



Detecting Multiple Change Points Using Adaptive Regression Splines With Application to Neural Recordings

Hazem Toutounji^{1*} and Daniel Durstewitz^{1,2}

¹ Department of Theoretical Neuroscience, Medical Faculty Mannheim, Bernstein Center for Computational Neuroscience, Central Institute of Mental Health, Heidelberg University, Mannheim, Germany, ² Faculty of Physics and Astronomy, Heidelberg University, Heidelberg, Germany

OPEN ACCESS

Edited by:

Markus Diesmann,
Forschungszentrum Jülich,
Helmholtz-Gemeinschaft Deutscher
Forschungszentren (HZ), Germany

Reviewed by:

Massimiliano Tamborino,
Johannes Kepler University of Linz,
Austria
Junji Ito,

Forschungszentrum Jülich,
Helmholtz-Gemeinschaft Deutscher
Forschungszentren (HZ), Germany

*Correspondence:

Hazem Toutounji
hazem@ini.ethz.ch

†Present Address:

Hazem Toutounji
Institute of Neuroinformatics,
University of Zurich and ETH Zurich,
Zurich, Switzerland

Received: 13 February 2018

Accepted: 11 September 2018

Published: 04 October 2018

Citation:

Toutounji H and Durstewitz D (2018)
Detecting Multiple Change Points
Using Adaptive Regression Splines
With Application to Neural Recordings.
Front. Neuroinform. 12:67.
doi: 10.3389/fninf.2018.00067

Time series, as frequently the case in neuroscience, are rarely stationary, but often exhibit abrupt changes due to attractor transitions or bifurcations in the dynamical systems producing them. A plethora of methods for detecting such *change points* in time series statistics have been developed over the years, in addition to test criteria to evaluate their significance. Issues to consider when developing change point analysis methods include computational demands, difficulties arising from either limited amount of data or a large number of covariates, and arriving at statistical tests with sufficient power to detect as many changes as contained in potentially high-dimensional time series. Here, a general method called *Paired Adaptive Regressors for Cumulative Sum* is developed for detecting multiple change points in the mean of multivariate time series. The method's advantages over alternative approaches are demonstrated through a series of simulation experiments. This is followed by a real data application to neural recordings from rat medial prefrontal cortex during learning. Finally, the method's flexibility to incorporate useful features from state-of-the-art change point detection techniques is discussed, along with potential drawbacks and suggestions to remedy them.

Keywords: change point, cumulative sum, adaptive regression splines, nonstationary, bootstrap test, block-permutation, behavior, spike counts

1. INTRODUCTION

Stationary data are the exception rather than the rule in many areas of science (Paillard, 1998; Elsner et al., 2004; Shah et al., 2007; Aston and Kirch, 2012; Stock and Watson, 2014; Fan et al., 2015; Latimer et al., 2015; Gärtner et al., 2017). Time series statistics often change, sometimes abruptly, due to transitions in the underlying system dynamics, adaptive processes or external factors. In neuroscience, both behavioral *time series* (Smith et al., 2004; Durstewitz et al., 2010; Powell and Redish, 2016) and their neural correlates (Roitman and Shadlen, 2002; Durstewitz et al., 2010; Gärtner et al., 2017) exhibit strongly nonstationary features which relate to important cognitive processes such as learning (Smith et al., 2004; Durstewitz et al., 2010; Powell and Redish, 2016) and perceptual decision making (Roitman and Shadlen, 2002; Latimer et al., 2015; Hanks and Summerfield, 2017). As such, identifying nonstationary features in behavioral and neural time series becomes necessary, both for interpreting the data in relation to the potential

influences generating those features, and for removing those features from the data in order to perform statistical analyses that assume stationary observations (Hamilton, 1994; Shumway and Stoffer, 2010). Abrupt jumps in time series statistics form one important class of nonstationary events. These are often caused by bifurcations, which, in turn, may occur with gradual changes in parameters of the underlying system (Strogatz, 2001). Consequently, they are of wide interest to both statistical data analysis and the study of dynamical systems, and are commonly referred to as *change points* (CPs; Chen and Gupta, 2012).

Detecting CPs has a long and varied history in statistics, and we will not attempt to exhaustively survey the different approaches, including regression models (Quandt, 1958; Brown et al., 1975), Bayesian techniques (Chernoff and Zacks, 1964) and *cumulative sum* (CUSUM) statistics (Page, 1954; Basseville, 1988), to name but a few, within the limited scope of this article. Instead, we refer the reader to the excellent reviews on the topic (Bhattacharya, 1994; Chen and Gupta, 2012; Aminikhanghahi and Cook, 2017) and focus on the *offline* CUSUM class of methods (Hinkley, 1971a, as opposed to sequential CUSUM methods, Page, 1954, that locate a CP online, while the time series is evolving), specifically methods that aim at detecting CPs in the *mean* of the time series. CUSUM-based methods are powerful, easy to implement, and are backed up by an extensive literature, theoretical results and various extensions to multiple CPs and multivariate scenarios, making them an ideal starting point. These methods assume that the time series is piecewise stationary in the statistic under consideration (e.g., piecewise constant mean) and rely on a cumulative sum transformation of the time series. Commonly, *at-most-one-change* (AMOC) is identified by maximum-type statistics (Kirch, 2007) at the extremum of the curve resulting from that transformation (Basseville, 1988; Antoch et al., 1995).

