tau(\lambda) = \frac{S_\lambda}{\tilde{T}-\lambda}$.

The complexity of the exact computation of $\hat{f}_\tau$, described in Algorithm 5, is quadratic in time and space, which is a drawback regarding any practical use. Another version of the implementation of $\hat{f}_\tau$ is given in Algorithm 7 which is more convenient for an online context since it is linear in time and space.

Indeed, by observing the evolution of the breakpoints over time (not shown in this thesis), it appears that the position of the last breakpoint is the most likely to change, while that of the other breakpoints are generally stable. This leads us to modify the characterization of the "assigned segment change" event by considering only the change caused by the last breakpoint instead of the entire segmentation.

$$\forall t, \lambda \in [\![1, T]\!]^2 \qquad \mathbb{1}[W_{t,t-\lambda}^T] = \mathbb{1}\left[\exists t' \in [\![t, T]\!], \quad \hat{b}_t < \hat{b}_{t'} \leq t - \lambda\right] \tag{\textbf{Last}}$$

Under this assumption, the computation of $\hat{f}_\tau(\lambda)$ can be simplified using Proposition 4.3.

Proposition 4.3. *Let $(X_t)_{1\leq t\leq T}$ be a time series of length T. Let $(\hat{\tau}(t))_{1\leq \tilde{t}\leq \tilde{T}}$ be the sequence of successive segmentations of the time series. Let $(\mathbb{1}[W_{t,u}^T])_{1\leq t\leq T, 1\leq u\leq T}$ the family of "assigned segment change" events. Assume that the assumption (**Last**) is verified. Then the estimator $\hat{f}_\tau$*

described in Eq. 4.29 is computed as

$$\hat{f}_\tau(\lambda) = \frac{1}{\tilde{T}} \sum_{\tilde{t}=1}^{\tilde{T}} \mathbb{1}\left[r_{\tilde{t}} > \lambda\right] \tag{4.30}$$

where $r_{\tilde{t}}$ is the maximum distance from the the end of the time series with X_t having segment reassigned. It is computed as:

$$r_{\tilde{t}} = \max_{t' > \tilde{t}, \hat{b}_{t'} > \hat{b}_{\tilde{t}}, \hat{b}_{t'} < \tilde{t}} \tilde{t} - \hat{b}_{t'} \tag{4.31}$$

Notice that $r_{\tilde{t}}$ does not depend on λ. It is sufficient to calculate $r_{\tilde{t}}$ once for all λ. Therefore, it's easy to deduce the value of $f_\tau(\lambda)$ for all λ. The most demanding part is the computation of $r_{\tilde{t}}$. Two implementations of $r_{\tilde{t}}$ computation are proposed. Algorithm 6 gives the most naive version, each $r_{\tilde{t}}$ is calculated one after the other. The problem is that the calculation of $r_{\tilde{t}}$ is itself linear in the length of the series. Therefore, the time complexity is quadratic. Algorithm 7 improves the computation by swapping the two loops. This limits the total number of comparisons performed. In the second loop, $\tilde{t}$ takes on the values between b'_t and t'. The number of values taken by $\tilde{t}$ is the length of a segment, not a the length of the time series. Algorithmic complexity is therefore linear.

Proof of Proposition 4.3. Based on Eq. 4.29, the estimator $\hat{f}_\tau$ is given by:

$$\hat{f}_\tau(\lambda) = \frac{1}{\tilde{T}} \sum_{t=1}^{\tilde{T}} \mathbb{1}\left[W_{\tilde{t},\tilde{t}-\lambda}^{\tilde{T}}\right]$$

With assumption (**Last**) it gives:

$$\mathbb{1}\left[W_{\tilde{t},\tilde{t}-\lambda}^{\tilde{T}}\right] = \mathbb{1}\left[\exists t' \in [\![\tilde{t}, \tilde{T}]\!], \quad \hat{b}_{\tilde{t}} < \hat{b}_{t'} \leq \tilde{t} - \lambda\right]$$

The inequality $\hat{b}_{\tilde{t}} < \hat{b}_{t'} \leq \tilde{t} - \lambda$ is equivalent to $\hat{b}_{\tilde{t}} < \hat{b}_{t'}$ and $\tilde{t} - \hat{b}_{t'} > \lambda$ which gives

$$\mathbb{1}\left[W_{\tilde{t},\tilde{t}-\lambda}^{\tilde{T}}\right] = \mathbb{1}\left[\bigcup_{t' > \tilde{t}, \hat{b}_{t'} > \hat{b}_{\tilde{t}}, \hat{b}_{t'} < \tilde{t}} \tilde{t} - \hat{b}_{t'} > \lambda\right]$$

Since, a set contains a number greater than λ, if and only if its maximum is greater than λ, it gives:

$$\mathbb{1}\left[W_{\tilde{t},\tilde{t}-\lambda}^{\tilde{T}}\right] = \mathbb{1}\left[\left(\max_{t' > \tilde{t}, \hat{b}_{t'} > \hat{b}_{\tilde{t}}} \tilde{t} - \hat{b}_{t'}\right) > \lambda\right]$$

Since $\lambda > 0$, when $\hat{b}_{\tilde{t}} < \hat{b}_{t'}$ and $\tilde{t} - \hat{b}_{t'} > \lambda$ it also implies that $\tilde{t} \geq \hat{b}_{t'}$ so $\mathbb{1}\left[\left(\max_{t' > \tilde{t}, \hat{b}_{t'} > \hat{b}_{\tilde{t}}} \tilde{t} - \hat{b}_{t'}\right) > \lambda\right] = \mathbb{1}\left[\left(\max_{t' > \tilde{t}, \hat{b}_{t'} > \hat{b}_{\tilde{t}}, \hat{b}_{t'} < \tilde{t}} \tilde{t} - \hat{b}_{t'}\right) > \lambda\right]$. The number $r_{\tilde{t}}$ is introduced as equal to $\max_{t' > \tilde{t}, \hat{b}_{t'} > \hat{b}_{\tilde{t}}, \hat{b}_{t'} < \tilde{t}} \tilde{t} -$

$\hat{b}_{t'}$. The $\hat{f}_\tau$ estimator can be written as follows

$$\hat{f}_\tau(\lambda) = \frac{1}{\tilde{T}} \sum_{\tilde{t}=1}^{\tilde{T}} \mathbb{1}\left[r_{\tilde{t}} > \lambda\right]$$

□

Algorithm 6 Naive computation of $r_{\tilde{t}}$.

```
for t̃ in [[1, T̃]] do
    for t' in [[t̃, T̃]] do
        if b̂_{t'} < t̃ and b̂_{t̃} > b̂_{t'} then
            r_{t̃} = max(r_{t̃}, t̃ − b̂_{t'})
        end if
    end for
end for
```

Algorithm 7 Efficient computation of $r_{\tilde{t}}$.

```
for t' in [[1, T̃]] do
    for t̃ in [[b̂_{t'}, t']] do
        if b̂_{t̃} > b̂_{t'} then
            r_{t̃} = max(r_{t̃}, t̃ − b̂_{t'})
        end if
    end for
end for
```

4.6.1.2 Application on simulated data

In the previous Section 4.6.1.1, two algorithms for estimating the probability of a segment assignment change were described: the exact estimation using Algorithm 5 and an efficient estimation using Algorithm 7. In this section, these different methods are assessed by experiments on simulated data.

Description of the experiment The following notations are used: Let T be the length of the time series, θ the average segment length, Δ the size jumps to generate a breakpoint and σ the standard deviation of the data point within a segment.

Time series are generated according to the following rules:

- The number of breakpoints follows the exponential distribution $D - 1 \sim Exp(T/\theta)$.
- Each breakpoint position is generated according to uniform distribution $\forall i \in [1, D-1], \tau_i \sim U(1, T)$
- The mean of the time series μ_i is piecewise constant with respect to the segmentation τ_i, with $\mu_{\tau_i} - \mu_{\tau_i+1} = \xi\Delta\sigma$
- The time series is generated according to the following rule $X_t \sim \mathcal{N}(\mu_t, \sigma)$

Then $\hat{f}_\tau(\lambda)$ is estimated using the two different methods: Algorithm 5 and Algorithm 7.

Results and analysis: Figure 4.8 gives the estimated probability of segment assignment change according to the two estimation Algorithms 5 and 7. The two algorithms give results that are almost the same, as shown in Figure 4.8. The selected $\lambda_\eta^\star$ is equal to 143, in the two cases. This supports assumption that (**Last**) is verified. In practice, we recommend to use the Algorithm 7 method since it is more computationally efficient. To compute the probability $\hat{f}_\tau(\lambda)$ on a PC (4 CPU, 16GB), the Algorithm 7 gives results within 30 seconds compared to the exact computation which gives the results within 15mn, for a time series of length 10^4.

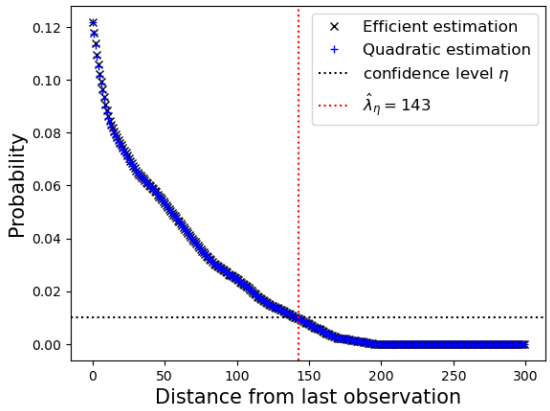
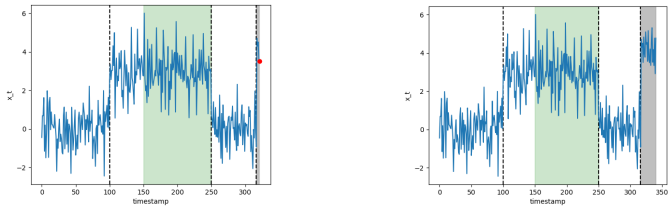


Figure 4.8: Probability of assignment change as a function of distance to time series end.

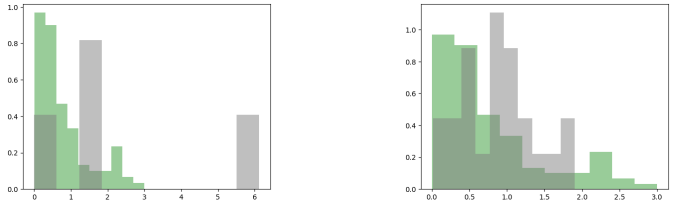
4.6.2 Estimate the probability that the point will have a status different from that of the oracle if the point is assigned to the same segment as the oracle.

As introduced in Section 4.3.3.2, $f_d(\ell)$ is the probability that the status of a point changes under the conditions the last breakpoint remains unchanged and the segment cardinality is equal to ℓ. This probability $f_d(\ell)$ is necessary to build the active set containing data points with uncertain status, as described in Algorithm 3. In this section, a procedure to estimate $f_d(\ell)$ is proposed.



(a) Starting new segment with small length.

(b) Data points added to the current segment becoming larger.



(c) Z-score for small segment

(d) Z-score for larger segment

Figure 4.9: Atypicity score estimation according to the length of the current segment.

Figure 4.9 illustrates how the the length of the current segment has an influence on the accuracy of the atypicity score estimation and consequently on the uncertainty of a data point status. Indeed, Figure 4.9a shows a time series with a newly detected current segment highlighted in gray color, and a calibration set in green color inside the previous segment. The atypicity

scores, z-scores based on the mean and the standard deviation, are shown in Figure 4.9c computed for the current segment in gray and the calibration set in green. Since the current segment has few points, its z-score estimation shows a high discrepancy compared to the score distributions of the calibration set, despite the fact that there are no anomalies. As shown in Figure 4.9d, when new data points are added, the estimation of the abnormality score of the current segment is more accurate.

This example highlights that the status of a data point can change even if the breakpoints remain unchanged, whereas Section 4.6.1 deals only with the case where the change in status is due to a change in the detected breakpoints. Uncertainty also comes from having too few points in the current segment, leading to score estimation errors. Estimating the probability $\hat{f}_d(\ell)$ is useful to build the active set that takes this into account. The following section suggests a procedure to estimate the probability $f_d(\ell)$.

4.6.2.1 Description of the method

The method is based on the learning phase using a set $\mathcal{D}$ of historical detected segments having a low probability to change (using final step T). This training set D is defined by,

$$\mathcal{D} = \{(X_1, \dots, X_{\hat{\tau}_1(T)}), (X_{\hat{\tau}_1(T)+1}, \dots, X_{\hat{\tau}_2(T)}), \dots, (X_{\hat{\tau}_D(T)+1}, \dots, X_{\hat{\tau}_{D+1}(T)})\} \tag{4.32}$$

In the following, the training procedure is based on six different steps needed to estimate the $\hat{f}_d(\ell)$ probability. Let $\overline{a}$ be the NCM (Non Conformity measure), used to define the atypicity score, as described in Section 4.5. As a reminder, $\overline{a}(S, x)$, measures the "nonconformity" between the set S and the point x.

Training procedure: The principle of the training phase is to simulate, using resampling, numerous examples where the current segment changes from a length ℓ to the final length. At each case, anomaly detection is applied to the test set of cardinality m and the proportion of statuses that have changed by modifying the length of the current segment is measured. The status obtained from the maximum size segment is the oracle status. By comparing it with the status obtained with the ℓ size segment, the confidence score can be approximated. Since the breakpoints are assumed to be stable, the simulation is inspired by the description of the detector given in Section 2, without the parts concerning breakpoint detection. These steps are repeated B times. Let $b \in [\![1, B]\!]$:

Step 1: Figure 4.10a illustrates that two segments are resampled drawing them uniformly from the historical data set $\mathcal{D}$. $\mathcal{S}_{1,b}$ is considered as the calibration set and $\tilde{\mathcal{S}}_{2,b}$ as a current segment if the whole time series where observed (see Definition 4.5) in the simulation:

$$\mathcal{S}_{1,b}, \tilde{\mathcal{S}}_{2,b} \sim U(\mathcal{D})$$

Step 2: The current segment $\tilde{\mathcal{S}}_{2,b}$ is sub-sampled into a smaller segment of length ℓ and noted $S_{2,\ell,b}$, as shown in Figure 4.10b. $S_{2,\ell,b}$ is considered as the same segment than $\tilde{\mathcal{S}}_{2,b}$ without knowledge of the whole time series, having only ℓ points, .

$$\mathcal{S}_{2,\ell,b} \sim U(\tilde{\mathcal{S}}_{2,b})$$

.

Step 3: The current segment $\tilde{\mathcal{S}}_{2,b}$ is sub-sampled into an other segments of length m and noted

$\overline{S}_{2,m,b}$. $\overline{S}_{2,m,b}$ is considered as the test set.

$$\overline{\mathcal{S}}_{2,m,b} \sim U(\tilde{\mathcal{S}}_{2,b})$$

Step 4: As illustrates in Figure 4.10c and 4.10d, the scores of the three segments are computed:

- The score of $\mathcal{S}_{1,b}$:

$$\forall i \in [\![1,n]\!], X_i \in \mathcal{S}_{1,b}, \quad c_{i,b} = \overline{a}(Y_i, \mathcal{S}_{1,b} \backslash \{X_i\})$$

- The score of the test set using $\tilde{\mathcal{S}}_{2,b}$ as training set:

$$\forall i \in [\![1,m]\!], Y_i \in \overline{S}_{2,m,b}, \quad \tilde{s}_{i,b} = \overline{a}(Y_i, \tilde{\mathcal{S}}_{2,b} \backslash \{Y_i\})$$

- The score of the test set using $S_{2,\ell,b}$ as a training set:

$$\forall i \in [\![1m]\!], Y_i \in \overline{S}_{2,m,b}, \quad s_{i,\ell,b} = \overline{a}(Y_i, S_{2,\ell,b} \backslash \{Y_i\})$$

Step 5: Figure 4.10e illustrates that the empirical p-values are computed for the two scores obtained from the test set using the same calibration set:

- p-values of the test set when using the complete current segment as training set: $\forall i \in [\![1,m]\!], \tilde{p}_{i,b} = \frac{1}{n} \sum_{j=1}^{n} \mathbb{1}[\tilde{s}_{i,b} < c_{j,b}]$
- p-values of the test set when using the length ℓ sub-sample of the current segment as training set: $\forall i \in [\![1,m]\!], p_{i,\ell,b} = \frac{1}{n} \sum_{j=1}^{n} \mathbb{1}[s_{i,\ell,b} < c_{j,b}]$

Step 6: Detect the anomalies in the two cases, by applying the Benjamini-Hochberg procedure $\hat{\varepsilon}_{BH_\alpha}$ on the estimated p-values, as shown in Figure 4.10f:

- In case the training set is the entire current segment:

$$\forall i \in [\![1,m]\!], \quad \tilde{d}_{i,b} = \mathbb{1}[\tilde{p}_{i,b} < \hat{\varepsilon}_{BH_\alpha}(\tilde{p}_{1,b}, ..., \tilde{p}_{m,b})]$$

- In case the training set is the sub-sample of cardinality ℓ:

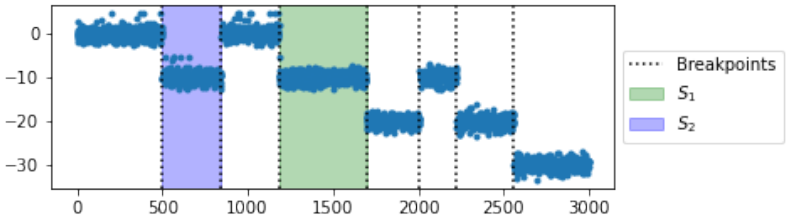
$$\forall i \in [\![1,m]\!], \quad d_{i,\ell,b} = \mathbb{1}[p_{i,\ell,b} < \hat{\varepsilon}_{BH_\alpha}(p_{1,\ell,b}, ..., p_{m,\ell,b})]$$

Step 7: The number of decisions that differ between the two cases, $\tilde{\mathcal{S}}_{2,b}$ or $\mathcal{S}_{2,\ell,b}$ used as the training set, is computed:

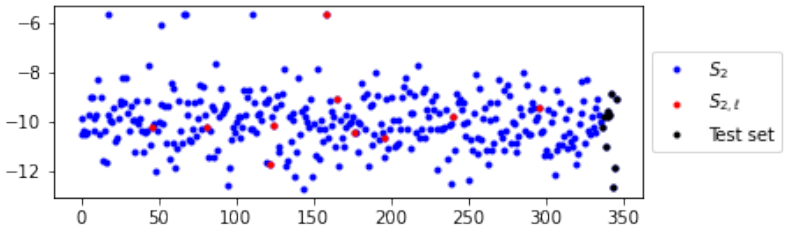
$$n_d = \sum_{i=1}^{m} \mathbb{1}[\tilde{d}_{i,b} \neq d_{i,\ell,b}]$$

The training procedure simulates the behavior of the online anomaly detector: $\mathcal{S}_1$ plays the role of the calibration set. $\tilde{\mathcal{S}}_2$ plays the role of current segment with knowledge of the whole time series. $\mathcal{S}_{2,\ell}$ plays the role of the current segment at the beginning of a new segment, that contains only ℓ points. The first m elements $Y_1, \ldots, Y_m$ from $\tilde{\mathcal{S}}_2$ constitute the test set.

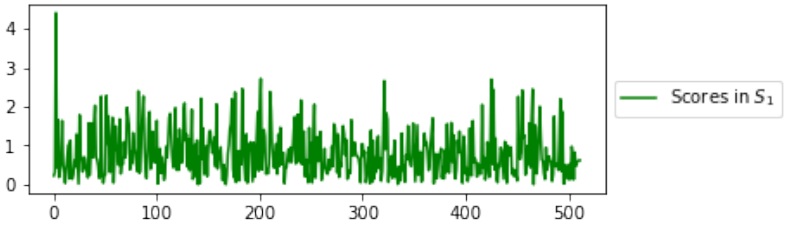
Assuming stationarity and piecewise dependence, as stated in Definition 4.3, by repeating this resampling process many times, as the length of the time series converges to infinity, the



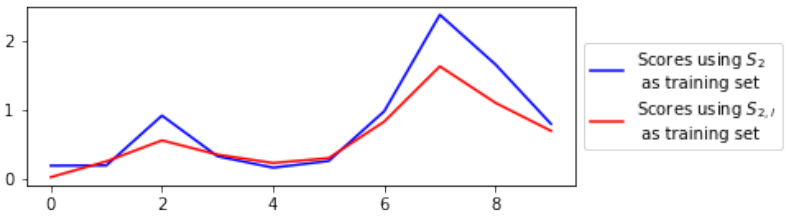
(a) Step 1: Segments resampling



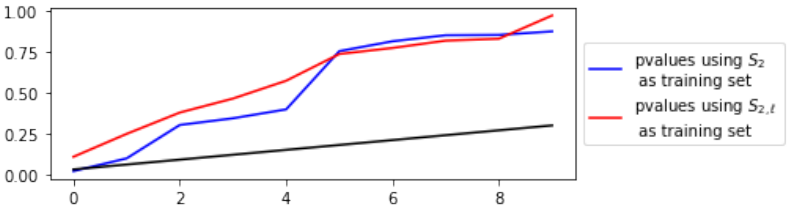
(b) Step 2 and 3: Sub-sampling



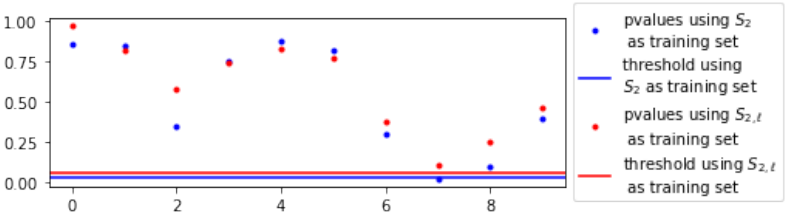
(c) Step 4: Calibration set scoring



(d) Step 5: Test set scoring



(e) Step 6: p-value estimation



(f) Step 7: Anomaly detection

Figure 4.10: Illustration of the different steps of the training procedure to estimate the status change probability under stable breakpoints.

proportion of status changes converges to the expectation, according to the law of large numbers [44]:

$$\lim_{T,B\to\infty} \frac{1}{mB} \sum_{b=1}^{B} \sum_{j=1}^{m} \mathbb{1}[\tilde{d}_{j,b} \neq d_{j,\ell,b}] = \mathbb{E}_{\mathcal{S}_1,\mathcal{S}_2\sim U(\mathcal{D})} \mathbb{E}_{\mathcal{S}_{2,\ell}\sim U(\mathcal{S}_2)} \sum_{i=1}^{m} \mathbb{1}[d_i \neq \tilde{d}_i]$$

Under the assumptions of score stationarity stated in Definition 4.3, the limit is equal to $f_d(\ell)$. Indeed, under score stationarity, the calibration set can be built from any segment of the time series. This implies that it is possible to use described training procedure as an estimator of $\hat{f}_d(\ell)$.

$$\hat{f}_d(\ell) = \frac{1}{mB} \sum_{b=1}^{B} \sum_{j=1}^{m} \mathbb{1}[\tilde{d}_{j,b} \neq d_{j,\ell,b}] \approx f_d(\ell)$$

4.6.2.2 Application on simulated data

The training procedure in Section 4.6.2.1 is applied for different scoring functions adapted to different types of time series considered in Section 4.5. : The goal is to check if the estimation approach of $\hat{f}_d(\ell)$ can be applied to different scoring functions.

