UiO : University of Oslo

Martin Tveten

Scalable change and anomaly detection in cross-correlated data

Thesis submitted for the degree of Philosophiae Doctor

Department of Mathematics
Faculty of Mathematics and Natural Sciences

2021

Series of dissertations submitted to the
Faculty of Mathematics and Natural Sciences, University of Oslo
No. 1234

ISSN 1234-5678

Cover: Hanne Baadsgaard Utigard.
Print production: Reprosentralen, University of Oslo.

Preface

This doctoral project started in August 2017 as a continuation of my master's thesis. Both projects have been supervised mainly by Ingrid Glad as part of Big Insight, a Centre for Research-Based Innovation funded by the Norwegian Research Council (project 237718). This thesis is the final result, consisting of four articles encapsulated by an introduction to provide context and background.

Being a PhD student has been an invaluable and enjoyable experience, although at times frustrating. Initially, I was supposed to get a head start by turning parts of my master's thesis into a paper during the first semester. Almost two years later, one paper had grown into two (Paper I and Paper II), and I never seemed to be satisfied with them. At this point, I was invited to join an applied Big Insight project in collaboration with the Norwegian Computing Centre and ABB on monitoring the temperature of ship motors (Paper III). Simultaneously, Ingrid got me in contact with Idris Eckley at Lancaster University. He and Paul Fearnhead were kind enough to welcome me to Lancaster as part of the Statscale program during the autumn of 2019 for a highly productive, inspiring and slightly rainy three months, where they initiated and supervised me on Paper IV. In the end, I feel like all the work has come nicely together into the common themes of changes, anomalies, scalable computation and cross-correlation.

The root cause of this thesis is Ingrid, who sparked my interest in statistics and almost five years ago introduced me to changepoint models. She has been exceedingly supportive throughout and is always enthusiastic, positive and solution-oriented. I am grateful to my co-supervisor Nils Lid Hjort's keen eyes for details and for letting me into his office to answer questions whenever needed. Thanks to Idris and Paul for being open to collaboration, encouraging and interested, as well as for the regular supervision on Skype. I also thank Kristoffer Hellton and Morten Stakkeland for letting me play around with the ship sensor data, Solveig Engebretsen for running my code countless times at any time of the day, and the remaining co-authors Ola Haug and Magne Aldrin. It was great fun working with Jonas Moss on the kdensity R package, implementing the 25 year old doctoral work of Ingrid (supervised by Nils). The past three years have been vastly enriched—both socially and academically—by my fellow students and colleagues in the statistics group at the Mathematics Department in Oslo, Big Insight, and the Statscale room in Lancaster. For teaching me basic C++, I owe Daniel Grose a skiing lesson. Finally, I am grateful to my friends for helping me recharge my batteries over weekends, to Trude for enduring me during thesis work and lock-down, and to my family for always supporting what I do.

Martin Tveten
Blindern, January 2021

List of Papers

Paper I

Tveten, M. (2019). Which principal components are most sensitive in the change detection problem? *Stat*, 8(e252).

Paper II

Tveten, M. and Glad, I. K. (2019). Online detection of sparse changes in high-dimensional data streams using tailored projections. *Manuscript*.

Paper III

Hellton, K. H., Tveten, M., Stakkeland, M., Engebretsen, S., Haug, O. and Aldrin, M. (2020). Real-time prediction of propulsion motor overheating using machine learning. *Submitted for publication*.

Paper IV

Tveten, M., Eckley, I. A. and Fearnhead, P. (2020). Scalable changepoint and anomaly detection in cross-correlated data with an application to condition monitoring. *Invited to submit a revision to Annals of Applied Statistics.*

Additional paper

The following paper (considered outside of the thesis' scope) was also written during the doctoral training period:

Moss, J. and Tveten, M. (2019). kdensity: An R package for kernel density estimation with parametric starts and asymmetric kernels. *Journal of Open Source Software*, 4(42), 1566.

Contents

Chapter 1

Introduction

Both in science and industry, the sizes of data sets are growing. But without appropriate tools for turning the data into insight, the value of harvesting more data is severely limited. This has created a surge in demand for statistical methods capable of handling enormous data sets, both in the sense of offering reasonable computing time as well as being methodologically sound. That is, modern statistical methods should not only be consistent, powerful and accurate, but also computationally *scalable*.

Apart from consisting of many measurements, big data sets can be extremely diverse. In long, multivariate time series, the typical assumption of stationarity frequently does not hold in practice. The problem of detecting if and when some distributional properties of the data change over time has therefore found increasing applied interest in recent years. For example, Eckley et al. (2020) use change detection methodology to remotely detect the location of gas emission sources utilising data obtained from sensors mounted on an airplane, Gao et al. (2020) use it for monitoring the surface-temperature of organ transplants, and Lévy-Leduc and Roueff (2009) search for anomalies in large amounts of network traffic data. Other areas where detecting changes has become an integral part include software reliability engineering (Mendiratta et al., 2019), research on telecommunications networks (Bardwell et al., 2019) and econometrics (Hlávka et al., 2017).

The motivating application for this thesis is detection of anomalies in sensor-monitored machinery. In this setting, several sensors are placed on different locations of a machine, for instance a pump or a motor, to measure the temperature, pressure or other quantities of interest over time. The machine is monitored to detect if it is not operating as supposed to, either to optimise performance or to avoid costly or dangerous failures. This applied problem translates well to a statistical change detection problem, as a significant change in the sensor data relative to its normal behaviour often signals that something is off with the machine. For example, if the hourly mean temperature of a motor is higher than it normally is, this may indicate that something is wrong with the cooling system. An idealised example of such temperature monitoring is shown to the left in Figure 1.1. For illustrational purposes, there are only four sensors in this example, but in practice, there may be several hundred sensors making measurements every second. Monitoring the sensor readings by eye is therefore not feasible. In addition, subtler changes can be detected when combining the information across all the sensors in a principled way.

A feature of the sensor data encountered in the present thesis we particularly focus on is cross-correlation—correlation between the sensors at any given time, due to, e.g., the proximity of the sensors (Figure 1.1, right). Handling and

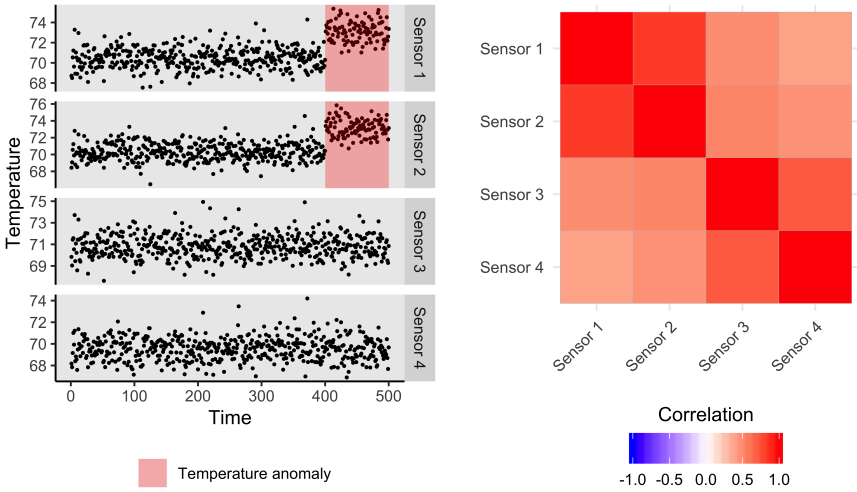

Figure 1.1: A multivariate times series of simulated temperature recordings from four imagined sensors on a ship's motor. Around time-point 400, a part of the cooling system breaks down. As a result, the temperature recordings of sensor 1 and 2 increase to a consistently higher level; the mean temperature has changed. The robustly estimated correlations between the sensors are shown in the matrix to the right. As some sensors are imagined to be relatively close to each other, the correlation between them is strong and positive. See Paper III and Paper IV for a similar type of anomaly detection in real data.

understanding the impact of cross-correlation when combining information from all sensors is important to obtain accurate and trustworthy results. Detecting changes in the correlation structure itself may also be of interest. Moreover, cross-correlation has received relatively little attention in the change detection literature so far, despite its near ubiquitous presence in high-dimensional time series.

There are two different modes of change detection resulting in different but related statistical problems. In *online* change detection, data are collected and analysed in real-time, and the aim is to control the rate of false alarms, but detect true changes as quickly as possible. *Offline* change detection, on the other hand, concerns the retrospective analysis of a historical data set, with the aim of accurately estimating the number and locations of changes. In the sensor-monitoring example, an online method would be used as the real-time monitoring system of the motor, while an offline method could be used to analyse and prepare a training data set for the online method.

We study both online and offline change and anomaly detection for cross-correlated, multivariate time series. Our contributions lie in the intersection of computation and methodology in the form of novel methods that are scalable to scenarios with many sensors or other variables. We also apply change detection

methods to new real world problems. Throughout, the focus is mainly on frequentist methods and parametric models. However, alternatives outside this scope will be touched upon along the way in this introduction.

The rest of the thesis is organised as follows: Chapters 2 and 3 provide background material for putting the papers into context. Chapter 2 starts by formally defining the change detection problem in the offline setting. General computational and methodological frameworks are then introduced in the univariate setting as a stepping stone to the more complex multivariate methods. Next, the anomaly detection problem is presented as a special case of the change detection problem. We finish the chapter by pointing to methods and problems surrounding the scope of the thesis. Chapter 3 partly builds on Chapter 2 to introduce the online version of change detection in a similar fashion. Summaries of the four papers then follow in Chapter 4, emphasising their main contributions. In Chapter 5, I discuss parts of my work in more detail and point to important venues of future research. The four papers in full length conclude the thesis.

Before we continue, some general remarks on notation is due. For a compact presentation, we write $x_{s:e} := \{x_s, \ldots, x_e\}$, where $s < e$. Bold types are used to indicate that an object is a vector rather than a scalar, for example $\mathbf{x}_{s:e} := \{\mathbf{x}_s, \ldots, \mathbf{x}_e\}$, where $\mathbf{x}_t = (x_t^{(1)}, \ldots, x_t^{(p)})^\intercal$. We also let $[n] := \{1, \ldots, n\}$ for $n \in \mathbb{N}$.

Chapter 2

Offline change and anomaly detection

Offline change detection methods take as input a p-variate time series of fixed length, $\mathbf{x}_t$ for $t = 1, \ldots, n$, and aim to answer one or a mix of the four problems:

(P1) Is the data stationary or does its distribution change over time?

(P2) If there are changes, how many changes are there?

(P3) Given a number of changes, at what times do they occur?

(P4) Given the times of change, how does the distribution change?

To be precise, consider the following general model for $\mathbf{x}_1, \ldots, \mathbf{x}_n$: Let the *changepoints* $1 < \tau_1 < \ldots < \tau_K < n$ denote $K < n$ unknown time-points where the data-generating mechanism for $\mathbf{x}_t$ changes abruptly. As a consequence, the observations are divided into $K + 1$ stationary segments with different distribution functions $F_0(\mathbf{x}), \ldots, F_K(\mathbf{x})$. I.e., the data follow a piecewise stationary distribution given by

$$\mathbf{x}_t \sim F_k \quad \text{for } t = \tau_k + 1, \ldots, \tau_{k+1} \text{ and } k = 0, \ldots, K, \tag{2.1}$$

where we define $\tau_0 := 0$ and $\tau_{K+1} := n$. In this model, (P1) is the *testing* problem of whether $K = 0$ or $K > 0$, while (P2)-(P4) are the problems of *estimating* K, $\tau_1, \ldots, \tau_K$ and $F_0, \ldots, F_K$, respectively, preferably combined with measures of estimation uncertainty. Depending on the problem at hand, the ideal goal is to construct the most powerful test or the most accurate estimator.

In most of this thesis, we will not consider models quite as general as (2.1). Firstly, we will mostly work with real-valued vector observations that can be described by a parametric family of densities $f(\mathbf{x}|\boldsymbol{\theta})$, where f is constant, changes occur in the parameter vector $\boldsymbol{\theta}$. Now the model in (2.1) becomes

$$\mathbf{x}_t \sim f(\mathbf{x}|\boldsymbol{\theta}_k) \quad \text{for } t = \tau_k + 1, \ldots, \tau_{k+1} \text{ and } k = 0, \ldots, K, \tag{2.2}$$

where $\boldsymbol{\theta}_{k-1} \neq \boldsymbol{\theta}_k$ for all k. Secondly, we primarily focus on models where the $\mathbf{x}_t$'s are independent in time. Thirdly, as mentioned in the introduction, our focus lies on frequentist methods. Some Bayesian alternatives are given at the end of this chapter.

The prototypical setup is to let f be a normal density with mean $\boldsymbol{\theta}$, as detecting changes in the mean is arguably the most important problem in practice. Plenty of other setups exist, however, for example changes in variance (Hsu, 1977; Inclán and Tiao, 1994), covariance matrix (Wang et al., 2018),

parameters of vector autoregressive models (Wang et al., 2020b), Poisson rates (Henderson and Matthews, 1993), parameters in exponential families (Worsley, 1986). Non-parametric methods for detecting general distributional changes as in (2.1) also exist (Pettitt, 1979; Csörgő and Horváth, 1988).

This chapter gives a brief overview of important methodological developments on the described offline change detection problem. There are a number of more comprehensive reviews in the literature that can be consulted for more details, for instance Truong et al. (2020), Aminikhanghahi and Cook (2017), Niu et al. (2016), Jandhyala et al. (2013), Chen and Gupta (2011), as well as in the theses of e.g. Tickle (2020), Maeng (2019) and Maidstone (2016). We begin by an introduction to some general ideas and frameworks for change detection in the univariate setting.

2.1 General ideas and frameworks—univariate data

Due to the literature on change detection being so vast, there are several ways of categorising all the different change detection methods. Following the review article of Truong et al. (2020) and the work of Killick et al. (2012), I have chosen to structure the exposition based on viewing the offline change detection problem as a problem of optimising a constrained or penalised cost. From this point of view, an offline change detection method consists of three elements: A cost for fitting observations to a specific model, $C(x_{s:e}) \geq 0$, a penalty or constraint for the complexity of the model to avoid overfitting, $P(\tau_{1:K}) \geq 0$, and a search procedure for solving

$$\min_{\tau_{1:K}} \left[\sum_{k=0}^{K} C(x_{(\tau_k+1):\tau_{k+1}}) + P(\tau_{1:K}) \right]. \tag{2.3}$$

The minimising arguments of (2.3) are the changepoint estimates $\hat{\tau}_{1:\hat{K}}$, where $\hat{K}$ is the estimated number of changepoints. In this section we think of the x_t's as univariate observations to fix ideas, but the general framework (2.3) easily carries over to the multivariate setting of Section 2.2.

Note that within this framework, (P1) is answered implicitly through the estimates $\hat{\tau}_{1:\hat{K}}$; the null hypothesis of stationarity is accepted if $\hat{K} = 0$ and rejected otherwise. (P2) and (P3) are solved directly, while (P4) is often answered by construction of the cost function or by a post-processing step given the estimated segmentation.

The cost function is a measure of how well the observations fit the model—the lower the cost, the better the fit—and there is an abundance of costs with different properties available. A prominent example from the changepoint literature is the log-likelihood cost (e.g. Hinkley (1970), Gombay and Horvath (1994), Eckley et al. (2011) and Aue and Horváth (2013)), defined by

$$C(x_{s:e}) = -2 \sup_{\boldsymbol{\theta}} \sum_{t=s}^{e} \log f(x_t|\boldsymbol{\theta}) \tag{2.4}$$

for independent and identically distributed (i.i.d.) observations. Using the log-likelihood cost results in a penalised maximum likelihood approach to change detection. As in many other contexts, the maximum likelihood approach results in estimators with desirable properties, such as consistency of the estimated changepoints under the true model and certain regularity conditions (He and Severini, 2010). The maximum likelihood approach to offline change detection can be traced back to Hinkley (1970), who studied (P3) (estimating the location of a change) in the case of a single change in the mean of Gaussian data with known variance. Other examples of costs include quadratic loss (Chen and Gupta, 2011), absolute loss (Bai, 1995), outlier-robust costs (Huber, 2004; Hušková, 2013; Chakar et al., 2017; Fearnhead and Rigaill, 2019) and nonparametric costs (Zou et al., 2014b). A selection of common costs can be found in Truong et al. (2020).

The penalty function measures the complexity of a given changepoint model. It is essential in obtaining an accurate estimate of the number of changes, K, as it governs how much the cost must be reduced for it to be worth adding an additional changepoint, thereby increasing the model complexity. Excluding a penalty in the change detection problem with an unknown number of changes would result in maximal overfitting as the optimum of (2.3) would be to add a changepoint at every observation, i.e. $\hat{\tau}_{1:\hat{K}} = [n-1]$. The most common penalty function is linear in the number of changepoints; $P(\tau_{1:K}) = \beta K$. This penalty includes standard model selection tools like Akaike's information criterion (Akaike, 1974) when $\beta = 2d$ and the Bayesian or Schwarz' information criterion (Schwarz, 1978) when $\beta = d \log n$, where d is the number of additional parameters in the model per changepoint added. An example of a non-linear penalty that is tailored to the change in mean problem is the modified Bayesian information criterion (Zhang and Siegmund, 2007), given by $P(\tau_{1:K}) = 3K \log n + \sum_{k=0}^{K} \log\big((\tau_{k+1} - \tau_k)/n\big)$. This penalty favours models with evenly spaced changes. More examples of penalties will emerge as we go along in this chapter.

When it comes to search methods, there are particularly two popular classes of algorithms we will treat in more detail. The first approach is based on *model selection* and solves (2.3) *exactly* by a dynamic programming scheme. The second and oldest approach solves (2.3) *approximately* by recursively applying *tests* for the existence of a single changepoint to narrower and narrower windows of the data. After presenting these two classes of algorithms, we go on a quick tour of notable alternatives.

Dynamic programming-based methods Multiple change detection methods based on dynamic programming define recursions for finding the exact optimum of (2.3). The optimal partitioning method of Jackson et al. (2005) is a cornerstone among such algorithms. It can only be used for linear-in-K penalties, but in return, it finds the optimum in $O(n^2)$ time, provided computation of the cost does not depend on n. This is the case for most costs as long as independence between observations in different segments is assumed. The key to optimal partitioning is to define $F(t)$ as the optimal penalised cost for data $x_{1:t}$. It starts

by $F(0) = -\beta$, and then proceeds by computing

$$F(t) = \min_{i<t} \big\{ F(i) + C\big(x_{(i+1):t}\big) + \beta \big\}. \tag{2.5}$$

The optimal cost is given by $F(n)$.

Although a reduction from exponential to quadratic in n computing time is remarkable, it is still prohibitive for sufficiently large n. Motivated by this, the pruned exact linear time (PELT) algorithm of Killick et al. (2012) refines optimal partitioning by only considering relevant i's in the minimisation in (2.5) at each step t. This is made possible by the observation that adding a changepoint always reduces the cost. Therefore, if at time $t_2 > t_1$, the inequality

$$F(t_1) + C(x_{(t_1+1):t_2}) + \beta \geq F(t_2) \tag{2.6}$$

holds, then t_1 can never be the most recent changepoint for all $t_3 > t_2$. In other words, t_1 can be "pruned" from the set of candidate changepoints after time t_2. The effect of pruning in practice is roughly to automatically discard times before a true changepoint. Consequently, PELT can scale linearly in n if the expected number of true changepoints also scales linearly with n, but it remains quadratic like optimal partitioning in the worst-case scenario of no changes. Parallelisation can further reduce the computational burden (Tickle et al., 2020), though at the price of sacrificing exactness of the solution. Even without parallelisation, the computational savings achieved by PELT is massive for many practical problems, making it an increasingly popular method. We also derive a PELT type algorithm in Paper IV.

If only changes in a single parameter is of interest, a very fast alternative to the inequality type pruning in PELT is so-called functional pruning in the functional pruning optimal partitioning algorithm of Maidstone et al. (2017). This type of pruning results in a substantial increase in candidate changepoints being pruned, irrespective of the true number of changes present. Functional pruning optimal partitioning can also be used to fit models where parameters are dependent across segments, as opposed to PELT.

As noted, optimal partitioning, PELT and functional pruning optimal partitioning can only be used with a linear penalty. If a non-linear penalty is preferred, the segment neighbourhood algorithm of Auger and Lawrence (1989) is an alternative. Segment neighbourhood passes through the data recursively as optimal partitioning, but also conditions on the number of changepoints in a particular segment. That is, it starts by computing the optimal segmentation for a single change, before recursively updating the optimal segmentation for one added change until a user-input maximum number of changes $\overline{K} < n$ is reached. Consequently, segment neighbourhood requires $O(\overline{K}n^2)$ operations to find the optimum. If K is completely unknown, this means cubic scaling in n, which limits its use to small data sets. As for optimal partitioning, the speed of segment neighbourhood can be improved by pruning techniques (Rigaill, 2010; Maidstone et al., 2017). Using a linear penalty with PELT or functional pruning optimal partitioning, however, remains a vastly more computationally viable option for large data sets.

Binary segmentation-based methods Another large class of multiple change detection algorithms emerges from the following idea: Let $T(\tau, x_{1:n})$ be a test statistic for a changepoint at τ in the series of observations $x_{1:n}$. This could be any of your favourite tests for a difference in distribution between the sample $x_{1:\tau}$ and $x_{(\tau+1):n}$—a t-test or likelihood ratio test for example. A natural test for the presence of a single changepoint is then to compute $\hat{T}(x_{1:n}) = \max_\tau T(\tau, x_{1:n})$ and compare it with a threshold b. If $\hat{T}(x_{1:n})$ is above b, a change is detected and estimated to be located at the maximising changepoint, $\hat{\tau}$. By splitting the sample at $\hat{\tau}$, the same procedure can be applied to each of the two segments $x_{1:\hat{\tau}}$ and $x_{(\hat{\tau}+1):n}$ to identify further changes, and so forth on each segment as long as the test is significant. This is the binary segmentation algorithm and it "is arguably the most established search method used within the changepoint literature" (Killick et al., 2012). It is often attributed to Vostrikova (1981), Scott and Knott (1974) and Edwards and Cavalli-Sforza (1965).

The way binary segmentation approximates the optimisation problem (2.3) becomes more apparent by considering test statistics of the form

$$T(\tau, x_{1:n}) = C(x_{1:n}) - C(x_{1:\tau}) - C(x_{(\tau+1):n}). \tag{2.7}$$

For a log-likelihood cost, (2.7) is the likelihood ratio test. Maximising this test over τ is the same as finding the single changepoint which provides the greatest decrease in cost. The threshold b governs how much the cost must be reduced when adding a changepoint for it to be considered a change, and can thus be viewed as a linear penalty in the number of changepoints.

There are at least three advantages of using binary segmentation. Firstly, it is computationally fast, only requiring $O(n \log n)$ operations. Secondly, it is easy to implement and modular. Thirdly, it is conceptually simple as it essentially reduces the multiple changepoint problem to a single changepoint problem, which can be further reduced to a (multiple) testing problem. Binary segmentation has also been shown to be consistent (Venkatraman, 1993) in scenarios where adjacent changepoints are sufficiently far apart. In total, this makes binary segmentation applicable to a wide range of old and new change detection problems. All that is needed is a test statistic for discriminating between distributional features of interest.

The main disadvantage of binary segmentation is so-called masking, which is due to its particular approximative nature. A typical example is when changes occur frequently and two close-by changes cancel each other out in the test for a single change. Generally, masking refers to change scenarios where at least one change is missed.

As a result, several tweaked versions of binary segmentation have recently been proposed to make it robust to a larger range of changepoint configurations. Circular binary segmentation of Olshen et al. (2004) is an early modification for detecting changes that switch back and forth between two distributional regimes. Later, the wild binary segmentation algorithm of Fryzlewicz (2014) has drawn much attention as it provably provides error-rate-optimal changepoint estimates (both (P2) and (P3)) in a certain sense (Wang et al., 2018, 2020a). Rather

than deterministically splitting each segment at the optimal single changepoint, wild binary segmentation draws intervals at random to search for a single change. Achieving the mentioned optimal rates, however, may require a very large amount of intervals, hence losing the computational advantage over the exact search methods, like PELT. Recent further improvements include wild binary segmentation 2 (Fryzlewicz, 2020), the narrowest-over-threshold method (Baranowski et al., 2019) and seeded binary segmentation (Kovács et al., 2020).

It should be mentioned that the most popular test statistic to use within binary segmentation is the cumulative sum (CUSUM) statistic. It can be traced all the way back to the first articles on change detection by Page (1954, 1955), who considered the online version of (P1) (testing for the presence of a change) in the context of industrial quality control. Hinkley (1971) later considered (P3) (estimating the location of a change) for Page's CUSUM in the offline setting with a single change.

In modern offline change detection literature (e.g. Wang and Samworth (2018); Fryzlewicz (2014); Aue and Horváth (2013)), the CUSUM statistic mostly does not refer to Page's CUSUM, but to the statistic

$$T(\tau, x_{1:n}) = \sqrt{\frac{\tau(n-\tau)}{n}} \left(\frac{1}{n-\tau} \sum_{t=\tau+1}^{n} x_t - \frac{1}{\tau} \sum_{t=1}^{\tau} x_t \right). \tag{2.8}$$

This statistic is equivalent to the positive root of the likelihood ratio statistic for a single change at τ in the mean of Gaussian data with known variance, and it serves as a blueprint for many other change detection tests. For example, Inclán and Tiao (1994) derive a test for a change in the variance by using cumulative sums of x_t^2, and Lee et al. (2003) further extend this idea by switching x_t in (2.8) with an appropriate function $g(x_t)$ for detecting a general parameter of interest. The simple form of CUSUM tests is what drives their popularity, as it facilitates both quick computation and theoretical analysis. An important result is that a large class of CUSUMs converge in distribution to a Brownian bridge (e.g. Lee et al. (2003)), which is helpful for tuning the threshold b in certain scenarios.

Not all CUSUMs fit nicely into the story of costs, penalisation and search methods. However, some CUSUMs are related to likelihood ratios (Inclán and Tiao, 1994) and squared error loss. As such, they can be viewed as another layer of approximation in (2.3) in addition to binary segmentation. Despite being approximative in general, the theoretical results on the consistency and optimality of wild or plain binary segmentation mentioned here use CUSUM type test statistics (Venkatraman, 1993; Wang et al., 2018, 2020a).

Other search methods There is a growing number of search methods and approaches apart from those we have seen so far based on dynamic programming and binary segmentation. We now briefly present a selection of these alternatives.

Binary segmentation can be described as a "top-down" search method as it starts with the entire stretch of data, before splitting it into smaller and smaller pieces. A natural alternative is therefore a "bottom-up" search method, where one initially starts with a changepoint at every observation, before merging

segments until some criterion is met. Such methods are still new to the change detection field, only recently having been explored by Matteson and James (2014) and Fryzlewicz (2018). These articles, however, suggest that such methods can be competitive with binary segmentation type methods, especially in scenarios with frequent changes.

Another alternative set of methods related to binary segmentation are moving sum methods, proposed for change detection by Preuss et al. (2015) and Eichinger and Kirch (2018), building on similar approaches to testing, e.g. Hušková and Slabý (2001). Moving sum methods, like binary segmentation, are based on testing for a single changepoint, but do so by sliding a window of a certain bandwidth across the time series, testing for a change at the window's midpoint. Given an appropriate bandwidth, moving sum methods can also be shown to be consistent for the number and location of changes, and are quick to compute as well as conceptually simple. Their main drawback is that performance crucially depends on a well-tuned bandwidth parameter.

Other model selection approaches also exist, where the simultaneous multiscale changepoint estimator for detecting changes in the mean proposed by Frick et al. (2014) has recieved much attention. Their take on the change detection problem is to minimise the number of changepoints over all potential piecewise constant mean signals within the acceptance region of a multiscale test. They show that this corresponds to a certain penalised cost, facilitating quick computation, and prove that the family-wise error rate of the number of estimated changes is controlled. Moreover, confidence sets for the locations of the changepoints as well as the piecewise constant mean can also be constructed. Pein et al. (2017) extend the simultaneous multiscale changepoint estimator to heterogeneous data, and Li et al. (2016) propose a related method for controlling the false discovery rate rather than the family-wise error rate, as control of family-wise error rate often leads to underestimating the number of changes. Unfortunately, the framework underpinning these multiscale methods only works for univariate data.

A model selection penalty that is linear in the number of changepoints is connected to an L_0-penalty on the sums of differences of a piecewise constant mean. Harchaoui and Lévy-Leduc (2010) exploit the link between L_0 and L_1 penalisation to create a computationally efficient changepoint estimator, similar to the famous LASSO regression estimator (Tibshirani, 1996). However, the L_1-penalty does not balance type I and type II error optimally for change detection (Cho and Fryzlewicz, 2011).

The final class of change detection methods based on model selection we mention is the data-driven penalty selection methods based on "slope heuristics" of Birgé and Massart (2001, 2007), described in Baudry et al. (2012). These methods aim to automatise tuning of penalties, which is often a delicate problem in practice. Their detection performance is good, but they are restricted to small data sets due to poor computational scaling in the sample size.

2.2 Multivariate methods

Data recordings are increasingly often multivariate and high-dimensional rather than univariate in the current "big data" era. This has led to a massive growth in research on multivariate change detection methods over the past ten years. Before reviewing a selection of the literature, we highlight some of the additional challenges connected to multivariate changepoint analysis compared to the univariate setting.

A naive way of detecting multivariate changes is to apply a univariate method to each time series and put a changepoint at each time-point the ensemble of univariate methods detects a change. However, such an approach would suffer from many false positives due to multiple testing, it does not account for dependence between the variables, and it is not be able to borrow strength across signals to detect changes that are small in each variable, but large when seen as a whole. Moreover, the ensemble of univariate methods might not scale well computationally as the number of variables, p, grows. These are the main reasons for taking what we can call a "fully" multivariate changepoint approach.

Now recall the problem formulation in this chapter's introduction, the changepoint models (2.1) and (2.2) in particular. The space of possible distributions per segment, F_k, is now vastly more complex; imagine the possibility of different marginal distributions per variable and different forms of dependence between them. Even under a family of parametric models $f(\mathbf{x}|\boldsymbol{\theta})$, the number of choices for f and ways in which $\boldsymbol{\theta}$ can change becomes exponentially larger in p. A specific additional question in the multivariate setting that has been addressed in the literature (e.g. Jirak (2015) and Fisch et al. (2019b)), and we pursue in this thesis, is the following:

(P5) Given that there is a change, which of the p variables change?

In the case where $\theta_k^{(i)}$ is the k'th segment mean for variable i, for example, the aim is to estimate the subsets $\mathbf{J}_k \subseteq [p]$ of non-zero elements in $\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}$ for $k = 1, \ldots, K$. Indicating which variables change is important to be able to diagnose what the cause of a change may be.

Complicating things further, there is a big difference between trying to detect changes that occur in more or less than $c\sqrt{p}$ variables, for some non-zero constant c (see e.g. Enikeeva and Harchaoui (2019), Cai et al. (2011) or Jeng et al. (2013)). If more than or exactly $c\sqrt{p}$ variables change, we are in a *dense* regime, and if less than $c\sqrt{p}$ variables change, we are in a *sparse* regime. The intuition behind there being two regimes can be explained as follows: In the dense regime, many variables change such that it is beneficial to aggregate information equally across all variables in the search of a change. If this type of aggregation is used in the sparse regime, on the other hand, the noise from the non-changing variables is more likely to drown out the signal from the few changing variables, making the detection problem harder. The boundary between the two regimes just happens to be at $c\sqrt{p}$ in the limit as $p \to \infty$ for changes in the mean of i.i.d. Gaussian observations with known variance. The consequence is that different methods

are optimal for separating the null hypothesis of no change from sparse and dense alternatives, respectively. In addition, it is primarily in the sparse regime it is relevant to ask (P5). It is likely that a boundary between sparse and dense changes also exists for other types of changes and data distributions, but the the exact nature of such a general law is an open problem, to the best of my knowledge.

Changes in the mean As in the univariate setting, changes in the mean vector is the most well-studied problem. Early contributions all consider tests for a single, dense change. As we have seen in Section 2.1, all such tests can be embedded in a binary segmentation type algorithm to detect multiple changes. Srivastava and Worsley (1986) study the likelihood ratio test for a single change in the mean of multivariate Gaussian data when the correlation matrix is unknown but constant. Horváth et al. (1999) later consider a scaled version of the same statistic, but derives its limiting distribution under a more general model with temporally m-dependent noise.

A large portion of modern work concentrates on the problem of testing for a single change, but from a high-dimensional angle. This either means that $p \to \infty$ in theoretical analysis of the method, or that interest lies on methods that are computationally scalable to potentially very large p. Many such tests are based on aggregating information across local test statistics per variable, $T(\tau, x_{1:n}^{(1)}), \ldots, T(\tau, x_{1:n}^{(p)})$, where $T(\cdot, \cdot)$ often is the CUSUM (2.8), but could in principle be any test. Early high-dimensional work focused on models assuming independence between variables $i = 1, \ldots, p$—what we call *cross-independence*—and assumed that the change is dense. For example, Bai (2010), Horváth and Hušková (2012) and Zhang et al. (2010) all propose an L_2-aggregation of their local statistics under these assumptions. The two former allow for temporal dependence and deal with estimation of a change whose presence is known *a priori*, i.e., (P3) assuming that a change has occured somewhere. Zhang et al. (2010) consider the testing problem (P2) and formulate a model where the change is allowed to be sparse, but their test statistic does not deal with the potential sparsity of the change, nor (P5).

Subsequently, the problem of detecting sparse changes in cross-independent models received increasing attention, as in many practical problems it is clear that only a few variables are likely to be affected. A typical example is the detection of DNA copy number variants, where some variants might only be shared across a few samples. Siegmund et al. (2011) incorporated a prior guess p_0 on the fraction of affected variables. Cho and Fryzlewicz (2015) use a hard-thresholded L_1-aggregation of local CUSUM statistics. Jirak (2015) proposes an L_∞-aggregation, i.e., the maximum of the absolute local CUSUM statistics, and is the first to study (P5). Enikeeva and Harchaoui (2019) propose a statistic based on ordered local CUSUM statistic in combination with an L_2-aggregated CUSUM test to obtain optimal rates of convergence for both sparse and dense changes in independent Gaussian data. Cho (2016) suggests to aggregate the ordered local CUSUMs by another coordinate-wise CUSUM transformation. Lastly,

Wang and Samworth (2018) derive an optimal projection (i.e., aggregation) of CUSUMs, and offer a consistent estimator of this projection direction by a sparse singular value decomposition on the CUSUM transformed data. Note that Jirak (2015) and Wang and Samworth (2018) also extend their methods to allow for cross-dependence.

There are few penalised cost-based methods for the high-dimensional setting. Two contributions in this direction are Fisch et al. (2019b) and Tickle (2020, Chapter 4), who derive methods for detecting both sparse and dense changes in cross-independent data that are easy to adapt to any parametric model for the marginal distributions.

Most recent high-dimensional literature considering cross-dependent data focus on dense changes (Westerlund, 2019; Bhattacharjee et al., 2019; Li et al., 2019; Wang and Shao, 2020). An interesting exception is Maeng (2019, Chapter 5), who also considers temporal dependence, but does not estimate which variables are affected (problem (P5)). An approach for detecting both sparse and dense changes in the mean of cross-correlated data that is computationally scalable and indicates which variables are affected is generally missing in the literature. We aim to fill this gap by a penalised cost approach in Paper IV.

Changes in the covariance matrix Assessing stability of the covariance matrix of multivariate observations has gained significant recent interest. One reason is that many methods for detecting changes in the mean assume that the covariance matrix is constant over time. The thorough analyst should therefore assess whether this assumption holds. Changes in the covariance matrix—or, equivalently, the precision matrix—may also be of independent interest. Kao et al. (2018), for instance, list several practical problems within finance and economics where this is the case.

Methods for detecting changes in the covariance matrix were first proposed for quality control purposes, e.g. the Gaussian likelihood ratio approach of Sullivan and Woodall (2000) or other control charts (see the review article of Yeh et al. (2005)). An early maximum likelihood treatment of the multiple changes in mean and covariance matrix problem is Maboudou-Tchao and Hawkins (2013), who additionally use the segment neighbourhood algorithm as their search method. Even though it is not connected to a specific publication, note that it is relatively straightforward to plug the Gaussian likelihood with unknown mean and covariance matrix and a linear penalty into the penalised cost (2.3) and optimise with PELT, for instance.

The CUSUM-based work of Aue et al. (2009) marks the starting point of the modern, more theoretically oriented line of research on offline covariance change detection methods. Their method and analysis is impressive as it also considers temporal dependence. Bai (2010) considers changes in the variances (in addition to the means), but not in a general covariance matrix. Later, CUSUM-based methods for covariance changes have been investigated by Cho and Fryzlewicz (2015), Kao et al. (2018), Wang et al. (2018) and Dette et al. (2020). All these methods assume that the mean is constant and the change is dense, except the

very recent work of Dette et al. (2020), where potential sparsity is addressed.

Other recent approaches are proposed by Roy et al. (2017), who consider changes in sparse Markov random field models, which includes sparse precision matrices in Gaussian data as a special case, Avanesov and Buzun (2018), who offer a moving sum-based method applicable both in the offline and online setting, and Wang et al. (2019), who utilise U-statistics and self-normalisation to detect changes in both the mean and covariance matrix. Lastly, Grundy et al. (2020) propose a method for detecting changes in the means and variances of high-dimensional (Gaussian) data by mapping the data into two dimensions—one highlighting changes in mean, and the other highlighting changes in the variance.

Research on changes in high-dimensional covariance matrices is still on an infant stage compared to changes in the mean. The $p(p-1)/2$ parameters involved makes the problem much tougher computationally, and almost all published work has only considered the scenario of dense changes. In Paper I and Paper II we investigate how the classical principal component analysis can be used to alleviate the computational burden. We also consider sparse changes in the covariance matrix.

Changes in other features In many practical situations it can be hard to know both the distribution of the data as well as exactly what type of distributional change is of interest. Hence, deriving nonparametric methods for detecting changes in multivariate data is a hot topic. Needless to say, this is a hard problem in general, both theoretically and computationally, but even more so in high-dimensional settings where the curse of dimensionality kicks in. Be aware that nonparametric methods can be used for detecting the already discussed changes in mean and covariance matrix, but is expected to be less powerful compared to methods specifically made for a particular type of change.

Examples of contemporary multivariate nonparametric change detection methods are the approach based on hierachical clustering and distance measures of Matteson and James (2014), the kernel-based methods of Harchaoui and Cappe (2007), Arlot et al. (2019) and Padilla et al. (2020), the graph-based methods of Chen and Zhang (2015), Chu and Chen (2019) and Liu and Chen (2020), as well as Zhang et al. (2017), who use energy statistics and the Kolmogorov-Smirnov test. Note that all these methods assume that observations are independent in time, and no distinction is made between sparse and dense changes.

We also remark that detection of changes in the quite general class of vector autoregressive models is investigated in Kirch et al. (2015), Safikhani and Shojaie (2020) and Wang et al. (2020b). In addition, Liu et al. (2020) very recently proposed a framework based on U-statistics and CUSUMs for detecting a change in any high-dimensional parameter, with power against sparse and dense changes simultaneously.

2.3 Changepoint-based anomaly detection

One of the many applications of changepoint models is anomaly detection. That is, detecting significant deviations from some baseline behaviour of the data. For example, Olshen et al. (2004) use a changepoint model to detect DNA copy number variations, which might indicate cancer or other diseases; Fisch et al. (2019a) detect an exoplanet based on inferring changes in lightcurve data from a star; and we detect overheating of a ship's propulsion motor in Paper III.

The general changepoint models (2.1) or (2.2) are only useful for detecting certain types of anomalies. In the comprehensive review of Chandola et al. (2009), anomalies are divided into three classes: Global, contextual and collective (the names of the classes are from Fisch et al. (2019a)). Global and contextual anomalies are defined as single observations not conforming to either the global or local pattern of the data. E.g., a temperature measurement of 40°C in Oslo is a global anomaly as it would be a highly unusual temperature any time of the year, whereas a measurement of 10°C would only be a contextual anomaly during the winter. Following the terminology of Fisch et al. (2019a,b), we call both global and contextual anomalies *point anomalies* as they are both single outlying observations. Collective anomalies are collections of related observations that are anomalous only when viewed together. For example, an average temperature of 13°C during April in Oslo, compared to the normal of around 10°C. It is primarily collective anomalies the general changepoint models are capable of detecting, while the presence of point anomalies is known to cause trouble in the form of inaccurate additional changepoints being added (Fearnhead and Rigaill, 2019). In addition, the general changepoint model does not utilise the fact that there is a common baseline distribution for the data in many anomaly detection applications.

On the other hand, classical outlier detection techniques and many existing anomaly detection methods from the machine learning community are not suitable for detecting collective anomalies (Chandola et al., 2009). These methods are made with the aim of detecting point anomalies, and often does not consider the relatedness of observations, for example their time-ordering.

Based on these observations Fisch et al. (2019a,b) develop the penalised cost-based framework *collective and point anomalies* (CAPA) for jointly detecting both point and collective anomalies. The anomaly model first assumes that $\mathbf{x}_t$ has a baseline distribution $f(\mathbf{x}|\boldsymbol{\theta}_0)$. Each of the K anomalies are then modelled by two changepoints; one change from the baseline distribution at time s_k, and one change back at time e_k, where $\{(s_k, e_k]\}_{k=1}^{K}$ form non-overlapping intervals. Such changepoints are known as *epidemic* changepoints in the literature (Kirch et al., 2015). This model can be described by

$$\mathbf{x}_t \sim \begin{cases} f(\mathbf{x}|\boldsymbol{\theta}_1) & \text{for } t = s_1 + 1, \ldots, e_1 \\ \quad\vdots & \\ f(\mathbf{x}|\boldsymbol{\theta}_K) & \text{for } t = s_K + 1, \ldots, e_K \\ f(\mathbf{x}|\boldsymbol{\theta}_0) & \text{otherwise,} \end{cases} \tag{2.9}$$

where $\boldsymbol{\theta}_k \neq \boldsymbol{\theta}_0$ for $k = 1, \ldots, K$, $s_k < e_k$ and $s_{k+1} \geq e_k$. In this model, point anomalies are simply defined as anomalies of length 1, i.e., when $s_k = e_k$, while collective anomalies have length greater than 1; $e_k - s_k \geq 2$. To distinguish the two cases, let $\{(s_k, e_k)\}_{k=1}^K$ refer to the collective anomalies, while $O \subseteq [n]$ denotes the set of point anomaly locations. As in the general changepoint model, the aim is to estimate K, $\{(s_k, e_k)\}_{k=1}^K$ and O, as well as $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K$. The baseline parameter $\boldsymbol{\theta}_0$ is assumed to be known, but it is estimated robustly from the data in practice.

Inference on the positions of the anomalies from data is done by using a PELT type algorithm for efficiently solving

$$\max_{K, \{(s_k, e_k)\}_{k=1}^K, O} \left[\sum_{k=1}^{K} S(s_k, e_k) + \sum_{t \in O} S'(\mathbf{x}_t) \right], \tag{2.10}$$

subject to $e_k - s_k \geq 2$ and no overlap between the intervals specified by $\{(s_k, e_k)\}_{k=1}^K$ and O. In (2.10), $S(s, e)$ is the *penalised saving* for introducing an anomaly, defined as the cost-based test statistic

$$S(s, e) := C(\mathbf{x}_{(s+1):e}, \boldsymbol{\theta}_0) - \min_{\boldsymbol{\theta}} C(\mathbf{x}_{(s+1):e}, \boldsymbol{\theta}) - \beta, \tag{2.11}$$

where β is a penalty for adding an anomaly. $S'(\mathbf{x}_t)$ is the penalised saving for adding a point anomaly at t, and is defined as $S(t-1, t)$, but with a separate penalty β'. Note that maximising the penalised savings in (2.10) is equivalent to minimising the penalised cost. Also, $S(s, e)$ with the log-likelihood cost corresponds to the likelihood ratio test of whether $\mathbf{x}_{(s+1):e}$ has parameter $\boldsymbol{\theta}_0$ or not, with threshold β. Fisch et al. (2019a,b) derive penalties for collective and point anomalies based on controlling the false positive rate in independent Gaussian data. In practice, the penalty can be tuned to achieve a desired false positive rate on a training set consisting exclusively of baseline observations, if available.

The article of Fisch et al. (2019b) concerns anomaly detection in multivariate data, where it might be that only a sparse subset $\mathbf{J}_k \subseteq [p]$ of variables are anomalous, as in the general changepoint model. In this case, the penalty in (2.11) is switched with a penalty function $P(|\mathbf{J}|)$ such that the method becomes powerful for detecting both sparse and dense anomalies. In Paper IV, we extend their method by allowing explicit modelling of cross-dependence.

It should be noted that several other authors tackle the problem of detecting epidemic changes, for instance Olshen et al. (2004), Zhang et al. (2010), Kirch et al. (2015), Aston and Kirch (2018), and Zhao and Yau (2019). Methods from sparse mixture detection are also suitable for detecting epidemic changes, e.g. Jeng et al. (2013) who utilise the higher-criticism test of Donoho and Jin (2004). Yet other methods aim to be robust against outliers (Fearnhead and Rigaill, 2019), or include inference regarding point anomalies (Maeng and Fryzlewicz, 2019).

2.4 Other approaches and related problems

So far in this chapter, we have covered frequentist methodology for detecting abrupt changes in piecewise stationary data, where the changes are aligned across variables in the multivariate setting. We will finish by pointing to important related work outside this scope.

There are several directions of Bayesian changepoint analysis. One school of thought formulates the changepoint problem as a hidden Markov model with a fixed number of states, each state corresponding to a stationary segment between changes (Chib, 1998). Inference is done by Markov chain Monte Carlo (MCMC) and if the number of changepoints is unknown, reversible jump MCMC (Green, 1995) can be used to explore the model space. More recently, Ko et al. (2015) proposed to use a Dirichlet process prior on the transition probabilities of the hidden Markov model, avoiding the prespecification of the number of states, and allowing for uncertainty measures both on the number and locations of changepoints.

Another class of Bayesian changepoint methods uses the product-partition model, of which prominent examples are Barry and Hartigan (1993) and Fearnhead (2006). Here, the prior is put on the time between changepoints instead of the transition probabilities. These approaches seek to avoid the difficulties of setting up appropriate MCMC algorithms, and rather build models that allow for quick and exact simulation from the posterior distribution of the number and locations of changepoints. Bardwell and Fearnhead (2017) recently proposed such a Bayesian method for detecting possibly sparse anomalous segments. We will also mention a few examples of related Bayesian online methods at the end of Chapter 3.

Somewhere between frequentist and Bayesian statistics lie methods for constructing confidence distributions (Schweder and Hjort, 2016). That is, distributions over the parameter space that can be used to visualise confidence intervals at all confidence levels simultaneously. Cunen et al. (2018) propose a framework for constructing confidence distributions for a single changepoint. As the literature on obtaining uncertainty measures for changepoints outside the Bayesian school is scarce, such methods could prove to be valuable.

When it comes to detecting changes in other models than covered here and changes of different types, the literature is growing. Examples include detecting changes in the covariates of regression models (Maeng, 2019; Lee et al., 2016; Leonardi and Bühlmann, 2016), changes in network models (Zhao et al., 2019; Bhattacharjee et al., 2020), multivariate changes that does not align perfectly in time between variables (Fisch et al., 2019b; Bardwell et al., 2019; Eckley et al., 2020), as well as fitting piecewise linear models rather than piecewise constant ones (Fearnhead et al., 2019; Maeng and Fryzlewicz, 2019).

Chapter 3

Online change and anomaly detection

In the online mode of change detection, observations are processed *sequentially* as they arrive, as opposed to the offline setting where an entire data set is collected before analysed *retrospectively*. Looking back at problems (P1)-(P5) posed for offline methods, online methods are primarily concerned with updating inference regarding (P1)—testing whether a change has occured or not—for every new observation $\mathbf{x}_t$ given inference based on $\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}$, potentially for $t \to \infty$. The aim is to detect that a true change has occured as quickly as possible, while controlling the rate of false alarms if not. When a change has been declared, offline methods can be used to answer the remaining questions (P2)-(P5). Nevertheless, online methods typically also output an estimate of the most recent changepoint and how the distribution has changed as a byproduct of testing for the presence of a change.

The vast majority of existing online change detection methods are constructed for solving some version of the following sequential hypothesis testing problem:

$$\begin{aligned} H_0 &: \mathbf{x}_t \sim F_0 \text{ for } t = 1, 2, \ldots. \\ H_1 &: \text{There is a } \tau \geq 0 \text{ such that} \\ &\quad \mathbf{x}_t \sim F_0 \text{ for } t = 1, \ldots, \tau, \\ &\quad \mathbf{x}_t \sim F_1 \text{ for } t = \tau + 1, \tau + 2, \ldots, \end{aligned} \tag{3.1}$$

where $\tau = 0$ refers to the alternative hypothesis of all observations stemming from F_1. Note that this is the same model as (2.1) with $K \in \{0, 1\}$ and $n \to \infty$. It is typically assumed that there is a training set of m observations known to be generated from F_0 available. Most commonly, this training set is used to pre-estimate F_0, before considering F_0 to be known in the sequential problem (3.1). Alternatively, F_0 is assumed unknown and its estimation brought into the sequential problem to account for its estimation uncertainty, in which case the training set is taken as the first m observations in (3.1) and the restriction $\tau \geq m$ added to H_1. F_1 can also be modelled as either known or unknown, depending on the situation. As in the offline chapter, we primarily concentrate on the parametric problem where F_k has a parametric density $f(\mathbf{x}|\boldsymbol{\theta}_k)$, $k = 0, 1$.

We remark that in the online context, the difference between an anomaly and a change introduced in Section 2.3 is not as useful due to F_0 being thought of as a baseline distribution in either case. Thus, when we use "changes" in this chapter, we might just as well have used "anomalies".

A sequential or online change detection method for solving (3.1) is a stopping

time $N \in \mathbb{N} \cup \{\infty\}$. All methods we consider are of the form

$$N = \inf\{t \geq 1 : T(\mathbf{x}_{1:t}) > b_t\}, \tag{3.2}$$

where b_t is a threshold function governing whether a test for a change at time t, $T(\mathbf{x}_{1:t})$, is significant or not.

To specify what is meant by "controlling false alarms" and "quick detection", let P^τ and E^τ denote probability and expectation under the model (3.1) when there is a true changepoint at τ. In particular, P^∞ and E^∞ mean that there is no changepoint and correspond to probability and expectation under H_0. A typical goal for a sequential method N is to find b_t such that the *average run length* (ARL) $E^\infty[N]$ is controlled at a user-specified level γ, and rank methods based on their (worst-case) *expected detection delay* (EDD), given by

$$\bar{E}^\tau[N] := \sup_\tau E^\tau[N - \tau | N > \tau]. \tag{3.3}$$

The lower EDD or response time, the better. The ARL can be viewed as the analog to controlling Type I error in the offline setting, while minimising EDD corresponds to maximising power. It is also a common goal to minimise the worst-worst-case EDD, due to Lorden (1971), defined as

$$\sup_\tau \operatorname*{ess\,sup}_{\mathbf{x}_1,\ldots,\mathbf{x}_\tau} E^\tau[(N - \tau)^+ | \mathbf{x}_1, \ldots, \mathbf{x}_\tau]. \tag{3.4}$$

However, it is often overly conservative and difficult to work with analytically, so the EDD in (3.3) has become more popular. Polunchenko and Tartakovsky (2012) can be consulted for a discussion on most classical performance measures.

A naive way of constructing a method for the online problem would be to apply one of the offline methods from Chapter 2 to the entire batch of data for every new observation. However, doing so results in a highly dependent and complicated multiple testing task, and as the sample size potentially goes to infinity, it is not feasible computationally. Thus, in addition to detecting changes quickly, an algorithm for online change detection should have computational complexity not depending on the current sample size t when updating inference from one observation to the next (Chen et al., 2020).

In the rest of this chapter, a brief overview of online change detection methods is given. The literature on online change detection is far sparser than its offline counterpart. Nevertheless, useful recent surveys include Aminikhanghahi and Cook (2017) and Polunchenko and Tartakovsky (2012), and the two books Siegmund (1985) and Basseville and Nikiforov (1993) give a thorough introduction to classical sequential methods. Our main focus is on methods that are related to the work in the papers of this thesis and fit within the online change detection framework just described. Section 3.1 introduces the most popular classical online methods in the univariate setting, before the multivariate setting is covered in Section 3.2. Section 3.3 provides pointers to recent research on related problems and methods outside the current scope.

3.1 Classical methods—univariate data

CUSUM methods CUSUM statistics play an equally important role in online as in offline change detection. As mentioned in Section 2.1, around (2.8), the CUSUM referred to in the online literature is not the same as in the offline literature, but they have a lot in common. Most importantly, both can be written in terms of cumulative sums and arise from likelihood ratio tests. The offline CUSUM originates from a likelihood ratio test between two unknown means in Gaussian data with known variance, whereas the online CUSUM of Page (1954) arises from a likelihood ratio test between two simple hypotheses;

$$T(x_{1:t}) = \max_{k<t} \sum_{i=k+1}^{t} \log \frac{f(x_i|\boldsymbol{\theta}_1)}{f(x_i|\boldsymbol{\theta}_0)}, \tag{3.5}$$

where $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$ are fixed pre- and post-change parameters. An online CUSUM method is then obtain by plugging (3.5) into (3.2), together with a constant threshold $b_t = b$ tuned to achieve an appropriate ARL.

A major contributor to the CUSUM's popularity is the fact that it can be written in the following recursive form:

$$T(x_{1:t}) = S_t = \left(S_{t-1} + \log \frac{f(x_t|\boldsymbol{\theta}_1)}{f(x_t|\boldsymbol{\theta}_0)} \right)^+, \tag{3.6}$$

where $S_0 = 0$ and $(\cdot)^+ := \max(0, \cdot)$. This recursion is obtained by viewing the CUSUM (3.5) as a repeated sequential probability ratio test with lower boundary 0 and upper boundary b (Basseville and Nikiforov, 1993, p. 38). Every time $T(x_{1:(t-1)})$ is below 0—i.e., the null hypothesis of no change is accepted—the test is restarted. In addition, the CUSUM's simple form facilitates theoretical analysis. As $t \to \infty$ the CUSUM behaves like a Brownian motion (Siegmund, 1985), which can guide the selection of the threshold b. It has also been proven that the CUSUM is optimal in terms of minimising the worst-worst-case EDD (3.4) asymptotically as the ARL $\gamma \to 0$ (Lorden, 1971), and for every $\gamma > 0$ (Moustakides, 1986).

The most problematic aspect of Page's CUSUM is that it not only assumes the pre-change distribution to be known, but also the post-change distribution, which is rarely the case in practice. A number of tweaks to the CUSUM have therefore been proposed since its initial release, aiming at adapting to unknown distributions while retaining the simple computational form. In Paper III, we use the post-change adapting CUSUM of Lorden and Pollak (2008) for detecting overheating in ship engines. Other examples of CUSUMs adapting to unknown pre- or post-change parameters are Pollak and Siegmund (1991) and McDonald (1990).