Extending the CUSUM method to multiple CPs usually involves repetitive partitioning of the time series upon each detection (*binary segmentation* methods; Scott and Knott, 1974; Bai, 1997; Olshen et al., 2004; Fryzlewicz, 2014; Cho and Fryzlewicz, 2015). This segmentation procedure, however, may hamper detection in later iterations as the reduction in number of observations depletes statistical power exponentially fast as more CPs are to be retrieved. In this article, we develop the PARCS (*Paired Adaptive Regressors for Cumulative Sum*) method which offers a straightforward extension that leverages the full time series in order to detect multiple CPs, thus providing a new solution to this issue. PARCS rests on the fact that a CUSUM transformation of the data relates to computing an integral transformation of the piecewise constant mean time series model, resulting in a *piecewise linear* mean function that bends at potential CPs and could be approximated by *adaptive regression spline* methods (Friedman and Silverman, 1989; Friedman, 1991; Stone et al., 1997). Namely, rather than attempting to approximate the discontinuous time series mean directly (Efron et al., 2004; Vert and Bleakley, 2010), the PARCS model is an approximation to the continuous CUSUM-transformed time series by a piecewise linear function. The bending points of the PARCS model are

each defined by a pair of non-overlapping piecewise linear regression splines that are first selected by a two-stage iterative procedure.

The PARCS model is further refined by a nonparametric CP significance test based on bootstraps (Dumbgen, 1991; Antoch and Hušková, 2001; Hušková, 2004; Kirch, 2007; Matteson and James, 2014). While analytically derived parametric tests may usually be preferable over bootstrap-based tests due to better convergence and coverage of the tails, in the current CP setting closed form expressions for parametric tests are hard to come by and are usually replaced by approximations (Gombay and Horváth, 1996; Horváth, 1997). In this case, tests based on bootstraps are preferable since they are known to converge faster to the limit distribution of the test statistic (often they are also not as conservative as parametric approximations for datasets of a relatively small size; Csörgö and Horváth, 1997; Antoch and Hušková, 2001; Kirch, 2007). In order to accommodate the possibility of *temporally dependent* noise in the data (Picard, 1985; Antoch et al., 1997; Horváth, 1997), model selection is carried out by a nonparametric *block-permutation* bootstrap procedure (Davison and Hinkley, 1997; Hušková and Slabý, 2001; Kirch, 2007) developed specifically for PARCS, which relies on a test statistic that quantifies the amount of bending at each candidate CP. Since model estimation is based on linear regression, PARCS is also effortlessly extended to spatially independent, *multivariate* time series.

The article is structured as follows. Section 2.1 introduces the CUSUM method for AMOC detection. We then develop the PARCS method, presenting in Section 2.2 the procedure for inferring a nested model that allows for significance testing of multiple CPs, followed in Section 2.3 by an outline of the nonparametric permutation test procedure for refining the PARCS model further. Results in Section 3 illustrate that PARCS improves on several issues inherent in classical methods for change point analysis. In Section 3.1, we compare the PARCS approach to the CUSUM method in detecting a single CP, followed in Section 3.2 by a comparison with standard binary segmentation in detecting multiple CPs. We also demonstrate in Section 3.3 that PARCS is successful in detecting CPs in spatially independent, multivariate time series. We then present in Section 3.4 an example from the neurosciences, in which neural and behavioral CPs are compared during operant rule-switching learning (Durstewitz et al., 2010). Finally, we discuss in Section 4 the PARCS approach in relation to other state-of-the-art CP detection methods, along with drawbacks and potential extensions.

2. METHODS

This section outlines the CUSUM method and the PARCS extension to multiple CPs, in addition to a nonparametric permutation technique to test for the statistical significance of CPs as identified by PARCS. For generality, the formulation assumes temporally dependent observations in the time series, independent observations being a special case.

2.1. CUSUM: Cumulative Sum of Differences to the Mean

A class of methods for identifying a single CP in the mean relies on computing a CUSUM transformation of the time series $\mathbf{x} = \{x_t\}_{1:T}$. A useful formulation that allows for dependent observations in the time series is given by the moving average (MA) step model (Antoch et al., 1997; Horváth, 1997; Kirch, 2007),

$$x_t = b + w \cdot \mathbf{1}_{t-c} + \sum_{\tau \geq 0} \kappa_\tau \epsilon_{t-\tau} \quad ; \quad \kappa_0 = 1, \epsilon_t \sim \mathcal{N}(0, \sigma^2), \quad (1)$$

where a jump in the time series mean from *baseline* b to $b + w$ occurs after time step c , the change point. The *step parameter* or *weight* w is positive (negative) when the time series mean increases (decreases) following c . The largest integer τ such that noise coefficient $\kappa_\tau \neq 0$ defines a finite order q of the MA process, which is 0 for temporally independent observations. We will assume that the MA process is stationary, which will always be the case if it is finite, with ϵ_t independent and identically distributed (i.i.d.) random variables (for an infinite process, points x_t for $t \leq 0$ may be considered unobserved, and coefficients κ_τ have to fulfill certain conditions to make the process stationary, as given, for instance, in Shumway and Stoffer, 2010). The Gaussian noise assumption in the MA process can be relaxed, as long as the noise process has zero mean and finite, constant variance (see Lombard and Hart, 1994; Antoch et al., 1997; Horváth, 1997; Kirch, 2007, for theoretical results on the more general form of dependent noise). The discrete Heaviside step function, $\mathbf{1}_{t-c}$, is defined by,

$$\mathbf{1}_i = \begin{cases} 1 & \text{if } i > 0, \\ 0 & \text{otherwise.} \end{cases}$$

