Journal of Statistical Research *https://doi.org/10.3329/jsr.v58i1.75425*
2024, Vol. 58, No. 1, pp. 197-219 *ISSN 0256 - 422 X*

CHANGE POINT DETECTION VIA GAUSSIAN MIXTURE MODELS

WANGSHU TU

School of Mathematics and Statistics
Carleton University, 1125 Colonel By Dr, Ottawa, Ontario, Canada K1S 5B6

Email: wangshu.tu@carleton.ca

UTKARSH J. DANG

Department of Health Sciences
Carleton University, 1125 Colonel By Dr, Ottawa, Ontario, Canada K1S 5B6

Email: utkarsh.dang@carleton.ca

SANJEENA SUBEDI*

School of Mathematics and Statistics
Carleton University, 1125 Colonel By Dr, Ottawa, Ontario, Canada K1S 5B6

Email: sanjeena.dang@carleton.ca

SUMMARY

Change point detection aims to find abrupt changes in time series data. These changes denote substantial modifications to the process; these can be modeled as a change in the distribution (in location, scale, or trend). Traditional changepoint detection methods often rely on a cost function to assess if a change occurred in a series. Here, change point detection is investigated in a mixture-model-based clustering framework and a novel change point detection algorithm is developed using a finite mixture of regressions with concomitant variables. Through the introduction of a label correction mechanism, the unstructured clustering-based labels are treated as ordered and distinct segment labels. This approach can detect change points in both univariate and multivariate time series, and different kinds of change can be captured using a parsimonious family of models. Performance is illustrated on both simulated and real data.

Keywords and phrases: Finite mixture model, change point detection, multivariate, EM algorithm

* Corresponding author

1 Introduction

Finding change points entails identifying abrupt, large, or rapid changes, that involve a shift in location/trend, scale, or distribution in the data-generating process. Those changes frequently require further investigation because they may indicate a substantial change to the process. These change points separate the time series into distinct, homogeneous segments. Changepoint detection has previously been noted to be a key task for large data analysis (National Research Council, 2013). Change point detection has been proposed and used in multiple applications including medical status monitoring (Robinson et al., 2010; Liu et al., 2017), climatology (Beaulieu and Killick, 2018), quality control (Page, 1954; Lai, 1995), signal and image analysis (Han et al., 2012; Fryzlewicz, 2014), and finance (Taylor and Letham, 2018), etc.

Traditional change point detection methods, at their core, determine if a difference in some metric is larger than some threshold by comparing the model fit of a model with change points (single or multiple) versus no change point (or fewer change points). However, many approaches/implementations exist for change point detection. Wald (1945) developed a sequential probability ratio test, that made use of a likelihood ratio test to determine if the parameters of two density functions differ. Based on the control chart demonstration by Shewhart and Deming (1940), Page (1954) proposed a Cumulative Sum (CUSUM) control chart as a nonparametric approach for identifying the presence of a change point. Direct density ratio methods have been established (Kuncheva, 2011; Kawahara and Sugiyama, 2012; Liu et al., 2013; Alippi et al., 2016), which replace the density function with nonparametric dissimilarity measures to simplify the estimation in likelihood ratio and expand the approach to additional scenarios. Other approaches include product partition models (Loschi and Cruz, 2005; Loschi et al., 2010) and the Bayesian change point detection approach (Prescott Adams and MacKay, 2007; Lau and Yamamoto, 2010; Malladi et al., 2013; Roberts and Zhao, 2022). Limitations of some of these can include reliance on a greedy algorithm, restrictions to a univariate series, or a strong reliance on cost and penalty functions, and repeated testing.

By treating each distinct segment as a cluster, the change point detection problem can also be reframed as a clustering problem. As such, the number of segments/clusters can be unknown, and the goal of the analysis is to identify segments such that observations are homogeneous within segments and heterogeneous between segments. Some efforts have been made towards clustering of time series data with change points (Samé et al., 2011; Zakaria et al., 2012; Tran, 2019), while other approaches have been based on k-means (Li et al., 2020; Dawn et al., 2021). Although some of these methods utilize existing clustering techniques, they can also rely on repeated testing to find multiple change points. Other efforts including those that used mixture model-based clustering and change point detection include Joseph and Wolfson (1993) and Joseph et al. (1996), Zhu and Melnykov (2022) and Sarkar and Zhu (2022).

Due to its modelling versatility, the finite mixture model (MacLahlan and Peel, 2000) is the most popular framework for investigating heterogeneity in data (McNicholas and Murphy, 2008; Lin, 2009; Subedi et al., 2020; Tu and Subedi, 2022a,b; Dang et al., 2023). A finite mixture model assumes that the data can be represented as a mixture of G components where each component follows a unique distribution. By treating each segment of consecutive measurements as a distinct cluster, the change point detection problem can be modelled as a clustering problem. Thus, each

subgroup or component can be seen as a segment of consecutive measurements, and a change point can be defined when the distribution changes. It can also be extended conveniently to a multivariate form and integrated into a mixture of regressions framework (Wedel, 2002).

Herein, we propose a novel change point detection algorithm using a Gaussian mixture model which can detect change points in both univariate and multivariate scenarios. In model-based clustering, observations within a cluster are typically independent and unordered. By introducing a label correction algorithm, our algorithm can ensure that the cluster label corresponds to distinct segments, preserving the sequential nature of time series and facilitating the detection of multiple change points. A family of parsimonious models is developed, which allows for change point detection on several common scenarios (change in mean, variance and slope), and the best-fitted model among the fitted models can be chosen using a model selection criteria. Our approach is also relatively robust to outliers. The manuscript is structured as follows: Section 2 describes the main algorithm and Section 3 introduces the family of models and model selection criteria. Simulation and real data studies are presented in Sections 4 and 5. Section 6 concludes with a discussion.

2 Methodology

Let $\mathbf{t}$ denote time, and $\mathbf{X}$ denote a $n \times p$ matrix consisting of measurements of p variables at n different time points. To account for time via the finite mixture model, and capture changes in trend, a finite mixture of regressions with concomitant variables (FMRC; Wedel, 2002) framework is adopted where time $\mathbf{t}$ is treated as a concomitant variable. The density function of a G-component FMRC can be written as

$$f(\mathbf{x}_i|\boldsymbol{\Theta}) = \sum_{g=1}^{G} \pi_{ig} f_g(\mathbf{x}_i|\boldsymbol{\beta}_{0g} + \mathbf{t}_i\boldsymbol{\beta}_{1g}, \boldsymbol{\Sigma}_g),$$

where $\pi_{ig} = \exp(\alpha_{0g} + \alpha_{1g}\mathbf{t}_i)/\sum_{g=1}^{G}\exp(\alpha_{0g} + \alpha_{1g}\mathbf{t}_i)$, the mixing weight, follows a multinomial logit model with the first component as the baseline. Here, $f_g(\mathbf{x}_i|\boldsymbol{\beta}_{0g} + \mathbf{t}_i\boldsymbol{\beta}_{1g}, \boldsymbol{\Sigma}_g)$ is a Gaussian density function of the g^{th} component, and $\boldsymbol{\beta}_{0g}$ and $\boldsymbol{\beta}_{1g}$ are p-dimensional vectors of intercepts and regression coefficients, $\boldsymbol{\Sigma}_g$ is a $p \times p$ covariance matrix, and $\boldsymbol{\Theta} = (\alpha_{01}, \ldots, \alpha_{0G}, \alpha_{11}, \ldots, \alpha_{1G}, \boldsymbol{\beta}_{01}, \ldots, \boldsymbol{\beta}_{0G}, \boldsymbol{\beta}_{11}, \ldots, \boldsymbol{\beta}_{1G}, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_G)$ represents the model parameters.