Generalised likelihood ratio methods An alternative class of online change detection methods for handling unknown parameters in both the pre- and post-change distribution are generalised likelihood ratio (GLR) methods. They

incorporate maximum likelihood estimation of the unknown parameters. Hence, in the case of known pre-change parameter and unknown post-change parameter, GLR methods are defined by test statistics of the form

$$T(x_{1:t}) = \max_{k<t} \sup_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \left[\sum_{i=k+1}^{t} \log \frac{f(x_i|\boldsymbol{\theta})}{f(x_i|\boldsymbol{\theta}_0)} \right], \tag{3.7}$$

where $\boldsymbol{\Theta}$ is a subset of the parameter space (see Basseville and Nikiforov (1993) or Lai (1995)). In the case of exponential families and composite alternative hypotheses, such statistics are optimal in the sense of Lorden (1971). Additionally assuming an unknown pre-change parameter brings us back to a statistic of the form (2.7) with log-likelihood cost (2.4), maximised over all $\tau < t$ for each new observations x_t. Unfortunately, in either case, the maximisation over the parameter space for each t and $k < t$ implies that plain GLR methods have computational complexity growing to infinity with the sample size.

Several solutions to alleviate the computational burden of GLR methods have been proposed, of which some are listed in the introduction of Lai (1995). The perhaps most widely used solution is to restrict the maximisation over candidate changepoints k to a set $\mathcal{K} \subseteq [t-1]$, for example a window of length w; $\mathcal{K} = \{k \geq 0 : t - w < k < t\}$. The effect of using a window is that only changes of a certain minimum size can be detected, with wider windows allowing for detectability of smaller changes and vice versa. Lai (1995) discusses how $\mathcal{K}$ can be constructed such that a vanishingly small amount of performance is lost. Such tricks bound the number of operations GLR methods need to update inference from one observation to the next, but the computational burden remains significantly larger than for CUSUM methods.

Moreover, note that it is significantly more complicated to evaluate the distribution of a GLR stopping time (3.2) than one based on a CUSUM. This is the case even for a change in mean in Gaussian data with known variance and pre-change mean, although Siegmund and Venkatraman (1995) derive approximations to the ARL that are quite accurate.

Other methods Several other methods have frequently been used for change detection, many originating from statistical process control. Two prominent examples are the Shewart's chart (Shewhart, 1925) and the exponentially weighted moving average chart (Hunter, 1986).

An alternative to the GLR statistic for an unknown post-change parameter is the so-called "mixture" or "weighted" likelihood ratio approach of Pollak and Siegmund (1975). Rather than maximising over the unknown parameter in (3.7), the mixture likelihood ratio approach integrates the likelihood ratio with respect to some probability distribution of the post-change $\boldsymbol{\theta}$.

The final classical method we mention is the Shiryaev-Roberts chart, due to Shiryaev (1963) and Roberts (1966). The Shiryaev-Roberts chart is a Bayesian analog to Page's CUSUM, and is given by exchanging the maximisation with summation in (3.5). It is a rather popular method as it is provably optimal in a certain Bayesian sense (see Polunchenko and Tartakovsky (2012, Section 4)).

3.2 Multivariate methods

We now present some important contributions to online change detection in multivariate data. As in the offline setting, the point of taking a multivariate approach is to be able to detect smaller changes more reliably than would be possible by a set of univariate methods. The distinction between sparse and dense changes is just as relevant in the online setting, as well as the additional challenge (P5) of identifying which variables are changing.

Changes in the mean For the prototypical change in mean setting, a major line of research on multivariate methods considers different ways of aggregating sequential changepoint tests applied to each univariate time series $x_{1:t}^{(j)}$, for $j = 1, \ldots, p$. Roughly, Tartakovsky et al. (2006), Siegmund and Yakir (2008) and Mei (2010) propose aggregation-based tests for dense changes, while Xie and Siegmund (2013), Liu et al. (2017) and Chan (2017) focus on sparse changes. All these works consider individual tests of either CUSUM, GLR or Shiryaev-Roberts type, except Liu et al. (2017) who consider aggregation of any individual test of choice. Chan (2017) proves that his GLR-based method is optimal for detecting positive mean changes in the worst-worst-case sense of Lorden (1971). Alternative methods include the higher-criticism-based method of Zou et al. (2014a), the sketching- and dimension reduction-based method of Cao et al. (2019), as well as the recently proposed method of Chen et al. (2020), who combine CUSUMs both over variables and different post-change sizes of the means.

Changes in the covariance matrix Online detection of changes in the covariance matrix has yet to receive sufficient attention in the modern literature. An overview of methods for this problem from statistical process control is given by Yeh et al. (2005), and Sullivan and Woodall (2000) as well as Hawkins and Zamba (2009) study the GLR for detecting general changes in the mean and/or covariance matrix of multivariate normal data. Recent contributions are the moving sum-based approach of Avanesov and Buzun (2018) and the CUSUM-based method of Xie et al. (2018) for detecting changes in a spiked covariance matrix model. To the best of my knowledge, all existing methods are constructed to be efficient for dense alternatives. Detecting sparse changes in the covariance matrix is a problem we investigate in Paper II.

Changes in other features For sequentially detecting changes in other features than the mean or covariance matrix, one strategy is to decide on a likelihood for the data and construct a multivariate CUSUM or GLR test in a similar way as described for the univariate case in Section 3.1. Alternatively, one of the aggregation strategies for changes in the mean can be applied to any univariate or lower dimensional likelihoods of choice, as suggested by Liu et al. (2017). Recent nonparametric methods are the kernel-based method of Li

et al. (2015) and the method based on windowed Kolmogorov-Smirnov tests of Madrid Padilla et al. (2019).

3.3 Other approaches

Not all the online literature fall nicely within the framework of controlling the ARL and minimising EDD (3.3). One deficiency of the classical methods is that the probability of declaring a change goes to one as t goes to infinity, i.e., a false alarm will eventually be raised. As a remedy, Chu et al. (1996) propose a different framework enabling control of $P^{\infty}(T < \infty)$ at a chosen significance level α under the asymptotic regime of m—the size of the training set—going to infinity. This approach has gained popularity in recent years, with methodology applicable in very general data scenarios being put forward by e.g. Aue et al. (2012), Kirch and Tadjuidje Kamgaing (2015) and Gösmann et al. (2020).

As online methods aim to update inference incrementally as data arrive, a Bayesian formulation in terms of updating the posterior distribution for every new observation seems a very natural one. Adams and MacKay (2007) and Fearnhead and Liu (2007) initialised the line of research on such methods. They utilise the product partition model as in the offline setting, and put a prior on the length between successive changepoints. These Bayesian methods are closer in spirit to offline methods as they aim to estimate the number and locations of changepoints, but in an online fashion, rather than detecting changes as quickly as possible. In addition, they have the advantage of providing uncertainty quantification of all unknown parameters, given the prior. Recent contributions to this class of methods include Ruggieri and Antonellis (2016), who introduce less informative priors, the multivariate anomaly detector of Bardwell and Fearnhead (2017) and the outlier-robust methodology of Knoblauch et al. (2018).

Chapter 4

Summaries of the papers

4.1 Paper I

Tveten, M. (2019). Which principal components are most sensitive in the change detection problem? *Stat*, 8(e252).

Principal component analysis (PCA) is arguably the most common method for reducing the dimensionality of multivariate data. It has been used for numerous applications both in statistics and machine learning, and it is therefore no surprise that it also forms the basis of many multivariate anomaly detection methods. In this short article, the behaviour of PCA within the change detection problem is investigated through a notion of each pre-change principal component's *sensitivity* to a change.

To be precise, consider the single changepoint setup where $\mathbf{x}_t \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ for $t = 1, \ldots, \tau$, and $\mathbf{x}_t \sim N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ for $t = \tau + 1, \ldots, n$. Now let $\{\lambda_j, \mathbf{v}_j\}_{j=1}^p$ denote the normalised eigensystem of the pre-change $\boldsymbol{\Sigma}_0$, ordered decreasingly in λ_j. Our objects of interest are the *pre-change principal components* $y_{j,t} = \mathbf{v}_j^\intercal \mathbf{x}_t$. Before a change, the distribution of $y_{j,t}$ is $p(y) = N(y|0, \lambda_j)$, while after a change, the distribution of y_j is $q(y) = N(y|\mathbf{v}_j^\intercal \boldsymbol{\mu}_1, \mathbf{v}_1^\intercal \boldsymbol{\Sigma}_1 \mathbf{v}_1)$, where it is assumed without loss of generality that $\boldsymbol{\mu}_0 = \mathbf{0}$. The sensitivity to a change of the j'th pre-change principal component is then defined as the Hellinger distance between its marginal distribution before and after a change, given by $H(p_j, q_j)$.

The main contribution of this paper is to prove that for bivariate normal data, the least varying pre-change principal component, $y_{2,t}$, is the most sensitive for a range of pre-change covariance matrices $\boldsymbol{\Sigma}_0$, and changes $\boldsymbol{\mu}_1$ and $\boldsymbol{\Sigma}_1$. Most notably, $y_{2,t}$ is almost always the most sensitive if only a single parameter of the original distribution changes, i.e., in cases where one of the means, one of the variances or the correlation parameter of the original data change. This result suggests that the least varying pre-change components should be used for detecting sparse distributional changes in higher dimensional data as well.

4.2 Paper II

Tveten, M. and Glad, I. K. (2019). Online detection of sparse changes in high-dimensional data streams using tailored projections. *Manuscript.*

This article builds on the insights from Paper I to propose a method for tailoring the choice of pre-change principal components to a specific change or anomaly detection problem. We call this method tailored PCA (TPCA),

and it is implemented in the accompanying R package `tpca`. In addition, we combine TPCA with an extension of the online change detection scheme of Xie and Siegmund (2013) to create a method for detecting changes both in the mean and the covariance matrix of potentially high-dimensional data. As mentioned in Chapter 3.2, online detection of changes in variance and correlation is an understudied subject. Note that TPCA can also be used in the offline setting. It is especially suitable for anomaly detection because of the explicit assumption of a baseline parameter that the choice of pre-change principal components can be based upon.

To select pre-change principal components by TPCA, the following ingredients are needed: A pre-change covariance matrix, $\mathbf{\Sigma}_0$, a divergence measure, a distribution over the post-change parameter space called the *change distribution* and a cutoff value $c \in [0, 1]$. In the paper, we use the Hellinger distance throughout in agreement with Paper I, but the `tpca` R-package allows any measure of divergence to be used. Using the notation of Section 4.1, we aim to rank the principal components' sensitivity to changes by

$$P_j := \mathbb{P}\left(\underset{1\leq i\leq p}{\operatorname{argmax}}\, H(p_i, q_i) = j | \mathbf{\Sigma}_0\right) \tag{4.1}$$

for $j = 1, \ldots, p$, where the probability is taken with respect to the change distribution. In practice, simulations from the change distribution is used to estimate P_j. TPCA selects the pre-change principal components indexed by

$$\mathcal{J} = \min_{\mathcal{I}\subseteq\{1,\ldots,p\}} \sum_{j\in\mathcal{I}} P_j \geq c. \tag{4.2}$$

In our simulated test scenarios, $\mathcal{J}$ almost always corresponds to a small subset of the least varying pre-change principal components, often facilitating a dimension reduction of $80 - 98\%$ for $c \in [0.8, 0.999]$.

In the simulations for assessing the performance of our TPCA-based online change detection method, we focus on detecting both sparse and dense changes in the mean, variance and correlation. If the correlation coefficients in $\mathbf{\Sigma}_0$ is sufficiently large, we find evidence of our method being able to detect changes quicker from a small set of principal components than the baseline method of Xie and Siegmund (2013). I.e., we observe quicker detection and computation simultaneously. For weaker pre-change cross-correlation, this clear advantage is not present, but significant dimension reduction is still possible without a great loss in performance.

At the end of the paper, we illustrate how our method can be used on time-dependent data by using dynamic PCA in place of the classic PCA, and compare our method to dynamic PCA as used within stochastic process control. This illustration is performed on a realistically simulated dataset of the Tennessee Eastman Process. We find that, in settings where there is no extra validation set for tuning the detection threshold, our method is superior to the classical dynamic PCA method.

4.3 Paper III

Hellton, K. H., Tveten, M., Stakkeland, M., Engebretsen, S., Haug, O. and Aldrin, M. (2020). Real-time prediction of propulsion motor overheating using machine learning. ***Submitted for publication.***

In this paper, online change detection methodology is applied to predict overheating in electrical propulsion motors onboard marine vessels. Technology that protects the motors from overheating is obviously critical for the safety of a ship and those on board. The data used in this study contain observations from four vessels, each with three motors and six temperature sensors at various locations per motor, over time periods ranging from 80 to 294 days.

Almost all of the data is collected during normal operating conditions, but there is one known overheating event in one of the vessels' motors. The main contribution of this paper is to show that by using mostly basic statistical tools, the onset of similar overheating events can be detected reliably 60-90 minutes in advance, and thereby avoided in the future. Parts of the method have already been implemented as a new thermal protection function on several ships.

First, we construct a simple but general linear model for predicting the sensor-observed temperatures from other operating variables of the vessel under normal conditions—power and speed of the motors, for example. Then the six series of residuals of the actual temperature observations and the predictions are monitored simultaneously for large, positive changes in the mean by a combination of an adaptive version of the CUSUM (Lorden and Pollak, 2008) and the shrinkage-aggregation framework proposed by Liu et al. (2017). If a sufficiently large, positive change in the residuals' mean is detected, this is taken as an initial sign of overheating, and an alarm is raised.

In this application, it is not only important to be able to detect an emerging overheating event in a timely fashion, but also to keep false alarms to an absolute minimum. If false alarms are too frequent, the operators of the vessel is likely to put a piece of tape over the red lamp meant to indicate an impending fault, which, needless to say, could be catastrophic. Consequently, a methodological contribution of this article is an automatic tuning procedure for the change detection algorithm that takes as input the acceptable number of false alarms in the fault-free training data. This tuning procedure uses information about the known fault—making it a supervised method—and thus risks to overfit to the single observed overheating event. A mechanism for balancing early detection with a countering of overfitting is therefore also built in.

4.4 Paper IV

Tveten, M., Eckley, I. A. and Fearnhead, P. (2020). Scalable change-point and anomaly detection in cross-correlated data with an application to condition monitoring. ***Invited to submit a revision to Annals of Applied Statistics.***

We study and propose methods for the offline multiple anomaly and change detection problems in multivariate data when variables are cross-correlated and changes occur in an unknown subset of the mean components. In addition, we demonstrate the anomaly detection method's usefulness for sensor-based condition monitoring of an industrial process pump. The paper is accompanied by the R package `capacc`, providing efficient implementations of our methods.

The first main methodological contribution of the paper is the derivation of the penalised cost-based methods CAPA-CC (collective and point anomalies in cross-correlated data) and CPT-CC (changepoints in cross-correlated data) for solving each of these problems in a computationally efficient manner. Both methods are built on a particular approximation of the penalised saving (2.11) corresponding to a penalised Gaussian likelihood ratio tests for a single anomaly or change. Encapsulating these tests for a single change or anomaly, CPT-CC uses a binary segmentation type algorithm to detect multiple changes, while CAPA-CC uses a PELT type algorithm to detect multiple anomalies.

An approximation of the penalised saving is necessary for a moderately large p, as the exact maximum likelihood estimator of a subset mean in correlated data corresponds to a combinatorial optimisation problem, as far as we can see. The approximation we propose is motivated from the form of the maximum likelihood estimator and corresponds to what is known as an unconstrained *binary quadratic program*. Such binary quadratic programs are of the form

$$\max_{\mathbf{u}\in\{0,1\}^p} \mathbf{u}^\intercal \mathbf{A}\mathbf{u} + \mathbf{u}^\intercal \mathbf{b} + c, \tag{4.3}$$

where $\mathbf{A}$ is a real, symmetric, $(p\times p)$-dimensional matrix, $\mathbf{b}$ is a real, p-dimensional vector and c is a real scalar. A second major result in the paper, of possibly independent interest, is a dynamic programming algorithm requiring $O(p2^r)$ operations for obtaining an exact solution to (4.3) when $\mathbf{A}$ is r-banded. This algorithm is inspired by the optimal partitioning algorithm (2.5) in the way of proceeding recursively through the variables $d = 1, \ldots, p$ and conditioning on the optimal penalised saving for variables $1, \ldots, d-1$ at each d.

In our problems, $\mathbf{A}$ is banded if the precision matrix $\mathbf{Q}$ is banded. As a consequence, a banded estimate of $\mathbf{Q}$ is required for our methods to be scalable. To obtain an estimate of a desired band we utilise a robust version of the GLASSO algorithm. An important result from our simulation study is that our method performs advantageously compared to other methods in terms of power and estimation accuracy in a range of data settings, also when a truly dense precision matrix is approximated by a banded estimate.

The simulation study also points to interesting facts about which scenarios incorporating cross-correlations is favourable in the change or anomaly detection analysis compared to ignoring it. Surprisingly, if the change is dense and the changed mean components have similar values, ignoring cross-correlations results in a more powerful method.

Chapter 5

Discussion

In Chapters 2 and 3, we introduced the offline and online change detection problems, respectively, and briefly summarised parts of the statistical literature on these topics. The literature review is only meant to provide context for Papers I–IV—summarised in Chapter 4—and is by no means exhaustive. In this chapter, I discuss the papers critically, pointing to limitations and possible improvements not already mentioned in the papers. It is therefore advantageous to read the papers in full length in advance. The chapter is concluded by a discussion of some open challenges and future directions of the change detection field in general.

5.1 Discussion of the papers

Paper I In this paper, I used the Hellinger distance between distributions to define sensitivity to changes partly because it proved simple to work with. It would have been interesting to obtain similar results using the Kullback-Leibler divergence, however, as it is more directly linked to properties of change detection methods. For example, for online methods, Lorden (1971) showed that the optimal worst-worst-case detection delay (3.4) is governed by

$$\bar{D}(g, f) := \frac{\log \gamma}{I(g, f)}, \tag{5.1}$$

asymptotically as γ (the ARL) goes to infinity, where $I(g, f)$ is the Kullback-Leibler divergence from the pre-change distribution f to the post-change distribution g;

$$I(g, k) := \int \log \frac{g(x)}{f(x)} g(x) dx. \tag{5.2}$$

Thus, comparing the Kullback-Leibler divergences $I(p_j, q_j)$, where p_j and q_j are the pre-change and post-change distributions of principal component j as in Section 4.1, can be directly translated to *how much* quicker a particular change can be detected by each principal component. By using the Hellinger distance, we only get to know the ordering of which principal component will be the most efficient to monitor.

Paper II Our TPCA method is a tool for testing the usefulness of the knowledge and concepts from Paper I in practice. Empirically, it seems to work well, but unfortunately, we have little theory to support it. For instance, it would be beneficial to get some measure of uncertainty on the selected subset $\mathcal{J}$ in (4.2), and some guidance on the number of Monte Carlo simulations needed to

approximate the distribution (4.1) well. For our chosen measure for ranking the axes (4.1)—the probability of a principal component being the most sensitive with respect to a distribution over changes—this is hard to obtain. Thus, in a future version of the manuscript, an option is to change the selection criterion for which principal components to monitor to one that can offer more in terms of guarantees on performance.

One such alternative selection criterion we have started to explore is based on the Kullback-Leibler divergence and its connection to the detection delay (5.1). The idea is to keep enough principal components such that a minimum of $c100\%$ of the information about changes occuring according to a change distribution is conserved with probability $1-\alpha$, for chosen $c, \alpha \in (0,1)$. This can be formalised as the problem of finding the minimal $\mathcal{J}$ such that

$$P\Big(\sum_{j\in\mathcal{J}} I_j \Big/ \sum_{j=1}^{p} I_j \geq c\Big) \geq 1-\alpha \tag{5.3}$$

holds, where $I_j := I(p_j, q_j)$. Through this criterion, we can specify how much loss in detection speed is permissible at some probability $1-\alpha$, as $\bar{D} \geq \log\gamma/(c\sum_{j=1}^{p} I_j)$ with probability $1-\alpha$ when monitoring the (5.3)-selected principal components. Moreover, for a multivariate Gaussian change distribution for the mean, $\boldsymbol{\mu} \sim N(\boldsymbol{\theta}, \boldsymbol{\Gamma})$, combined with a Kullback-Leibler divergence between two Gaussians in I_j, the distribution (5.3) for a fixed $\mathcal{J}$ is possible to derive analytically; it can be expressed as the probability distribution of a quadratic form $\boldsymbol{\mu}^\intercal \mathbf{A}\boldsymbol{\mu}$, known to be distributed as a linear combination of independent non-central chi-square random variables. Motivated by the results of the minor principal components being the most sensitive, an approximate minimisation over $\mathcal{J}$ can be performed by starting from $\mathcal{J} = \{p\}$, and progressively adding more and more varying components until the criterion (5.3) is met. Results of this flavour could be useful as computationally efficient default settings, and to approximate more complicated change and data distributions.

More generally, we would like to address the choice of change distribution more thoroughly in the future. In the current manuscript, the change distribution used throughout represents little prior information, but it might seem quite arbitrary. As mentioned in the previous paragraph, finding a choice of change distribution that enables selection of the tailored principal components in a less brute force manner than Monte Carlo simulation would be highly beneficial. Such change distributions could then be studied under misspecified scenarios to assess the value of setting up a more complicated change distribution.

In the simulation study, we have divided results into classes of "low" and "high" correlation based on the value of the α_d parameter in the method of Joe (2006) for generating random correlation matrices being less than or greater than 1. The motivation for using this method was to obtain a large range of different correlation matrices. However, it is not that easy to interpret the size of the correlations in each class. Selecting a few simpler classes of correlation matrices as test beds, as we did in Paper IV, might therefore provide more informative

results in terms of how strong the correlation must be for TPCA to perform better than the mixture procedure of Xie and Siegmund (2013).

Today, there are also more methods it would be relevant to compare performance with, especially the methods mentioned in the paragraph on detecting changes in the covariance matrix of Section 3.2.

Paper III The aim of this paper was to propose a method for detecting when a ship's motor is about to fail. Specifically, the method had to be able to predict an observed fault in a historical data set sufficiently early in advance with a minimal amount of false alarms, be generalisable to other ships and motors, as well as be simple conceptually and simple to implement on the on-board system of the ship. The two latter requirements lead us to using simple i.i.d. Gaussian models for the data, both when constructing the model for the motor temperature and when monitoring the residuals. These modelling assumptions were justified because the size of the change in mean signalling the observed fault was large enough to be detected early with few false alarms, despite the threshold having to absorb all aspects of the data not captured by the i.i.d. Gaussian model. The lesson here, from a practical point of view, is that much can be achieved by a very simple model.

However, other failures may not be equally pronounced as the one in our test set. In failure cases with smaller changes, more effort must be put on modelling the data. There are at least three improvements that would make detection of significantly smaller changes possible, if we disregard the requirement of implementational and conceptual simplicity. First, as mentioned in the discussion section of the paper, there is a consistent bias in the temperature residuals for each sensor. This is due to the model for generating the residuals being based on the average temperature over the six sensors, such that individual differences between the sensors are lost. One way of reducing the bias is thus to construct a temperature model for each sensor by including a training period for each motor. Note that a part of the bias is already handled by the parameter ρ in the adaptive CUSUM, but lowering ρ is also of interest to be able to detect smaller changes. A second improvement is to model the temporal dependence explicitly in the change detection method. The improvement is likely to be remarkable as the temperature residuals are very strongly auto-correlated as a consequence of the motor temperature being a slowly varying process relative to the once per second sampling rate. Thirdly, the spatial dependence between the sensors is also strong, so modelling it would further increase detection power (as the results in Paper IV show).

On the other hand, there will always be behaviour of the temperature sensor data not captured by even an extremely complex model. From the point of view of a change detection method, such deviations from the model will often be interpreted as evidence for a change. Thus, a reformulation of the change detection problem relevant to this application is to only detect *relevant changes*. The ρ-parameter in the adaptive CUSUM in practice filters out too small changes, but another alternative is to incorporate the relevant size of a change directly in

the hypothesis testing problem. For a change in mean in univariate data, this means studying null hypotheses such as

$$H_0 : |\mu_0 - \mu_1| \leq \Delta,$$

where μ_0 and μ_1 are the pre- and post-change means. Initial work on change detection problems of this form has already been carried out by Dette and Gösmann (2018).

Paper IV Much of the discussion of Paper III also applies to the application of condition monitoring a process pump in Paper IV. Specifically, modelling of temporal dependence and a more sophisticated model for removing trends in the data associated with the operational state of the pump is likely to increase performance. By "operational state", I mean, for instance, the volume fractions of the different fluids being pumped, their flow rate, the power of the pump, and so forth.

An online version of CAPA-CC is needed to be able to monitor the pump in real-time. In this particular application, the current offline version is primarily useful for analysing historical data of the pump, either to prepare a training set for an online method, or to explore when the pump has been running suboptimally in the past, perhaps discovering previously unknown anomalous segments. Fisch et al. (2020) recently showed how the univariate CAPA method can be made sequential, and similar ideas can be used to create an online counterpart of CAPA-CC.

On the methodological side there are also numerous possibilities for extensions. In the penalised cost framework of our methods, we use a pointwise minimum between a linear and a constant penalty on the number of changing variables. Akin to the optimal partitioning algorithm in (2.5), the restriction to linear penalties in the sparse regime is what allows for quick computation of the penalised saving for a fixed changepoint or anomalous segment. There may, however, be scenarios where a non-linear penalty is preferred, and Fisch et al. (2019b) show that for intermediately sparse changes—that is, for $p^{-1/2} < |\mathbf{J}| \leq p^{-3/4}$—in cross-independent Gaussian data, a third, non-linear penalty regime is needed for optimal power. It is possible to accommodate for non-linear penalties in our method by deriving a segment neighbourhood analog to our optimal partitioning-inspired algorithm for computing the penalised saving. (The segment neighbourhood algorithm is described at the end of the paragraph on dynamic programming-based methods in Section 2.1.) By this, I mean that in addition to sequentially conditioning on the optimal penalised saving until variable $d \leq p$, one can also condition on the number of changing variables, starting from finding the single variable that increases the penalised saving the most, before proceeding recursively until a maximum number of changing variables, $\bar{J}$, is reached. Such an algorithm would scale quadratically in p if $\bar{J}$ grows linearly in p. This is prohibitive for large p, but for moderately sized p, as in our 5-dimensional pump data example, it may have practical value.

As CAPA-CC and CPT-CC are based on the multivariate Gaussian model, it is of course relevant to explore other costs, both likelihood-based costs and

others. It would be interesting to seek more general models where our algorithm for solving binary quadratic programs can be used to approximate tests for subset changes, or if it can find use in other tasks involving variable selection.

I am open to suggestions on how to obtain stronger theoretical results on the quality of the approximate versus the exact penalised saving.

5.2 Open challenges in change detection

We conclude this introduction by discussing some interesting open challenges in the change detection field.

A major issue with the vast majority of frequentist change detection methods is that they only provide point estimates of changepoints, without a measure of confidence in these estimates. The quality of changepoint estimates is mainly assessed by proving their consistency and deriving convergence rates. As pointed out by Paul Fearnhead in the discussion on Frick et al. (2014), additional challenges with confidence intervals for changepoints arise when the the number of changes are unknown, as is often the case. Should the confidence intervals be constructed with respect to a fixed number of changepoints? How should uncertainty on the number of changepoints be incorporated? And how can confidence intervals for the number of changepoints be constructed? The confidence intervals of Frick et al. (2014) rely on their method consistently estimating the number of changepoints, and then asymptotic confidence intervals for the changepoints are constructed conditional on the estimated number of changes. Continuing to paraphrase Paul Fearnhead, it is not clear how to interpret such confidence intervals in many real data settings, as there is often significant uncertainty regarding the number of changes. The confidence distribution approach of Cunen et al. (2018) would face similar challenges as they assume there is maximally a single changepoint. Bayesian methods, on the other hand, are able to incorporate uncertainty on both the number and location of changepoints simultaneously, and may be the only option for full uncertainty quantification. However, such Bayesian inference is of course conditional on the often subjectively specified prior.

In many applied problems, including both the ship motor and pump monitoring problems of Paper III and Paper IV, the mean function of the data is not constant or linear between changepoints, but contains local fluctuations or trends of a complicated functional form. In our problems, a portion of these trends can be ascribed to a time-varying context of the machines; the temperature of the motor naturally increases as the motor's power increases, for example. Given the true relationship between the power and the temperature of the motor, this trend could be removed entirely. In practice, however, relationships of this sort have to be modelled and estimated, and variables explaining the trend might not always be recorded. Some trends or local fluctations will, consequently, often remain, no matter how hard one tries to remove them. Thus, change and anomaly detection methods that allow the mean function between changepoints to be smoothly time-varying or stochastic are likely to be useful in practice.

Combine this with modelling of temporal dependence and outlier-robustness, and the practical usefulness will increase even further. Initial work in this direction has been carried out by Romano et al. (2020) for univariate data. Multivariate and online versions are still to be explored.

Online or sequential change detection is still an underexplored problem compared to the offline problem, despite the origin of change detection being sequential. The current online field is mainly focused on the speed of detecting a single change. However, in several applied settings, it is more important that detection is reliable in terms of avoiding false alarms than quick, so long as detection is "quick enough". Consequently, a formulation of the online change detection problem starting from what is sufficiently quick detection before minimising the probability of false alarms might be fruitful. Moreover, constructing online versions of the algorithms for existing offline methods could be useful for adapting the online field to multiple change scenarios, thereby bridging the gap between the two settings.

Bibliography

Adams, R. P. and MacKay, D. J. C. (2007). Bayesian online changepoint detection. *arXiv:0710.3742 [stat.ML]*. arXiv: 0710.3742.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723. Conference Name: IEEE Transactions on Automatic Control.

Aminikhanghahi, S. and Cook, D. J. (2017). A survey of methods for time series change point detection. *Knowledge and Information Systems*, 51(2):339–367.

Arlot, S., Celisse, A., and Harchaoui, Z. (2019). A Kernel Multiple Change-point Algorithm via Model Selection. *Journal of Machine Learning Research*, 20:1–56.

Aston, J. A. D. and Kirch, C. (2018). High dimensional efficiency with applications to change point tests. *Electronic Journal of Statistics*, 12(1):1901–1947. Publisher: The Institute of Mathematical Statistics and the Bernoulli Society.

Aue, A. and Horváth, L. (2013). Structural breaks in time series. *Journal of Time Series Analysis*, 34(1):1–16. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9892.2012.00819.x.

Aue, A., Horváth, L., Kühn, M., and Steinebach, J. (2012). On the reaction time of moving sum detectors. *Journal of Statistical Planning and Inference*, 142(8):2271–2288.

Aue, A., Hörmann, S., Horváth, L., and Reimherr, M. (2009). Break detection in the covariance structure of multivariate time series models. *Annals of Statistics*, 37(6B):4046–4087. Publisher: Institute of Mathematical Statistics.

Auger, I. E. and Lawrence, C. E. (1989). Algorithms for the optimal identification of segment neighborhoods. *Bulletin of Mathematical Biology*, 51(1):39–54.

Avanesov, V. and Buzun, N. (2018). Change-point detection in high-dimensional covariance structure. *Electronic Journal of Statistics*, 12(2):3254–3294. Publisher: The Institute of Mathematical Statistics and the Bernoulli Society.

Bai, J. (1995). Least Absolute Deviation Estimation of a Shift. *Econometric Theory*, 11(3):403–436. Publisher: Cambridge University Press.

Bai, J. (2010). Common breaks in means and variances for panel data. *Journal of Econometrics*, 157(1):78–92.

Baranowski, R., Chen, Y., and Fryzlewicz, P. (2019). Narrowest-over-threshold detection of multiple change points and change-point-like features. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 81(3):649–672. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/rssb.12322.

Bardwell, L. and Fearnhead, P. (2017). Bayesian Detection of Abnormal Segments in Multiple Time Series. *Bayesian Analysis*, 12(1):193–218.

Bardwell, L., Fearnhead, P., Eckley, I. A., Smith, S., and Spott, M. (2019). Most Recent Changepoint Detection in Panel Data. *Technometrics*, 61(1):88–98.

Barry, D. and Hartigan, J. A. (1993). A Bayesian Analysis for Change Point Problems. *Journal of the American Statistical Association*, 88(421):309–319. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01621459.1993.10594323.

Basseville, M. and Nikiforov, I. V. (1993). *Detection of abrupt changes: theory and application*, volume 104. Prentice Hall, Englewood Cliffs.

Baudry, J.-P., Maugis, C., and Michel, B. (2012). Slope heuristics: overview and implementation. *Statistics and Computing*, 22(2):455–470.

Bhattacharjee, M., Banerjee, M., and Michailidis, G. (2019). Change Point Estimation in Panel Data with Temporal and Cross-sectional Dependence. *arXiv:1904.11101 [math.ST]*.

Bhattacharjee, M., Banerjee, M., and Michailidis, G. (2020). Change Point Estimation in a Dynamic Stochastic Block Model. *Journal of Machine Learning Research*, 21:1–59.

Birgé, L. and Massart, P. (2001). Gaussian model selection. *Journal of the European Mathematical Society*, 3(3):203–268.

Birgé, L. and Massart, P. (2007). Minimal penalties for Gaussian model selection. *Probability Theory and Related Fields*, 138(1-2):33–73.

Cai, T. T., Jeng, X. J., and Jin, J. (2011). Optimal detection of heterogeneous and heteroscedastic mixtures. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(5):629–662. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2011.00778.x.

Cao, Y., Thompson, A., Wang, M., and Xie, Y. (2019). Sketching for sequential change-point detection. *EURASIP Journal on Advances in Signal Processing*, 2019(1):42.

Chakar, S., Lebarbier, E., Lévy-Leduc, C., and Robin, S. (2017). A robust approach for estimating change-points in the mean of an AR(1) process. *Bernoulli*, 23(2):1408–1447. Publisher: Bernoulli Society for Mathematical Statistics and Probability.

Chan, H. P. (2017). Optimal sequential detection in multi-stream data. *The Annals of Statistics*, 45(6):2736–2763.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):15:1–15:58.

Chen, H. and Zhang, N. (2015). Graph-based change-point detection. *Annals of Statistics*, 43(1):139–176. Publisher: Institute of Mathematical Statistics.

Chen, J. and Gupta, A. K. (2011). *Parametric Statistical Change Point Analysis: With Applications to Genetics, Medicine, and Finance*. Birkhäuser, New York.

Chen, Y., Wang, T., and Samworth, R. J. (2020). High-dimensional, multiscale online changepoint detection. *arXiv:2003.03668 [stat.ME]*. arXiv: 2003.03668.

Chib, S. (1998). Estimation and comparison of multiple change-point models. *Journal of Econometrics*, 86(2):221–241.

Cho, H. (2016). Change-point detection in panel data via double CUSUM statistic. *Electronic Journal of Statistics*, 10(2):2000–2038.

Cho, H. and Fryzlewicz, P. (2011). Multiscale interpretation of taut string estimation and its connection to Unbalanced Haar wavelets. *Statistics and Computing*, 21(4):671–681.

Cho, H. and Fryzlewicz, P. (2015). Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 77(2):475–507.

Chu, C.-S. J., Stinchcombe, M., and White, H. (1996). Monitoring Structural Change. *Econometrica*, 64(5):1045–1065. Publisher: [Wiley, Econometric Society].

Chu, L. and Chen, H. (2019). Asymptotic distribution-free change-point detection for multivariate and non-Euclidean data. *Annals of Statistics*, 47(1):382–414. Publisher: Institute of Mathematical Statistics.

Csörgő, M. and Horváth, L. (1988). 20 Nonparametric methods for changepoint problems. In *Handbook of Statistics*, volume 7 of *Quality Control and Reliability*, pages 403–425. Elsevier.

Cunen, C., Hermansen, G., and Hjort, N. L. (2018). Confidence distributions for change-points and regime shifts. *Journal of Statistical Planning and Inference*, 195:14–34.

Dette, H. and Gösmann, J. (2018). Relevant change points in high dimensional time series. *Electronic Journal of Statistics*, 12(2):2578–2636. Publisher: The Institute of Mathematical Statistics and the Bernoulli Society.

Dette, H., Pan, G. M., and Yang, Q. (2020). Estimating a Change Point in a Sequence of Very High-Dimensional Covariance Matrices. *Journal of the American Statistical Association*. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01621459.2020.1785477.

Donoho, D. and Jin, J. (2004). Higher criticism for detecting sparse heterogeneous mixtures. *The Annals of Statistics*, 32(3):962–994.

Eckley, I., Kirch, C., and Weber, S. (2020). A novel change point approach for the detection of gas emission sources using remotely contained concentration data. *Annals of Applied Statistics*.

Eckley, I. A., Fearnhead, P., and Killick, R. (2011). Analysis of changepoint models. In Barber, D., Cemgil, A. T., and Chiappa, S., editors, *Bayesian Time Series Models*, pages 205–224. Cambridge University Press, Cambridge.

Edwards, A. W. F. and Cavalli-Sforza, L. L. (1965). A Method for Cluster Analysis. *Biometrics*, 21(2):362–375. Publisher: [Wiley, International Biometric Society].

Eichinger, B. and Kirch, C. (2018). A MOSUM procedure for the estimation of multiple random change points. *Bernoulli*, 24(1):526–564. Publisher: Bernoulli Society for Mathematical Statistics and Probability.

Enikeeva, F. and Harchaoui, Z. (2019). High-dimensional change-point detection under sparse alternatives. *Annals of Statistics*, 47(4):2051–2079. Publisher: Institute of Mathematical Statistics.

Fearnhead, P. (2006). Exact and efficient Bayesian inference for multiple changepoint problems. *Statistics and Computing*, 16(2):203–213.

Fearnhead, P. and Liu, Z. (2007). On-line inference for multiple changepoint problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):589–605.

Fearnhead, P., Maidstone, R., and Letchford, A. (2019). Detecting Changes in Slope With an L_0 Penalty. *Journal of Computational and Graphical Statistics*, 28(2):265–275. Publisher: Taylor & Francis.

Fearnhead, P. and Rigaill, G. (2019). Changepoint Detection in the Presence of Outliers. *Journal of the American Statistical Association*, 114(525):169–183.

Fisch, A. T. M., Bardwell, L., and Eckley, I. A. (2020). Real Time Anomaly Detection And Categorisation. *arXiv:2009.06670 [stat.ME]*.

Fisch, A. T. M., Eckley, I. A., and Fearnhead, P. (2019a). A linear time method for the detection of point and collective anomalies. *arXiv:1806.01947 [stat.ML]*.

Fisch, A. T. M., Eckley, I. A., and Fearnhead, P. (2019b). Subset Multivariate Collective And Point Anomaly Detection. *arXiv:1909.01691 [stat.ME]*.

Frick, K., Munk, A., and Sieling, H. (2014). Multiscale change point inference. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 76(3):495–580. Publisher: [Royal Statistical Society, Wiley].

Fryzlewicz, P. (2014). Wild binary segmentation for multiple change-point detection. *Annals of Statistics*, 42(6):2243–2281.

Fryzlewicz, P. (2018). Tail-greedy bottom-up data decompositions and fast multiple change-point detection. *Annals of Statistics*, 46(6B):3390–3421. Publisher: Institute of Mathematical Statistics.

Fryzlewicz, P. (2020). Detecting possibly frequent change-points: Wild Binary Segmentation 2 and steepest-drop model selection. *Journal of the Korean Statistical Society*.

Gao, Z., Du, P., Jin, R., and Robertson, J. L. (2020). Surface temperature monitoring in liver procurement via functional variance change-point analysis. *Annals of Applied Statistics*, 14(1):143–159. Publisher: Institute of Mathematical Statistics.

Gombay, E. and Horvath, L. (1994). An application of the maximum likelihood test to the change-point problem. *Stochastic Processes and their Applications*, 50(1):161–171. Publisher: Elsevier.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732. Publisher: Oxford Academic.

Grundy, T., Killick, R., and Mihaylov, G. (2020). High-dimensional changepoint detection via a geometrically inspired mapping. *Statistics and Computing*, 30(4):1155–1166.

Gösmann, J., Kley, T., and Dette, H. (2020). A new approach for open-end sequential change point monitoring. *arXiv:1906.03225 [math.ST]*. arXiv: 1906.03225.

Harchaoui, Z. and Cappe, O. (2007). Retrospective Mutiple Change-Point Estimation with Kernels. In *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*, pages 768–772, Madison, WI, USA. ISSN: 2373-0803.

Harchaoui, Z. and Lévy-Leduc, C. (2010). Multiple Change-Point Estimation With a Total Variation Penalty. *Journal of the American Statistical Association*, 105(492):1480–1493. Publisher: Taylor & Francis.

Hawkins, D. M. and Zamba, K. D. (2009). A Multivariate Change-Point Model for Change in Mean Vector and/or Covariance Structure. *Journal of Quality Technology*, 41(3):285–303.

He, H. and Severini, T. A. (2010). Asymptotic properties of maximum likelihood estimators in models with multiple change points. *Bernoulli*, 16(3):759–779. Publisher: Bernoulli Society for Mathematical Statistics and Probability.

Henderson, R. and Matthews, J. N. S. (1993). An Investigation of Changepoints in the Annual Number of Cases of Haemolytic Uraemic Syndrome. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 42(3):461–471. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.2307/2986325.

Hinkley, D. V. (1970). Inference about the change-point in a sequence of random variables. *Biometrika*, 57(1):1–17.

Hinkley, D. V. (1971). Inference about the change-point from cumulative sum tests. *Biometrika*, 58(3):509–523. Publisher: Oxford Academic.

Hlávka, Z., Hušková, M., Kirch, C., and Meintanis, S. G. (2017). Fourier–type tests involving martingale difference processes. *Econometric Reviews*, 36(4):468–492.

Horváth, L. and Hušková, M. (2012). Change-point detection in panel data. *Journal of Time Series Analysis*, 33(4):631–648.

Horváth, L., Kokoszka, P., and Steinebach, J. (1999). Testing for Changes in Multivariate Dependent Observations with an Application to Temperature Changes. *Journal of Multivariate Analysis*, 68(1):96–119.

Hsu, D. A. (1977). Tests for Variance Shift at an Unknown Time Point. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 26(3):279–284. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.2307/2346968.

Huber, P. J. (2004). *Robust Statistics*. John Wiley & Sons. Google-Books-ID: e62RhdqIdMkC.

Hušková, M. (2013). Robust Change Point Analysis. In Becker, C., Fried, R., and Kuhnt, S., editors, *Robustness and Complex Data Structures*, pages 171–190. Springer, Berlin, Heidelberg.

Hušková, M. and Slabý, A. (2001). Permutation tests for multiple changes. *Kybernetika*, 37(5):605–622. Publisher: Institute of Information Theory and Automation AS CR.

Hunter, J. S. (1986). The Exponentially Weighted Moving Average. *Journal of Quality Technology*, 18(4):203–210. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/00224065.1986.11979014.

Inclán, C. and Tiao, G. C. (1994). Use of Cumulative Sums of Squares for Retrospective Detection of Changes of Variance. *Journal of the American Statistical Association*, 89(427):913–923. Publisher: Taylor & Francis.

Jackson, B., Scargle, J., Barnes, D., Arabhi, S., Alt, A., Gioumousis, P., Gwin, E., Sangtrakulcharoen, P., Tan, L., and Tsai, T. T. (2005). An algorithm for optimal partitioning of data on an interval. *IEEE Signal Processing Letters*, 12(2):105–108. Conference Name: IEEE Signal Processing Letters.

Jandhyala, V., Fotopoulos, S., MacNeill, I., and Liu, P. (2013). Inference for single and multiple change-points in time series. *Journal of Time Series Analysis*, 34(4):423–446.

Jeng, X. J., Cai, T. T., and Li, H. (2013). Simultaneous discovery of rare and common segment variants. *Biometrika*, 100(1):157–172.

Jirak, M. (2015). Uniform change point tests in high dimension. *The Annals of Statistics*, 43(6):2451–2483.

Joe, H. (2006). Generating random correlation matrices based on partial correlations. *Journal of Multivariate Analysis*, 97(10):2177–2189.

Kao, C., Trapani, L., and Urga, G. (2018). Testing for instability in covariance structures. *Bernoulli*, 24(1):740–771. Publisher: Bernoulli Society for Mathematical Statistics and Probability.

Killick, R., Fearnhead, P., and Eckley, I. A. (2012). Optimal Detection of Changepoints With a Linear Computational Cost. *Journal of the American Statistical Association*, 107(500):1590–1598.

Kirch, C., Muhsal, B., and Ombao, H. (2015). Detection of Changes in Multivariate Time Series With Application to EEG Data. *Journal of the American Statistical Association*, 110(511):1197–1216.

Kirch, C. and Tadjuidje Kamgaing, J. (2015). On the use of estimating functions in monitoring time series for change points. *Journal of Statistical Planning and Inference*, 161:25–49.

Knoblauch, J., Jewson, J. E., and Damoulas, T. (2018). Doubly Robust Bayesian Inference for Non-Stationary Streaming Data with β-Divergences. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31*, pages 64–75. Curran Associates, Inc.

Ko, S. I. M., Chong, T. T. L., and Ghosh, P. (2015). Dirichlet Process Hidden Markov Multiple Change-Point Model. *Bayesian Analysis*, 10(2):275–296. Publisher: International Society for Bayesian Analysis.

Kovács, S., Li, H., Bühlmann, P., and Munk, A. (2020). Seeded Binary Segmentation: A general methodology for fast and optimal change point detection. *arXiv:2002.06633 [stat.ME]*.

Lai, T. L. (1995). Sequential Changepoint Detection in Quality Control and Dynamical Systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(4):613–658.

Lee, S., Ha, J., Na, O., and Na, S. (2003). The Cusum Test for Parameter Change in Time Series Models. *Scandinavian Journal of Statistics*, 30(4):781–796. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-9469.00364.

Lee, S., Seo, M. H., and Shin, Y. (2016). The lasso for high dimensional regression with a possible change point. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, 78(1):193–210.

Leonardi, F. and Bühlmann, P. (2016). Computationally efficient change point detection for high-dimensional regression. *arXiv:1601.03704 [stat.ME]*. arXiv: 1601.03704.

Li, H., Munk, A., and Sieling, H. (2016). FDR-control in multiscale change-point segmentation. *Electronic Journal of Statistics*, 10(1):918–959. Publisher: The Institute of Mathematical Statistics and the Bernoulli Society.

Li, J., Xu, M., Zhong, P.-S., and Li, L. (2019). Change Point Detection in the Mean of High-Dimensional Time Series Data under Dependence. *arXiv:1903.07006 [stat.ME]*.

Li, S., Xie, Y., Dai, H., and Song, L. (2015). M-Statistic for Kernel Change-Point Detection. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28*, pages 3366–3374. Curran Associates, Inc.

Liu, B., Zhou, C., Zhang, X., and Liu, Y. (2020). A unified data-adaptive framework for high dimensional change point detection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(4):933–963. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/rssb.12375.

Liu, K., Zhang, R., and Mei, Y. (2017). Scalable SUM-Shrinkage Schemes for Distributed Monitoring Large-Scale Data Streams. *Statistica Sinica*, 29:1–22.

Liu, Y.-W. and Chen, H. (2020). A Fast and Efficient Change-point Detection Framework for Modern Data. *arXiv:2006.13450 [stat.ME]*. arXiv: 2006.13450.

Lorden, G. (1971). Procedures for reacting to a change in distribution. *The Annals of Mathematical Statistics*, 42(6):1897–1908.

Lorden, G. and Pollak, M. (2008). Sequential Change-Point Detection Procedures That are Nearly Optimal and Computationally Simple. *Sequential Analysis*, 27(4):476–512.

Lévy-Leduc, C. and Roueff, F. (2009). Detection and localization of change-points in high-dimensional network traffic data. *The Annals of Applied Statistics*, 3(2):637–662.

Maboudou-Tchao, E. M. and Hawkins, D. M. (2013). Detection of multiple change-points in multivariate data. *Journal of Applied Statistics*, 40(9):1979–1995. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/02664763.2013.800471.

Madrid Padilla, O. H., Athey, A., Reinhart, A., and Scott, J. G. (2019). Sequential Nonparametric Tests for a Change in Distribution: An Application to Detecting Radiological Anomalies. *Journal of the American Statistical Association*, 114(526):514–528. Publisher: Taylor & Francis.

Maeng, H. (2019). *Adaptive multiscale approaches to regression and trend segmentation*. PhD Thesis, The London School of Economics and Political Science.

Maeng, H. and Fryzlewicz, P. (2019). Detecting linear trend changes and point anomalies in data sequences. *arXiv:1906.01939 [stat.ME]*. arXiv: 1906.01939.

Maidstone, R. (2016). *Efficient analysis of complex changepoint problems*. PhD Thesis, Lancaster University.

Maidstone, R., Hocking, T., Rigaill, G., and Fearnhead, P. (2017). On optimal multiple changepoint algorithms for large data. *Statistics and Computing*, 27(2):519–533.

Matteson, D. S. and James, N. A. (2014). A Nonparametric Approach for Multiple Change Point Analysis of Multivariate Data. *Journal of the American Statistical Association*, 109(505):334–345. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01621459.2013.849605.

McDonald, D. (1990). A cusum procedure based on sequential ranks. *Naval Research Logistics*, 37(5):627–646. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/1520-6750%28199010%2937%3A5%3C627%3A%3AAID-NAV3220370504%3E3.0.CO%3B2-F.

Mei, Y. (2010). Efficient scalable schemes for monitoring a large number of data streams. *Biometrika*, 97(2):419–433.

Mendiratta, V., Liu, Z., Bhattacharjee, M., and Zhou, Y. (2019). Detecting and Diagnosing Anomalous Behavior in Large Systems with Change Detection Algorithms. In *2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, pages 47–52.

Moustakides, G. V. (1986). Optimal stopping times for detecting changes in distributions. *The Annals of Statistics*, 14(4):1379–1387.

Niu, Y. S., Hao, N., and Zhang, H. (2016). Multiple Change-Point Detection: A Selective Overview. *Statistical Science*, 31(4):611–623. Publisher: Institute of Mathematical Statistics.

Olshen, A. B., Venkatraman, E. S., Lucito, R., and Wigler, M. (2004). Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics*, 5(4):557–572.

Padilla, O. H. M., Yu, Y., Wang, D., and Rinaldo, A. (2020). Optimal nonparametric multivariate change point detection and localization. *arXiv:1910.13289 [math.ST]*. arXiv: 1910.13289.

Page, E. S. (1954). Continuous inspection schemes. *Biometrika*, 41(1/2):100–115.

Page, E. S. (1955). A test for a change in a parameter occurring at an unknown point. *Biometrika*, 42(3/4):523–527.

Pein, F., Sieling, H., and Munk, A. (2017). Heterogeneous change point inference. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, 79(4):1207–1227.

Pettitt, A. N. (1979). A Non-Parametric Approach to the Change-Point Problem. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(2):126–135. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.2307/2346729.

Pollak, M. and Siegmund, D. (1975). Approximations to the Expected Sample Size of Certain Sequential Tests. *The Annals of Statistics*, 3(6):1267–1282. Publisher: Institute of Mathematical Statistics.

Pollak, M. and Siegmund, D. (1991). Sequential detection of a change in a normal mean when the initial value is unknown. *Annals of Statistics*, 19(1):394–416. Publisher: Institute of Mathematical Statistics.

Polunchenko, A. S. and Tartakovsky, A. G. (2012). State-of-the-Art in Sequential Change-Point Detection. *Methodology and Computing in Applied Probability*, 14(3):649–684.

Preuss, P., Puchstein, R., and Dette, H. (2015). Detection of Multiple Structural Breaks in Multivariate Time Series. *Journal of the American Statistical Association*, 110(510):654–668. Publisher: Taylor & Francis.

Rigaill, G. (2010). Pruned dynamic programming for optimal multiple change-point detection. *arXiv:1004.0887 [stat.CO]*.

Roberts, S. W. (1966). A Comparison of Some Control Chart Procedures. *Technometrics*, 8(3):411–430. Publisher: Taylor & Francis.

Romano, G., Rigaill, G., Runge, V., and Fearnhead, P. (2020). Detecting Abrupt Changes in the Presence of Local Fluctuations and Autocorrelated Noise. *arXiv:2005.01379 [stat.ME]*. arXiv: 2005.01379.

Roy, S., Atchadé, Y., and Michailidis, G. (2017). Change point estimation in high dimensional Markov random-field models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(4):1187–1206.

Ruggieri, E. and Antonellis, M. (2016). An exact approach to Bayesian sequential change point detection. *Computational Statistics & Data Analysis*, 97:71–86.

Safikhani, A. and Shojaie, A. (2020). Joint Structural Break Detection and Parameter Estimation in High-Dimensional Nonstationary VAR Models. *Journal of the American Statistical Association*. Publisher: Taylor & Francis.

Schwarz, G. (1978). Estimating the Dimension of a Model. *Annals of Statistics*, 6(2):461–464. Publisher: Institute of Mathematical Statistics.

Schweder, T. and Hjort, N. L. (2016). *Confidence, Likelihood, Probability*. Cambridge University Press. Google-Books-ID: vPx9CwAAQBAJ.

Scott, A. J. and Knott, M. (1974). A Cluster Analysis Method for Grouping Means in the Analysis of Variance. *Biometrics*, 30(3):507–512. Publisher: [Wiley, International Biometric Society].

Shewhart, W. A. (1925). The Application of Statistics as an Aid in Maintaining Quality of a Manufactured Product. *Journal of the American Statistical Association*, 20(152):546–548. Publisher: Taylor & Francis _eprint: https://www.tandfonline.com/doi/pdf/10.1080/01621459.1925.10502930.

Shiryaev, A. N. (1963). On Optimum Methods in Quickest Detection Problems. *Theory of Probability & Its Applications*, 8(1):22–46. Publisher: Society for Industrial and Applied Mathematics.

Siegmund, D. (1985). *Sequential Analysis: Tests and Confidence Intervals*. Springer, New York, USA.

Siegmund, D. and Venkatraman, E. S. (1995). Using the generalized likelihood ratio statistic for sequential detection of a change-point. *The Annals of Statistics*, 23(1):255–271.

Siegmund, D. and Yakir, B. (2008). Detecting the emergence of a signal in a noisy image. *Statistics and Its Interface*, 1(1):3–12.

Siegmund, D., Yakir, B., and Zhang, N. R. (2011). Detecting simultaneous variant intervals in aligned sequences. *The Annals of Applied Statistics*, 5(2A):645–668. Publisher: Institute of Mathematical Statistics.

Srivastava, M. S. and Worsley, K. J. (1986). Likelihood Ratio Tests for a Change in the Multivariate Normal Mean. *Journal of the American Statistical Association*, 81(393):199–204. Publisher: Taylor & Francis _eprint: https://www.tandfonline.com/doi/pdf/10.1080/01621459.1986.10478260.

Sullivan, J. H. and Woodall, W. H. (2000). Change-point detection of mean vector or covariance matrix shifts using multivariate individual observations. *IIE Transactions*, 32(6):537–549.

Tartakovsky, A. G., Rozovskii, B. L., Blažek, R. B., and Kim, H. (2006). Detection of intrusions in information systems by sequential change-point methods. *Statistical Methodology*, 3(3):252–293.

Tibshirani, R. (1996). Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.2517-6161.1996.tb02080.x.

Tickle, S. (2020). *Changepoint detection for data intensive settings*. PhD Thesis, Lancaster University.

Tickle, S. O., Eckley, I. A., Fearnhead, P., and Haynes, K. (2020). Parallelization of a Common Changepoint Detection Method. *Journal of Computational and Graphical Statistics*, 29(1):149–161. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/10618600.2019.1647216.