Identifying the presence of a CP requires testing the null hypothesis, $H_0 : w = 0$, against the alternative, $H_1 : w \neq 0$ (Lombard and Hart, 1994; Antoch et al., 1995). This begins by inferring the time of the step according to a CP locator statistic. A typical offline CP locator statistic is the maximum point of the weighted absolute cumulative sum of differences to the mean (Horváth, 1997; Antoch and Hušková, 2001),

$$\hat{c} = \arg \max_{0 < t < T} \left(\frac{T}{t(T-t)} \right)^\gamma \left| \sum_{\tau=1}^t (x_\tau - \langle \mathbf{x} \rangle) \right|, \quad (2)$$

where $\langle \mathbf{x} \rangle$ is the arithmetic mean of the time series (see **Figure 1A**). The first term on the right-hand side corrects for bias toward the center, where more centrally-located points are down-weighted by an amount controlled by parameter $\gamma \in [0, 0.5]$. Other CUSUM-based locator statistics exist with different bias-correcting terms and cumulative sum transformations (Bhattacharya, 1994; Antoch et al., 1997; Kirch, 2007; Jirak, 2012). As outlined in the Discussion, PARCS may be modified to include such bias-correcting terms as well. However, as we will demonstrate, PARCS can significantly reduce center bias even without recourse to such a term. To show this, we will mostly deal with the generic case, $\gamma = 0$, when comparing

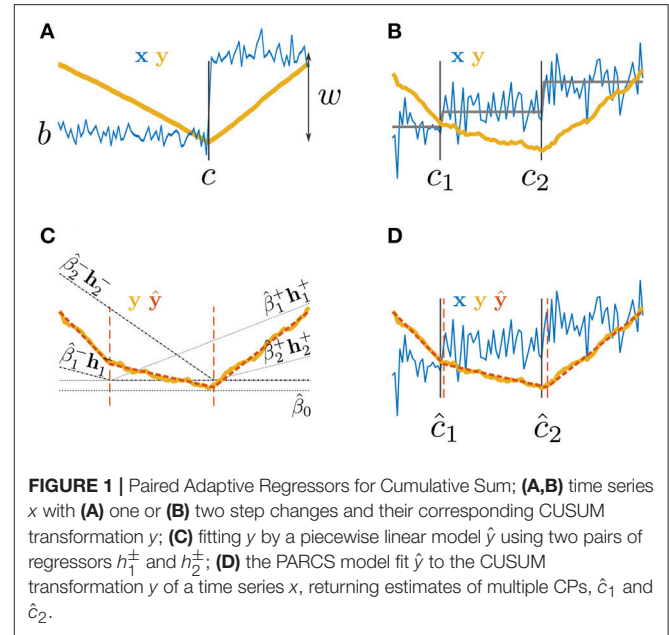


FIGURE 1 | Paired Adaptive Regressors for Cumulative Sum; **(A,B)** time series x with **(A)** one or **(B)** two step changes and their corresponding CUSUM transformation y ; **(C)** fitting y by a piecewise linear model \hat{y} using two pairs of regressors h_1^+ and h_2^+ ; **(D)** the PARCS model fit \hat{y} to the CUSUM transformation y of a time series x , returning estimates of multiple CPs, \hat{c}_1 and \hat{c}_2 .

PARCS to the CUSUM transformation as defined in Equation 2. This has the added advantage of avoiding having to select an optimal power or an optimal weight factor, a choice that usually depends on prior assumptions on the CP's potential location (Bhattacharya, 1994). As such, and unless stated otherwise, the term *CUSUM transformation* will refer, thereof, to the cumulative sum of differences to the mean,

$$y_t \triangleq \sum_{\tau=1}^t (x_\tau - \langle \mathbf{x} \rangle), \quad (3)$$

where the maximum value,

$$S = \max_{0 < t < T} \left| \sum_{\tau=1}^t (x_\tau - \langle \mathbf{x} \rangle) \right| = \max_{0 < t < T} |y_t|, \quad (4)$$

defines a test statistic by which it is decided whether to reject the null hypothesis.