The unobserved group membership of each observation is treated as missing data. We define the membership indicator variable $\boldsymbol{Z}_i$ as

$$Z_{ig} = \begin{cases} 1 & \text{observation } i \in g^{th} \text{ component,} \\ 0 & \text{otherwise.} \end{cases}$$

The complete-data likelihood is as follows:

$$L(\boldsymbol{\Theta}) = \prod_{i=1}^{n}\prod_{g=1}^{G}\left\{\pi_{ig} f_g(\mathbf{x}_i|\boldsymbol{\beta}_{0g} + \mathbf{t}_i\boldsymbol{\beta}_{1g}, \boldsymbol{\Sigma}_g)\right\}^{z_{ig}}.$$

We define $\mathbf{T}$ as the design matrix such that the first column of $\mathbf{T}$ is a n dimensional vector of 1s and the second column of $\mathbf{T}$ is time $\mathbf{t}$ with $\mathbf{T}_i$ being the i^{th} row of $\mathbf{T}$ and we define $\boldsymbol{\zeta}_g$ as a $2 \times p$ dimensional matrix where the first row of $\boldsymbol{\zeta}_g$ is $\boldsymbol{\beta}_{0g}$ and the second row of $\boldsymbol{\zeta}_g$ is $\boldsymbol{\beta}_{1g}$. Then, the complete-data likelihood can be written as

$$L(\boldsymbol{\Theta}) = \prod_{i=1}^{n} \prod_{g=1}^{G} \left\{ \pi_{ig} f_g(\mathbf{x}_i | \mathbf{T}_i \hat{\boldsymbol{\zeta}}_g, \boldsymbol{\Sigma}_g) \right\}^{z_{ig}}.$$

2.1 The EM algorithm

Parameter estimation can be done via an expectation-maximization (EM) algorithm (Dempster et al., 1977). An EM algorithm comprises two steps: an E-step where the expected value of the complete-data log-likelihood is computed and an M-step where the expected value of the complete-data log-likelihood is maximized to obtain the maximum likelihood estimate of the model parameters.

The complete data log-likelihood can be written as

$$\begin{aligned} l(\boldsymbol{\Theta}) = \log L(\boldsymbol{\Theta}) &= \log \prod_{i=1}^{n} \prod_{g=1}^{G} \left\{ \pi_{ig} f_g(\mathbf{x}_i | \mathbf{T}_i \hat{\boldsymbol{\zeta}}_g, \boldsymbol{\Sigma}_g) \right\}^{z_{ig}} \\ &= \sum_{i=1}^{n} \sum_{g=1}^{n} z_{ig} \log \pi_{ig} + \sum_{i=1}^{n} \sum_{g=1}^{n} z_{ig} \log f_g(\mathbf{x}_i | \mathbf{T}_i \hat{\boldsymbol{\zeta}}_g, \boldsymbol{\Sigma}_g). \end{aligned}$$

The expected value of the complete-data log-likelihood requires the expected value of the missing data z_{ig} conditional on $\mathbf{x}_i$. The expected value of the z_{ig} conditional on $\mathbf{x}_i$ for a mixture of Gaussian distributions is

$$\mathbb{E}(z_{ig} \mid \mathbf{x}_i) = \hat{z}_{ig} = \frac{\pi_g f_g(\mathbf{x}_i | \mathbf{T}_i \hat{\boldsymbol{\zeta}}_g, \boldsymbol{\Sigma}_g)}{\sum_{h=1}^{G} \pi_h f_h(\mathbf{x}_i | \mathbf{T}_i \hat{\boldsymbol{\zeta}}_h, \boldsymbol{\Sigma}_h)}. \tag{2.1}$$

In the M-step, using the expected value of the complete-data log-likelihood, we obtain the maximum likelihood estimates of the model parameters $\boldsymbol{\zeta}_g$ and $\boldsymbol{\Sigma}_g$:

$$\begin{aligned} \hat{\boldsymbol{\zeta}}_g &= (\sum_{i=1}^{n} \hat{z}_{ig} \mathbf{T}_i^T \mathbf{T}_i)^{-1} \sum_{i=1}^{n} \hat{z}_{ig} \mathbf{T}_i^T \mathbf{x}_i, \\ \hat{\boldsymbol{\Sigma}}_g &= \frac{\sum_{i=1}^{n} \hat{z}_{ig} (\mathbf{x}_i - \mathbf{T}_i \hat{\boldsymbol{\zeta}}_g)(\mathbf{x}_i - \mathbf{T}_i \hat{\boldsymbol{\zeta}}_g)^T}{\sum_{i=1}^{n} \hat{z}_{ig}}, \\ \text{and thus, } \hat{\boldsymbol{\beta}}_{0g} &= \hat{\boldsymbol{\zeta}}_g[1,] \text{ and } \hat{\boldsymbol{\beta}}_{1g} = \hat{\boldsymbol{\zeta}}_g[2,]. \end{aligned} \tag{2.2}$$

The $\boldsymbol{\alpha}$ in π_{ig} can be estimated by fitting a multinomial regression of $\hat{z}$ with time $\mathbf{t}$. The E and M steps are iterated until convergence. We determine the convergence of the EM algorithm based on Aitken's acceleration (Aitken, 1926).

2.2 Changepoint detection

The estimated $\hat{z}_{ig}$ is a soft classifier, i.e., it is the posterior probability of the i^{th} observation belonging to the g^{th} component. Here, we view each component as a segment in $\mathbf{x}$ such that the observations in one segment are a sequence of consecutive observations generated from the same distribution, i.e., with the same component membership. Then, we can define a new variable $\mathbf{z}^*$ as the membership of the segment and a changepoint is defined as the first time point when the component membership switches.

In traditional mixture model-based clustering, the final cluster assignment is often done using hard partitioning of $\hat{z}_{ig}$ where the i^{th} point is assigned to the component with the largest weight. Directly using the hard classification of $\hat{z}_{ig}$ from the EM algorithm for a segment has several issues. First, the number of segments will not necessarily equal the number of components. For instance, a sequence with $\mathbf{z} = (1, \ldots, 2, \ldots, 1, \ldots)$ has 3 segments but only 2 components. Secondly, the cluster assignment of the i^{th} observation is considered independent of the j^{th} observation where $j = 1, \ldots, n$ and $j \neq i$. Thus, the cluster assignment will not always form distinct, contiguous segments. Moreover, if the data is noisy or has potential outliers, there may not be clear separation between the segments. In such a scenario, $\mathbf{z}^*$ could look like $(1, 2, 1, 1, 1, 2, 1, 2, 2, 2)$. To reduce the effects of atypical/noise observations, we first apply a moving average with degree m on $\mathbf{X}$ to smooth the data resulting in a $\mathbf{x}^{new}$ such that

$$\mathbf{x}_i^{new} = \frac{\sum_{j=i}^{i+m-1} \mathbf{x}_j}{m}.$$

Note that $\mathbf{x}^{new}$ has a shorter length with time ranging from 1 to $n-m+1$. As long as the degree m is smaller than the number of observations in the last segment, the last $m-1$ observations will always belong to the last segment. After smoothing, we run cluster analysis using a Gaussian mixture model on $\mathbf{x}^{new}$ and obtain $\hat{\mathbf{z}}^*$. Based on the smoothed $\mathbf{x}^{new}$, and cluster labels $\hat{\mathbf{z}}^*$, we define the following terms:

- The vector containing all change points is denoted as $\boldsymbol{c}$ with length G^* such that $\boldsymbol{c}_g$ is the g^{th} changepoint and $\boldsymbol{c}_1 = 1$.
- w: window size.
- $\hat{\mathbf{z}}^*[\boldsymbol{c}_g : \boldsymbol{c}_g + w - 1]$ represents a vector of length w containing labels of w observations at and after the c_g^{th} changepoint.
- l: the label that has the highest proportion inside the c_g^{th} window.
- s: stopping number, $i.e., s = 0, 1$.

The basic idea of the proposed changepoint detection algorithm is to find the position where the membership label switches, i.e., a new segment starts. First, we find the label that has the highest proportion (l) inside the first (or previous) segment, then use l to define the segment. Recall that here we define a change point as the position where a new segment starts. We then compute the number of l for all subsequent rolling windows from left to right of size w. Let l_j be the number of l in j^{th} rolling window. Ideally, we would like to see $l_j = 0$ at some time point, so that the label

l completely disappears, indicating a change has happened. But in reality, a complete change may be too restrictive. Instead, we use the stopping number s, which is the number of l in windows such that the segment changes if $l_j \leq s$. For example, with $w = 5$ and $s = 0$, in a dataset with labels $(1, 2, 1, 1, 1, 2, 1, 2, 2, 2, \ldots)$ the first changepoint will be detected at 8^{th} position (identifying the changepoint after label "1" completely disappears i.e., $(1, 2, 1, 1, 1, 2, 1, \mathbf{2}, 2, 2, \ldots)$) whereas with $w = 5$ and $s = 1$, in the same dataset, the first changepoint will be detected at 6^{th} position (identifying the changepoint when 1 out of 5 observations has label l_j i.e., $(1, 2, 1, 1, 1, \mathbf{2}, 1, 2, 2, 2, \ldots)$. Thus $s > 0$ allows some contamination in the next segment. Depending on the combination of w and s, the position of inferred change points could change. The algorithm to detect changepoints is summarized below:

Algorithm 1. **Algorithm 1**: Algorithm for changepoints detection

$g = 1$ and $c_1 = 1$.
while $g < g_{max}$ **do**
    Compute the number of labels following the last changepoint in the next window: $\hat{\mathbf{z}}^*[c_g : c_g + w - 1]$
    Let l be the label that has the highest proportion inside the c_g^{th} window.
    Compute the number of l for all subsequent rolling windows of size w.
    Let l_j be the number of l in the j^{th} rolling window. Then, we find the first window such that $l_j \leq s$.
    if no window satisfies $l_j \leq s$ or $c_{g+1} - c_g < 3$ **then**
        Break
    end if
    Define the left end of that window to be c_{g+1}.
    $g = g + 1$
end while
Output: $\boldsymbol{c}$
end

Here, g_{max} is the maximum number of segments that can be detected, set to $g_{max} = 30$ in our work.