Description of the experiment Different series that require different scoring functions are considered: Gaussian and Mixture of Gaussian.

- Figure 4.11a shows a Gaussian white noise with anomalies in distribution tail.

$$\begin{aligned} \forall t \in [\![1, T]\!], \quad & A_t \sim Ber(\pi), \\ & \text{if } A_t = 0, X_t \sim \mathcal{N}(0, 1) \\ & \text{else } X_t = \Delta \end{aligned}$$

  The z-score applied on X_t to detect anomalies that are in the tail of the distribution, is computed by,

$$\overline{a}(X_t, S) = |X_t - \hat{\mu}_S|/\hat{\sigma}_S \tag{4.33}$$

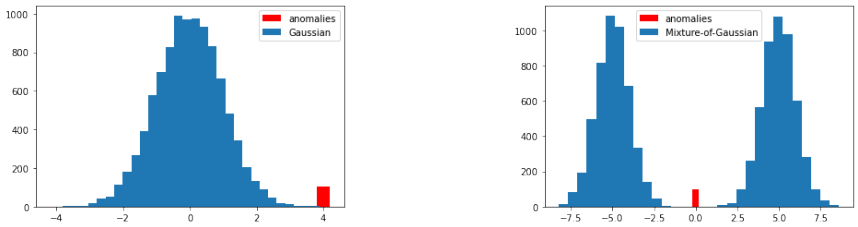
  where S is a segment of data, $\hat{\mu}_S$ the mean estimator on S and , $\hat{\mu}_S$ the standard deviation on S

- Figure 4.11b shows a Mixture of Gaussians with anomalies between the distribution modes.

$$\begin{aligned} \forall t \in [\![1, T]\!], \quad & A_t \sim Ber(\pi), \\ & \text{if } A_t = 0, X_t \sim 0.5\mathcal{N}(\Delta, 1) + 0.5\mathcal{N}(-\Delta, 1) \\ & \text{else } X_t = 0 \end{aligned}$$

  The kernel based score, inspired from other works on kernel based anomaly detection [68, 141], applied to detect anomalies having large distance from the normal data, is computed by,

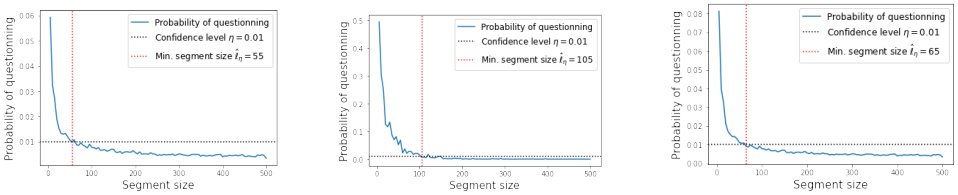
$$\overline{a}(X_t, S) = \frac{1}{|S|^2} \sum_{s,s'\in S^2} K(s, s') - \frac{2}{|S|} \sum_{s\in S} K(X_t, s) + K(X_t, X_t) \tag{4.34}$$



(a) Gaussian with anomalies in tail (b) Gaussian mixture with anomalies in center

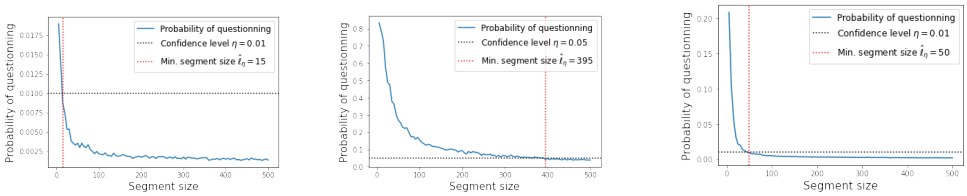
Figure 4.11: Different time series distributions and anomalies.

Results and analysis As stated previously, two types of time series are considered in the experiments: results of Gaussian data shown in Figure 4.12 and results of Gaussian mixture data shown in Figure 4.13. For both, three line charts representing the probability of status change as a function of the current segment length in relation to the initial status: (a) the status is normal, (b) the status is abnormal and (c) unknown status.



(a) normal status (b) abnormal status (c) unknown status

Figure 4.12: Probability that status changes under stable breakpoints as a function of segment length, for Gaussian data.



(a) normal status (b) abnormal status (c) unknown status

Figure 4.13: Probability that status change under stable breakpoint as a function of segment length, for Gaussian mixture data.

For Gaussian data and in the unknown status, Figure 4.12c shows clearly that the probability of status change decreases with the length of the current segment. This probability is higher when the status is abnormal, as shown in Figure 4.12b. Nevertheless, with a segment length of 100, the probability is less than 1%. For Gaussian mixture data and in the abnormal status scenario shown in Figure 4.13b, the length of the current segment needs to be at least equal to 500 to get a probability of changing status around 5%. For the normal status scenario in Figure 4.13a, the probability of changing quickly decreases to 0. The results are also promising in the unknown status scenario in Figure 4.13c, where the change probability is low.

Conclusion A solution to compute the probability of status change under "stable" breakpoints has been built. Empirical results show that the choice of an optimal $\hat{\ell}_\eta$ which reduces the uncertainty of a data point status depends on the type of data and the scoring function that is used. The method can help to select an atypicity score. A good atypicity score, satisfying requirements discussed in Section 4.5 (been robust and efficient) should have low $\hat{\ell}_\eta$ value.

4.7 Calibration set

Section 4.3.1 introduces the notion of calibration set by giving a high level description of the Breakpoint Based Anomaly Detector. It is a collection of data points representing the reference behavior, inspired by Conformal Anomaly Detection[100, 86]. It is built using data from the current segment, or from another segment in the history with a similar distribution probability compared to the current segment. The cardinality of the calibration set follows two constraints:

- it should be large enough to ensure that the p-values are estimated with sufficient precision to generate a low false positive and false negative rate.
- it should not be too large to maximize the homogeneity of the data and to limit computation time.

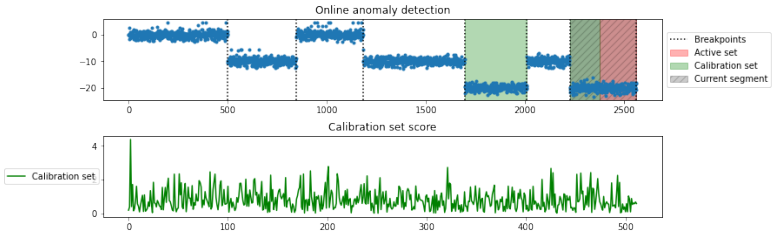


Figure 4.14: Illustration of the current segment, the active set and the calibration set.

Previously, in **Score** it was assumed that the scores of different segments followed the same distribution, but this is not the case in practice, so in order to reduce the bias induced, segments with a similar distribution are searched for. As shown in Figure 4.14, while data are collected online, the length of the current segment after the new breakpoint is too small to build the whole calibration set. By identifying similar segments and merging them to build the calibration set, the current segment can be completed with enough data points to estimate the p-values accurately. Similar segments are found using a similarity function, like the Bhattacharyya distance proposed in [15]. This similarity function is defined between two segments S_1, S_2 with means μ_1 and μ_2 and standard deviations σ_1 and σ_2 by:

$$sim(\mathcal{S}_1, \mathcal{S}_2) = -\frac{1}{8\sigma^2}(\mu_1 - \mu_2)^2 - \frac{1}{2}\ln\frac{\sigma}{\sqrt{\sigma_1\sigma_2}} \tag{4.35}$$

The similarity function allows to sort all historical segments according to their similarity to the current segment. First, the similarity of each segment to the current segment is calculated. This allows to assign to each data point X_u the variable that characterizes the similarity s_u. By definition, the sequence (s_u) is constant on each segment $X_{\tau_i(t)}^{\tau_{i+1}(t)-1}$ and maximal on the current

segment $X^t_{\hat{b}_t}$.

$$\forall i \in \hat{[\![}1, D_t[\![}, \forall u \in [\![\hat{\tau}_i(t), \hat{\tau}_{i+1}(t) - 1[\![}, \quad sim_u = sim(X^{\hat{\tau}_{i+1}(t)-1}_{\hat{\tau}_i(t)}, X^t_{\hat{b}_t}) \tag{4.36}$$

To build a calibration set of cardinality n, it is initialized using the scores of data points of the current segment that are not assigned to the active set. The data points scores with a "normal" status from the previous segments are added to the calibration set in descending order of similarity until n scores are reached. After having described how a calibration set of a given cardinality n is built, Section 4.8 describes how the optimal cardinality n is chosen.

4.8 p-value estimation and threshold selection

After having defined the active set and the calibration set, the empirical p-values of each data point of the active set are computed using the calibration set. The threshold is chosen using the p-values of the active set to ensure the control of the FDR at a given level α. Finally, the status of each data point of the active set is reevaluated comparing its p-value to the threshold.

In Chapter 3 we detail a new strategy for controlling the FDR of an anomaly detector in the online framework. This goal is achieved by efficiently controlling the modified FDR criterion (mFDR) of subseries so that the FDR value of the full time series is controlled at the prescribed level α. A modified version of the Benjamini-Hochberg procedure was designed. Instead of applying BH to the active set with a slope α, it is applied with a slope $\alpha' = \frac{\alpha}{1+\frac{1-\alpha}{m\pi}}$, where m denotes the length of the active set, α is the desired global FDR level, and π refers to the proportion of anomalies. Since α' depends on π, an estimation of π (or expert knowledge) is required to detect anomalies. Some guidelines are provided in [159]. Notice that when π is given, the fix threshold $\frac{\pi\alpha}{1+\pi-\alpha}$ control the FDR at level α, this is equivalent to using BH with a subseries of length $m = 1$.

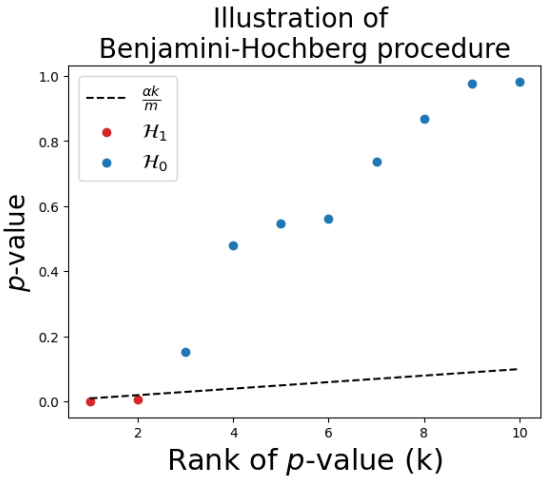


Figure 4.15: Example of Benjamini-Hochberg procedure.

The calibration set is used to compute the p-values. The FDR and the FNR of the modified BH procedure is very sensitive to the cardinality of the calibration set used to estimate the p-value. In Section 3.3.2, we study under which conditions the cardinality of the calibration set

ensures a control of the FDR. Given m the cardinality of the active set and α' the modified slope for BH, the calibration set cardinality has to be chosen among:

$$n \in \{\nu \frac{m}{\alpha'} - 1, \quad \nu \in \mathbb{N}^*\} \tag{4.37}$$

As explained more deeply in Section 3.3.2, the number of false negatives decreases with higher ν. But a larger ν also increases the computation time, which can make any real-time decision difficult. We recommend to try different values of ν, to monitor the decision time and to choose the largest ν which allows real time decisions.

4.9 Empirical study

An anomaly detector based on breakpoint detection has been proposed in Section 4.3. The core components have been separately elaborated and evaluated in Sections 4.4, 4.5, 4.6, 4.7 and 4.8 . In this section, the performance of the whole anomaly detector is assessed. The experiments are conducted in several steps. First, the anomaly detector is applied to several synthetic time series. The flexibility of the detector is evaluated and the roles played by the kernel and the atypicity score are highlighted. Second, the anomaly detector is applied by choosing different hyperparameters involved in the core components, not necessarily the same as those proposed in the previous analyses. The relevance of the different components and their associated analyses are evaluated. Third, the anomaly detector is applied by replacing some estimators with true knowledge in order to explore more deeply the reasons for the errors made by the anomaly detector. Finally, the anomaly detector is evaluated against alternative anomaly detectors.

An experimental framework is designed to conduct the experiments and to evaluate different aspects of the anomaly detector. The framework described in Section 4.9.1 is adapted for different time series and anomaly detector parameters.

4.9.1 Experimental framework

Let's consider a time series generation process and an anomaly detector. The following steps are repeated on different samples of the time series:

1. Generate the time series, according to the the first reference distribution $\mathcal{P}_{0,1}$, the proportion of anomalies π, the alternative distribution $\mathcal{P}_{1,1}$ and the transition rule describing how the parameters of the reference distribution will change between two segments.
   (a) The number $D-1$ of breakpoints is generated by $Exp(T/\theta)$, where θ is the average distance between two breakpoints.
   (b) The position of the $D-1$ breakpoints follows $U([1,T])$. In addition to the previous step, this implies that the process of breakpoint positions is a Poisson process.
   (c) The rule is applied iteratively to get the reference and alternative distributions for each segment. Two types of rules are considered:
      - Breakpoint in the mean with a jump size of Δ. For each i in $[\![1, D-1]\!]$, let μ_i be the mean of the reference distribution in the ith segment. The mean of a segment is equal to the mean of the previous one shifted with Δ.

$$\forall i \in [\![1, D-1]\!], \quad \mu_{i+1} = \mu_i + \zeta_i \Delta \tag{4.38}$$

With ζ_i, a random variable following the Rademacher distribution and defining the sign of the jump.

- Breakpoint in the variance with a jump scale size of Δ. For each i in $[\![1, D-1]\!]$, let σ_i be the standard deviation of the reference distribution in the ith segment. The standard deviation of a segment is equal to the standard deviation of the previous segment multiplied or divided by Δ.

$$\forall i \in [\![1, D-1]\!], \quad \sigma_{i+1} = \exp(\zeta_i \ln \Delta/2) * \sigma_i \tag{4.39}$$

With ζ_i, a random variable following the Rademacher distribution and defining if the standard deviation is multiplied or divided by Δ.

In this modeling, the locations of the breakpoints are taken from an iid probability distribution, and the transition in the distribution between two segments is generated by an iid model, which allows uncertainty to be controlled as discussed in Section 4.6.

(d) The position of anomalies are generated by a Bernoulli distribution: $A_t \sim Ber(\pi)$

(e) All the values of the time series are computed as follows:

$$\forall i \in [\![1, D]\!], \quad \forall t \in [\![\tau_i, \tau_{i+1}[\![, \quad \begin{cases} X_t \sim \mathcal{P}_{0,i}, & \text{if } A_t = 0 \\ X_t \sim \mathcal{P}_{1,i}, & \text{otherwise} \end{cases}$$

2. Apply the anomaly detector on the generated time series. Three core components need to be defined:
   (a) the appropriate kernel to identify the breakpoints using KCP,
   (b) the scoring function a
   (c) and parameters n for the length of the calibration set and λ and ℓ to define the active set.
3. Compare the detections with true anomalies and calculate the proportion of false discoveries and of false negatives.

The two criteria FDR and FNR are estimated as the average of the FDP and of the FNP over all repetitions. In the experiments of this section, the length of the time series is $T = 3000$ and it is ensured that the segments contain at least 100 points, deleting breakpoints if necessary.

The experimental framework is used in different scenarios: At Section 4.9.2, different synthetic time series are tested and analyzed. At Section 4.9.3, the effect of hyperparameter choice on performance is evaluated. At Section 4.9.4 the causes of underperformances of the anomaly detector are studied. Finally, in Section 4.9.5, the proposed anomaly detector is compared to alternative anomaly detectors using various public data collections.

4.9.2 Application on synthetic data

The goal of this section is to check if the breakpoint based anomaly detector is able to detect anomalies with a controlled FDR considering different scenarios of time series. For the first scenario, Gaussian time series are considered with breakpoints in the mean and anomalies in the tail of the distribution. This simplest scenario is used as a reference before evaluating a more complex one. The second scenario considers Gaussian mixture time series with breakpoints in

the mean and anomalies in the center of the distribution between the two Gaussian modes. In this case, the detector is checked for anomalies that are not present in the tail of the distribution. For the third scenario, 2D Gaussian time series with breakpoints in the covariance are used to evaluate the detector on multidimensional data. Indeed, the breakpoint in the covariance ensures that breakpoints and anomalies cannot be detected by applying the anomaly detector to each dimension. The third scenario evaluates the detector on heteroscedastic time series, considering Gaussian time series with breakpoints in the mean and in the variance. For the last scenario, Gaussian data with breakpoints in the variance are used to evaluate how the anomaly detector can be applied with changes in the variance, which is a more difficult case study.

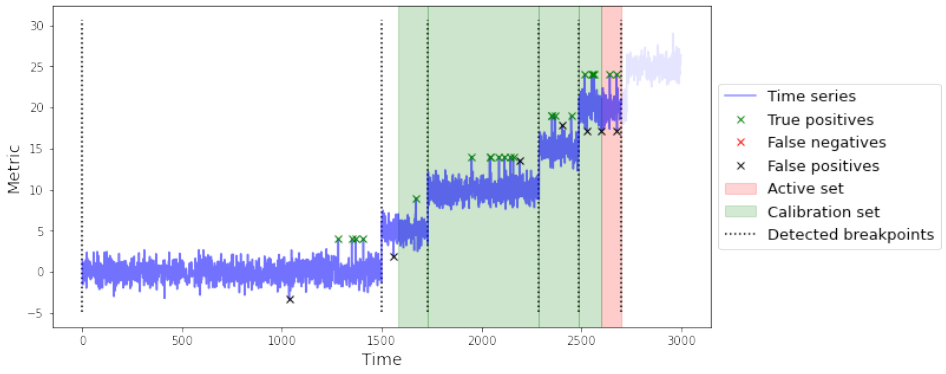
4.9.2.1 Gaussian time series with breakpoints in the mean

This scenario considers Gaussian data with breakpoints in the mean. The z-score is used to capture anomalies. The ability of the detector to control the FDR with a low FNR on different difficulties is assessed by varying the desired level of FDR control α and the size of the shift between two segments Δ.

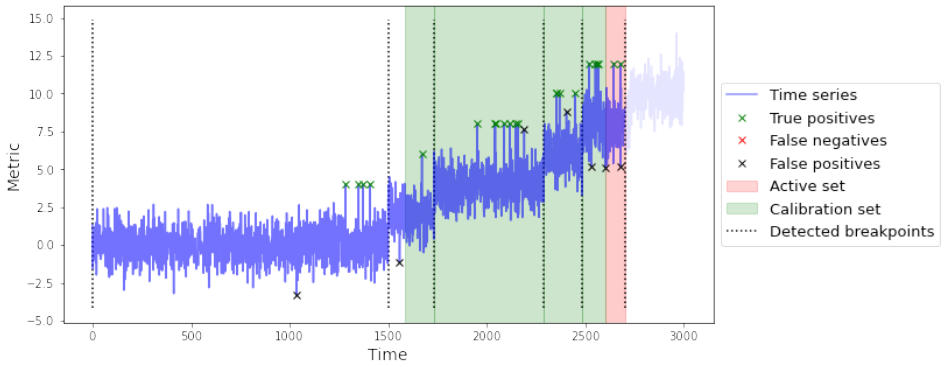
Description of the experiment By applying the framework of Section 4.9.1, multiple choices have been made:

- The Gaussian distribution is considered as the reference $\mathcal{P}_{0,1}$ and the proportion of anomalies is equal to $\pi = 0.01$. These anomalies are generated in the tail of the reference distribution and follow $\Delta'\zeta$, where ζ is the Rademacher distribution and $\Delta' = 4$ is the spike size of the anomalies.
- The transition rule between two breakpoints is a jump in the mean of size Δ taking values in $\{2, 3, 5\}$.
- For the breakpoint detector, the Gaussian kernel with bandwidth estimated using the median heuristic is considered, as presented in Section 4.4. The z-score is used as the scoring function with the mean estimated using the median estimator and the standard deviation estimated using the biweight midvariance estimator, as defined in Section 4.5.1.1.
- According to preliminary experiments in Section 4.6, the active set is built using $\hat{\lambda} = \hat{\ell} = 100$. Based on the rules defined in Section 4.8, Benjamini-Hochberg is applied on the active set with the modified parameter $\alpha' = \frac{\alpha}{1+\frac{1-\alpha}{m\pi}}$. The calibration set is built according to the rules of Section 4.8, where the value n is chosen equal to $m/\alpha' - 1$. Two cases are considered $\alpha = 0.2$ and $\alpha = 0.1$. In the case $\alpha = 0.2$, then the following values are chosen $\alpha' = 0.1$ and $n = 999$. In the case $\alpha = 0.1$, then $\alpha' = 0.05$ and $n = 1999$.