Given potentially dependent observations, $q > 0$, as defined by the model in Equation 1, nonparametric bootstrap testing proceeds by *block-permutation* (Davison and Hinkley, 1997; Hušková, 2004; Kirch, 2007), such that temporal dependence in the data is preserved (see Section 3.2). The candidate CP \hat{c} is identified according to Equation 2 and its associated test statistic S is computed by Equation 4. Estimates \hat{b} and \hat{w} are retrieved from the arithmetic means of \mathbf{x} before and after \hat{c} using the model in Equation 1. By subtracting $\hat{w} \cdot \mathbf{1}_{t-\hat{c}}$ from the time series \mathbf{x} we arrive at a time series \mathbf{x}_0 that provides an estimate of the null distribution. The stationary time series \mathbf{x}_0 is split into n blocks of size k , chosen such that temporal dependencies are mostly preserved in the permuted time series (Davison and Hinkley, 1997). One way to do so is to select the block size to be larger than the order of the underlying MA process, $q + 1$ (since the

autocorrelation function of an MA(q) process cuts off at order q ; Davison and Hinkley, 1997). This requires identifying the order q which can be determined from the H_0 -conform time series \mathbf{x}_0 by inspecting its autocorrelation function (Fan and Yao, 2003) for different time lags τ . The autocorrelation function's asymptotic distribution (Kendall and Stuart, 1983),

$$\text{acorr}(\mathbf{x}_0; \tau) \sim \mathcal{N}(-1/(T - \tau), +1/(T - \tau)), \quad (5)$$

provides a test statistic for deciding the largest time lag q at which to reject the null hypothesis $H_0: \text{acorr}(\mathbf{x}_0; q) = 0$, given some preset significance level $\alpha \in [0, 1]$. The resulting blocks are randomly permuted and each permutation is CUSUM-transformed according to Equation 3 to compute an H_0 -conform sample S_0 of the test statistic S in Equation 4 (note that we do not know the true step parameter w or the true CP c , of course, such that this procedure will only yield an estimate of the H_0 distribution). A sufficiently large number B of permutations results in samples S_i of an H_0 -conform empirical distribution function (EDF) $F(S_0) \triangleq \sum_{i=1}^B \mathbf{1}_{S_0 - S_i} / B$ that weighs every sample S_i equally. The candidate CP \hat{c} is detected when the test statistic S as computed from the original time series \mathbf{x} satisfies $S \geq F^{-1}(1 - \alpha)$, where α is a preset significance level and $F^{-1}(1 - \alpha)$ the inverse of the EDF, defined as the $(1 - \alpha)B^{\text{th}}$ largest value out of B permutations (Davison and Hinkley, 1997; Durstewitz, 2017).

2.2. PARCS: Paired Adaptive Regressors for Cumulative Sum

The PARCS method for estimating multiple CPs rests on the fact that the integral of a piecewise constant function is piecewise linear. The AMOC model as defined in Equation 1 assumes a piecewise stationary MA process, consisting of two segments with constant mean. A process consisting of $M + 1$ segments generalizes Equation 1 to data containing at-most- M -change,

$$x_t = b + \sum_{m=1}^M w_m \cdot \mathbf{1}_{t-c_m} + \sum_{\tau \geq 0} \kappa_\tau \epsilon_{t-\tau} \quad ; \quad \kappa_0 = 1, \epsilon_t \sim \mathcal{N}(0, \sigma^2). \quad (6)$$

The CUSUM transformation $\mathbf{y} = \{y_t\}_{1:T}$ of this process as given by Equation 3 corresponds to the numerical integration of a piecewise stationary process $\mathbf{x} - \langle \mathbf{x} \rangle$. That is, \mathbf{y} is approximately (due to the noise) piecewise linear (exactly piecewise linear in the mean; see Figure 1B). If points $\{c_m\}_{1:M}$ at which \mathbf{y} bends were known, the latter can be fitted by a weighted sum of local piecewise linear basis functions or *splines*, centered at the *knots* $\{c_m\}_{1:M}$,

$$h_{t,c_m}^+ = \begin{cases} t - c_m & \text{if } t > c_m \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad h_{t,c_m}^- = \begin{cases} c_m - t & \text{if } t < c_m \\ 0 & \text{otherwise} \end{cases}$$

This fit corresponds to modeling the expected value of \mathbf{y} , conditioned on spline pair set $\mathcal{H} = \{\mathbf{h}_{c_m}^\pm\}_{1:M}$, resulting in model inference,

$$E[y_t | \mathcal{H}] \approx \hat{y}_{t,M} = \hat{\beta}_0 + \sum_{m=1}^M \hat{\beta}_m^+ h_{t,c_m}^+ + \sum_{m=1}^M \hat{\beta}_m^- h_{t,c_m}^-$$

which is a simple regression problem that can be solved by estimating the intercept $\hat{\beta}_0$ and coefficients $\hat{\beta}_m^\pm$ that minimize the mean-square-error,

$$\text{mse}_M(\mathbf{y}, \hat{\mathbf{y}}_M) = \frac{1}{T} \sum_{t=1}^T (y_t - \hat{y}_{t,M})^2. \quad (7)$$

However, in the multiple CP detection setting (assuming M is known), optimal knot placement is not known *a priori*, but can be inferred by *adaptively* adjusting knot locations (Friedman and Silverman, 1989; Friedman, 1991; Stone et al., 1997) to maximally satisfy the goodness-of-fit criterion in Equation 7. In other words, and as shown in Figures 1C,D, the problem of identifying multiple CPs is replaced by the equivalent problem of inferring the order- M PARCS model (or PARCS $_M$ model),

$$\hat{\mathbf{y}}_M = \hat{\beta}_0 + \sum_{m=1}^M \hat{\beta}_m^+ \mathbf{h}_{c_m}^+ + \sum_{m=1}^M \hat{\beta}_m^- \mathbf{h}_{c_m}^-, \quad (8)$$

with associated M -tuple $\hat{\mathbf{c}} \triangleq (\hat{c}_m)_{1:M}$ that best fits the CUSUM transformation of the time series. Regression coefficients in model 8 are real numbers, while knots in the present time series context are positive integers, excluding the first and last time steps, $\hat{c}_m \in \{2, 3, \dots, T - 1\}$.