Truong, C., Oudre, L., and Vayatis, N. (2020). Selective review of offline change point detection methods. *Signal Processing*, 167:107299.

Venkatraman, E. S. (1993). *Consistency results in multiple change-point problems*. PhD Thesis, Stanford University.

Vostrikova, L. J. (1981). Detecting disorder in multidimensional random processes. *Soviet Mathematics Doklady*, 24:55–59.

Wang, D., Yu, Y., and Rinaldo, A. (2018). Optimal Covariance Change Point Localization in High Dimension. *arXiv:1712.09912 [math.ST]*. arXiv: 1712.09912.

Wang, D., Yu, Y., and Rinaldo, A. (2020a). Univariate mean change point detection: Penalization, CUSUM and optimality. *Electronic Journal of Statistics*, 14(1):1917–1961. Publisher: The Institute of Mathematical Statistics and the Bernoulli Society.

Wang, D., Yu, Y., Rinaldo, A., and Willett, R. (2020b). Localizing Changes in High-Dimensional Vector Autoregressive Processes. *arXiv:1909.06359 [math.ST]*. arXiv: 1909.06359.

Wang, R. and Shao, X. (2020). Dating the Break in High-dimensional Data. *arXiv:2002.04115 [math.ST]*. arXiv: 2002.04115.

Wang, R., Volgushev, S., and Shao, X. (2019). Inference for Change Points in High Dimensional Data. *arXiv:1905.08446 [math.ST]*. arXiv: 1905.08446.

Wang, T. and Samworth, R. J. (2018). High dimensional change point estimation via sparse projection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(1):57–83.

Westerlund, J. (2019). Common Breaks in Means for Cross-Correlated Fixed-T Panel Data. *Journal of Time Series Analysis*, 40(2):248–255.

Worsley, K. J. (1986). Confidence regions and tests for a change-point in a sequence of exponential family random variables. *Biometrika*, 73(1):91–104. Publisher: Oxford Academic.

Xie, L., Moustakides, G. V., and Xie, Y. (2018). First-Order Optimal Sequential Subspace Change-Point Detection. In *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 111–115.

Xie, Y. and Siegmund, D. (2013). Sequential multi-sensor change-point detection. *The Annals of Statistics*, 41(2):670–692.

Yeh, A. B., Lin, D. K. J., and McGrath, R. N. (2005). Multivariate Control Charts for Monitoring Covariance Matrix: A Review. *Quality Technology & Quantitative Management*, 3(4):415–436.

Zhang, N. R. and Siegmund, D. O. (2007). A Modified Bayes Information Criterion with Applications to the Analysis of Comparative Genomic Hybridization Data. *Biometrics*, 63(1):22–32. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1541-0420.2006.00662.x.

Zhang, N. R., Siegmund, D. O., Ji, H., and Li, J. Z. (2010). Detecting simultaneous changepoints in multiple sequences. *Biometrika*, 97(3):631–645. Publisher: Oxford Academic.

Zhang, W., James, N. A., and Matteson, D. S. (2017). Pruning and Nonparametric Multiple Change Point Detection. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 288–295. ISSN: 2375-9259.

Zhao, Z., Chen, L., and Lin, L. (2019). Change-point detection in dynamic networks via graphon estimation. *arXiv:1908.01823 [stat.ME]*. arXiv: 1908.01823.

Zhao, Z. and Yau, C. Y. (2019). Alternating Pruned Dynamic Programming for Multiple Epidemic Change-Point Estimation. *arXiv:1907.06810 [stat.ME]*, 1907.

Zou, C., Wang, Z., Zi, X., and Jiang, W. (2014a). An Efficient Online Monitoring Method for High-Dimensional Data Streams. *Technometrics*, 57(3):374–387.

Zou, C., Yin, G., Feng, L., and Wang, Z. (2014b). Nonparametric maximum likelihood approach to multiple change-point problems. *Annals of Statistics*, 42(3):970–1002. Publisher: Institute of Mathematical Statistics.

Papers

Paper I

Which principal components are most sensitive in the change detection problem?

Martin Tveten

Published in *Stat*, December 2019, volume 8, issue e252, DOI: 10.1002/sta4.252.

Received: 12 August 2019 | Revised: 6 September 2019 | Accepted: 12 September 2019
DOI: 10.1002/sta4.252

WILEY

ORIGINAL ARTICLE

Which principal components are most sensitive in the change detection problem?

Martin Tveten

Department of Mathematics, University of Oslo, Oslo, Norway

Correspondence
Martin Tveten, Department of Mathematics, University of Oslo, Niels Henrik Abels hus, Moltke Moes vei 35, 0851 Oslo, Norway.
Email: martintv@math.uio.no

Funding information
Norges Forskningsråd, Grant/Award Number: 237718

Principal component analysis (PCA) is often used in anomaly detection and statistical process control tasks. For bivariate normal data, we prove that the minor projection (the least varying projection) of the PCA-rotated data is the most sensitive to distributional changes, where sensitivity is defined as the Hellinger distance between the projections' marginal distributions before and after a change. In particular, this is almost always the case if only one parameter of the bivariate normal distribution changes, that is, the change is sparse. Simulations indicate that the minor projections are the most sensitive for a large range of changes and pre-change settings in higher dimensions as well, including changes that are very sparse. This motivates using only a few of the minor projections for detecting sparse distributional changes in high-dimensional data.

KEYWORDS
machine learning, quality control, statistical process control

1 | INTRODUCTION

It is popular to use principal component analysis (PCA) for anomaly detection and stochastic process control (SPC). Using PCA in SPC goes back to the work of Jackson and Morris (1957) and Jackson and Mudholkar (1979), and its various extensions (see Ketelaere et al., 2015 and Rato et al., 2016, for an overview) have been successfully applied to many real data situations. Within the machine learning literature on anomaly detection, Mishin et al. (2014) use PCA for temperature monitoring at Johns Hopkins, Harrou et al. (2015) apply PCA-based anomaly detection to find segments with abnormal rates of patient arrivals at an emergency department, and Camacho et al. (2016) relate PCA-based monitoring in SPC to modern anomaly detection in statistical networks. PCA has also been studied in the setting of change detection in multivariate functional data with the aim of detecting faulty profiles in a forging manufacturing process (Wang et al., 2018). Pimentel et al. (2014) provide an extensive review of novelty detection techniques and applications, and it is pointed to PCA being very useful for detecting outliers in this setting, for a large range of real world examples, covering industrial monitoring, video surveillance, text mining, sensor networks, and IT security. Moreover, many authors (Huang et al., 2007; Lakhina et al., 2004; Pimentel et al., 2014) acknowledge that it is most often the residual subspace of PCA that is most useful for outlier detection. On a similar note, Kuncheva and Faithfull (2014) offer an interesting alternative way to use PCA for change detection problems.

Most PCA-based methods utilize PCA in the intended way of creating a model based on retaining a small number of the most varying projections onto eigenvectors of the covariance matrix. As a consequence, the data are split into a model subspace that explains most of the variance in the data and a residual subspace. It is not self-evident that this is the best way to use PCA as a dimension reduction tool for change detection, so Kuncheva and Faithfull (2014) pose the question of which projections are the most sensitive to distributional changes in the data. Sensitivity is measured by a statistical divergence between the marginal distributions of projections before and after a change. They give a brief two-dimensional theoretical example that motivates monitoring the minor projections (the least varying projections) to detect anomalies that manifest in the form of sustained changes in the distribution of the data. An important feature of such an approach is that it can potentially be used to choose a subspace based on criteria linked to change detection, rather than on retaining data variance, hopefully yielding a better change and anomaly detection methods. The goal of this article is to give a more complete treatment of and extend the bivariate problem of Kuncheva and Faithfull (2014) in order to better understand the projections' sensitivity to changes under a simple setup and then study how these results carry over to higher dimensions by simulations.

There are three main differences between our approach and the approach of Kuncheva and Faithfull (2014). First, we express the projections' sensitivity to changes as functions of the parameters of the original data rather than of the parameters of the projections. The reason for this choice is that the original data are the object of the main interest, whereas the projections are ancillary. Our approach allows one to change individual parameters of the original data independently and see how this affects the marginal distributions of the projections as a consequence. We argue that this is more informative. Second, we study a much larger space of possible changes, including changes in only one parameter at a time. Such change scenarios where only a few of the dimensions change are called *sparse changes*, and they are the subject of much current interest (Chan, 2017; Liu et al., 2017; Wang et al., 2018; Wang & Samworth, 2018; Xie & Siegmund, 2013). Third, we measure sensitivity by the normal Hellinger distance between the marginal distributions of projections before and after a change, whereas Kuncheva and Faithfull (2014) use the normal Bhattacharyya distance. See Section 2 for an explanation of this choice.

In short, we find the following. For bivariate data, we prove that if only one of the two components' means changes in any direction, one component's variance increases, or the correlation between the components changes, the minor projection is the most sensitive. The principal projection is the most sensitive if one of the components' variance decreases and the correlation is not too close to 1. Lastly, if both means change, which projection is the most sensitive depends on the relative directions and sizes of change, and when both variances change by an equal amount, both projections are equally sensitive. Thus, on average (with all change scenarios up to a certain size equally likely), the minor projection is the most sensitive, mainly due to the sparse change scenarios. Our simulations confirm that the trend of the minor projections being more sensitive on average also holds for higher dimensions. Moreover, and most importantly, the minor projections seem to be quite sensitive even to very sparse changes. This knowledge carries large potential for creating more efficient change or anomaly detection methods.

The rest of the article is organized as follows: Section 2 formulates the problem precisely, Section 3 contains the theoretical results about sensitivity to changes in two dimensions, and in Section 4, we explore sensitivity in higher dimensions by simulations. The proofs are found in Appendix A.

2 | PROBLEM FORMULATION

Consider independent observations $x_t \in \mathbb{R}^D$, $t = 1, \dots, n$, and let $\kappa \in \{1, \dots, n-1\}$ be a change-point. For $t \leq \kappa$, the observations have mean $\boldsymbol{\mu_0}$ and covariance matrix Σ_0, whereas for $t > \kappa$, the data have mean μ_1 and covariance matrix Σ_1. Assume without loss of generality that the data are standardized with respect to the pre-change parameters, so that $\boldsymbol{\mu_0} = \mathbf{0}$ and Σ_0 is a correlation matrix with correlation parameter ρ. For $D = 2$, the changed mean is given by $\boldsymbol{\mu_1} = (\mu_1, \mu_2)^t$, and the changed covariance matrix can be expressed in terms of Σ_0 and parameter-wise multiplicative change factors as

$$\Sigma_1 = \begin{pmatrix} a_{11}^2 & a_{11}a_{22}a_{12}\rho \\ a_{11}a_{22}a_{12}\rho & a_{22}^2 \end{pmatrix},$$

where

$$-1 < \rho, a_{12}\rho < 1 \text{ and } \rho \neq 0. \tag{1}$$

For example, if $a_{11} = 2$, it means that the standard deviation of the first component has doubled compared with what it was originally in Σ_0. Similarly, $a_{12} = 0.5$ means that the correlation is half as strong after the change. Note that we exclude the degenerate cases of correlations equal to -1 and 1.

Next, let $\{\lambda_j, v_j\}_{j=1}^D$ be the normalized eigensystem of Σ_0, ordered by $\lambda_1 \geq \dots \geq \lambda_D$. The orthogonal projections $y_{j,t} = v_j^t x_t$, with progressively decreasing variances λ_j, are our main objects of interest.

The general problem is to find out which of the D projections are the most sensitive to different distributional changes defined by $(\boldsymbol{\mu_1}, \Sigma_1)$, for each pre-change correlation matrix Σ_0. In the bivariate case, $(\Sigma_0, \boldsymbol{\mu_1}, \Sigma_1)$ is fully specified by $(\rho, \mu_1, \mu_2, a_{11}, a_{12}, a_{22})$. Note that a collection of the most and least varying $y_{j,t}$'s is referred to as the *principal projections* and *minor projections*, respectively.

We define sensitivity to changes as the normal Hellinger distance between the marginal distribution of a projection before and after a change. The squared Hellinger distance between two normal distributions $p(x) = N(x|\xi_1, \sigma_1^2)$ and $q(x) = N(x|\xi_2, \sigma_2^2)$ is given by

$$H^2(p, q) = 1 - \sqrt{\frac{2\sigma_1\sigma_2}{\sigma_1^2 + \sigma_2^2}} \exp\left\{-\frac{1}{4}\frac{(\xi_1 - \xi_2)^2}{\sigma_1^2 + \sigma_2^2}\right\}.$$

The formal definition of sensitivity to changes is contained in Definition 1.

Definition 1. For $j = 1, \dots, D$, let p_j and q_j denote the marginal pre- and post-change density functions of $y_{j,t}$, respectively, given by

$$p_j(y) = N(y \mid v_j^T \mu_0, v_j^T \Sigma_0 v_j) = N(y|0, \lambda_j),$$
$$q_j(y) = N(y \mid v_j^T \mu_1, v_j^T \Sigma_1 v_j).$$

The sensitivity of the jth projection based on Σ_0 to the change specified by $(\boldsymbol{\mu_1}, \Sigma_1)$ is defined as $H(p_j, q_j)$, abbreviated by H_j or $H_j(\Sigma_0, \boldsymbol{\mu_1}, \Sigma_1)$.

Our aim in the next section is to determine which pre-change parameters and changes the inequality $H_2 > H_1$ holds for when $D = 2$ in light of Definition 1.

Remark

(i) Kuncheva and Faithfull (2014) also define sensitivity as a divergence between distributions before and after a change but use the Bhattacharyya distance. The closely related Hellinger distance was chosen here because it turns out to be simpler to prove the sensitivity propositions because of Lemma 1 (see Appendix A). It is also an advantageous feature of the Hellinger distance that it is a true metric and takes values in $[0, 1]$. That it is a true metric implies for instance that a change in variance from 1 to $a > 1$ is an equally large change as from 1 to $1/a$ for the normal distribution. We find this an appealing feature because it is also a property of the generalized likelihood ratio test for a change in the mean and/or variance of normal data (see Hawkins & Zamba, 2005, for the corresponding test statistic).

(ii) One of the differences between our approach and the work of Kuncheva and Faithfull (2014) can now be stated more precisely. Our aim is to study the sensitivity of the $y_{j,t}$'s as functions of parameters of the original data x_t. Kuncheva and Faithfull (2014), on the other hand, study (additive) changes in the parameters of y_t directly; for instance, λ_j changing to $\lambda_j + a$ for all j, but without relating this a back to which Σ_1's this change corresponds to.

3 | BIVARIATE RESULTS

This section contains all the bivariate results about sensitivity to changes. The detailed proofs are given in Appendix A.

For changes in the mean in two-dimensional data, Proposition 1 gives the condition for determining which projection is the most sensitive, as well as the results for some special cases.

Proposition 1. *Let $a_{11} = a_{22} = a_{12} = 1$ and $\mu_1, \mu_2 \in \mathbb{R}$ while not both being 0 simultaneously (only the mean changes). $H_2 > H_1$ if and only if $(\mu_1 - \mu_2)^2/(\mu_1 + \mu_2)^2 > (1 - |\rho|)/(1 + |\rho|)$.*

In particular, for all $|\rho| \in (0, 1)$,

1. *$H_2 > H_1$ if one of μ_1 and μ_2 is 0 whereas the other is not (one mean changes).*
2. *$H_2 > H_1$ if $\mu_1 = -\mu_2 = \mu \neq 0$ (equal changes in opposite directions).*
3. *$H_2 < H_1$ if $\mu_1 = \mu_2 = \mu \neq 0$ (equal changes in the same direction).*

When both variances change by the same amount, Proposition 2 tells us that both projections are equally sensitive no matter what the pre-change correlation or size of the change is.

Proposition 2. *Let $\mu_1 = \mu_2 = 0$, $a_{12} = 1$ and $a_{11} = a_{22} = a \neq 1$ (both variances change equally). For any $|\rho| \in (0, 1)$ and $a > 0$, $H_2 = H_1$.*

The picture becomes more complicated when only one variance changes (Proposition 3). If the variance increases, the minor projection is always the most sensitive. On the other hand, if the variance decreases, the principal projection is mostly the most sensitive but not always if the pre-change correlation is high (greater than $\sqrt{3}/2$). In total, this gives a slight edge to the minor projection.

Proposition 3. *Let $\mu_1 = \mu_2 = 0$, $a_{12} = 1$, and either $a_{11} = 1$ and $a_{22} = a \neq 1$, or $a_{11} = a$ and $a_{22} = 1$, where $a > 0$ (one variance changes).*

1. *For any $|\rho| \in (0, 1)$ and $a > 1$ (variance increase), $H_2 > H_1$.*
2. *When $|\rho| \in (0, 1)$ and $a \in (0, 1)$ (variance decrease), $H_2 < H_1$ in most cases. The only exception is if $|\rho| \in (\sqrt{3}/2, 1)$ and $a \in (0, \sqrt{4\rho^2 - 3})$, where $H_2 > H_1$.*

Finally, for a change in correlation, the minor projection is the most sensitive in most cases (Proposition 4). Only if the correlation changes direction and becomes stronger is the principal projection more sensitive.

Proposition 4. *Let $\mu_1 = \mu_2 = 0$, $a_{11} = a_{22} = 1$ and $a_{12} = a \neq 1$ such that (1) holds (the correlation changes). Then $H_2 > H_1$ for any $|\rho| \in (0, 1)$ and $a > -1$.*

4 | EXPLORING HIGHER DIMENSIONS

In the two-dimensional case, we saw that which projection is the most sensitive depends both on the change $(\boldsymbol{\mu_1}, \Sigma_1)$ and on the pre-change correlation matrix Σ_0. For a higher dimension D, solving inequalities like above for all the parameters in $(\Sigma_0, \boldsymbol{\mu_1}, \Sigma_1)$ quickly becomes tedious and uninformative. Therefore, we use simulation to obtain Monte Carlo estimates $E[H_j(\Sigma_0, \boldsymbol{\mu_1}, \Sigma_1)]$ instead, where we vary which parameters that change, the size of the changes, and the sparsity of the change (the number of dimensions that change). Let $\rho_{i,d}$ for $i \neq d$ denote the off-diagonal elements of Σ_0, μ_d be the dth element of $\boldsymbol{\mu_1}$, and σ_d be the dth diagonal element of Σ_1. Then our simulation protocol to get such estimates is as follows:

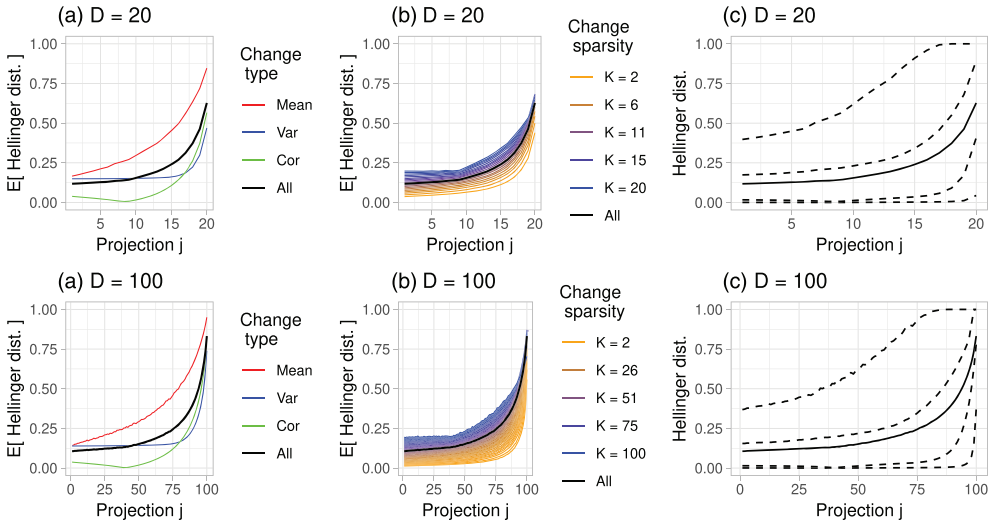

FIGURE 1 A summary of the sensitivity results obtained by the simulation protocol for $D = 20$ for $D = 100$. (a) Monte Carlo estimates of $E[H_j]$ for uniformly drawn changes in the mean, variance, and (decreases in) correlation, as well as uniformly drawn pre-change correlation matrices Σ_0. (b) Same as (a), but now the average sensitivity is conditional on the sparsity of the change, rather than the type of parameter. (c) 0.05, 0.25, 0.75, and 0.95 percentiles (the dashed lines, from bottom to top) of the distribution of H_j, together with $E[H_j]$ (solid line). Note that the percentiles are over Σ_0, $\boldsymbol{\mu_1}$, and Σ_1 simultaneously

1. Draw a correlation matrix Σ_0 uniformly from the space of correlation matrices by the method of Joe (2006) (clusterGeneration::rcorrmatrix in R).
2. Draw a change sparsity $K \sim \text{Unif}\{2, \dots, D\}$.
3. Draw a random subset $\mathcal{D} \subseteq \{1, \dots, D\}$ of size K.
4. Draw an additive change in mean $\mu \sim \text{Unif}(-3, 3)$, and set $\mu_d = \mu$ for $d \in \mathcal{D}$, whereas $\Sigma_1 = \Sigma_0$.
5. Draw a multiplicative change in standard deviation $\sigma \sim \frac{1}{2}\text{Unif}(1/3, 1) + \frac{1}{2}\text{Unif}(1, 3)$ (equal probability of decrease and increase in standard deviation) and set $\sigma_d = \sigma$ for $d \in \mathcal{D}$, keeping the remaining parameters constant.
6. Draw a multiplicative change in correlation $a \sim \text{Unif}(0, 1)$ and change $\rho_{i,d}$ to $a\rho_{i,d}$ for all $i \neq d \in \mathcal{D}$. The other parameters are kept constant.
7. For each of the three change scenarios 4–6, calculate $H_j(\Sigma_0, \boldsymbol{\mu_1}, \Sigma_1)$, $j = 1, \dots, D$.
8. Repeat 2–7 10^3 times.
9. Repeat 1–8 10^3 times.

Averaging the simulated H_js yields estimates of $E[H_j]$, and we can condition on the type of parameter that changes and the change sparsity to see what the sensitivity is expected to be for different classes of changes. (Note that we only consider decreases in correlation. This is to avoid getting too many indefinite Σ_1's. If indefinite Σ_1's still occur, we find the closest positive-definite one by Higham's algorithm (Higham, 2002), implemented in the Matrix::nearPD-function in R.

Figure 1 shows that the trend of the minor components being the most sensitive continues for $D = 20$ and $D = 100$. This holds for changes in the mean, variance, and correlation (a) as well as all the different change sparsities (b). From the quantile plots (c), however, observe that a lot of variation is hidden in these averages, meaning that which projection is the most sensitive will depend on the specific Σ_0 and change $(\boldsymbol{\mu_1}, \Sigma_1)$, as in the bivariate case.

5 | CONCLUDING REMARKS

We have presented bivariate theory demonstrating that the minor projection of PCA-rotated data is usually the most sensitive to changes, especially if the change is sparse. Simulations confirm this to be the case on average for higher dimensions as well, but, in general, the sensitivity strongly varies with the pre-change correlation matrix and the specific change.

In future work, we aim to exploit these insights for creating computationally efficient change detection methods for high-dimensional data. The most promising and surprising part of our results is that even very sparse changes seem to be quite noticeable in the minor projections. This is important for change detection in high-dimensional data because a change rarely affects all dimensions or parameters at once. Most often, only a few parameters among many will change, and therefore, the problem of sparse changes will be the most relevant. One interpretation of

the results presented here is that for detecting sparse changes in the mean vector and/or covariance matrix of a high-dimensional data set or of a sequentially arriving data stream, it is potentially sufficient to search for changes in a few selected minor projections. This might lead to major improvements, not only computationally but also in terms of detection accuracy or speed. Choosing which minor projections to use for a specific change detection problem is the subject of ongoing work.

This work is funded by the Norwegian Research Council centre Big Insight, Project 237718. The author would also like to thank Ingrid Glad for useful input on the presentation of the material.

SUPPORTING INFORMATION AND DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available as part of the supporting information for this online article: **R code** .R file with the code for reproducing (and easily extending) the simulation study and Figure 1.

ORCID

Martin Tveten https://orcid.org/0000-0002-4236-633X

REFERENCES

Camacho, J., Pérez-Villegas, A., García-Teodoro, P., & Maciá-Fernández, G. (2016). PCA-based multivariate statistical network monitoring for anomaly detection. *Computers & Security*, *59*, 118–137. https://doi.org/10.1016/j.cose.2016.02.008

Chan, H. P. (2017). Optimal sequential detection in multi-stream data. *The Annals of Statistics*, *45*(6), 2736–2763. https://doi.org/10.1214/17-AOS1546

Harrou, F., Kadri, F., Chaabane, S., Tahon, C., & Sun, Y. (2015). Improved principal component analysis for anomaly detection: Application to an emergency department. *Computers & Industrial Engineering*, *88*, 63–77. https://doi.org/10.1016/j.cie.2015.06.020

Hawkins, D. M., & Zamba, K. D (2005). Statistical process control for shifts in mean or variance using a changepoint formulation. *Technometrics*, *47*(2), 164–173. https://doi.org/10.1198/004017004000000644

Higham, N. J. (2002). Computing the nearest correlation matrix—A problem from finance. *IMA Journal of Numerical Analysis*, *22*(3), 329–343. https://doi.org/10.1093/imanum/22.3.329

Huang, L., Nguyen, X., Garofalakis, M., Jordan, M. I., Joseph, A., & Taft, N. (2007). In-network PCA and anomaly detection. In Schölkopf, B., Platt, J. C., & Hoffman, T. (Eds.), *Advances in Neural Information Processing Systems 19*. MA, USA: MIT Press, pp. 617–624.

Jackson, J. E., & Morris, R. H. (1957). An application of multivariate quality control to photographic processing. *Journal of the American Statistical Association*, *52*(278), 186–199.

Jackson, J. E., & Mudholkar, G. S. (1979). Control procedures for residuals associated with principal component analysis. *Technometrics*, *21*(3), 341–349. https://doi.org/10.1080/00401706.1979.10489779

Joe, H. (2006). Generating random correlation matrices based on partial correlations. *Journal of Multivariate Analysis*, *97*(10), 2177–2189. https://doi.org/10.1016/j.jmva.2005.05.010

Ketelaere, B. D., Hubert, M., & Schmitt, E. (2015). Overview of PCA-based statistical process-monitoring methods for time-dependent, high-dimensional data. *Journal of Quality Technology*, *47*(4), 318–335. https://doi.org/10.1080/00224065.2015.11918137

Kuncheva, L. I., & Faithfull, W. J. (2014). PCA Feature Extraction for Change Detection in Multidimensional Unlabeled Data. *IEEE transactions on neural networks and learning systems*, *25*(1), 69–80. https://doi.org/10.1109/TNNLS.2013.2248094

Lakhina, A., Crovella, M., & Diot, C. (2004). Diagnosing network-wide traffic anomalies. In *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, ACM, New York, USA, pp. 219–230. https://doi.org/10.1145/1015467.1015492

Liu, K., Zhang, R., & Mei, Y. (2017). Scalable SUM-shrinkage schemes for distributed monitoring large-scale data streams. *Statistica Sinica*, *29*, 1–22. https://doi.org/10.5705/ss.202015.0316

Mishin, D., Brantner-Magee, K., Czako, F., & Szalay, A. S. (2014). Real time change point detection by incremental PCA in large scale sensor data. In *2014 IEEE High Performance Extreme Computing Conference (HPEC)*, pp. 1–6. https://doi.org/10.1109/HPEC.2014.7040959

Pimentel, M. A. F., Clifton, D. A., Clifton, L., & Tarassenko, L. (2014). A review of novelty detection. *Signal Processing*, *99*, 215–249. https://doi.org/10.1016/j.sigpro.2013.12.026

Rato, T., Reis, M., Schmitt, E., Hubert, M., & De Ketelaere, B. (2016). A systematic comparison of PCA-based statistical process monitoring methods for high-dimensional, time-dependent processes. *AIChE Journal*, *62*(5), 1478–1493. https://doi.org/10.1002/aic.15062

Wang, Y., Mei, Y., & Paynabar, K. (2018). Thresholded multivariate principal component analysis for phase I multichannel profile monitoring. *Technometrics*, *60*(3), 360–372. https://doi.org/10.1080/00401706.2017.1375993

Wang, T., & Samworth, R. J. (2018). High dimensional change point estimation via sparse projection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *80*(1), 57–83. https://doi.org/10.1111/rssb.12243

Xie, Y., & Siegmund, D. (2013). Sequential multi-sensor change-point detection. *The Annals of Statistics*, *41*(2), 670–692. https://doi.org/10.1214/13-AOS1094

How to cite this article: Tveten M. Which principal components are most sensitive in the change detection problem?. *Stat*. 2019;8:e252. https://doi.org/10.1002/sta4.252

APPENDIX A: PROOFS

Before turning to the proofs of the propositions in Section 3, the expressions for the pre- and post-change means and variances of each projection are needed. The normalized eigenvectors (principal axes) and corresponding eigenvalues (variance in the data along a given principal axis) of Σ_0 are quickly verified to be

$$\begin{aligned} \lambda_1 &= 1+\rho, \quad \mathbf{v}_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \\ \lambda_2 &= 1-\rho, \quad \mathbf{v}_2 = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ 1 \end{pmatrix}. \end{aligned} \tag{A1}$$

Note that which principal axis is the dominant one depends on the sign of ρ. If ρ is positive, $\mathbf{v}_1$ is the dominant one, but $\mathbf{v}_2$ is dominant if ρ is negative.

From the projections in (A1), the parameters of the projections before and after a change can be expressed as functions of the original correlation matrix and multiplicative change factors. For the principal component, the original and changed variances become as follows, respectively:

$$\begin{aligned} o_1^2 &= 1+\rho, \\ c_1^2 &= \frac{1}{2}a_{11}^2 + \frac{1}{2}a_{22}^2 + a_{11}a_{22}a_{12}\rho. \end{aligned} \tag{A2}$$

The expressions for the variances of the minor component are identical up to one switched sign:

$$\begin{aligned} o_2^2 &= 1-\rho, \\ c_2^2 &= \frac{1}{2}a_{11}^2 + \frac{1}{2}a_{22}^2 - a_{11}a_{22}a_{12}\rho. \end{aligned} \tag{A3}$$

Observe that if $\rho < 0$, then o_2 and c_2 would be equal to o_1 and c_1 with positive ρ, and vice versa. Thus, for $\rho \in (-1, 1)$, the general expressions are obtained by replacing ρ with $|\rho|$. Lastly, the changed mean components are given by

$$\begin{aligned} m_1 &= \frac{1}{\sqrt{2}}(\mu_1 + \mu_2), \\ m_2 &= \frac{1}{\sqrt{2}}(\mu_1 - \mu_2). \end{aligned} \tag{A4}$$

We first prove Proposition 1 for changes in the mean.

Proof of Proposition 1. Let $p_1(x) = N(x \mid 0, o_1^2)$, $q_1(x) = N(x \mid m_1, o_1^2)$, $p_2(x) = N(x \mid 0, o_2^2)$, and $q_2(x) = N(x \mid m_2, o_2^2)$, where m_i, o_i are as in (A2), (A3), and (A4), with ρ replaced by $|\rho|$ as noted above. The Hellinger distances between the distributions before and after a change along each principal axis are given by for $j = 1, 2$

$$H_j^2 = H^2(p_j, q_j) = 1 - \exp\left\{-\frac{1}{8o_j^2}m_j^2\right\}.$$

Then some algebra results in the inequality we needed to prove:

$$\begin{aligned} H_2 &> H_1 \\ \frac{1}{8(1-|\rho|)}\frac{(\mu_1-\mu_2)^2}{2} &> \frac{1}{8(1+|\rho|)}\frac{(\mu_1+\mu_2)^2}{2} \\ \frac{(\mu_1-\mu_2)^2}{(\mu_1+\mu_2)^2} &> \frac{1-|\rho|}{1+|\rho|} \end{aligned}$$

From this inequality, the three special cases (i), (ii), and (iii) are immediately given. □

In the proofs concerning changes in the covariance matrix, we will make use of the following lemma. It reduces the inequality of Hellinger distances to a simpler inequality of ratios of variances.

Lemma 1. *Let p_1, q_1, p_2, q_2 be 0-mean normal distribution functions with variances $\sigma_{p_1}^2, \sigma_{q_1}^2, \sigma_{p_2}^2$, and $\sigma_{q_2}^2$, respectively. Furthermore, let*

$$\log r_j = \left| \log \frac{\sigma_{q_j}^2}{\sigma_{p_j}^2} \right|, \quad j = 1, 2.$$

Then $H(p_2, q_2) > H(p_1, q_1)$ if and only if $\log r_2 > \log r_1$.

Proof. First observe that when the means are 0, then we can write the Hellinger distance between two normal distributions as the following.

$$\begin{aligned} H^2(p, q) &= 1 - \left(\frac{2\sigma_p \sigma_q}{\sigma_p^2 + \sigma_q^2} \right)^{1/2} \\ &= 1 - \sqrt{2} \left(\frac{\sigma_p}{\sigma_q} + \frac{\sigma_q}{\sigma_p} \right)^{-1/2} \\ &= 1 - \sqrt{2} \left(\frac{\sigma_p^2}{\sigma_q^2} + \frac{\sigma_q^2}{\sigma_p^2} + 2 \right)^{-1/4}. \end{aligned}$$

This gives us the inequality

$$\begin{aligned} &H(p_2, q_2) > H(p_1, q_1) \\ &\frac{\sigma_{p_2}^2}{\sigma_{q_2}^2} + \frac{\sigma_{q_2}^2}{\sigma_{p_2}^2} > \frac{\sigma_{p_1}^2}{\sigma_{q_1}^2} + \frac{\sigma_{q_1}^2}{\sigma_{p_1}^2}. \end{aligned}$$

By setting $r_2 = \sigma_{p_2}^2 / \sigma_{q_2}^2$ and $r_1 = \sigma_{p_1}^2 / \sigma_{q_1}^2$, the inequality can be written as

$$r_2 + r_2^{-1} > r_1 + r_1^{-1}.$$

Now assume first that $r_1, r_2 > 1$, that is, $\sigma_{p_j}^2 > \sigma_{q_j}^2$. Then we see that

$$\begin{aligned} r_2 + r_2^{-1} &> r_1 + r_1^{-1} \\ r_2 - r_1 + \frac{r_1 - r_2}{r_1 r_2} &> 0 \\ (r_2 - r_1) \left(1 - \frac{1}{r_1 r_2} \right) &> 0. \end{aligned}$$

By the assumption that $r_1, r_2 > 1$, this inequality holds if and only if $r_2 > r_1$.

Finally, note that by interchanging $\sigma_{p_j}^2$ and $\sigma_{q_j}^2$, the same result is obtained when $\sigma_{q_j}^2 \geq \sigma_{p_j}^2$. Thus, to make the result hold in general, we can set

$$r_j = \exp \left\{ \left| \log \frac{\sigma_{q_j}^2}{\sigma_{p_j}^2} \right| \right\}, \quad j = 1, 2,$$

which is an expression for the ratio between variances where the largest of the variances is always in the numerator. Therefore, we get that $\log r_2 > \log r_1$ is equivalent to $H_2 > H_1$. □

The rest of this article contains the individual proofs of the remaining propositions in the main body of the text.

Proof of Proposition 2. Let $\log r_j$ for $j = 1, 2$ be defined as in Lemma 1. When assuming that $a_{12} = 1$ and $a_{11} = a_{22} = a \neq 1$, we get that

$$\log r_2 = \left| \log \frac{a^2/2 + a^2/2 - |\rho| a^2}{1 - |\rho|} \right| = |\log a^2|,$$

and

$$\log r_1 = \left| \log \frac{a^2/2 + a^2/2 + |\rho| a^2}{1 + |\rho|} \right| = |\log a^2|.$$

Hence, by arguments along the lines of the proof of Lemma 1, we see that $H_2 = H_1$ no matter what $|\rho|$ or a is. □

Proof of Proposition 3. Using the formulas for the variances of the projections (A2) and (A3), the inequality we have to study according to Lemma 1 becomes the following:

$$\begin{aligned}\left|\log\frac{a^2-2a|\rho|+1}{2(1-|\rho|)}\right| &> \left|\log\frac{a^2+2a|\rho|+1}{2(1+|\rho|)}\right| \\ \left|\log\left[\frac{(1-a)^2}{2(1-|\rho|)}+a\right]\right| &> \left|\log\left[\frac{(1-a)^2}{2(1+|\rho|)}+a\right]\right|.\end{aligned} \tag{A5}$$

First, we have to find the sign of the expressions inside the absolute values for each a and $|\rho|$. For the left-hand side, we get

$$\begin{aligned}\frac{(1-a)^2}{2(1-|\rho|)}+a &= 1 \\ a &= 1 \text{ and } a = 2|\rho|-1.\end{aligned}$$

Thus, for $a > 1$ and $a < 2|\rho| - 1$, the left-hand side is positive, whereas negative in between. For the right-hand side, the expression inside the absolute value signs are positive for $a > 1$ and $a < -(1 + 2|\rho|)$. Because $a > 0$, however, the relevant root for the right-hand side is only $a = 1$. In total, this gives us three regions of $(a, |\rho|)$-values to check inequality (A5): $a > 1$ and $|\rho| \in (0, 1)$, $a \in (2|\rho| - 1, 1)$ and $|\rho| \in (0, 1)$, and $a \in (0, 2|\rho| - 1)$ and $|\rho| \in (1/2, 1)$.

$a > 1$ and $|\rho| \in (0, 1)$:

The absolute value signs can now be dissolved, so that inequality (A5) becomes

$$\frac{(1-a)^2}{(1-|\rho|)} > \frac{(1-a)^2}{(1+|\rho|)}.$$

Because $|\rho| \in (0, 1)$, we see that the inequality holds for any $a > 1$. Hence, $H_2 > H_1$ in this scenario, when the variance increases.

$a \in (2|\rho| - 1, 1)$ and $|\rho| \in (0, 1)$:

In this case, inequality (A5) becomes

$$\frac{(1-a)^2}{(1-|\rho|)} < \frac{(1-a)^2}{(1+|\rho|)}.$$

That is, it does not hold for any of the a's or $|\rho|$'s within the relevant region. Note that when $|\rho| < 1/2$, a is kept between $(0, 1)$.

$a \in (0, 2|\rho| - 1)$ and $|\rho| \in (1/2, 1)$:

Now we get the inequality

$$\frac{(1-a)^2}{2(1-|\rho|)}+a > \left(\frac{(1-a)^2}{2(1+|\rho|)}+a\right)^{-1},$$

which is equivalent to

$$a^4 - a^2(4\rho^2 - 2) + 4\rho^2 - 3 > 0. \tag{A6}$$

The roots of the function on the left-hand side are $a = \pm 1$ and $a = \pm\sqrt{4\rho^2 - 3}$, but the only relevant root for $a \in (0, 2|\rho| - 1)$ and $|\rho| \in (1/2, 1)$ is $a_0 := \sqrt{4\rho^2 - 3}$.

Next, for $|\rho| < \sqrt{3}/2$, the root a_0 moves into the complex plane, and the function on the left-hand side of (A6) is always less than 0 for the relevant a's. That is, $H_2 < H_1$ in this case. If $|\rho| > \sqrt{3}/2$, on the other hand, then (A6) holds for $a \in (0, a_0)$, but not for $a \in (a_0, 2|\rho| - 1)$.□

Proof of Proposition 4. In this scenario, the inequality to check due to Lemma 1 and expressions (A2) and (A3) is

$$\left|\log\frac{1-a|\rho|}{1-|\rho|}\right| > \left|\log\frac{1+a|\rho|}{1+|\rho|}\right|. \tag{A7}$$

To dissolve the absolute value signs, we first have to see for which values of a and $|\rho|$ the expressions inside are positive or negative. It is easily verified that the expression inside the left-hand side absolute value is positive for $a < 1$, whereas the right-hand side is positive if $a > 1$, both being negative otherwise.

First assume that $a < 1$. Then inequality (A7) becomes

$$\begin{aligned}\frac{1-a|\rho|}{1-|\rho|} &> \frac{1+|\rho|}{1+a|\rho|} \\ 1-(a\rho)^2 &> 1-\rho^2 \\ a^2 &< 1.\end{aligned}$$

Hence, $a \in (-1, 1)$ yields $H_2 > H_1$. On the other hand, if $a > 1$, we obtain

$$\frac{1-|\rho|}{1-a|\rho|} > \frac{1+a|\rho|}{1+|\rho|}$$
$$a^2 > 1,$$

which is always true. Thus, in total, $H_2 < H_1$ only if $a < -1$. □

Paper II

Online detection of sparse changes in high-dimensional data streams using tailored projections

Martin Tveten and Ingrid K. Glad

Manuscript. Openly available on arXiv: 1908.02029 [stat.ME].

II

Online Detection of Sparse Changes in High-Dimensional Data Streams Using Tailored Projections

Martin Tveten and Ingrid K. Glad,
Department of Mathematics, University of Oslo

August 7, 2019

Abstract

When applying principal component analysis (PCA) for dimension reduction, the most varying projections are usually used in order to retain most of the information. For the purpose of anomaly and change detection, however, the least varying projections are often the most important ones. In this article, we present a novel method that automatically tailors the choice of projections to monitor for sparse changes in the mean and/or covariance matrix of high-dimensional data. A subset of the least varying projections is almost always selected based on a criteria of the projection's sensitivity to changes.

Our focus is on online/sequential change detection, where the aim is to detect changes as quickly as possible, while controlling false alarms at a specified level. A combination of tailored PCA and a generalized log-likelihood monitoring procedure displays high efficiency in detecting even very sparse changes in the mean, variance and correlation. We demonstrate on real data that tailored PCA monitoring is efficient for sparse change detection also when the data streams are highly auto-correlated and non-normal. Notably, error control is achieved without a large validation set, which is needed in most existing methods.

Keywords: Statistical Process Control (SPC), Principal Component Analysis, Anomaly Detection, Change-point Detection, Bootstrap/Resampling.
R packages: `tpca`, `tpcaMonitoring` and `tdpcaTEP` are available from https://github.com/Tveten. The packages include all code to easily reproduce our results.

Additional supplementary materials are available online (see the list at the end of the article).

1. INTRODUCTION

1.1 Motivation

The exploding availability of cheap sensors has created a need for new methods to harvest insight from them. In many applications, these sensors are deployed in large networks for online monitoring of a system. Concrete examples include temperature monitoring of a data center at Johns Hopkins (Mishin et al. 2014), plant-wide monitoring of industrial processes (Ge 2017) and semiconductor manufacturing (Zou et al. 2014). Similar technology is also used within video segmentation (Kuncheva and Faithfull 2014), solar flare detection (Liu et al. 2015), medical monitoring, DNA protein sequence analysis, network intrusion detection and speech recognition. Our own motivation comes from condition monitoring of ships, where around 100-500 sensors are placed to measure the ship's state in terms of propulsion, temperatures, pressure and other physical quantities.

For many applications, there is a need for quick detection of anomalies that arise in the form of sustained changes in the data distribution. E.g., the pressure in a valve is too high, which should be attended to as quickly as possible, or a small number of sensors suddenly becomes faulty. This illustrates that changes may (and perhaps most often) only occur in a small subset of the sensors in an entire network. Thus, lately, several authors (see Section 1.4) have worked on the problem of change detection from the angle of only a small, unknown set of affected sensors, or so-called *sparse* changes. Mostly, they focus on changes in the mean of independent normal data, or assume all parameters in the model to be known, both before and after a change.

However, in some applications (Hawkins and Zamba 2009; Woodall and Montgomery 2014; Kuncheva and Faithfull 2014), sparse changes both in the mean and in the covariance matrix of the sensors are of interest. For instance, if a certain level of stability in a process is required, or because one has learned from historical data or experts that a group of sensors should be correlated in a specific way. Additonally, parameters are unknown in most cases and must be estimated. If estimation uncertainty is not accounted for, many false alarms will be raised, which is highly undesirable. The problem we address in this article is therefore sequential detection of sparse changes in the mean and/or covariance structure of high-dimensional data streams, with all parameters unknown. To make the method scalable, principal component analysis is incorporated and studied within this change detection framework.

1.2 Problem Formulation

Imagine a system being monitored by D sensors at times indexed by t, yielding a multivariate data stream of observations $\mathbf{x}_t \in \mathbb{R}^D$. First, there is a training period where m observations $\mathbf{x}_{-m+1}, \ldots, \mathbf{x}_0$ of the system under normal conditions are generated. From $t \geq 1$ the data stream $\mathbf{x}_t$ is monitored *online* or *sequentially* for a change in its joint distribution. The change is thought of as being a consequence of an anomaly in the system. Importantly, the anomaly might be local, and therefore only affect a small number of sensors. The aim is to detect these anomalies as soon as possible, but false alarms should be kept at a controlled level.

For simplicity, our modelling assumptions are mainly as follows, but extensions to handle time-dependency and non-normality are presented and tested in Section 5. First, there is a training period where m independent $N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ observations $\mathbf{x}_t$ are gathered. As monitoring ensues, observations keep arriving from the null distribution until a change-point $\kappa \geq 0$, after which the distribution of $\mathbf{x}_t$ changes to $N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ for all $t > \kappa$. A key element is the assumption that only a subset $\mathcal{D} \subseteq \{1, \ldots, D\}$ of the sensors are affected by a change, following the perspective of Xie and Siegmund (2013). The *subset of affected streams* is defined by

$$\mathcal{D} = \{d : \ \mu_{0,d} \neq \mu_{1,d} \text{ or } (\boldsymbol{\Sigma}_0)_{d,*} \neq (\boldsymbol{\Sigma}_1)_{d,*}\}, \tag{1}$$

where $(A)_{d,*}$ denotes the d-th row of a matrix. In other words, we assume that the change in mean vector $\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0$ and/or change in covariance matrix $\boldsymbol{\Sigma}_1 - \boldsymbol{\Sigma}_0$ has a sparsity structure. This

sparse online change-point problem is summarized by the following sequential hypothesis test:

$$\begin{aligned} H_0: &\quad \mathbf{x}_t \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0), \quad t = -m+1, -m+2 \dots \\ H_1: &\quad \text{There is a } \kappa \geq 0 \text{ such that} \\ &\quad \mathbf{x}_t \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0), \quad t = -m+1, \dots, \kappa \\ &\quad \mathbf{x}_t \sim N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), \quad t = \kappa+1, \kappa+2, \dots, \end{aligned} \tag{2}$$

where only parameters for $d \in \mathcal{D}$ change, and κ, $\mathcal{D}$, $\boldsymbol{\mu}_0$, $\boldsymbol{\Sigma}_0$, $\boldsymbol{\mu}_1$ and $\boldsymbol{\Sigma}_1$ are *all unknown*. Our primary interest is the high-dimensional, sparse problem, where D is high and $|\mathcal{D}|$ is relatively small. Ultimately, we end up with a stopping rule for (2) of the form

$$T = \inf\{t : \Lambda_t \geq b\}, \tag{3}$$

where Λ_t is a running test statistic of all observations, including the training set.

Note that the assumption of independence in time is less restrictive than one may think. Rather than thinking of $\mathbf{x}_t$ as the raw observations, they can be thought of as residuals from a spatio-temporal model, learned in advance. The monitoring procedure then raises an alarm when the spatio-temporal model does not explain the incomming data well anymore.

To describe how the stopping rules T are evaluated, let $\mathbb{P}^\kappa$ and $\mathbb{E}^\kappa$ denote probability and expectation when there is a true change-point at κ. In particular, $\mathbb{P}^\infty$ and $\mathbb{E}^\infty$ mean probability and expectation under H_0. For a chosen monitoring length n, we control the *probability of false alarms* (PFA) at a given level,

$$\mathbb{P}^\infty(T \leq n) \leq \alpha. \tag{4}$$

(This measure of false alarms compared to the more common average run length (ARL) is discussed in Section 3.) Then, if a change actually occurs at κ, the aim is to detect it as quickly as possible, measured by the (conditional) *expected detection delay* (EDD),

$$\mathbb{E}^\kappa[T - \kappa | T > \kappa]. \tag{5}$$

The EDD is the expected sample size to detect a given change. The lower it is, the better.

If one disregards the sparsity of the change, a solution to the problem (2) can be obtained through relatively straightforward generalized likelihood ratio methodolgy (Sullivan and Woodall 2000; Hawkins and Zamba 2009). However, these methods are not efficient in the high-dimensional and sparse change setting for several reasons. Firstly, they do not incorporate prior information about the sparsity of a change, yielding slow detection. Secondly, they scale poorly with D in terms of detection speed. To see this, let t denote the current time, $k < t$ be a candidate change-point, so that $t - k$ is the number of observations used in estimating $\boldsymbol{\Sigma}_1$. Then $t - k > D$ for a non-degenerate maximum likelihood estimate. This means that the most recent candidate change-point k will grow further apart from the current time t as D grows, resulting in very slow detection. Even if one uses regularization techniques to circumvent a singular maximum likelihood estimate, there would still be a need of an increasing amount of observations for a reliable estimate. Thirdly, they are not scalable computationally as D grows because of the burden of computing many increasingly larger covariance matrices.

Dimension reduction tools are often employed to overcome high-dimensional challenges, but they have not been studied much in the online change-point detection context. Therefore, our main objective in this work is to take a common and well understood dimension reduction tool, principal component analysis (PCA), use knowledge about how it reacts to (sparse) changes in the mean and covariance matrix, and come up with an efficient way to use it for online change detection.

Our strategy for solving the change-point problem (2) with tailored PCA is as follows:

1. Obtain the sample principal axes $\hat{\mathbf{v}}_j$, $j = 1, \dots, D$, from the training set $\mathbf{x}_{-m+1}, \dots, \mathbf{x}_0$.

(The sample principal axes are the eigenvectors of the sample covariance matrix $\hat{\mathbf{\Sigma}}_0$.)

2. Figure out which of the projections onto sample principal axes that are most sensitive to a given set of relevant or possible changes. Pick the J most sensitive and disregard the rest.

3. For $t > 0$, monitor the mean and variance of the projections $y_{j,t} = \hat{\mathbf{v}}_j^\intercal \mathbf{x}_t$, $j = 1, \ldots, J$. In this way, the problem of detecting changes in the entire covariance matrix is reduced to detecting changes in marginal variances.

This procedure is first studied within the modelling and evaluation framework described above to get some understanding under a clean setup, before an extension to more realistic data is proposed in Section 5.

Point (2) above is the main focus of Sections 2, while point (3) together with false alarm control is handled in Section 3. Empirical results from simulation studies are presented in Section 4. Lastly, in Section 5, our method is extended to tackle non-normal and time-dependent data, and benchmarked on the Tennessee Eastman process.

1.3 Main Contributions

There are two main contributions of this work:

1. A principled approach to automatically choosing which principal axes to keep for a specific change detection task, readily implemented in an R package. This was an open problem posed by Kuncheva and Faithfull (2014).

2. An online monitoring scheme that extends the scheme of Xie and Siegmund (2013) for sparse, positive changes in the mean of independent data, to detection of sparse changes in the mean and/or covariance matrix of time-dependent data. Our scheme is scalable, and includes all sources of estimation uncertainty when finding a threshold that meets a specified probability of false alarms, without the need of a large validation set.

Expanding on Kuncheva and Faithfull (2014) and Tveten (2019), we find that a subset of the least varying projections tend to be selected for a wide range of change scenarios and pre-change covariance matrices. We also conclude that monitoring the projections y_{jt} offer a solution to all the discussed shortcomings of a direct approach; quicker detection and computation can be attained because there are less parameters to estimate online, and information about change sparsity can be incorporated in our method for choosing projections.

1.4 Connections with Prior Work

The work in this article intersects with many fields, including anomaly and novelty detection in the machine learning world, statistical offline and online change/change-point detection, and statistical process control (SPC).

As the previous section suggests, the work in this article is mainly inspired by Xie and Siegmund (2013), Kuncheva and Faithfull (2014) and Tveten (2019). We follow Xie and Siegmund (2013) approximately in formulating the change-point problem. The difference is that they are interested in the case where the variance is known and constant, there is no correlation between the streams, and only positive changes in the mean are of interest.

On the other hand, Kuncheva and Faithfull (2014) motivated the use and study of PCA for our problem by arguing that the least varying projections were the most useful through a bivariate example. Tveten (2019) elaborate and, sometimes, correct their picture by letting the answer depend on the pre-change covariance matrix and a more comprehensive list of possible change scenarios. In contrast to Kuncheva and Faithfull (2014), Tveten (2019) traces changes that occur in the distribution of $\mathbf{x}_t$ through the projection, and see how the distribution of the projections $y_{j,t}$ changes as a result. We build on this to develop the general method for choosing projections to monitor online for changes presented here.

The problem of sequential detection of sparse changes has recieved much recent interest beyond Xie and Siegmund (2013), which we have drawn upon in some way or another. Most of the research in this direction, however, is either concerned with changes in the mean of independent normals (or a known covariance matrix) (Zou et al. 2014; Wang and Mei 2015; Chan 2017), or assumes that both the pre- and post-change distributions are known (Mei 2010; Banerjee and Veeravalli 2015; Fellouris and Sokolov 2016). The work of Mei et al. (2017) is interesting and relevant in that no assumptions on the distributions are made, but it is not a fully multivariate approach.

Our motivation for the choice of performance metrics comes from discussions by Lai (1995) and Lai and Xing (2010). These works also study generalized likelihood ratio approaches where parameters have to be estimated, discuss window lengths as well as obtaining thresholds by bootstrapping. All of which is relevant to the present article.

All of the mentioned articles fall in a tradition that was initiated by Page (1955) and later expanded by Lorden (1971) and Moustakides (1986). The significant contribution of Siegmund (1985) should also be mentioned.

There is also a connection from our work to Kirch and Tadjuidje Kamgaing (2015) and Dette and Gösmann (2018), who study online change-point detection within a more recently developed theoretical framework. They consider monitoring statistics that incorporate a training set, and control the probability of false alarms under the asymptotic scheme of the number of training samples going to infinity. Their setup is very general, and contains much less rigid assumptions than the works we have mentioned so far, but does not consider sparse changes explicitly. Moreover, we control false alarms for a finite number of training samples.

Relevant litterature also exists within stochastic process control. Hawkins and Zamba (2009) and Sullivan and Woodall (2000) consider the same change-point problem as in this paper, but without incorporating an assumption about the sparsity of a change. Chan and Zhang (2001) also study the detection of changes in the mean and/or covariance matrix by the use of projection pursuit as a dimension reduction tool, rather than PCA, but assume the pre-change parameters to be known. Additionally, there are plenty of control charts based on PCA (see for example the reviews Weese et al. (2015) and Rato et al. (2016)). These are, however, not set within the online change-point detection framework of controlling the false alarm rate and measuring detection delays, and they only handle sparse changes implicitly.

Within the realm of anomaly detection in machine learning, PCA has been used in numerous ways. The work in Qahtan et al. (2015) is closely related to Kuncheva and Faithfull (2014), but they use PCA in the standard way where only the most varying projections are selected. Lakhina et al. (2004) and Huang et al. (2007) use PCA to detect anomalies in (traffic) networks, and, like us, they find that it is the residual subspace of PCA that is most useful. This fact is also pointed to in the extensive review of novelty detection techniques and applications in Pimentel et al. (2014). A difference from these works to us is that what they consider as anomalies are outliers in a trained model, not changes in distribution. And, most importantly, we do not use the entire residual subspace, but rather the subspace of it that is most sensitive to a user-defined set of relevant distributional changes. Other examples of PCA-based anomaly detection procedures are Ferrer (2007), Mishin et al. (2014) and Harrou et al. (2015). None of the articles mentioned in this paragraph considers the speed of detection, which is a major difference to our objective.