Results and analysis Figure 4.16 shows an example of anomaly detection for one time series. The x-axis is the timestamp and the y-axis the value of the generated time series, shown in blue. The light blue data points are those that are not observed at the time the results are presented. The vertical black lines are the detected breakpoints, the red band is the subseries defined as the active set, the green band is the subseries used to build the calibration set. Detected anomalies are the green crosses, false positives are the black crosses and red crosses are the false negatives. As shown in Figure 4.16a, there are no false negative and the false positives seem to be a small fraction of the true detected anomalies. As expected, the breakpoints are positioned exactly where the means of the series change. The active set contains the most recent observations. And the calibration set gathers data from several segments since the current segment does not contain enough data.



(a) $\Delta = 5$



(b) $\Delta = 2$

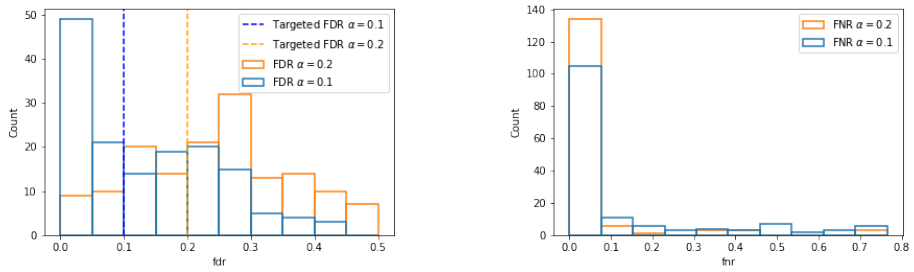
Figure 4.16: Application of our anomaly detector on Gaussian time series having breakpoints in the mean, for different shift size values Δ.

Table 4.1 gives the FDR and the FNR after having applied the anomaly detector to a collection of $B = 50$ Gaussian time series with breakpoint in the mean for different shift sizes Δ. The FNR is always close to 0. This is necessary to ensure the FDR control with the modified BH procedure. For all the cases, the FDR remains close to the desired α level. The FDR is well influenced by the choice of α level but less by the value of Δ. However, it is always slightly higher than alpha. Indeed, for $\Delta = 5$, it is equal to 0.23 instead of $\alpha = 0.20$, as shown in Table 4.1.

α	Δ	FDR	FNR
0.10	2	0.133	0.123
	3	0.134	0.111
	5	0.129	0.106
0.20	2	0.242	0.039
	3	0.242	0.042
	5	0.236	0.037

Table 4.1: FDR and FNR for anomaly detection on Gaussian time series having breakpoints in the mean according to the α level and the shift size Δ.

The histogram in Figure 4.17 shows more detailed results applied to the collection of time series for different values of α parameter. Figure 4.17a shows the distribution of the FDR values compared to the target FDR in vertical lines. Figure 4.17b shows the distribution of the FNR values. As shown in Figure 4.17a,the performance of the anomaly detector is poor for some time series since the FDR values are higher and far from the target FDR. This explains why the measured FDR is slightly higher than the targeted FDR in Table 4.1. The diagnosis of this inefficiency will be examined in Section 4.9.4. In the next Sections 4.9.2.2, 4.9.2.3, 4.9.2.3, 4.9.2.4 and 4.9.2.5 the anomaly detector is applied and checked to more complex time series.



(a) FDR (b) FNR

Figure 4.17: Histograms of the FDR and FNR for different targeted FDR α levels.

4.9.2.2 Gaussian mixture time series with breakpoints in the mean

In this section, the aim is to show how to handle anomalies that occur between two modes of a Gaussian mixture. These anomalies, which do not occur in the tail of a distribution, cannot be detected by z-scores because they are close to the mean. Therefore, it is necessary to adapt to this new situation by using another atypicity score, such as the kNN score introduced in [86]. Indeed, in this case, anomalies can be characterized by their distance from other segment data.

Description of the experiment The anomaly detector applied to Gaussian mixture data considers the reference distribution $\mathcal{P}_{0,1} = 0.5\mathcal{N}(\Delta', 1) + 0.5\mathcal{N}(-\Delta', 1)$, with an anomaly spike size of $\Delta' = 6$. The anomalies are chosen to be equal to 0 to ensure they lie in the middle between the two Gaussian distributions.

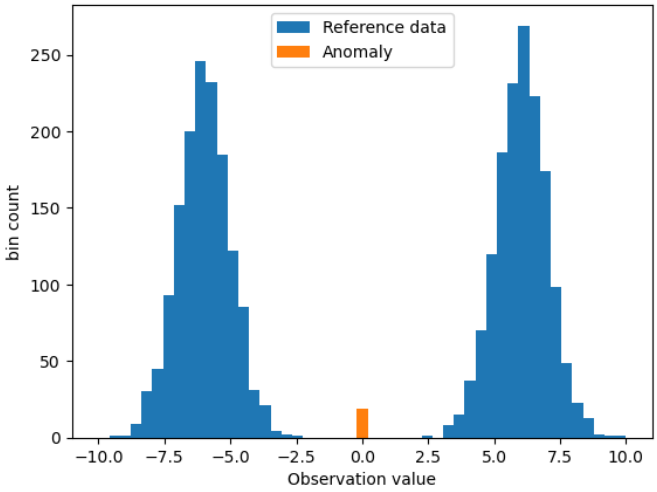


Figure 4.18: Histogram that represent the Gaussian mixture reference distribution with anomalies in the center.

As explained previously, to adapt to this new difficulty of time series data with Gaussian mixture, the atypicity score needs to be chosen accordingly. The kNN score introduced in [86] is applied. To ensure that the distribution of the score is the same between two segments and not affected by segment cardinality, the kNN distance is computed after having resampled $B_s = 100$ points from the segment. To obtain a robust score, the number k of nearest neighbors should be chosen carefully because the kNN distance should not be affected by the presence of anomalies in the segment. In particular, the k nearest neighbors of an anomaly should not be an anomaly, otherwise the distance will be close to 0, which leads to a false positive. By choosing $k = 10$ and ensuring $k/B = 0.1 >> 0.01 = \pi$, this issue is avoided with high probability. Experimental parameters not specified in this section have the same values as in Section 4.9.2.1.

Results and analysis The result in Figure 4.19 clearly shows that for this example, the anomaly detector is able to detect the breakpoints, in the dashed black lines, and the anomalies, represented by the green crosses, with few false positives. The anomaly detector has been applied to 50 time series and the results are summarized in Table 4.2. The FDR is controlled at the desired level of 0.1 or 0.2 while the FNR is slightly higher compared to the Gaussian case in Table 4.1. This is probably due to the kNN score, which is less efficient than the z-score.

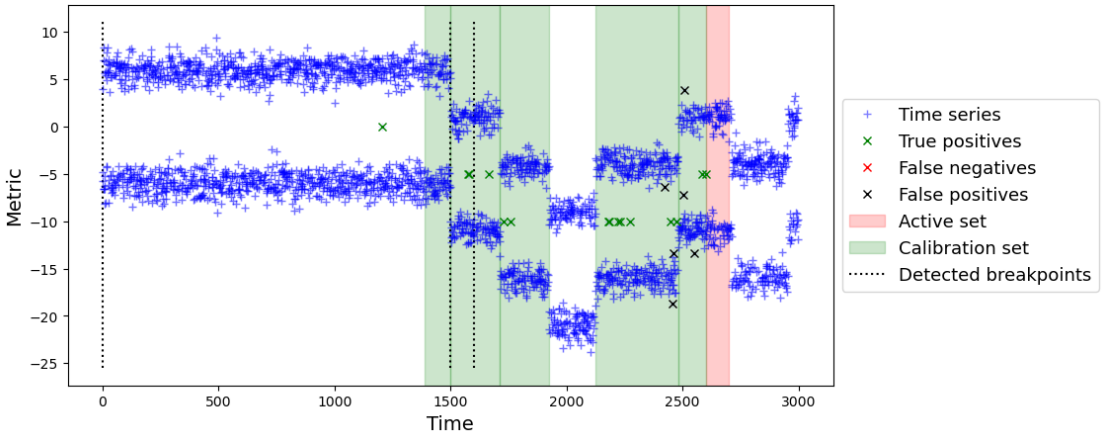


Figure 4.19: Application of our anomaly detector on Gaussian mixture time series having breakpoints in the mean.

α	FDR	FNR
0.1	0.118	0.246
0.2	0.202	0.137

Table 4.2: FDR and FNR for anomaly detection on Gaussian mixture time series with breakpoints in the mean according to α level.

4.9.2.3 2D Gaussian time series with breakpoint in the covariance

In this section, the aim is to show how to handle anomalies that occur on multidimensional data. Previously, the kernel method in KCP demonstrated high accuracy in detecting breakpoints for univariate time series data. Hopefully, the paper [113] shows that this kernel method is also applicable to multivariate time series. Once the time series is segmented, a scalar atypicity score is computed for the multidimensional data points. An alternative would be to apply univariate anomaly detection to each univariate time series. However, some breakpoints, such as those occurring in the covariance, cannot be detected by this alternative method.

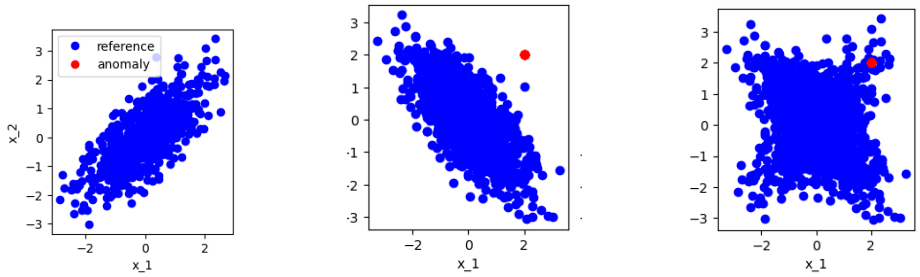
Description of the experiment Data are generated according the following rule:

$$\forall t \in [\![1, T]\!], \quad X_t = \begin{pmatrix} X_{1,t} \\ X_{2,t} \end{pmatrix} \sim \mathcal{N}(0, \Sigma_t) \tag{4.40}$$

With the covariant matrix equal to:

$$\Sigma_t = \begin{cases} \begin{pmatrix} 1 & 0.7 \\ 0.7 & 1 \end{pmatrix} & \text{if } t \leq \tau_1 \\ \begin{pmatrix} 1 & -0.7 \\ -0.7 & 1 \end{pmatrix} & \text{else} \end{cases} \tag{4.41}$$

The reference distribution generates two-dimensional Gaussian data X_t. For each component the mean is 0 and the standard deviation is 1. To simplify the generation, one breakpoint τ_1 is considered linked to the change of the covariance from 0.7 to -0.7. Figure 4.20a shows that the covariance is positive before the breakpoint and negative after the breakpoint in Figure 4.20a. Anomalies are considered in the second segment, and are set to the value $(1, 1)$. This value has interesting properties to evaluate the capacity of the anomaly detector. First, "1" appears as a typical value at each one dimensional component of the time series. This implies that the anomalies cannot be detected by working on each component independently. Second, the value $(1, 1)$ is fairly typical before the breakpoint τ_1, as shown Figure 4.20a. Consequently, the breakpoint detector enables the detection of anomalies while they are hidden in the data mixture as shown in Figure 4.20c.



(a) Before the breakpoint. (b) After the breakpoint. (c) Data mixture.

Figure 4.20: 2D Gaussian data with breakpoint in covariance matrix

For this scenario, the Gaussian kernel is used to detect the breakpoint in the covariance. As a characteristic kernel, it should detect the change in the covariance, which is the change at the second moment order. The median heuristic is used to select the bandwidth. Since each component cannot be treated independently to detect anomalies, the Mahalanobis distance [108] is preferred over the Euclidean distance. The Mahalanobis distance is defined as the following, where $\hat{\mu}$ is the estimated mean vector and $\hat{\Sigma}$ is the estimated covariance matrix.

$$s_t = \sqrt{(X_t - \hat{\mu})^T \hat{\Sigma}^{-1} (X_t - \hat{\mu})}$$

To ensure a good atypicity score, the estimator of the covariance has to be robust and efficient, as shown in Section 4.5. Inspired by the results of Section 4.6.1.2, the biweight-midcovariance [119] is used to estimate each coefficient of the matrix $\hat{\Sigma}$.

Results and analysis The result is represented for one example in Figure 4.21. The multidimensional time series is represented using one plot for each dimension. The anomaly detector successfully detects the breakpoints in the dashed black lines, and the anomalies that are represented by green dots with few false positives. The anomaly detector has been applied to 50 time series and the results are summarized in Table 4.3. The FNR is close to 0 and the FDR is smaller than expected, 0.12 instead of 0.2. This confirms that the detector can be applied to multidimensional data with minor adaptation.

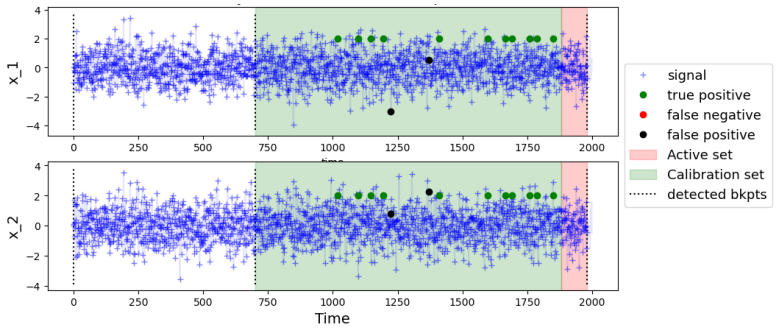


Figure 4.21: Application of our anomaly detector on 2D Gaussian time series having breakpoints in the covariance.

α	FDR	FNR
0.2	0.126	0.054

Table 4.3: FNR and FDR for anomaly detection on 2D Gaussian time series with breakpoint in the covariance.

4.9.2.4 Gaussian data with breakpoints in the mean and in the variance

In the experiments conducted so far, all scenarios considered homoscedastic time series where the change is in the mean while the variance is constant between two segments. In this section, the case of heteroscedasticity in time series is studied where the variance changes between two segments. Therefore, time series will have parts where the variance is very low and parts where it is very high. The struggle is that a kernel may be good at detecting breakpoints in a low variance context, but have difficulty when the variance is high, and vice versa. Therefore, several kernels are tested by varying the bandwidth size. Kernel methods are used instead of methods specialized in detecting breakpoints in the variance, because the aim is to keep a method capable of detecting any type of breakpoint.

Experiment description. Let's consider a time series generation process and an anomaly detector described in Section 4.9.1. To adapt to the heteroscedasticity hypothesis, the transition rule is modified so that at each breakpoint the variance changes as follows, where Δ_σ is the variance shift size equal to 2:

$$\sigma_{i+1} = \exp(\zeta_{\sigma,i} \ln \Delta_\sigma) * \sigma_i$$

To ensure that the variance covers a wide range of values, the variable ζ_i is chosen asymmetric. In the case of this experiment, ζ_i has a probability of 0.9 of being $+1$. Thus, the variance is more likely to increase than to decrease at each breakpoint. To ensure the visibility of the breakpoint in the mean to any variance, the size of the shift in the mean needs to be proportional to the maximum of the variance of the segment before and after the breakpoint, as described in the following:, where Δ_μ is the mean shift size equal to 2:

$$\mu_{i+1} = \zeta_{\mu,i+1} \Delta_\mu \max(\sigma_i, \sigma_{i+1}) + \mu_i$$

According to the median heuristic, breakpoints are easily detected by a Gaussian kernel when the standard deviation of the data is of the same order as the bandwidth h. Several kernels are tested:

- Gaussian kernel with bandwidth $h = 1$. This kernel with a small bandwidth is relevant to detect breakpoints when the variance of the time series is small, but may fail when the variance is high.
- Gaussian kernel with bandwidth $h = 100$. In this situation, the kernel is more relevant to detect breakpoints when the variance is high.
- To consider both scenarios, where breakpoints appear in some parts of the series with high variance and in parts of the series with low variance, a linear combination of the two Gaussian kernels may be a good response. This kernel is characteristic as a sum of two characteristic kernels and is defined by:

$$K(x, y) = 0.5K_{h_1}(x, y) + 0.5K_{h_2}(x, y) \tag{4.42}$$

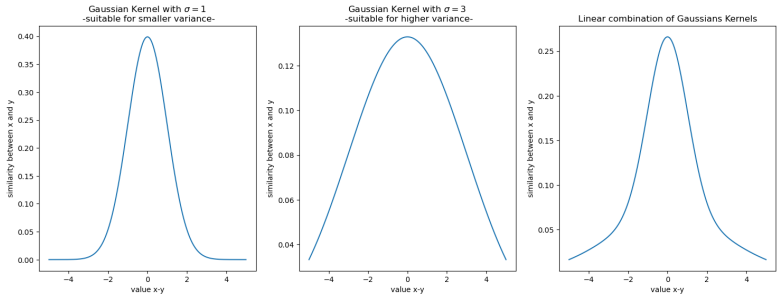


Figure 4.22: Illustration of the different kernels

Result analysis The anomaly detector is applied three times to the same time series, changing only the kernel used in Figures 4.23, 4.24 and 4.25:

- Figure 4.23 illustrates the result using the Gaussian kernel with small bandwidth, $h = 1$. The breakpoint was not detected at ①, which leads to a false negative ② and a large number of false positives at ③.
- Figure 4.24 illustrates the result using the Gaussian kernel with large bandwidth, $h = 100$. At the position ①, the breakpoint with low variance is not detected. It leads to false positives at ② because data with different variances belong to the same calibration set.
- Figure 4.25 illustrates the result when using the linear combination of the two Gaussian kernels. All breakpoints are detected, reducing the number of false positives and false negatives.

The anomaly detector has been applied to 50 time series and the FDR and FNR results are summarized in Table 4.4. Different kernels, bandwidth h, are considered in combination with α levels in $\{0.1, 0.2\}$:

- Gaussian kernel (labeled Gaussianh) with bandwidth h in $\{1, 10, 100\}$
- Linear combination of two Gaussians (labeled CombG1G100)

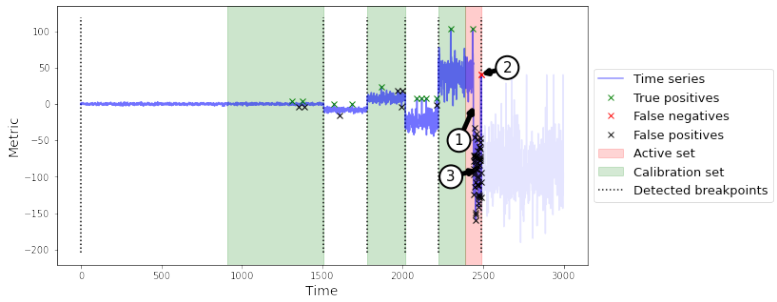


Figure 4.23: Application of our anomaly detector on Gaussian time series having breakpoints in the mean and in the variance, using a Gaussian kernel having a small bandwidth.

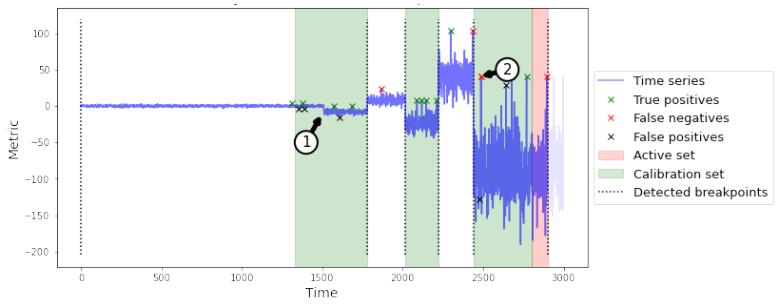


Figure 4.24: Application of our anomaly detector on Gaussian time series having breakpoints in the mean and in the variance, using a Gaussian kernel having a large bandwidth.

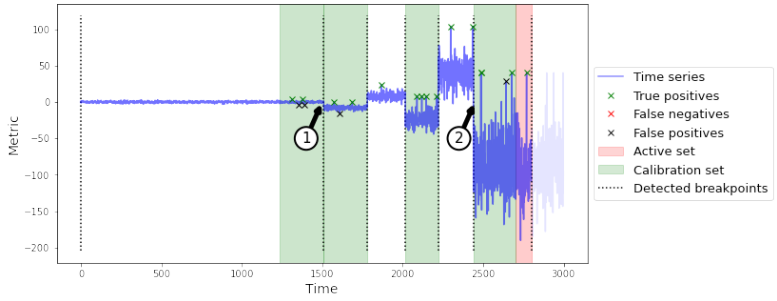


Figure 4.25: Application of our anomaly detector on Gaussian time series having breakpoints in the mean and in the variance, using a linear combination of two Gaussian kernels.

The performances are strongly influenced by the kernel bandwidth: The FNR is lower when using the Gaussian kernel with bandwidth $h = 1$ or using the combination of Gaussian kernels while it is high when using kernels with larger bandwidth. The FDR is slightly higher than expected α for all tested kernels. However, the FDR is smaller when using the combination of Gaussians compared to the Gaussian kernel with $h = 1$. Thus, anomaly detection remains possible when the variance of the time series changes under heteroscedasticity. However, there is no general way to build a dedicated kernel that responds to this scenario, but combining specialized kernels to adapt to the different regimes of the time series seems to be a promising approach.

α	Kernel	FDR	FNR
0.10	Gaussian1	0.188	0.054
	Gaussian10	0.127	0.136
	Gaussian100	0.148	0.456
	CombG1G100	0.134	0.057
0.20	Gaussian1	0.323	0.017
	Gaussian10	0.232	0.102
	Gaussian100	0.232	0.397
	CombG1G100	0.253	0.018

Table 4.4: FDR and FNR for anomaly detection on Gaussian time series with breakpoints in the mean and in the variance according to the α level and the chosen kernel

4.9.2.5 Gaussian data with breakpoints in the variance

In this section, the more challenging scenario of time series with changes in variance without a shift in mean is addressed.