Fitting the PARCS $_M$ model is based on a forward/backward spline selection strategy (Smith, 1982) with added CP ranking stage and proceeds as outlined in Algorithm 1. Starting with an empty PARCS $_0$ model, containing only the intercept $\hat{\beta}_0$, a *forward* sweep increases model complexity to a forward upper bound order $L > M$ by adding at each iteration the spline pair \mathbf{h}_c^\pm , not yet contained in the model, that decreases residual mean-square-error the most. A reasonable heuristic for setting L is 2–3 times M (assuming M is known or given some liberal guess). This is followed by a backward *pruning* iteration, in which the spline pair whose removal increases residual mean-square-error the least is dropped from the model. Pruning removes those knots that were added at the beginning of the forward phase which became redundant as the model was refined by later additions (Friedman and Silverman, 1989). This stage continues until the number of knots reaches the preset final upper bound of model complexity M , i.e., $L - M$ knots are pruned. Knots are then sorted in descending order according to the amount of explained variance. The *ranking* iteration returns a nested model by pruning the PARCS $_M$ model further, down to the PARCS $_0$ model. The first knot to be pruned, reducing the number of knots to $M - 1$, explains the least variance and is placed last as \hat{c}_M in the M -tuple $\hat{\mathbf{c}}$. The last knot to be pruned explains the most variance and is placed first as \hat{c}_1 . Note that regression coefficients are re-estimated every time a knot is added to or removed from the PARCS model.

The model can be effortlessly extended to the multiple response setting in the case of spatially independent time series (extension to a nondiagonal MA covariance matrix, Stone et al., 1997, will be considered elsewhere). Given N independent,

Algorithm 1: Procedure for inferring the PARCS_M model with forward/backward spline selection (first/second loop) and CP ranking (third loop). Regression coefficients are computed by least squares estimation, conditioned on the set of knot locations of predefined size M that minimizes mean-square-error. Final knot locations are specified by eliminating spurious knots through block-permutation bootstrapping as described in Section 2.3.

Input: L, M and \mathbf{y}
Output: $\hat{\mathbf{c}} \triangleq (\hat{c}_m)_{1:M}$ and $\hat{\mathbf{y}}_M$
 $\hat{\mathcal{H}}, \mathcal{H} \leftarrow \emptyset$
for $m \leftarrow 1$ **to** L **do** // forward stage
 $\hat{c}_m \leftarrow \arg \min_{1 < c < T} \{ \text{mse}_m(\mathbf{y}, \hat{\mathbf{y}}_{\mathcal{H} \cup \mathbf{h}_c^\pm}) | \mathbf{h}_c^\pm \notin \mathcal{H} \}$
 $\mathcal{H} \leftarrow \mathcal{H} \cup \mathbf{h}_{\hat{c}_m}^\pm$
for $m \leftarrow L$ **to** $M + 1$ **do** // pruning stage
 $\hat{c} \leftarrow \arg \min_c \{ \text{mse}_{m-1}(\mathbf{y}, \hat{\mathbf{y}}_{\mathcal{H} \setminus \mathbf{h}_c^\pm}) \}$
 $\hat{\mathbf{c}} \leftarrow \hat{\mathbf{c}} \setminus \hat{c}$ and $\mathcal{H} \leftarrow \mathcal{H} \setminus \mathbf{h}_{\hat{c}}^\pm$
 $\hat{\mathbf{y}}_M \leftarrow \hat{\beta}_0 + \sum_{m=1}^M \hat{\beta}_m^+ \mathbf{h}_{\hat{c}_m}^+ + \sum_{m=1}^M \hat{\beta}_m^- \mathbf{h}_{\hat{c}_m}^-$
for $m \leftarrow M$ **to** 1 **do** // ranking stage
 $\hat{c}_m \leftarrow \arg \min_c \{ \text{mse}_{m-1}(\mathbf{y}, \hat{\mathbf{y}}_{\mathcal{H} \setminus \mathbf{h}_c^\pm}) \}$
 $\mathcal{H} \leftarrow \mathcal{H} \setminus \mathbf{h}_{\hat{c}_m}^\pm$

piecewise stationary MA processes with common CPs $\{c_m\}_{1:M}$,

$$x_{t,n} = b_n + \sum_{m=1}^M w_{mn} \cdot \mathbf{1}_{t-c_m} + \sum_{\tau \geq 0} \kappa_\tau \epsilon_{t-\tau,n};$$

$$\kappa_0 = 1, \epsilon_{t,n} \sim \mathcal{N}(0, \sigma^2), \tag{9}$$

where $n = 1, \dots, N$, the corresponding multivariate CUSUM transformation $\mathbf{y}_t = \{y_{t,n}\}_{1:N}$ is fitted by the multiple response, PARCS_M model, conditioned on common spline pairs,

$$E[y_{t,n} | \mathcal{H}] \approx \hat{y}_{t,M,n} = \hat{\beta}_{0n} + \sum_{m=1}^M \hat{\beta}_{mn}^+ h_{t,\hat{c}_m}^+ + \sum_{m=1}^M \hat{\beta}_{mn}^- h_{t,\hat{c}_m}^-,$$

using **Algorithm 1**. Returning CPs that are common to all variables \mathbf{x}_n is done by using the goodness-of-fit criterion in Equation 7, averaged over all responses \mathbf{y}_n .