2.3 Overall algorithm

The overall algorithm, referred to from hereon as change point mixture model (`cpmm`) algorithm, where change point detection is incorporated within the EM algorithm is summarized below:

Algorithm 2. **Algorithm 2**: Algorithm for changepoint mixture model **Input**: $\mathbf{x}$, G, m, w, s

Initialization:

Apply moving average with degree m on $\mathbf{x}$.
Initialize cluster label $\mathbf{z}^*$ based on *mclust* (Scrucca et al., 2016) or *flexmix* (Grün and Leisch, 2008) with concomitant variables with G.
Apply **Algorithm 1** for changepoint detection with input $\mathbf{z}^*$, w and s.
return $\mathbf{c}$.

Main:

Create G^* segments with $\mathbf{c}$ and initialize another FMRC model with G^* components.
Run the EM algorithm outlined in Section 2.1 on the new G^*-component model to obtain the estimates of the model parameters and $\hat{z}_{ig,G^*}$ based on the new G^*-component model and obtain new $\mathbf{z}^*_{G^*}$.
Set G_{new} as the total number of unique labels in the $\mathbf{z}^*_{G^*}$.
while $G_{new} < G^*$ **do**
    Restart the EM algorithm outlined in Section 2.1 with G_{new}.
    Get $\hat{z}_{ig}$ and $\mathbf{z}^*_{G_{new}}$.
    Set $G^* = G_{new}$ and redefine G_{new} as the total number of unique labels in the $\mathbf{z}^*_{G_{new}}$.
end while
Apply **Algorithm 1** for changepoint detection with input $\mathbf{z}^*_{G_{new}}$, w, and s.

Output: $\mathbf{c}_{G_{new}}$ which is the vector of final changepoints.
end

Using the final $\mathbf{c}_{G_{new}}$, we define the final G_{new} segments and then obtain the final estimate of the $\hat{\mathbf{Z}}$ matrix $Z_{ig} = 1$ if an observation belongs to g segment and 0 otherwise. Using $\hat{\mathbf{Z}}$, the estimates of the model parameters $\boldsymbol{\beta}_{0g}$, $\boldsymbol{\beta}_{1g}$, and $\boldsymbol{\Sigma}_g$ can be obtained.

3 Family of models and model selection

Constraints can be imposed on $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$ for parsimony; these are outlined in Table 1 resulting in a family of four models.

The model "trend" is the unconstrained model introduced in Section 2 with details of the parameter estimation provided in Section 2.1. In the model "VV", we impose the constraint that the slope is 0, i.e., the segments are modelled with unique means and covariance matrices. Imposing constraints that the means are equal across segments but the covariance differs results in the model "EV" while imposing constraints that the means across the segments are different but covariance is the same results in the model "VE". Details on the parameter estimation of the three latter models are provided in the Appendix.

Ideally, detection of the segments should not be impacted if observations are ordered $\mathbf{x}_1, \mathbf{x}_2, \ldots,$ $\mathbf{x}_n$ or in reverse order: $x_n, x_{n-1}, \ldots, x_1$. Hence, we also reverse the ordering of the data and fit all

Table 1: Specification of different constraints in the family of models.

Model name	$\boldsymbol{\beta}_{0g}$	$\boldsymbol{\beta}_{1g}$	$\boldsymbol{\Sigma}_g$	Number of parameters
trend	$\boldsymbol{\beta}_{0g} = \boldsymbol{\beta}_{0g}$	$\boldsymbol{\beta}_{1g} = \boldsymbol{\beta}_{1g}$	$\boldsymbol{\Sigma}_g = \boldsymbol{\Sigma}_g$	$G^* \left(\frac{p(p+1)}{2} + 2p\right) + 2(G^* - 1)$
VV	$\boldsymbol{\beta}_{0g} = \boldsymbol{\beta}_{0g}$	$\boldsymbol{\beta}_{1g} = 0$	$\boldsymbol{\Sigma}_g = \boldsymbol{\Sigma}_g$	$G^* \left(\frac{p(p+1)}{2} + p\right) + 2(G^* - 1)$
VE	$\boldsymbol{\beta}_{0g} = \boldsymbol{\beta}_{0g}$	$\boldsymbol{\beta}_{1g} = 0$	$\boldsymbol{\Sigma}_g = \boldsymbol{\Sigma}$	$G^* p + \frac{p(p+1)}{2} + 2(G^* - 1)$
EV	$\boldsymbol{\beta}_{0g} = \boldsymbol{\beta}_0$	$\boldsymbol{\beta}_{1g} = 0$	$\boldsymbol{\Sigma}_g = \boldsymbol{\Sigma}_g$	$G^* \left(\frac{p(p+1)}{2}\right) + p + 2(G^* - 1)$

models in the family to the reversed data as well.

As the number of segments is unknown, we run our algorithm with a range of values for G. Moreover, the optimal model from the family of models for a given dataset is also unknown. We also run these algorithms to allow for some small contamination of the segments to make the algorithm more robust to outliers. Hence, we run all four models for $s = \{0, 1\}$, window size $w = \{5, 10\}$, number of observations in moving average $m = \{1, 5, 10\}$, and $G = \{2, 3\}$ using both original and reverse ordering of the data and the best-fitting model is chosen based on a model-selection criterion *a posteriori*. Note that G is crucial only for initialization of Algorithm 2. The final number of segments is determined by $\mathbf{c}$ and it can be larger than G. The Bayesian information criterion (BIC; Schwarz, 1978) is the most widely used model selection criterion and is considered to be consistent and efficient under certain regularity conditions (Keribin, 2000; Fraley and Raftery, 1998). Mathematically,

$$\text{BIC} = -2L + \psi \log n,$$

where L is the log-likelihood, ψ is the number of free parameters, and n is the total number of time points. The model with the lowest BIC is chosen as the best model.

4 Simulation studies

Here, we demonstrate the performance of *cpmm* in both univariate and multivariate scenarios and provide comparisons with other existing approaches with R implementations. For each simulation scenario, we generate 100 datasets and fit the family of models for a range of initial G using both the original ordering of the data and the reversed ordering of the data. The goal here is to have a high true positive rate (i.e., identifying a true changepoint as a changepoint) and a low false positive rate (i.e., identifying a data point as a changepoint when it is not a true changepoint). Since the changepoints split data into segments, we utilize the adjusted Rand index (ARI; Hubert and Arabie, 1985) to measure the agreement between the true (when known) and estimated segments. An ARI value of 1 stands for perfect agreement and has an expected value of 0 under random classification.

4.1 Univariate simulations

For the univariate case, we conduct six sets of simulations (Figure 1). The performance of our proposed algorithm is compared to the following four methods available in R: *mcp* (Lindeløv, 2020), *change point* (Killick and Eckley, 2014; Killick et al., 2022), *EnvCpt* (Killick et al., 2021) and *stepR* (Pein et al., 2022). Figure 1 shows examples of one example dataset generated under each scenario:

- Simulation Scenario 1 (4 segments): The datasets are generated using a mixture of normal distributions such that different segments have different means but the same variance.
- Simulation Scenario 2 (4 segments): The datasets are generated using a mixture of normal distributions such that different segments have the same mean but different variances.
- Simulation Scenario 3 (3 segments): The datasets are generated using a mixture of regression models with different slopes. The slopes of adjacent segments are simulated with contrasting directionality, i.e., $\boldsymbol{\beta}_{1g}$ from the neighbouring segments have opposite signs.
- Simulation Scenario 4 (3 segments): The datasets are generated using a mixture of regressions with different slopes. However, the slopes are changing gradually and towards the same direction (all $\boldsymbol{\beta}_{1g}$ have the same signs). Scenario 4 is considered a challenging case according to Beaulieu and Killick (2018).
- Simulation Scenario 5 (7 segments): Here, the datasets are generated with noise, similar to Simulation Scenario 2 from Fearnhead and Rigaill (2019). To add noise, we randomly replace 50 points with noise from a t-distribution with 5 degrees of freedom.
- Simulation Scenario 6 (4 segments): The datasets are generated using 4 different skew-normal distributions (i.e., 4 segments) from package: *sn*. The parameters of skew-normal distributions are ξ, ω, and α, where ξ and ω control the location and scale of the distribution and α controls the skewness. All four segments are generated to have slightly different location parameters. The first and last segments are both generated to have left-skewness with the same scale, the second segment is generated to have a larger scale and right-skewness, and the third segment is generated with a smaller scale value but with right-skewness as well.