Description of the experiment To generate the data, a Gaussian distribution is used as the reference one. The breakpoints in the variance are generated according to the rule described in Eq. 4.39. Since the variance of the time series changes along the time series, it may be difficult to detect all the breakpoints with the same kernel. To evaluate the detector in this scenario, it is based on the same kernels defined in Section 4.9.2.4 and on the z-score atypicity function.

Results and analysis Figures 4.26 and 4.27 show two examples of anomaly detection. In Figure 4.26, all the breakpoints are successfully detected, allowing correct anomaly detection with few false positives. In Figure 4.27, the procedure fails and no breakpoint is detected in ①. After the change with higher variance, all data are considered as anomalies. It is an evidence that the efficiency of the anomaly detector is strongly influenced by its ability to detect the true breakpoints.

Table 4.5 summarizes the FDR and FNR results obtained for 50 time series using the same kernels and α levels as in Table 4.4. In all cases, the anomaly detection shows a poor accuracy, since on one side there is a lack of control of the FDR with respect to the target value alpha, and on the other side the FNR is very high. However, the best FNR and FDR values are obtained for the combination of Gaussian kernels, which allows better detection of breakpoints in the variance.

These results show how challenging the case of time series with breakpoints in the variance is. Indeed, the change in the variance is much harder to detect than the shift in the mean presented

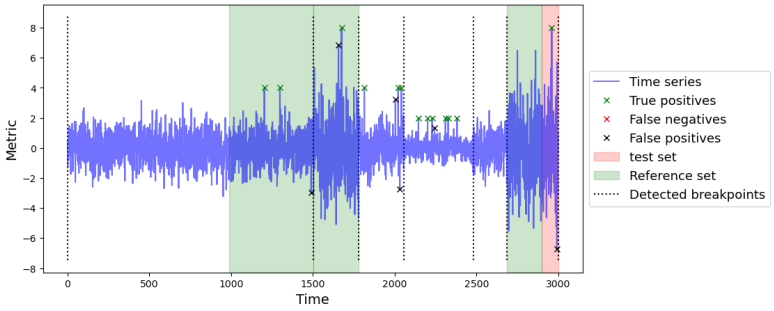


Figure 4.26: Example of successful anomaly detection on time series with breakpoints in the variance.

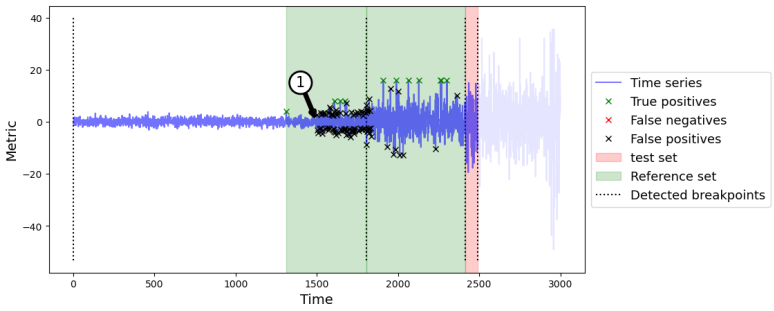


Figure 4.27: Examples of failure of anomaly detection on time series with change in the variance.

α	Kernel	FDR	FNR
0.10	Gaussian1	0.272	0.321
	Gaussian10	0.806	0.712
	Gaussian100	0.835	0.599
	CombG1G100	0.229	0.298
0.20	Gaussian1	0.313	0.241
	Gaussian10	0.649	0.511
	Gaussian100	0.685	0.396
	CombG1G100	0.282	0.225

Table 4.5: FDR and FNR for anomaly detection on Gaussian time series with breakpoint in the variance according α level and chosen kernel.

in Section 4.9.2.1. One approach is to carefully tune the kernel by choosing the right combination of kernels to enable the detection of specific types of breakpoints.

4.9.3 How hyperparameter choices affect the the anomaly detector performances?

The goal of this section is to show how incorrect hyperparameter values of the anomaly detector lead to a degradation of the anomaly detector's performances. This evaluation is done for three core components: the variance estimator, the cardinality of the calibration set, and the cardinality of the active set. The hyperparameters of these components are intentionally set very far from the recommendations given in Sections 4.5 and 4.6 and the consequences are observed and discussed to confirm the recommendations, the rules and the analyses stated in the chapter.

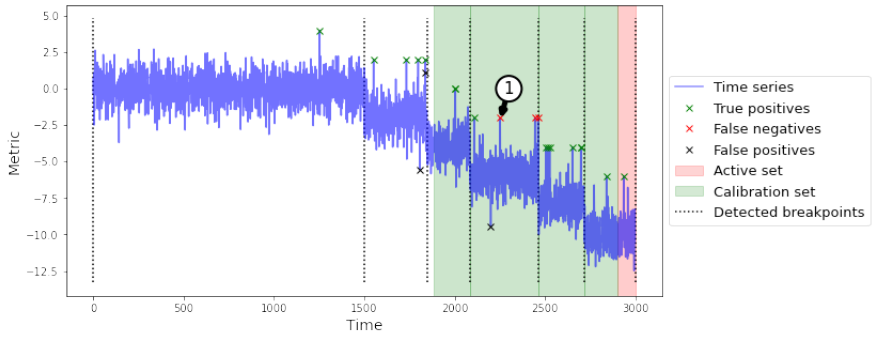
4.9.3.1 Bad choice of variance of segments estimator

In Section 4.5, it was established that a good atypicity score should respect two properties: robustness to the presence of anomalies in the training set and efficiency. In this scenario, a bad choice is made for an atypicity score that does not respect the requirements of being robust and efficient. To simulate this case and evaluate the effects, the experiment with Gaussian time series having breakpoint in the mean introduced in Section 4.9.2.1 is reused by replacing the biweight estimator of the variance in the z-score function by the MLE estimator or the MAD estimator. Indeed, MLE estimator is efficient but not robust, and the MAD estimator is robust but not efficient while the biweight midvariance is robust and efficient.

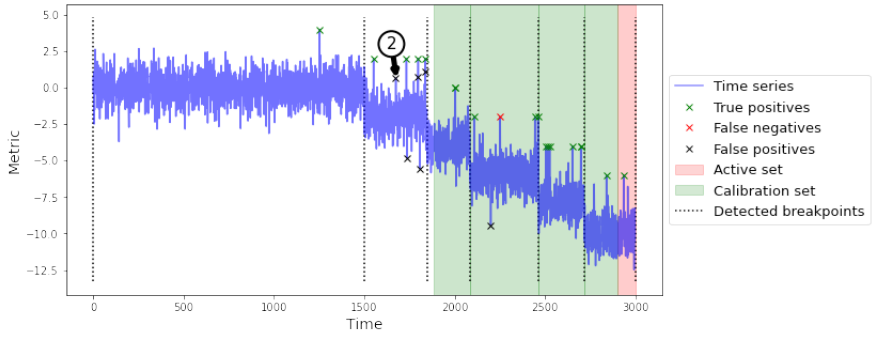
Result and analysis To analyze and compare the effect of the different estimators, the same example is considered in Figure 4.28, for different variance estimators. Since the MLE estimator is not robust, Figure 4.28a at ① shows false negatives due to variance overestimation caused by the presence of anomalies in the current segment. The choice of the robust MAD estimator reduces the false negatives while it generates a higher number of false positives as shown in Figure 4.28b at ②. The variance is underestimated due to the lack of data points and the lower efficiency of MAD. The Biweight estimator is advantageous as it is both robust and efficient and is able to reduce false positives and false negatives, as shown in Figure 4.5.1.

In Figure 4.29, the boxplots represent the distribution of FNR and the FDR over a set of 50 time series based on the standard deviation estimator (MLE, MAD or BW). Paired permutation tests [54] are used to compare the performances of two estimators. For each pair of estimators, the hypothesis tested is: "The mean FDR (or FNR) is the same using these two variance estimators". The results are represented by adding a symbol ("ns" the difference is not significant, "*" significance at 5%, "**" significance at 1%, "***" significance at 0.1%) between the two tested estimators.

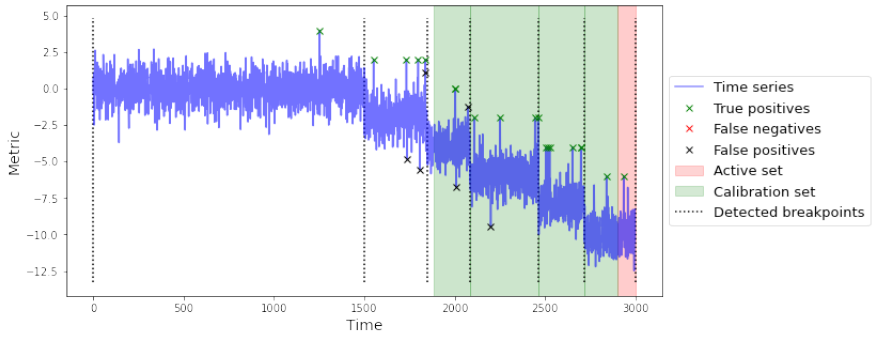
The FDR and FNR results are summarized in Table 4.6. The FNR is significantly higher when the MLE variance estimator is used compared to the more robust MAD and biweight midvariance estimators, which have close performances. However, the FDR is significantly higher when the MAD is used compared to the biweight midvariance estimator, for which the FDR is better controlled.



(a) MLE



(b) MAD

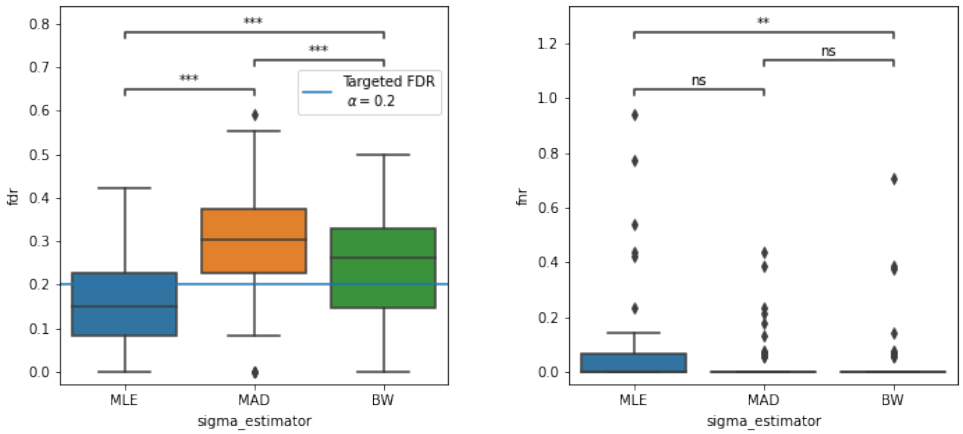


(c) BW

Figure 4.28: Application of our anomaly detector on Gaussian time series having breakpoints in the mean, using different variance estimators.

sigma_estimator	FDR	FNR
MLE	0.16	0.08
MAD	0.29	0.04
BW	0.24	0.04

Table 4.6: FNR and FDR according to the choice of the variance estimator.



(a) FDR (b) FNR

Figure 4.29: Boxplots of the FNR and FDR according to the choice of the variance estimator.

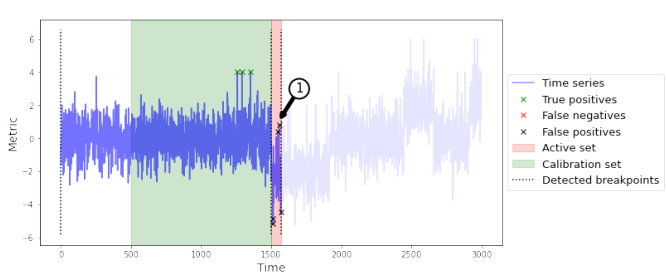
4.9.3.2 Bad choice for active set cardinality

Section 4.6 introduced the notion of an active set to deal with the uncertainty of status. It also provides rules to compute the cardinality of the calibration test. In this section the relevance of this rule is evaluated.

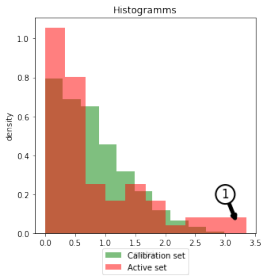
Description of the experiment To evaluate the performance degradation due to a bad choice of the active set cardinality, the experiment framework introduced in Section 4.9.2.1 is reused. According to the results of the experiments in Section 4.6.1.2 and Section 4.6.2.2, status can be ensured with strong confidence with an active set cardinality equal to $m = 100$. For each time series generated, two anomaly detectors are applied, one with an active set cardinality equal to 100 and the second with an active set cardinality equal to 10.

Results and analysis In order to understand how the active set improves the anomaly detector, the results are observed at two different instants: at time $t = 1570$ in Figures 4.30a and 4.30b, and at time $t = 1600$ in Figures 4.30c and 4.30d. The histograms of the z-scores of the calibration set in green and the active set in red are shown in Figures 4.30b and 4.30d. At time $t = 1570$, the new current segment contains few points, resulting in a variance estimation error and an overestimation of the z-score of the active set in ① Figure 4.30b and false positives in ① Figure 4.30a. At time $t = 1600$, the segment has acquired new data points, the variance estimate is improved and the z-score is not overestimated in ② Figure 4.30d. The number of false positives is reduced in ② Figure 4.30c. The status of the data point at $t = 1570$ is corrected at time $t = 1600$ because the active set is large enough, otherwise its status would be fixed to the wrong one.

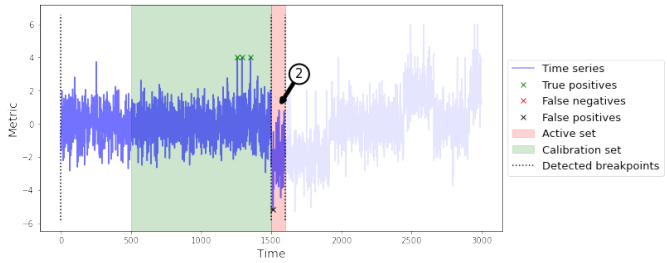
Figure 4.31 illustrates the boxplots of the FDR distribution according to the active set cardinality. The results, summarized in Table 4.7, show that the FDR is significantly higher when the active set has a cardinality of $m = 10$. On the contrary, using a cardinality of $m = 100$ allows to control the FDR at the desired level $\alpha = 0.2$. This experiment illustrates the benefits



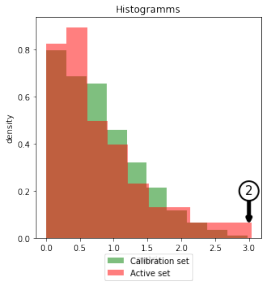
(a) Example of false positive, $t = 1570$



(b) Example of false positive



(c) Example of accurate detection, $t = 1600$



(d) Example of false negative

Figure 4.30: Abnormality status update after new data points acquisition in the current segment.

of following the recommendations in Section 4.6 to improve anomaly detection performances.

α	m	FDR	FNR
0.2	10	0.529	0.006
	100	0.186	0.053

Table 4.7: FDR and FNR mean according to the active set cardinality.

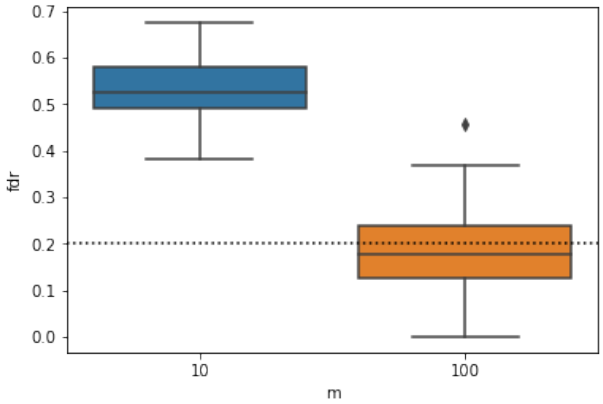


Figure 4.31: FDR boxplots according to the active set cardinality

4.9.3.3 Bad choice of cardinality for the calibration set

It was established in Section 4.8 that the FDR can only be controlled if the cardinality of the calibration set takes specific values. This section verifies this claim in the case of breakpoint based anomaly detection.

Description of the experiment To evaluate the degradation of the FDR control due to a bad choice of the calibration set cardinality, time series are generated according to the framework design introduced in Section 4.9.2.1. For each generated time series, with the target FDR $\alpha = 0.2$ (resp. $\alpha = 0.1$) and the active set cardinality $m = 100$, two anomaly detectors are applied: one with a calibration set cardinality equal to 999 (resp. 1999) respecting the recommendation. The second with a calibration set cardinality equal to 1000 (resp. 2000), not respecting the recommendation. Indeed, since the proportion of anomalies is equal to $\pi = 0.01$, the goal of a FDR equal to $\alpha = 0.2$ (resp. $\alpha = 0.1$) can be achieved using Benjamini-Hochberg with $\alpha' = 0.1$ (resp. 0.05) according to Section 4.8. The calibration set cardinality should then be equal to 999 (resp. 1999) according to Eq. 4.37.

Results and analysis The results in Table 4.8 show that the FDR is controlled at the desired level for $n = 999$ and $n = 1999$, while the FNR is higher. This confirms that the FDR can only be controlled by selecting the parameter n using the rule in Section 4.8. To reduce the FNR while maintaining control of the FDR, the values n must be chosen among the values $\{1999, 2999, 3999, \ldots\}$ as discussed in Section 3.3.2.

α	n	FDR	FNR
0.2	999	0.21	0.030
	1000	0.30	0.0
0.1	1999	0.1	0.1
	2000	0.16	0.03

Table 4.8: FDR and FNR according to the calibration set cardinality.

4.9.4 Diagnose the causes of underperformance

Our Breakpoint based anomaly detector has been tested on different time series data in Section 4.9.2, it shows good performances to ensure low FNR with an FDR almost controlled in different cases. However, the FDR is never completely under control, and is always slightly higher than expected. This section examines why this lack of complete control of the FDR occurs by replacing some estimators with knowledge of the true values and evaluate the effect on the FDR.

4.9.4.1 Description of the experiment

The BKAD is applied to the synthetic time series, where some estimators are replaced by true knowledge, called oracle version. Three estimators are chosen to be replaced by their oracle versions:

- The breakpoint estimator: can be replaced by the true breakpoint position,
- The mean and standard deviation estimators: can be replaced by their true values,
- The anomaly removed: As described in Section 4.7 when building the calibration set, estimated anomalies are removed to avoid biasing the estimation of the p-values. The oracle version of this is to remove the true anomalies.

Using the framework from Section 4.9.1, five anomaly detectors are applied to each time series. Multiple combinations of the true knowledge (marked "O") versus estimated values (marked "E") are used to produce different versions of anomaly detectors described in Table 4.9. As an

example, for detector 3, the breakpoints and anomalies in the calibration set are detected using their true values, but the segment mean and variance parameters are estimated.

Detector	Breakpoint	Mean and variance	Anomaly Removing
Detector 1	O	O	O
Detector 2	O	O	E
Detector 3	O	E	O
Detector 4	E	E	O
Detector 5	E	E	E

Table 4.9: Description of the different detectors.

The significance of the results is checked using permutation tests. It is possible that the cause of this underperformance depends on the data distribution or on breakpoint types. Different probability distributions are tested with different kinds of shifts.

4.9.4.2 Results and analysis

The complete empirical results can be found in 4.11. The performances of the different detectors are evaluated on a different laws generating the time series (Student, Gaussian, Mixture of Gaussians noted MoG). The FDR and FNR distributions are represented by a boxplot with the significance differentiating two detectors ("ns" the difference is not significant, "*" significance at 5%, "**" significance at 1%, "***" significance at 0.1%).

In the following paragraphs, the effects of the various core components are studied: breakpoint detector, mean and variance segment estimator and anomalies removed from the calibration set.

Breakpoint Estimation Table 4.10 shows the performance of anomaly detectors 3 and 4 (see Table 4.9) for different types of data and shifts. The only difference between the two detectors is that Detector 3 uses a breakpoint detector while Detector 4 has knowledge of true breakpoints. The bold values highlight the cases where the difference between the two estimators is significant. Table 4.10 illustrates that the breakpoint estimation does not strongly affect the FDR performance except in the case where breakpoints occur in the variance. This is expected since breakpoints in the variance are more difficult to to detect, as discussed earlier in Section 4.9.2.5. FNR increases in few cases where the breakpoint positions are estimated.

Type of shift	law	α	Breakpoints	FDR	FNR
Mean	Gaussian	0.10	E	0.104	0.105
			O	0.100	0.091
		0.20	E	0.182	0.054
			O	0.176	0.054
	Student	0.10	E	0.119	0.066
			O	0.117	0.065
		0.20	E	0.199	0.033
			O	0.198	0.032
	MoG	0.10	E	0.113	0.131
			O	0.108	0.124
		0.20	E	0.186	0.072
			O	0.190	0.071
Mean and var.	Gaussian	0.10	E	**0.114**	0.090
			O	0.099	0.078
		0.20	E	**0.188**	0.051
			O	0.167	0.040
Variance	Gaussian	0.10	E	**0.200**	**0.214**
			O	0.110	0.109
		0.20	E	**0.257**	**0.128**
			O	0.174	0.062

Table 4.10: Anomaly detector performances with and without knowledge of true breakpoint positions, according different time series.

Segment mean and variance parameters Table 4.11 shows the performance of anomaly detectors 1 and 3 (see Table 4.9) for different types of data and shifts. The only difference between the two detectors is that Detector 3 estimates the mean and the variance parameters of the segments while Detector 1 has knowledge of the true parameters. According to Table 4.11, the estimators do not strongly affect the FDR of the anomaly detector. There are few significant differences, displayed in bold, which are smaller than in Table 4.10.

type of shift	law	α	Mean and variance	FDR	FNR
Mean	Gaussian	0.10	E	0.082	0.043
			O	0.105	0.000
		0.20	E	0.167	0.000
			O	0.188	0.000
	Student	0.10	E	0.117	0.065
			O	0.113	0.056
		0.20	E	0.198	0.032
			O	0.196	0.026
Mean and var.	Gaussian	0.10	E	**0.091**	0.000
			O	0.107	0.000
		0.20	E	**0.167**	0.000
			O	0.200	0.000

Table 4.11: Anomaly detector performances with knowledge of the true segment mean and standard deviation values and with estimation of these parameters, according different time series.