2.3. PARCS Model Selection by Block-Permutation Bootstrap

The piecewise linear PARCS formulation, Equation 8, of the CUSUM transformation in Equation 3 bends at the CPs. Due to the presence of noise in the original time series \mathbf{x} , some noise realizations may appear as slight bends in the CUSUM-transformed time series, leading PARCS to return false CPs. As such, the amount of bending at knot \hat{c}_m can be used as a test statistic for bootstrap significance testing that can refine the PARCS model further. No bending indicates either a constant fit, $\hat{\beta}_m^+ = \hat{\beta}_m^- = 0$, or a smooth linear fit, $\hat{\beta}_m^+ = -\hat{\beta}_m^-$ (see

also **Figure 1C**). Thus, a suitable test statistic that quantifies the amount of bending at \hat{c}_m is given by,

$$S = |\hat{\beta}_m^+ + \hat{\beta}_m^-|, \tag{10}$$

where for multivariate time series, the test statistic is the average over all time series.

Before describing the block-permutation bootstrap method for PARCS, we outline a procedure for identifying the order q of the MA noise process, provided as pseudocode in **Algorithm 2**. First, an H_0 -conform time series $\mathbf{x}_0 = \{x_{t,0}\}_{1:T}$ is computed by regressing out the PARCS model $\hat{\mathbf{y}}_M$ of Equation 8 from the CUSUM-transformed time series \mathbf{y} and then inverting the CUSUM transformation. This is followed by inspecting the autocorrelation function of \mathbf{x}_0 for different time lags τ . The largest time lag at which the null hypothesis $H_0 : \text{acorr}(\mathbf{x}_0; q) = 0$ is rejected, given some preset significance level $\alpha \in [0, 1]$ is then returned as the order q , given some predefined upper bound of MA order, Q .

Algorithm 2: Identifying the order q of the MA process, given some upper bound Q . The H_0 -conform time series \mathbf{x}_0 is estimated before entering the loop. The loop increases the autocorrelation time lag and exits when the autocorrelation of \mathbf{x}_0 is not significantly different from 0 anymore.

Input: $\mathbf{y}, \hat{\mathbf{c}}, Q$ and α
Output: $q \leq Q$
 $q \leftarrow Q$
 $\mathbf{y}_0 \leftarrow \mathbf{y} - \left(\hat{\beta}_0 + \sum_{m=1}^M \hat{\beta}_m^+ \mathbf{h}_{\hat{c}_m}^+ + \sum_{m=1}^M \hat{\beta}_m^- \mathbf{h}_{\hat{c}_m}^- \right)$
 $x_{t,0} \leftarrow y_{t,0} - y_{t-1,0} + \langle \mathbf{x} \rangle$ for $t = 1, \dots, T$ where $y_{0,0} = 0$
for $\tau \leftarrow 1$ **to** Q **do**
 $F_\tau \leftarrow \text{CDF}(\mathcal{N}(-1/(T-\tau), +1/(T-\tau)))$
if $\text{acorr}(\mathbf{x}_0; \tau) \in [F_\tau^{-1}(\alpha/2), F_\tau^{-1}(1-\alpha/2)]$ **then**
 $q \leftarrow \tau - 1$
break

Given the M -tuple CP set $\hat{\mathbf{c}}$ returned by **Algorithm 1** and an estimate of the dependent normal noise order q by **Algorithm 2**, a block-permutation bootstrap test returns the subset $\hat{\mathcal{C}}$ of significant CPs, as outlined in **Algorithm 3**. First, an H_0 -conform time series \mathbf{x}_0 is computed. For each CP $\hat{c}_m \in \hat{\mathbf{c}}$, starting with the one ranked highest, all CP-splines already deemed significant by the bootstrap test are regressed out of \mathbf{y} . A PARCS model with the remaining knots, including \hat{c}_m , is estimated and the test statistic S , evaluated at \hat{c}_m according to Equation 10, is computed. Knot \hat{c}_m is tested for significance against an H_0 -conform EDF, estimated through block-permutation bootstrapping: A total of B bootstrap samples is generated from the H_0 -conform series \mathbf{x}_0 by randomly permuting blocks of size $k = q + 1$. For each of these $i = 1 \dots B$ bootstrap samples test statistic S_i is evaluated at knot location \hat{c}_m , yielding an EDF $F(S_0)$ which assigns equal probability $1/B$ to each bootstrapped S_i . A significant \hat{c}_m is then added to $\hat{\mathcal{C}}$, or rejected as false discovery otherwise. The procedure repeats for the CP next

in the rank order. Similar to **Algorithm 1**, regression coefficients are re-estimated every time a knot is added to or removed from the PARCS model.

Algorithm 3: Block-permutation bootstrap procedure for PARCS, given block size k . The H_0 -conform time series \mathbf{x}_0 is estimated before entering the loop. The loop iterates over the rank-ordered CPs to test for each CP's significance.