Note that some competitive approaches have some limitations, which need to be taken into account while fitting these models on the simulated datasets. The package *mcp* provides a change point detection algorithm based on a regression framework which can model various complex structures of the data, however, it requires prespecifying the number of segments and the structure of segments, such as constant, slope or AR process as well as the order of the segments. The other three methods can detect change points automatically: *stepR* uses the step function to model time-series data. *change point* can detect changes in mean and variance, while *EnvCpt* utilizes a similar search algorithm as *change point* but uses the univariate normal distribution to model each segment with and without the AR process introduced.

For each dataset, for *cpmm*, we ran all four models in the family for a range of G using both the original ordering and reverse ordering of the datasets and the best model was chosen using BIC. As

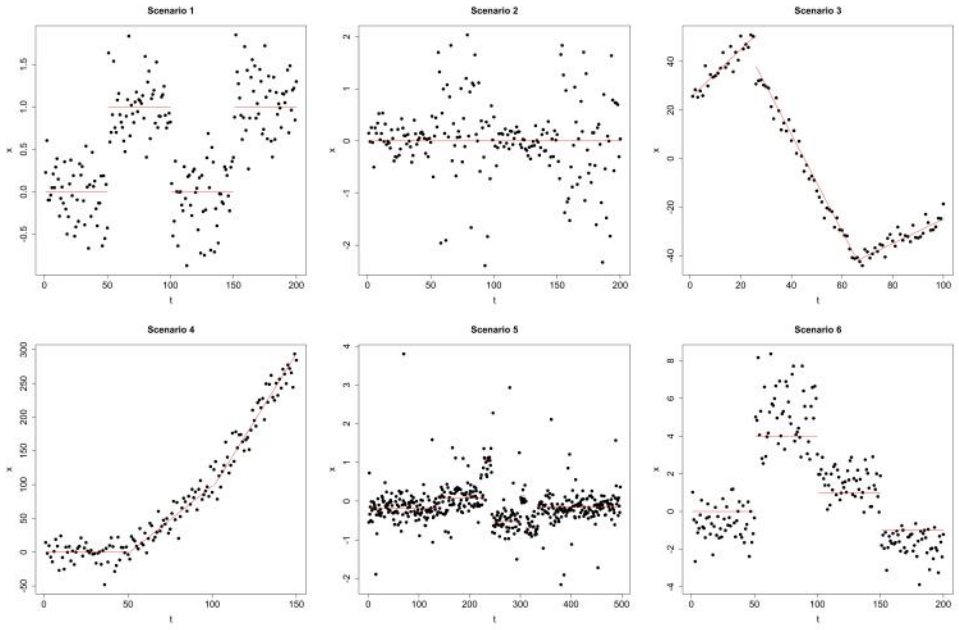


Figure 1: Example datasets from each of the six simulation scenarios for the univariate case. Red lines indicate either the mean (or location parameter in Scenario 6) or the slope.

mcp requires prespecifying the number of segments and the structure of segments, such as constant, slope or AR process and the order of the segments, it was provided for all scenarios. To have a fair comparison, we use function $cpt.meanvar$ with method "PELT" for *changepoint*, and only run mean and trend detection for *EnvCpt*. The default setting is selected for *stepR*.

The results from all five approaches are summarized using density plots (see Figure 2), which show the high-density regions of the change points identified by each approach and using a table (see Table 2), which shows the proportions of changepoints that were identified within $\pm\ k$ (with $k = 1, 3, 5$) observations of the true change point out of the changepoints that were identified.

Although *mcp* required prespecifying the number of segments and the structure of segments (i.e., specifying whether to fit the model with a slope or mean), it did not have the best performance in five out of six scenarios: Scenarios 1, 2, 3, 5, and 6. According to the density plot in Figure 2 and Table 2, we can see that all methods perform well in Simulation Scenario 1. All changepoints detected by *cpmm* are within 5 observations of the true changepoint and *EnvCpt* and *stepR* have the highest ARI. For Simulation Scenario 2, as *stepR* cannot model the changes in variances, it did not perform well while *EnvCpt* had the highest ARI as well as the highest proportion of changepoints detected. Our proposed approach yields competitive performance based on both the proportion of changepoints detected as well as ARI. As *changepoint* and *stepR* don't model slopes, poorer performance is expected in Scenarios 3 and 4. When there is a change is in the slope, *cpmm* outperforms all other methods in Scenario 3. Although *mcp* performed slightly better than *cpmm* in Scenario 4,

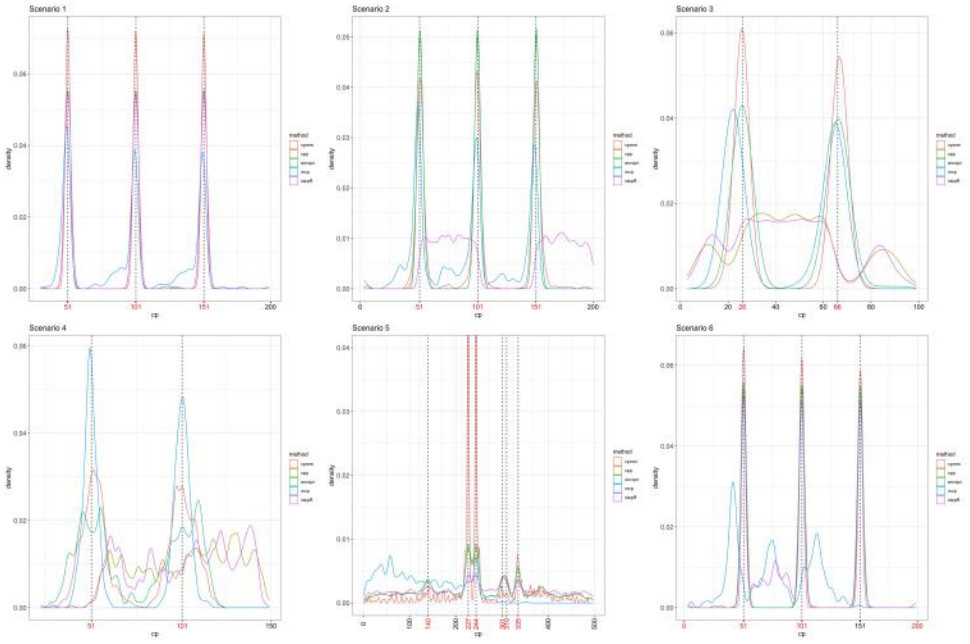


Figure 2: Density plots showing the high-density regions of the selected changepoints for the six univariate simulation scenarios using all 5 approaches (our proposed approach and four competitive approaches). The x-axis indicates the time points and the vertical dashed lines are the true change points.

note that the number of changepoints (presence of three segments) and the structure of the model (i.e., information that all three segments have different slopes) are provided to the *mcp* algorithm, whereas *cpmm* is an unsupervised implementation. Note that *cpmm* also had better performance than *EnvCpt* in this challenging scenario. In scenario 5, *cpmm* has the best performance in selecting the correct number of change points although a lower ARI compared to *changepoint* and *EnvCpt*. In scenario 6 where the dataset is generated from a mixture of skew-normal distributions, when looking at change points detected within ± 1 of the true change points, the *EnvCpt* algorithm outperforms all other approaches whereas when looking at changepoints detected within ± 3 or ± 5 of the true changepoints, *cpmm*, *EnvCpt*, and *changepoint* have similar performance. In terms of picking out the changepoints in the neighbourhood of the true changepoint, *stepR* did not perform as well as the other approaches. `mcp` on the other did not perform well on this dataset. In terms of ARI, *EnvCpt* yielded the highest ARI while a similar ARI was observed for *cpmm* and *changepoint*.

4.2 Multivariate case

Here, we demonstrate the performance of our proposed model on simulated multivariate datasets generated under two different scenarios. Changepoint detection on multivariate data is typically

Table 2: Summary of the proportion of changepoints that were detected within $\pm\, k$ (with $k = 1, 3, 5$) observations of the true changepoints. The average ARI along with standard deviations (sd) from all five approaches are also provided.