2. TAILORING THE CHOICE OF PRINCIPAL AXES TO CHANGE DETECTION

In this section, the insights from Tveten (2019) about the sensitivity of projections to various changes and the dependence on the pre-change covariance matrix are knit together into an algorithm that decides which projections to use for a given change-point problem. Such an automatic choice of projections is what we mean by *tailoring* PCA for change detection. In the next section we test it in the online change detection setting.

What do we mean by sensitivity to changes? Akin to Kuncheva and Faithfull (2014) and

Tveten (2019), we define it by a divergence between the marginal distribution of each projection before and after a change. Here we follow Tveten (2019), who use the Hellinger distance. The squared Hellinger distance between two normal distributions $p(x) = N(x|\xi_1, \sigma_1)$ and $q(x) = N(x|\xi_2, \sigma_2)$ is given by

$$H^2(p, q) = 1 - \sqrt{\frac{2\sigma_1\sigma_2}{\sigma_1^2 + \sigma_2^2}} \exp\left\{-\frac{1}{4}\frac{(\xi_1 - \xi_2)^2}{\sigma_1^2 + \sigma_2^2}\right\}.$$

A desirable feature of the Hellinger distance between two normals is that it is symmetric with respect to whether the variance increases or decreases in the sense that a multiplicative increase of the variance by a factor $a \geq 1$ changes the distribution as much as a decrease by the factor $1/a$. This is also a property of the generalized likelihood ratio procedure for detecting changes in the mean and/or variance we use for monitoring later. There could be reasons for using other divergences, however, so in the accompanying R package, any divergence can be specified. Our own experiments suggest that the overall conclusions will not be significantly different by using for example the KL-divergence or Bhattacharyya distance.

Formally, the definition of sensitivity to changes we use, as defined in Tveten (2019), is as follows. Recall that $\boldsymbol{\mu}_0$ and $\boldsymbol{\Sigma}_0$ are the pre-change mean and covariance matrix of $\mathbf{x}_t$, while $\boldsymbol{\mu}_1$ and $\boldsymbol{\Sigma}_1$ are the post-change parameters. Without loss of generality, assume that $\mathbf{x}_t$ is standardized with respect to the pre-change parameters, so that $\boldsymbol{\mu}_0 = \mathbf{0}$ and $\boldsymbol{\Sigma}_0$ is a correlation matrix. Next, let $\{\lambda_j, \mathbf{v}_j\}_{j=1}^D$ be the normalized eigensystem of $\boldsymbol{\Sigma}_0$, where it has been sorted such that $\lambda_1 \geq ... \geq \lambda_D$. Then the orthogonal projections onto the pre-change principal axes are given by $y_{j,t} = \mathbf{v}_j^\intercal \mathbf{x}_t$, for $j = 1, \ldots, D$. Assuming $\mathbf{x}_t$ is multivariate normal, $y_{j,t}$ has marginal pre- and post-change density functions

$$\begin{aligned} p_j(y) &= N(y|\ \mathbf{v}_j^\intercal \boldsymbol{\mu}_0, \mathbf{v}_j^\intercal \boldsymbol{\Sigma}_0 \mathbf{v}_j) = N(y|\ 0, \lambda_j) \\ q_j(y) &= N(y|\ \mathbf{v}_j^\intercal \boldsymbol{\mu}_1, \mathbf{v}_j^\intercal \boldsymbol{\Sigma}_1 \mathbf{v}_j), \end{aligned} \tag{6}$$

respectively. Given a correlation matrix $\boldsymbol{\Sigma}_0$, the *sensitivity of the j'th projection to the change specified by* $(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ is defined as $H(p_j, q_j)$, abbreviated by H_j. Importantly, note that the sensitivity as defined here is a function of the pre- and post-change parameters of the original data $\mathbf{x}_t$: $\boldsymbol{\Sigma}_0$, $\boldsymbol{\mu}_1$ and $\boldsymbol{\Sigma}_1$.

Using this definition of sensitivity, Tveten (2019) proved that for bivariate normal data, the least varying projection is the most sensitive if one of the means change, one of the variances increases, or the correlation changes. If one variance decrease, then the most varying projection is the most sensitive unless the pre-change correlation is larger than $\sqrt{3}/2 \approx 0.87$, On the other hand, when both means or both variances change, there are no clear winner among the projections. Thus, we hypothesize that the least varying projections are particularly useful if changes have some sparsity structure, which they almost always are in the high-dimensional setting. The general take-away, however, is that which projections are most sensitive depends on the pre-change correlation matrix and the exact nature of the change.

The tailored PCA (TPCA) method is motivated from the bivariate results. In short, the procedure is as follows. First, an estimate of the pre-change correlation matrix, $\hat{\boldsymbol{\Sigma}}_0$, must be obtained from a training set. Then simulate B changes from a customizable *change distribution* $p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1|\hat{\boldsymbol{\Sigma}}_0)$, measure each projection's sensitivity to each change, $(H_1, \ldots, H_D)^{(b)}$, $b = 1, \ldots, B$, and summarize the sensitivity in a way that yields a meaningful ranking of the principal axes/projections. A selection of projections can then be made from the ranking.

In principle, any distribution for $p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1|\hat{\boldsymbol{\Sigma}}_0)$ could be used, but the space of all possible combinations of changes is extremely vast. Therefore, we make some restrictions to simplify the space of changes. First, we restrict ourselves to consider only one *change type* at a time. The change type can then be seen as a single-trial multinomially distributed random variable $\mathbf{C}$ with

probabilities p_μ (mean), p_σ (variance) and p_ρ (correlation). Secondly, let $K = |\mathcal{D}| \in \{1, 2, \ldots, D\}$ be the *change sparsity*, where $\mathcal{D}$ is defined as in (1), which indicates how many dimensions that are affected by a change. I.e., the number of non-zero elements in $\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0$ and $\boldsymbol{\sigma}_1 - \boldsymbol{\sigma}_0$, where $\boldsymbol{\sigma}$ is the diagonal of $\boldsymbol{\Sigma}$. For a change in correlation, a change sparsity of K means that all the correlations between the K affected dimensions change. Thirdly, given a change sparsity K, we assume throughout that the exact subset of affected streams $\mathcal{D}$ is uniformly distributed over all combinations of size K. Fourthly, there is the *change size* of each type of change:

- $\mu_d \in \mathbb{R}$ is the size of an additive change in the mean in the d'th component for $d \in \mathcal{D}$.
- $\sigma_d \in \mathbb{R}_{>0}$ is the size of a multiplicative change in the standard deviation in the d'th component for $d \in \mathcal{D}$.
- a_{di} such that $a_{di}\rho_{di} \in [0, 1)$ for $d \neq i \in \mathcal{D}$ is the size of a multiplicative change in each pre-change correlation. (Not all changes of this element-wise sort will result in a positive definite correlation matrix. See the supplementary material for how we deal with this.)

Note that for practical purposes it is reasonable to restrict the domain of the change sizes to sizes that are actually relevant, but the above outlines the theoretical scope of the post-change parameter subspace we consider.

A change distribution $p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1 | \boldsymbol{\Sigma}_0)$ can now be characterized by a distribution over the parameters $(\mathbf{C}, K, \mathcal{D}, \mu_d, \sigma_d, a_{di})$. Due to space limitations, we will only show results for a change distribution that represents very little prior information about the nature of a change. The two minor exceptions are that we assume interest is restricted to change sparsities $K \leq D/2$ and that the correlation can only decrease. This change distribution is given by

$$\begin{aligned}
\mathbf{C} &\sim \text{Multinom}(p_\mu = 1/3, p_\sigma = 1/3, p_\rho = 1/3) \\
K &\sim \text{Unif}\{1, \ldots, D/2\} \\
\mathcal{D}|K &\sim \text{Unif}\{\mathcal{D} \subseteq \{1, \ldots, D\} : |\mathcal{D}| = K\} \\
\mu_d|\mathcal{D}, \mathbf{C} &\overset{iid}{\sim} \text{Unif}[-1.5, 1.5],\ d \in \mathcal{D} \\
\sigma_d|\mathcal{D}, \mathbf{C} &\overset{iid}{\sim} \frac{1}{2}\text{Unif}[1/2.5, 1] + \frac{1}{2}\text{Unif}[1, 2.5],\ d \in \mathcal{D} \\
a_{di}|\mathcal{D}, \mathbf{C} &\overset{iid}{\sim} \text{Unif}[0, 1],\ d \neq i \in \mathcal{D}.
\end{aligned} \tag{7}$$

The supplementary material contains simulations that show that the exhibited results are fairly robust to the choice of change distribution. In the accompanying R-package `tpca`, one can easily set up uniform change distributions as in 7 over other sets of change scenarios.

By using change distribution (7) and a randomly generated 20-dimensional pre-change correlation matrix, Figure 1 illustrates that the least varying projections are the most sensitive on average, but that notable variation is hidden on the level of the exact nature of a change. To capture this variation, our idea is to estimate how often projection j is the most sensitive one for a given correlation matrix, and use this to rank the projections. That is, we want to estimate

$$P_j := \mathbb{P}\left(\underset{i \in \{1,\ldots,D\}}{\text{argmax}}\, H(p_i, q_i) = j \,\middle|\, \boldsymbol{\Sigma}_0 \right),\ \text{for } j = 1, \ldots, D. \tag{8}$$

In this way, the probability of omitting a projection that is maximally sensitive to a particular change can be controlled.

To automate the choice of a projections, a cutoff value $c \in [0, 1]$ can be selected such that the projections with the highest probability of being the most sensitive are picked until the sum of probabilities is greater than c. Then $1 - c$ corresponds to the probability of not picking a projection that is maximally sensitive to a change. Figure 2 displays the estimated probabilities

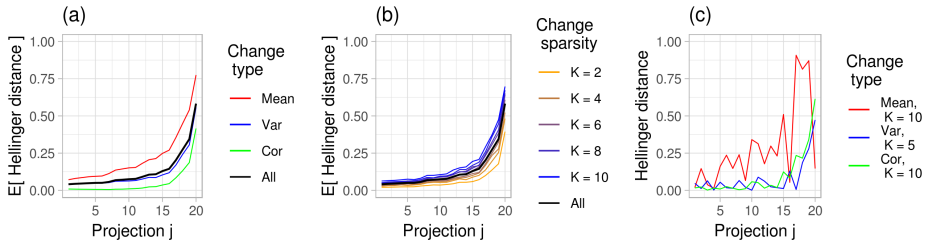

FIGURE 1. (a) and (b) display Monte Carlo estimates of $E[H_j|\mathbf{\Sigma}_0]$, $j = 1, \ldots, 20$ for a randomly generated $\mathbf{\Sigma}_0$, with respect to the change distribution (7). (a) show results conditional on change type and (b) on change sparsity. 10^4 Monte Carlo samples were used. Note that $j = 1$ and $j = 20$ are the most and least varying projections, respectively. (c) displays H_j for one randomly selected change in each class of change type, to illustrate what each outcome that is averaged over to obtain (a) and (b) looks like.

$\hat{P}_j$ corresponding to the same simulations as in Figure 1. Observe that even for c close to 1, only a few of the least varying projections would be selected for all change types. However, more axes would be selected for general changes and changes in the mean than for changes in the variance or correlation. Consult the supplementary material for more simulations regarding which axes that are selected.

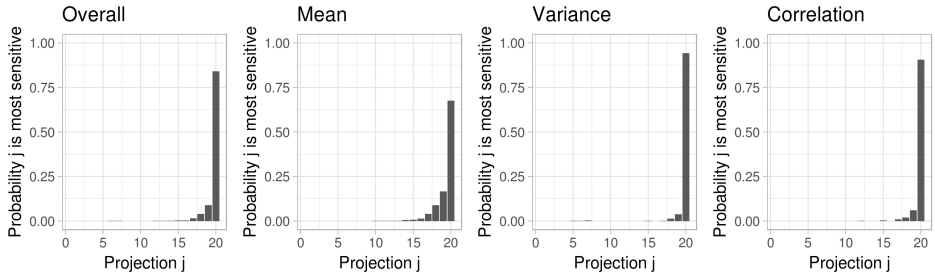

FIGURE 2. Monte Carlo estimates of P_j with respect to the same $\mathbf{\Sigma}_0$ as in Figure 1 and same draws from change distribution (7). The three right-most figures show the contributions to the overall probabilities (left) for each change type.

To summarize, Algorithm 1 describes the tailoring procedure in detail. We call it the TPCA algorithm, and it is implemented in the R package `tpca`. For online monitoring of data streams, it is intended as a final step in the training phase. In the training phase, an estimate $\hat{\mathbf{\Sigma}}_0$ of the pre-change correlation matrix is obtained, a change distribution $p(\boldsymbol{\mu}_1, \mathbf{\Sigma}_1|\hat{\mathbf{\Sigma}}_0)$ is set up to represent the changes of interest, and a cutoff c is chosen. Then the tailoring algorithm is run to determine which principal axes $\mathcal{J} \in \{1, \ldots, D\}$ to project the incomming data onto. Ultimately, monitoring of $\hat{\mathbf{v}}_j^\mathsf{T}\mathbf{x}_t$, $j \in \mathcal{J}$ ensues, which we deal with next.

Algorithm 1 Tailored PCA (TPCA) for Change Detection

Input: $\boldsymbol{\Sigma}_0$, $p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1|\boldsymbol{\Sigma}_0)$, c, B

1: Compute (sorted) eigenvalues and eigenvectors $\{\lambda_j, \mathbf{v}_j\}_{j=1}^{D}$ of $\boldsymbol{\Sigma}_0$
2: **for** $b \in \{1, \ldots, B\}$ **do**
3: $\quad (\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)^{(b)} \sim p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1|\boldsymbol{\Sigma}_0)$
4: $\quad (H_1^{(b)}, \ldots, H_D^{(b)}) \leftarrow (H(p_1, q_1^{(b)}), \ldots, H(p_D, q_D^{(b)})$ ▷ p_j and q_j are given in (6).
5: **end for**
6: $\hat{P}_j \leftarrow \frac{1}{B}\sum_{b=1}^{B} I\left\{\underset{i\in\{1,\ldots,D\}}{\operatorname{argmax}} H_i^{(b)} = j\right\}$
7: $\mathcal{J} \leftarrow \{j : \sum_{i\in\mathcal{J}} \hat{P}_i \geq c \text{ such that } |\mathcal{J}| \text{ is minimal}\}$.

Return: $\mathcal{J}$ and $\{\lambda_j, \mathbf{v}_j\}_{j\in\mathcal{J}}$

3. ONLINE MONITORING

In this section, focus is shifted back to the sparse online change-point problem (2). First, the monitoring statistic we will use for performance analysis is presented, before we turn to handling the uncertainty stemming from estimating the eigensystem. We have chosen a monitoring statistic that can be set up to handle sparse changes in the mean and/or variance vectors directly in the original data $\mathbf{x}_t$ as well as indirectly in the projections. In this way, we obtain a fair benchmark for the TPCA method, both in terms of detection speed and dimension reduction capabilities. What the monitoring statistic can not do when applied directly to the original data is to detect changes in the cross-stream correlations of $\mathbf{x}_t$. We view this ability as an advantage of PCA-based procedures. Whether such an ability is important or not depends on the application.

3.1 A Mixture Procedure for Detecting Changes in the Mean and/or Variance

Our monitoring statistic generalizes the mixture generalized likelihood ratio (GLR) detection procedure of Xie and Siegmund (2013) from only detecting positive mean shifts to detecting all changes in the mean and/or variance. The key component in their mixture procedure is the incorporation of a prior guess about the sparsity of the change. As before, we assume there is a training set of size m with observations from the null distribution available, and that only an unknown proportion $p = |\mathcal{D}|/D$ of the streams are affected by a change. The mixture procedure arises from the following hypothesis testing setup:

$$\begin{aligned} H_0: &\quad \mathbf{x}_t \sim N\big(\boldsymbol{\mu}_0,\ \operatorname{diag}\{\boldsymbol{\sigma}_0^2\}\big), \quad t = -m+1, -m+2 \ldots \\ H_1: &\quad \text{There is a } \kappa \geq 0 \text{ such that} \\ &\quad \mathbf{x}_t \sim N\big(\boldsymbol{\mu}_0,\ \operatorname{diag}\{\boldsymbol{\sigma}_0^2\}\big), \quad t = -m+1, \ldots, \kappa \\ &\quad \mathbf{x}_t \sim N\big(\boldsymbol{\mu}_1,\ \operatorname{diag}\{\boldsymbol{\sigma}_1^2\}\big), \quad t = \kappa+1, \kappa+2, \ldots, \end{aligned} \tag{9}$$

where $\mu_{0,d} \neq \mu_{1,d}$ and/or $\sigma_{0,d}^2 \neq \sigma_{1,d}^2$ only for $d \in \mathcal{D} \subseteq \{1, \ldots, D\}$. The key component in the mixture procedure of Xie and Siegmund (2013) is to substitute the unknown p with a prior guess p_0, which acts as the probability that each stream n belongs to the class of affected streams or not. Note that it is assumed that changes occur in the mean and/or variance simultaneously, and then persist for all $t > \kappa$.

The mixture log-likelihood ratio statistic for a change in the mean and/or variance is derived in the following. With an assumed change-point at $\kappa = k \geq 0$ the global log-likelihood ratio is on the form

$$\Lambda_{k,t}(p_0) = \sum_{d=1}^{D} \log\left[1 - p_0 + p_0 \exp\left\{\ell_{d,k,t}\right\}\right], \tag{10}$$

where $\ell_{d,k,t}$ is the maximized likelihood ratio statistic for each stream d. So with probability $1-p_0$ all observations in stream d are assumed to come from the same distribution, while with probability p_0, the distribution of a stream can be different before and after k. Denote the maximum likelihood estimators for the mean and variance of each stream d by

$$\bar{x}_{d,i,l} := \frac{1}{l-i}\sum_{j=i+1}^{l} x_{d,j} \quad \text{and} \quad S^2_{d,i,l} := \frac{1}{l-i}\sum_{j=i+1}^{l}(x_{d,j}-\bar{x}_{d,i,l})^2.$$

Then standard calculations lead us to

$$\ell_{d,k,t} = -\frac{m+k}{2}\log\frac{S^2_{d,-m,k}}{S^2_{d,-m,t}} - \frac{t-k}{2}\log\frac{S^2_{d,k,t}}{S^2_{d,-m,t}}. \tag{11}$$

See for instance Hawkins and Zamba (2005, p. 166). Note that $\Lambda_{k,t}(p_0)$ also depends on m although it is suppressed in the notation.

Ideally, a change would be declared if $\max_k \Lambda_{k,t}(p_0)$ raises above a threshold b. However, a minor correction is preferable to prevent unwanted behavior, namely, that declaration of a change is much more likely for small sample sizes $t-k$. This is so because the distribution of $\Lambda_{k,t}(p_0)$ strongly depends on the number of observations used to estimate the post-change parameters. For example, the variance of $\Lambda_{198,200}(p_0)$ is much larger than $\Lambda_{100,200}(p_0)$, making a realization from $\Lambda_{198,200}(p_0)$ more likely to be above b than $\Lambda_{100,200}(p_0)$. An often used remedy is to find a Bartlett correction (Hawkins and Zamba 2005, p. 166), where one finds a multiplicative correction factor $C(k,t)$, such that the expected value of the statistic under the null hypothesis equates to its asymptotic expected value. The asymptotic expected value of $\Lambda_{k,t}(p_0)$ under the null hypothesis is, alas, unknown. However, the asymptotic expected value of $2\ell_{d,k,t}$ is 4 due to the classical result by Wilks. Using that for a chi-square distributed X with a degrees of freedom, $\mathbb{E}[\log X] = \log 2 + \psi(a/2)$, where ψ is the digamma function, a correction factor for each stream d is given exactly by

$$\begin{aligned}
\mathbb{E}\big[2\ell_{d,k,t}/C(k,t)\big] &= 4 \\
2C(k,t) &= -(m+t)\log(m+t) + (m+t)\psi\left([m+t-1]/2\right) \\
&\quad + (m+k)\log(m+k) - (m+k)\psi\left([m+k-1]/2\right) \\
&\quad + (t-k)\log(t-k) - (t-k)\psi\left([t-k-1]/2\right).
\end{aligned}$$

In total, the global corrected mixture log-likelihood ratio statistic becomes

$$\Lambda^C_{k,t}(p_0) := \sum_{d=1}^{D}\log\left[1-p_0+p_0\exp\left\{\ell_{d,k,t}/C(k,t)\right\}\right]. \tag{12}$$

It further defines the stopping time that constitute the detection procedure,

$$T(p_0,b) := \inf\Big\{t \geq 2 : \max_{0\leq k\leq t-2}\Lambda^C_{k,t}(p_0) \geq b\Big\}. \tag{13}$$

For comparing performance, we can now apply $T(p_0,b)$ to the original data $\mathbf{x}_t$ with various choices of p_0, and to the projections $\mathbf{y}_t$ with $p_0 = 1$ (since we want to see how TPCA handles sparsity on its own).

In practice, we will also restrict the set of k's that the maximum is taken over to a set $\mathcal{K} = \{k : 2 \leq t-k \leq w+1\}$, where w is called the *window size* and denotes the number of previous time-points that are considered as candidate change-points. This is to limit memory usage and not allow the algorithm to become slower and slower indefinitely as t grows. Choices for the set $\mathcal{K}$ and the effect of the window size is discussed in Lai (1995). Here, we will use

$w = 200$ throughout, in line with Xie and Siegmund (2013).

3.2 Monitoring by TPCA

Algorithm 2 summarizes how TPCA is used in conjunction with the mixture monitoring procedure (13) to solve the original change-point detection problem (2). Observe that the monitored observations are the standardized projections;

$$z_{j,t} = \hat{\mathbf{v}}_j^\intercal \mathbf{S}_0^{-1}(\mathbf{x}_t - \hat{\boldsymbol{\mu}}_0)/\sqrt{\hat{\lambda}_j}, \tag{14}$$

for $j \in \mathcal{J}$, where $\hat{\boldsymbol{\mu}}_0$ is the training sample mean, $\mathbf{S}_0$ is the diagonal matrix of training sample standard deviations, and $\{\hat{\lambda}_j, \hat{\mathbf{v}}_j\}_{j\in\mathcal{J}}$ is the sample eigensystem of the training sample correlation matrix. In other words, the observations $\mathbf{x}_t$ are first standardized by $\mathbf{u}_t = \mathbf{S}_0^{-1}(\mathbf{x}_t - \hat{\boldsymbol{\mu}}_0)$, since PCA is not invariant to scaling. These standardized observations then form the basis of PCA, and we get the projections $y_{j,t} = \hat{\mathbf{v}}_j^\intercal \mathbf{u}_t$. Lastly, the projections are normalized by $z_{j,t} = y_{j,t}/\sqrt{\hat{\lambda}_j}$. The reason for normalizing the projections is numerical stability, since the variance of $y_{j,t}$ for j close to D will typically be very small.

Algorithm 2 Monitoring by TPCA

Input: b, $p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1|\boldsymbol{\Sigma}_0)$ and $\{\mathbf{x}_s\}_{s=-m+1}^0$.

1: Compute $\hat{\boldsymbol{\mu}}_0$, $\mathbf{S}_0$ and the correlation matrix $\hat{\boldsymbol{\Sigma}}_0$ from $\{\mathbf{x}_s\}_{s=-m+1}^0$.
2: $\mathcal{J}$ and $\{\hat{\lambda}_j, \hat{\mathbf{v}}_j\}_{j\in\mathcal{J}} \leftarrow$ the results of applying Algorithm 1 to $\hat{\boldsymbol{\Sigma}}_0$ with $p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1|\hat{\boldsymbol{\Sigma}}_0)$.
3: $z_{j,t} \leftarrow \hat{\mathbf{v}}_j^\intercal \mathbf{S}_0^{-1}(\mathbf{x}_s - \hat{\boldsymbol{\mu}}_0)/\sqrt{\hat{\lambda}_j}$, for $t = -m+1, \ldots, 0$ and $j \in \mathcal{J}$.
4: $t \leftarrow 0$ and $\Lambda_{\max,t}^C(1) = 0$.
5: **while** $\Lambda_{\max,t}^C(1) < b$ **do**
6: $\quad t \leftarrow t + 1$ and new data $\mathbf{x}_t$ arrives.
7: $\quad \mathbf{z}_t \leftarrow (z_{j,t}) \leftarrow \hat{\mathbf{v}}_j^\intercal \mathbf{S}_0^{-1}(\mathbf{x}_t - \hat{\boldsymbol{\mu}}_0)/\sqrt{\hat{\lambda}_j}$ for $j \in \mathcal{J}$.
8: $\quad \Lambda_{\max,t}^C(1) \leftarrow \max_{k\in\mathcal{K}} \Lambda_{k,t}^C(1)$ based on $\{\mathbf{z}_s\}_{s=-m+1}^t$.
9: **end while**

Return: t

It is important to note that the estimates $\hat{\boldsymbol{\mu}}_0$, $\mathbf{S}_0$ and $\{\hat{\lambda}_j, \hat{\mathbf{v}}_j\}_{j\in\mathcal{J}}$ are not updated as more data arrives. Ideally they would be updated for every new observation $\mathbf{x}_t$, but then the procedure would lose its sequential nature; all projections $z_{j,s}$ for all s would have to be recalculated at every step, as well as everything based on them. Estimating the quantities needed for the projections only once in combination with incorporating the estimation uncertainty when calibrating the threshold b is a solution that allows for both recursive computations on the projections and control of false alarms under a correctly specified model.

3.3 Controlling False Alarms

How can one set the threshold b? As in regular hypothesis testing there is a trade-off between false positives and false negatives. There are several sequential analogs, but recall that we use the probability of false alarm (4) and the expected detection delay (5), respectively, motivated by the discussion in Lai (1995). A threshold b can now be found by choosing a segment length n and a probability of false alarm α, then solving $\alpha = \mathbb{P}^\infty[T(p_0, b) \leq n]$ for b. The EDD of the stopping rules can then be compared, where the goal is as low EDD as possible.

Remark. The *average run length* (ARL), defined as $\mathbb{E}^\infty[T(b)]$, is perhaps a more commonly used measure of false alarms. However, in many applications, a false alarm is very undesirable, and Lai (1995) argues that a more informative measure of false alarms is to consider the probability of no false alarm during a typical, steady-state period of operation. For example, an average run

length of 1000 does not necessarily mean that the probability of a false alarm during the first 100 observations is low. Another advantage is that the PFA is much more tractable to compute by Monte Carlo simulation. Finally, also pointed out by Lai (1995, p. 631), the two quantities are related approximately by

$$\mathbb{E}^{\infty}[T(b)] \approx n/\mathbb{P}^{\infty}[T(b) \leq n],$$

for the stopping rule we consider.

Finding thresholds for monitoring the raw data can be done by a relatively straight forward bootstrap procedure. Thresholds for monitoring the PCA projections, however, are slightly more complicated to attain, so this is what we focus on below. The accompanying R package `tpcaMonitoring` can be consulted for all implementational details.

Complications arise for monitoring the projections because uncertainty due to estimating the principal axes from the training data has to be incorporated. If not, there will be false alarms due to estimation error rather than an actual change in the distribution. Importantly, the estimation variance of the sample principal components can not necessarily be disregarded even for high sample sizes. This is seen from the asymptotic distribution of the eigenvectors of a sample covariance matrix $\mathbf{\Sigma}$. Recall that $\{\lambda_j, \mathbf{v}_j\}_{j=1}^{d}$ and $\{\hat{\lambda}_j, \hat{\mathbf{v}}_j\}_{j=1}^{d}$ are the population and sample eigensystems, respectively. Then $\hat{\mathbf{v}}_j$ is asymptotically multivariate normal with mean $\mathbf{v}_j$ and covariance matrix

$$\mathbf{\Gamma}_j = \frac{\lambda_j}{n} \sum_{l \neq j} \frac{\lambda_l}{(\lambda_j - \lambda_l)^2} \mathbf{v}_j \mathbf{v}_j^{\intercal}, \tag{15}$$

given that the λ_j's are all distinct eigenvalues (Muirhead 1982, p. 405). Hence, if there is a small gap between two population eigenvalues, the variance can be large even for large sample sizes.

The estimation uncertainty can be incorporated by the following bootstrapping procedure:

1. Input: Training data $\{\mathbf{x}_s\}_{s=-m+1}^{0}$ assumed to be $N(\boldsymbol{\mu}_0, \mathbf{\Sigma}_0)$, b, n and α.
2. Obtain estimates $\hat{\boldsymbol{\mu}}_0$ and $\hat{\mathbf{\Sigma}}_0$ from the training data.
3. Run the TPCA algorithm (Algorithm 1) on $\hat{\mathbf{\Sigma}}_0$ to get the indices $\mathcal{J} \in \{1, \ldots, D\}$.
4. Draw a bootstrap training sample $\{\tilde{\mathbf{x}}_s\}_{s=-m+1}^{0}$, where $\tilde{\mathbf{x}}_s \overset{iid}{\sim} N(\hat{\boldsymbol{\mu}}_0, \hat{\mathbf{\Sigma}}_0)$.
5. Run Algorithm 2 on $\{\tilde{\mathbf{x}}_s\}_{s=-m+1}^{0}$ and equally distributed monitoring observations $\tilde{\mathbf{x}}_t \overset{iid}{\sim} N(\hat{\boldsymbol{\mu}}_0, \hat{\mathbf{\Sigma}}_0)$, $t = 1, \ldots, n$. One exception is that $\mathcal{J}$ of $\hat{\mathbf{\Sigma}}_0$ is reused to select projections.
6. Record $I\{T(1, b) \leq n\}$.
7. Repeat 4 - 6 B times.
8. Average the B $I\{T(1, b) \leq n\}$'s to get an estimate $\hat{\alpha}$ of $\mathbb{P}^{\infty}[T(1, b) \leq n]$.

Finally, repeat for different b's until $\hat{\alpha}$ is close enough to α within a desired margin of error.

This parametric bootstrap procedure also opens the door for other ways of robustifying the threshold; pick another distribution to bootstrap training and monitoring samples from than the normal, and run the otherwise exact same simulations. For example the multivariate t-distribution or the empirical distribution function (i.e., a nonparametric bootstrap).

A drawback of using bootstrapping to get a threshold b for TPCA monitoring is that each threshold is conditional on the exact training set, which principal axes $\mathcal{J}$ as well as the window size w. Thus, to obtain exact error control under the aassumption of a correctly specified model, a new threshold must be found by simulation for every training set. Luckily, these simulations depend most strongly on the number of projections $|\mathcal{J}|$ rather than D, making it scalable. Setting up and running these simulations is the cost of incorporating all sources of uncertainty and achieving exact error control.

4. NUMERICAL PERFORMANCE ANALYSIS

In this section, the detection performance of TPCA monitoring is assessed through an extensive simulation study. The three questions we want to answer are: In terms of EDD, how well does TPCA monitoring compare to another method that also explicitly handles sparse changes? What is gained by using TPCA compared to simply picking the least varying projections? How much can the dimension be reduced by without compromising on detection speed?

4.1 Setup

In all the simulations we present here, $n = 100$ and $\alpha = 0.01$ with 95% confidence (an ARL of approximately 10^4) and $w = 200$. The main simulation study is performed for $D = 100$ and $m = 200$, while some results for $D = 500$ with $m = 1000$ are presented briefly at the end of the results section. Four different classes of methods were run on each change scenario: The mixture procedure on the raw data with method parameters $p_0 = 0.03, 0.1, 0.3, 1$, the $J = 1, 2, 3, 5, 10, 20$ most (Max PCA) and least (Min PCA) varying projections, as well as TPCA with cutoffs $c = 0.8, 0.9, 0.95, 0.99, 0.995, 0.999$. For TPCA, change distribution (7) was used with some modifications to see if incorporating information had any effect. To be precise, we assumed knowledge about which change type was of interest, so for changes in mean, for example, we set $p_\mu = 1$ and the others to 0. In addition, we set $K \leq D/2$ to emphasize sparse changes.

All the different change scenarios were considered for 30 randomly chosen pre-change correlation matrices $\mathbf{\Sigma}_0$, with varying strengths of correlation. For each correlation matrix, a training set of $m = 200$ observations was drawn independently from $N(\mathbf{0}, \mathbf{\Sigma}_0)$. 15 matrices fall into a "low correlation" group and 15 into a "high correlation" group (Figure 3). "Low" and "high" refers to different choices of the α_d parameter in the method of Joe (2006) for generating random correlation matrices, where $\alpha_d < 1$ ($\alpha_d > 1$) yields a higher (lower) probability of large correlations in the space of correlation matrices. The α_d's are evenly spread between 1 and 50 in the "low" group, while between 0.05 and 0.95 in the "high" group.

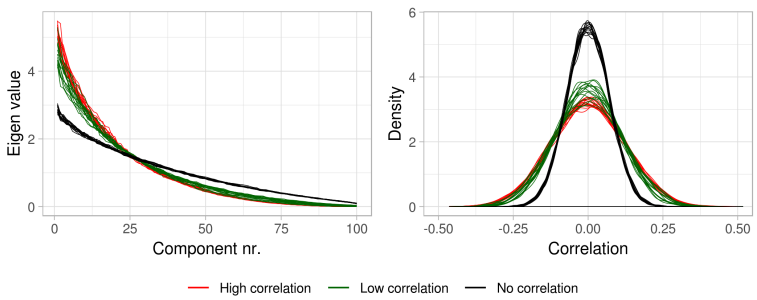

FIGURE 3. Scree plots (left) and corresponding correlation density plots (right) of the 30 random training $\hat{\mathbf{\Sigma}}_0$'s used in the simulations. As a reference for the spread of the matrices, 15 estimates based on 200 standard normal samples are also shown in black.

After a change-point at $\kappa = 0$, observations to monitor were drawn independently from $N(\boldsymbol{\mu}_1, \mathbf{\Sigma}_1)$. For all change types and sizes, the proportion of affected streams was varied over $p = |\mathcal{D}|/D = 0.02, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 0.95, 0.98$. Which $|\mathcal{D}|$ dimensions that were changed was (uniformly) randomized in every simulated change. Considered changes in the mean were $\mu_d = 0.5, 0.7, 1, 1.3$ for $d \in \mathcal{D}$, where $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_0$. To explain the changes in variance, note that any covariance matrix $\mathbf{\Sigma}$ can be decomposed into its variance and correlation part by $\mathbf{\Sigma} = \mathbf{CRC}$, where $\mathbf{R}$ is the correlation matrix corresponding to $\mathbf{\Sigma}$, and $\mathbf{C}$ is a diagonal matrix with the standard deviations $\boldsymbol{\sigma}$ on its diagonal. Using this relationship, keeping the mean and the correlation matrix constant, the affected standard deviation components were changed to

$\sigma_d = 0.5, 0.75, 1.5, 2$. Finally, correlations ρ_{di} were changed multiplicatively by factors $a_{di} = 0, 0.25, 0.5, 0.75$ for $d \neq i \in \mathcal{D}$, while $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_0$ and $\boldsymbol{\sigma}_1 = \boldsymbol{\sigma}_0$.

In total, the setup consists of a grid of 108 change scenarios (combinations of change type, change size and change sparsity). 500 simulations of each change scenario is performed, and all the 22 combinations of methods and method parameters (p_0, J or c) are run on every simulated data set to estimate the EDD. Finally, everything is repeated for each of the 30 training sets, including finding new thresholds for all the PCA-based method. This is important to have in mind to grasp the upcomming figures and results, which are compact summaries of a vast amount of simulations. Also note that the figures showing EDDs have $log(p)$ on the x-axis to highlight the sparse change scenarios.

4.2 Results

When the correlations are high, monitoring the least varying projections through TPCA or Min PCA can detect almost all the tested changes immediately with an EDD of 2-3 (Figure 4 and 5, and Table 1). Particularly, even the sparsest ($p = 0.02$), smallest changes in the mean and variance ($\mu_d = 0.5$ and $\sigma_d = 0.75, 1.5$) can be detected at this speed by monitoring only the two least varying projections. I.e., a dimension reduction of 98% can be obtained, while gaining in detection speed compared to the mixture procedure. The sparsest changes in correlation is the only notable exception, where the EDD is 100-300 time-steps, depending on the size of the change. Monitoring the most varying projections leads to considerably slower detection.

Table 1

High correlation: Average EDD per change type for each method's best method parameters (in parenthesis), as a summary of Figure 4. The average is taken over change sparsity, change size and the 15 full runs with different training sets. To display robustness, the listed method parameters are the ones that are within 1 time unit of the method's minimum average EDD.

	EDD			
Change type	Max PCA(J)	Min PCA(J)	TPCA(c)	Mixture(p_0)
Mean	27.4 (20)	1.6 (2, 3, 5, 10)	1.8 (0.8, 0.9, 0.95, 0.99 0.995, 0.999)	15.2 (0.03, 0.1)
Variance	198.9 (20)	2 (2, 3, 5)	2.1 (0.8, 0.9, 0.95, 0.99 0.995, 0.999)	8 (0.03)
Correlation	50.2 (20)	10.8 (20)	22.4 (0.999)	

As the correlations between streams become smaller, the performance of the least varying projections deteriorate (Figure 6 and 5, and Table 2). In general, 10-20 projections are needed to attain a comparable performance with the mixture procedure; slightly worse performance for the denser, larger changes and better for the the sparse, small changes. Thus, when the correlations are low, some compromise on detection speed must mostly be made to reduce the dimension, but a reduction of $80 - 90\%$ will often bring the EDD within 10 time units of the mixture procedure. The most noticable difference from the high correlation scenario occurs when the variance decreases, where the most varying projections now are the most sensitive. Note that this behaviour is in line with the two-dimensional results of Tveten (2019, p. 5). For changes in correlation when the correlations are small, we see that an even higher c than 0.999 is needed for TPCA to pick enough axes to detect the sparse changes as efficiently as Min and Max PCA with 20 projections.

In terms of detection speed alone, there is no advantage of using TPCA compared to simply picking the axes of the least varying projections as in Min PCA; more or less the same projections are monitored under both schemes. However, Tables 1 and 2 point to the fact that TPCA will automatically choose a reasonable subset of projections quite robustly with respect to the cutoff

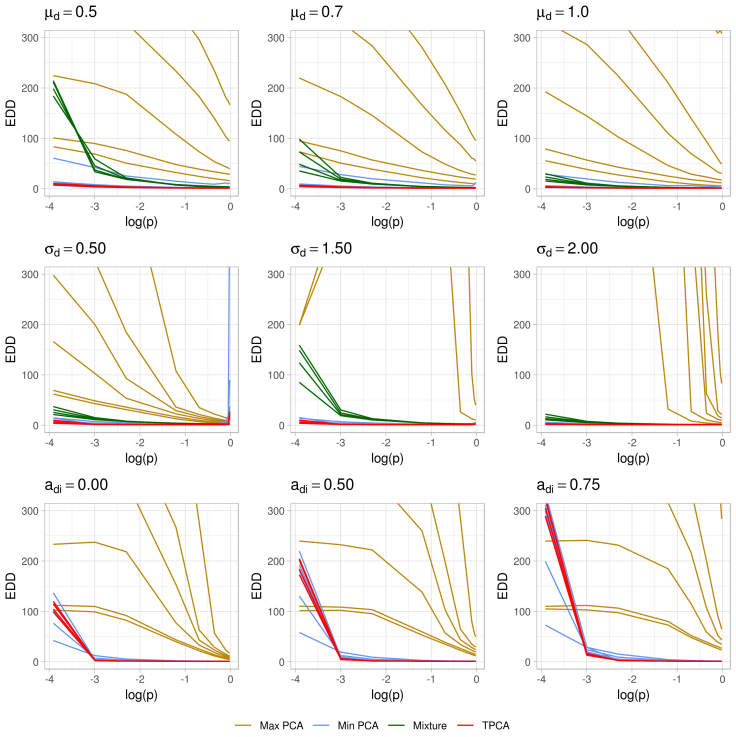

FIGURE 4. **High correlation:** EDD for changes in the mean (first row), variance (second row) and correlation (third row) of varying change sparsity and change size. Each line shows the EDD based on 500 simulations for a single method parameter, averaged over the 15 full runs with different high-correlation training sets. Note that the mixture procedure applied to the raw data can not detect changes in correlation and is thus absent from the last line of figures.

c given some knowledge about which changes are of interest. This robustness is not observed to the same degree by picking projections manually with Min PCA. The exception to this rule is for decreases in variance and changes in correlation of weakly correlated data.

Lastly, a hint towards the generalizability of these results to higher dimensions than $D = 100$ is given in Figure 7. For $D = 500$, the change sparsities tested was $p = 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.3$. Observe that TPCA and Min PCA are still able to detect the very sparse changes in the 500-dimensional data stream at almost the same speed as the sparse ones in the 100-dimensional stream.

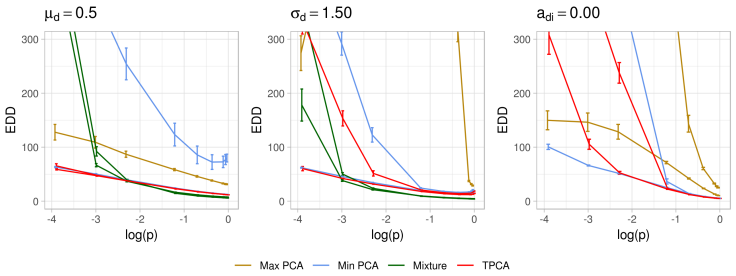

FIGURE 5. An illustration of the errors involved in Figures 4 and 6: Average EDDs with average 95% confidence intervals for a subset of method parameters. The averages are taken over all the 30 training sets, meaning that each confidence limit is an average of 30 individual limits. EDD estimates that are high or of sparse changes are generally more uncertain.

TABLE 2

Low correlation: Average EDD per change type for each method's best method parameters (in parenthesis), as a summary of Figure 6. The average is taken over change sparsity, change size and the 15 full runs with different training sets. To display robustness, the listed method parameters are the ones that are within 1 time unit of the method's minimum average EDD.

	EDD			
Change type	Max PCA(J)	Min PCA(J)	TPCA(c)	Mixture(p_0)
Mean	20.6 (20)	17.9 (20)	17.6 (0.9, 0.95, 0.99, 0.995, 0.999)	16 (0.03)
Variance	184.3 (20)	158.8 (10)	154.8 (0.995)	8.3 (0.03)
Correlation	30 (20)	28.5 (20)	52.3 (0.999)	

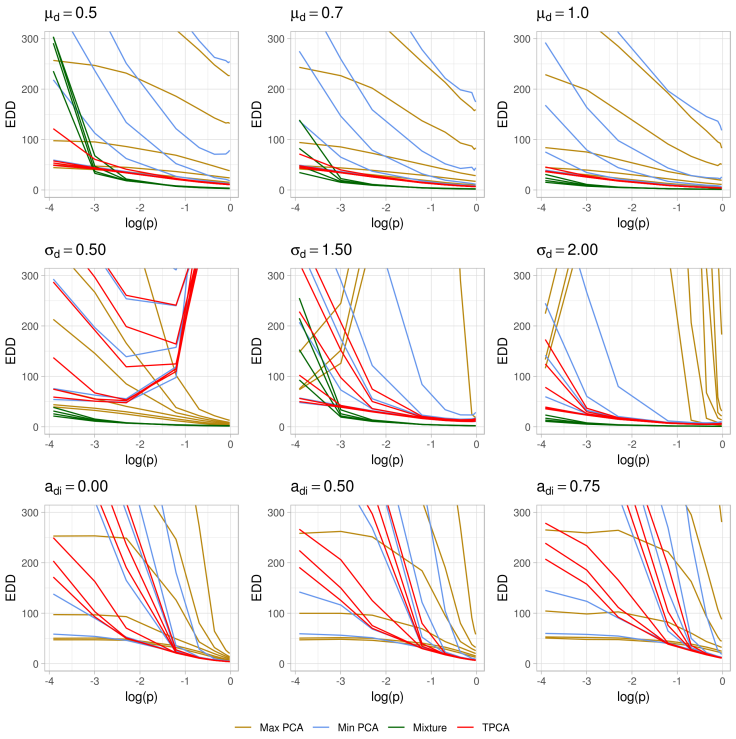

FIGURE 6. **Low correlation:** EDD for changes in the mean (first row), variance (second row) and correlation (third row) of varying change sparsity and change size. Each line shows the EDD based on 500 simulations for a single method parameter, averaged over the 15 full runs with different low-correlation training sets. Note that the mixture procedure applied to the raw data can not detect changes in correlation and is thus absent from the last line of figures.

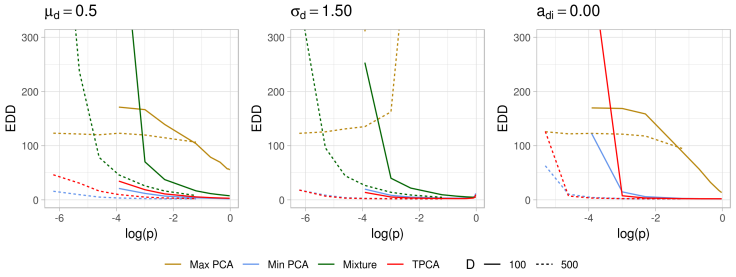

FIGURE 7. A comparison of EDDs for $D = 100, 500$ with $m = 2D$ for a single training set generated from a randomly drawn correlation matrix using $\alpha_d = 1$. Only the best choices of method parameters are shown. These figures are representative of the pattern also seen for $\mu_d = 0.7, 1$, $\sigma_d = 0.5, 2$ and $a_{di} = 0.5, 0.75$, which are omitted due to space limitations.

5. TESTING ON THE TENNESSEE EASTMAN PROCESS

In this section, TPCA is extended to handle time-dependent data and compared to dynamic PCA (DPCA) as used in the stochastic process control (SPC) litterature and the method of Kuncheva and Faithfull (2014) (what we have called Min PCA or Min DPCA below). See for example Vanhatalo et al. (2017) or Rato et al. (2016) for an introduction to DPCA in SPC. The methods are tested on the well-known Tennessee Eastman Process (TEP): a model of an industrial chemical process used to generate realistic data from a large system (Downs and Vogel 1993). As a test bed, we will use the available TEP dataset from Rieth et al. (2017), which includes fault free training sets of 500 observations as well as faulty test sets of 960 observations with a change-point at $\kappa = 160$. Each observation is a sample from the process with 3 min intervals and consists of 41 direct measurements (xmeas) and 11 controlled input variables (xmv), 52 in total. There are 500 complete test sets for each of 20 different faults. Most faults cause sparse distributional changes, where faults 1-7 are changes in mean, fault 8-12 are changes in the variance, and the rest are of various other types (Rato et al. 2016). As before, we measure the EDD of detecting these faults after a short training period, now on $m = 500$ observations. We stress that this means no extra validation set is available for fine-tuning, which DPCA generally depends on.

To let TPCA account for the highly auto-correlated TEP observations, we extended it in similar fashion as PCA is extended to DPCA. I.e., a lag l is chosen, and the observation vectors $\mathbf{x}_t$ are lag-extended to $\tilde{\mathbf{x}}_t = (\mathbf{x}_{t-l}^\intercal, \dots, \mathbf{x}_t^\intercal)^\intercal$ before they are fed to PCA. This induces a VAR model with lag l on the data. Then, a change distribution for $\mathbf{x}_t$ can be set up like before, where each simulated change now corresponds to $l+1$ duplicate changes in the parameters of $\tilde{\mathbf{x}}_t$. In this way, pre- and post-change parameters of $\tilde{\mathbf{x}}_t$ are obtained, which can be used to measure the sensitivity of the projections $\tilde{\mathbf{v}}_j^\intercal \tilde{\mathbf{x}}_t$, where $\tilde{\mathbf{v}}_j$, $j = 1, \dots, D(l+1)$, are the eigenvectors of the correlation matrix of $\{\tilde{\mathbf{x}}_t\}_{t=-m+l+1}^0$. We call this method tailored dynamic PCA (TDPCA), and we will also compare it to simply picking the J most (Max DPCA) and least (Min DPCA) varying projections, like previously. It is also possible to implement a change distribution over changes in the auto-correlations, but for simplicity we keep using change-distribution (7) and close relatives. When $p_\mu = 1$, $p_\sigma = 1$ and $p_\mu = p_\sigma = p_\rho = 1/3$, we denote the methods by TDPCA(mean), TDPCA(var) and TDPCA(unif), respectively.

The main additional challenge comes from setting a valid detection threshold when the observations are not independent multivariate normal. We tackle this by switching the parametric bootstrap procedure of Section 3 with a non-parametric block bootstrap (Kunsch 1989). Thresholds are set to meet a PFA of $\alpha = 0.01$ on $n = \kappa - l$ observations with 90% confidence, for comparisons with the empirical probability of false alarms $\hat{\alpha}$.

Vanhatalo et al. (2017, p. 10) suggest lags $l = 2, 3$ or 5 for DPCA on the TE process. In our case, using lags 2 and 3 were not sufficient to capture most autocorrelation, so we proceeded with $l = 5$ for all methods as well as in the bootstrap. This yields 312-dimensional lag-extended observations with a change-point at $\kappa - l = 155$.

Before proceding to the results, DPCA must be fit into the framework of controlling the PFA and measuring EDD. DPCA is not a change-point method, and only measures how non-conforming each new data point $\tilde{\mathbf{x}}_t$ is to the trained DPCA model by two statistics. The Hotelling's $T^2(\tilde{\mathbf{x}}_t)$ is the squared Mahalanobis distance of $\tilde{\mathbf{x}}_t$ in the DPCA model subspace, and the $Q(\tilde{\mathbf{x}}_t)$-statistic measures the orthogonal distance of $\tilde{\mathbf{x}}_t$ to the DPCA subspace (Rato et al. 2016). The corresponding stopping rule for DPCA is $T_{\text{DPCA}} = \min(T_1, T_2)$, where

$$T_1 = \inf\{t \geq 0 : T^2(\tilde{\mathbf{x}}_t) > T^2_{\alpha_n}\} \text{ and } T_2 = \inf\{t \geq 0 : Q(\tilde{\mathbf{x}}_t) > Q_{\alpha_n}\}\},$$

$T^2_{\alpha_n}$ and Q_{α_n} being percentiles of each statistic's distribution. The percentiles depend on n to fulfill the PFA requirement $\mathbb{P}^\infty(T_{\text{DPCA}} \leq n) \leq \alpha$. By assuming that false alarms are equally likely for both statistics and applying a union bound, we get that $\alpha_n = \alpha/(2n)$. This is a

slightly conservative percentile. Since we do not have a validation set to find the two thresholds precisely, we use the common approximation of T^2 being χ^2_r, where r is the number of retained components in the DPCA model, and the approximation of the Q-statistic given by Jackson and Mudholkar (1979). Since the approximation for the T^2 assumes normality and temporal independence of the retained (most varying) projections, we expect it to be overly optimistic for the TEP data, but counter-weighted somewhat by the conservative bound on α_n. Thus, it is a realistic setup in the case of no validation set, but we don't expect it to work very well in terms of false alarm control.

Lastly, in the results below, we have set the cumulative percentage of variance explained in DPCA to 95%, $J = 20$ in Min and Max DPCA, $c = 0.9$ in TPCA(mean) and TPCA(uniform), and $c = 0.99$ in TPCA(var). The reason for the cutoff-values is that these also result in approximately 20 projections being chosen, to be comparable with Min and Max PCA. Note that it is generally better to set c too high than too low, as too few projections being chosen can be detrimental, while including a few more projections than necessary only slows detection slightly. Our experiments suggest that a dimension reduction of $7 - 10\%$ is a good choice.

The results are summarized in Table 3 and Figure 8. As expected, the proportions of false alarms for DPCA is much higher than the nominal 0.01, which disqualifies it for use in this setting. The same is the case for Max DPCA, which suffers from the most varying projections being long-range auto-correlated. Among the methods that achieve appropriate error control, one of the TDPCA variants are the quickest in all cases. In particular, note that TDPCA still beats DPCA in 13 out of 20 cases despite the considerably stricter control on false alarms. When we gave DPCA the luxury of a massive validation set to find more accurate thresholds, we still found that TDPCA beats DPCA in 15 out of 20 cases. Unexpectedly, there is no systematic relation between the type of change and whether TDPCA(mean) and TDPCA(var) is best, they are mostly almost equal, and only slightly faster than TDPCA(unif). Given the results of Section 4, it is slightly surprising that TDPCA is significantly better than Min DPCA.

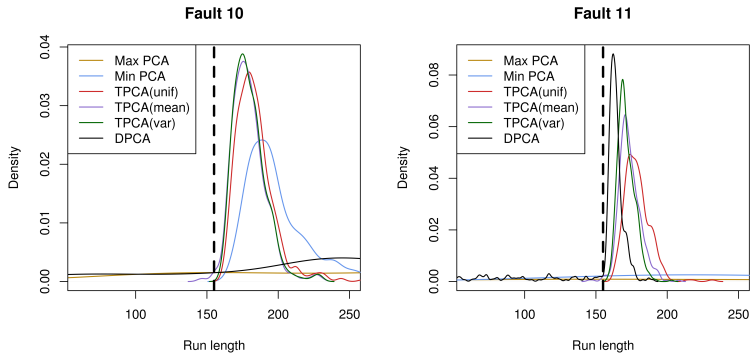

FIGURE 8. Kernel density estimates of run lengths for faults 10 and 11 of the TE process. The dashed line marks the change-point.

TABLE 3
EDD results for the TEP data. The quickest methods among those that achieve acceptable error control ($\hat{\alpha} \leq 0.01$) are in bold.

Fault	Min DPCA	TDPCA(unif)	TDPCA(var)	TDPCA(mean)	Max DPCA	DPCA
$\hat{\alpha}$	0.000	0.000	0.000	0.008	0.156	0.182
1	14.6	7.4	7.4	**5.4**	17.0	6.2
2	42.4	22.8	19.7	**17.0**	25.3	13.9
3	800.0	769.4	695.5	**663.4**	668.5	416.7
4	757.8	20.3	**9.6**	12.9	127.9	1.0
5	2.9	2.0	**1.8**	2.6	27.0	2.1
6	1.9	1.2	**1.0**	**1.0**	23.3	1.0
7	5.1	2.6	2.7	**1.9**	9.8	1.0
8	49.0	24.7	24.2	**20.0**	40.7	23.0
9	800.0	766.4	695.6	**661.8**	663.4	401.6
10	45.0	28.6	25.0	**24.6**	317.0	158.0
11	308.0	24.0	**16.3**	18.6	635.9	9.2
12	9.1	7.8	**7.3**	7.8	34.6	9.1
13	81.7	47.2	43.3	**39.6**	61.6	45.5
14	54.2	27.0	22.4	**20.1**	9.7	2.3
15	800.0	313.7	**80.8**	208.3	672.3	377.1
16	31.5	17.7	16.3	**15.5**	603.4	219.3
17	44.9	36.4	34.5	**33.5**	46.5	35.0
18	60.4	51.6	48.7	**48.0**	73.8	51.6
19	723.2	14.6	13.5	**9.8**	692.1	19.1
20	52.9	43.5	**38.4**	40.0	505.0	48.2

6. CONLUDING REMARKS

The problem of detecting sparse changes in the mean and/or covariance matrix of high-dimensional data is a problem that admits no efficient direct solution because of the number of samples necessary to estimate the covariance matrix. Monitoring projections of the incoming data onto the pre-change principal axes offers an indirect solution that is also computationally scalable. Which projections to monitor for specific distributional changes is not self-evident, and this choice is what TPCA offers an answer to. We have seen that TPCA's choice of projections work well in almost all cases studied when modelling assumptions are correct, the exception being sparse changes in the correlation, and decreases in variance when the correlations are weak. Monitoring the TPCA projections work especially well if the data streams are strongly correlated, where most changes, even very sparse and small ones, can be detected almost immediately in 100-dimensional normal data. On the other hand, if correlations are weak, some performance is lost by dimension reduction, but one still gains the ability to detect changes in correlation without loosing too much in detection speed for changes in the mean and variance compared to the benchmark mixture procedure.