Input: $\mathbf{y}, \hat{c}, k, B$ and α

Output: $\hat{\zeta} \subseteq \hat{c}$

$\hat{\zeta} \leftarrow \emptyset$

$\mathbf{y}_0 \leftarrow \mathbf{y} - \left(\hat{\beta}_0 + \sum_{m=1}^M \hat{\beta}_m^+ \mathbf{h}_{\hat{c}_m}^+ + \sum_{m=1}^M \hat{\beta}_m^- \mathbf{h}_{\hat{c}_m}^- \right)$

$x_{t,0} \leftarrow y_{t,0} - y_{t-1,0} + \langle \mathbf{x} \rangle$ for $t = 1, \dots, T$ where $y_{0,0} = 0$

for $m \leftarrow 1$ **to** M **do**

$\mathbf{y}_{\hat{c} \setminus \hat{\zeta}} \leftarrow \mathbf{y} - \left(\hat{\beta}_0 + \sum_{\mu=1}^{|\hat{\zeta}|} \hat{\beta}_\mu^+ \mathbf{h}_{\hat{\zeta}_\mu}^+ + \sum_{\mu=1}^{|\hat{\zeta}|} \hat{\beta}_\mu^- \mathbf{h}_{\hat{\zeta}_\mu}^- \right)$

$\hat{\mathbf{y}}_{\hat{c} \setminus \hat{\zeta}} \leftarrow \hat{\beta}_0 + \sum_{\mu=m}^M \hat{\beta}_\mu^+ \mathbf{h}_{\hat{c}_\mu}^+ + \sum_{\mu=m}^M \hat{\beta}_\mu^- \mathbf{h}_{\hat{c}_\mu}^-$

$S \leftarrow |\hat{\beta}_m^+ + \hat{\beta}_m^-|$

$F(S_0) \leftarrow \text{BlockPermutationBootstrap}(\mathbf{x}_0, \hat{c}_m, k, B)$

if $S \geq F^{-1}(1 - \alpha)$ **then**

$\hat{\zeta} \leftarrow \hat{\zeta} \cup \hat{c}_m$

3. RESULTS

We first evaluate the PARCS method on synthetic data in single and multiple CP detection settings, followed by a real data example on detecting behavioral and neural change points during rule learning.

3.1. Alleviating CUSUM Bias in AMOC Detection

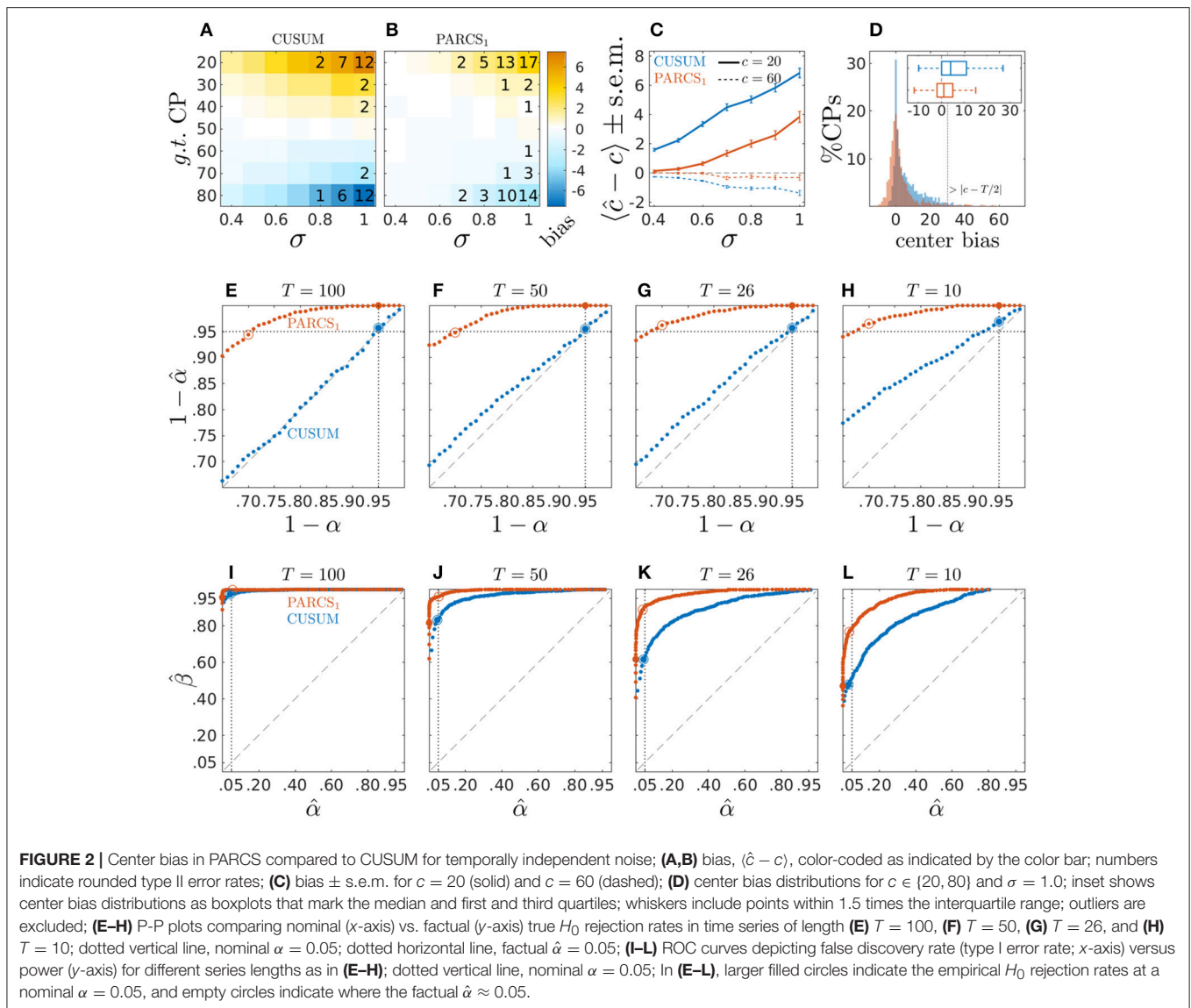
We first compare the CUSUM method for detecting a single CP to the PARCS approach in order to evaluate the effect of each method on the center bias in CP detection. Both white and MA Gaussian noise are considered. We also compare PARCS to the CUSUM locator statistic of Equation 2 with $\gamma = 0.5$ (the maximum likelihood estimator of CP location under the assumption of i.i.d. Gaussian noise) and identify conditions under which one method is preferable over the other.