Univariate	$\pm\, k$	*cpmm*	*changepoint*	*EnvCpt*	*stepR*	*mcp*
Scenario 1	± 1	0.91	0.98	0.99	0.98	0.63
	± 3	1	0.99	0.99	0.99	0.73
	± 5	1	0.99	0.99	0.99	0.79
ARI	Mean (sd)	0.97 (0.02)	0.96 (0.01)	0.99 (0.01)	0.99 (0.02)	0.88 (0.15)
Scenario 2	± 1	0.48	0.83	0.84	0.07	0.38
	± 3	0.77	0.97	0.98	0.13	0.60
	± 5	0.91	0.98	0.99	0.20	0.70
ARI	Mean (sd)	0.90 (0.07)	0.95 (0.02)	0.98 (0.02)	0.60 (0.06)	0.80 (0.20)
Scenario 3	± 1	0.72	0.06	0.68	0.09	0.38
	± 3	0.93	0.12	0.93	0.15	0.58
	± 5	0.99	0.20	0.98	0.23	0.92
ARI	Mean (sd)	0.93 (0.05)	0.34 (0.04)	0.91 (0.05)	0.35 (0.03)	0.84 (0.09)
Scenario 4	± 1	0.18	0.03	0.11	0.02	0.34
	± 3	0.38	0.08	0.23	0.07	0.66
	± 5	0.56	0.13	0.41	0.13	0.88
ARI	Mean (sd)	0.79 (0.11)	0.49 (0.05)	0.74 (0.11)	0.49 (0.05)	0.89 (0.07)
Scenario 5	± 1	0.45	0.33	0.30	0.16	0.08
	± 3	0.58	0.39	0.41	0.22	0.12
ARI	Mean (sd)	0.55 (0.10)	0.65 (0.14)	0.65 (0.14)	0.45 (0.08)	0.40 (0.16)
Scenario 6	± 1	0.87	0.96	0.98	0.69	0.06
	± 3	0.98	0.99	1	0.72	0.09
	± 5	0.99	0.99	1	0.73	0.14
ARI	Mean (sd)	0.96 (0.06)	0.96 (0.01)	0.99 (0.01)	0.93 (0.05)	0.54 (0.08)

more challenging than changepoint detection on univariate data, and many available R algorithms are restricted to univariate data only.

- Multivariate Scenario 1: We consider the first multivariate scenario as the same as the example in *ecp* (James and Matteson, 2014). In this scenario, the means are the same across segments

but the variances are different; this would be analogous to our "EV" model. The observations are drawn from a 3-dimensional multivariate Gaussian distribution with $\boldsymbol{\beta}_0 = \boldsymbol{\beta}_1 = (0, 0, 0)$, and the following covariance matrix:

$$\boldsymbol{\Sigma}_1 = \begin{pmatrix} 1, 0, 0 \\ 0, 1, 0 \\ 0, 0, 1 \end{pmatrix}, \quad \boldsymbol{\Sigma}_2 = \begin{pmatrix} 1, 0.9, 0.9 \\ 0.9, 1, 0.9 \\ 0.9, 0.9, 1 \end{pmatrix}, \quad \boldsymbol{\Sigma}_3 = \begin{pmatrix} 1, 0, 0 \\ 0, 1, 0 \\ 0, 0, 1 \end{pmatrix}.$$

Assuming that the variables are independent, each univariate variable follows a $N(0, 1)$. Thus, applying univariate changepoint methods will not model the data adequately. Thus, it is crucial to model the data as multivariate data using a multivariate changepoint detection approach.

- Multivariate Scenario 2: In the second multivariate scenario, the observations are drawn from an 8-dimensional multivariate Gaussian distribution with 3 segments. The covariance matrix of the three segments is set to be the same across segments but the intercepts and the slopes for the j^{th} variable (where $j = 1, \ldots, 8$) are generated from $N(0, 10)$ and $N(0, 0.5)$, for all segments respectively.

We compared the performance of our approach with the multivariate changepoint detection approach in *ecp* which utilizes a divergence measurement to determine if two independent variables are identically distributed.

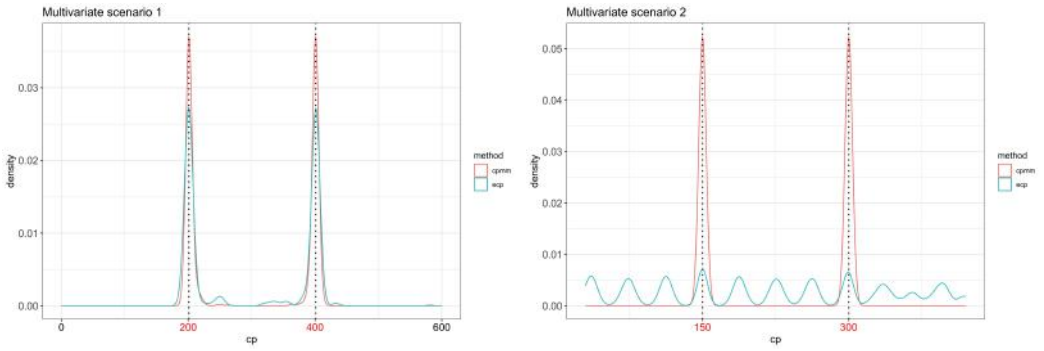


Figure 3: Density plots show the high-density regions of the selected changepoints for the two multivariate simulation scenarios using both *cpmm* and ecp approaches. The x-axis indicates the time points and the vertical dashed lines are the true change points.

As evident in Figure 3 and Table 3, *cpmm* performs very well in both multivariate scenarios. In Scenario 1, where one expects the univariate case to not perform satisfactorily, both *cpmm* and *ecp* show good performance. However *cpmm* is slightly better in inferring changepoints in the neighbourhood of the true changepoints. In Scenario 2, where the change in each dimension is a mixture of different slopes, *ecp* could not always correctly identify the changepoint. *cpmm* on the other hand detects the changepoints well (within ± 1 observations of the true changepoint) and has a high ARI.

Table 3: Summary of the proportion of changepoints that were detected within $\pm\, k$ (with $k = 1, 3, 5$) observations of the true changepoints. The average ARI along with standard deviations (sd) from both *cpmm* and *ecp* approaches are also provided.

Multivariate	$\pm k$	*cpmm*	*ecp*
Scenario 1	± 1	0.68	0.58
	± 3	0.86	0.76
	± 5	0.92	0.85
ARI	Mean (sd)	0.98 (0.03)	0.95 (0.10)
Scenario 2	± 1	1	0.17
	± 3	1	0.18
	± 5	1	0.18
ARI	Mean (sd)	0.99 (0.01)	0.31 (0.02)

5 Real data analysis

In the real data analysis, we applied *cpmm* to two publicly available datasets: `Well-log` dataset and `House Price Index` dataset.

5.1 Well-log dataset

We apply our method to the well-log dataset (Ruanaidh and Fitzgerald, 1996), which is a univariate time series dataset. Models were fitted with $G = \{2, \ldots, 6\}$, window size $w = \{10, 20\}$, $m = \{1, 10, 20\}$, and stop number $s = \{0, 1\}$. This data comprises measurements on rocks from a probe being lowered into a bore-hole. As the probe passes through different rock strata, there is a sudden change in the measurements. The original motivation for collecting this data was to detect the changes in real-time when the rock strata are being drilled so that once the change is detected, appropriate adjustments to the settings of the drill can be made. Several studies have analyzed this dataset (Ruanaidh and Fitzgerald, 1996; Wyse et al., 2011; Fearnhead, 2006; Ruggieri and Antonellis, 2016). Fearnhead and Rigaill (2019) pointed out that there is another version of this dataset which contains several outliers. Visualization of the two versions of the dataset is provided in Figure 4: the plot on the left is the dataset with outliers, and the one on the right is the version where these outliers have been removed.