On the TEP data, we saw that the dynamic version of TPCA combined with a non-parametric block-bootstrap procedure to robustly set thresholds worked well in detecting a wide range of changes quickly. Importantly, this was achieved without a large validation set, which is often needed to make the classic SPC tool DPCA work properly. For error control to be achieved, however, enough lags must be included so that the TDPCA projections are not subject to major auto-correlation. In terms of detection speed, TDPCA improves upon the method of Kuncheva and Faithfull (2014) (Min DPCA), and is also slightly quicker than DPCA in most cases. The superiority of TDPCA over DPCA is most notable when the changes are small and sparse,

whereas most changes in the TEP data are sparse, but large.

It should also be mentioned that we have only considered the question of when to raise an alarm. After an alarm has been raised, it is of course natural to ask which parameters and which dimensions/sensors that changed. This question is left for future research, but relevant litterature already exists in e.g. Hawkins and Zamba (2009) and Lakhina et al. (2004).

Other interesting follow-up questions include: (1) How does TPCA work combined with more sophisticated tools for handling time-dependence? (2) We have studied a general pre-change covariance matrix. What if it has a known structure of a certain form, like blockwise-dependence? (3) Can the insight about PCA for change detection provided here be extended to other dimension reduction tools?

ACKNOWLEDGEMENTS

This work is partially funded by the Norwegian Research Council centre Big Insight, project 237718. We would also like to thank Kristoffer H. Hellton and Steve Marron for useful discussions regarding the theory of PCA.

SUPPLEMENTARY MATERIALS

Appendix: A) Sensitivity under other change distributions, B) Dealing with indefinite post-change correlation matrices. (.pdf document)

R-package `tpca`: The TPCA routine for selecting projections (Algorithm 1). Also includes the dynamic version of TPCA used in the TEP example. (available from https://github.com/Tveten/tpca)

R-package `tpcaMonitoring`: Includes an implementation of Algorithm 2 and a single function to reproduce the entire simulation study in Section 4. The package also contains .txt files with the results from our own run of the simulation study, to quickly recreate the figures. (available from https://github.com/Tveten/tpcaMonitoring)

R-package `tdpcaTEP`: All the code to easily reproduce the TEP results. (available from https://github.com/Tveten/tdpcaTEP)

REFERENCES

Banerjee, T., and Veeravalli, V. V. (2015), "Data-Efficient Quickest Change Detection in Sensor Networks," *IEEE Transactions on Signal Processing*, 63(14), 3727–3735.

Chan, H. P. (2017), "Optimal sequential detection in multi-stream data," *The Annals of Statistics*, 45(6), 2736–2763.

Chan, L. K., and Zhang, J. (2001), "Cumulative Sum Constrol Charts for the Covariance Matrix," *Statistica Sinica*, 11(3), 767–790.

Dette, H., and Gösmann, J. (2018), "A likelihood ratio approach to sequential change point detection," *arXiv:1802.07696 [math, stat]*, . arXiv: 1802.07696.

Downs, J. J., and Vogel, E. F. (1993), "A plant-wide industrial process control problem," *Computers & Chemical Engineering*, 17(3), 245–255.

Fellouris, G., and Sokolov, G. (2016), "Second-Order Asymptotic Optimality in Multisensor Sequential Change Detection," *IEEE Transactions on Information Theory*, 62(6), 3662–3675.

Ferrer, A. (2007), "Multivariate Statistical Process Control Based on Principal Component Analysis (MSPC-PCA): Some Reflections and a Case Study in an Autobody Assembly Process," *Quality Engineering*, 19(4), 311–325.

Ge, Z. (2017), "Review on data-driven modeling and monitoring for plant-wide industrial processes," *Chemometrics and Intelligent Laboratory Systems*, 171, 16–25.

Harrou, F., Kadri, F., Chaabane, S., Tahon, C., and Sun, Y. (2015), "Improved principal component analysis for anomaly detection: Application to an emergency department," *Computers & Industrial Engineering*, 88, 63–77.

Hawkins, D. M., and Zamba, K. D. (2005), "Statistical Process Control for Shifts in Mean or Variance Using a Changepoint Formulation," *Technometrics*, 47(2), 164–173.

Hawkins, D. M., and Zamba, K. D. (2009), "A Multivariate Change-Point Model for Change in Mean Vector and/or Covariance Structure," *Journal of Quality Technology*, .

Huang, L., Nguyen, X., Garofalakis, M., Jordan, M. I., Joseph, A., and Taft, N. (2007), "In-Network PCA and Anomaly Detection," in *Advances in Neural Information Processing Systems 19*, eds. B. Schölkopf, J. C. Platt, and T. Hoffman, MA, USA: MIT Press, pp. 617–624.

Jackson, J. E., and Mudholkar, G. S. (1979), "Control Procedures for Residuals Associated With Principal Component Analysis," *Technometrics*, 21(3), 341–349.

Joe, H. (2006), "Generating random correlation matrices based on partial correlations," *Journal of Multivariate Analysis*, 97(10), 2177–2189.

Kirch, C., and Tadjuidje Kamgaing, J. (2015), "On the use of estimating functions in monitoring time series for change points," *Journal of Statistical Planning and Inference*, 161, 25–49.

Kuncheva, L. I., and Faithfull, W. J. (2014), "PCA Feature Extraction for Change Detection in Multidimensional Unlabeled Data," *IEEE Transactions on Neural Networks and Learning Systems*, 25(1), 69–80.

Kunsch, H. R. (1989), "The Jackknife and the Bootstrap for General Stationary Observations," *The Annals of Statistics*, 17(3), 1217–1241.

Lai, T. L. (1995), "Sequential Changepoint Detection in Quality Control and Dynamical Systems," *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(4), 613–658.

Lai, T. L., and Xing, H. (2010), "Sequential Change-Point Detection When the Pre- and Post-Change Parameters are Unknown," *Sequential Analysis*, 29(2), 162–175.

Lakhina, A., Crovella, M., and Diot, C. (2004), Diagnosing Network-wide Traffic Anomalies,, in *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '04, ACM, New York, USA, pp. 219–230.

Liu, K., Mei, J., and Shi, J. (2015), "An Adaptive Sampling Strategy for Online High-Dimensional Process Monitoring," *Technometrics*, .

Lorden, G. (1971), "Procedures for Reacting to a Change in Distribution," *The Annals of Mathematical Statistics*, 42(6), 1897–1908.

Mei, Y. (2010), "Efficient scalable schemes for monitoring a large number of data streams," *Biometrika*, 97(2), 419–433.

Mei, Y., Liu, K., and Zhang, R. (2017), "Scalable SUM-Shrinkage Schemes for Distributed Monitoring Large-Scale Data Streams," *Statistica Sinica*, .

Mishin, D., Brantner-Magee, K., Czako, F., and Szalay, A. S. (2014), Real time change point detection by incremental PCA in large scale sensor data,, in *2014 IEEE High Performance Extreme Computing Conference (HPEC)*, pp. 1–6.

Moustakides, G. V. (1986), "Optimal Stopping Times for Detecting Changes in Distributions," *The Annals of Statistics*, 14(4), 1379–1387.

Muirhead, R. J. (1982), *Aspects of Multivariate Statistical Theory*, New York, USA: John Wiley & Sons.

Page, E. S. (1955), "A test for a change in a parameter occurring at an unknown point," *Biometrika*, 42(3/4), 523–527.

Pimentel, M. A. F., Clifton, D. A., Clifton, L., and Tarassenko, L. (2014), "A review of novelty detection," *Signal Processing*, 99, 215–249.

Qahtan, A. A., Alharbi, B., Wang, S., and Zhang, X. (2015), A PCA-Based Change Detection Framework for Multidimensional Data Streams: Change Detection in Multidimensional Data Streams,, in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, ACM, New York, USA, pp. 935–944.

Rato, T., Reis, M., Schmitt, E., Hubert, M., and De Ketelaere, B. (2016), "A systematic comparison of PCA-based Statistical Process Monitoring methods for high-dimensional, time-dependent Processes," *AIChE Journal*, 62(5), 1478–1493.

Rieth, C. A., Amsel, B. D., Tran, R., and Cook, M. B. (2017), "Additional Tennessee Eastman Process Simulation Data for Anomaly Detection Evaluation," , . type: dataset.

Siegmund, D. (1985), *Sequential analysis*, New York, USA: Springer.

Sullivan, J. H., and Woodall, W. H. (2000), "Change-point detection of mean vector or covariance matrix shifts using multivariate individual observations," *IIE Transactions*, 32(6), 537–549.

Tveten, M. (2019), "Which principal components are most sensitive to distributional changes?" *arXiv:1905.06318 [math, stat]*, . arXiv: 1905.06318.

Vanhatalo, E., Kulahci, M., and Bergquist, B. (2017), "On the structure of dynamic principal component analysis used in statistical process monitoring," *Chemometrics and Intelligent Laboratory Systems*, 167, 1–11.

Wang, Y., and Mei, Y. (2015), "Large-Scale Multi-Stream Quickest Change Detection via Shrinkage Post-Change Estimation," *IEEE Transactions on Information Theory*, 61(12), 6926–6938.

Weese, M., Waldyn, M., Megahead, F. M., and Jones-Farmer, L. A. (2015), "Statistical Learning Methods Applied to Process Monitoring: An Overview and Perspective," *Journal of Quality Technology*, .

Woodall, W. H., and Montgomery, D. C. (2014), "Some Current Directions in the Theory and Application of Statistical Process Monitoring," *Journal of Quality Technology*, .

Xie, Y., and Siegmund, D. (2013), "Sequential Multi-Sensor Change-Point Detection," *The Annals of Statistics*, 41(2), 670–692.

Zou, C., Wang, Z., Zi, X., and Jiang, W. (2014), "An Efficient Online Monitoring Method for High-Dimensional Data Streams," *Technometrics*, .

APPENDIX A: SENSITIVITY OF PROJECTIONS UNDER OTHER CHANGE DISTRIBUTIONS

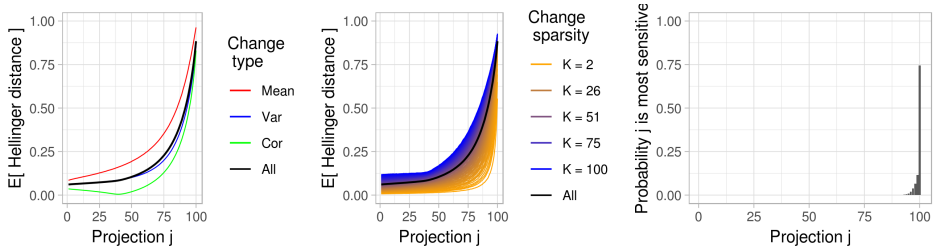

FIGURE 9. Monte Carlo estimates of $E[H_j]$ and P_j with respect to the change distribution (7) and uniformly drawn pre-change covariance matrices $\mathbf{\Sigma}_0$, with $D = 100$. (This is the same figures as shown in the main body of the article, for easier comparison with the figures below.) 10^3 randomly drawn $\mathbf{\Sigma}_0$'s were used, as well as 10^3 Monte Carlo draws from the change distribution for each $\mathbf{\Sigma}_0$

Here we present a simulation study for investigating the robustness of our results to change distribution (7), restated below for completeness.

$$
\begin{aligned}
\mathbf{C} &\sim \text{Multinom}(p_\mu = 1/3, p_\sigma = 1/3, p_\rho = 1/3) \\
K &\sim \text{Unif}\{1, \ldots, D/2\} \\
\mu_d | K, \mathbf{C} &\overset{iid}{\sim} \text{Unif}[-1.5, 1.5],\ d \in \mathcal{D} \\
\sigma_d | K, \mathbf{C} &\overset{iid}{\sim} \frac{1}{2}\text{Unif}[1/2.5, 1] + \frac{1}{2}\text{Unif}[1, 2.5],\ d \in \mathcal{D} \\
a_{di} | K, \mathbf{C} &\overset{iid}{\sim} \text{Unif}[0, 1),\ d \neq i \in \mathcal{D}.
\end{aligned}
\tag{7}
$$

In the simulations, we set $D = 100$, and draw 10^3 $\mathbf{\Sigma}_0$'s uniformly from the space of correlation matrices (Joe 2006). For each $\mathbf{\Sigma}_0$, the sensitivity to changes is assessed as in Section 2, by means of 10^3 draws from a change distribution and calculating summary statistics of the Hellinger distances. However, we now average the sensitivity results over the 10^3 $\mathbf{\Sigma}_0$'s, to get an average picture over many pre-change conditions. Each of the figures presented below are therefore averages over 10^3 figures like the ones shown in Section 2 The average sensitivity results for change distribution (7) are shown in Figure 9.

There are four alternative change distributions we look at.

- **Larger changes:** Mean interval $[-3, 3]$, standard deviation interval $[1/4, 4]$ (with the same split between decreases and increases) and correlation interval $[0, 0.5]$.
- **Smaller changes:** Mean interval $[-0.5, 0.5]$, standard deviation interval $[1/1.5, 1.5]$ and correlation interval $[0.5, 1]$.
- **Equal changes:** The same intervals as in (7), but with all affected dimensions changing with the same value. For example, only one change size μ is drawn, and $\mu_d = \mu$ for all $d \in \mathcal{D}$.

The change parameters not mentioned have the same distribution as in (7)

Figures 10 11 and 12 show the results for the larger, smaller and equal changes, respectively. Overall, the sets of figures are very similar. If one difference is to be mentioned, it is that for cutoff values c close to 1, both the larger and smaller changes would have resulted in slightly more projections being chosen by TPCA.

Lastly, Figure 13 shows the average sensitivity results with the same change distribution (7) but now with $D = 200$,

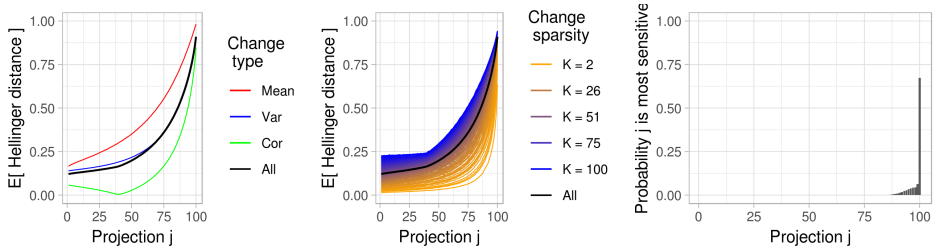

FIGURE 10. Monte Carlo estimates of $E[H_j]$ and P_j with respect to the change distribution (7) and uniformly drawn $\mathbf{\Sigma}_0$'s, but with **larger changes**

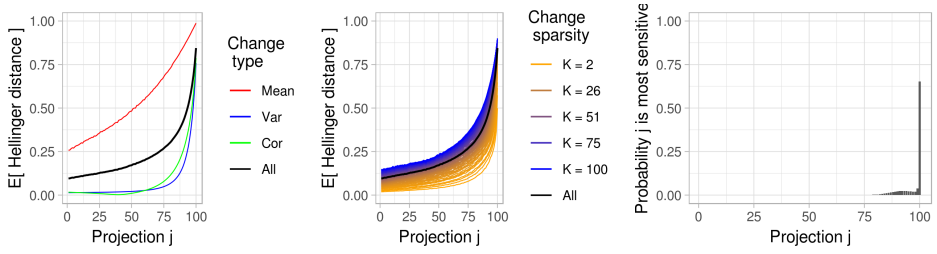

FIGURE 11. Monte Carlo estimates of $E[H_j]$ and P_j with respect to the change distribution (7) and uniformly drawn $\mathbf{\Sigma}_0$'s, but with **smaller changes**

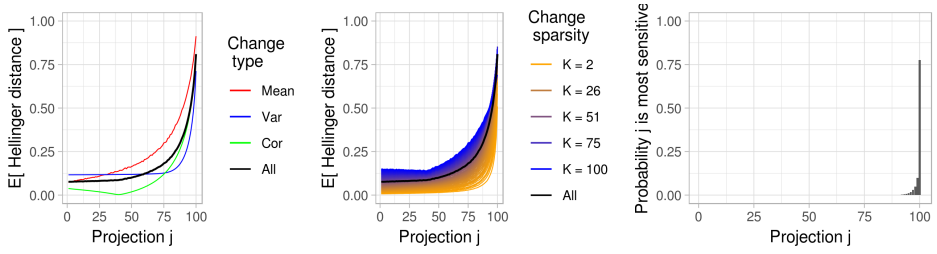

FIGURE 12. Monte Carlo estimates of $E[H_j]$ and P_j with respect to the change distribution (7) and uniformly drawn $\mathbf{\Sigma}_0$'s, but with a **equal changes** across all affected dimensions

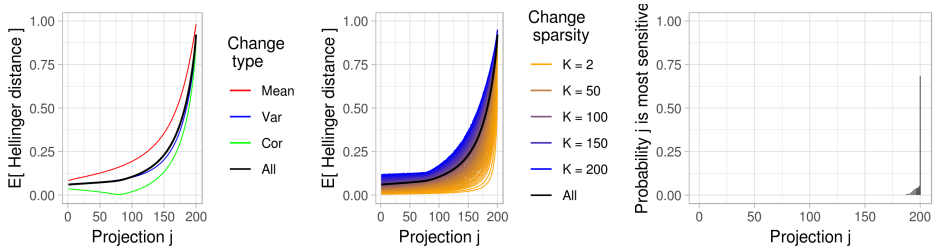

FIGURE 13. Monte Carlo estimates of $E[H_j]$ and P_j with respect to the change distribution (7) and uniformly drawn $\mathbf{\Sigma}_0$'s, but now with $D = 200$.

APPENDIX B: DEALING WITH INDEFINITE POST-CHANGE CORRELATION MATRICES

When we change entries ρ_{di} in the correlation matrix $\mathbf{\Sigma}_0$ by multiplying them with factors a_{di}, it is not guaranteed that the changed matrix is positive definite. To overcome this, we have used the function `nearPD` in the `Matrix` package of R. This is a function that finds the nearest positive definite matrix to the input matrix in sup norm. To obtain a correlation matrix, the diagonal is then put to 1.

Paper III

Real-time prediction of propulsion motor overheating using machine learning

Hellton, K. H., Tveten, M., Stakkeland, M., Engebretsen, S., Haug, O. and Aldrin, M.

Submitted for publication.

III

ARTICLE

Real-time prediction of propulsion motor overheating using machine learning

K. H. Hellton[a], M. Tveten[b], M. Stakkeland[b,c], S. Engebretsen[a], O. Haug[a] and M. Aldrin[a]

[a]SAMBA, Norwegian Computing Center, Oslo, Norway; [b]Institute of Mathematics, University of Oslo, Oslo, Norway; [c]ABB, Oslo, Norway

ARTICLE HISTORY
Compiled October 6, 2020

ABSTRACT
Thermal protection in marine electrical propulsion motors is commonly implemented by installing temperature sensors on the windings of the motor. An alarm is issued once the temperature reaches the alarm limit (H), while the motor shuts down once the trip limit (HH) is reached. Field experience shows that this protection scheme in some cases is insufficient, as the motor may already be damaged before reaching the trip limit. In this paper, we develop a machine learning algorithm to predict overheating based on past data collected from a class of identical vessels. All methods were implemented to comply with real-time requirements of the on-board protective systems with minimal need for memory and computational power. Our two-stage overheating detection algorithm first predicts the temperature in a normal state using linear regression fitted to regular operation motor performance measurements, with exponentially smoothed predictors to account for time dynamics. Then it identifies and monitors temperature deviations between the observed and predicted temperatures using an adaptive cumulative sum (CUSUM) procedure. Using data from a real fault case, the monitor alerts between 60 to 90 minutes before failure occurs, and is able to detect the emerging fault at temperatures well below the current alarm limits.

KEYWORDS
Overheating, Anomaly detection, CUSUM, Linear regression, Temperature monitoring

1. Introduction

The common safety practice for prevention of overheating in marine electrical propulsion motors is based on physically mounted temperature sensors on the windings of the motor, often by resistor temperature detection (RTD) devices (IEEE Standard 3004.8 2016). An alarm is issued once the temperature reaches the alarm limit (H) and the motor shuts down if the trip limit (HH) is reached. In the propulsion control software, there is a fixed HH limit (typically at 155 °C), that only allows for a single point of critical temperature level protection. Field experience shows that these procedures can be insufficient, in particular with respect to timeliness, as the motor may already be damaged before the trip limit is reached. Based on the mounting spot of the RTD

CONTACT K. H. Hellton Email: hellton@nr.no

sensors, the efficiency, accuracy, and timeliness of the protection system can vary – the highest temperature of the windings may be on a different location than the monitoring points. For instance, a hotspot may develop at a location where excess heat is not transferred effectively to the monitoring spots, and hence overheating might not be detected by the sensors before a critical fault has occurred. There is hence a need for more adaptive and dynamic monitoring of overheating. The modelling of excess temperature development or overheating has traditionally been based on physical models of the system utilising thermodynamics or electrical parameters (see e.g. Gnacinski 2008; Maftei et al. 2009; Lystianingrum, Hredzak, and Agelidis 2016; Pawlus et al. 2017), but these model-based approaches may be difficult to develop.

In this paper, we instead demonstrate how past data and machine learning, following a data-driven approach, can be used for timely prediction of overheating in high performance marine propulsion motors. The main target is to implement a real-time thermal protection function that can detect an abnormal state prior to reaching the HH temperature limit. We focus on using a relatively simple and transparent model, which can be easily implemented in practice, without requiring substantial computational power and memory. The monitor uses measurements that are readily available and can be implemented based on existing instrumentation on industrial grade computation engines commonly applied in the on-board system. Using data from a known fault incidence, we illustrate the usefulness of our monitor in detecting faults earlier and at lower temperatures than the standard procedures. Such data-driven approaches to condition monitoring are increasingly used for anomaly detection, fault identification and prognostics in marine vessels (Vanem and Brandsæter 2019).

1.1. System overview

We consider thermal protection for propulsion motors on ships with diesel-electric propulsion systems. The rating and dimension of these motors depend on the size and design of the ship, and the rating of the motor ranges from kilowatts to several megawatts of generated power. Protection of such motors is important both from a cost and safety point-of-view. From the cost perspective, damage to a propulsion motor may result in costly repairs or replacements, in addition to a loss or reduction of the ability to provide the intended fiscal services over a period of time. From the safety perspective, a partial loss of propulsion at a critical moment due to single motor failure may also lead to a safety hazard due to reduced manoeuvrability of the ship. The critical function of this class of motors motivated the development of the novel machine learning-based protection function described in this study.

A schematic overview of the overall system we consider can be found in Figure 1. An on-board freshwater cooling system is used to control the temperature of the motor and other units. The motor itself is cooled by air, which is circulated by one or more fans. Heat is transferred from the hot air to the water-based cooling system through a heat exchanger, as seen in Figure 1. Cooling air temperature is measured on the inlet and outlet of the heat exchanger. The rotation speed of the motor is either measured directly, or provided by the Propulsion Control Unit (PCU), which is a controller application integrated with the propulsion frequency converter controlling the speed and power of the propulsion motors. The mechanical torque and electrical power are calculated and provided by the PCU. The system overview, shown in Figure 1, is fairly generic and will be suitable for other applications beyond electrical motors and diesel-electric propulsion systems on ships. More instrumentation may be available in

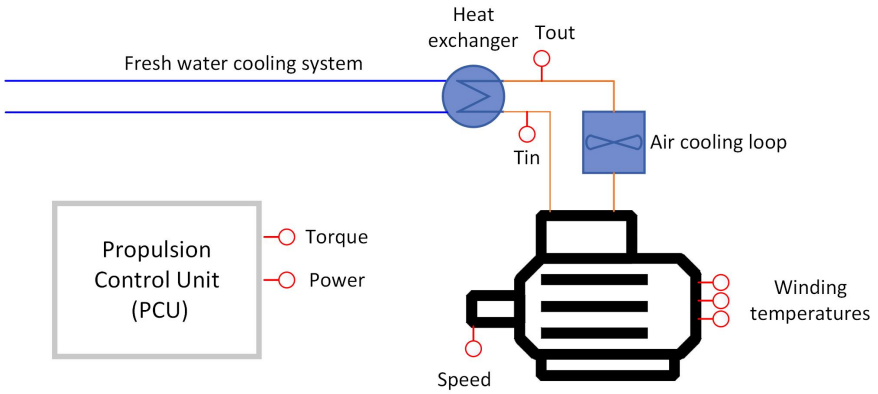

Figure 1. Schematic overview of the system. The cooling air inlet and outlet temperatures are abbreviated Tin and Tout, respectively.

some applications, but a minimum set was chosen for training and implementation, in order for the protection function developed to be as general as possible.

2. Data

2.1. Training data

The data made available by ABB consist of temperature and performance recordings from medium voltage (MV) electrical propulsion motors and the surrounding cooling system aboard four vessels of the same design. Each vessel has three different motors of two different classes, named class I and II in this paper, one motor of class I and two motors of class II. The temperature measurements (°C) on the windings of the motors were recorded by six sensors in total, located in two separate triplets of windings; referred to as U_1, V_1, W_1, and U_2, V_2, W_2. Several variables of the performance and the motor's surrounding cooling system could be measured, but the following subset of variables was recorded consistently across all the vessels: power of the motor (% of maximum nominal power), speed of the motor (% of maximum nominal speed), and mechanical torque (% of maximum nominal torque), together with the inlet temperature (°C), and the outlet temperature (°C) of the cooling air in the cooling system.

2.2. Preprocessing

The data from each vessel and motor were collected at different time periods in 2017 and 2018. The time periods of collected data vary for the different vessels: 125 days, around 4 months, for the first vessel; 80 days, around 2.5 months, for the second vessel; 294 days, around 10 months, for the third vessel; and 262 days, around 9 months, for the fourth vessel.

The data were collected using the ABB Remote Diagnostics System (RDS), an edge device whose function is to collect data from the on-board devices and transfer it via

a satellite link to cloud storage, enabling remote troubleshooting and data analysis. The on-board protection systems collect data at regular, sub-second intervals, while the RDS queries the data using different data collection schemes. In the considered training data, all measurements were collected using an asynchronous sampling regime, where the values were polled at regular intervals, but only stored when the difference between the current and the previously stored value exceeded a given threshold. The threshold is configured differently for the different measurements. Under the asynchronous setup, substantial amounts of data are collected during dynamic periods, while none or few samples are available during stationary periods, for instance at zero power, or when running at a constant power over a long period of time. The main strength of the asynchronous sampling regime is a reduction in storage capacity and bandwidth for data transfer during stationary periods, while still being able to record data with a relatively high bandwidth in dynamic periods (Losada, Rubio and Bencomo 2015). The main weakness is reduced robustness to data gaps, as missing measurements cannot be separated from stationary periods without additional information.

The temperature, performance, and cooling system measurements were thus recorded at non-uniform time intervals with gaps ranging from milliseconds to several hours or days. To obtain regularly sampled data, required for the machine learning analysis, all recordings were mapped onto a regular grid with 1 second sampling time, over the range spanned by the timestamps of all the measurements. The synchronisation of the temporal scale was applied to each ship separately as the collected data covered different periods. With several recordings within the same second, the last observed value was chosen.

To impute the missing values in the regular time-aligned data, we applied a last value carried forward (LVCF) interpolation principle, separately for each variable. This complies with the asynchronous sampling regime in that a sequence of non-recorded measurements will be replaced by its previous recorded value. Alternatively, given a synchronous sampling regime or a combination of asynchronous and synchronous measurements, a linear interpolation approach could have been used. There are, however, also instances of missing data due to malfunctioning registration. We, therefore, determined an upper threshold of 48 hours for the length of the periods to be interpolated. The value of 48 hours was chosen to cover periods of long-distance voyages with stable conditions observed in the data. In addition, for the slowly changing temperature measurements, interpolation was only carried out if the difference between measurements at either end of the period was below 3°C. Changes in temperature larger than this threshold suggested that the missing observations were due to malfunctioning, and not the asynchronous sampling procedure. For the rapidly changing measurements, i.e. power, speed, and torque, a reasonable restriction could not be defined and interpolation was performed regardless of the change in value.

Before the synchronisation of the data, there were approximately 650 000 measurements for the temperature measurements and 3 million observations for the power, speed, and torque measurements. After synchronisation, but before interpolation, there were around 450 000 complete observations with registered measurements in all variables. After interpolation, there were around 78 million complete observations.

Finally, overheating can only occur when the motor is in fact running, and predicting the motor temperature at zero power corresponds to predicting ambient temperature. Therefore the final preprocessing step was to censor all observations where the power was approximately zero, set at the practical limit of the power being less than 1%.

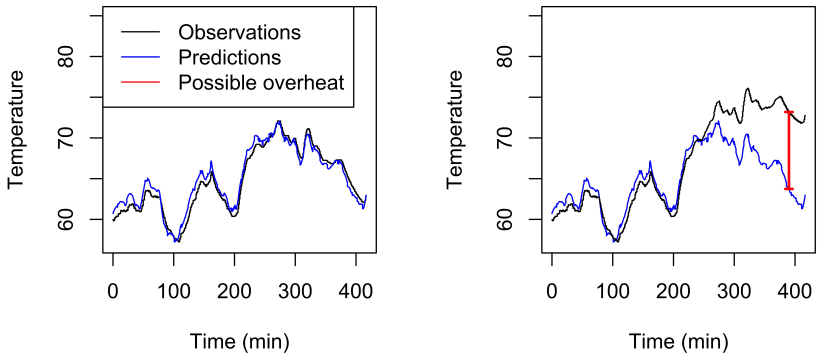

Figure 2. Illustration of the general framework for overheating detection. The left panel shows the observed and predicted temperatures under normal conditions. The right panel shows an observed temperature significantly exceeding the predicted temperature, indicating a possible overheating event.

3. Methods

Our general framework for detecting heat development is to compare a prediction of the temperature in a normal state to the actually observed temperature, and monitor deviations between the two. If the observed temperatures are significantly higher than the predicted temperatures, we may suspect overheating. The approach is illustrated in Figure 2. The left panel shows the observed and predicted temperatures under normal conditions, where they largely agree. The right panel, on the other hand, shows a hypothetical scenario where the observed temperature significantly exceeds the predicted temperature, indicating a possible overheating event. The novel contribution is to predict the winding temperature using a machine learning model to emulate the physical system. The prediction is based on available training data of motor performance and cooling system measurements under normal conditions to describe how the temperature should behave.

We therefore propose a framework for overheating detection consisting of two parts:

(1) first build a predictive machine learning model for the winding temperature based on observed historical data,
(2) then monitor and detect deviations in the observed winding temperatures from the predicted normal state values. When deviations exceed a certain threshold, an alarm is issued.

In the predictive step, we train the machine learning model to predict the average motor temperature T_m (averaged over all windings) to ensure generalisability as the locations of sensors and windings may differ across vessels and motors. In the detection step, we monitor the deviations of the observed temperature on the individual windings, in order to detect overheating events as early as possible. All analyses in the study were performed using the statistical software `R`.

3.1. Modelling temperature using machine learning

The first step is to train a machine learning model to emulate the physical system of the motor. We use ordinary least squares (OLS) linear regression (Chambers 1992; Hastie, Tibshirani and Friedman 2001) as the machine learning algorithm, to comply with the practical constraints of the on-board protective system with limited computational power and memory. The OLS model predicts the average motor temperature T_m, or the output y_t, at time t as a linear function of M input variables $x_{i,t}$ at time t:

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_M x_{M,t} + \varepsilon_t,$$

with a noise term ε_t. The initial input variables are the power, speed and torque of the motor and the cooling air inlet temperature. In addition, the cooling air outlet temperature is also available, but the role and subsequent exclusion of the cooling air outlet temperature from the considered model is discussed in Section 5.

The model is fitted by least squares using the QR factorisation method (Hansen, Pereyra and Scherer 2013). To optimise the predictive ability of the final model, we assess reasonable transformations of the input variables and select the best in terms of prediction error.

3.1.1. Transformation of input variables

We first determine the relevant transformations of the input variables for the OLS regression algorithm. Expert knowledge regarding the physical system of the motor guides the inclusion of relevant transformations. First, it is assumed that the impact of the motor speed and torque on the winding temperature would be the same irrespective of the direction of rotation, such that only the absolute values of the speed and torque variables are considered as inputs. Further, the power of the motor is always positive.

The motor performance measurements of speed, torque, and power are characterised by large and abrupt changes, as seen in the left panel of Figure 3. The individual sensor and mean temperatures are, on the other hand, slowly varying measurements as the motor, being a block of metal, heats up through conduction. We therefore consider time-lagged, smoothed transformations of the volatile input variables, as such smoothed variables will be more informative of the temperature, accounting for the time dynamics of the system. For our purpose, exponential smoothing, or exponentially weighted moving average (EWMA) (Brown 1956; Holt 1957), is seen to be a good choice for constructing such lagged input variables. Importantly, exponential smoothing can be recursively defined, requiring minimal memory, such that the transformation is easily implementable in an industrial real-time system. Alternative smoothing approaches, such as fixed-window moving averages, would in comparison require more memory.

EWMA smooths the time series using an exponential window function. It temporally lags the original input variable u by recursively adding current variable values to previous aggregates, multiplied by a smoothing factor $0 < \theta < 1$. More formally, the exponentially smoothed input variable, x_t at time-step t, of the original input variable, u_t, is given by

$$x_t = (1 - \theta)x_{t-1} + \theta u_t, \quad u_0 = x_0.$$

The smoothing factor θ determines the time constant of the system, τ, where the

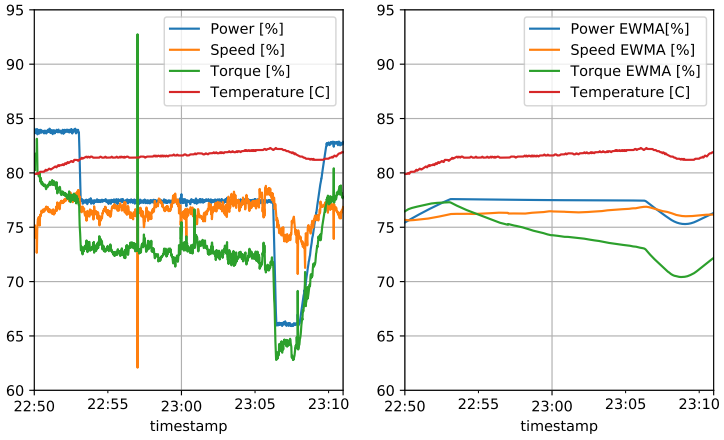

Figure 3. The left panel shows an example of abrupt changes in speed, power, and torque compared to the slowly changing mean temperature at an occurrence of acceleration of speed and torque. The right panel shows the corresponding exponentially smoothed variables of speed, power, and torque, together with the non-smoothed mean temperature.

relationship between θ, τ, and the sampling interval ΔT is given by

$$\theta = 1 - e^{-\Delta T/\tau} \simeq \frac{\Delta T}{\tau}, \quad \tau \gg \Delta T.$$

The time constant, τ, of an exponential moving average is hence given by $\tau = \Delta T / \log(1-\theta)$, and represents the amount of time it takes the smoothed response of a unit set function to reach 63.2% of the original signal. The EWMA characterises the solution to a first-order ordinary differential equation, and therefore gives a good approximation to physical systems such as the heat transfer models.

When constructing the exponentially smoothed variables, a mechanism for handling the remaining missing or censored observations is needed. We choose to reset the exponential smoothing if an observation x_k is missing, meaning that the smoothing is initialised by setting y_0 equal to the first value after the missing observations. After a reset, the smoothing requires time to stabilise, such that the 30 first minutes are subsequently censored.

3.1.2. Selection of input variables

We then select the best input variables to be included in the OLS algorithm. As prediction is our main aim, we evaluate the different models based on predictive ability using cross-validation. Standard model selection approaches, such as evaluating Akaike's information criterion (AIC) would be less practical due to the large number of observations (Claeskens and Hjort 2008). In cross-validation, different parts of the training data are consecutively held out from the model fitting and predicted based on the remaining data. The prediction error is then assessed by the root mean squared error (RMSE) averaged over all parts.

Table 1. Estimated parameters for the temperature prediction models. The time constant τ is measured in minutes.

Variable	Class I motor	Class II motor
Constant	$2.7 \cdot 10^{1}$	$2.4 \cdot 10^{1}$
TaIn	$8.4 \cdot 10^{-1}$	$7.8 \cdot 10^{-1}$
Power^2 ($\tau = 28$)	$-3.5 \cdot 10^{-3}$	$1.0 \cdot 10^{-3}$
Speed ($\tau = 28$)	$-4.0 \cdot 10^{-1}$	$-9.6 \cdot 10^{-2}$
Speed^2 ($\tau = 28$)	$5.7 \cdot 10^{-3}$	$2.2 \cdot 10^{-3}$
Torque^2 ($\tau = 28$)	$9.9 \cdot 10^{-3}$	$5.4 \cdot 10^{-3}$

In our setting, the part of the training data held out could comprise either the data for one whole vessel, one motor class of a vessel, or one individual motor. Due to differing operating modes and varying sea conditions, the variability in motor operation between different vessels is substantially larger than the variability between motors within the same vessel. The two class II motors are, in addition, likely to run in the same mode within the same vessel. As we specifically aim to assess how the predictive performance generalises to a previously unobserved vessel, we perform cross-validation leaving out each vessel, i.e. the single class I motor and the pair of class II motors, in each cross-validation iteration. The cross-validation scheme therefore holds out all three motors in one vessel for each iteration. As the amount of available data varies between vessels, a weighted version of the root mean squared error is used. The vessel-specific weights equal the proportion of observations for each vessel of the total data, separately for the motor classes I and II. We further assume the physical systems of the class I and II motors to be equal, but that they may run under different operating regimes. The same input variables are therefore used for all motors, but with parameters estimated separately for the two motor classes.

We follow a forward and backward step-wise model selection strategy (Hastie, Tibshirani and Friedman 2001), testing increasingly complex models and comparing them in terms of the cross-validated RMSE. The variables are included in the following hierarchy:

(1) Cooling air inlet temperature
(2) Linear power
(3) Squared power
(4) Linear speed
(5) Squared speed
(6) Linear torque
(7) Squared torque
(8) Interaction between linear speed, power and torque

All power, speed, and torque terms were included as exponentially smoothed variables as investigations showed that non-smoothed input variables always gave worse prediction performance. Details on the separate cross-validation errors for each step, when including different input variables, are provided in the Supplementary material. After a final backward step, excluding the variables not improving the prediction, the final model uses five input variables: cooling air inlet temperature, exponentially smoothed squared power, linear and squared speed, and squared torque.

As part of the model selection procedure, a conditionally optimal time constant, τ, was initially estimated separately for each input variable. To facilitate a physical interpretation of the model, all τ values were fixed to the same value, found to be $\tau = 28$ min by minimising the cross-validation RMSE. The model with a common

time constant was seen to give only slightly worse prediction performance than the individual time constant model, see the Supplementary material for further details. The class I and II motor model fits are summarised in Table 1. We note that the sign of the estimated effect of the squared power differed between the models, which is likely due to the high correlations between the different input variables. The adjusted R^2 of the two models is 0.941 and 0.931, respectively. This close match between our predictive model and the observed data can be seen in the illustration in the left part of Figure 2.

3.2. Fault detection algorithm

Given the prediction models for the normal state of the system, the second step is to monitor the deviations between the observed temperature and the predicted temperatures. We propose an online monitoring algorithm for the temperature deviations based on the framework of Lorden and Pollak (2008) and Liu, Zhang and Mei (2017). We further develop a novel tuning procedure to automatically select the parameters of the monitoring algorithm.

The prediction models are trained on the mean temperature, but we monitor the observed temperature deviations on the *individual* sensors to detect overheating events as early as possible. For the fault detection algorithm, the aim is to detect, as quickly as possible, whether any of the temperatures in N sensors suddenly rises to an abnormally high level compared to the prediction. In our case, the number of sensors is equal to the number of individual windings, $N = 6$.

We use the notation $y_{j,t}$ for the observed temperature of the individual sensor $j = 1, \ldots, N$ at time t, and $\hat{y}_t$ for the predicted average temperature across the sensors at time t produced by the models in Table 1. The deviations of the observed temperature from the predicted temperature at time t, referred to as the residuals, are then given by

$$e_{j,t} = y_{j,t} - \hat{y}_t, \quad j = 1, \ldots, N.$$

The goal is to detect whether the mean of the residual distribution for any of the sensors has changed sufficiently far from 0 in the positive direction. Such a large deviation is shown schematically in the right panel of Figure 2.

We monitor each sensor using a local monitoring statistic, $z_{j,t}$, which is a function of the temperature residuals of the jth sensor up until time t: $e_{j,1}, \ldots, e_{j,t}$. We then construct a global monitoring statistic for all sensors, G_t, by applying a set of filtering or shrinkage functions, $h_j \geq 0$, on the local monitoring statistics of each sensor and summing their individual contributions,

$$G_t = \sum_{j=1}^{N} h_j(z_{j,t}). \tag{1}$$

Finally, the global monitoring statistic, G_t, is compared to an alarm threshold, b, where the alarm is raised when the statistic exceeds the threshold value.

The detection algorithm is required to detect true faults quickly with as few false alarms as possible and to detect faults that are only visible in a single sensor. At the same time, we need the algorithm to be computationally efficient and conceptually simple. Further, it should also generalise to different motors and vessels without

motor-specific tuning. As also stated earlier, the simplicity of the monitoring system is important for the operator's understanding of the system and for implementation, as the monitoring system is coded in the on-board vessel system and must be able to run in real-time. We specifically select the local monitoring statistic $z_{j,t}$ and the filtering functions h_j of the detection algorithm to comply with these criteria.

3.2.1. Choice of monitoring statistic

For the local monitoring statistic, we use the adaptive cumulative sum (CUSUM) statistic introduced by Lorden and Pollak (2008). We choose the adaptive CUSUM because of its simplicity and computational efficiency, in addition to a proven ability to detect distributional changes of unknown magnitude quickly. Alternative monitoring statistics include the standard CUSUM statistic (Page 1954, 1955), the EWMA control chart (Roberts 1959) or other sequential change-point detection statistics (Basseville and Nikiforov 1993).

We assume the residuals to be independent and normally distributed, such that the adaptive CUSUM statistic is given by

$$z_{j,t} = \max\left(z_{j,t-1} + \hat{\mu}_{j,t}e_{j,t} - \frac{1}{2}\hat{\mu}_{j,t}^2, 0\right), \tag{2}$$

for each sensor j. In addition, the overall distribution of the residuals is standardised to have a mean of 0 and a standard deviation of 1. The values $\hat{\mu}_{j,t}$ are adaptive means, recursively estimated for each sensor, given by

$$\hat{\mu}_{j,t} = \max\left(\frac{s_{j,t}}{n_{j,t}}, \rho\right), \quad s_{j,t} = \begin{cases} s_{j,t-1} + e_{j,t-1}, & z_{j,t-1} > 0, \\ 0, & z_{j,t-1} = 0, \end{cases} \tag{3}$$

where $n_{j,t} = n_{j,t-1} + 1$, if $z_{j,t-1} > 0$, and otherwise $n_{j,t} = 0$, if $z_{j,t-1} = 0$, and with initial values $z_{j,0} = s_{j,0} = e_{j,0} = 0$. Note that when $s_{j,t} = n_{j,t} = 0$, we define $s_{j,t}/n_{j,t} = 0$.

The adaptive means and the monitoring statistic $z_{j,t}$ are therefore dependent on a user-determined parameter, $\rho > 0$, representing the smallest relevant change. If there is evidence that a change occurred, such that $z_{j,t-1} > 0$, the mean is estimated by a recursively updated average. The update starts from a candidate change-point given by the most recent time i where $z_{j,i} = 0$ for $1 \leq i \leq t-2$. If there is no evidence of a change, such that $z_{j,t-1} = 0$, the average is reset to 0. The statistic will further ignore irrelevant changes when ρ is selected appropriately. When the monitoring statistic is zero, $z_{j,t-1} = 0$, the consecutive value only increases, $z_{j,t} > 0$, if $e_{j,t} > \rho/2$.

The assumed normality and independence of the residuals is an oversimplification, resulting in a misspecified model. But due to the large amounts of available training data and the fact that the temperature faults of interest correspond to large changes in the mean, the simplistic model still yields good results in practice. Further, the threshold b is set based on the number of false alarms in the training data, irrespective of the model assumption of the CUSUM statistic. The model misspecification therefore does not result in loss of control of the false alarms, but rather speed of detection. Given that the relevant changes in the means are relatively large, any improvement in timeliness achieved by applying a more complex residual model appears to be small.

In the standard CUSUM, changes in the mean have to be pre-specified, and the recursive estimation of the mean in the adaptive CUSUM is more flexible. The adaptive

CUSUM is therefore less prone to degrading performance due to a misspecified μ compared to the standard CUSUM, and it achieves two goals simultaneously: ρ may be specified at the lowest possible level to filter out all small, non-relevant changes, while at the same time maintaining near optimal detection speed for changes of mean greater than ρ. This point is further discussed in Section 3.2.3. This feature is important when only one fault is available for testing, as new faults will be different in terms of the size of the change in mean. In the standard CUSUM, μ must be balanced between these two goals, not being optimal for any of them. The standard CUSUM is also prone to overfitting μ to the observed fault, suggesting that the adaptive CUSUM generalises better to other vessels and faults.

3.2.2. Choice of global monitoring statistic

For the global monitoring statistic, we use the maximum over all sensors

$$G_t = \max_j(z_{j,t}), \quad j = 1, \ldots, N, \tag{4}$$

which is given by the order-thresholding filtering function, $h(z) = z\mathbf{1}\{z \geq z_{(1)}\}$, where $z_{(1)}$ is the largest order statistic or maximum of $z_1, \ldots, z_N$. The order-thresholding applied to the sum in Equation (1) truncates the terms not corresponding to the maximum to zero. Alternative filtering functions such as hard- and soft-thresholding, $h(z) = z\mathbf{1}\{z \geq a\}$ and $h(z) = \max(z - a, 0)$, respectively, depend on an additional constant a.

The max function is chosen to allow for quick detection of faults affecting only a single sensor, i.e. emerging hotspots, as it is known to be more efficient than the sum or average of sensors for such faults (Mei 2010; Xie and Siegmund 2013; Liu, Zhang and Mei 2017). Soft- or hard-thresholding may yield faster detection speed for faults affecting all sensors (Liu, Zhang and Mei 2017), but as the max filtering does not introduce additional tuning parameters, it allows for better generalisability to new and previously unobserved vessels. The fault detection algorithm is summarised in Algorithm 1.

3.2.3. Setting the detection threshold and minimum change size

To apply the detection algorithm in practice, we are required to determine the detection threshold, b, and the minimum change size, ρ. These parameters are tuned to detect a fault as early as possible, while controlling the number of false alarms, and at the same time setting ρ as low as possible without severely compromising the detection speed. The latter counteracts overfitting due to the limited number of faults, only one single incident, and can therefore improve generalisability.

The detection threshold, b, is set relative to the number of acceptable false alarms, m, in the fault-free training data. We define a potential false alarm event as the contiguous time-points where the statistic G_t raises above 0 for a certain period of time, before going back to 0 again. What governs the threshold b is the maximum value of G_t in each such region. To be precise, if I_j for $j = 1, \ldots, k$ denote the k potential false alarm events in the training data, then the value of G_t at the peak over each interval is given by $\hat{G}_j = \max_{i \in I_j} G_i$. A threshold can then be obtained by setting b to the $(m+1)$th largest $\hat{G}_j$. As the threshold depends on both ρ and m, we use the notation $b(\rho, m)$ when it is useful to make this dependence explicit.

The time of an alarm corresponds to the first time the monitoring statistic exceeds

Algorithm 1 Maximised adaptive CUSUM for temperature fault detection

Input: ρ, b

$t = z_{j,0} = s_{j,0} = n_{j,0} = 0$ for $j = 1, \ldots, N$.
while $\max_j z_{j,t} < b$ **do**
  $t = t + 1$.
  Input set of standardised temperature residuals $(e_{1,t}, \ldots, e_{N,t})$.
  for $j = 1, \ldots, N$ **do**
    if $z_{j,t-1} > 0$ **then**
      $s_{j,t} = s_{j,t-1} + e_{j,t-1}$.
      $n_{j,t} = n_{j,t-1} + 1$.
    else
      $s_{j,t} = n_{j,t} = 0$.
    end if
    $\hat{\mu}_{j,t} = \max\left(\frac{s_{j,t}}{n_{j,t}}, \rho\right)$.
    $z_{j,t} = \max\left(z_{j,t-1} + \hat{\mu}_{j,t} e_{j,t} - \frac{1}{2}\hat{\mu}_{j,t}^2, 0\right)$.
  end for
end while

Return: t

the threshold, denoted as a function of ρ and m by

$$A(\rho, m) = \min\{t \geq 1 : G_t > b(\rho, m)\}.$$

Given that F is the time of a true fault, detecting the fault as early as possible while allowing m false alarms, can be formulated as maximising

$$T(\rho, m) = F - A(\rho, m),$$

with respect to ρ for a given m. As multiple values of ρ and corresponding thresholds $b(\rho, m)$ may achieve approximately the same time to failure T, we select the smallest ρ maximising T within a user-specified error margin δ:

$$\hat{\rho}(m) = \min\left\{\rho > 0 : \max_{\rho > 0}\left\{T(\rho, m)\right\} - T(\rho, m) \leq \delta\right\}.$$

The corresponding threshold for a specific number of false alarms m is then given by $b(\hat{\rho}, m)$. A grid search over ρ was used to find an approximately optimal $\hat{\rho}$.

The error margin δ is introduced to reduce overfitting of ρ to the one single fault. We experienced that an error margin of 1 second resulted in ρ being set too high for possible future faults, because a slightly higher ρ resulted in a few seconds quicker detection. With δ, one can specify, for example, that all detection times within 60 seconds of the optimal detection time are good enough. We found that a δ of three minutes provided a decent counter-balance to maximally overfitting ρ to our single fault case presented in the next section.

4. Performance on real failure case

In this section, we present the results of the detection algorithm 1) applied to a real overheating failure case. The failure occurred in one of the vessels available in the training data, following the system described in Section 1.1, but outside the training period. The upper panel of Figure 4 shows the residuals of the $N = 6$ temperature sensors in the four hour period before the motor fails. The lower panel of Figure 4 shows in the same period the corresponding individual CUSUM statistics (gray lines) and the maximised adaptive CUSUM statistic (black line) following Algorithm 1. The time of the motor failure is indicated by the red line. The missing values in the residuals and the CUSUM statistic are due to the motor being shut off, resulting in zero power, and the subsequent initialisation of the exponentially smoothed variables (removing 30 minutes of observations). In the lower panel of Figure 4, the detection times prior to the failure allowing for $m = 0, 1, 2$ and 3 false alarms in the training data are shown. The adaptive CUSUM statistics are shown for $\rho = 13$, although the optimal alarm for a given number of false alarms is set at different values ranging from 12.4 to 17.8, obtained by the procedure described in Section 3.2.3.

If no false alarms are allowed in the training period, the fault can be detected 57 minutes before the motor failure (green line, corresponding to $\rho = 17.8$). The alarm is raised when the mean temperature is 104.2°C and the maximum temperature over the six sensors is 111.3°C. By allowing for one and two false alarm events in the training period, the detection time remains approximately the same (56 minutes), but ρ may be lowered to 16.2 and 12.4, respectively, following the tuning strategy in Section 3.2.3. The lower ρ values improve the generalisability of the monitoring algorithm to new faults. Finally, if we allow for three false alarms in the entire training period, the fault may be detected already 86 minutes prior to the failure (blue line, corresponding to $\rho = 17.2$). The alarm is then raised when the mean temperature is 91.0°C and the maximum temperature over the six sensors is 98.3°C. Further allowing for four to ten false alarms did not change the detecting time, but lowered the optimal ρ value.

5. Discussion

We have demonstrated how a data-driven approach of using past data and machine learning can provide timely prediction of overheating in marine vessels. Based on assessing a real failure case, our proposed alarm algorithm may detect a fault between 60 and 90 minutes *before* the actual occurrence and at temperatures well below the current alarm limits, depending on the number of allowed false alarms. By using a machine learning approach, one can capture predictive relationships, interactions or feedback-loops, that may be unknown or non-intuitive to experts of the physical model. Physical knowledge was only included in the model building step to guide the selection and transformation of candidate input variables considered by the model.

The aim of this work was to use data from four vessels to create two models (class I and II) that could be used on all vessels in the fleet. The model was hence trained on the average winding temperature, while the monitoring algorithm itself is implemented on each individual winding. It should therefore be noted that better performance could probably be obtained by training winding-specific models. For all motors, the winding temperatures had a variation of around 3–4°C, which appears to be consistent for each motor, but no pattern could be found across motors. The bias is, therefore, likely caused by installation or manufacturing effects, and individual models for each winding

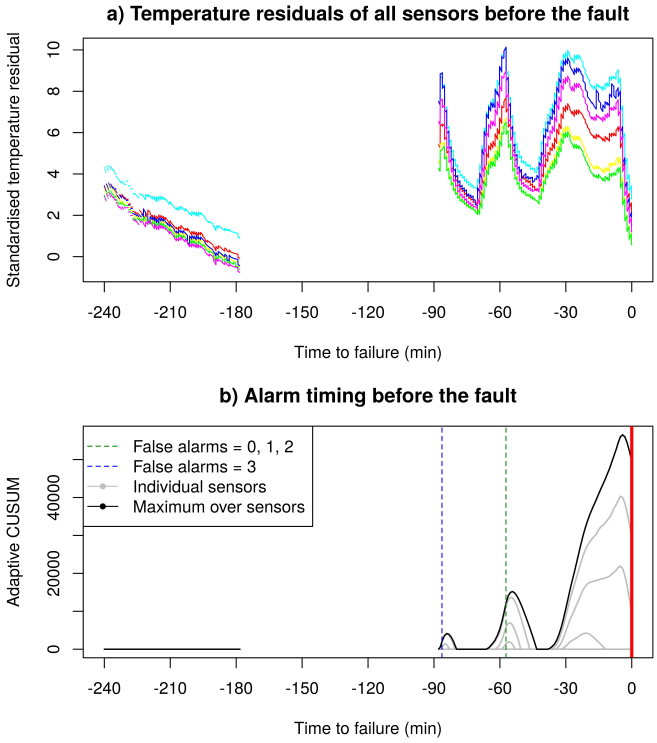

Figure 4. **a)** The residuals of each of the six temperature sensors before the fault. **b)** The corresponding adaptive CUSUM statistics per sensor $z_{j,t}$ (gray lines) and their maximum G_t (black lines). The detection times prior to the fault for $m = 0, 1, 2$ accepted false alarms in the training data is indicated by the green dashed line and $m = 3$ by the blue dashed line. The time of the motor failure is shown by the red line. For illustrational purposes, the adaptive CUSUM statistics are shown for a single value, $\rho = 13$, though the alarms are given for different values depending on m, $\hat{\rho}(m) = 17.8, 16.2, 12.4, 17.2$ for $m = 0, 1, 2, 3$, following the procedure described in Section 3.2.3.

would be able to filter out these biases and hence improve the monitor performance. However, as these differences are motor-specific, this would require retraining of the model for each new vessel.