Anomalies Removing Table 4.12 represents the results for different detectors considering different laws, alpha levels and kind of shift. The four detectors are chosen to identify the effect of removing detected anomalies from the calibration set instead of removing the true anomalies, in case other components are estimators and in case other components are oracles. Note that for Gaussian Mixture (MoG), the "Mean and Variance" component is marked with a "X", since the kNN atypicity score does not use mean and variance parameters. It is clear that the control of the FDR is worse when the calibration set is built based on detected anomalies. Indeed, the false positives and false negatives detected at time t will badly affect the detection at time $t+1$. Despite the fact that a robust score is chosen, these observations lead to a conclusion that the p-value estimator is sensitive to:

- False negatives: If there is a missed anomaly in the calibration set, the p-values of all data points in the active set will be underestimated. This situation leads to generate more false negatives, which will confound the calibration sets of subsequent instants.
- False positives: The p-value estimator is also sensitive to false positives due to the way the calibration set is constructed. As a reminder, detected anomalies are replaced by a random points belonging to a segment similar to the current segment. The problem arises when an anomaly is falsely detected. Generally speaking a false positive is a point with a high score. When a false positive is replaced with a random point, its score will be statistically lower. Thus, removing the false positives from the calibration set reduces the average score in the calibration set and consequently reduces the p-values of the data points in the active set. This leads to more false positives, which will affect the construction of calibration sets at later times.

Type of shift	Law	α	Breakpoint	Mean and variance	**Anomaly removing**	FDR	FNR
Mean	Gaussian	0.1	E	E	E	**0.134**	0.123
			E	E	O	0.104	0.105
			O	O	E	**0.165**	0.041
			O	O	O	0.121	0.048
		0.2	E	E	E	**0.242**	0.039
			E	E	O	0.182	0.054
			O	O	E	**0.301**	0.018
			O	O	O	0.197	0.018
	Student	0.1	E	E	E	**0.158**	0.059
			E	E	O	0.119	0.066
			O	O	E	**0.154**	0.035
			O	O	O	0.113	0.056
		0.2	E	E	E	**0.289**	0.026
			E	E	O	0.199	0.033
			O	O	E	**0.301**	0.013
			O	O	O	0.196	0.026
	MoG	0.1	E	X	E	0.118	0.246
			E	X	O	0.113	0.131
			O	X	E	0.103	0.294
			O	X	O	0.108	0.124
		0.2	E	X	E	0.202	0.137
			E	X	O	0.186	0.072
			O	X	E	0.221	0.111
			O	X	O	0.190	0.071
Mean and var.	Gaussian	0.1	E	E	E	0.134	0.057
			E	E	O	0.114	0.090
			O	O	E	**0.955**	0.022
			O	O	O	0.119	0.054
		0.2	E	E	E	**0.253**	0.018
			E	E	O	0.188	0.051
			O	O	E	**0.961**	0.021
			O	O	O	0.205	0.029

Table 4.12: Anomaly detector performances with and without knowledge of true anomalies for removing anomalies, according different time series.

Conclusion The conclusion of this analysis is that most of the underperformance relative to the ideal case, such as higher than expected FDR, is explained by the non-robustness of the empirical p-value estimator and the contamination of the calibration set by false negatives and false positives.

4.9.5 Evaluation against competitors

After studying the conditions that must be met to ensure high detection performance and control of the FDR in the previous sections, the breakpoint detection based anomaly detector (BKAD) proposed in this chapter is compared to alternative anomaly detectors from the literature on different data collections. The goal is to determine if and under which conditions the new anomaly detector can improve the state of the art.

4.9.5.1 Methods

BKAD is evaluated against state-of-the-art anomaly detectors presented in the review [143]. The most representative unsupervised anomaly detectors for univariate time series data are selected. The implementation of [143] is used, with default hyperparameters. The detectors selected are those that theoretically capable of detecting anomalies in piecewise iid data. These algorithms fall into two categories: the one that build a context such as a segment, a sliding window or a cluster, and on the other that use subseries instead of single points. On the other hand, predictive or regression models are of little interest on piecewise iid data.

Median [10] A sliding windows is used to estimate the median and dispersion parameter of last observations. The atypicity score used is the z-score. The main difference with the BKAD approach is the use of sliding windows instead of using a breakpoint detector to define the segments.

CBLOF [76] Cluster based local outlier factor identifies the cluster to which individual points belong, then it computes the local outlier factor associated with that cluster. The use of clusters is similar to that of breakpoints in that it attempts to group similar points together, but has no temporal notion.

Sub. IF [106] The method divides the time series in subsequences and uses Isolation Forest on the subsequences set.

DWT [162] Method based on wavelet to remove noise. Atypicity score is computed using the Gaussian distribution on the Discrete Wavelet Transform, with Haar wavelet. Anomalies can be detected as abnormal Haar coefficients.

Sub. LOF [30] The method divides the time series in subsequences and uses Local Outlier Factor on the subsequences set.

FFT [134] Method based on Fast Fourier Transform. It uses Local outlier factor on the Fast Fourier Transform of the subsequences. Anomalies can be detected as abnormal frequency coefficients.

4.9.5.2 Threshold

After applying these different methods, an atypicity score is obtained. This score is sufficient to compute the AUC metric, but does not allow detection and calculation of the FDR and FNR without thresholds. To calculate these thresholds, the method introduced in Chapter 3 is used, which guarantees FDR control at a fixed α level in case the time series of scores is iid. The threshold of BKAD is chosen as described in Section 4.8. Here α is set to 0.2 for all detectors and time series.

4.9.5.3 Data

To ensure a comprehensive analysis, different kind of time series data are considered:

- Time series with breakpoints
- Time series with seasonality
- Residual from time-series with seasonality
- Real data time series

Time series with breakpoints The time series with breakpoints are generated according to the experimental design presented in Section 4.9.1 with the following hyperparameters: the reference distribution is Gaussian $\mathcal{P}_{0,1} = \mathcal{N}(0,1)$ and all anomalies follow the law of 4ζ, where zeta follows the Rademacher distribution. Anomalies are generated with a proportion of $\pi = 0.01$. The breakpoint positions are generated according to the Poisson process with an average segment length of 125. To avoid having too few segments, breakpoints are removed if a segment has less than 100 points. For the benchmark *breakpoint-mean*, breakpoints occur in the mean with a $\Delta = 2$. And, for the benchmark *breakpoint-var*, breakpoints occur in the variance with $\Delta = 1.5$.

Time series with seasonality To study how the anomaly detector behaves on time series not following the statistical model introduced in Section 4.2.1, time series with seasonality and trend are considered.

Let the following components be given:

1. $R_t \sim \mathcal{N}(0, \sigma)$, the residual, $\sigma = 1$
2. $A_t \sim B(\pi)$ the abnormality variable, $\pi = 0.01$
3. $S_{1,t} = A_1 \sin(2\pi f_1 t)$ the seasonality with long period, where the amplitude A_1 and the frequency f_1 are random variables, $A_1 \in \{1, 3, 5\}$ and $f_1 \in \{5, 10, 20\}$
4. $S_{2,t} = a_{21} A_1 \sin(2\pi w_{21} f_1 t)$ the seasonality with short period, where the frequency multiple w_{21} and the amplitude attenuation are random variable, $a_{21} \in \{0.5, 0.3, 0.1\}$ and $w_{21} \in \{2, 3, 5\}$
5. $\sigma_t = \sin(t) + 1.5$ the seasonal variance
6. $T_t = Bt$ the linear trend

The following collections are generated:

1. *simple-seasonality*: $X_t = S_{1,t} + (1 - A_t)R_t + A_t\zeta_t\Delta'$
2. *complex-seasonality*: $X_t = S_{1,t} + S_{2,t} + (1 - A_t)R_t + A_t\zeta_t\Delta'$

3. *variance-seasonality*: $X_t = ((1 - A_t)R_t + A_t\zeta_t\Delta')\sigma_t$
4. *trend-seasonality*: $X_t = T_t + S_t + (1 - A_t)R_t + A_t\zeta_t\Delta'$

Residual from time series with seasonality In practical applications, to simplify the detection of anomalies, seasonality and other predictable patterns are removed during a preprocessing step. To evaluate how the anomaly detector performances are affected by this preprocessing step, a new benchmark is built from the residuals extracted for each time series in the "Time series with seasonality" benchmark. In this experiment, the residual is extracted by removing the trend and the seasonality using the "seasonal_decompose" function from the Python library *statsmodels*.

Time series from real data: The anomaly detectors are evaluated on various time series datasets coming from different sources. The Numenta Anomaly Benchmark (NAB) from [99], the dodger dataset from the UCI at [85] and Mars Science Laboratory (MSL) and Soil Moisture Active Passive (SMAP) provided by NASA in [82] are used to build the complete benchmark.

4.9.5.4 Metrics

To measure the performances of different anomaly detectors, two metrics are reported: The Area Under Curve (AUC)[28, 75] in Table 4.13 and the FDR/FNR in Table 4.14. The advantage of the Area Under the Curve (AUC) is to be able to evaluate the anomaly detector without evaluating the threshold selection method. However, this can also be a limitation for real use, since a threshold is needed for practical applications. To determine the ability of anomaly detectors to control the false positive rate to a desired level while keeping the false negative rate low, the FDR and FNR metrics are reported. The disadvantage of these metrics is that it can be difficult to compare two detectors if one performs better on FDR and the other on FNR. Furthermore, they only take into account values for a single threshold, which have to be precised for detectors that return only an atypicity score. The threshold policy used for this experiment is the one implemented in Chapter 3, as stated in Section 4.9.5.2.

4.9.5.5 Results and analysis

The results are summarized in two tables. Table 4.13 represents the AUC metric according to benchmarks and anomaly detectors and Table 4.14 represents the FDR and FNR metrics.

Benchmark	BKAD(Ours)	Median	Sub. IF	DWT	Sub. LOF	LOF	VALMOD	CBLOF	FFT
Breakpoint in mean	**1.00**	0.95	0.64	0.61	0.65	0.70	0.42	0.80	0.83
Breakpoint in variance	**0.98**	0.89	0.52	0.54	0.60	0.56	0.36	0.78	0.16
Simple seasonality	0.88	**0.98**	0.56	0.57	0.71	0.68	0.47	0.73	0.72
Complex seasonality	0.94	**0.98**	0.57	0.55	0.72	0.79	0.45	0.85	0.64
Seasonality in variance	**1.00**	0.99	0.54	0.57	0.56	0.87	0.43	0.92	0.71
Seasonality and trend	0.88	**0.98**	0.53	0.57	0.71	0.63	0.47	0.67	0.72
Res. simple seasonality	**0.99**	0.98	0.62	0.57	0.69	0.92	0.47	0.99	0.89
Res. complex seasonality	**1.00**	0.99	0.63	0.56	0.71	0.94	0.45	1.00	0.91
Res. seasonality and trend	**0.99**	0.98	0.61	0.56	0.69	0.93	0.47	0.99	0.89
DODGER	0.56	0.30	**0.67**	0.65	0.54	0.51	0.41	0.48	0.30
NAB	0.57	0.45	0.66	**0.73**	0.67	0.48	0.47	0.54	0.20
NASA-MSL	0.57	0.56	**0.84**	0.81	0.61	0.56	0.48	0.68	0.56
NASA-SMAP	0.60	0.39	0.83	**0.90**	0.69	0.51	0.61	0.61	0.47

Table 4.13: AUC metric according to the anomaly detectors on benchmarks.

Benchmark	BKAD(Ours) FDR	BKAD(Ours) FNR	Median FDR	Median FNR	LOF FDR	LOF FNR	CBLOF FDR	CBLOF FNR	Sub. LOF FDR	Sub. LOF FNR	Sub. IF FDR	Sub. IF FNR	DWT FDR	DWT FNR	FFT FDR	FFT FNR
Breakpoint in mean	0.25	0.04	0.07	0.48	0.52	0.70	0.83	0.52	0.99	0.81	0.99	0.44	0.99	0.01	0.93	0.46
Breakpoint in variance	0.36	0.17	0.19	0.36	0.72	0.71	0.72	0.37	0.95	0.54	0.91	0.58	0.93	0.28	0.89	0.27
Simple seasonality	0.34	0.47	0.21	0.45	0.48	0.76	0.62	0.72	0.99	0.73	0.99	0.96	0.97	0.30	0.92	0.58
Complex seasonality	0.27	0.44	0.19	0.44	0.37	0.64	0.41	0.64	0.99	0.68	0.99	0.95	0.95	0.43	0.95	0.65
Seasonality and trend	0.50	0.46	0.22	0.45	0.58	0.86	0.77	0.78	0.99	0.77	0.79	0.89	0.97	0.11	0.92	0.57
Seasonality in variance	0.34	0.35	0.10	0.23	0.33	0.50	0.32	0.48	0.99	0.86	0.95	0.94	0.95	0.20	0.93	0.63
Res. simple seasonality	0.26	0.42	0.19	0.56	0.39	0.70	0.25	0.41	0.99	0.77	0.99	0.94	0.95	0.35	0.93	0.61
Res. complex seasonality	0.17	0.15	0.12	0.31	0.58	0.80	0.21	0.14	0.99	0.72	0.99	0.92	0.97	0.32	0.93	0.45
Res. seasonality and trend	0.26	0.43	0.20	0.57	0.44	0.73	0.29	0.41	0.99	0.79	0.97	0.94	0.95	0.36	0.92	0.64
dodger	0.41	0.66	0.79	0.99	0.89	0.09	0.97	1.00	0.71	0.92	0.39	0.91	0.70	0.55	0.89	0.09
NAB	0.61	0.91	0.62	0.85	0.67	0.58	0.50	0.82	0.64	0.74	0.55	0.62	0.77	0.27	0.87	0.25
NASA-MSL	0.78	0.91	0.62	0.84	0.71	0.72	0.45	0.72	0.62	0.82	0.49	0.70	0.65	0.42	0.68	0.49
NASA-SMAP	0.69	0.92	0.76	0.63	0.80	0.27	0.70	0.52	0.75	0.64	0.65	0.44	0.83	0.05	0.80	0.33

Table 4.14: FDR and FNR metrics according to the anomaly detectors on benchmarks.

The BKAD detector gets the highest AUC scores on series with breakpoints ("Breakpoint in mean" and "Breakpoint in variance"), as shown in Table 4.13. It can also be seen that this detector remains efficient even when the time series contain seasonality ("simple seasonality", "complex seasonality",...). This shows the benefits of splitting the time series into simpler segments based on breakpoints, even if it does not follow the model introduced in Section 4.2.1. The results show the importance of preprocessing the data. Indeed, the performance of the detector increases when it is applied to the residuals of the seasonal series instead of the original seasonal time series, as shown for "Res. simple seasonality" or "Res. complex seasonality". Nevertheless, Table 4.14 shows that it can be difficult to obtain control of the FDR and FNR even for the best AUC score. This illustrates that FDR control relies heavily on the (piecewise) iid hypothesis. Finally, BKAD is not very efficient on tested real data such as ("DODGER", "NAB", ...) containing anomalies which do not follow the formalism introduced in Section 4.2.1. The most efficient methods: "Sub. IF" and "DWT", define an atypicity score on subseries instead of data points. An interesting approach for the future might be to find a better preprocessing to apply it to real data and improve the anomaly detection.

4.10 Conclusion

In this chapter, an online anomaly detector has been developed that detects anomalies and controls the FDR at a given level α on piecewise stationary time series. The research was conducted to address three challenges:

- Changes in the reference distribution: the changes are detected using a breakpoint detector. Anomalies are retrieved in each homogeneous segment by defining an atypicity score and a calibration set.
- Uncertainty: Due to the online nature of the detection, the abnormality status of the data points is uncertain. The notion of an active set is introduced to collect the data points that need to be re-evaluated since there status is too uncertain.
- and control of the FDR: modified Benjamini-Hochberg procedure is applied to the active set to control the FDR on the entire time series.

The result of our research is a modular anomaly detector where all core components have been studied through theoretical or empirical analysis to optimize their performance. The detector has been evaluated on a variety of scenarios to understand its strengths and limitations. It demonstrates state-of-the-art capabilities to detect anomalies on time series presenting a distribution shift. The main drawback of our method is that it relies on non-robust estimation of p-values.

Also, the piecewise stationary hypothesis is often not respected in practice. Further work concerns the integration of a robust p-value estimator and the development of a preprocessing step to apply the anomaly detector to time series that are not piecewise stationary.

4.11 Figures related to experiment of Section 4.9.4

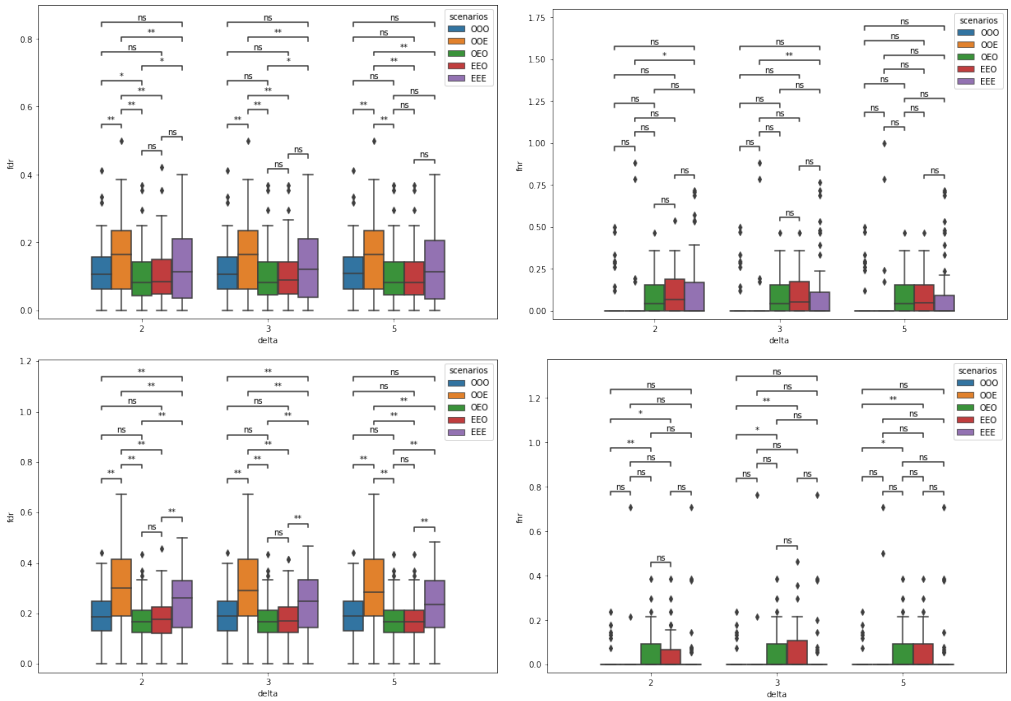


Figure 4.32: Boxplots of FDR and FNR for anomaly detection on Gaussian time series having breakpoint in the mean according to the different Detectors described in Table 4.9 and shift size Δ. Top-left: FDR while $\alpha = 0.1$, Top-right: FNR while $\alpha = 0.1$, Bottom-left: FDR while $\alpha = 0.2$, Top-right: FNR while $\alpha = 0.2$.

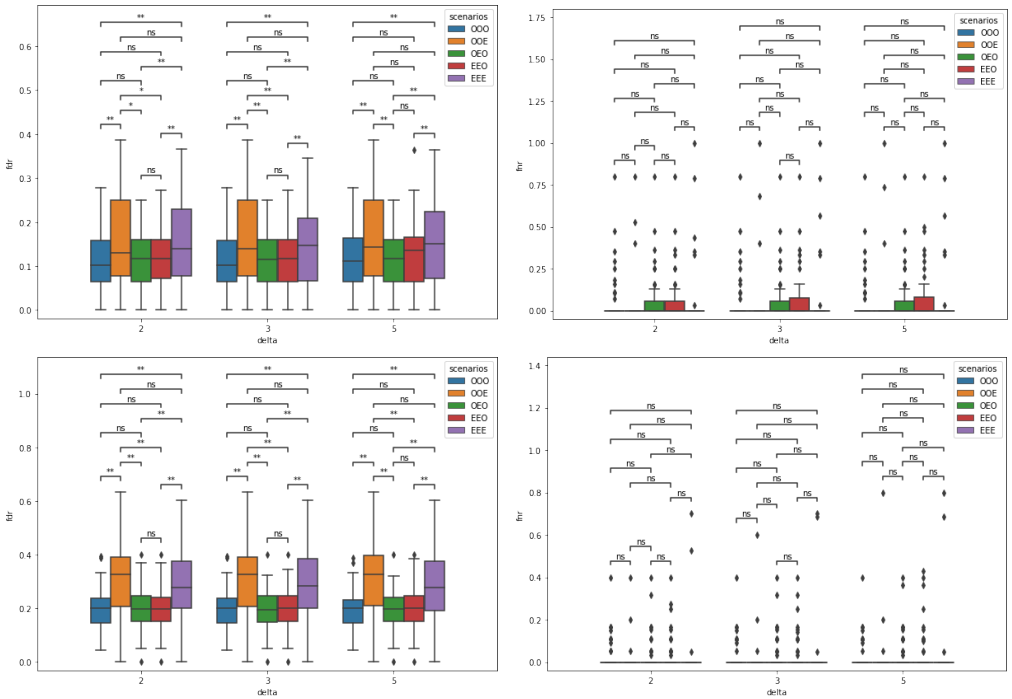


Figure 4.33: Boxplots of FDR and FNR for anomaly detection on Student time series having breakpoint in the mean according to the different Detectors described in Table 4.9 and shift size Δ. Top-left: FDR while $\alpha = 0.1$, Top-right: FNR while $\alpha = 0.1$, Bottom-left: FDR while $\alpha = 0.2$, Top-right: FNR while $\alpha = 0.2$

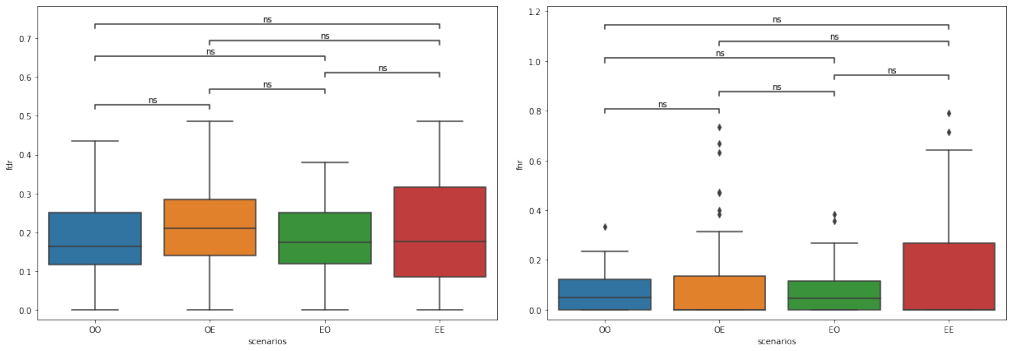


Figure 4.34: Boxplots of FDR and FNR for anomaly detection on Gaussian Mixture time series having breakpoint in the mean according to the different Detectors described in Table 4.9. Left: FDR while $\alpha = 0.2$, right: FNR while $\alpha = 0.2$.