We applied our methods along with the comparator approaches from packages *changepoint*, *EnvCpt* and *stepR* on both versions of the `Well-log` dataset. Visualization of the changepoints detected in the dataset without outliers is presented in Figure 5 and the dataset with outliers is

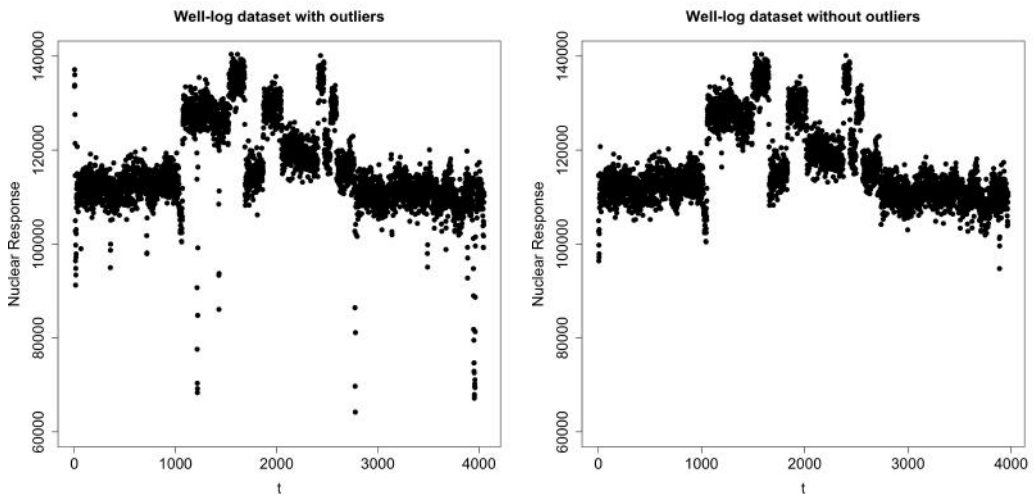


Figure 4: Visualization of the two versions of the `Well-log` dataset.

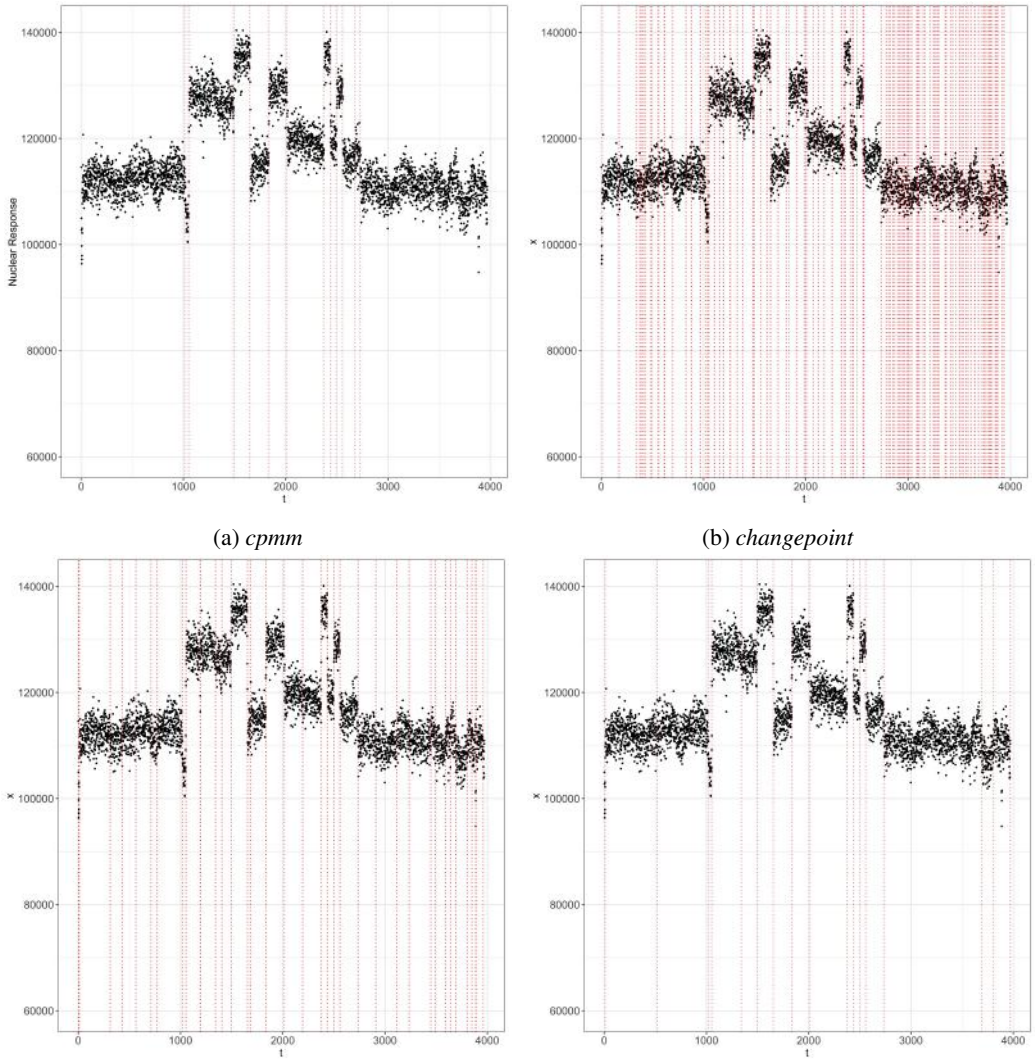
presented in Figure 6.

Based on Figures 5 and 6, we can see that *cpmm* is the most conservative approach. In terms of the number of changepoints detected, visual inspection of Figures 5 and 6 reveals that *changepoint* detected the most number of changepoints followed by *stepR*; in both cases, some observations are identified as changepoints even though the shift in the data looks to be quite small. Although *EnvCpt* and *cpmm* perform similarly, *cpmm* is more robust to outliers as there is a smaller difference in changepoints detected in datasets with and without outliers. Furthermore, the performance of *cpmm* agreed better with the results from Fearnhead and Rigaill (2019).

5.2 `House Price Index` dataset

We consider the changes in the UK House Price Index (HPI) for four countries: England, Northern Ireland, Scotland, and Wales. The UK `House Price Index` (HPI) dataset comprises house sales data from various sources such as the HM Land Registry, Registers of Scotland, and Land and Property Services Northern Ireland. Based on a hedonic regression model, the Office for National Statistics computes an index which produces estimates of the change in house prices for each period and all types of property for each country. This 4-dimensional data runs from April 1968 to September 2022 and is available from `https://landregistry.data.gov.uk/app/ukhpi`. As the data are reported monthly and quarterly depending on different periods, we compute the average index for each quarter and treat it as quarterly data. Hence, the resulting dataset runs from the second quarter in 1968 to the third quarter in 2022 with 218 time points in total. As this is multivariate data when the data from multiple countries are analyzed together, we compare the result of our approach with the *ecp* algorithm.

Figure 7 shows that the segments seem to have gradually increasing slopes, thus making this