A drawback of the model-based approach is that data is needed to build the models for a specific cooling system configuration and motor type. Data is needed both for building machine learning models, and for parameter identification in the case of a physics-based model. The model cannot be implemented on a new configuration directly. However, we believe that the selection of parameters and methodology could be applicable to systems similar to this particular class of identical vessels.

Based on a single fault, it may be difficult to assess how the probability of detection and the detection time will generalise to other overheating events. Further assessments of the procedure is therefore needed, both on more vessels and faults. Importantly, both a larger number and a wider range of faults should be used to validate how well the detection framework generalises beyond the current fault case. If a substantial number of fault cases can be obtained, machine learning models may also be applied directly to

predict alarms, instead of monitoring the deviation from the normal state. Additional fault cases may also improve the estimates of the probability of detection and the timeliness of our procedure.

For the prediction of the motor temperature, there were several models, or combinations of input variables, that gave similar or identical prediction performance. It is reasonable to expect that the exact ordering of the different models would change if more data were included. The final model is likely to depend on the order of which the input variables were included. Hence, there may be no one single preferred model, clearly outperforming and superior to the rest. However, we aimed to select the final model consistently by including the input variables lowering the prediction error, while ensuring a parsimonious model. It should be noted that models including the temperature of the air outlet of the cooling system in the model, gave a prediction error in terms of RMSE that was lower by a factor of 0.5. The air outlet temperature was strongly correlated with the winding temperatures, and hence has high predictive power. Any normal or anomalous increase in winding temperature will, however, also lead to an increase in the air outlet temperature. The main aim of the derived model was to detect observed temperatures that are higher than predicted to raise an alarm. Including air outlet temperature as a covariate inherently introduces a risk of masking overheating cases, which was supported by the fact that the time to detect was not reduced by including the air outlet temperature even though the nominal prediction error (RMSE) of the model was significantly lower. The air outlet temperature was therefore omitted from the final model.

More complex machine learning approaches may also be utilised in the prediction step. This could include recurrent neural network and deep learning, such as the popular Long Short Term Memory (LSTM) models or Gated Recurrent Units (Hochreiter and Schmidhuber 1997; Cho et al. 2014). These methods, however, require excessive computational time and memory not available at the current on-board system implementation. Future work needs to assess whether such complex algorithms may improve the predictive performance in the initial modelling step of our framework.

Acknowledgements

This work was supported by Norwegian Research Council centre Big Insight project 237718. The authors would like to thank Bo-Won Lee and Jaroslaw Nowak at ABB Norway for valuable support in the development of this project.

Disclosure

The authors have nothing to disclose.

References

Basseville M, Nikiforov IV. 1993. Detection of abrupt changes: theory and application. Englewood Cliffs, Prentice Hall.

Brown RG. 1956. Exponential Smoothing for Predicting Demand. Cambridge, Massachusetts: Arthur D. Little Inc.

Chambers JM. 1992. Linear models. Chapter 4 of Statistical Models in S eds J. M. Chambers and T. J. Hastie: Wadsworth & Brooks/Cole.

Cho K, Van Merrinboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2014).

Claeskens G, Hjort NL. 2008. Model selection and model averaging. Cambridge University Press: Cambridge.

Gnacinski P. 2008. Prediction of windings temperature rise in induction motors supplied with distorted voltage. Energy Conversion and Management, 49(4):707-717.

Hansen PC, Pereyra V, Scherer G. 2013. Least squares data fitting with applications. JHU Press.

Friedman J, Hastie T, Tibshirani R. 2001. The elements of statistical learning: Data mining, Inference, and Prediction. (Vol. 1, No. 10). New York City (NY): Springer.

Hochreiter S, Schmidhuber J. 1997. Long short-term memory. Neural computation, 9(8):1735-1780.

Holt CC. 1957. Forecasting Trends and Seasonal by Exponentially Weighted Averages. Office of Naval Research Memorandum. 52.

Recommended Practice for Motor Protection in Industrial and Commercial Power Systems. IEEE Standard 3004.8, 2016. New York City (NY): IEEE.

Izenman AJ. 2008. Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning. New York City (NY): Springer.

Maftei C, Moreira L, Guedes Soares C. 2009. Simulation of the dynamics of a marine diesel engine. Journal of Marine Engineering & Technology. 8:2943.

Liu K, Zhang R, Mei Y. 2017. Scalable SUM-Shrinkage Schemes for Distributed Monitoring Large-Scale Data Streams. Statistica Sinica. 29:1–22.

Lorden G. 1971. Procedures for reacting to a change in distribution. The Annals of Mathematical Statistics. 42(6):1897–1908.

Lorden G, Pollak M. 2008. Sequential change-point detection procedures that are nearly optimal and computationally simple. Sequential Analysis. 27(4):476–512.

Losada MG, Rubio FR, Bencomo SD (Eds.). 2015. Asynchronous control for networked systems. Heidelberg: Springer.

Lystianingrum V, Hredzak B, Agelidis VG. 2016. Multiple-Model-Based Overheating Detection in a Supercapacitors String. IEEE Transactions on Energy Conversion, 31(4):1413-1422.

Mei Y. 2010. Efficient scalable schemes for monitoring a large number of data streams. Biometrika. 97(2):419–433.

Moustakides GV. 1986. Optimal stopping times for detecting changes in distributions. The Annals of Statistics. 14(4):1379–1387.

Page ES. 1954. Continuous inspection schemes. Biometrika. 41:100–115.

Page ES. 1955. A test for a change in a parameter occurring at an unknown point. Biometrika. 42:523–527.

Pawlus W, Birkeland JT, Van Khang H, Hansen MR. 2017. Identification and experimental validation of an induction motor thermal model for improved drivetrain design. IEEE Transactions on Industry Applications, 53(5):4288-4297.

Roberts SW. 1959. Control chart tests based on geometric moving averages. Technometrics. 1:239–250.

Vanem E., Brandsæter D. 2019. Unsupervised anomaly detection based on clustering methods and sensor data on a marine diesel engine. Journal of Marine Engineering & Technology.

Xie Y., Siegmund D. 2013. Sequential Multi-Sensor Change-Point Detection. The Annals of Statistics. 41(2):670-692.

Paper IV

Scalable changepoint and anomaly detection in cross-correlated data with an application to condition monitoring

Tveten, M., Eckley, I. A., and Fearnhead, P.

Invited to submit a revision to Annals of Applied Statistics. Openly available on arXiv: 2010.06937 [stat.ME].

Scalable changepoint and anomaly detection in cross-correlated data with an application to condition monitoring

Martin Tveten * Idris A. Eckley † Paul Fearnhead †

October 15, 2020

Abstract

Motivated by a condition monitoring application arising from subsea engineering we derive a novel, scalable approach to detecting anomalous mean structure in a subset of correlated multivariate time series. Given the need to analyse such series efficiently we explore a computationally efficient approximation of the maximum likelihood solution to the resulting modelling framework, and develop a new dynamic programming algorithm for solving the resulting Binary Quadratic Programme when the precision matrix of the time series at any given time-point is banded. Through a comprehensive simulation study, we show that the resulting methods perform favourably compared to competing methods both in the anomaly and change detection settings, even when the sparsity structure of the precision matrix estimate is misspecified. We also demonstrate its ability to correctly detect faulty time-periods of a pump within the motivating application.

Keywords: Anomaly; Binary Quadratic Programme; Changepoints; Cross-correlation; Outliers.

*Department of Mathematics, University of Oslo, Oslo, Norway.
†Mathematics and Statistics, Lancaster University, Lancaster, LA1 4YF, UK.

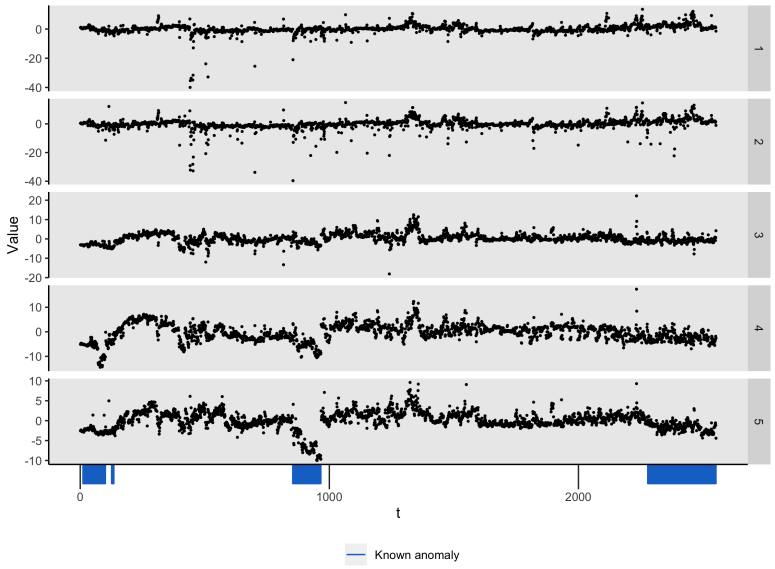

Figure 1: Pump data after preprocessing with four known segments of suboptimal operation marked by a blue rug. The correlation between variables 1 and 2 is 0.89 and the pairwise correlations between variables 3, 4 and 5 are all above 0.6.

1 Introduction

Modern machinery can be perplexingly complicated and interlinked. The interruption of one machine may cause downtime of a whole operation, in addition to a repair being both costly, time consuming and arduous. This has spawned an enormous interest in (remote) condition monitoring of industrial equipment to detect deviations from its normal operation, such that optimal uptime can be achieved and impending faults discovered before they occur. Overviews of condition monitoring techniques for different equipment exist for pump-turbines (Egusquiza et al., 2015), wind turbines (Tchakoua et al., 2014), and audio and vibration signals (Henriquez et al., 2014), among others. A common theme is the decision problem of when the machinery is running abnormally—a problem that lends itself well to statistical changepoint analysis.

The current work is motivated by a problem of detecting time-intervals (segments) of suboptimal operation of an industrial process pump. We will refer to these segments as "anomalies" or "anomalous segments", because they correspond to deviations from some predefined normal pump behaviour. The pump is equipped with sensors that measure temperatures and pressures over time at various locations. Other operational variables such as the flow rate and volume fractions for the different fluids being pumped are also recorded. If present, the aim is to estimate the start- and end-point of anomalies, as well as indicate which variables are anomalous. This is useful information to the operators of the pump to pin-point the source of historical problems and learn from it. Another reason for performing such an analysis is to create a clean reference data set to train a model of the equipment's baseline behaviour on, before deploying the method for online condition monitoring. The particular data set we consider contains four anomalies that have been manually labelled by the engineers based on retrospectively looking for signs in the data of degrading performance.

The starting-point of our methodology is to assume that during normal operation of the pump, the data follows a baseline stationary distribution, and during suboptimal operation, the

mean of the distribution changes abruptly for some period of time before it reverts back to the baseline mean. This is known as an *epidemic* changepoint model in the literature (Kirch et al., 2015). A challenge with the pump data is that the mean changes as a consequence of what is being pumped and other operating conditions in addition to suboptimal operation. To decrease the dependence on the operating conditions and thus increase the signal from changes due to suboptimal operation, we divide the variables into sets of *state* variables and *monitoring* variables, and regress the monitoring variables onto the state variables (similar to Klanderman et al. (2020)). The remaining five-variate time series of monitoring residuals are shown in Figure 1, where the known anomalies are marked on the time axis. Observe that the strength of the known anomalies vary as well as which variables seem to be affected. It is also apparent that the mean changes outside of the known anomalous segments. Detecting and estimating these segments is also important as they may correspond to previously unknown anomalies or constitute data for which the current model between state and monitoring variables fit poorly, and hence point to how it should be improved.

The pump data after preprocessing also exhibit strong cross-correlation due to the proximity of the sensors, with the correlation of variables 1 and 2 being 0.89 and the pairwise correlations between variables 3, 4 and 5 all being above 0.6. Most existing methods for detecting a change or anomaly in a subset of variables ignore cross-correlation (though see Wang and Samworth, 2018). If not accounted for, however, cross-correlation will hamper the detection of more subtle anomalies as illustrated by the simulated example in Figure 2. The point of doing multivariate changepoint detection is to borrow strength between variables to detect smaller changes than would be possible if each variable were considered separately, and including cross-correlation if sufficiently strong will increase the power of detection. This is particularly true for sparse changes, which has also been observed by Liu et al. (2019).

Our main methodological contribution is therefore to develop a novel test statistic based on a penalised cost approach for detecting multiple anomalies/epidemic changes in a subset of means of cross-correlated time series. The test is designed to be powerful for both sparse and dense alternatives, as well as to be computationally fast and scalable. This is crucial for our method to also be useful for anomaly detection problems of higher dimensionality than our process pump example. Anomalies are then detected by using the test within a PELT-type algorithm Killick et al. (2012) to optimise exactly over all possible start- and end-points of anomalies. We also show how the same ideas can be applied to the related classical multiple changepoint problem where there is no baseline behaviour.

Through the work on making the method scalable, we derive an algorithm which may be of independent interest within combinatorial optimisation. Our test statistic is an approximation to the maximum likelihood solution of our problem, formulated as what is known as an unconstrained Binary Quadratic Program (BQP). We show that such optimisation problems can be solved exactly by a dynamic programming (DP) algorithm scaling linearly in the number of variables, p, if the matrix in the quadratic part of the objective function is sparse in a banded fashion. In the anomaly detection problem, this corresponds to having a banded precision matrix. We present a simple pre-processing step for obtaining a banded estimate of the precision matrix of our data, and show empirically that detecting the changepoints using such an estimate leads to gains in power over methods that ignore cross-correlation even when the banded assumption is incorrect.

A further challenge in many applications, such as the pump data of Figure 1, is the presence of outliers. If left unattended, it is well-known that they will interfere with the detection of changes (Fearnhead and Rigaill, 2019). To handle outliers, we incorporate the distinction between point and collective anomalies, introduced in the CAPA (Collective And Point Anomalies) and MVCAPA (MultiVariate CAPA) methods of Fisch et al. (2019a,b). A point anomaly is defined as an anomalous segment of length one—a single anomalous observation—while a collective anomaly is an anomalous segment of length two or longer. This distinction enables

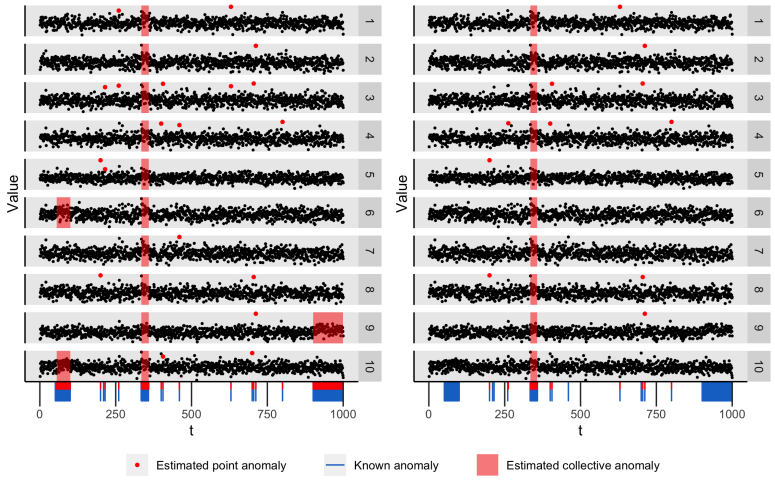

Figure 2: Modelling cross-correlation increases detection power for a fixed Type I error probability, especially for sparse changes. Both plots show the same set of 1000 simulated observations from a 10-variate Gaussian distribution with a global constant correlation of 0.5, containing three collective anomalies at $t \in (50, 100], (333, 358], (900, 1000]$, affecting the means of variables $\{6, 10\}$, $\{1, \ldots, 10\}$ and $\{9\}$, respectively, and 12 point anomalies affecting two random variables each. The left plot displays the estimates of collective and point anomalies of our method, which incorporates cross-correlations, while the right plot shows estimates when the method ignores cross-correlations. As both methods were tuned to achieve 0.05 probability of a false positive under the global correlation null model, the two sparse anomalies are not detected in the right plot as a trade-off with error control.

the method to classify sporadic outliers as point anomalies rather than confusing them with a collective anomaly. We call our anomaly and changepoint detection algorithms CAPA-CC and CPT-CC respectively, short for Collective And Point Anomalies in Cross-Correlated data and ChangePoinTs in Cross-Correlated data, respectively.

To the best of our knowledge, there are no other methods designed specifically for the multiple point and collective anomaly detection problem in multivariate, cross-correlated data with both sparse and dense anomalies. Current approaches to detect collective anomalies assume independence across series (Fisch et al., 2019b; Jeng et al., 2013). Alternatively, methods like Kirch et al. (2015) model correlated series, but focus on detecting changes in the cross-correlation.

For the general changepoint problem of a sparse or dense change in the mean, the literature is mostly concentrated on methods that either allow for sparse changes but assume cross-independence (Xie and Siegmund, 2013; Jirak, 2015; Cho and Fryzlewicz, 2015; Cho, 2016; Bardwell et al., 2019), or allow cross-dependence but assume changes are dense (Horváth and Hušková, 2012; Li et al., 2019; Bhattacharjee et al., 2019; Westerlund, 2019). The inspect method of Wang and Samworth (2018) is a notable exception from this rule as it is designed to estimate sparse changes in the mean of potentially cross-correlated data. Whilst general changepoint methods can also be used for the anomaly detection problem, some power is expected to be lost as there is no assumption of a shared baseline parameter.

The rest of the paper is organised as follows: We first describe the anomaly and changepoint detection problems in detail in Section 2, before considering the anomaly detection problem in detail in Section 3. Particular focus is put on the single collective anomaly case and our BQP solving algorithm for approximating the maximum likelihood solution. We then briefly describe

how the same ideas can be applied to the general changepoint detection problem in Section 4, In Section 5, we cover a useful strategy for robustly estimating the precision matrix with a given sparsity structure, and Section 6 contains an extensive simulation study for assessing the performance of our methods. We conclude by presenting the analysis of the pump data in Section 7.

2 Problem description

Suppose we have n observations $\{\mathbf{x}_t\}_{t=1}^n$ of p variables $\mathbf{x}_t = (x_t^{(1)}, \ldots x_t^{(p)})$, where each $\mathbf{x}_t$ has mean $\boldsymbol{\mu}_t$ and a common precision matrix $\mathbf{Q}$ encoding the conditional dependence structure between the variables. Our interest is in either detecting collective anomalies or changepoints that are characterised by a change in the mean of the data. We will first set up the anomaly detection problem, before describing the changepoint problem in terms of it.

In our anomaly detection problem, segments of the data will be considered anomalous if the mean $\boldsymbol{\mu}_t$ is different from a baseline mean $\boldsymbol{\mu}_0$. Let K be the number of anomalous segments, where the kth anomaly, for $k = 1, \ldots, K$, starts at observation $s_k + 1$, ends at observations e_k, and affects the components in a subset $\mathbf{J}_k \subseteq [p]$. So, the model assumes that the mean vectors $\boldsymbol{\mu}_t$ are given by

$$\mu_t^{(i)} = \begin{cases} \mu_1^{(i)} & \text{if } s_1 < t \leq e_1 \text{ and } i \in \mathbf{J}_1, \\ \quad \vdots & \\ \mu_K^{(i)} & \text{if } s_K < t \leq e_K \text{ and } i \in \mathbf{J}_K, \\ \mu_0^{(i)} & \text{otherwise,} \end{cases} \tag{1}$$

where $e_k \leq s_{k+1}$, such that no overlapping anomalous segments are allowed. In some cases, one may also be given information about the minimum and maximum segment length of an anomaly, $l \geq 1$ and $M > l$, respectively, such that $l \leq e_k - s_k \leq M$ for all k. Our aim is to infer the number of anomalies K, as well as their locations within the data $(s_k, e_k, \mathbf{J}_k)_{k=1}^K$ together with the anomalous means $\boldsymbol{\mu}_k^{(i)}$, for $i \in \mathbf{J}_k$, in a computationally efficient manner.

In the corresponding changepoint problem, on the other hand, there is no concept of a baseline mean. It is thus the special case of (1) where the end of a segment is also the start of a new one, i.e., $e_k = s_{k+1}$ and $e_K = n$. To distinguish the two problems, we will denote the changepoints as $\tau_k := s_k$ in the changepoint problem. The aim can therefore be stated as estimating the number of changepoints K, their locations $(\tau_k, \mathbf{J}_k)_{k=1}^K$, and the segment means $\boldsymbol{\mu}_k$, for $k = 0, \ldots, K$.

As is common, in the anomaly detection problem, we assume that the baseline parameter $\boldsymbol{\mu}_0$ is known and for both problems we assume that the precision matrix $\mathbf{Q}$ is known. In practice, these will be estimated from the data using robust statistical methods described in Section 5. Later, to enable quick computation, we will also assume that $\mathbf{Q}$ or an estimate of $\mathbf{Q}$ is sparse in a banded fashion. A sparse precision matrix corresponds to cases where only a few of the variables are conditionally dependent.

3 Detecting anomalies

3.1 A single collective anomaly

In this section, we consider the anomaly detection problem described in Section 2 for $K \leq 1$. Our approach is to model the data as being realisations of multivariate Gaussian random variables, independent over time, and to use a penalised likelihood approach to detect an anomaly.

We will use the following notation: For a p-vector $\mathbf{x}$ and set $\mathbf{J} \subseteq [p]$, $\mathbf{x}^{(\mathbf{J})} := (x^{(i)})_{i \in \mathbf{J}}$ and $\mathbf{x}(\mathbf{J}) := \left(x^{(i)} I\{i \in \mathbf{J}\}\right)_{i=1}^p$, where $I\{i \in \mathbf{J}\}$ is the indicator function. For a matrix $\mathbf{X}$, $\mathbf{X}_{\mathbf{J},\mathbf{K}}$

denotes the sub-matrix of rows $\mathbf{J}$ and columns $\mathbf{K}$. Both $-\mathbf{J}$ and $\mathbf{J}^c$ refer to the complement of a set $\mathbf{J}$. The k-subscripts enumerating the anomalies/changepoints will be skipped when the referenced anomaly or changepoint is clear from the context.

Define the cost of introducing an anomaly from time-point $s+1$ to e in variables $\mathbf{J}$ as twice the negative log-likelihood of multivariate Gaussian data,

$$\begin{aligned} C\left(\mathbf{x}_{(s+1):e}, \boldsymbol{\mu}(\mathbf{J})\right) &= -2 \sum_{t=s+1}^{e} \log f\left(\mathbf{x}_t | \boldsymbol{\mu}(\mathbf{J})\right) \\ &\propto \sum_{t=s+1}^{e} \left(\mathbf{x}_t - \boldsymbol{\mu}(\mathbf{J})\right)^{\intercal} \mathbf{Q} \left(\mathbf{x}_t - \boldsymbol{\mu}(\mathbf{J})\right). \end{aligned} \tag{2}$$

Now, for ease of presentation, without loss of generality we assume $\boldsymbol{\mu}_0 = \mathbf{0}$. Then the log-likelihood ratio statistic of the observations $\mathbf{x}^{(\mathbf{J})}_{(s+1):e}$ being anomalous is given by

$$S(s, e, \mathbf{J}) = C(\mathbf{x}_{(s+1):e}, \mathbf{0}) - \min_{\boldsymbol{\mu}(\mathbf{J})} C\left(\mathbf{x}_{(s+1):e}, \boldsymbol{\mu}(\mathbf{J})\right). \tag{3}$$

We refer to $S(s, e, \mathbf{J})$ as the *saving* of allowing the observations $\mathbf{x}^{(\mathbf{J})}_{(s+1):e}$ to have a different mean than $\mathbf{0}$. In a maximum likelihood spirit, the aim is to maximise the savings $S(s, e, \mathbf{J})$ over start-points s, end-points e, and subset $\mathbf{J}$, and infer the anomalous segment thereof. However, as we vary $\mathbf{J}$ we are optimising over differing numbers of means in the anomalous segment – and the savings will always increase as we optimise over more parameters. One way of dealing with this is to introduce a penalty that is the function of the number of anomalous variables, $P(|\mathbf{J}|)$, and maximise the penalised savings in stead. This gives us the following anomaly detection statistic:

$$S := \max_{l \leq s-e \leq M} S(s, e) := \max_{l \leq s-e \leq M} \max_{\mathbf{J}} \left[S(s, e, \mathbf{J}) - P(|\mathbf{J}|)\right]. \tag{4}$$

Recall that l and M is the minimum and maximum segment length, respectively. An anomaly is declared if (4) is positive, and the maximising $(s, e, \mathbf{J})$ is a point-estimate of the anomaly's position in the data.

Throughout this article, we use a piecewise linear penalty function of the form

$$P(|\mathbf{J}|) = \min(\alpha_{\text{sparse}} + \beta|\mathbf{J}|, \alpha_{\text{dense}}) = \begin{cases} \alpha_{\text{sparse}} + \beta|\mathbf{J}|, & |\mathbf{J}| < k^* \\ \alpha_{\text{dense}}, & |\mathbf{J}| \geq k^* \end{cases}, \tag{5}$$

where $k^* = (\alpha_{\text{dense}} - \alpha_{\text{sparse}})/\beta$. We will refer to $|\mathbf{J}| < k^*$ as being in the *sparse regime* and $|\mathbf{J}| \geq k^*$ as being in the *dense regime*. Such a penalty function ensures that our method can be powerful against both sparse and dense alternatives. In addition, we can apply the results from Fisch et al. (2019b) where it is shown that, if our modelling assumptions are correct, setting $\alpha_{\text{dense}} = p + 2\sqrt{p\psi} + 2\psi$, $\alpha_{\text{sparse}} = 2\psi$ and $\beta = 2\log(p)$, for $\psi = \log(n)$, results in a false positive rate that tends to 0 as n grows. Furthermore Fisch et al. (2019b) show that scaled versions of these rates are appropriate in many situations where the modelling assumptions do not hold, such as when there is dependence over time.

Furthermore, note that $[p]$ is always the maximiser in the dense regime, and that β is the additional penalty for adding an extra variable to the anomalous subset in the sparse regime. We will exploit these properties when deriving an efficient optimisation algorithm in Section 3.2.

To compute the anomaly detection statistic S, we need the maximum likelihood estimator (MLE) $\hat{\boldsymbol{\mu}}(\mathbf{J})$ of $\boldsymbol{\mu}(\mathbf{J})$, where the means of variables $j \in \mathbf{J}$ are allowed to vary freely while the others are restricted to 0. Optimising the multivariate Gaussian likelihood (2) with respect to such a subset restricted mean results in the following MLE for the mean components in $\mathbf{J}$:

$$\hat{\boldsymbol{\mu}}^{(\mathbf{J})}_{(s+1):e} = \bar{\mathbf{x}}^{(\mathbf{J})}_{(s+1):e} + \mathbf{Q}^{-1}_{\mathbf{J},\mathbf{J}} \mathbf{Q}_{\mathbf{J},-\mathbf{J}} \bar{\mathbf{x}}^{(-\mathbf{J})}_{(s+1):e}. \tag{6}$$

The corresponding p-vector $\hat{\boldsymbol{\mu}}(\mathbf{J})$ is constructed by placing $\hat{\boldsymbol{\mu}}^{(\mathbf{J})}$ at indices $\mathbf{J}$ and 0's elsewhere. Finally, putting the MLE back into the expression for the saving, and suppressing the subscripts $(s+1):e$ to not clutter the display, gives us that

$$S(s, e, \mathbf{J}) = (e-s)(2\bar{\mathbf{x}} - \hat{\boldsymbol{\mu}}(\mathbf{J}))^{\mathsf{T}}\mathbf{Q}\hat{\boldsymbol{\mu}}(\mathbf{J}). \tag{7}$$

Unfortunately, the complicated form of the MLE (6) means that the number of operations required for finding the exact maximum penalised saving over subsets $\mathbf{J}$ scales on the order of $O(2^p)$. The optimisation problem is not only combinatorial, but also nonlinear, and as far as we know, there is no reformulation of the saving 7 that would make the problem notably more tractable. We thus opt for an approximation to the saving 7 to achieve scalability.

3.2 Approximate savings for anomaly detection

Our idea for a computationally efficient approximation of the subset-maximised penalised savings $S(s, e)$, is to replace the MLE in (7) with the subset-truncated sample mean,

$$\bar{\mathbf{x}}(\mathbf{J}) = \bar{\mathbf{x}} \circ \mathbf{u}, \tag{8}$$

where $\mathbf{u} = (I\{i \in \mathbf{J}\})_{i=1}^{p}$ and $\circ$ is the element-wise (Hadamard) product. That is, under the sparse regime, we aim to maximise the approximate penalised saving;

$$\tilde{S}(s, e) := \max_{\mathbf{J}} \left[\tilde{S}(s, e, \mathbf{J}) - P(|\mathbf{J}|)\right] = \max_{\mathbf{J}} \left[(e-s)(2\bar{\mathbf{x}} - \bar{\mathbf{x}}(\mathbf{J}))^{\mathsf{T}}\mathbf{Q}\bar{\mathbf{x}}(\mathbf{J}) - \beta|\mathbf{J}|\right] - \alpha_{\text{sparse}}. \tag{9}$$

Under the dense regime, the exact maximum is given by $S(s, e, [p]) - \alpha_{\text{dense}}$.

An important motivation for using $\bar{\mathbf{x}}(\mathbf{J})$ is that finding $\tilde{S}(s, e)$ corresponds to what is known as a *binary quadratic program* (BQP). The unconstrained version of such optimisation problems are of the form

$$\max_{\mathbf{u} \in \{0,1\}^p} \mathbf{u}^{\mathsf{T}}\mathbf{A}\mathbf{u} + \mathbf{u}^{\mathsf{T}}\mathbf{b} + c, \tag{10}$$

where $\mathbf{A}$ is a real, symmetric, $(p \times p)$-dimensional matrix, $\mathbf{b}$ is a real, p-dimensional vector and c is a real scalar. BQPs are NP-hard in general (Garey and Johnson, 1979), even if $\mathbf{A}$ is positive definite. If $\mathbf{A}$ is r-banded, however, we show that BQPs can be solved with $O(p2^r)$ operations. Proposition 1 confirms that $\max_{\mathbf{J}}[\tilde{S}(s, e, \mathbf{J}) - P(|\mathbf{J}|)]$ is indeed a BQP.

Proposition 1. *Let $\alpha, \beta \geq 0$, $\bar{\mathbf{x}} \in \mathbb{R}^p$ and $\bar{\mathbf{x}}(\mathbf{J}) = \mathbf{u} \circ \bar{\mathbf{x}}$, where $\mathbf{u}$ is a binary vector with 1 at positions $\mathbf{J}$ and 0 elsewhere. Then solving*

$$\max_{\mathbf{J}} \left[(e-s)(2\bar{\mathbf{x}} - \bar{\mathbf{x}}(\mathbf{J}))^{\mathsf{T}}\mathbf{Q}\bar{\mathbf{x}}(\mathbf{J}) - \beta|\mathbf{J}|\right] - \alpha \tag{11}$$

corresponds to a BQP with $\mathbf{A} = -(e-s)\bar{\mathbf{x}}\bar{\mathbf{x}}^{\mathsf{T}} \circ \mathbf{Q}$, $\mathbf{b} = 2(e-s)(\bar{\mathbf{x}} \circ \mathbf{Q}\bar{\mathbf{x}}) - \beta$ and $c = -\alpha$.

To explain the dynamic program (Algorithm 1) for solving the BQP when the precision matrix $\mathbf{Q}$, and hence $\mathbf{A}$, is r-banded, it is illustrative to consider the case of $r = 1$. The key idea is that if we cycle through the variables in turn, then the choice of which of the variables $d, \ldots, p$ are anomalous will depend on the variables $1, \ldots, d-1$ only through whether variable $d-1$ is anomalous or not. Thus we can obtain a recursion by considering these two possibilities separately.

In the case of $r = 1$, the BQP for $\max_{\mathbf{J}}[\tilde{S}(s, e, \mathbf{J}) - P(|\mathbf{J}|)]$ is given by

$$\max_{\mathbf{u} \in \{0,1\}^p} \sum_{d=1}^{p}(b_d + A_{d,d})u_d + 2\sum_{d=2}^{p} A_{d,d-1}u_d u_{d-1} + c, \tag{12}$$

where $A_{d,i} = (e-s)Q_{d,i}\bar{x}_d\bar{x}_i$ for $i = d, d-1$, $b_d = 2(e-s)\bar{x}_d \sum_{i=d-1}^{d+1} Q_{d,i}\bar{x}_i - \beta$, and $c = -\alpha$. Let $\tilde{S}_1(d)$ and $\tilde{S}_0(d)$ be the maximal approximate penalised savings of variables $1, \ldots, d \leq p$

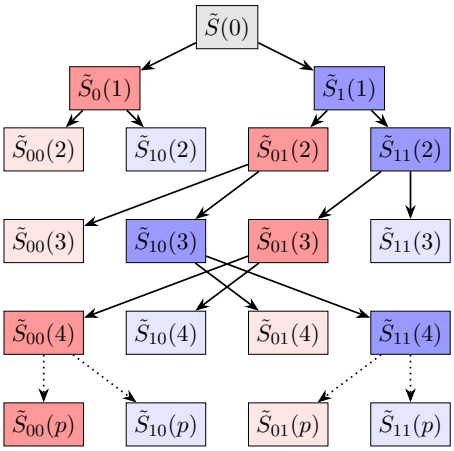

Figure 3: The unbalanced binary tree structure of the dynamic program for solving (11) for 1-banded $\mathbf{Q}$ and fictitious data. The blue and red nodes refer to conditioning on whether the current variable/level d is anomalous ($u_d = 1$) or not anomalous ($u_d = 0$), respectively. At each level, the darker coloured nodes are the selected parent within every colour group, while the edges correspond to the step of growing children nodes. Observe that the maximum value to the BQP in this example is $\tilde{S}_{00}(p)$, with argument $\mathbf{u} = (1, 1, 0, 0, \ldots, 0)$.

conditional on variable d being anomalous ($u_d = 1$) or not ($u_d = 0$) for a fixed s and e. Moreover, we write $\tilde{S}_{(0,u)}(d)$ and $\tilde{S}_{(1,u)}(d)$ for $u = 0, 1$ when additionally conditioning on variable $d-1$ being 0 or 1. Then, by initialising from $\tilde{S}(0) := c$, $\tilde{S}_0(1) = \tilde{S}(0)$ and $\tilde{S}_1(1) = \tilde{S}(0) + b_1 + A_{1,1}$, the following two-stage recursion holds for $d = 2, \ldots, p$:

$$\begin{aligned} \tilde{S}_{(0,u)}(d) &= \tilde{S}_u(d-1), \\ \tilde{S}_{(1,u)}(d) &= \tilde{S}_u(d-1) + b_d + A_{d,d} + 2uA_{d,d-1}, \end{aligned} \tag{13}$$

for $u = 0, 1$, and

$$\tilde{S}_u(d) = \max\left(\tilde{S}_{(u,0)}(d), \tilde{S}_{(u,1)}(d)\right), \tag{14}$$

such that $\max(\tilde{S}_0(p), \tilde{S}_1(p)) = \max_{\mathbf{J}}[\tilde{S}(s, e, \mathbf{J}) - P(|\mathbf{J}|)]$ when $r = 1$. Note that the computational complexity of finding the optimum in this case is only $O(p)$.

To extend the recursion to more general precision matrices, observe that the dynamic program given by (13) and (14) can be described by an unbalanced binary tree (Figure 3). Initialisation occurs at levels 0 and 1 of the tree. Thereafter, two selected nodes at level $d-1$ grow children nodes according to (13), before two of the four nodes at level d is selected as parents for the next level by the max operation in (14). The path from the maximum node at the final level back to the root encodes the optimal $\mathbf{u}$. In the following, we will refer to the vector of 0's and 1's along the path from a certain node back up to the root as the "position" or "argument" of a node.

By using the tree description, it is easier to generalise the algorithm to any neighbourhood structure of each variable d. When $r = 1$, we only have to consider the two options of variable $d-1$ being 0 or 1 at every step d, whereas for a general band, we have to consider all combinations of variables $d-r, \ldots, d-1$ being 0 or 1. A further adaptation to the precision matrix at hand can be made by excluding those variables among $d-r, \ldots, d-1$ that will never be visited again, at each step d. To be precise, let us define the neighbours of variable d by $N_d := \{i : A_{d,i} \neq 0\}$, and the potential lower neighbours of d by $P_d^{<} := \{\max(1, d-r), \ldots, d-1\}$ for $d \geq 2$ and

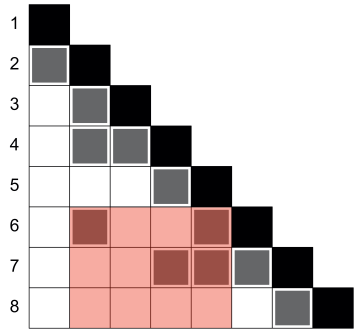

Figure 4: An example 4-banded $\mathbf{A}$ matrix where the diagonal is black, other non-zero elements are gray, and zero-elements are white. The red region illustrates how the extended neighbours of $d = 6$ are found; the column indices of the red region correspond to $P_6^< = \{2, 3, 4, 5\}$, but variable 3 can be excluded as it is not in any of the coming neighbourhoods, making $M_6 = \{2, 4, 5\}$. The other extended neighbourhoods in this example are $M_1 = \emptyset$, $M_2 = \{1\}$, $M_3 = \{2\}$, $M_4 = \{2, 3\}$, $M_5 = \{2, 4\}$, $M_7 = \{4, 5, 6\}$ and $M_8 = \{7\}$.

$P_1^< := \emptyset$. At each step d, we have to condition on all 0-1-combinations of the variables in

$$M_d := P_d^< \setminus \big(\cup_{i=d}^{d+r} N_i \big)^{\mathsf{c}} = P_d^< \cap \big(\cup_{i=d}^{d+r} N_i \big). \tag{15}$$

We call the variables in M_d the *extended neighbours* of d. See Figure 4 for an example of how the M_d's are constructed.

To accomodate for more complicated neighbourhood structures, we have to extend the scalar indicators u needed when $r = 1$, to vector indicators $\mathbf{u}_d \in \{0, 1\}^{|M_d|}$ that give us the position of a node in the tree relative to M_d. I.e., $\mathbf{u}_d$ tells us which extended neighbours of d are on (1) or off (0). At each level d, all $2^{|M_d|}$ possible on-off-combinations must be conditioned on, resulting in $2^{|M_d|+1}$ recursive updates, given by

$$\begin{aligned} \tilde{S}_{(0,\mathbf{u}_d)}(d) &= \tilde{S}_{\mathbf{u}_d}(d-1) \\ \tilde{S}_{(1,\mathbf{u}_d)}(d) &= \tilde{S}_{\mathbf{u}_d}(d-1) + b_d + A_{d,d} + 2\mathbf{u}_d^{\mathsf{T}} \mathbf{A}_{d,M_d}, \end{aligned} \tag{16}$$

where $(0, \mathbf{u}_d)$ and $(1, \mathbf{u}_d)$ indicates the positions of the 0-child and 1-child nodes relative to M_d. All these children nodes constitute the nodes at level d, and we will refer to them by $\{\tilde{S}(d)\}$.

The parent-selecting step in the general case also becomes more complex since the extended neighbourhoods can evolve in many different ways. To explain this step in detail, we use the notation position$(\tilde{S}(d))$ to refer to the 0-1-vector that gives the position of a given node in our binary tree representation of the algorithm. For example, position$(\tilde{S}_{10}(4)) = (1, 1, 0, 1)$ in Figure 3. Now the parent for each $\mathbf{u}_d$ are determined by maximising over the variables that will never be visited again;

$$\tilde{S}_{\mathbf{u}_d}(d-1) = \max_{\mathbf{v} \in \mathbf{V}} \tilde{S}_{\mathbf{v}}(d-1), \tag{17}$$

where $\mathbf{V} = \{\mathbf{v} \in \text{positions}(\{\tilde{S}(d-1)\}) : \mathbf{v}^{(M_d)} = \mathbf{u}_d\}$ is the set of positions at level $d-1$ that match the on-off pattern indicated by $\mathbf{u}_d$ relative to M_d.

The final algorithm is summarised in Algorithm 1 and 2. Note that we also keep track of the minimum number of anomalous variables at each level d through the term $\underline{k}$. In this way, the recursions can be stopped as soon as the anomaly is guaranteed to lie in the dense regime. For an r-banded matrix the computational complexity is bounded by $O\big(\sum_{d=1}^{p} 2^{|M_d|}\big) \leq O(p2^r)$, and if the anomaly is estimated as dense, the number of operations may be substantially less.

Algorithm 1 Dynamic programming BQP solver for banded matrices

Input: $\mathbf{A}$, $\mathbf{b}$, c, $\{M_d\}_{d=1}^p$, k^*

$d = 1$, $\underline{k} = 0$, $\tilde{S}(0) = c$.
while $d \leq p$ and $\underline{k} \leq k^*$ **do**
  for $\mathbf{u}_d \in \{0,1\}^{|M_d|}$ **do**
    $\mathbf{V} = \{\mathbf{v} \in \text{positions}(\{\tilde{S}(d-1)\}) : \mathbf{v}^{(M_d)} = \mathbf{u}_d\}$.
    $\tilde{S}_{\mathbf{u}_d}(d-1) = \max_{\mathbf{v} \in \mathbf{V}} \tilde{S}_{\mathbf{v}}(d-1)$.
    $\tilde{S}_{(0,\mathbf{u}_d)}(d) = \tilde{S}_{\mathbf{u}_d}(d-1)$.
    $\tilde{S}_{(1,\mathbf{u}_d)}(d) = \tilde{S}_{\mathbf{u}_d}(d-1) + b_d + A_{d,d} + 2\mathbf{u}_d^\intercal \mathbf{A}_{d,M_d}$.
  end for
  $\underline{k} = \min_{\mathbf{v} \in \text{positions}\{\tilde{S}(d)\}} \mathbf{v}^\intercal \mathbf{1}$.
  $d = d + 1$.
end while
$\tilde{\mathbf{J}} = \text{argmax}\{\tilde{S}(p)\}$.
$\tilde{S} = \max\{\tilde{S}(p)\}$.
return: $\tilde{S}$, $\tilde{\mathbf{J}}$.

Algorithm 2 The approximate penalised saving for anomaly detection used in CAPA-CC

Input: $\bar{\mathbf{x}}$, $\mathbf{Q}$, $\{M_d\}_{d=1}^p$, β, α_{sparse}, α_{dense}, k^*, e, s.

$\mathbf{A} = -(e-s)\bar{\mathbf{x}}\bar{\mathbf{x}}^\intercal \circ \mathbf{Q}$.
$\mathbf{b} = 2(e-s)(\bar{\mathbf{x}} \circ \mathbf{Q}\bar{\mathbf{x}}) - \beta$.
$c = -\alpha_{\text{sparse}}$
$\tilde{S}$, $\tilde{\mathbf{J}}$ from Algorithm 1 with input ($\mathbf{A}$, $\mathbf{b}$, c, $\{M_d\}_{d=1}^p$, k^*)
$S = S(s, e, [p]) - \alpha_{\text{dense}}$.
if $\tilde{S} \geq S$ **return:** $\tilde{S}, \tilde{\mathbf{J}}$.
else return: S, $[p]$.

3.3 Properties of the approximation

Our main evaluation of the approximation's performance is done through simulations, where in Section B.1 in the Supplementary Material we demonstrate that the approximation and the MLE give almost equal results for low p. Some properties regarding how $\tilde{S}(s, e)$ compares to $S(s, e)$, however, can be derived theoretically.

Firstly, under the dense penalty regime, the approximate MLE is equal to the MLE because the optimal $\mathbf{J}$ is $[p]$ in both cases, making $\hat{\boldsymbol{\mu}}(\mathbf{J}) = \bar{\mathbf{x}}$. Thus, we are only approximating the savings under the sparse penalty regime.

Secondly, $\tilde{S}(s, e) \leq S(s, e)$ for all start- and end-points s and e. This follows by definition of the MLE, which is present in $S(s, e)$; $\hat{\boldsymbol{\mu}}(\mathbf{J})$ is the minimiser in (3), and consequently, no other estimator can make the saving larger. Using the approximation will therefore not increase the probability of falsely detecting anomalies. The only effect it may have is a reduction in power.

In addition to the lower bound of 0 on the approximation error, Proposition 2 gives an upper bound which is useful for distilling what drives a potential decrease in performance. The proof is given in Section A.2 in the Supplementary Material.

Proposition 2. *Let* $\mathbf{W}(\mathbf{J})$ *be the matrix where* $\mathbf{W}(\mathbf{J})_{\mathbf{J},-\mathbf{J}} = \mathbf{Q}_{\mathbf{J},\mathbf{J}}^{-1}\mathbf{Q}_{\mathbf{J},-\mathbf{J}}$ *and is* 0 *elsewhere, and* $\hat{\mathbf{J}} = \text{argmax}_{\mathbf{J}}\big[S(s, e, \mathbf{J}) - P(|\mathbf{J}|)\big]$. *Then the following bound on the approximation error holds for all* $s < e$*:*

$$0 \leq S(s, e) - \tilde{S}(s, e) \leq (e-s)\lambda_{\max}\big(\mathbf{Q}\mathbf{W}(\hat{\mathbf{J}})\big)\|\bar{\mathbf{x}}_{(s+1):e}(\hat{\mathbf{J}}^{\mathbf{c}})\|^2. \tag{18}$$

The right-hand side of (18) indicates that the worst-case scenarios for our approximation are sparse changes in strongly correlated data. The right-hand side of (18) suggests that the relative

approximation error will be largest for sparse changes in strongly correlated data—as this is the situation that $\|\bar{\mathbf{x}}_{(s+1):e}(\hat{\mathbf{J}}^{\mathbf{c}})\|^2$ is largest (see Section A.2 in the Supplementary Material). The simulation results in Section B.1 in the Supplementary Material support this conclusion that the greatest difference in performance occurs when there is a sparse change in strongly correlated data, although the difference is small in the tested settings.

3.4 Multiple point and collective anomalies

We can extend the described method for detecting a single collective anomaly to detecting multiple collective anomalies, and also to allow for point anomalies within the normal segments. To incorporate point anomalies, we follow the approach of Fisch et al. (2019a,b) by defining point anomalies as collective anomalies of length 1. Thus, the optimal approximate saving of a point anomaly at time t can be defined as

$$\tilde{S}'(t) = \max_{\mathbf{J}} \left[\tilde{S}(t, t, \mathbf{J}) - \beta'|\mathbf{J}|\right]. \tag{19}$$

In accordance with Fisch et al. (2019b), we set $\beta' = 2\log p + 2\psi$, where $\psi = \log n$ as in Section 3.1. As for the collective anomaly penalty function, β' can be scaled by a constant factor to achieve appropriate error control.

We can now extend our penalised likelihood framework. The estimates for the collective anomalies, $\tilde{K}$ and $(\tilde{s}_k, \tilde{e}_k, \tilde{\mathbf{J}}_k)$ for $k = 1, \ldots, \tilde{K}$, and point anomalies, $\tilde{O}$ and $\tilde{\mathbf{J}}_t$ for $t \in \tilde{O}$, can then be obtained by minimising the penalised cost

$$\max_{K \in [\lfloor n/l \rfloor], s_k, e_k} \sum_{k=1}^{K} \tilde{S}(s_k, e_k) + \max_{O \subseteq [n]} \sum_{t \in O} \tilde{S}'(t), \tag{20}$$

subject to $\hat{e}_k - \hat{s}_k \geq l \geq 2$, $\hat{e}_k \leq \hat{s}_{k+1}$ and $(\cup_k [\hat{s}_k + 1, \hat{e}_k]) \cap O = \emptyset$.

The optimisation problem (20) can be solved exactly by a pruned dynamic program, using ideas from the PELT algorithm of Killick et al. (2012). Defining $C(m)$ as the maximal penalised approximate savings for observations $\mathbf{x}_{1:m}$, the basis for our PELT algorithm is the following recursive relationship:

$$C(m) = \max\left(C(m-1), \max_{0 \leq t \leq m-l}\left[C(t) + \tilde{S}(t, m)\right], C(m-1) + \tilde{S}'(t)\right), \tag{21}$$

for $C(0) = 0$. The first term in the outer maximum corresponds to no anomaly at m, the second term to a collective anomaly ending at m, and the third term to a point anomaly at m.

The computationally costly part of (21) is the maximisation over all possible starting-points t in the term for collective anomalies. Due to this term, the runtime of this dynamic program scales quadratically in n. If one specifies a maximum segment length M, however, the runtime is reduced to $O(Mn)$ at the risk of missing collective anomalies that are longer than M. The PELT algorithm is able to prune those t's in the term for the collective anomalies that can never be the maximisers, thus reducing computational cost whilst maintaining exactness. Proposition 3 gives a condition for when t can be pruned. The proof is given in the Supplementary Material.

Proposition 3. *If there exists an $m \geq t - l$ such that*

$$C(t) + \tilde{S}(t, m) + \alpha_{dense} \leq C(m) \tag{22}$$

then, for all $m' \geq m + l$, $C(m') \geq C(t) + \tilde{S}(t, m')$.

Proposition 3 states that if (22) is true for some $m \geq t - l$, t can never be the optimal changepoint for future times $m' \geq m + l$, and can therefore be skipped in the dynamic program. Killick et al. (2012) shows that if the number of changepoints increases linearly in n, then such

a pruned dynamic program can scale linearly. In the worst case of no changepoints, however, the scaling is still quadratic in n.

Calculating $C(n)$ in (21) by PELT with savings computed from Algorithm 2 constitutes our CAPA-CC algorithm.

4 Changepoint detection

In this section, we derive a test statistic for the single changepoint detection problem that utilises the approximation used for anomaly detection. CPT-CC detects multiple changepoints by embedding the test for a single changepoint within binary segmentation or a related algorithm, such as wild binary segmentation Fryzlewicz (2014) or seeded binary segmentation Kovács et al. (2020).

Recall that the single changepoint problem is like the anomaly detection problem, with the exception that $e = n$ and all means are unknown. In addition, we use τ to denote the changepoint. Without loss of generality assume the sample mean for each series is 0. To be able to use the same approximation as in the anomaly detection case we will base our cost on the log-likelihood under the assumption that the mean of the data is 0 for each series if there is no change. The resulting changepoint saving for a fixed τ and $\mathbf{J}$ is given by

$$\begin{aligned} S(\tau, \mathbf{J}) :&= C\left(\mathbf{x}_{1:n}, \mathbf{0}\right) - \min_{\boldsymbol{\mu}(\mathbf{J})} C\left(\mathbf{x}_{1:\tau}, \boldsymbol{\mu}(\mathbf{J})\right) - \min_{\boldsymbol{\mu}(\mathbf{J})} C\left(\mathbf{x}_{(\tau+1):n}, \boldsymbol{\mu}(\mathbf{J})\right) \\ &= S(1, \tau, \mathbf{J}) + S(\tau+1, n, \mathbf{J}), \end{aligned} \tag{23}$$

where C is defined in (2) and $S(s, e, \mathbf{J})$ in (3). Note that $\mathbf{J}$ is the same both before and after a changepoint to restrict the change vector $\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0$ to be nonzero only in $\mathbf{J}$. Mirroring the anomaly detection case, we obtain our changepoint test statistic by subtracting the penalty function and maximising over τ and $\mathbf{J}$;

$$\max_{l \le \tau \le n-l} S(\tau) := \max_{l \le \tau \le n-l} \left[\max_{\mathbf{J}} S(\tau, \mathbf{J}) - P(|\mathbf{J}|) \right], \tag{24}$$

where $l \ge 1$ is the minimum segment length as before.

Next, we once again replace the MLE of $\boldsymbol{\mu}(\mathbf{J})$ with the subset-truncated sample mean, $\bar{\mathbf{x}}(\mathbf{J})$, defined in (8). The optimal approximate penalised savings for the single changepoint problem is thus given by

$$\tilde{S}(\tau) = \max_{\mathbf{J}} \left[\tau(2\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_1(\mathbf{J}))^\intercal \mathbf{Q} \bar{\mathbf{x}}_1(\mathbf{J}) + (n-\tau)(2\bar{\mathbf{x}}_2 - \bar{\mathbf{x}}_2(\mathbf{J}))^\intercal \mathbf{Q} \bar{\mathbf{x}}_2(\mathbf{J}) - \beta|\mathbf{J}| - \alpha \right], \tag{25}$$

where $\bar{\mathbf{x}}_1 := \bar{\mathbf{x}}_{1:\tau}$ and $\bar{\mathbf{x}}_2 := \bar{\mathbf{x}}_{(\tau+1):n}$. A changepoint is detected when $\max_\tau \tilde{S}(\tau) > 0$. Proposition 4 confirms that (25) is also a BQP, such that Algorithm 1 can be used to find the optimum efficiently for a banded precision matrix $\mathbf{Q}$. Algorithm 3 summarises the method.

Proposition 4. *Let $\alpha, \beta \ge 0$, $\bar{\mathbf{x}} \in \mathbb{R}^p$ and $\bar{\mathbf{x}}(\mathbf{J}) = \mathbf{u} \circ \bar{\mathbf{x}}$, where $\mathbf{u}$ is 1 at $\mathbf{J}$ and 0 elsewhere. Then solving (25) corresponds to a BQP with $c = -\alpha$ and*

$$\begin{aligned} \mathbf{A} &= -\tau(\bar{\mathbf{x}}_1 \bar{\mathbf{x}}_1^\intercal \circ \mathbf{Q}) - (n-\tau)(\bar{\mathbf{x}}_2 \bar{\mathbf{x}}_2^\intercal \circ \mathbf{Q}) \\ \mathbf{b} &= 2\tau(\bar{\mathbf{x}}_1 \circ \mathbf{Q}\bar{\mathbf{x}}_1) + 2(n-\tau)(\bar{\mathbf{x}}_2 \circ \mathbf{Q}\bar{\mathbf{x}}_2) - \beta. \end{aligned}$$

5 Robustly estimating the mean and precision matrix

In practice we need an estimate $\mathbf{Q}$, and, in the anomaly problem, of $\boldsymbol{\mu}_0$, as they are very rarely known *a priori*. We will use the median of each series $\mathbf{x}_{1:n}^{(i)}$ to estimate $\boldsymbol{\mu}_0^{(i)}$. To estimate$\mathbf{Q}$ we

Algorithm 3 The approximate penalised saving for changepoint detection used in CPT-CC

Input: $\bar{\mathbf{x}}$, $\mathbf{Q}$, $\{M_d\}_{d=1}^p$, β, α_{sparse}, α_{dense}, k^*, e, s.