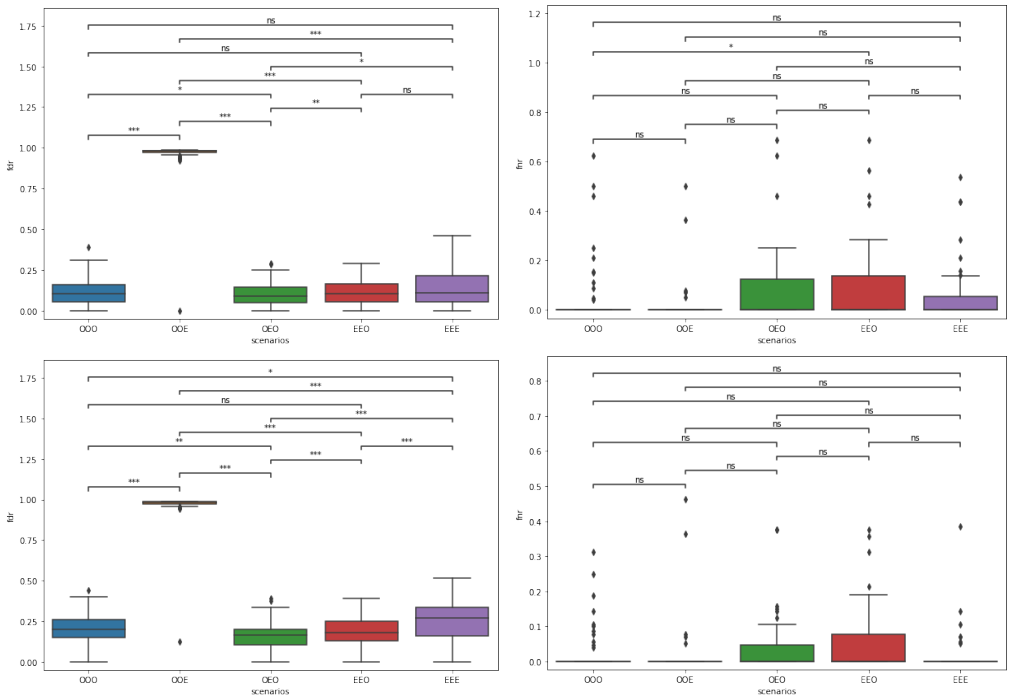


Figure 4.35: Boxplots of FDR and FNR for anomaly detection on Gaussian time series having breakpoint in the mean and variance according to the different Detectors described in Table 4.9. Top-left: FDR while $\alpha = 0.1$, Top-right: FNR while $\alpha = 0.1$, Bottom-left: FDR while $\alpha = 0.2$, Top-right: FNR while $\alpha = 0.2$

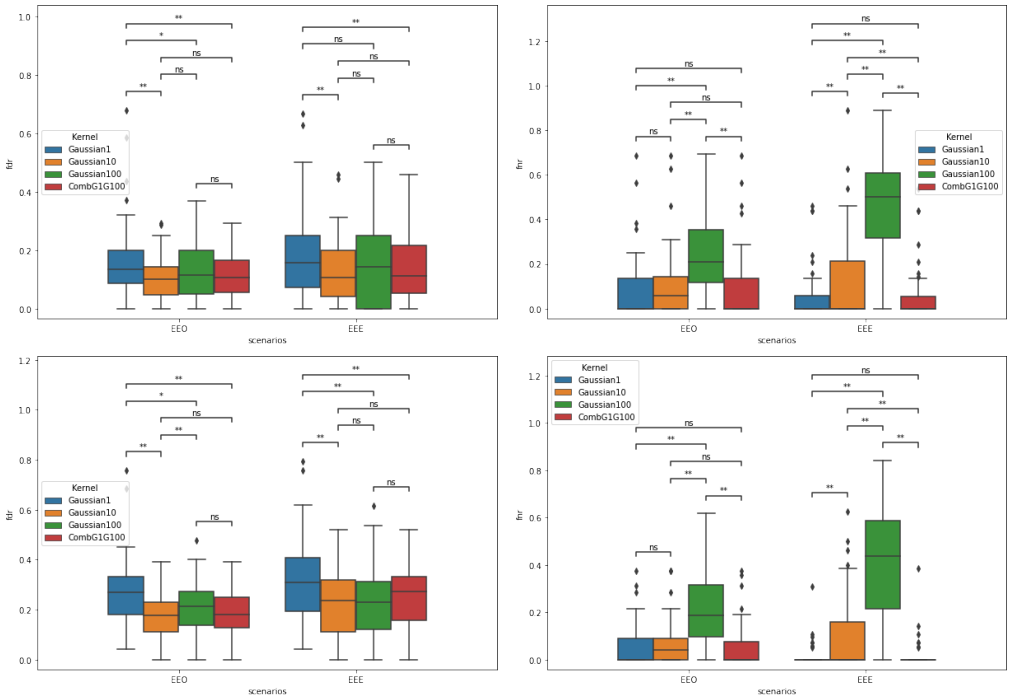


Figure 4.36: Boxplots of FDR and FNR for anomaly detection on Gaussian time series hawing breakpoint in the mean and and in the variance according to the chosen Kernel. Top-left: FDR while $\alpha = 0.1$, Top-right: FNR while $\alpha = 0.1$, Bottom-left: FDR while $\alpha = 0.2$, Top-right: FNR while $\alpha = 0.2$

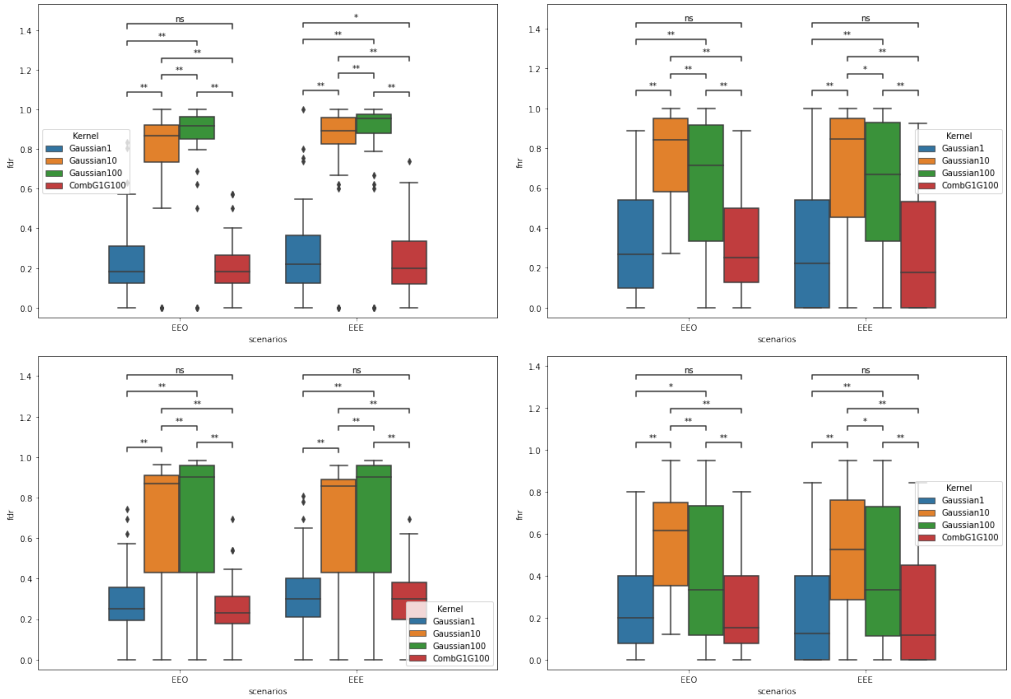
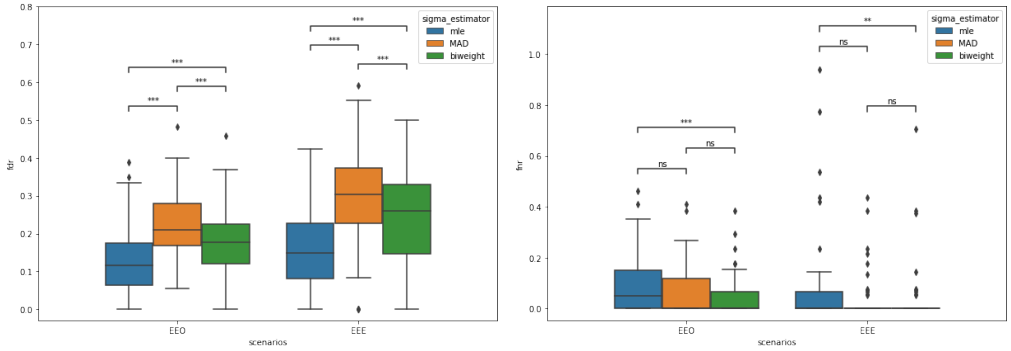


Figure 4.37: Boxplots of FDR and FNR for anomaly detection on Gaussian time series having breakpoints in the variance according to different the chosen Kernel. Top-left: FDR while $\alpha = 0.1$, Top-right: FNR while $\alpha = 0.1$, Bottom-left: FDR while $\alpha = 0.2$, Top-right: FNR while $\alpha = 0.2$



(a) FDR (b) FNR

Figure 4.38: Boxplots of the FNR and FDR according to the chosen variance estimator.

4.12 Proofs for FDR control

4.12.1 Proof of Theorem 4.1

Proof of Theorem 4.1. Similar to the proof of Theorems 3.3 and 3.4, the FDP is written as the ratio of R_1^t and FP_1^t.

$$FDP_1^\infty = \lim_{t\to\infty} FDP_1^t = \lim_{t\to\infty} \frac{FP_1^t}{R_1^t} = \lim_{t\to\infty} \frac{\sum_{u=1}^t (1-A_u) d_{u,t}}{\sum_{u=1}^t d_{u,t}} \tag{4.43}$$

In the next part of the proof the numerator and denominator are made to converge so that the mFDR expression appears. The main steps of the proof are:

1. First, it is shown that for any u, as long as t is large enough, then $s_{u,t} = \tilde{a}(X_u, i_u)$.
2. Then, it is deduced that for any u, as long as t is large enough, then $p_{u,t}$ are identically distributed.
3. Similarly, it is shown that for any u, as long as t is large enough, then $d_{u,t}$ are identically distributed.
4. Finally, since $d_{u,t}$ is identically distributed and respects a mixing property, an ergodicity theorem allows to conclude that $1/tR_1^t$ converges to $\mathbb{E}[d_{1,\infty}]$

Some notations are introduced:

- Q_s the distribution of $\tilde{a}(X_u, i_u)$.
- Q_p the distribution of the p-value $p = \hat{p}_e(s, \mathcal{S}_{cal})$, when s and all elements of $\mathcal{S}_{cal}$ follow Q_s and are independent.
- Q_d the distribution of the status $d = \mathbb{1}[p_1 < \varepsilon(p_1, \dots, p_m)]$, when all p-values $p_1, \dots, p_m$ are computed according to the same calibration set and follow Q_p.

Step 1: The scores are iid distributed for t sufficiently large.

Let $u \in [\![1, \infty]\!]$. u belongs to a unique true segment i, delimited by the breakpoints τ_i and τ_{i+1}.

When $t > u + \lambda^*$, according to (**Segmentation**) $\hat{\tau}(t) \cap [\![1, u]\!] = \tau \cap [\![1, u]\!]$, then Eq. 4.3 gives:

$$s_{u,t} = \overline{a}(X_u, \{X_{\tau_i}, \dots, X_{min(\tau_i+\ell^*, t)}\}) \tag{4.44}$$

Furthermore, when $t > u + max(\lambda^*, \ell^*) = u + m$, there are more than ℓ^* data points in the segment, then (**Score**) gives:

$$s_{u,t} = \overline{a}(X_u, \{X_{\tau_i}, \dots, X_{\tau_i+\ell^*}\}) = \tilde{a}(X_u, i) \tag{4.45}$$

The true score $\tilde{a}(X_u, i)$ is iid by assumption. For these reasons $s_{u,t}$ follows Q_s.

Since the p-value is calculated by comparing the score of a data point with the scores from a calibration set, it can be deduced that the p-values are identically distributed.

Step 2: The p-values are identically distributed for t sufficiently large.

Let u be in $[\![1, T]\!]$, according to Eq. 4.4 the p-value is estimated as the following:

$$p_{u,t} = p(s_{u,t}, \mathcal{S}_t) \tag{4.46}$$

$$\text{with } \mathcal{S}_t = \{s_{h(t-m,1),t}, \dots, s_{h(t-m,n),t}\} \tag{4.47}$$

By definition of h: $h(t-m, j) \leq t-m$, and thus according to the previous paragraph, all elements of $\mathcal{S}_t$ are identically distributed. All p-values associated with a score that follows the $\mathcal{Q}_s$ distribution follow the same distribution. In particular, all p-values $p_{u,t}$ follow the same distribution Q_p, as soon as $t > u + m$.

The status of a point depends only on the p-value of the point and the p-values used to calculate the data-driven threshold. It has been shown that the p-values follow the same distribution. In the next paragraph, it is deduced that the statuses are also identically distributed.

Step 3: The decision series $(d_{u,t})$ is identically distributed for t sufficiently large.

Let i be in $[\![1, D]\!]$ and defining a segment $[\![\tau_i, \tau_{i+1} - 1]\!]$. The cases are separated according to the position of the point in the real segment: at the end of the segment or in the middle of the segment.

Case 1: The data point belongs to the end of the segment, $u \in [\![\tau_{i+1} - m, \tau_{i+1} - 1]\!]$. Two steps are required. First it is shown that $d_{u,t}$ verifies the property for $t = u + m$, then it is shown that for any $t > u + m$, $d_{u,t} = d_{u,u+m}$.

First, let $t = \tau_{i+1} + m$. According to (**Segmentation**): $\hat{\tau}(t) \cap [\![1, \tau_{i+1}]\!] = \tau \cap [\![1, \tau_{i+1}]\!]$. Thus, the current segment at time t is equal to $[\![\tau_{i+1}, t]\!]$ and has exactly m points. Then, since the rule of closing the previous segment from Eq. 4.6 is applied, it gives

$$d_{u,t} = \mathbb{1}[\hat{p}_e(s_{u,t}, \mathcal{S}_{cal,t}) < \varepsilon(\hat{p}_e(s_{\tau_{i+1}-m,t}, \mathcal{S}_{cal,t}), \dots, \hat{p}_e(s_{\tau_{i+1}-1,t}, \mathcal{S}_{cal,t}))]$$

Since all score variables $s_{\tau_{i+1}-m,t}, \dots, s_{\tau_{i+1}-1,t}$ follow the distribution $\mathcal{Q}_s$, according to (**Score**), and all p-values are computed using the same calibration set $\mathcal{S}_{cal,t}$, then $d_{u,t}$ follows $\mathcal{Q}_d$.

Then, for $t > \tau_{i+1} + m$, $d_{u,t} = d_{u,t-1}$. This implies that the limit value $d_{u,\infty}$ is equal to $d_{u,\tau_{i+1}+m}$ which follows the distribution $\mathcal{Q}_d$.

Case 2: The data point belongs to the middle of the segment, $u \in [\![\tau_i, \tau_{i+1} - m]\!]$.

To prove that $d_{u,t}$ follows the distribution $\mathcal{Q}_d$ for all $t \geq u + m$, we have three steps. First, it is shown that the property holds for $t = u + m$. Then it is shown that the status $d_{u,t}$ may possibly be modified for t in $]\!]u + m, u + 2m]\!]$, but that $d_{u,t}$ always follows the $\mathcal{Q}_d$ distribution. Finally, it is verified that $d_{u,t}$ is constant from $t > u + 2m$.

First, let $t = u + m$. Although by hypothesis $[\![u, u + m]\!]$ contains no true breakpoints, the $\hat{\tau}_t$ detector can detect a false breakpoint. Cases are separated according to whether a breakpoint was detected or not.

- Case 2a, there is no (false) breakpoint in $[\![u, u+m]\!]$. Thus, the current segment is $[\![\tau_i, u+m]\!]$, according Eq. 4.5.

  $$d_{u,t} = \mathbb{1}[\hat{p}_e(s_{u,t}, \mathcal{S}_{cal,t}) < \varepsilon(\hat{p}_e(s_{u,t}, \mathcal{S}_{cal,t}), \dots, \hat{p}_e(s_{u+m,t}, \mathcal{S}_{cal,t}))] \tag{4.48}$$

  $d_{u,t}$ follows $\mathcal{Q}_d$

- Case 2b, there is a (false) breakpoint in $[\![u, u+m]\!]$, this breakpoint is noted $\hat{b}_t$. The current segment $[\![\hat{b}_t, u+m]\!]$ contains less than m points. Thus, according to Eq. 4.6, $d_{u,t}$ takes the value:

$$d_{u,t} = \mathbb{1}[\hat{p}_e(s_{u,t}, \mathcal{S}_{cal,t}) < \varepsilon(\hat{p}_e(s_{\hat{b}_t-m,t}, \mathcal{S}_{cal,t}), \ldots, \hat{p}_e(s_{\hat{b}_t-1,t}, \mathcal{S}_{cal,t}))] \tag{4.49}$$

  $d_{u,t}$ follows $\mathcal{Q}_d$.

When $t = u + m$, in both cases $d_{u,t}$ follows the distribution Q_d. Next, check that this property remains true even when t is greater than $u + m$.

Then, for t in $]\!]u + m, u + 2m]\!]$, the cases are split again according to the detection of a breakpoint in $[\![u, u + m]\!]$:

- Case 2a': There are no detected breakpoint in $[\![u, u + m]\!]$. According Eq. 4.7, in this case the status is not updated and $d_{u,t} = d_{u,t-1}$.

- Case 2b': A (false) breakpoint is detected in $[\![t-m, u+m]\!] \subset [\![u, u+m]\!]$ and noted $\hat{b}_t$. The current segment $[\![\hat{b}_t, u + m]\!]$ contains less than m points, $d_{u,t}$ is updated according to the rule:

$$d_{u,t} = \mathbb{1}[\hat{p}_e(s_{u,t}, \mathcal{S}_{cal,t}) < \varepsilon(\hat{p}_e(s_{\hat{b}_t-m,t}, \mathcal{S}_{cal,t}), \ldots, \hat{p}_e(s_{\hat{b}_t-1,t}, \mathcal{S}_{cal,t}))] \tag{4.50}$$

  Once again, $d_{u,t}$ follows the $\mathcal{Q}_d$ distribution.

Finally, for $t > u + 2m$, assuming (**Segmentation**), there is no breakpoint in $[\![u, u+m]\!]$ and therefore: $d_{u,t} = d_{u,t-1}$

By induction $d_{u,t}$ follows the law $\mathcal{Q}_d$ as soon as t is greater than $u + m$. Therefore the series of final decisions $d_{u,\infty}$ is identically distributed and follows the law $\mathcal{Q}_d$.

Step 4: Numerator, denominator and ratio convergence After having proved that the status series $d_{u,\infty}$ is identically distributed, the following shows that the empirical mean of the series converges to its expectation.

It was shown in the previous step that $d_{u,\infty}$ is identically distributed. Furthermore, $d_{u_1,\infty} \perp d_{u_2,\infty}$ if $|u_1 - u_2| > m + n$. Using the corollary of Theorem 3 in [22] this gives almost surely convergence:

$$\lim_{t\to\infty} \frac{1}{t} R_1^t = \lim_{t\to\infty} \frac{1}{t} \sum_{u=1}^{t} d_{u,\infty} \tag{4.51}$$

$$= \mathbb{E}[d_{1,\infty}] \tag{4.52}$$

Similarly, it gives the almost surely convergence of the numerator FP_1^t.

$$\lim_{t\to\infty} \frac{1}{t} FP_1^t = \lim_{t\to\infty} \frac{1}{t} \sum_{u=1}^{t} A_u d_{u,\infty} \tag{4.53}$$

$$= \mathbb{E}[A_1 d_{1,\infty}] \tag{4.54}$$

Since both the numerator and the denominator converge almost surely, this leads to the

almost sure convergence of the ratio which is the FDP:

$$\lim_{t\to\infty} FDP_t = \frac{\mathbb{E}[A_1 d_{1,\infty}]}{\mathbb{E}[d_{1,\infty}]} \tag{4.55}$$

Step 5: Link with mFDR So far it has been proved that the FDP of the complete series converges to the ratio of the expectation of $d_{1,\infty}$ and $A_1 d_{1,\infty}$. In the following, this ratio is linked to the mFDR computed on a subseries of size m.