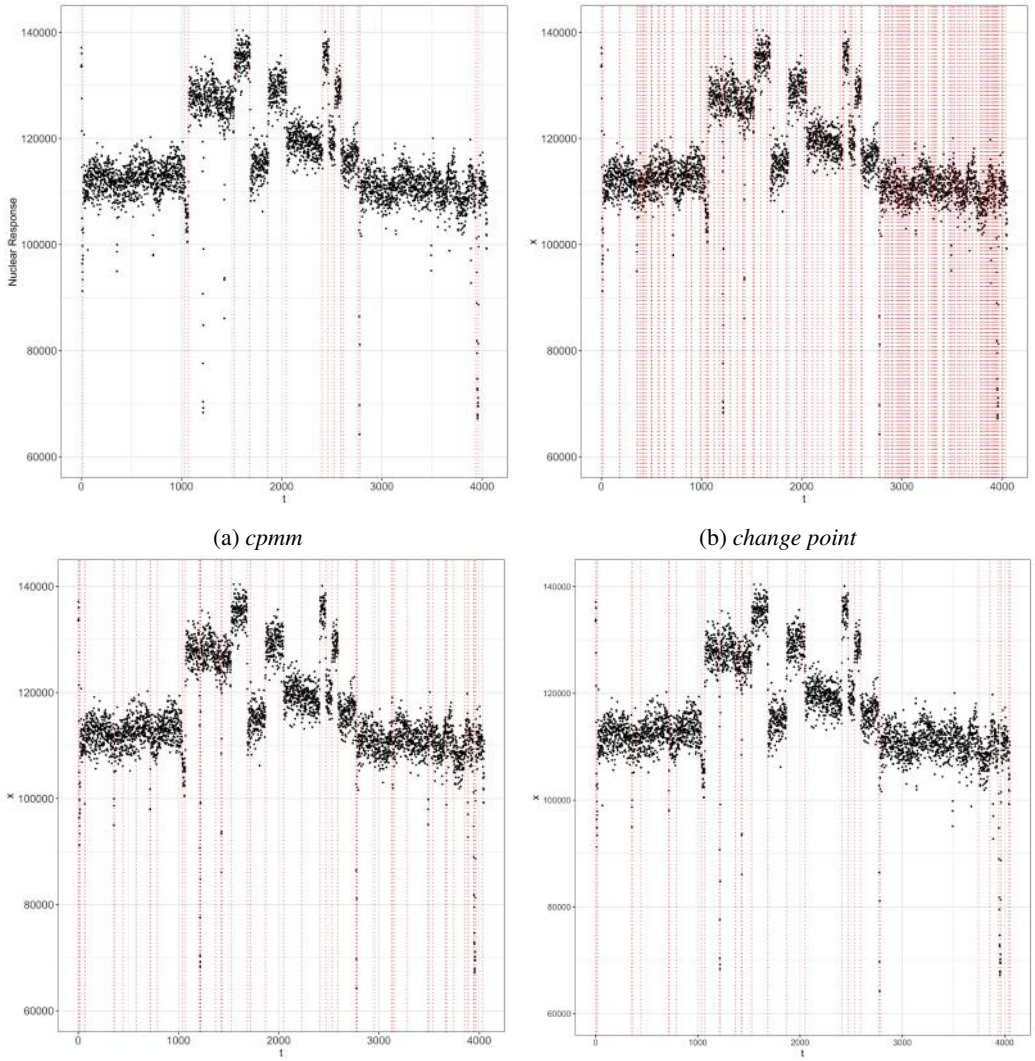
(a) *cpmm* (b) *changepoint*

(c) *EnvCpt* (d) *stepR*

Figure 5: Visualization of the changepoints detected by all four approaches on the `Well-log` dataset without outliers. The red line indicates the changepoints that were detected.

dataset a harder segment detection problem.

Visualization of the changepoints identified by both approaches is shown in Figure 7. Some of the changepoints detected by *cpmm* and ecp are close (within a 1-year difference) to each other: *cpmm* selects 1978_q2, 1988_q1, and 2002_q4 while *ecp* selects 1979_q2, 1988_q2, and 2003_q1. *ecp* can sometimes miss the dramatic change e.g., the decrease in slope around the fourth quarter

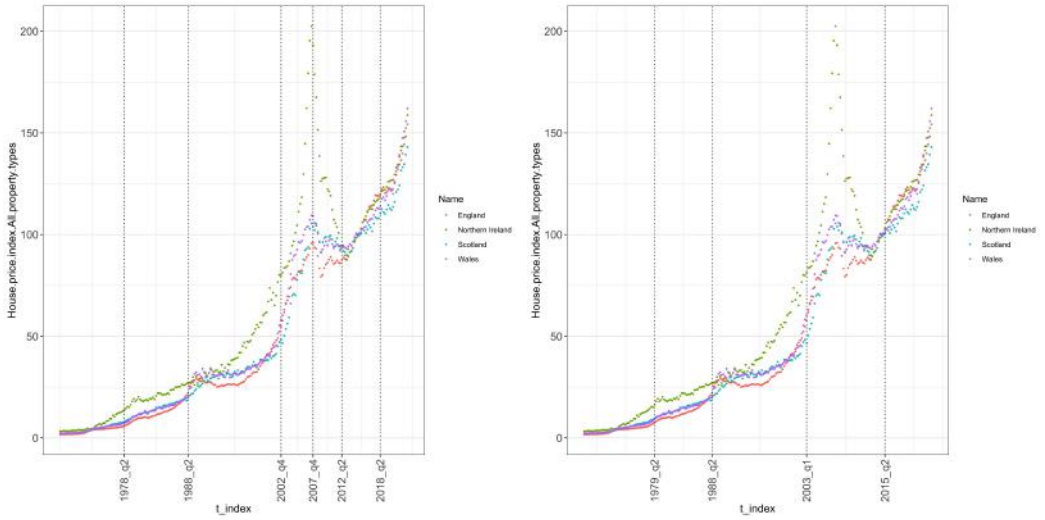


(a) *cpmm* (b) *change point*

(c) *EnvCpt* (d) *stepR*

Figure 6: Visualization of the changepoints detected by all four approaches on the `Well-log` dataset with outliers. The red line indicates the changepoints that were detected.

in 2007 - this was identified by *cpmm*. In addition, *cpmm* further picked another changepoint at 2018_q2 - visual inspection of Figure 7 shows the rate of slope seems to change around 2018_q2 as well.



(a) *cpmm* (b) *ecp*

Figure 7: Visualization of the changepoints detected by both `cpmm` and `ecp` approaches on the `House Price Index` dataset. The y-axis represents the housing index and the x-axis represents the quarter of the year. The dotted line indicates the changepoints that were detected.

6 Conclusion

In this manuscript, we propose and develop a novel changepoint detection algorithm that utilizes a finite mixture model and a label correction method based on a moving window. By considering changepoint detection as a clustering problem, the novel approach can automatically detect the position of changepoints as well as the number of changepoints. Due to the flexibility of the finite mixture model, it can model changes in mean, variance and trend (i.e., slopes) in both univariate and multivariate cases. The novel approach for label correction proposed here allows for clusters to be defined such that the number of segments will equal the number of components and observations in the components are consecutive. Furthermore, the algorithm is also relatively robust to the effects of outliers on change point detection. While the proposed label correction is implemented in a mixture-model-based clustering framework, this label correction method can be used in any clustering algorithm to force the label to be continuous and distinct. Through simulation studies, we show that the proposed approach provides competitive performance compared to existing algorithms, and is better when modelling trend changes and datasets with outliers. On real data, we also demonstrate the robust performance of the proposed approach in the presence of data with outliers. Using the UK house index dataset, we show its ability to identify meaningful changepoints for multivariate real data.

Our novel approach to changepoint detection connects the changepoint detection algorithm to a clustering problem and allows a natural unified framework for changepoint detection for both

univariate and multivariate data. However, in the multivariate scenario when p is large, the model can be computationally intensive especially when computing $\mathbf{\Sigma}^{-1}$. Moreover, to estimate $\mathbf{\Sigma}$ properly, the minimal number of points needed in each segment is going to increase as p increases. Thus, efficient estimation of $\mathbf{\Sigma}$ is necessary when p becomes large. Similar to any mixture model-based clustering, our approach can be sensitive to the initialization of the EM algorithm, especially for larger dimensional datasets. Some future work could focus on extending these models to allow for additional flexibility in modelling heavy-tailed and/or skewed data by utilizing heavy-tailed or skewed mixture models. Furthermore, future work will also include incorporating more complex structures into the model such as the AR process.

Appendix

Here we provide the parameter estimations for the remaining three models in the model family:

- VV model: This model can be considered as a special case of the trend model with $\boldsymbol{\beta}_{1g} = 0$. Thus, the updates of the parameters become:

$$\hat{\boldsymbol{\beta}}_{0g} = \frac{\sum_{i=1}^{n} \hat{z}_{ig}\mathbf{x}_i}{\sum_{i=1}^{n} \hat{z}_{ig}}$$

$$\hat{\mathbf{\Sigma}}_g = \frac{\sum_{i=1}^{n} \hat{z}_{ig}(\mathbf{x}_i - \hat{\boldsymbol{\beta}}_{0g})(\mathbf{x}_i - \hat{\boldsymbol{\beta}}_{0g})^T}{\sum_{i=1}^{n} \hat{z}_{ig}}.$$

- VE model: This model is a special case of "VV" model but with the constraint that

$$\mathbf{\Sigma}_1 = \cdots = \mathbf{\Sigma}_G = \mathbf{\Sigma}.$$

  Thus, the update for $\boldsymbol{\beta}_{0g}$ is exactly the same as the VV model but the update of $\mathbf{\Sigma}$ is given by:

$$\hat{\mathbf{\Sigma}}_g = \hat{\mathbf{\Sigma}} = \frac{\sum_{g=1}^{G}\sum_{i=1}^{n} \hat{z}_{ig}(\mathbf{x}_i - \hat{\boldsymbol{\beta}}_{0g})(\mathbf{x}_i - \hat{\boldsymbol{\beta}}_{0g})^T}{n}.$$

- EV model: This model is a special case of the "VV" model but with the constraint that

$$\boldsymbol{\beta}_{01} = \cdots = \boldsymbol{\beta}_{0G} = \boldsymbol{\beta}_0.$$

  Thus, the estimate of $\boldsymbol{\beta}_0$ is given by:

$$\hat{\boldsymbol{\beta}}_{0g} = \hat{\boldsymbol{\beta}}_0 = \frac{\sum_{i=1}^{n} \mathbf{x}_i}{n}.$$

References

Aitken, A. C. (1926), "A series formula for the roots of algebraic and transcendental equations," *Proceedings of the Royal Society of Edinburgh*, 45, 14–22.

Alippi, C., Boracchi, G., Carrera, D., and Roveri, M. (2016), "Change detection in multivariate datastreams: likelihood and detectability loss," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, AAAI Press, IJCAI'16, p. 1368–1374.