$\mathbf{A} = -\tau(\bar{\mathbf{x}}_1\bar{\mathbf{x}}_1^\intercal \circ \mathbf{Q}) - (n-\tau)(\bar{\mathbf{x}}_2\bar{\mathbf{x}}_2^\intercal \circ \mathbf{Q})$

$\mathbf{b} = 2\tau(\bar{\mathbf{x}}_1 \circ \mathbf{Q}\bar{\mathbf{x}}_1) + 2(n-\tau)(\bar{\mathbf{x}}_2 \circ \mathbf{Q}\bar{\mathbf{x}}_2) - \beta.$

$c = -\alpha_{\text{sparse}}$

$\tilde{S}$, $\tilde{\mathbf{J}}$ from Algorithm 1 with input ($\mathbf{A}$, $\mathbf{b}$, c, $\{M_d\}_{d=1}^p$, k^*)

$S = S(s, e, [p]) - \alpha_{\text{dense}}$.

if $\tilde{S} \geq S$ **return:** $\tilde{S}, \tilde{\mathbf{J}}$.

else return: S, $[p]$.

use a robust version of the GLASSO algorithm (Friedman et al., 2008). This algorithm takes as input an estimate of the covariance matrix, $\hat{\mathbf{\Sigma}}$, and an adjacency matrix $\mathbf{W}$. An estimate $\hat{\mathbf{Q}}(\mathbf{W})$ of $\mathbf{Q}$ is then computed by maximising the penalised log-likelihood

$$\log\det\Theta - \operatorname{tr}(\hat{\mathbf{\Sigma}}\Theta) - \|\Gamma \circ \Theta\|_1 \tag{26}$$

over non-negative definite matrices Θ, where we set $\gamma_{ij} = 0$ if $w_{ij} = 1$ or $i = j$ and $\gamma_{ij} = \infty$ (or some very high number) otherwise. This can be seen as producing the closest estimate of $\mathbf{Q}$ based on $\hat{\mathbf{\Sigma}}^{-1}$ subject to the sparsity pattern imposed by $\mathbf{W}$. To compute $\hat{\mathbf{Q}}$ efficiently, we use the R package glassoFast (Sustik and Calderhead, 2012).

As input for $\hat{\mathbf{\Sigma}}$ we use an estimate, $\mathbf{S}$, of the covariance in the raw data that is robust to the presence of anomalies. Our robust estimator is constructed from the Gaussian rank correlation and the maximum absolute deviation, as suggested by Öllerer and Croux (2015). To be precise, let $\operatorname{mad}(\mathbf{x}^{(i)})$ be the maximum absolute deviation of all measurements of variable i, and

$$r_{\text{Gauss}}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) := r\left(\Phi^{-1}\left(R\big(\mathbf{x}^{(i)}\big)/(n+1)\right), \Phi^{-1}\left(R\big(\mathbf{x}^{(j)}\big)/(n+1)\right)\right) \tag{27}$$

be the Gaussian rank correlation between variables i and j, where r is the sample Pearson correlation, and $R\big(\mathbf{x}\big)$ is a vector of the ranks of each x_t within $\mathbf{x}$. Then the robust pairwise covariances are estimated by

$$s_{ij} = \operatorname{mad}\big(\mathbf{x}_{1:n}^{(i)}\big)\operatorname{mad}\big(\mathbf{x}_{1:n}^{(j)}\big) r_{\text{Gauss}}\big(\mathbf{x}_{1:n}^{(i)}, \mathbf{x}_{1:n}^{(j)}\big). \tag{28}$$

For changepoint detection, we input $\mathbf{S}/2$, where $\mathbf{S}$ is computed on the differenced data.

A number of different considerations can go into choosing $\mathbf{W}$. From a modelling perspective, selecting $\mathbf{W}$ corresponds to deciding on a model for the conditional independence structure; $w_{ij} = 0$ means variables are assumed to be conditionally independent, while $w_{ij} = 1$ means variables are conditionally dependent. For spatial data, for example, the choice of $\mathbf{W}$ is the same as choosing the neighbourhood structure in a CAR model, where $w_{ij} = 1$ if and only if spatial region i is a neighbour of spatial region j. In our process pump example this would mean specifying which sensors are neighbours.

Computational considerations can also guide the choice of $\mathbf{W}$, however. As we have seen, CAPA-CC and CPT-CC scale exponentially in the band of $\mathbf{Q}$. Hence, the band of $\mathbf{W}$ governs the run-time of our algorithms to a large extent. A reasonable default choice of $\mathbf{W}$ is therefore a low value of r in the r-banded adjacency matrix $\mathbf{W}(r)$, defined by

$$w_{ij} = \begin{cases} 1 & \text{if } 0 < |i-j| \leq r, \\ 0 & \text{otherwise.} \end{cases} \tag{29}$$

In the simulations of the next section, we illustrate that good performance can be achieved even when specifying $\mathbf{W}$ to have a much narrower band than the true $\mathbf{Q}$.

In cases where the precision matrix is sparse but not banded, bandwidth reduction algorithms such as the Cuthill-McKee algorithm (Cuthill and McKee, 1969) and the Gibbs-Poole-Stockmeyer algorithm (Lewis, 1982) can be a useful pre-processing step before running CAPA-CC or CPT-CC.

6 Simulation study

We next turn to examine the power and estimation accuracy of CAPA-CC and CPT-CC in a range of data settings. In almost all cases, we test the robustness of the methods to incorrectly specifying the adjacency matrix in the precision matrix estimate. Like before, we concentrate on anomaly detection first, and changepoint detection afterwards.

For both problems, we have chosen a widely used one-parameter version of the *conditional autoregressive* (CAR) model called the *row-standardised* CAR model as our primary testbed (see for instance Ver Hoef et al. (2018) for a concise introduction). This CAR model is given by

$$\mathbf{Q}_{\mathrm{CAR}}(\rho, \mathbf{W}) := \mathrm{diag}(\mathbf{W}\mathbf{1}) - \rho\mathbf{W}, \tag{30}$$

where $\mathbf{W}$ is an adjacency matrix as before. $\mathbf{Q}_{\mathrm{CAR}}$ is then standardised so that $\mathbf{Q}^{-1}$ becomes a correlation matrix, and we let $\boldsymbol{\mu}_0 = \mathbf{0}$ throughout. Conveniently, the sparsity structure of $\mathbf{Q}_{\mathrm{CAR}}$ follows directly from the design of $\mathbf{W}$. In our simulations, we consider data with precision matrices corresponding to the r-banded neighbourhood structures given in (29) and regular lattice neighbourhood structures. To define the $m \times m$ lattice adjacency matrix, let (u, v) for $0 \leq u, v \leq m$ denote the coordinate of a node in the lattice. The neighbourhood of (u, v) is considered to be $\{(u-1, v), (u+1, v), (u, v-1), (u, v+1)\}$. Coordinates are then enumerated by $i = (u-1)m + v$, such that the square lattice adjacency matrix $\mathbf{W}_{\mathrm{lat}}$ can be defined by $w_{ij} = 1$ if i and j are neighbours and 0 otherwise. For the sake of brevity, we also define $\mathbf{Q}_{\mathrm{lat}}(\rho) := \mathbf{Q}_{\mathrm{CAR}}(\rho, \mathbf{W}_{\mathrm{lat}})$ and $\mathbf{Q}(\rho, r) := \mathbf{Q}_{\mathrm{CAR}}(\rho, \mathbf{W}(r))$. In addition to the CAR models, we will also test performance under the constant correlation model, given by

$$\mathbf{Q}_{\mathrm{con}}(\rho) := (\rho\mathbf{1}\mathbf{1}^{\mathsf{T}} + (1-\rho)\mathbf{I})^{-1}. \tag{31}$$

Note that we use $\mathbf{W}^*$ to refer to the true adjacency matrix of the data.

If more than one series changes, the power of different methods may depend on the how similarly each series change. To investigate this we consider the following ways of simulated anomalous or post-change means, $\boldsymbol{\mu}_k$, $k = 1, \ldots, K$: $\boldsymbol{\mu}_k^{(\mathbf{J}_k)} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{J}_k, \mathbf{J}_k})$, where $\boldsymbol{\Sigma}$ is the data covariance matrix, and $\boldsymbol{\mu}_k^{(\mathbf{J}_k)} \sim N(\mathbf{0}, (\mathbf{Q}_{\mathrm{con}}(\rho))^{-1})$. We refer to changes being drawn from the former and latter classes, respectively, by $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$ and $\boldsymbol{\mu}_{(\rho)}$. Note that $\rho = 0$ and $\rho = 1$ correspond to the special cases of the means being independent and equal for the changing variables, respectively.

After sampling a mean vector, it is scaled by a constant to achieve a specific signal strength $\vartheta_k := \|\boldsymbol{\mu}_k - \boldsymbol{\mu}_0\|_2 = \|\boldsymbol{\mu}_k\|_2$ for anomalies, and $\vartheta_{k,k-1} := \|\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k-1}\|_2$ for changes. Moreover, unless stated otherwise, we let $\mathbf{J}_k = \{1, 2, \ldots, J_k\}$, where $J_k \in [p]$ denotes the number of changing variables.

In all the simulations, the penalty functions or detection thresholds are tuned to achieve $\alpha = 0.05 \pm 0.02$ probability of a false positive in data simulated from the appropriate null distribution. In the case of a penalty function, we find $b > 0$ such that $bP(|\mathbf{J}|)$ meets this criteria. Throughout, we also set the minimum segment length $l = 2$ and the maximum segment length $M = 100$.

6.1 Single anomaly detection

To the best of our knowledge, there are no other statistical methods tailored for jointly detecting sparse and dense anomalies in correlated multivariate data. A comparison between methods for

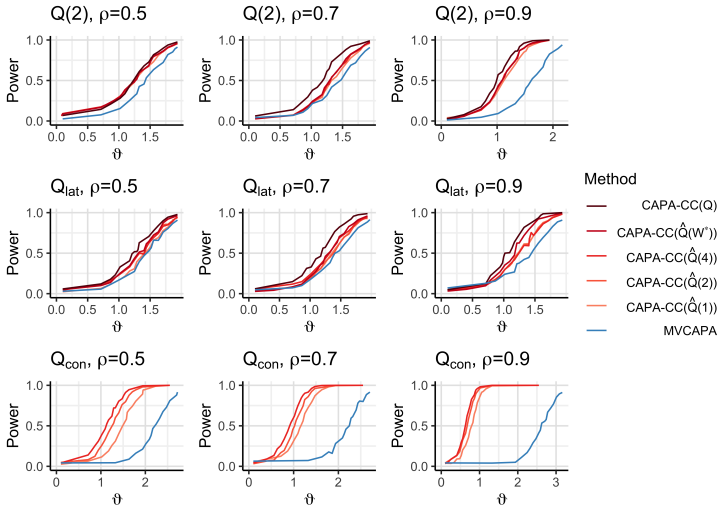

Figure 5: Power curves for correct and misspecified versions of CAPA-CC for a single known anomaly at $(s, e) = (100, 110)$ when $J = 1$ and $p = 100$. The existing MVCAPA method for iid variables is marked in blue, and the red colours correspond to versions CAPA-CC. A lighter red colour roughly means increasing misspecification of the precision matrix's structure. Results for 2-banded, lattice and globally constant correlation precision matrices are shown from top to bottom, with increasing ρ from left to right. Other parameters: $n = 200$, $\alpha = 0.05$, and 500 repetitions were used during tuning and for each point along the power curves.

independent multivariate data was performed by Fisch et al. (2019b), where their MVCAPA method was shown to generally outperform other competitors. Hence, we focus on comparing correctly and various incorrectly specified version of CAPA-CC with MVCAPA in this section, to see the worth of incorporating cross-correlations, and discover the trade-offs between the two methods.

We evaluate methods in terms of power to detect an anomaly of increasing signal strength, and also assess the correctness of the estimated subset of anomalous variables, $\mathbf{J}$.

6.1.1 Independence vs. dependence

As the performance of the anomaly detection methods we consider ultimately hinges on the performance of a test statistics at each pair (s, e), we compare performance assuming that the location of the collective anomaly is known *a priori*. That is, we fix the collective anomaly at $(s, e) = (n/2 + 1, n/2 + 10)$, and compare the power of $\tilde{S}(s, e)$ with the corresponding test statistic assuming cross-independence used within MVCAPA. In CAPA-CC, we test using the true precision matrix $\mathbf{Q}$, an estimate based on the true adjacency structure $\hat{\mathbf{Q}}(\mathbf{W}^*)$, as well as misspecified banded adjacency structures with $r = 1, 2, 4$. The power at each point along the power curve is estimated from 1000 ($p = 10$) or 500 ($p = 100$) simulated datasets, and the same datasets were used for all methods. The full set of tested scenarios include all combinations of $\{(n, p), \mathbf{Q}, \rho, J, \boldsymbol{\mu}_{(\cdot)}\}$ for $(n, p) = (100, 10), (200, 100)$, $\mathbf{Q} = \mathbf{Q}(2), \mathbf{Q}_{\text{lat}}, \mathbf{Q}_{\text{con}}$, $\rho = 0.3, 0.5, 0.7, 0.9, 0.99$, $J = 1, \lfloor\sqrt{p}\rfloor, p$, and change classes $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$, $\boldsymbol{\mu}_{(0)}$, $\boldsymbol{\mu}_{(0.8)}$, $\boldsymbol{\mu}_{(0.9)}$ and $\boldsymbol{\mu}_{(1)}$. In addition, we have also varied which series are anomalous for selected scenarios. Note that CAPA-CC($\mathbf{Q}$) represents the performance of an oracle method. For larger n relative to p, however, the difference between CAPA-CC($\mathbf{Q}$) and CAPA-CC($\hat{\mathbf{Q}}(\mathbf{W}^*)$) will decrease.

A first main finding, illustrated in Figure 5, is that for detecting a single anomalous variable,

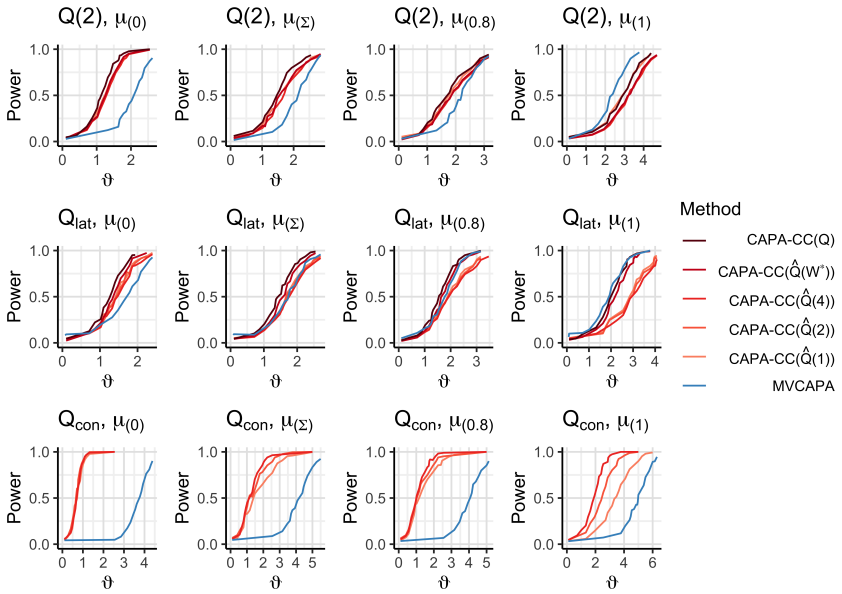

Figure 6: Power curves for a single known anomaly at $(s, e) = (100, 110)$ when $J = 10$, $p = 100$ and $\rho = 0.9$ with the same set of methods as in Figure 5. From left to right, the columns of plots show results for the anomalous means being sampled from $N(\mathbf{0}, \mathbf{I})$, $N(\mathbf{0}, \mathbf{\Sigma})$, $N(\mathbf{0}, \mathbf{Q}_{\text{con}}^{-1}(0.8))$ and the right-most column is equivalent to $\mu^{(i)} = \mu$ for all $i \in \mathbf{J}$. From top to bottom are results for 2-banded, lattice and global constant correlation data precision matrices. Other parameters: $n = 200$, $\alpha = 0.05$, and 500 repetitions were used during tuning and for each point along the power curves.

incorporating correlations lead to higher power, also when misspecifying the structure of the precision matrix estimate. The stronger the correlation, the higher the gain in power. For a collection of densely correlated variables, even using a 1-banded estimate of the precision matrix leads to a big improvement in power for sparse anomalies (the bottom row of plots).

The picture for more than one anomalous variable is more complex. Figure 6 displays the results for different precision matrices and classes of changes for $J = 10$, $p = 100$ and $\rho = 0.9$. Observe that for all precision matrices, CAPA-CC is superior for anomalous means sampled from the independent normal distribution ($\boldsymbol{\mu}_{(0)}$) and when they are sampled from a normal distribution with the data correlation matrix ($\boldsymbol{\mu}_{(\mathbf{\Sigma})}$). The power of CAPA-CC decreases, however, when the anomalous means have very similar or equal values, as in the case of means being sampled from to $\boldsymbol{\mu}_{(0.8)}$ and $\boldsymbol{\mu}_{(1)}$. Surprisingly, for the special case of equally sized anomalous means and a banded or lattice precision matrix, MVCAPA is more powerful than using the true model for the precision in CAPA-CC($\mathbf{Q}$). For $J = 100$, this is also the case for equal changes in the global constant correlation model (Figure 16 in the Supplementary Material). As we will see in Section 6.2, the same phenomenon can be observed for other methods as well, and we discuss it further in Section 8. For low values of ρ, we observe almost no difference between the two methods, which is why we focus on $\rho \geq 0.5$. For higher values of ρ than 0.9, the gain from incorporating correlations in the method increases. For $p = 10$, the corresponding results look qualitatively similar. See Section B.2 of the Supplementary Material for more details.

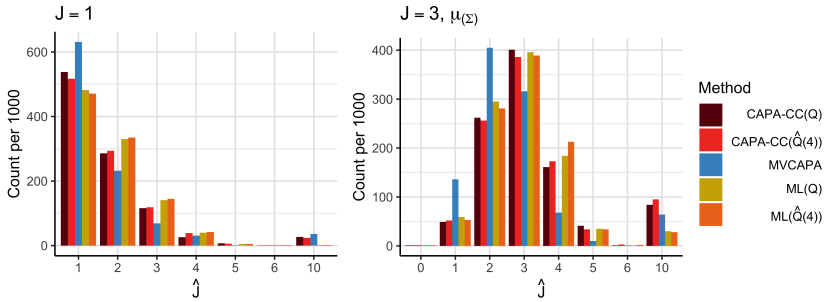

Figure 7: Estimated sizes of $\mathbf{J}$ for $\mathbf{J} = \{1\}$ (left) and $\mathbf{J} = \{1, 2, 3\}$ (right) when $p = 10$ and the location of the anomaly is assumed known. Other parameters: $n = 100$, $\mathbf{Q} = \mathbf{Q}(2, 0.9)$, $s = 10$, $e = 20$, $\vartheta = 2$, $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$, $\alpha = 0.005$.

6.1.2 Variable selection

Although CAPA-CC is not designed to estimate $\mathbf{J}$ consistently, it is worth investigating the behaviour of $\hat{\mathbf{J}}$ so that it is interpreted with sufficient caution. Note that we now use $\hat{\mathbf{J}}$ to refer to the output estimate of $\mathbf{J}$ for all algorithms. Also recall that we let $J := |\mathbf{J}|$ and $\hat{J} := |\hat{\mathbf{J}}|$.

For $p = 10$ and 100, the precision and recall of $\hat{\mathbf{J}}$ from MVCAPA as well as both true and misspecified versions of CAPA-CC were compared in the single known anomaly setting, described in Section 6.1.1. We also included the exact ML method for $p = 10$. Under a 2-banded precision matrix model, we see from Tables 3 and 4 in the Supplementary Material that both CAPA-CC and the exact ML method tend to have higher recall, but slightly lower precision, than MVCAPA. The reason for this is illustrated in Figure 7, where it can be observed that all the methods that incorporate cross-correlations overestimate J more frequently than MVCAPA. In particular, CAPA-CC more often estimates anomalies as dense. This effect is seen more clearly for $p = 100$ (Figure 20 in the Supplementary Material), where estimating J becomes increasingly hard as J grows closer to the boundary k^* between sparse and dense changes. Moreover, we found that the estimated subset is quite sensitive to the scaling of the penalties relative to the signal strength ϑ. If a more accurate estimate of $\mathbf{J}$ is desired, we thus recommend running a post-processing step by optimising the penalised saving for each anomalous segment using only the sparse penalty regime.

6.2 Single changepoint detection and estimation

We now look at changepoint detection and estimation in a single changepoint scenario, where we also compare our method to the inspect method of Wang and Samworth (2018). We focus on using CPT-CC with a $\hat{\mathbf{Q}}(4)$ precision matrix. I.e., we assume that the precision matrix model is misspecified in the rest of the section since this is most realistic, but note that improved performance on the order of what can be seen in Figures 5 and 6 could be achieved by selecting a more correct model. The version of inspect that assumes independence is available in the R package `InspectChangepoint`, and we refer to it by inspect($\mathbf{I}$). Wang and Samworth (2018) also discuss how inspect can be extended to include cross-correlations, and we have implemented this version into inspect($\hat{\mathbf{Q}}$). The inspect method does not require $\mathbf{Q}$ to be sparse, and thus we estimate it using the same robust estimator (28) that we plug into the GLASSO method for estimating $\mathbf{Q}$ in CPT-CC.

For comparing the method's power, we assume that the changepoint τ is known a priori, like in the anomaly setting. We let $\tau = n - 30$ within the same scope of data scenarios as in the anomaly setting of Section 6.1.1. A brief summary of the results are given by Figures

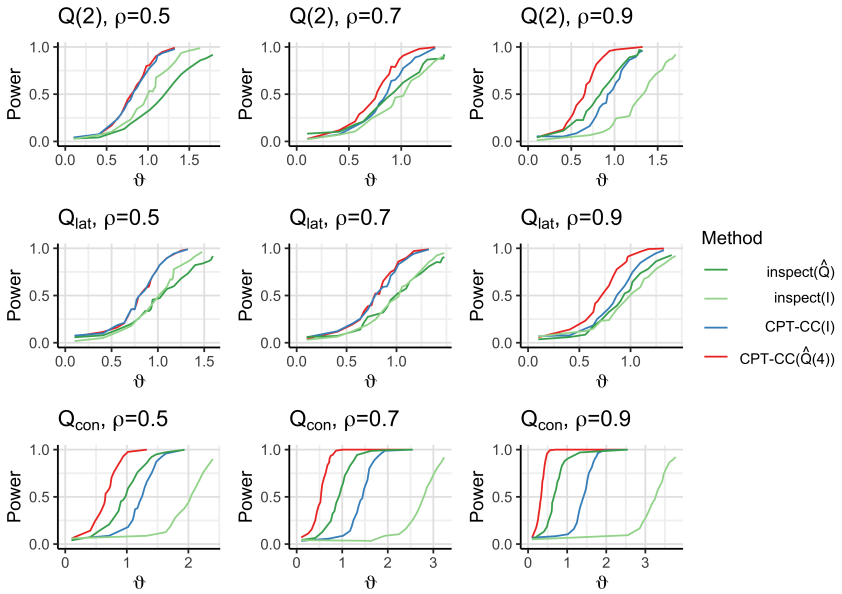

Figure 8: Power curves for a single known changepoint at $\tau = 170$ when $J = 1$ and $p = 100$. Results for 2-banded, lattice and globally constant correlation precision matrices are shown from top to bottom, with increasing ρ from left to right. Other parameters: $n = 200$, $\alpha = 0.05$, and 1000 simulated data sets were used during tuning and power estimation.

8 and 9, which correspond to the ones shown in the anomaly setting. The main conclusion is that CAPA-CC($\hat{\mathbf{Q}}(4)$) generally is the most powerful method for the models we consider. An exception from this rule can be observed for changes where more than $J/p \gtrapprox 0.1$ adjacent variables change in a very similar ($\boldsymbol{\mu}_{(\rho)}$ with $\rho \geq 0.9$) or equal way, where MVCAPA is better. Interestingly, observe that the same pattern of whether it is best to include correlations or not also holds when comparing the two versions of inspect. For more details, see Section B.3 in the Supplementary Material.

We also compare the RMSE of estimated changepoints for the four methods. For these comparisons, to avoid conflation due to method having different powers, we assume the existence of a single changepoint to be known *a priori* (as recommended by Fearnhead and Rigaill (2020)), and let all methods output their estimate of the changepoint location. For the CPT-CC methods, this is not unproblematic because testing for the existence of a changepoint is built into the method through the penalty function. As a consequence, estimates of unsignificant changepoints (when $\tilde{S} < 0$) tend to be placed at either end of the data set. For a fairer comparison, we have therefore chosen to set $\vartheta = 3$, such that almost all changes are significant. A subset of the results for $\boldsymbol{\mu}^{(\mathbf{J})} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{J},\mathbf{J}})$ are given in Table 1. We see that CAPA-CC($\hat{\mathbf{Q}}(4)$) also performs well in terms of RMSE, but that inspect is more competitive, especially for the medium sparse changes of $J = 10$. For the other change classes, the same trends as seen in the power simulations can be observed (see Section B.3 in the Supplementary Material), but with a stronger performance of inspect for $J = 10$; CAPA-CC($\hat{\mathbf{Q}}(4)$) is almost uniformly better for $\boldsymbol{\mu}_{(0)}$, while for $\boldsymbol{\mu}_{(1)}$, MVCAPA is better when $J = 100$, and either inspect($\mathbf{I}$) or inspect($\mathbf{Q}$) is best for $J = 10$.

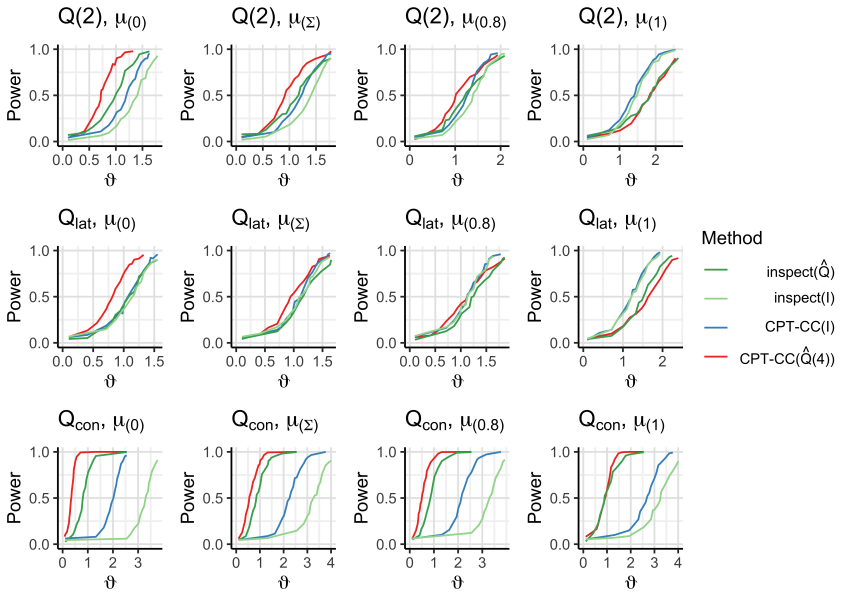

Figure 9: Power curves for a single known changepoint at $\tau = 170$ when $J = 10$ and $p = 100$. Results for 2-banded, lattice and globally constant correlation precision matrices are shown from top to bottom, with increasingly similar entries in the changed mean from left to right. Other parameters: $n = 200$, $\rho = 0.9$, $\alpha = 0.05$, and 1000 simulated data sets were used during tuning and power estimation.

6.3 Multiple anomaly detection

The simulation study is concluded by comparing the *adjusted rand index* (ARI) of CAPA-CC($\hat{\mathbf{Q}}(4)$), MVCAPA, inspect($\hat{\mathbf{Q}}$) and inspect($\mathbf{I}$) in a multiple anomaly setting with and without point anomalies (Hubert and Arabie, 1985). In this setting, the methods are used to classify observations either as anomalous (point or collective) or normal. The ARI measures the accuracy of the classification, but adjusts for the sizes of the classes. It is therefore suitable in an unbalanced classification problem such as ours.

Since inspect is not specifically made for the anomaly setting, as opposed to MVCAPA and CAPA-CC, we do not expect it to be competitive. However, since it could be used for the purpose, we include it to measure the gain of using a dedicated anomaly detection method rather than a generic changepoint detection method. Our heuristic for turning inspect into an anomaly classifier is as follows: If the sample mean of an estimated segment has L_2 norm greater than 1, the observations within the segment are classified as anomalous, and if the L_2 norm is smaller than or equal to 1, they are classified as normal.

Table 2 displays the results for $p = 100$, $n = 1000$ with three collective anomalies at $\{(s_k, e_k)\}_{k=1}^3 = \{(300, 330), (600, 620,), (900, 910)\}$, $\mathbf{J}_1 = \{1\}$, $\mathbf{J}_2 = \{1, \ldots, 10\}$ and $\mathbf{J}_3 = \{1, \ldots, 10, 46, \ldots, 55, 91, \ldots, 100\}$, with change sizes $(\vartheta_1, \vartheta_2, \vartheta_3) = (2, 4, 6)$. The results are again very favourable for CAPA-CC($\hat{\mathbf{Q}}(4)$). In the scenarios with point anomalies in particular, a lot is gained by using CAPA-CC or MVCAPA in favour of inspect. For $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$, the ARI's compare similarly as for $\boldsymbol{\mu}_{(0)}$, and for $(\vartheta_1, \vartheta_2, \vartheta_3) = (1.5, 3, 4.5)$ and $(\vartheta_1, \vartheta_2, \vartheta_3) = (1, 2, 3)$ the results are even more in CAPA-CC($\hat{\mathbf{Q}}(4)$)'s favour. The corresponding results for $p = 10$ are qualitatively similar in all respects. See Section B.4 in the Supplementary Material for details.

$\mathbf{Q}$	ρ	J	CPT-CC($\hat{\mathbf{Q}}(4)$)	CPT-CC($\mathbf{I}$)	inspect($\hat{\mathbf{Q}}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	1	**0.51**	0.55	1.38	0.55
$\mathbf{Q}(2)$	0.9	1	**0.21**	0.50	0.81	0.59
$\mathbf{Q}_{\text{lat}}$	0.5	1	0.54	0.58	1.26	**0.51**
$\mathbf{Q}_{\text{lat}}$	0.9	1	**0.37**	0.59	0.89	0.52
$\mathbf{Q}_{\text{con}}$	0.5	1	**0.29**	6.32	0.77	0.78
$\mathbf{Q}_{\text{con}}$	0.9	1	**0.00**	10.22	0.13	2.82
$\mathbf{Q}(2)$	0.5	10	0.75	0.72	1.68	**0.60**
$\mathbf{Q}(2)$	0.9	10	**0.84**	1.41	1.72	1.40
$\mathbf{Q}_{\text{lat}}$	0.5	10	0.64	0.66	1.68	**0.55**
$\mathbf{Q}_{\text{lat}}$	0.9	10	**0.78**	1.07	1.41	0.83
$\mathbf{Q}_{\text{con}}$	0.5	10	**0.74**	13.44	0.92	2.21
$\mathbf{Q}_{\text{con}}$	0.9	10	0.20	27.31	**0.18**	8.43
$\mathbf{Q}(2)$	0.5	100	**0.73**	0.77	3.07	1.11
$\mathbf{Q}(2)$	0.9	100	**0.73**	1.71	3.80	2.12
$\mathbf{Q}_{\text{lat}}$	0.5	100	**0.77**	0.79	3.02	1.11
$\mathbf{Q}_{\text{lat}}$	0.9	100	**1.05**	2.98	3.71	1.92
$\mathbf{Q}_{\text{con}}$	0.5	100	**2.63**	65.10	5.89	16.20
$\mathbf{Q}_{\text{con}}$	0.9	100	**8.38**	113.87	18.99	38.35

Table 1: RMSE of changepoint estimates for $p = 100$, $n = 200$, $\tau = 140$, $\vartheta = 3$, and $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$ changes. The smallest value is given in bold. 1000 random datasets were used for each RMSE estimate.

7 Pump data analysis

We now return to the problem of inferring anomalous segments and variables in the pump data described in the introduction. Recall that the data was preprocessed by regressing a set of monitoring variables onto a set of state variables, such that we are left with five series of residuals to detect anomalies in (Figure 1). Some of the residuals are strongly correlated (Figure 10), suggesting that incorporating cross-correlations when modelling them is advantageous based on our simulation study.

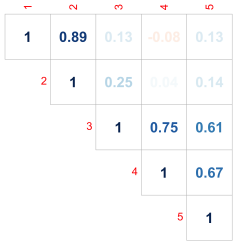

Figure 10: The robustly estimated correlation matrix (see (28)) of the pump data after preprocessing.

Before running CAPA-CC on the pump data, the penalties must be tuned and input parameters selected. The tuning of the penalties accounts for all features in the data that we have not modelled, e.g. auto-correlation, a non-stationary correlation matrix and trends in the data's mean not associated with segments of suboptimal operation. As we do not have training data guaranteed to only contain baseline observations, we instead tune the penalties such that the

$\mathbf{Q}$	ρ	$\boldsymbol{\mu}_{(\cdot)}$	Pt. anoms	CAPA-CC($\hat{\mathbf{Q}}(4)$)	CAPA-CC($\mathbf{I}$)	inspect($\hat{\mathbf{Q}}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	0	–	**0.82**	0.80	0.52	0.41
$\mathbf{Q}(2)$	0.5	0	✓	**0.84**	0.83	0.21	0.08
$\mathbf{Q}(2)$	0.9	0	–	**0.90**	0.72	0.69	0.00
$\mathbf{Q}(2)$	0.9	0	✓	**0.91**	0.78	0.23	−0.01
$\mathbf{Q}(2)$	0.5	0.8	–	0.71	**0.78**	0.49	0.32
$\mathbf{Q}(2)$	0.5	0.8	✓	0.75	**0.82**	0.18	0.09
$\mathbf{Q}(2)$	0.9	0.8	–	**0.75**	0.70	0.64	0.00
$\mathbf{Q}(2)$	0.9	0.8	✓	**0.79**	0.73	0.21	−0.02
$\mathbf{Q}_{\text{lat}}$	0.5	0	–	**0.82**	0.79	0.56	0.42
$\mathbf{Q}_{\text{lat}}$	0.5	0	✓	**0.77**	0.75	0.25	0.22
$\mathbf{Q}_{\text{lat}}$	0.9	0	–	**0.85**	0.73	0.60	0.03
$\mathbf{Q}_{\text{lat}}$	0.9	0	✓	**0.83**	0.71	0.30	0.06
$\mathbf{Q}_{\text{lat}}$	0.5	0.8	–	0.70	**0.75**	0.47	0.36
$\mathbf{Q}_{\text{lat}}$	0.5	0.8	✓	0.69	**0.74**	0.34	0.26
$\mathbf{Q}_{\text{lat}}$	0.9	0.8	–	**0.74**	0.71	0.56	0.04
$\mathbf{Q}_{\text{lat}}$	0.9	0.8	✓	**0.70**	0.70	0.36	0.06
$\mathbf{Q}_{\text{con}}$	0.5	0	–	**0.88**	0.01	0.31	0.00
$\mathbf{Q}_{\text{con}}$	0.5	0	✓	**0.90**	0.16	0.11	0.00
$\mathbf{Q}_{\text{con}}$	0.9	0	–	**1.00**	0.00	0.39	0.00
$\mathbf{Q}_{\text{con}}$	0.9	0	✓	**1.00**	0.10	0.14	0.00
$\mathbf{Q}_{\text{con}}$	0.5	0.8	–	**0.72**	0.00	0.32	0.00
$\mathbf{Q}_{\text{con}}$	0.5	0.8	✓	**0.76**	0.12	0.14	0.00
$\mathbf{Q}_{\text{con}}$	0.9	0.8	–	**0.85**	0.01	0.43	0.00
$\mathbf{Q}_{\text{con}}$	0.9	0.8	✓	**0.88**	0.09	0.15	0.00

Table 2: ARI of classifying normal and anomalous observations when $p = 100$, $n = 1000$, $(\vartheta_k)_{k=1}^3 = (2, 4, 6)$, $\{(s_k, e_k)\}_{k=1}^3 = \{(300, 330), (600, 620,), (900, 910)\}$ and $\mathbf{J}_1 = \{1\}$, $\mathbf{J}_2 = \{1, \ldots, 10\}$, $\mathbf{J}_3 = \{1, \ldots, 10, 46, \ldots, 55, 91, \ldots, 100\}$, based on 100 repetitions. Point anomalies are placed at 10 fixed locations, each randomly affecting a single variable with size sampled from $N(0, 4 \log p)$. The largest value for each data setting is given in bold.

correct number of anomalies are output to see how they align with the known ones. We do this by adjusting the scaling factor of the collective anomaly penalty function, b, while keeping the point anomaly scaling at 1. This tuning procedure resulted in a scaling factor of $b = 11$. For the remaining inputs, we set $\mathbf{Q}$ to the inverse of the correlation matrix in Figure 10, a minimum segment length $l = 5$, and use no maximum segment length.

The final result is shown in Figure 11. Before interpreting the output, it is important to know that the start points of the known anomalies are more uncertain than the end points; the end point is the time where the pump was brought back to normal operation, whereas the start point has been set based on a retrospective analysis by the engineers. With this in mind, we observe that three out of four estimated collective anomalies are within three separate known anomalous segments, with the estimated end points being more accurate than the estimated start points. The short known anomaly from $t = 125$ to $t = 135$ is missed as there is virtually no signal of it in the data. The estimated anomaly from $t = 1306$ to $t = 1362$, however, does not overlap with a known anomaly, but it clearly looks anomalous by eye. This segment is also of interest to detect since it may correspond to an unknown segment of suboptimal operation. If not, this segment points to a part of the data that fits our linear regression model poorly, indicating that a more sophisticated model might be in order if fewer false alarms are required. In general we expect that a better model for linking the state variables with the monitoring

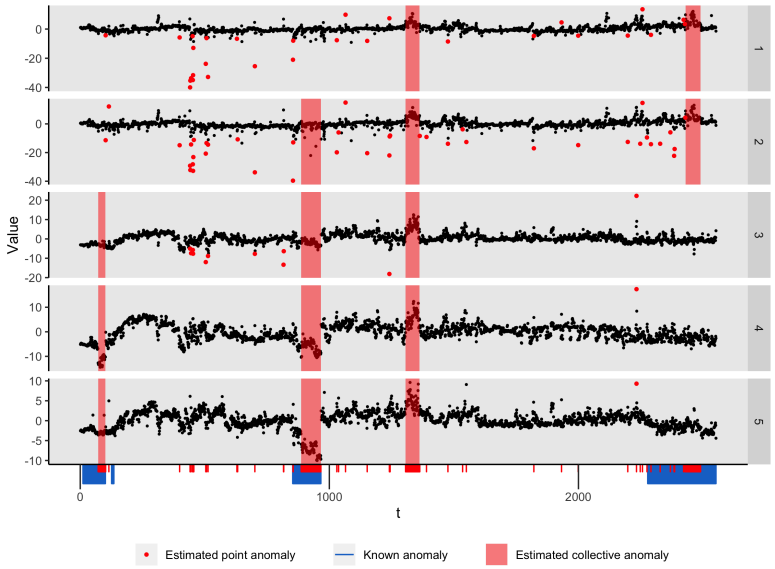

Figure 11: The four most significant estimated collective anomalies in the five residual times series derived from the pump data. Tuning parameters: $b = 11$, $b_{\text{point}} = 1$, $l = 5$ and $M = \infty$.

variables would improve the results even further because more of the trend in the mean not associated with the known anomalies would be absorbed by the model rather than leaking into the residuals.

In addition, notice the importance of including point anomalies in the analysis for this application. Rerunning CAPA-CC on the data without inferring point anomalies resulted in four additional false collective anomalies being inferred for $b = 11$.

8 Conclusions

In this article, we have proposed computationally efficient penalised cost-based methods for detecting multiple sparse and dense anomalies or changes in the mean of cross-correlated data. In addition to estimating the locations of the changepoints, the methods indicate which components are affected by a change. This is important to understand why and how changes or anomalies have occured. At the computational core of these methods lies a novel dynamic programming algorithm for solving banded unconstrained binary quadratic programs which approximate the Gaussian likelihood ratio test for a subset mean change.

The motivation of our methodological development comes from condition monitoring of an industrial process pump, where strong cross-correlations between spatially adjacent sensor measurements could be observed. Although several modelling assumptions were violated, three out of four known anomalies could be detected, with only one potential false alarm, when analysing the data with CAPA-CC. Even better results can be expected by using a more accurate model to remove trends not associated with anomalies. Also of interest for this application is being able to detect collective anomalies in real-time. The CAPA framework we have adopted has been shown to be able to be applied in online settings (Fisch et al., 2020), and similar ideas could be used to produce a sequential version of CAPA-CC.

When assessing the method's performance empirically, special attention was paid to how incorporating cross-correlations in the model affected the results compared to ignoring it as

most existing methods do. We found that for low to medium levels of dependence there was almost no difference in power or estimation accuracy; e.g. for $\rho < 0.5$ in the 2-banded and lattice precision matrices, and $\rho < 0.2$ for the constant correlation matrix, in the case of $p = 100$ variables. For increasingly stronger dependence above these levels, either in the form of a denser precision matrix or higher correlation parameter, the benefit of including cross-correlation in the model of the data grows in almost all tested cases.

The exception to this rule is connected to the somewhat surprising finding that the shape of the change in mean across variables influences the magnitude of the advantage of including cross-correlations quite strongly. In positively correlated data, changes that affect many series and are of very similar, or the same, size for each series can be harder to detect when including cross-correlations in the model. For example, in a model with strong positive correlations, it is much harder to detect if a moderately large amount of variables changes by the same amount in the same direction, than if these variables changes by varying amounts in wildly varying directions. The intuition behind this is that in the former case, the change mimics the expected behaviour of the data given the variables' strong positive dependence, while in the latter, the change strongly violates the model's expectation. The model assuming independence, on the other hand, is completely agnostic to the shape of the changed mean vector. As a result, the benefits of including correlations in the model is small, or perhaps even negative, if variables in the data is strongly dependent, and interest lies on detecting moderately sparse to dense and similarly changing variables.

Acknowledgements

We are grateful to OneSubsea for sharing their data with us, and to Alex Fisch and Daniel Grose for helpful discussions. This work is partially funded by the Norwegian Research Council project 237718 (Big Insight) and EPSRC grant EP/N031938/1 (STATSCALE).

Supplementary Material

Supplementary material Proofs of the propositions, additional comments to Proposition 2, and detailed results from the simulation study.

Code Efficient implementations of the CAPA-CC and CPT-CC algorithms as well as the code for reproducing the simulation study is available in the R package `capacc`, downloadable at `https://github.com/Tveten/capacc`. CAPA-CC will be included in a future version of the R package `anomaly` on CRAN, which contains the CAPA family of methods.

References

Bardwell, L., Fearnhead, P., Eckley, I. A., Smith, S., and Spott, M. (2019). Most Recent Changepoint Detection in Panel Data. *Technometrics*, 61(1):88–98.

Bhattacharjee, M., Banerjee, M., and Michailidis, G. (2019). Change Point Estimation in Panel Data with Temporal and Cross-sectional Dependence. *arXiv:1904.11101 [math.ST]*.

Cho, H. (2016). Change-point detection in panel data via double CUSUM statistic. *Electronic Journal of Statistics*, 10(2):2000–2038.

Cho, H. and Fryzlewicz, P. (2015). Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 77(2):475–507.

Cuthill, E. and McKee, J. (1969). Reducing the bandwidth of sparse symmetric matrices. In *Proceedings of the 1969 24th national conference*, ACM '69, pages 157–172, New York, NY, USA. Association for Computing Machinery.

Egusquiza, E., Valero, C., Valentin, D., Presas, A., and Rodriguez, C. G. (2015). Condition monitoring of pump-turbines. New challenges. *Measurement*, 67:151–163.

Fearnhead, P. and Rigaill, G. (2019). Changepoint Detection in the Presence of Outliers. *Journal of the American Statistical Association*, 114(525):169–183.

Fearnhead, P. and Rigaill, G. (2020). Relating and comparing methods for detecting changes in mean. *Stat*, 9(1):e291.

Fisch, A. T. M., Bardwell, L., and Eckley, I. A. (2020). Real Time Anomaly Detection And Categorisation. *arXiv:2009.06670 [stat.ME]*.

Fisch, A. T. M., Eckley, I. A., and Fearnhead, P. (2019a). A linear time method for the detection of point and collective anomalies. *arXiv:1806.01947 [stat.ML]*.

Fisch, A. T. M., Eckley, I. A., and Fearnhead, P. (2019b). Subset Multivariate Collective And Point Anomaly Detection. *arXiv:1909.01691 [stat.ME]*.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Fryzlewicz, P. (2014). Wild binary segmentation for multiple change-point detection. *Annals of Statistics*, 42(6):2243–2281.

Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co, New York, NY, USA.

Henriquez, P., Alonso, J. B., Ferrer, M. A., and Travieso, C. M. (2014). Review of Automatic Fault Diagnosis Systems Using Audio and Vibration Signals. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(5):642–652.

Horváth, L. and Hušková, M. (2012). Change-point detection in panel data. *Journal of Time Series Analysis*, 33(4):631–648.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1):193–218.

Jeng, X. J., Cai, T. T., and Li, H. (2013). Simultaneous discovery of rare and common segment variants. *Biometrika*, 100(1):157–172.

Jirak, M. (2015). Uniform change point tests in high dimension. *The Annals of Statistics*, 43(6):2451–2483.

Killick, R., Fearnhead, P., and Eckley, I. A. (2012). Optimal Detection of Changepoints With a Linear Computational Cost. *Journal of the American Statistical Association*, 107(500):1590–1598.

Kirch, C., Muhsal, B., and Ombao, H. (2015). Detection of Changes in Multivariate Time Series With Application to EEG Data. *Journal of the American Statistical Association*, 110(511):1197–1216.

Klanderman, M. C., Newhart, K. B., Cath, T. Y., and Hering, A. S. (2020). Fault isolation for a complex decentralized waste water treatment facility. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 69(4):931–951.

Kovács, S., Li, H., Bühlmann, P., and Munk, A. (2020). Seeded Binary Segmentation: A general methodology for fast and optimal change point detection. *arXiv:2002.06633 [stat.ME]*.

Lewis, J. G. (1982). Algorithm 582: The Gibbs-Poole-Stockmeyer and Gibbs-King Algorithms for Reordering Sparse Matrices. *ACM Transactions on Mathematical Software (TOMS)*, 8(2):190–194.

Li, J., Xu, M., Zhong, P.-S., and Li, L. (2019). Change Point Detection in the Mean of High-Dimensional Time Series Data under Dependence. *arXiv:1903.07006 [stat.ME]*.

Liu, H., Gao, C., and Samworth, R. J. (2019). Minimax rates in sparse, high-dimensional changepoint detection. *arXiv:1907.10012 [math.ST]*.

Öllerer, V. and Croux, C. (2015). Robust High-Dimensional Precision Matrix Estimation. In Nordhausen, K. and Taskinen, S., editors, *Modern Nonparametric, Robust and Multivariate Methods*, pages 325–350. Springer International Publishing, Cham.

Sustik, M. A. and Calderhead, B. (2012). GLASSOFAST: An efficient GLASSO implementation. *UTCS Technical Report*, TR-12-29.

Tchakoua, P., Wamkeue, R., Ouhrouche, M., Slaoui-Hasnaoui, F., Tameghe, T. A., and Ekemb, G. (2014). Wind Turbine Condition Monitoring: State-of-the-Art Review, New Trends, and Future Challenges. *Energies*, 7(4):2595–2630.

Ver Hoef, J. M., Hanks, E. M., and Hooten, M. B. (2018). On the relationship between conditional (CAR) and simultaneous (SAR) autoregressive models. *Spatial Statistics*, 25:68–85.

Wang, T. and Samworth, R. J. (2018). High dimensional change point estimation via sparse projection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(1):57–83.

Westerlund, J. (2019). Common Breaks in Means for Cross-Correlated Fixed-T Panel Data. *Journal of Time Series Analysis*, 40(2):248–255.

Xie, Y. and Siegmund, D. (2013). Sequential multi-sensor change-point detection. *The Annals of Statistics*, 41(2):670–692.

Supplementary Material

A Proofs and additional comments

A.1 Proof of Proposition 1

First rewrite the optimisation problem in terms of the binary vector $\mathbf{u}$:

$$\tilde{S}(s,e) = \max_{\mathbf{u}} (e-s)\left[2\bar{\mathbf{x}}^\intercal \mathbf{Q}(\bar{\mathbf{x}} \circ \mathbf{u}) - (\bar{\mathbf{x}} \circ \mathbf{u})^\intercal \mathbf{Q}(\bar{\mathbf{x}} \circ \mathbf{u})\right] - \beta \mathbf{1}^\intercal \mathbf{u}.$$

The proof is completed by using properties of the Hadamard product and its relations to the regular matrix product to reexpress the optimal savings as

$$\begin{aligned}
\tilde{S}(s,e) &= \max_{\mathbf{u}} (e-s)\left[2\bar{\mathbf{x}}^\intercal \mathbf{Q}\mathrm{diag}(\bar{\mathbf{x}})\mathbf{u} - \mathbf{u}^\intercal(\bar{\mathbf{x}}\bar{\mathbf{x}}^\intercal \circ \mathbf{Q})\mathbf{u}\right] - \beta \mathbf{1}^\intercal \mathbf{u} \\
&= \max_{\mathbf{u}} \left[2(e-s)\bar{\mathbf{x}}^\intercal \mathbf{Q}\mathrm{diag}(\bar{\mathbf{x}}) - \beta\right]\mathbf{u} + \mathbf{u}^\intercal\left[-(e-s)\bar{\mathbf{x}}\bar{\mathbf{x}}^\intercal \circ \mathbf{Q}\right]\mathbf{u}. \\
&= \max_{\mathbf{u}} \mathbf{u}^\intercal\left[2(e-s)\bar{\mathbf{x}} \circ \mathbf{Q}\bar{\mathbf{x}} - \beta\right] + \mathbf{u}^\intercal\left[-(e-s)\bar{\mathbf{x}}\bar{\mathbf{x}}^\intercal \circ \mathbf{Q}\right]\mathbf{u}.
\end{aligned}$$

A.2 Proof of Proposition 2 with comments

Let $\tilde{\mathbf{J}} := \mathrm{argmax}_{\mathbf{J}}\left[\tilde{S}(s,e,\mathbf{J}) - P(|\mathbf{J}|)\right]$ and $\hat{\mathbf{J}} := \mathrm{argmax}_{\mathbf{J}}\left[S(s,e,\mathbf{J}) - P(|\mathbf{J}|)\right]$. In the following, we omit s and e in the notation of $S(s,e,\mathbf{J})$ and $\tilde{S}(s,e,\mathbf{J})$, such that $S(\hat{\mathbf{J}}) = S(s,e)$ and $\tilde{S}(\tilde{\mathbf{J}}) = \tilde{S}(s,e)$ in Proposition 2.

The lower bound $S(\hat{\mathbf{J}}) - \tilde{S}(\tilde{\mathbf{J}}) \geq 0$ follows from S using the exact MLE and $\tilde{S}$ using an alternative estimator. I.e., $S(\mathbf{J}) \geq \tilde{S}(\mathbf{J})$ for all subsets $\mathbf{J}$. Hence, $S(\hat{\mathbf{J}}) \geq S(\tilde{\mathbf{J}}) \geq \tilde{S}(\tilde{\mathbf{J}})$.

To obtain the upper bound, let $\Delta S(\mathbf{J}) := S(\mathbf{J}) - \tilde{S}(\mathbf{J})$. Now, as $\tilde{\mathbf{J}}$ is the maximiser of $\tilde{S}$, we get that

$$S(\hat{\mathbf{J}}) - \tilde{S}(\tilde{\mathbf{J}}) = \tilde{S}(\hat{\mathbf{J}}) + \Delta S(\hat{\mathbf{J}}) - \tilde{S}(\tilde{\mathbf{J}}) \leq \Delta S(\hat{\mathbf{J}}).$$

The final result immediately follows;

$$\begin{aligned}
\Delta S(\hat{\mathbf{J}}) &= (e-s)(2\bar{\mathbf{x}}(\hat{\mathbf{J}}^{\mathbf{c}}) - \mathbf{W}(\hat{\mathbf{J}})\bar{\mathbf{x}}(\hat{\mathbf{J}}^{\mathbf{c}}))^\intercal \mathbf{Q}\mathbf{W}(\hat{\mathbf{J}})\bar{\mathbf{x}}(\hat{\mathbf{J}}^{\mathbf{c}}) \\
&\leq 2(e-s)\bar{\mathbf{x}}(\hat{\mathbf{J}}^{\mathbf{c}})^\intercal \mathbf{Q}\mathbf{W}(\hat{\mathbf{J}})\bar{\mathbf{x}}(\hat{\mathbf{J}}^{\mathbf{c}}) \\
&\leq 2(e-s)\lambda_{\max}\big(\mathbf{Q}\mathbf{W}(\hat{\mathbf{J}})\big)\|\bar{\mathbf{x}}(\hat{\mathbf{J}}^{\mathbf{c}})\|^2.
\end{aligned}$$

The first inequality is due to the positive semi-definiteness of the quadratic form in the second term, while the second inequality is a standard result on quadratic forms. This concludes the proof.

The following arguments suggests that the worst-case scenario for the approximation is sparse changes in strongly correlated data, as is also observed in the simulations of Section B.1. First observe that $\|\bar{\mathbf{x}}_{(s+1):e}(\hat{\mathbf{J}}^{\mathbf{c}})\|^2$ grows as $|\hat{\mathbf{J}}|$ becomes smaller. Moreover, under the cross-correlated multivariate normal model with means equal to 0 and variances equal to 1,

$$(e-s)\big\|\bar{\mathbf{x}}_{(s+1):e}\big(\hat{\mathbf{J}}^{\mathbf{c}}\big)\big\|^2 = \sum_{j \in \hat{\mathbf{J}}^{\mathbf{c}}} (e-s)\big(\bar{x}^{(j)}_{(s+1):e}\big)^2, \tag{32}$$

where $(e-s)\big(\bar{x}^{(j)}_{(s+1):e}\big)^2$ for $j \in \hat{\mathbf{J}}^{\mathbf{c}}$ are dependent χ^2_1 random variables. By standard rules of expectation and variance, we get that the expected value of (32) is $|\hat{\mathbf{J}}^{\mathbf{c}}|$ and the variance is $\big(|\hat{\mathbf{J}}^{\mathbf{c}}| + \sum_{i \neq j \in \hat{\mathbf{J}}^{\mathbf{c}}} \omega_{i,j}\big)$, where $\omega_{i,j}$ is the pairwise covariance between $\big(\bar{x}^{(i)}_{(s+1):e}\big)^2$ and $\big(\bar{x}^{(j)}_{(s+1):e}\big)^2$. If we assume the zero-mean model to hold for the variables in the estimated non-anomalous variables $\hat{\mathbf{J}}^{\mathbf{c}}$, we thus see that approximation may get worse as the change becomes sparser and the strength of the correlation increases in positive direction.