This result comes from the permutation invariance of $\varepsilon(\cdot)$ which gives

$$\begin{aligned}\mathbb{E}\left[\sum_{u=1}^{m} \mathbb{1}[\hat{p}_{u,t} < \varepsilon(\hat{p}_{1,t},\dots,\hat{p}_{1,t})]\right] &= \sum_{u=1}^{m} \mathbb{E}\left[\mathbb{1}[\hat{p}_{u,t} < \varepsilon(\hat{p}_{1,t},\dots,\hat{p}_{1,t})]\right] \\ &= m\mathbb{E}\left[\mathbb{1}[\hat{p}_{1,t} < \varepsilon(\hat{p}_{1,t},\dots,\hat{p}_{1,t})]\right] \\ &= m\mathbb{E}\left[d_{1,t}\right]\end{aligned}$$

This implies that the FDP limit can be written as mFDR:

$$\lim_{t\to\infty} FDP_t = \frac{\mathbb{E}[A_1 d_{1,\infty}] \times m}{\mathbb{E}[d_{1,\infty}] \times m} = \frac{\mathbb{E}[FP_1^m]}{\mathbb{E}[R_1^m]} = mFDR_1^m \tag{4.56}$$

□

4.12.2 Proof of Corollary 4.1

Proof of Corollary 4.1. According to Theorem 4.1, the FDP of the complete time series is equal to the mFDR of the subseries:

$$\lim_{t\to\infty} FDP_1^t = mFDR_1^m$$

The following steps of the proof show that $mFDR_1^m = (1-\pi)\alpha$ using various results of Corollary 3.5 proof.

First, according to Proposition 3.3, using BH the $mFDR_1^m$ of the subseries can be expressed using the number of rejections and the FDR on the subseries.

$$mFDR_1^m = \frac{\mathbb{E}[R_1^m(1)]}{\mathbb{E}[R_1^m]} FDR_1^m \tag{4.57}$$

From the assumptions expressed in Eq. 4.9, it follows that

$$\frac{\mathbb{E}[R_1^m(1)]}{\mathbb{E}[R_1^m]} = 1 + \frac{1-\alpha}{m\pi}$$

As described in Definition 4.2, all p-values are calculated from a single calibration set, furthermore, the cardinality of the calibration set is equal to $n = \ell m/\alpha' - 1$. Then according to Theorem

3.4 in [110] and Corollary 3 in [95]:

$$FDR_1^m = (1-\pi)\alpha' = \left(1+\frac{1-\alpha}{m\pi}\right)^{-1}(1-\pi)\alpha$$

Injecting the value of FDR_1^m and $\frac{\mathbb{E}[R_1^m(1)]}{\mathbb{E}[R_1^m]}$ in Eq. 4.57, it gives:

$$mFDR_1^m = \left(1+\frac{1-\alpha}{m\pi}\right)\left(1+\frac{1-\alpha}{m\pi}\right)^{-1}(1-\pi)\alpha \tag{4.58}$$

$$= (1-\pi)\alpha \tag{4.59}$$

This result completes the proof. □

4.13 Proofs for uncertainty control

4.13.1 Proof of Proposition 4.1

Proof of Proposition 4.1. By assumption the probability $\mathbb{P}\left[W_{t-\lambda,t}\right]$ does not depend on t and is noted $f_\tau(\lambda)$. According to the second assumption, $f_\tau(\lambda)$ converges to 0 when λ tends to $+\infty$. Therefore, by definition of convergence:

$$\forall \eta > 0, \exists \lambda_\eta > 0 \quad \lambda \geq, \forall \lambda_\eta, \quad f_\tau(\lambda) \leq \eta.$$

Moreover, by definition $\lambda = t - u$, it follows that:

$$\forall \eta > 0, \quad \exists \lambda > 0, \quad \forall t \in [\![1, T]\!], \forall u \in [\![1, t]\!], \quad |u-t| \geq \lambda_\eta, \quad \mathbb{P}\left[W_{u,t}\right] \leq \eta.$$

The second result is proven using similar arguments. □

4.13.2 Proof of Theorem 4.2

Proof of Theorem 4.2. The two statements are proven separately. First, it is shown that for every η, the probability that the final status differs from the oracle status is less than 3η. Then, a mixing property allows to prove that the proportion of differences along the time series is less than 3η.

Proof of the first statement: To prove the first statement, two steps are taken: first, the final decision about the status of X_u is characterized. According to the way BKAD works, as described in Definition 4.4, there are two possibilities: either the final decision is taken when u belongs to the current segment and is not updated thereafter, as stated in Eq. 4.7. Or the status of X_u is updated when the segment to which u belongs is closed, as stated in Eq. 4.6. Once the final status has been characterized, the probability that it differs from the oracle status is calculated.

Let u be in $[\![1, T]\!]$.

In case there is t such that: $|b_t - t| < m$ and $u \in [\![\hat{b}_t - m, \hat{b}_t]\!]$. Let t' be the largest one. This corresponds to the case where the status of X_u is updated after detecting a breakpoint which

closes the segment to which u is assigned. According to Eq. 4.6

$$d_{u,t'} = \mathbb{1}[\hat{p}_{u,t'} < \varepsilon(\hat{p}_{u,b_{t'}-m}, \ldots, p_{u,b_{t'}})] \tag{4.60}$$

Otherwise, let $t' = u+m$. This corresponds to the case where the final status associated with X_u is taken at the time it belongs to the current segment. According to Eq. 4.7

$$d_{u,t'} = \mathbb{1}[\hat{p}_{u,t'} < \varepsilon(\hat{p}_{u,u}, \ldots, p_{u,t+u})] \tag{4.61}$$

By definition of t', $d_{u,t'}$ is the final status associated with X_u. Therefore, it is of interest to know the probability, noted $\mathbb{P}(\overline{V}_{u,t'})$, that $d_{u,t'}$ is equal to the oracle status.

According to the law of total probabilities on the event that the assigned segment change, noted $W_{\overline{u},t'}$ and defined in Eq. 4.13. $\overline{u} = b_{t'}$

$$\mathbb{P}(\overline{V}_{u,t'}) = \mathbb{P}(\overline{V}_{u,t'}|W_{\overline{u},t'})\mathbb{P}(W_{\overline{u},t'}) + \mathbb{P}(\overline{V}_{u,t'}|\overline{W}_{\overline{u},t'})\mathbb{P}(\overline{W}_{\overline{u},t'}) \tag{4.62}$$

$$\leq \mathbb{P}(\overline{V}_{u,t'}|W_{\overline{u},t'}) + \mathbb{P}(\overline{W}_{\overline{u},t'}) \tag{4.63}$$

Now let's upper bound the different terms on the right-hand side of Eq. 4.63. According the definition of $\ell_\eta \leq m$ in in Proposition 4.1:

$$\mathbb{P}(\overline{V}_{u,t'}|W_{\overline{u},t'}) \leq \eta \tag{4.64}$$

Then the probability of $X_{\overline{u}}$ changing its assigned segment at $[\![t', \infty[\![$ is written by distinguishing the time when this change occurs at $[\![t', t'+2m]\!]$ or at $]\!]t'+2m, \infty[\![$

$$\mathbb{P}(\overline{W}_{\overline{u},t'}) = \mathbb{P}(\exists t'' > t', \tau(t'') \cap [\![\tau_i(t'), \tau_{i+1}(t')[\![\neq \emptyset) \tag{4.65}$$

$$= \mathbb{P}(\exists t' + 2m \geq t'' > t', \tau_{t''} \cap [\![\tau_i(t'), \tau_{i+1}(t')[\![\neq \emptyset) \tag{4.66}$$

$$+ \mathbb{P}(\exists t'' > t' + 2m, \tau_{t''} \cap [\![\tau_i(t'), \tau_{i+1}(t')[\![\neq \emptyset) \tag{4.67}$$

According to Eq. 4.16 it gives:

$$\mathbb{P}(\exists t' + 2m \geq t'' > t', \tau_{t''} \cap [\![\tau_i(t'), \tau_{i+1}(t')[\![\neq \emptyset) \leq \eta \tag{4.68}$$

According to the definition on $\lambda_\eta \leq m$ in Proposition 4.1 and the assumption described by Eq. 4.17.

$$\mathbb{P}(\exists t'' > t' + 2m, \tau_{t''} \cap [\![\tau_i(t'), \tau_{i+1}(t')[\![\neq \emptyset) \leq \eta \tag{4.69}$$

Finally, by injecting the bounding terms in Eq. 4.63 using the results from Eq. 4.64, 4.68 and 4.69, it gives:

$$\mathbb{P}(\overline{V}_{u,t'}) \leq 3\eta \tag{4.70}$$

Proof of the second statement: According to the first part of the proof, the probability that the final status is different from the oracle status is less than 3η. But this property is local, valid for each u. The aim is to have a global property. For this, a property of ergodicity of $V_{u,t}$ is to be proved. According to [22], it suffices to show that there exists a number q such that if

$|t_1 - t_2| > q$ then V_{u,t_1} is independent of V_{u,t_2}.

According to Eq. 4.19, breakpoint positions are independent beyond a distance q_1. According to the ideal BKAD operation, for a given segmentation, the statuses d_{u,t_1} and d_{u,t_2} are independent if $|t_1 - t_2| > m + n$. It follows that the random variables in the series $V_{u,t}$ are independent if $|u_1 - u_2| > m + n + q_1$. This completes the proof. □

Chapter 5

Conclusion and perspectives

5.1 Conclusion

In the present thesis, different approaches have been studied in order to contribute to the improvement of anomaly detectors. The main contribution is an anomaly detector able to adapt to changes in the reference behavior and to theoretically control the FDR.

In Chapter 2, an attempt is made to improve the estimation of the p-value, which plays a key role in anomaly detection. The idea is to suggest a new procedure for selecting the bandwidth parameter for KDE. It is argued that minimization of the p-value estimation error is the most relevant selection criterion. Different approaches to estimating this criterion are explored, with the LOO approach giving the best results. Then, different estimators of the p-value are compared. It turns out that, because of the poor performance of the LOO estimator, the proposed procedure does not improve the estimation of the p-value.

In Chapter 3, a data-driven threshold selection procedure is proposed that allows FDR control on a complete iid time series using mFDR control on subseries. First, the behavior of BH when p-values are estimated is studied both theoretically and experimentally. This analysis highlights the importance of the cardinality of the calibration set in BH's performance, and suggests an optimal way to choose this cardinality. Second, it is proved theoretically that the FDR of a whole series can be controlled using a control of the mFDR on subseries. By showing that, under certain conditions, the BH procedure can be used to control the mFDR, a procedure for controlling the global FDR is obtained. This procedure is studied empirically, showing both its practical utility and its limitations, especially due to the non-robustness of the p-value estimator.

Finally, Chapter 4 presents a new anomaly detector which uses a breakpoint detector to identify instants where the reference distribution of the time series changes. This new detector is studied theoretically to show that the procedure developed in Chapter 3 allows FDR control even in this new context. The different components are studied separately to improve the performance of the anomaly detector. The anomaly detector is then extensively tested empirically to assess its capabilities and limitations. The detector's versatility is demonstrated, as it can be adapted to a wide range of time series. It also shows that the limit for controlling the FDR on real data is that the piecewise iid working hypothesis is rarely verified.

5.2 Perspectives

There are several directions that can be explored to extend this work:

- As seen in Section 4.9.4, the non-robustness of the p-value estimator is one of the main limitations of the anomaly detector. To our knowledge, there is no agnostic and robust p-value estimator that ensures control of the FDR using BH. A simple method to build a robust p-value estimator is to use the Median-of-Means. The calibration set can be divided into different blocks, and an estimation of the p-value is computed for each block. The different estimations are then merged by calculating their median. This estimator is easier to study than the one based on KDE [81]. The theoretical question is whether such an estimator can be used with BH to control the FDR at the desired level.
- Theorem 3.4 shows that the FDR of the entire time series can be controlled to a desired level α if the mFDR of the subseries of size m can be controlled to the same level. It was later shown that the BH procedure could be modified to control this mFDR. However, this procedure, described in Definition 3.6, uses the proportion of true anomalies π. This proportion cannot always be given by an expert, and its estimation is prone to error. It would be interesting to develop a procedure that controls the mFDR without using the proportion of π anomalies. The article [1] presents such an online procedure controlling the mFDR. Unfortunately, the α investing procedures are not permutation invariant, so it is not possible to use it with Theorem 3.4.
- In this work, only homogeneous segments with identical distribution have been considered. In practice, this property is difficult to verify. To extend the application of the anomaly detector, segments can be allowed to exhibit more complex behavior. For example, a segment can follow a linear trend or, more generally, a polynomial trend. Breakpoints can be detected by applying KCP to the finite difference of the series, or by using trend filtering methods [72]. The atypicity of a data point can be defined using a nonconformity measure on the finite difference time series, or by building a polynomial model on each detected segment.
- Another interesting question is the detection of collective anomalies. In our theoretical analysis and in our experiments, the anomaly labels are generated according to iid Bernoulli distributions. In practice, however, anomalies often occur in sequences. Several questions arise when anomalies occur in sequence. First, if there is a subset of abnormal points within a homogeneous segment, comparing the atypicity score to a threshold is not optimal. In fact, it's better to calculate an average of the scores (EWMA) [136] or to cumulate the scores (CUSUM) [66] to increase the power of the test. This raises the question of FDR control in this case [104]. Moreover, this assumes that the anomalies are in the middle of a normal data segment, but KCP will tend to isolate them in a homogeneous segment of abnormal data. The anomaly detector described in Chapter 4 would consider this segment as the new reference, and the anomalies would not be detected. Two approaches are possible for dealing with collective anomalies: if the anomaly subsequence is known to be short, KCP can be instructed to return only segments long enough to ensure that the anomalies are within a larger segment. Otherwise, it is possible to build an abnormal segment detector [167]. Features such as segment length or shift size can be used.

Bibliography

[1] Ehud Aharoni and Saharon Rosset. "Generalized α-investing: definitions, optimality results and application to public databases". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 76.4 (2014), pp. 771–794.

[2] Mohiuddin Ahmed, Abdun Naser Mahmood, and Jiankun Hu. "A survey of network anomaly detection techniques". In: *Journal of Network and Computer Applications* 60 (2016), pp. 19–31.

[3] Mohammed Alenezi and Martin J Reed. "Denial of service detection through TCP congestion window analysis". In: *World Congress on Internet Security (WorldCIS-2013)*. IEEE. 2013, pp. 145–150.

[4] Samaneh Aminikhanghahi and Diane J Cook. "A survey of methods for time series change point detection". In: *Knowledge and information systems* 51.2 (2017), pp. 339–367.

[5] Anastasios N Angelopoulos and Stephen Bates. "A gentle introduction to conformal prediction and distribution-free uncertainty quantification". In: *arXiv preprint arXiv:2107.07511* (2021).

[6] Sylvain Arlot, Alain Celisse, and Zaid Harchaoui. "A kernel multiple change-point algorithm via model selection". In: *Journal of machine learning research* 20.162 (2019).

[7] Alexander Aue and Claudia Kirch. "The state of cumulative sum sequential changepoint testing 70 years after Page". In: *Biometrika* 111.2 (2024), pp. 367–391.

[8] Alexander Aue et al. "Break detection in the covariance structure of multivariate time series models". In: (2009).

[9] Jushan Bai. "Estimating multiple breaks one at a time". In: *Econometric theory* 13.3 (1997), pp. 315–352.

[10] Sabyasachi Basu and Martin Meckesheimer. "Automatic outlier detection for time series: an application to sensor data". In: *Knowledge and Information Systems* 11 (2007), pp. 137–154.

[11] Stephen Bates et al. "Testing for outliers with conformal p-values". In: *The Annals of Statistics* 51.1 (2023), pp. 149–178.

[12] Jean-Patrick Baudry, Cathy Maugis, and Bertrand Michel. "Slope heuristics: overview and implementation". In: *Statistics and Computing* 22 (2012), pp. 455–470.

[13] Yoav Benjamini and Yosef Hochberg. "Controlling the false discovery rate: a practical and powerful approach to multiple testing". In: *Journal of the Royal statistical society: series B (Methodological)* 57.1 (1995), pp. 289–300.

[14] Yoav Benjamini and Daniel Yekutieli. “The control of the false discovery rate in multiple testing under dependency”. In: *Annals of statistics* (2001), pp. 1165–1188.

[15] Anil Bhattacharyya. “On a measure of divergence between two statistical populations defined by their probability distribution”. In: *Bulletin of the Calcutta Mathematical Society* 35 (1943), pp. 99–110.

[16] Lucien Birgé and Pascal Massart. “Minimal penalties for Gaussian model selection”. In: *Probability theory and related fields* 138 (2007), pp. 33–73.

[17] Lucien Birgé and Pascal Massart. “Minimum contrast estimators on sieves: exponential bounds and rates of convergence”. In: *Bernoulli* (1998), pp. 329–375.

[18] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Vol. 4. 4. Springer, 2006.

[19] Gilles Blanchard and Etienne Roquain. “Two simple sufficient conditions for FDR control”. In: (2008).

[20] Ane Blázquez-García et al. “A review on outlier/anomaly detection in time series data”. In: *ACM Computing Surveys (CSUR)* 54.3 (2021), pp. 1–33.

[21] James M Blum and Kevin K Tremper. “Alarms in the intensive care unit: too much of a good thing is dangerous: is it time to add some intelligence to alarms?” In: *Critical care medicine* 38.2 (2010), pp. 702–703.

[22] JR Blum, David Lee Hanson, and Lambert Herman Koopmans. *On the strong law of large numbers for a class of stochastic processes*. Sandia Corporation, 1963.

[23] Benjamin Bobbia, Paul Doukhan, and Xiequan Fan. “A Review on some weak dependence conditions”. In: *HAL, preprint* (2022).

[24] Tim Bollerslev. “Generalized autoregressive conditional heteroskedasticity”. In: *Journal of econometrics* 31.3 (1986), pp. 307–327.

[25] Rishi Bommasani et al. “On the opportunities and risks of foundation models”. In: *arXiv preprint arXiv:2108.07258* (2021).

[26] Adrian W Bowman. “An alternative method of cross-validation for the smoothing of density estimates”. In: *Biometrika* 71.2 (1984), pp. 353–360.

[27] George EP Box et al. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[28] Andrew P Bradley. “The use of the area under the ROC curve in the evaluation of machine learning algorithms”. In: *Pattern recognition* 30.7 (1997), pp. 1145–1159.

[29] Mohammad Braei and Sebastian Wagner. “Anomaly detection in univariate time-series: A survey on the state-of-the-art”. In: *arXiv preprint arXiv:2004.00433* (2020).

[30] Markus M Breunig et al. “LOF: identifying density-based local outliers”. In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 2000, pp. 93–104.

[31] Tom Brown et al. “Language models are few-shot learners”. In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[32] Teodora Sandra Buda, Bora Caglayan, and Haytham Assem. “Deepad: A generic framework based on deep learning for time series anomaly detection”. In: *Pacific-Asia conference on knowledge discovery and data mining*. Springer. 2018, pp. 577–588.

[33] Evgeny Burnaev and Vladislav Ishimtsev. "Conformalized density-and distance-based anomaly detection in time-series data". In: *arXiv preprint arXiv:1608.04585* (2016).

[34] Yang Cao et al. "Sequential change-point detection via online convex optimization". In: *Entropy* 20.2 (2018), p. 108.

[35] Kevin M Carter and William W Streilein. "Probabilistic reasoning for streaming anomaly detection". In: *2012 IEEE Statistical Signal Processing Workshop (SSP)*. IEEE. 2012, pp. 377–380.

[36] Mete Çelik, Filiz Dadaşer-Çelik, and Ahmet Şakir Dokuz. "Anomaly detection in temperature data using DBSCAN algorithm". In: *2011 international symposium on innovations in intelligent systems and applications*. IEEE. 2011, pp. 91–95.

[37] Alain Celisse and Stéphane Robin. "A cross-validation based estimation of the proportion of true null hypotheses". In: *Journal of Statistical Planning and Inference* 140.11 (2010), pp. 3132–3147.

[38] Alain Celisse et al. "New efficient algorithms for multiple change-point detection with reproducing kernels". In: *Computational Statistics & Data Analysis* 128 (2018), pp. 200–220.

[39] Varun Chandola, Arindam Banerjee, and Vipin Kumar. "Anomaly detection: A survey". In: *ACM computing surveys (CSUR)* 41.3 (2009), pp. 1–58.

[40] Yen-Chi Chen. "STAT 542: Multivariate Analysis, Lecture 7: Density Estimation". 2021.

[41] Taesung Choi et al. "Multivariate time-series anomaly detection using SeqVAE-CNN hybrid model". In: *2022 International Conference on Information Networking (ICOIN)*. IEEE. 2022, pp. 250–253.

[42] Kacper P Chwialkowski et al. "Fast two-sample testing with analytic representations of probability measures". In: *Advances in Neural Information Processing Systems* 28 (2015).

[43] Antonia Creswell et al. "Generative adversarial networks: An overview". In: *IEEE signal processing magazine* 35.1 (2018), pp. 53–65.

[44] Sándor Csörgö. "On the law of large numbers for the bootstrap mean". In: *Statistics & probability letters* 14.1 (1992), pp. 1–7.

[45] Maria Cvach. "Monitor alarm fatigue: an integrative review". In: *Biomedical instrumentation & technology* 46.4 (2012), pp. 268–277.

[46] Paul Deheuvels. "Estimation non paramétrique de la densité par histogrammes généralisés". In: *Revue de statistique appliquée* 25.3 (1977), pp. 5–42.

[47] Janez Demšar. "Statistical comparisons of classifiers over multiple data sets". In: *The Journal of Machine learning research* 7 (2006), pp. 1–30.

[48] Luc Devroye et al. "Sub-Gaussian mean estimators". In: (2016).

[49] Wei Dong et al. "Modeling LSH for performance tuning". In: *Proceedings of the 17th ACM conference on Information and knowledge management*. 2008, pp. 669–678.

[50] Robert F Engle. "Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation". In: *Econometrica: Journal of the econometric society* (1982), pp. 987–1007.

[51] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: *kdd*. Vol. 96. 34. 1996, pp. 226–231.

[52] Alexander TM Fisch, Lawrence Bardwell, and Idris A Eckley. "Real time anomaly detection and categorisation". In: *Statistics and Computing* 32.4 (2022), p. 55.

[53] RA Fisher. "The Design of Experiments, volume 6th Ed". In: *Hafner, New York, NY* (1951).

[54] Ronald Aylmer Fisher et al. *The design of experiments*. 7th Ed. Oliver, Boyd. London, and Edinburgh, 1960.