Beaulieu, C. and Killick, R. (2018), "Distinguishing trends and shifts from memory in climate data," *Journal of Climate*, 31, 9519–9543.

Dang, U. J., Gallaugher, M. P., Browne, R. P., and McNicholas, P. D. (2023), "Model-based clustering and classification using mixtures of multivariate skewed power exponential distributions," *Journal of Classification*, 40, 145–167.

Dawn, T., Roy, A., Manna, A., and Ghosh, A. K. (2021), "Some clustering-based change-point detection methods applicable to high dimension, low sample size data," *arXiv preprint arXiv:2111.14012*.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977), "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B*, 39, 1–38.

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

Fearnhead, P. and Rigaill, G. (2019), "Changepoint detection in the presence of outliers," *Journal of the American Statistical Association*, 114, 169–183.

Fraley, C. and Raftery, A. E. (1998), "How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis," *The Computer Journal*, 41, 578–588.

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

Grün, B. and Leisch, F. (2008), "FlexMix version 2: finite mixtures with concomitant variables and varying and constant parameters," *Journal of Statistical Software*, 28, 1–35.

Han, M., Vinh, L. T., Lee, Y.-K., and Lee, S. (2012), "Comprehensive context recognizer based on multimodal sensors in a smartphone," *Sensors*, 12, 12588–12605.

Hubert, L. and Arabie, P. (1985), "Comparing partitions," *Journal of Classification*, 2, 193–218.

James, N. A. and Matteson, D. S. (2014), "ecp: An R package for nonparametric multiple change point analysis of multivariate data," *Journal of Statistical Software*, 62, 1–25.

Joseph, L., Vandal, A. C., and Wolfson, D. B. (1996), "Estimation in the multipath change point problem for correlated data," *Canadian Journal of Statistics*, 24, 37–53.

Joseph, L. and Wolfson, D. B. (1993), "Maximum likelihood estimation in the multi-path change-point problem," *Annals of the Institute of Statistical Mathematics*, 45, 511–530.

Kawahara, Y. and Sugiyama, M. (2012), "Sequential change-point detection based on direct density-ratio estimation," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 5, 114–127.

Keribin, C. (2000), "Consistent estimation of the order of mixture models," *Sankhyā: The Indian Journal of Statistics, Series A (1961-2002)*, 62, 49–66.

Killick, R., Beaulieu, C., Taylor, S., and Hullait, H. (2021), *EnvCpt: Detection of Structural Changes in Climate and Environment TimeSeries*, R package version 1.1.3.

Killick, R. and Eckley, I. A. (2014), "changepoint: An R package for changepoint analysis," *Journal of Statistical Software*, 58, 1–19.

Killick, R., Haynes, K., and Eckley, I. A. (2022), *changepoint: An R package for changepoint analysis*, R package version 2.2.3.

Kuncheva, L. I. (2011), "Change detection in streaming multivariate data using likelihood detectors," *IEEE transactions on knowledge and data engineering*, 25, 1175–1180.

Lai, T. L. (1995), "Sequential changepoint detection in quality control and dynamical systems," *Journal of the Royal Statistical Society: Series B (Methodological)*, 57, 613–644.

Lau, H. F. and Yamamoto, S. (2010), "Bayesian online changepoint detection to improve transparency in human-machine interaction systems," in *49th IEEE Conference on Decision and Control (CDC)*, IEEE, pp. 3572–3577.

Li, Q., Yao, K., and Zhang, X. (2020), "A change-point detection and clustering method in the recurrent-event context," *Journal of Statistical Computation and Simulation*, 90, 1131–1149.

Lin, T. I. (2009), "Maximum likelihood estimation for multivariate skew normal mixture models," *Journal of Multivariate Analysis*, 100, 257–265.

Lindeløv, J. K. (2020), "mcp: An R package for regression with multiple change points," *OSF Preprints*.

Liu, S., Wright, A., and Hauskrecht, M. (2017), "Change-point detection method for clinical decision support system rule monitoring," in *Conference on Artificial Intelligence in Medicine in Europe*, Springer, pp. 126–135.

Liu, S., Yamada, M., Collier, N., and Sugiyama, M. (2013), "Change-point detection in time-series data by relative density-ratio estimation," *Neural Networks*, 43, 72–83.

Loschi, R. H. and Cruz, F. R. (2005), "Extension to the product partition model: computing the probability of a change," *Computational Statistics & Data Analysis*, 48, 255–268.

Loschi, R. H., Pontel, J. G., and Cruz, F. R. (2010), "Multiple change-point analysis for linear regression models," *Chilean Journal of Statistics*, 1, 93–112.

MacLahlan, G. and Peel, D. (2000), "Finite Mixture Models," *John Wiley & Sons*.

Malladi, R., Kalamangalam, G. P., and Aazhang, B. (2013), "Online Bayesian change point detection algorithms for segmentation of epileptic activity," in *2013 Asilomar Conference on Signals, Systems and Computers*, IEEE, pp. 1833–1837.

McNicholas, P. D. and Murphy, T. B. (2008), "Parsimonious Gaussian mixture models," *Statistics and Computing*, 18, 285–296.

National Research Council (2013), *Frontiers in Massive Data Analysis*, Washington, DC: The National Academies Press.

Page, E. S. (1954), "Continuous inspection schemes," *Biometrika*, 41, 100–115.

Pein, F., Hotz, T., Sieling, H., and Aspelmeier, T. (2022), *stepR: Multiscale change-point inference*, R package version 2.1-3.

Prescott Adams, R. and MacKay, D. J. (2007), "Bayesian online changepoint detection," *ArXiv e-prints*, arXiv–0710.

Roberts, E. and Zhao, L. (2022), "A Bayesian mixture model for changepoint estimation using ordinal predictors," *The International Journal of Biostatistics*, 18, 57–72.

Robinson, L. F., Wager, T. D., and Lindquist, M. A. (2010), "Change point estimation in multi-subject fMRI studies," *Neuroimage*, 49, 1581–1592.

Ruanaidh, J. J. O. and Fitzgerald, W. J. (1996), *Numerical Bayesian methods applied to signal processing*, Springer Science & Business Media.

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

Samé, A., Chamroukhi, F., Govaert, G., and Aknin, P. (2011), "Model-based clustering and segmentation of time series with changes in regime," *Advances in Data Analysis and Classification*, 5, 301–321.

Sarkar, S. and Zhu, X. (2022), "Multiple change point clustering of count processes with application to California COVID data," *Pattern Recognition Letters*, 157, 83–89.

Schwarz, G. (1978), "Estimating the dimension of a model," *The Annals of Statistics*, 6, 461–464.

Scrucca, L., Fop, M., Murphy, T. B., and Raftery, A. E. (2016), "mclust 5: clustering, classification and density estimation using Gaussian finite mixture models," *The R Journal*, 8, 289–317.

Shewhart, W. A. and Deming, W. (1940), "Statistical method from the viewpoint of quality control," *Philosophy of Science*, 7.

Subedi, S., Neish, D., Bak, S., and Feng, Z. (2020), "Cluster analysis of microbiome data by using mixtures of Dirichlet–multinomial regression models," *Journal of the Royal Statistical Society Series C*, 69, 1163–1187.

Taylor, S. J. and Letham, B. (2018), "Forecasting at scale," *The American Statistician*, 72, 37–45.

Tran, D.-H. (2019), "Automated change detection and reactive clustering in multivariate streaming data," in *2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF)*, IEEE, pp. 1–6.

Tu, W. and Subedi, S. (2022a), "A family of mixture models for biclustering," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 15, 206–224.

— (2022b), "Penalized logistic normal multinomial factor analyzers for high dimensional compositional data," *Journal of Statistical Research*, 56, 185–216.

Wald, A. (1945), "Sequential tests of statistical hypotheses," *The Annals of Mathematical Statistics*, 16, 117 – 186.

Wedel, M. (2002), "Concomitant variables in finite mixture models," *Statistica Neerlandica*, 56, 362–375.

Wyse, J., Friel, N., and Rue, H. (2011), "Approximate simulation-free Bayesian inference for multiple changepoint models with dependence within segments," *Bayesian Analysis*, 6, 501–528.

Zakaria, J., Mueen, A., and Keogh, E. (2012), "Clustering time series using unsupervised-shapelets," in *2012 IEEE 12th International Conference on Data Mining*, IEEE, pp. 785–794.

Zhu, X. and Melnykov, Y. (2022), "On finite mixture modeling of change-point processes," *Journal of Classification*, 39, 3–22.

Received: February 8, 2024

Accepted: July 13, 2024