Note that this analysis only contains half of the picture as $\lambda_{\max}\big(\mathbf{Q}\mathbf{W}(\hat{\mathbf{J}})\big)$ seems intractable to study theoretically even for simple examples of $\mathbf{Q}$. Our numerical experimentation, however, suggests that $\lambda_{\max}\big(\mathbf{Q}\mathbf{W}(\hat{\mathbf{J}})\big)$ also grows as the correlations increase. In addition, the simulation results in Section B.1 in the Supplementary Material agree with the conclusion that the greatest difference in performance occurs when there is a sparse change in strongly correlated data, although the difference is small in the tested low p settings.

A.3 Proof of Proposition 3

This proof follows the lines of the proof of Theorem 3.1 in Killick et al. (2012). First, recall the expression for the approximate savings,

$$\tilde{S}(s, e, \mathbf{J}) = (e - s)\left[2\bar{\mathbf{x}}^{\intercal}\mathbf{Q}\bar{\mathbf{x}}(\mathbf{J}) - \bar{\mathbf{x}}(\mathbf{J})^{\intercal}\mathbf{Q}\bar{\mathbf{x}}(\mathbf{J})\right],$$

and that we write $\tilde{S}(s, e) = \max_{\mathbf{J}}[\tilde{S}(s, e, \mathbf{J}) - P(\mathbf{J})]$ for the optimal penalised approximate savings. Next, observe that

$$\begin{aligned}
\max_{\mathbf{J}}\left[\tilde{S}(t, m, \mathbf{J}) - P(\mathbf{J})\right] + \max_{\mathbf{J}}\tilde{S}(m, m', \mathbf{J}) &\geq \max_{\mathbf{J}}\left[\tilde{S}(t, m', \mathbf{J}) - P(\mathbf{J})\right] \\
\max_{\mathbf{J}}\left[\tilde{S}(t, m, \mathbf{J}) - P(\mathbf{J})\right] + \max_{\mathbf{J}}\left[\tilde{S}(m, m', \mathbf{J}) - P(\mathbf{J})\right] + \max_{\mathbf{J}}P(\mathbf{J}) &\geq \max_{\mathbf{J}}\left[\tilde{S}(t, m', \mathbf{J}) - P(\mathbf{J})\right] \\
\tilde{S}(t, m) + \tilde{S}(m, m') + \max_{\mathbf{J}}P(\mathbf{J}) &\geq \tilde{S}(t, m') \qquad (33)
\end{aligned}$$

The inequality follows because of the basic fact that we are maximising over more parameters on the left-hand side than on the right-hand side, while adding the maximum penalty in the left-hand side guarantees that the additional penalty term is canceled out. As a consequence of (33), and assuming that

$$C(t) + \tilde{S}(t, m) + \max_{\mathbf{J}}P(\mathbf{J}) \leq C(m)$$

holds, we see that for all future times $m' \geq m + l$,

$$\begin{aligned}
C(t) + \tilde{S}(t, m) + \tilde{S}(m, m') + \max_{\mathbf{J}}P(\mathbf{J}) &\leq C(m) + \tilde{S}(m, m') \\
C(t) + \tilde{S}(t, m') &\leq C(m').
\end{aligned}$$

The proof is concluded by noting that for the penalty given in (5), $\max_{\mathbf{J}} P(\mathbf{J}) = \alpha_{\text{dense}}$.

A.4 Proof of Proposition 4

The proof follows the same steps as in the proof of Proposition 1 in Section A.1.

B Additional simulation results

B.1 Approximation vs. MLE

In this section, we compare the power of our approximation in CAPA-CC with the exact ML method in a data scenario with $n = 100$ observations from a $N(\boldsymbol{\mu}_t, \mathbf{Q}(\rho, 2)^{-1})$ distribution with a single collective anomaly at $(s, e) = (50, 60)$ when $p = 10$. As in Section 6.1.1 in the main text, we assume that the location of the anomaly is known. Within this setup, we focus on varying the change class, ρ, p and $J = |\mathbf{J}|$. The penalty function for a given precision matrix was tuned for CAPA-CC and reused in the ML method for computational reasons. Proposition 2 guarantees that CAPA-CC is in a disadvantage, if anything, under this choice.

As can be seen from Figure 12, almost no power is lost in the low dimensional setting by using our approximation rather than the exact ML method, both when using the true precision

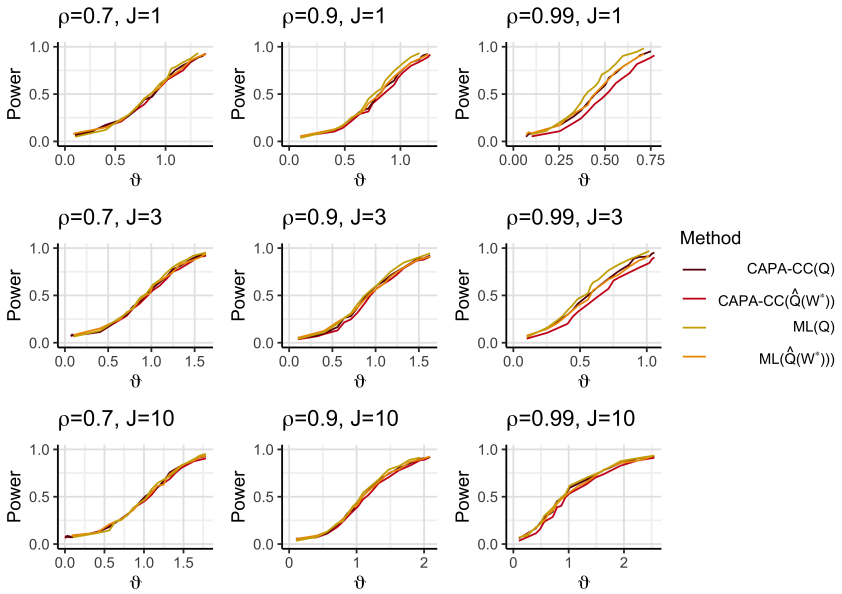

Figure 12: Power curves of our approximation and the exact ML method with the true precision matrix (dark red and gold lines) and an estimate using the true adjacency matrix (red and orange lines). Plot-wise from left to right, the correlation grows, and the number of anomalous variables grows from top to bottom. Other parameters: $n = 100$, $p = 10$, $\mathbf{Q} = \mathbf{Q}(2)$, $s = 50$, $e = 60$, change class $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$, and 1000 repetitions were used during tuning and power estimation.

matrix and when the precision matrix is estimated from the true adjacency matrix. It is only in the scenarios with a very high correlation of 0.99 and a relatively sparse change of $J = 1, 3$ that there is a notable difference between the two methods for each precision matrix $\mathbf{Q}$ and $\hat{\mathbf{Q}}(\mathbf{W}^*)$. We should point out that this difference may become bigger as p grows. For $p = 5, 10$ and15, however, the results are very similar (Figure 13). All the results for $\boldsymbol{\mu}_{(0)}$ (i.i.d.) and $\boldsymbol{\mu}_{(1)}$ (equal) changes were qualitatively similar to the results for $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$ shown here.

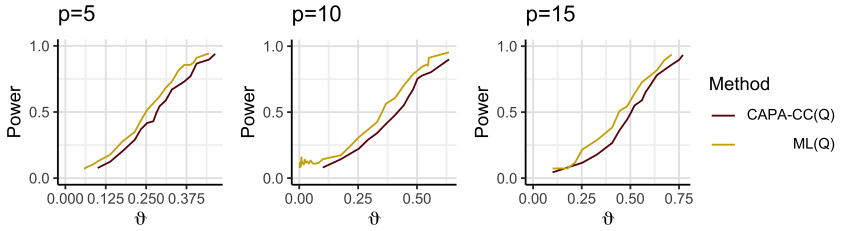

Figure 13: Power comparison of the approximation and the MLE with the true precision matrix for $p = 5, 10, 15$ in the worst-case scenario of a single changing variable in highly correlated data (the top right scenario in Figure 12). Other parameters: $n = 100$, $s = 50$, $e = 60$, and 1000 repetitions were used during tuning and power estimation.

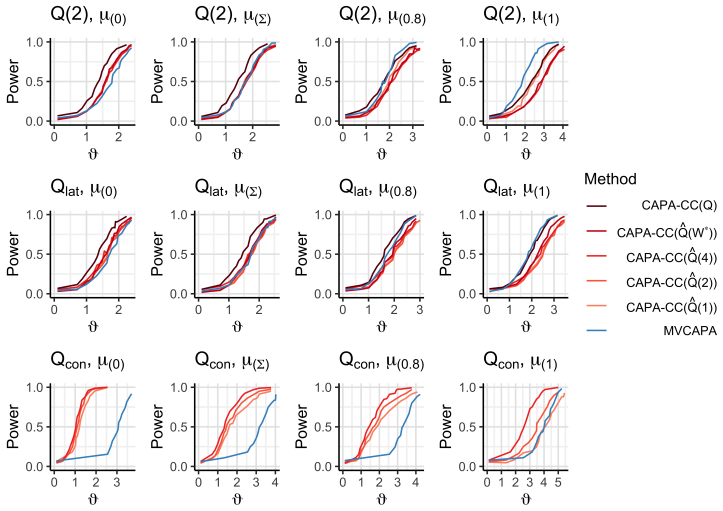

Figure 14: Power curves for $J = 10$, $p = 100$, $\rho = 0.7$, $n = 200$, $(s, e) = (100, 110)$, $\alpha = 0.05$.

B.2 Single anomaly detection

Figures 14-19 display additional simulation results for comparing power when incorporating dependence in the method versus ignoring it in the single anomaly setting of Section 6.1.1 in the main text. For more results on variable selection in the single anomaly setting, see Figure 20 and Tables 3 and 4.

B.3 Single changepoint detection and estimation

Further simulation results on power in the single changepoint setting is given in Figure 21-25. Tables 5-10 give additional results on the RMSE of changepoint estimates.

B.4 Multiple anomaly detection

A supplementary result on multiple anomaly detection is given in Table 11. The setup is precisely the same as in Table 2 in the main text, with the exception that the changes are of half the size.

Table 3: Average precision, recall and $\hat{J}$ over 1000 repetitions for $p = 10$ and $n = 100$. Other parameters: $\mathbf{Q} = \mathbf{Q}(2)$, $s = n/10$ and $e = s + 10$, $\boldsymbol{\mu}_{(\Sigma)}$, $\alpha = 0.005$.

J	ϑ	ρ	Method	$\hat{J}$	Precision	Recall
1	2	0.5	MVCAPA	1.66	0.73	1.00
1	2	0.5	CAPA-CC($\mathbf{Q}$)	1.97	0.66	1.00
1	2	0.5	ML($\mathbf{Q}$)	1.94	0.65	1.00
1	2	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	1.77	0.70	1.00
1	2	0.5	ML($\hat{\mathbf{Q}}(4)$)	1.77	0.70	1.00
1	2	0.9	MVCAPA	1.79	0.78	1.00
1	2	0.9	CAPA-CC($\mathbf{Q}$)	1.87	0.73	1.00
1	2	0.9	ML($\mathbf{Q}$)	1.77	0.71	1.00
1	2	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	1.89	0.72	1.00
1	2	0.9	ML($\hat{\mathbf{Q}}(4)$)	1.79	0.70	1.00
1	5	0.5	MVCAPA	1.66	0.74	1.00
1	5	0.5	CAPA-CC($\mathbf{Q}$)	1.92	0.68	1.00
1	5	0.5	ML($\mathbf{Q}$)	1.90	0.68	1.00
1	5	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	1.77	0.71	1.00
1	5	0.5	ML($\hat{\mathbf{Q}}(4)$)	1.73	0.71	1.00
1	5	0.9	MVCAPA	1.66	0.81	1.00
1	5	0.9	CAPA-CC($\mathbf{Q}$)	1.84	0.72	1.00
1	5	0.9	ML($\mathbf{Q}$)	1.75	0.71	1.00
1	5	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	1.85	0.73	1.00
1	5	0.9	ML($\hat{\mathbf{Q}}(4)$)	1.81	0.69	1.00
3	2	0.5	MVCAPA	2.80	0.83	0.70
3	2	0.5	CAPA-CC($\mathbf{Q}$)	3.25	0.78	0.74
3	2	0.5	ML($\mathbf{Q}$)	3.15	0.78	0.73
3	2	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	2.97	0.81	0.72
3	2	0.5	ML($\hat{\mathbf{Q}}(4)$)	2.88	0.81	0.70
3	2	0.9	MVCAPA	2.86	0.87	0.72
3	2	0.9	CAPA-CC($\mathbf{Q}$)	3.47	0.82	0.81
3	2	0.9	ML($\mathbf{Q}$)	3.05	0.83	0.77
3	2	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	3.55	0.80	0.81
3	2	0.9	ML($\hat{\mathbf{Q}}(4)$)	3.10	0.81	0.77
3	5	0.5	MVCAPA	3.42	0.85	0.88
3	5	0.5	CAPA-CC($\mathbf{Q}$)	3.85	0.80	0.90
3	5	0.5	ML($\mathbf{Q}$)	3.80	0.80	0.90
3	5	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	3.56	0.83	0.89
3	5	0.5	ML($\hat{\mathbf{Q}}(4)$)	3.47	0.84	0.89
3	5	0.9	MVCAPA	3.53	0.88	0.90
3	5	0.9	CAPA-CC($\mathbf{Q}$)	4.12	0.81	0.93
3	5	0.9	ML($\mathbf{Q}$)	3.63	0.83	0.92
3	5	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	4.16	0.80	0.93
3	5	0.9	ML($\hat{\mathbf{Q}}(4)$)	3.65	0.82	0.92

Table 4: Average precision, recall and $\hat{J}$ over 1000 repetitions for $p = 100$ and $n = 200$. Other parameters: $\mathbf{Q} = \mathbf{Q}(2)$, $s = n/10$ and $e = s + 10$, $\boldsymbol{\mu}_{(\Sigma)}$, $\alpha = 0.005$.

J	ϑ	ρ	Method	$\hat{J}$	Precision	Recall
1	2	0.5	MVCAPA	10.25	0.63	1.00
1	2	0.5	CAPA-CC($\mathbf{Q}$)	10.40	0.64	1.00
1	2	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	11.62	0.62	1.00
1	2	0.9	MVCAPA	4.45	0.80	1.00
1	2	0.9	CAPA-CC($\mathbf{Q}$)	10.74	0.69	1.00
1	2	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	13.73	0.66	1.00
1	5	0.5	MVCAPA	9.65	0.63	1.00
1	5	0.5	CAPA-CC($\mathbf{Q}$)	10.15	0.62	1.00
1	5	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	9.79	0.62	1.00
1	5	0.9	MVCAPA	5.21	0.81	1.00
1	5	0.9	CAPA-CC($\mathbf{Q}$)	11.44	0.68	1.00
1	5	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	11.84	0.67	1.00
5	2	0.5	MVCAPA	27.83	0.59	0.55
5	2	0.5	CAPA-CC($\mathbf{Q}$)	29.32	0.59	0.57
5	2	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	30.00	0.56	0.56
5	2	0.9	MVCAPA	11.35	0.79	0.42
5	2	0.9	CAPA-CC($\mathbf{Q}$)	34.55	0.56	0.63
5	2	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	38.06	0.50	0.63
5	5	0.5	MVCAPA	44.22	0.53	0.84
5	5	0.5	CAPA-CC($\mathbf{Q}$)	47.75	0.51	0.85
5	5	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	51.34	0.47	0.86
5	5	0.9	MVCAPA	21.58	0.78	0.79
5	5	0.9	CAPA-CC($\mathbf{Q}$)	55.21	0.46	0.91
5	5	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	58.26	0.42	0.91
10	2	0.5	MVCAPA	38.26	0.55	0.50
10	2	0.5	CAPA-CC($\mathbf{Q}$)	42.77	0.51	0.54
10	2	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	44.63	0.48	0.54
10	2	0.9	MVCAPA	14.15	0.76	0.28
10	2	0.9	CAPA-CC($\mathbf{Q}$)	49.74	0.43	0.60
10	2	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	52.24	0.40	0.60
10	5	0.5	MVCAPA	85.29	0.23	0.93
10	5	0.5	CAPA-CC($\mathbf{Q}$)	88.40	0.20	0.94
10	5	0.5	CAPA-CC($\hat{\mathbf{Q}}(4)$)	89.42	0.19	0.95
10	5	0.9	MVCAPA	51.88	0.55	0.78
10	5	0.9	CAPA-CC($\mathbf{Q}$)	94.77	0.15	0.98
10	5	0.9	CAPA-CC($\hat{\mathbf{Q}}(4)$)	95.97	0.14	0.98

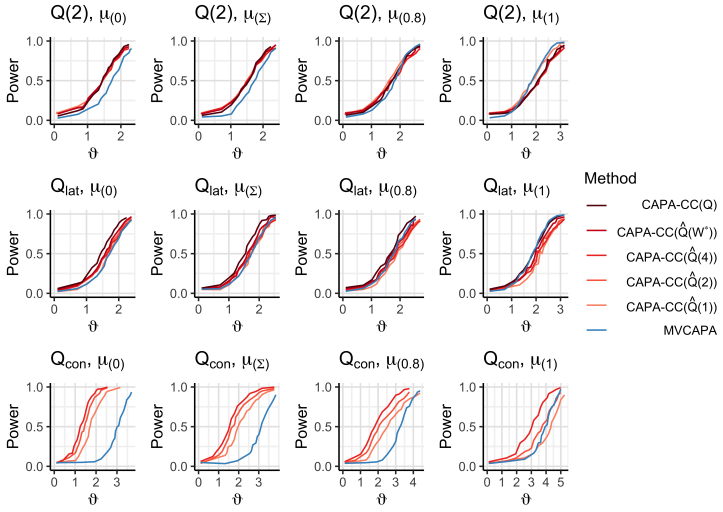

Figure 15: Power curves for $J = 10$, $p = 100$, $\rho = 0.5$, $n = 200$, $(s, e) = (100, 110)$, $\alpha = 0.05$.

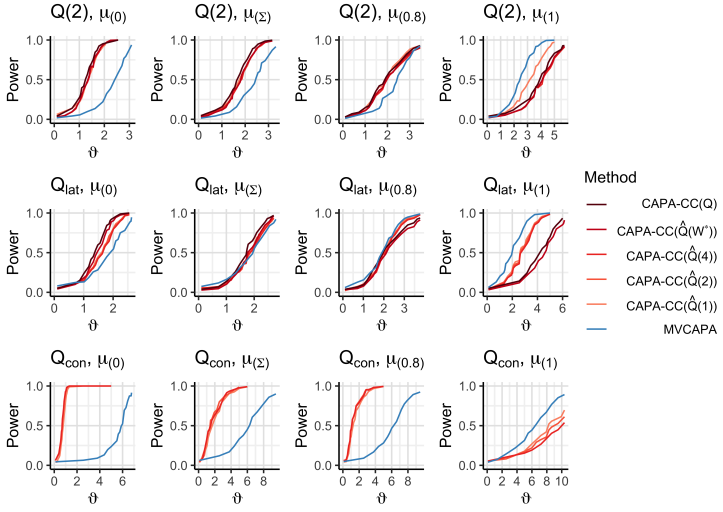

Figure 16: Power curves for $J = 100$, $p = 100$, $\rho = 0.9$, $n = 200$, $(s, e) = (100, 110)$, $\alpha = 0.05$.

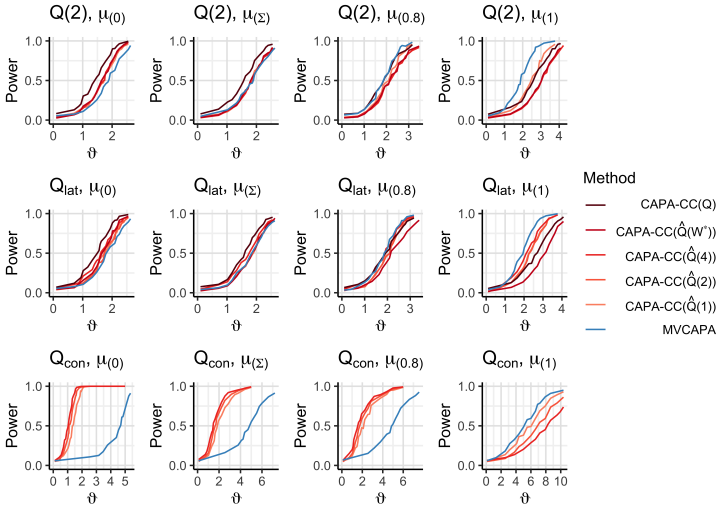

Figure 17: Power curves for $J = 100$, $p = 100$, $\rho = 0.7$, $n = 200$, $(s, e) = (100, 110)$, $\alpha = 0.05$.

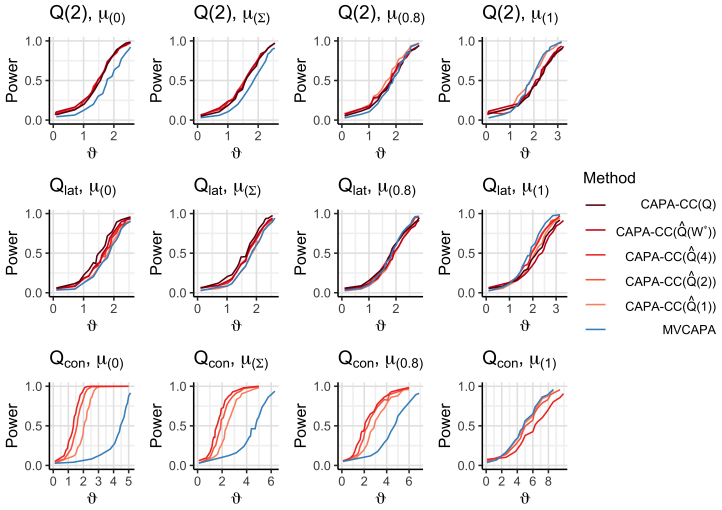

Figure 18: Power curves for $J = 100$, $p = 100$, $\rho = 0.5$, $n = 200$, $(s, e) = (100, 110)$, $\alpha = 0.05$.

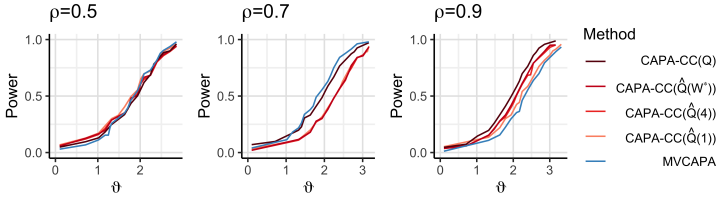

Figure 19: Power curves for $\mathbf{J} = \{1, 2, 3, 4, 50, 51, 52, 98, 99, 100\}$, $p = 100$, $n = 200$, $\mathbf{Q} = \mathbf{Q}(2)$ $(s, e) = (100, 110)$, $\boldsymbol{\mu}_1^{(\mathbf{J})} \sim \boldsymbol{\mu}_{(1)}$, $\alpha = 0.05$.

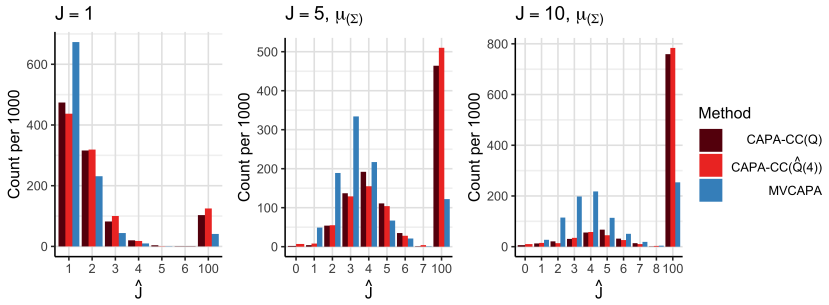

Figure 20: Estimated sizes of $\mathbf{J}$ for $\mathbf{J} = \{1\}$ (left) $\mathbf{J} = \{1, \ldots, 5\}$ (middle) and $\mathbf{J} = \{1, \ldots, 10\}$ when $p = 100$. Other parameters: $n = 200$, $\mathbf{Q} = \mathbf{Q}(2, 0.9)$, $s = 10$, $e = 20$, $\vartheta = 3$, $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$, $\alpha = 0.005$.

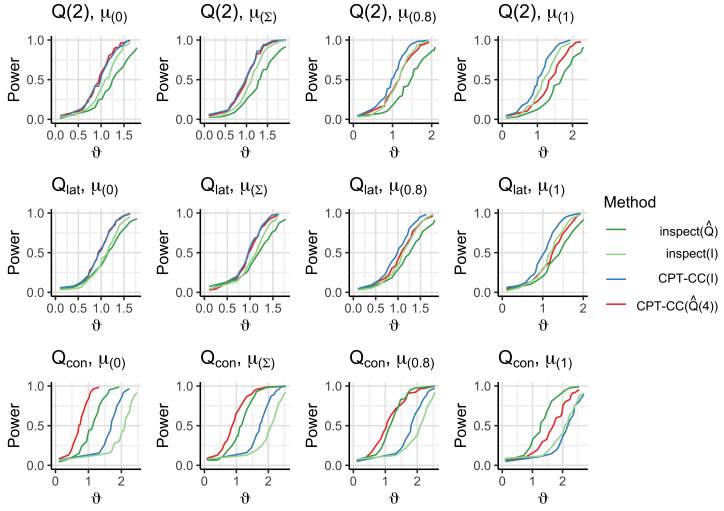

Figure 21: Power curves for a single known changepoint at $\tau = 170$ when $J = 10$, $p = 100$ and $\rho = 0.5$. Other parameters: $n = 200$, $\rho = 0.5$, $\alpha = 0.05$.

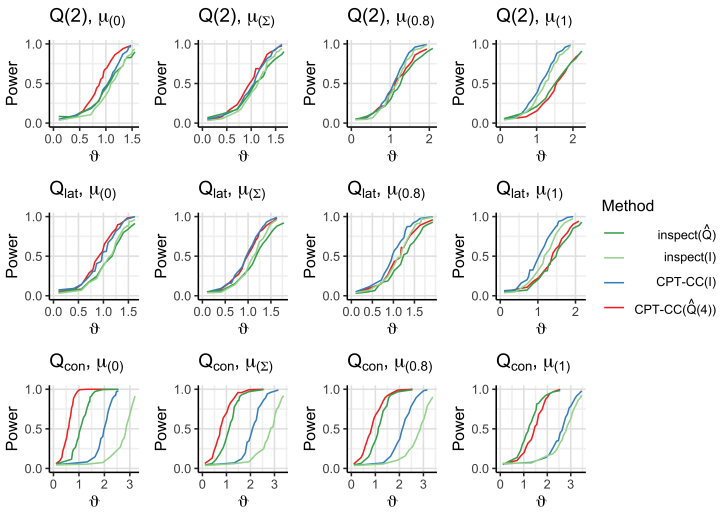

Figure 22: Power curves for a single known changepoint at $\tau = 170$ when $J = 10$, $p = 100$ and $\rho = 0.7$. Other parameters: $n = 200$, $\rho = 0.7$, $\alpha = 0.05$.

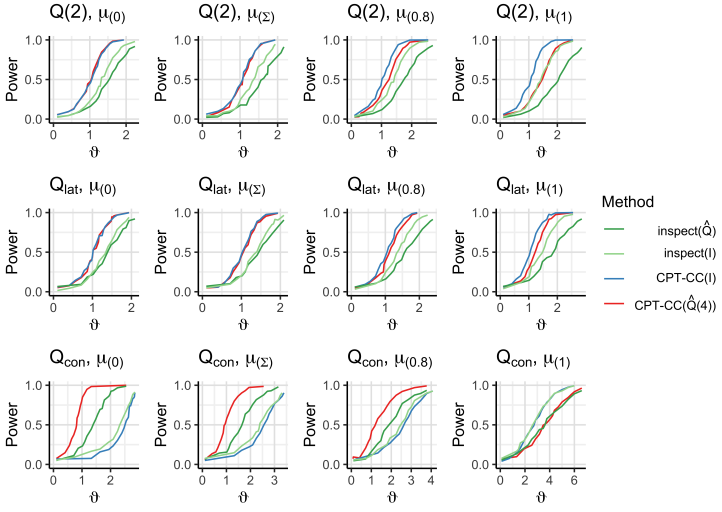

Figure 23: Power curves for a single known changepoint at $\tau = 170$ when $J = 100$, $p = 100$ and $\rho = 0.5$. Other parameters: $n = 200$, $\rho = 0.5$, $\alpha = 0.05$.

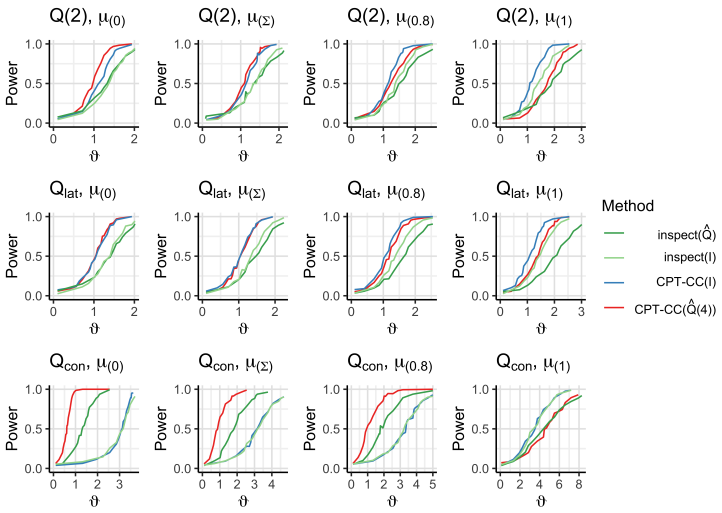

Figure 24: Power curves for a single known changepoint at $\tau = 170$ when $J = 100$, $p = 100$ and $\rho = 0.7$. Other parameters: $n = 200$, $\rho = 0.7$, $\alpha = 0.05$.

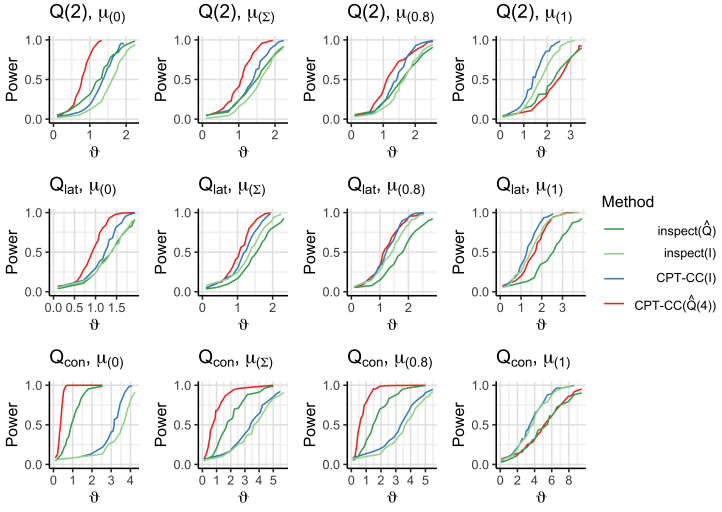

Figure 25: Power curves for a single known changepoint at $\tau = 170$ when $J = 100$, $p = 100$ and $\rho = 0.9$. Other parameters: $n = 200$, $\alpha = 0.05$.

Table 5: RMSE for $p = 100$, $n = 200$, $\tau = 140$, $\vartheta = 2$ and $\boldsymbol{\mu}_{(0)}$ changes. The smallest value is given in bold. 1000 random samples were used for each RMSE estimate.

$\mathbf{Q}$	ρ	J	CPT-CC($\hat{\mathbf{Q}}(\mathbf{W}(4))$)	inspect($\hat{\mathbf{Q}}$)	CPT-CC($\mathbf{I}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	1	**1.3**	3.4	1.4	1.4
$\mathbf{Q}(2)$	0.5	10	1.9	4.4	1.9	**1.7**
$\mathbf{Q}(2)$	0.5	100	**2.3**	9.5	2.6	5.4
$\mathbf{Q}(2)$	0.7	1	**1.2**	3.1	1.4	1.3
$\mathbf{Q}(2)$	0.7	10	**1.7**	3.7	2.2	1.8
$\mathbf{Q}(2)$	0.7	100	**1.8**	7.4	3.0	5.7
$\mathbf{Q}(2)$	0.9	1	**0.7**	2.1	1.5	1.3
$\mathbf{Q}(2)$	0.9	10	**1.0**	2.3	3.6	1.7
$\mathbf{Q}(2)$	0.9	100	**1.1**	5.5	4.9	6.8
$\mathbf{Q}_{\text{lat}}$	0.5	1	**1.3**	3.3	1.4	1.4
$\mathbf{Q}_{\text{lat}}$	0.5	10	2.1	4.4	2.3	**1.6**
$\mathbf{Q}_{\text{lat}}$	0.5	100	2.7	8.6	**2.7**	4.6
$\mathbf{Q}_{\text{lat}}$	0.7	1	1.4	2.8	1.4	**1.3**
$\mathbf{Q}_{\text{lat}}$	0.7	10	**1.7**	4.0	2.0	1.9
$\mathbf{Q}_{\text{lat}}$	0.7	100	**2.1**	8.8	2.5	5.7
$\mathbf{Q}_{\text{lat}}$	0.9	1	**1.0**	2.2	1.4	1.5
$\mathbf{Q}_{\text{lat}}$	0.9	10	**1.4**	3.3	2.4	1.8
$\mathbf{Q}_{\text{lat}}$	0.9	100	**1.8**	6.4	4.5	6.9
$\mathbf{Q}_{\text{con}}$	0.5	1	**0.8**	1.8	13.3	2.4
$\mathbf{Q}_{\text{con}}$	0.5	10	**1.1**	2.0	21.2	5.1
$\mathbf{Q}_{\text{con}}$	0.5	100	**1.1**	7.7	115.9	23.4
$\mathbf{Q}_{\text{con}}$	0.7	1	**0.5**	1.1	17.1	6.4
$\mathbf{Q}_{\text{con}}$	0.7	10	**0.6**	1.7	32.0	9.8
$\mathbf{Q}_{\text{con}}$	0.7	100	**0.5**	5.5	126.1	34.9
$\mathbf{Q}_{\text{con}}$	0.9	1	**0.1**	0.4	20.5	11.7
$\mathbf{Q}_{\text{con}}$	0.9	10	**0.1**	3.7	53.9	17.7
$\mathbf{Q}_{\text{con}}$	0.9	100	**0.1**	4.0	131.2	34.5

Table 6: RMSE for $p = 100$, $n = 200$, $\tau = 140$, $\vartheta = 2$ and $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$ changes. The smallest value is given in bold. 1000 random samples were used for each RMSE estimate.

$\mathbf{Q}$	ρ	J	CPT-CC($\hat{\mathbf{Q}}(\mathbf{W}(4))$)	inspect($\hat{\mathbf{Q}}$)	CPT-CC($\mathbf{I}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	1	**1.3**	3.4	1.4	1.4
$\mathbf{Q}(2)$	0.5	10	2.3	4.8	2.4	**1.9**
$\mathbf{Q}(2)$	0.5	100	**2.4**	9.7	2.7	5.0
$\mathbf{Q}(2)$	0.7	1	**1.2**	3.1	1.4	1.3
$\mathbf{Q}(2)$	0.7	10	2.7	5.6	**2.6**	2.6
$\mathbf{Q}(2)$	0.7	100	**2.4**	10.9	3.4	6.0
$\mathbf{Q}(2)$	0.9	1	**0.7**	2.1	1.5	1.3
$\mathbf{Q}(2)$	0.9	10	4.5	5.3	5.2	**3.1**
$\mathbf{Q}(2)$	0.9	100	**3.2**	13.4	9.5	10.5
$\mathbf{Q}_{\text{lat}}$	0.5	1	**1.3**	3.3	1.4	1.4
$\mathbf{Q}_{\text{lat}}$	0.5	10	2.4	6.1	2.3	**1.8**
$\mathbf{Q}_{\text{lat}}$	0.5	100	**2.5**	9.9	2.6	5.1
$\mathbf{Q}_{\text{lat}}$	0.7	1	1.4	2.8	1.4	**1.3**
$\mathbf{Q}_{\text{lat}}$	0.7	10	2.2	4.4	2.4	**1.9**
$\mathbf{Q}_{\text{lat}}$	0.7	100	**2.5**	10.1	3.3	5.2
$\mathbf{Q}_{\text{lat}}$	0.9	1	**1.0**	2.2	1.4	1.5
$\mathbf{Q}_{\text{lat}}$	0.9	10	3.0	4.0	4.5	**2.3**
$\mathbf{Q}_{\text{lat}}$	0.9	100	**3.1**	12.2	8.3	9.1
$\mathbf{Q}_{\text{con}}$	0.5	1	**0.8**	1.8	13.3	2.4
$\mathbf{Q}_{\text{con}}$	0.5	10	5.0	**3.0**	23.9	7.8
$\mathbf{Q}_{\text{con}}$	0.5	100	**12.3**	22.4	117.2	35.2
$\mathbf{Q}_{\text{con}}$	0.7	1	**0.5**	1.1	17.1	6.4
$\mathbf{Q}_{\text{con}}$	0.7	10	6.6	**3.0**	49.2	13.8
$\mathbf{Q}_{\text{con}}$	0.7	100	**17.3**	29.9	123.3	45.6
$\mathbf{Q}_{\text{con}}$	0.9	1	**0.1**	0.4	20.5	11.7
$\mathbf{Q}_{\text{con}}$	0.9	10	0.8	**0.5**	92.9	21.0
$\mathbf{Q}_{\text{con}}$	0.9	100	**23.6**	42.3	127.2	58.8

Table 7: RMSE for $p = 100$, $n = 200$, $\tau = 140$, $\vartheta = 2$ and $\boldsymbol{\mu}_{(1)}$ changes. The smallest value is given in bold. 1000 random samples were used for each RMSE estimate.

$\mathbf{Q}$	ρ	J	CPT-CC($\hat{\mathbf{Q}}(\mathbf{W}(4))$)	inspect($\hat{\mathbf{Q}}$)	CPT-CC($\mathbf{I}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	1	**1.3**	3.4	1.4	1.4
$\mathbf{Q}(2)$	0.5	10	11.5	8.9	3.9	**3.3**
$\mathbf{Q}(2)$	0.5	100	14.1	18.1	**3.5**	9.0
$\mathbf{Q}(2)$	0.7	1	**1.2**	3.1	1.4	1.3
$\mathbf{Q}(2)$	0.7	10	27.9	14.4	5.3	**5.3**
$\mathbf{Q}(2)$	0.7	100	36.8	21.6	**5.6**	12.4
$\mathbf{Q}(2)$	0.9	1	**0.7**	2.1	1.5	1.3
$\mathbf{Q}(2)$	0.9	10	43.9	17.7	11.9	**8.1**
$\mathbf{Q}(2)$	0.9	100	68.7	29.7	18.7	**18.3**
$\mathbf{Q}_{\text{lat}}$	0.5	1	**1.3**	3.3	1.4	1.4
$\mathbf{Q}_{\text{lat}}$	0.5	10	7.1	6.5	3.2	**2.8**
$\mathbf{Q}_{\text{lat}}$	0.5	100	5.3	20.2	**3.5**	9.5
$\mathbf{Q}_{\text{lat}}$	0.7	1	1.4	2.8	1.4	**1.3**
$\mathbf{Q}_{\text{lat}}$	0.7	10	11.4	8.9	3.8	**3.0**
$\mathbf{Q}_{\text{lat}}$	0.7	100	13.8	24.4	**7.0**	13.0
$\mathbf{Q}_{\text{lat}}$	0.9	1	**1.0**	2.2	1.4	1.5
$\mathbf{Q}_{\text{lat}}$	0.9	10	40.9	8.4	9.1	**4.5**
$\mathbf{Q}_{\text{lat}}$	0.9	100	37.6	28.1	19.1	**17.6**
$\mathbf{Q}_{\text{con}}$	0.5	1	**0.8**	1.8	13.3	2.4
$\mathbf{Q}_{\text{con}}$	0.5	10	42.8	**5.2**	79.8	15.4
$\mathbf{Q}_{\text{con}}$	0.5	100	79.7	**50.7**	116.5	52.6
$\mathbf{Q}_{\text{con}}$	0.7	1	**0.5**	1.1	17.1	6.4
$\mathbf{Q}_{\text{con}}$	0.7	10	23.1	**6.9**	114.2	23.9
$\mathbf{Q}_{\text{con}}$	0.7	100	82.9	**57.8**	118.2	62.7
$\mathbf{Q}_{\text{con}}$	0.9	1	**0.1**	0.4	20.5	11.7
$\mathbf{Q}_{\text{con}}$	0.9	10	**1.4**	4.2	128.4	29.1
$\mathbf{Q}_{\text{con}}$	0.9	100	84.1	**67.5**	124.7	71.2

Table 8: RMSE for $p = 100$, $n = 200$, $\tau = 140$, $\vartheta = 3$ and $\boldsymbol{\mu}_{(0)}$ changes. The smallest value is given in bold. 1000 random samples were used for each RMSE estimate.

$\mathbf{Q}$	ρ	J	CPT-CC($\hat{\mathbf{Q}}(\mathbf{W}(4))$)	CPT-CC($\mathbf{I}$)	inspect($\hat{\mathbf{Q}}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	1	**0.5**	0.6	1.4	0.5
$\mathbf{Q}(2)$	0.7	1	**0.4**	0.6	1.2	0.5
$\mathbf{Q}(2)$	0.9	1	**0.2**	0.5	0.8	0.6
$\mathbf{Q}_{\text{lat}}$	0.5	1	0.5	0.6	1.3	**0.5**
$\mathbf{Q}_{\text{lat}}$	0.7	1	**0.4**	0.5	1.2	0.5
$\mathbf{Q}_{\text{lat}}$	0.9	1	**0.4**	0.6	0.9	0.5
$\mathbf{Q}_{\text{con}}$	0.5	1	**0.3**	6.3	0.8	0.8
$\mathbf{Q}_{\text{con}}$	0.7	1	**0.1**	7.2	0.4	2.1
$\mathbf{Q}_{\text{con}}$	0.9	1	**0.0**	10.2	0.1	2.8
$\mathbf{Q}(2)$	0.5	10	0.6	0.7	1.4	**0.6**
$\mathbf{Q}(2)$	0.7	10	**0.6**	0.8	1.4	0.6
$\mathbf{Q}(2)$	0.9	10	**0.4**	0.7	1.0	0.6
$\mathbf{Q}_{\text{lat}}$	0.5	10	0.7	0.7	1.5	**0.6**
$\mathbf{Q}_{\text{lat}}$	0.7	10	**0.6**	0.8	1.4	0.6
$\mathbf{Q}_{\text{lat}}$	0.9	10	**0.5**	0.8	1.0	0.6
$\mathbf{Q}_{\text{con}}$	0.5	10	**0.3**	12.3	0.9	1.2
$\mathbf{Q}_{\text{con}}$	0.7	10	**0.2**	15.5	0.5	2.0
$\mathbf{Q}_{\text{con}}$	0.9	10	**0.0**	16.4	0.2	6.2
$\mathbf{Q}(2)$	0.5	100	**0.7**	0.7	2.9	1.0
$\mathbf{Q}(2)$	0.7	100	**0.5**	0.7	2.2	1.1
$\mathbf{Q}(2)$	0.9	100	**0.3**	1.1	1.5	1.4
$\mathbf{Q}_{\text{lat}}$	0.5	100	**0.7**	0.7	2.8	1.1
$\mathbf{Q}_{\text{lat}}$	0.7	100	**0.7**	0.8	2.2	1.2
$\mathbf{Q}_{\text{lat}}$	0.9	100	**0.5**	0.8	2.0	1.4
$\mathbf{Q}_{\text{con}}$	0.5	100	**0.4**	40.2	1.3	8.1
$\mathbf{Q}_{\text{con}}$	0.7	100	**0.1**	87.0	0.8	12.4
$\mathbf{Q}_{\text{con}}$	0.9	100	**0.0**	117.2	0.3	18.6

Table 9: RMSE for $p = 100$, $n = 200$, $\tau = 140$, $\vartheta = 3$ and $\boldsymbol{\mu}_{(\boldsymbol{\Sigma})}$ changes. The smallest value is given in bold. 1000 random samples were used for each RMSE estimate.

$\mathbf{Q}$	ρ	J	CPT-CC($\hat{\mathbf{Q}}(\mathbf{W}(4))$)	CPT-CC($\mathbf{I}$)	inspect($\hat{\mathbf{Q}}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	1	**0.5**	0.6	1.4	0.5
$\mathbf{Q}(2)$	0.7	1	**0.4**	0.6	1.2	0.5
$\mathbf{Q}(2)$	0.9	1	**0.2**	0.5	0.8	0.6
$\mathbf{Q}_{\text{lat}}$	0.5	1	0.5	0.6	1.3	**0.5**
$\mathbf{Q}_{\text{lat}}$	0.7	1	**0.4**	0.5	1.2	0.5
$\mathbf{Q}_{\text{lat}}$	0.9	1	**0.4**	0.6	0.9	0.5
$\mathbf{Q}_{\text{con}}$	0.5	1	**0.3**	6.3	0.8	0.8
$\mathbf{Q}_{\text{con}}$	0.7	1	**0.1**	7.2	0.4	2.1
$\mathbf{Q}_{\text{con}}$	0.9	1	**0.0**	10.2	0.1	2.8
$\mathbf{Q}(2)$	0.5	10	0.8	0.7	1.7	**0.6**
$\mathbf{Q}(2)$	0.7	10	**0.8**	1.0	1.8	0.8
$\mathbf{Q}(2)$	0.9	10	**0.8**	1.4	1.7	1.4
$\mathbf{Q}_{\text{lat}}$	0.5	10	0.6	0.7	1.7	**0.6**
$\mathbf{Q}_{\text{lat}}$	0.7	10	0.7	0.8	2.2	**0.7**
$\mathbf{Q}_{\text{lat}}$	0.9	10	**0.8**	1.1	1.4	0.8
$\mathbf{Q}_{\text{con}}$	0.5	10	**0.7**	13.4	0.9	2.2
$\mathbf{Q}_{\text{con}}$	0.7	10	0.7	15.2	**0.5**	4.1
$\mathbf{Q}_{\text{con}}$	0.9	10	0.2	27.3	**0.2**	8.4
$\mathbf{Q}(2)$	0.5	100	**0.7**	0.8	3.1	1.1
$\mathbf{Q}(2)$	0.7	100	**0.8**	1.0	3.2	1.5
$\mathbf{Q}(2)$	0.9	100	**0.7**	1.7	3.8	2.1
$\mathbf{Q}_{\text{lat}}$	0.5	100	**0.8**	0.8	3.0	1.1
$\mathbf{Q}_{\text{lat}}$	0.7	100	0.9	**0.8**	3.6	1.2
$\mathbf{Q}_{\text{lat}}$	0.9	100	**1.0**	3.0	3.7	1.9
$\mathbf{Q}_{\text{con}}$	0.5	100	**2.6**	65.1	5.9	16.2
$\mathbf{Q}_{\text{con}}$	0.7	100	**4.4**	99.5	10.8	27.4
$\mathbf{Q}_{\text{con}}$	0.9	100	**8.4**	113.9	19.0	38.3

Table 10: RMSE for $p = 100$, $n = 200$, $\tau = 140$, $\vartheta = 3$ and $\boldsymbol{\mu}_{(1)}$ changes. The smallest value is given in bold. 1000 random samples were used for each RMSE estimate.

$\mathbf{Q}$	ρ	J	CPT-CC($\hat{\mathbf{Q}}(\mathbf{W}(4))$)	CPT-CC($\mathbf{I}$)	inspect($\hat{\mathbf{Q}}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	1	**0.5**	0.6	1.4	0.5
$\mathbf{Q}(2)$	0.7	1	**0.4**	0.6	1.2	0.5
$\mathbf{Q}(2)$	0.9	1	**0.2**	0.5	0.8	0.6
$\mathbf{Q}_{\text{lat}}$	0.5	1	0.5	0.6	1.3	**0.5**
$\mathbf{Q}_{\text{lat}}$	0.7	1	**0.4**	0.5	1.2	0.5
$\mathbf{Q}_{\text{lat}}$	0.9	1	**0.4**	0.6	0.9	0.5
$\mathbf{Q}_{\text{con}}$	0.5	1	**0.3**	6.3	0.8	0.8
$\mathbf{Q}_{\text{con}}$	0.7	1	**0.1**	7.2	0.4	2.1
$\mathbf{Q}_{\text{con}}$	0.9	1	**0.0**	10.2	0.1	2.8
$\mathbf{Q}(2)$	0.5	10	1.6	1.2	3.2	**1.2**
$\mathbf{Q}(2)$	0.7	10	2.5	1.6	5.4	**1.5**
$\mathbf{Q}(2)$	0.9	10	5.7	3.0	6.5	**2.7**
$\mathbf{Q}_{\text{lat}}$	0.5	10	1.4	1.1	2.3	**0.8**
$\mathbf{Q}_{\text{lat}}$	0.7	10	1.7	1.2	2.4	**1.0**
$\mathbf{Q}_{\text{lat}}$	0.9	10	3.2	1.7	2.5	**1.4**
$\mathbf{Q}_{\text{con}}$	0.5	10	3.7	17.6	**0.9**	4.2
$\mathbf{Q}_{\text{con}}$	0.7	10	1.7	39.9	**0.6**	8.2
$\mathbf{Q}_{\text{con}}$	0.9	10	0.3	81.9	**0.2**	11.3
$\mathbf{Q}(2)$	0.5	100	1.8	**1.3**	8.7	2.7
$\mathbf{Q}(2)$	0.7	100	3.4	**2.0**	12.1	3.5
$\mathbf{Q}(2)$	0.9	100	30.2	**4.0**	18.5	6.8
$\mathbf{Q}_{\text{lat}}$	0.5	100	1.4	**1.2**	8.3	2.6
$\mathbf{Q}_{\text{lat}}$	0.7	100	2.3	**1.8**	12.6	3.0
$\mathbf{Q}_{\text{lat}}$	0.9	100	6.7	**5.5**	18.5	5.6
$\mathbf{Q}_{\text{con}}$	0.5	100	71.0	88.5	36.6	**36.2**
$\mathbf{Q}_{\text{con}}$	0.7	100	80.8	99.8	**42.5**	46.7
$\mathbf{Q}_{\text{con}}$	0.9	100	83.8	111.7	**49.7**	55.2

Table 11: ARI of classifying normal and anomalous observations when $p = 100$, $n = 1000$, $(\vartheta_k)_{k=1}^3 = (1, 2, 3)$, $\{(s_k, e_k)\}_{k=1}^3 = \{(300, 330), (600, 620,), (900, 910)\}$ and $\mathbf{J}_1 = \{1\}$, $\mathbf{J}_2 = \{1, \ldots, 10\}$, $\mathbf{J}_3 = \{1, \ldots, 10, 46, \ldots, 55, 91, \ldots, 100\}$, based on 100 repetitions. Point anomalies are placed at 10 fixed locations, each randomly affecting a single variable with size sampled from $N(0, 4 \log p)$. The largest value for each data setting is given in bold.

$\mathbf{Q}$	ρ	$\boldsymbol{\mu}_{(\cdot)}$	Pt. anoms	CAPA-CC($\hat{\mathbf{Q}}(4)$)	MVCAPA	inspect($\hat{\mathbf{Q}}$)	inspect($\mathbf{I}$)
$\mathbf{Q}(2)$	0.5	0	–	**0.27**	0.18	0.04	0.00
$\mathbf{Q}(2)$	0.5	0	✓	**0.39**	0.37	0.01	−0.02
$\mathbf{Q}(2)$	0.9	0	–	**0.60**	0.03	0.16	0.00
$\mathbf{Q}(2)$	0.9	0	✓	**0.68**	0.26	0.02	−0.02
$\mathbf{Q}(2)$	0.5	Σ	–	**0.23**	0.20	0.05	0.00
$\mathbf{Q}(2)$	0.5	Σ	✓	**0.40**	0.37	0.01	−0.02
$\mathbf{Q}(2)$	0.9	Σ	–	**0.53**	0.05	0.13	−0.00
$\mathbf{Q}(2)$	0.9	Σ	✓	**0.61**	0.26	0.03	−0.02
$\mathbf{Q}(2)$	0.5	0.8	–	**0.23**	0.17	0.05	0.00
$\mathbf{Q}(2)$	0.5	0.8	✓	**0.39**	0.35	0.01	−0.02
$\mathbf{Q}(2)$	0.9	0.8	–	**0.50**	0.07	0.13	0.00
$\mathbf{Q}(2)$	0.9	0.8	✓	**0.62**	0.27	0.02	−0.02
$\mathbf{Q}_{\text{lat}}$	0.5	0	–	**0.21**	0.12	0.06	0.00
$\mathbf{Q}_{\text{lat}}$	0.5	0	✓	**0.31**	0.23	0.04	0.06
$\mathbf{Q}_{\text{lat}}$	0.9	0	–	**0.34**	0.08	0.12	0.00
$\mathbf{Q}_{\text{lat}}$	0.9	0	✓	**0.40**	0.20	0.07	0.04
$\mathbf{Q}_{\text{lat}}$	0.5	Σ	–	**0.21**	0.12	0.05	0.00
$\mathbf{Q}_{\text{lat}}$	0.5	Σ	✓	**0.29**	0.25	0.08	0.07
$\mathbf{Q}_{\text{lat}}$	0.9	Σ	–	**0.34**	0.09	0.08	0.00
$\mathbf{Q}_{\text{lat}}$	0.9	Σ	✓	**0.33**	0.18	0.14	0.06
$\mathbf{Q}_{\text{lat}}$	0.5	0.8	–	**0.23**	0.14	0.05	0.00
$\mathbf{Q}_{\text{lat}}$	0.5	0.8	✓	**0.27**	0.22	0.09	0.07
$\mathbf{Q}_{\text{lat}}$	0.9	0.8	–	**0.33**	0.07	0.09	0.00
$\mathbf{Q}_{\text{lat}}$	0.9	0.8	✓	**0.38**	0.19	0.11	0.06
$\mathbf{Q}_{\text{con}}$	0.5	0	–	**0.47**	0.00	0.07	−0.00
$\mathbf{Q}_{\text{con}}$	0.5	0	✓	**0.53**	0.16	0.02	0.01
$\mathbf{Q}_{\text{con}}$	0.9	0	–	**0.83**	−0.00	0.28	−0.00
$\mathbf{Q}_{\text{con}}$	0.9	0	✓	**0.83**	0.10	0.08	−0.00
$\mathbf{Q}_{\text{con}}$	0.5	Σ	–	**0.44**	−0.00	0.06	−0.00
$\mathbf{Q}_{\text{con}}$	0.5	Σ	✓	**0.50**	0.11	0.03	0.00
$\mathbf{Q}_{\text{con}}$	0.9	Σ	–	**0.66**	−0.00	0.26	−0.00
$\mathbf{Q}_{\text{con}}$	0.9	Σ	✓	**0.71**	0.09	0.10	0.00
$\mathbf{Q}_{\text{con}}$	0.5	0.8	–	**0.43**	0.00	0.04	−0.00
$\mathbf{Q}_{\text{con}}$	0.5	0.8	✓	**0.52**	0.11	0.02	0.00
$\mathbf{Q}_{\text{con}}$	0.9	0.8	–	**0.68**	−0.00	0.30	0.00
$\mathbf{Q}_{\text{con}}$	0.9	0.8	✓	**0.71**	0.09	0.10	0.00