[55] Thomas Flynn and Shinjae Yoo. "Change detection with the kernel cumulative sum algorithm". In: *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE. 2019, pp. 6092–6099.

[56] Ralph Foorthuis. "On the nature and types of anomalies: a review of deviations in data". In: *International journal of data science and analytics* 12.4 (2021), pp. 297–331.

[57] Dean P Foster and Robert A Stine. "α-investing: a procedure for sequential control of expected false discoveries". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 70.2 (2008), pp. 429–444.

[58] Piotr Fryzlewicz. "Detecting possibly frequent change-points: Wild Binary Segmentation 2 and steepest-drop model selection". In: *Journal of the Korean Statistical Society* 49.4 (2020), pp. 1027–1070.

[59] Kenji Fukumizu et al. "Kernel choice and classifiability for RKHS embeddings of probability distributions". In: *Advances in neural information processing systems* 22 (2009).

[60] Axel Gandy and Georg Hahn. "MMCTest—a safe algorithm for implementing multiple Monte Carlo tests". In: *Scandinavian Journal of Statistics* 41.4 (2014), pp. 1083–1101.

[61] Damien Garreau, Wittawat Jitkrittum, and Motonobu Kanagawa. "Large sample analysis of the median heuristic". In: *arXiv preprint arXiv:1707.07269* (2017).

[62] Alexander Geiger et al. "Tadgan: Time series anomaly detection using generative adversarial networks". In: *2020 ieee international conference on big data (big data)*. IEEE. 2020, pp. 33–43.

[63] Zeineb Ghrib, Rakia Jaziri, and Rim Romdhane. "Hybrid approach for anomaly detection in time series data". In: *2020 international joint conference on neural networks (ijcnn)*. IEEE. 2020, pp. 1–7.

[64] Alexander Goldenshluger and Oleg Lepski. "Bandwidth selection in kernel density estimation: oracle inequalities and adaptive minimax optimality". In: (2011).

[65] Ian Goodfellow et al. "Generative adversarial nets". In: *Advances in neural information processing systems* 27 (2014).

[66] Pierre Granjon. "The CuSum algorithm-a small review". In: (2013).

[67] Robert M Gray and RM Gray. *Probability, random processes, and ergodic properties*. Vol. 1. Springer, 2009.

[68] Arthur Gretton et al. "A kernel two-sample test". In: *The Journal of Machine Learning Research* 13.1 (2012), pp. 723–773.

[69] AM Gross and John W Tukey. *The estimators of the Princeton robustness study*. Department of Statistics, Univ., 1973.

[70] Mickael Guedj et al. "Kerfdr: a semi-parametric kernel-based approach to local false discovery rate estimation". In: *BMC bioinformatics* 10.1 (2009), pp. 1–12.

[71] Sudipto Guha et al. “Robust random cut forest based anomaly detection on streams”. In: *International conference on machine learning*. PMLR. 2016, pp. 2712–2721.

[72] Adityanand Guntuboyina et al. “Adaptive risk bounds in univariate total variation denoising and trend filtering”. In: (2020).

[73] Wenge Guo and Shyamal Peddada. “Adaptive choice of the number of bootstrap samples in large scale multiple testing”. In: *Statistical applications in genetics and molecular biology* 7.1 (2008).

[74] Peter Hall and James Stephen Marron. “Estimation of integrated squared density derivatives”. In: *Statistics & Probability Letters* 6.2 (1987), pp. 109–115.

[75] James A Hanley and Barbara J McNeil. “The meaning and use of the area under a receiver operating characteristic (ROC) curve.” In: *Radiology* 143.1 (1982), pp. 29–36.

[76] Zengyou He, Xiaofei Xu, and Shengchun Deng. “Discovering cluster-based local outliers”. In: *Pattern recognition letters* 24.9-10 (2003), pp. 1641–1650.

[77] Nils-Bastian Heidenreich, Anja Schindler, and Stefan Sperlich. “Bandwidth selection for kernel density estimation: a review of fully automatic selectors”. In: *AStA Advances in Statistical Analysis* 97 (2013), pp. 403–433.

[78] Weiming Hu et al. “Anomaly detection using local kernel density estimation and context-based regression”. In: *IEEE Transactions on Knowledge and Data Engineering* 32.2 (2018), pp. 218–233.

[79] Shih-Ting Huang and Johannes Lederer. “DeepMoM: Robust Deep Learning With Median-of-Means”. In: *Journal of Computational and Graphical Statistics* 32.1 (2023), pp. 181–195.

[80] Peter J Huber. “Robust estimation of a location parameter”. In: *Breakthroughs in statistics: Methodology and distribution*. Springer, 1992, pp. 492–518.

[81] Pierre Humbert, Batiste Le Bars, and Ludovic Minvielle. “Robust kernel density estimation with median-of-means principle”. In: *International Conference on Machine Learning*. PMLR. 2022, pp. 9444–9465.

[82] Kyle Hundman et al. “Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding”. In: *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 2018, pp. 387–395.

[83] Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[84] Rob J Hyndman et al. “A state space framework for automatic forecasting using exponential smoothing methods”. In: *International Journal of forecasting* 18.3 (2002), pp. 439–454.

[85] Alexander Ihler, Jon Hutchins, and Padhraic Smyth. “Adaptive event detection with time-varying poisson processes”. In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2006, pp. 207–216.

[86] Vladislav Ishimtsev et al. “Conformal k-NN Anomaly Detector for Univariate Data Streams”. In: *Conformal and Probabilistic Prediction and Applications*. PMLR. 2017, pp. 213–227.

[87] Adel Javanmard and Andrea Montanari. “Online rules for control of false discovery rate and false discovery exceedance”. In: *The Annals of statistics* 46.2 (2018), pp. 526–554.

[88] Feiyu Jiang, Changbo Zhu, and Xiaofeng Shao. “Two-sample and change-point inference for non-Euclidean valued time series”. In: *Electronic Journal of Statistics* 18.1 (2024), pp. 848–894.

[89] M Chris Jones, James S Marron, and Simon J Sheather. “A brief survey of bandwidth selection for density estimation”. In: *Journal of the American statistical association* 91.433 (1996), pp. 401–407.

[90] Sevvandi Kandanaarachchi and Rob J Hyndman. “Leave-one-out kernel density estimates for outlier detection”. In: *Journal of Computational and Graphical Statistics* 31.2 (2022), pp. 586–599.

[91] Rebecca Killick, Paul Fearnhead, and Idris A Eckley. “Optimal detection of changepoints with a linear computational cost”. In: *Journal of the American Statistical Association* 107.500 (2012), pp. 1590–1598.

[92] Alexander Kirillov et al. “Segment anything”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 4015–4026.

[93] Nursel Koyuncu and Cem Kadilar. “Efficient estimators for the population mean”. In: *Hacettepe Journal of Mathematics and Statistics* 38.2 (2009), pp. 217–225.

[94] Viacheslav Kozitsin, Iurii Katser, and Dmitry Lakontsev. “Online forecasting and anomaly detection based on the ARIMA model”. In: *Applied Sciences* 11.7 (2021), p. 3194.

[95] Etienne Krönert, Alain Célisse, and Dalila Hattab. “FDR Control for Online Anomaly Detection”. In: *arXiv preprint arXiv:2312.01969* (2023).

[96] Etienne Krönert, Dalila Hattab, and Alain Celisse. “Breakpoint based online anomaly detection”. In: *arXiv preprint arXiv:2402.03565* (2024).

[97] Claire Lacour, Pascal Massart, and Vincent Rivoirard. “Estimator selection: a new method with applications to kernel density estimation”. In: *Sankhya A* 79 (2017), pp. 298–335.

[98] Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. “Numba: A llvm-based python jit compiler”. In: *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*. 2015, pp. 1–6.

[99] Alexander Lavin and Subutai Ahmad. “Evaluating real-time anomaly detection algorithms–the Numenta anomaly benchmark”. In: *2015 IEEE 14th international conference on machine learning and applications (ICMLA)*. IEEE. 2015, pp. 38–44.

[100] Rikard Laxhammar. “Conformal anomaly detection: Detecting abnormal trajectories in surveillance applications”. PhD thesis. University of Skövde, 2014.

[101] Rikard Laxhammar and Göran Falkman. “Inductive conformal anomaly detection for sequential detection of anomalous sub-trajectories”. In: *Annals of Mathematics and Artificial Intelligence* 74 (2015), pp. 67–94.

[102] Guillaume Lecué, Matthieu Lerasle, and Timlothée Mathieu. “Robust classification via MOM minimization”. In: *Machine learning* 109 (2020), pp. 1635–1665.

[103] Katarzyna Lewandowska et al. “Determining Factors of Alarm Fatigue among Nurses in Intensive Care Units—A Polish Pilot Study”. In: *Journal of Clinical Medicine* 12.9 (2023), p. 3120.

[104] Yanting Li and Fugee Tsung. “False discovery rate-adjusted charting schemes for multistage process monitoring and fault identification”. In: *Technometrics* 51.2 (2009), pp. 186–205.

[105] Yichen Li et al. “An Intelligent Framework for Timely, Accurate, and Comprehensive Cloud Incident Detection”. In: *ACM SIGOPS Operating Systems Review* 56.1 (2022), pp. 1–7.

[106] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. “Isolation forest”. In: *2008 eighth ieee international conference on data mining*. IEEE. 2008, pp. 413–422.

[107] Jian Ma. “Change Point Detection with Copula Entropy based Two-Sample Test”. In: *arXiv preprint arXiv:2403.07892* (2024).

[108] Prasanta Chandra Mahalanobis. “On the generalized distance in statistics”. In: *Sankhyā: The Indian Journal of Statistics, Series A (2008-)* 80 (2018), S1–S7.

[109] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. “The M4 Competition: 100,000 time series and 61 forecasting methods”. In: *International Journal of Forecasting* 36.1 (2020), pp. 54–74.

[110] Ariane Marandon et al. “Machine learning meets false discovery rate”. In: *arXiv preprint arXiv:2208.06685* (2022).

[111] David Mary and Etienne Roquain. “Semi-supervised multiple testing”. In: *Electronic Journal of Statistics* 16.2 (2022), pp. 4926–4981.

[112] Muhammad Mashuri et al. “PCA-based Hotelling’s T2 chart with fast minimum covariance determinant (FMCD) estimator and kernel density estimation (KDE) for network intrusion detection”. In: *Computers & Industrial Engineering* 158 (2021), p. 107447.

[113] David S Matteson and Nicholas A James. “A nonparametric approach for multiple change point analysis of multivariate data”. In: *Journal of the American Statistical Association* 109.505 (2014), pp. 334–345.

[114] Laura Mayoral. *Time Series Analysis: Introduction to time series and forecasting*. 2019.

[115] Eric Stefan Miele, Fabrizio Bonacina, and Alessandro Corsini. “Deep anomaly detection in horizontal axis wind turbines using graph convolutional autoencoders for multivariate time series”. In: *Energy and AI* 8 (2022), p. 100145.

[116] John A Miller et al. “A survey of deep learning and foundation models for time series forecasting”. In: *arXiv preprint arXiv:2401.13912* (2024).

[117] Stanislav Minsker. “Geometric median and robust estimation in Banach spaces”. In: (2015).

[118] ARIMA-GARCH Model. “Detection of network attacks using hybrid”. In: *Dependability Problems and Complex Systems: Proceedings of the Twelfth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX*. 2017, p. 1.

[119] Frederick Mosteller and John W Tukey. “Data analysis and regression. A second course in statistics”. In: *Addison-Wesley series in behavioral science: quantitative methods* (1977).

[120] Mohsin Munir et al. “DeepAnT: A deep learning approach for unsupervised anomaly detection in time series”. In: *Ieee Access* 7 (2018), pp. 1991–2005.

[121] Gerhard Münz, Sa Li, and Georg Carle. “Traffic anomaly detection using k-means clustering”. In: *Gi/itg workshop mmbnet*. Vol. 7. 9. 2007.

[122] Gyoung S Na, Donghyun Kim, and Hwanjo Yu. “Dilof: Effective and memory efficient local outlier detection in data streams”. In: *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 2018, pp. 1993–2002.

[123] Isack Thomas Nicholaus et al. “Anomaly detection of water level using deep autoencoder”. In: *Sensors* 21.19 (2021), p. 6679.

[124] Oscar Hernan Madrid Padilla et al. “Optimal nonparametric change point detection and localization”. In: *arXiv preprint arXiv:1905.10019* (2019).

[125] Oscar Hernan Madrid Padilla et al. “Optimal nonparametric multivariate change point detection and localization”. In: *IEEE Transactions on Information Theory* 68.3 (2021), pp. 1922–1944.

[126] Ewan S Page. “Continuous inspection schemes”. In: *Biometrika* 41.1/2 (1954), pp. 100–115.

[127] Eduardo HM Pena, Marcos VO de Assis, and Mario Lemes Proença. “Anomaly detection using forecasting methods arima and hwds”. In: *2013 32nd international conference of the chilean computer science society (sccc)*. IEEE. 2013, pp. 63–66.

[128] Belinda Phipson and Gordon K Smyth. “Permutation P-values should never be zero: calculating exact P-values when permutations are randomly drawn”. In: *Statistical applications in genetics and molecular biology* 9.1 (2010).

[129] James Pickands III. “Statistical inference using extreme order statistics”. In: *the Annals of Statistics* (1975), pp. 119–131.

[130] Bechir Raggad. “Fondements de la théorie des valeurs extrêmes, ses principales applications et son apport à la gestion des risques du marché pétrolier”. In: *Mathématiques et sciences humaines. Mathematics and social sciences* 186 (2009), pp. 29–63.

[131] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. “Efficient algorithms for mining outliers from large data sets”. In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 2000, pp. 427–438.

[132] Aaditya Ramdas et al. “Online control of the false discovery rate with decaying memory”. In: *Advances in neural information processing systems* 30 (2017).

[133] Aaditya Ramdas et al. “SAFFRON: an adaptive algorithm for online control of the false discovery rate”. In: *International conference on machine learning*. PMLR. 2018, pp. 4286–4294.

[134] Faraz Rasheed et al. “Fourier transform based spatial outlier mining”. In: *Intelligent Data Engineering and Automated Learning-IDEAL 2009: 10th International Conference, Burgos, Spain, September 23-26, 2009. Proceedings 10*. Springer. 2009, pp. 317–324.

[135] Quentin Rebjock et al. “Online false discovery rate control for anomaly detection in time series”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 26487–26498.

[136] SW Roberts. “Control chart tests based on geometric moving averages”. In: *Technometrics* 42.1 (2000), pp. 97–101.

[137] David S Robertson, James MS Wason, and Aaditya Ramdas. “Online multiple hypothesis testing”. In: *Statistical science: a review journal of the Institute of Mathematical Statistics* 38.4 (2023), p. 557.

[138] Peter J Rousseeuw and Mia Hubert. “Anomaly detection by robust statistics”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.2 (2018), e1236.

[139] Mats Rudemo. “Empirical choice of histograms and kernel density estimators”. In: *Scandinavian Journal of Statistics* (1982), pp. 65–78.

[140] Lukas Ruff et al. “A unifying review of deep and shallow anomaly detection”. In: *Proceedings of the IEEE* 109.5 (2021), pp. 756–795.

[141] Aleksandr Maratovich Safin and Evgeny Burnaev. "Conformal kernel expected similarity for anomaly detection in time-series data". In: *Advances in Systems Science and Applications* 17.3 (2017), pp. 22–33.

[142] Mahmoud Said Elsayed et al. "Network anomaly detection using LSTM based autoencoder". In: *Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks*. 2020, pp. 37–45.

[143] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. "Anomaly detection in time series: a comprehensive evaluation". In: *Proceedings of the VLDB Endowment* 15.9 (2022), pp. 1779–1797.

[144] David W Scott and George R Terrell. "Biased and unbiased cross-validation in density estimation". In: *Journal of the american Statistical association* 82.400 (1987), pp. 1131–1146.

[145] Glenn Shafer and Vladimir Vovk. "A Tutorial on Conformal Prediction." In: *Journal of Machine Learning Research* 9.3 (2008).

[146] Simon J Sheather and Michael C Jones. "A reliable data-based bandwidth selection method for kernel density estimation". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 53.3 (1991), pp. 683–690.

[147] Lewis H Shoemaker and Thomas P Hettmansperger. "Robust estimates and tests for the one-and two-sample scale models". In: *Biometrika* 69.1 (1982), pp. 47–53.

[148] David Siegmund and ES Venkatraman. "Using the generalized likelihood ratio statistic for sequential detection of a change-point". In: *The Annals of Statistics* (1995), pp. 255–271.

[149] Alban Siffer et al. "Anomaly detection in streams with extreme value theory". In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 2017, pp. 1067–1075.

[150] Bernard W Silverman. *Density estimation for statistics and data analysis*. Routledge, 1998.

[151] James Smith et al. "Anomaly detection of trajectories with kernel density estimation by conformal prediction". In: *Artificial Intelligence Applications and Innovations: AIAI 2014 Workshops: CoPA, MHDW, IIVC, and MT4BD, Rhodes, Greece, September 19-21, 2014. Proceedings 10*. Springer. 2014, pp. 271–280.

[152] Slawek Smyl. "A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting". In: *International Journal of Forecasting* 36.1 (2020), pp. 75–85.

[153] Jo M Solet and Paul R Barach. "Managing alarm fatigue in cardiac care". In: *Progress in Pediatric Cardiology* 33.1 (2012), pp. 85–90.

[154] Siwoon Son, Myeong-Seon Gil, and Yang-Sae Moon. "Anomaly detection for big log data using a Hadoop ecosystem". In: *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*. IEEE. 2017, pp. 377–380.

[155] Guillaume Staerman. "Functional anomaly detection and robust estimation". PhD thesis. Institut polytechnique de Paris, 2022.

[156] Robert G Staudte and Simon J Sheather. *Robust estimation and testing*. John Wiley & Sons, 2011.

[157] Douglas Steinley. "K-means clustering: a half-century synthesis". In: *British Journal of Mathematical and Statistical Psychology* 59.1 (2006), pp. 1–34.

[158] John D Storey. "A direct approach to false discovery rates". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 64.3 (2002), pp. 479–498.

[159] John D Storey, Jonathan E Taylor, and David Siegmund. "Strong control, conservative point estimation and simultaneous conservative consistency of false discovery rates: a unified approach". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 66.1 (2004), pp. 187–205.

[160] Wenguang Sun and T Tony Cai. "Oracle and adaptive compound decision rules for false discovery rate control". In: *Journal of the American Statistical Association* 102.479 (2007), pp. 901–912.

[161] Charles C Taylor. "Bootstrap choice of the smoothing parameter in kernel density estimation". In: *Biometrika* 76.4 (1989), pp. 705–712.

[162] Markus Thill, Wolfgang Konen, and Thomas Bäck. "Time series anomaly detection with discrete wavelet transforms and maximum likelihood estimation". In: *Intern. Conference on Time Series (ITISE)*. Vol. 2. 2017, pp. 11–23.

[163] Charles Truong, Laurent Oudre, and Nicolas Vayatis. "Selective review of offline change point detection methods". In: *Signal Processing* 167 (2020), p. 107299.

[164] Alexandre B. Tsybakov. "Nonparametric estimators". In: *Introduction to Nonparametric Estimation*. New York, NY: Springer New York, 2009, pp. 1–76. ISBN: 978-0-387-79052-7. DOI: 10.1007/978-0-387-79052-7_1. URL: https://doi.org/10.1007/978-0-387-79052-7_1.

[165] Shreshth Tuli, Giuliano Casale, and Nicholas R Jennings. "Tranad: Deep transformer networks for anomaly detection in multivariate time series data". In: *arXiv preprint arXiv:2201.07284* (2022).

[166] Gabriel Wallin, Jenny Häggström, and Marie Wiberg. "How Important is the Choice of Bandwidth in Kernel Equating?" In: *Applied psychological measurement* 45.7-8 (2021), pp. 518–535.

[167] Ming Wan et al. "Functional Pattern-Related Anomaly Detection Approach Collaborating Binary Segmentation with Finite State Machine". In: *CMC-COMPUTERS MATERIALS & CONTINUA* 77.3 (2023), pp. 3573–3592.

[168] Daren Wang, Yi Yu, and Alessandro Rinaldo. "Univariate mean change point detection: Penalization, cusum and optimality". In: (2020).

[169] Weinan Wang et al. "Online fdr controlled anomaly detection for streaming time series". In: *5th Workshop on Mining and Learning from Time Series (MiLeTS)*. 2019.

[170] Stanisław Węglarczyk. "Kernel density estimation and its application". In: *ITM web of conferences*. Vol. 23. EDP Sciences. 2018, p. 00037.

[171] Asaf Weinstein, Rina Barber, and Emmanuel Candes. "A power and prediction analysis for knockoffs with lasso statistics". In: *arXiv preprint arXiv:1712.06465* (2017).

[172] Ziyu Xu and Aaditya Ramdas. "Dynamic algorithms for online multiple testing". In: *Mathematical and Scientific Machine Learning*. PMLR. 2022, pp. 955–986.

[173] Nong Ye, Qiang Chen, and Connie M Borror. "EWMA forecast of normal system activity for computer intrusion detection". In: *IEEE Transactions on Reliability* 53.4 (2004), pp. 557–566.

[174] Susik Yoon et al. "Adaptive model pooling for online deep anomaly detection from a complex evolving data stream". In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2022, pp. 2347–2357.

[175] Martin Zhang, James Zou, and David Tse. "Adaptive monte carlo multiple testing via multi-armed bandits". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 7512–7522.

[176] Chong Zhou and Randy C Paffenroth. "Anomaly detection with robust deep autoencoders". In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 2017, pp. 665–674.

[177] Tian Zhou et al. "One fits all: Power general time series analysis by pretrained lm". In: *Advances in neural information processing systems* 36 (2024).

[178] Zeng-Guang Zhou and Ping Tang. "Continuous anomaly detection in satellite image time series based on z-scores of season-trend model residuals". In: *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE. 2016, pp. 3410–3413.

[179] Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. "A survey on unsupervised outlier detection in high-dimensional numerical data". In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* 5.5 (2012), pp. 363–387.