Univariate time series of length $T = 100$ are simulated according to the step model in Equation 1 with different levels of white Gaussian noise, $\sigma \in \{0.4, 0.5, \dots, 1.0\}$, and different ground truth CP locations, $c \in \{20, 30, \dots, 80\}$. Baseline is set to $b = 0$ and step parameter to $w = 1$. Note that in the step model with white Gaussian noise, increasing σ is equivalent to reducing w . A single CP was identified by using the CUSUM method and estimating the PARCS₁ model, both followed by bootstrap significance testing with $B = 10,000$ permutations, nominal significance level $\alpha = 0.05$, and blocks of size $k = 1$ (since noise is independent in this example). Each parameter configuration was repeated for 1,000 noise realizations.

We compare bias in CP detection toward the center of the time series in both the CUSUM and PARCS methods. We measure this *center bias* by $cb = (2 \cdot \mathbf{1}_{c-T/2} - 1) \cdot (c - \hat{c})$, which is positive when estimate \hat{c} falls onto the side located toward the center from c , and is negative otherwise. As expected given the choice $\gamma = 0$ in Equation 2, the CUSUM method shows a strong center bias which increases for lower signal-to-noise ratio and more peripheral CPs (see **Figure 2A**). The CUSUM method's power decreases for harder parameter settings (higher σ and more peripheral c) in that true CPs are missed in more of the realizations. The PARCS method results in a significant reduction in center bias but does not eliminate it completely, and yields more misses relative to CUSUM if both are run at the same nominal α level (see **Figure 2B**). Summary comparison between the two methods is shown in **Figure 2C** for two exemplary CP locations, $c \in \{20, 60\}$.

To fully appreciate the source of CUSUM center bias and its reduction by PARCS, time series realizations with the two hardest parameter settings ($c \in \{20, 80\}$ and $\sigma = 1.0$) are considered in **Figure 2D**, which compares the distribution of cb in the 81% of realizations in which both CUSUM and PARCS returned a CP. The histograms show a strongly skewed, heavy-tailed distribution for CUSUM, compared to a more symmetric distribution around 0 for PARCS, indicating only little bias. Most of the center bias in PARCS is accounted for by outliers. This is illustrated by excluding outliers in the boxplots, which show a median of 1 time step center bias in PARCS against median center bias of 4 time steps in the case of CUSUM. Note that measuring center bias as defined above does not differentiate between biased detections and false discoveries where, in extreme cases, a CP may be detected beyond the middle point $T/2$ of the time series, corresponding to center bias greater than $|c - T/2|$. However, this scenario rarely occurred in the simulation results reported here.

While PARCS reduces center bias, the simulation results above indicate that it behaves more conservatively than CUSUM at the same nominal α level. In principle, false H_1 rejection rates (type II errors) may be reduced by adjusting the α level, at the same time producing more false discoveries. In order to assess how well the nominal significance level α agrees with the empirical type I error rate (false discoveries), 1,000 white Gaussian noise realizations of length $T = 100$ are simulated with $\sigma = 1.0$. Conclusions drawn from this analysis are largely the same for larger signal-to-noise ratio (results not shown). A single CP was extracted using the CUSUM method and estimating the PARCS₁ model on these time series conforming to the null hypothesis $H_0: w = 0$. Type I error rates at different nominal α levels are shown in **Figure 2E** as *probability-probability* (P-P) plots, depicting the nominal, $1 - \alpha$, against the empirical, $1 - \hat{\alpha}$, probabilities of accepting the null hypothesis when the null hypothesis is true. While the empirical type I error rate of CUSUM perfectly agrees with the nominal significance level, for PARCS, in contrast, the empirical rate of false discoveries tends toward 0% for $\alpha = 0.05$ and remains smaller than 1% for α as large as 0.18. This entails that PARCS behaves highly conservatively, and that the nominal α level may be adjusted considerably upward without strongly influencing the false discovery rate. On the other hand, despite being more conservative, **Figure 2I** shows

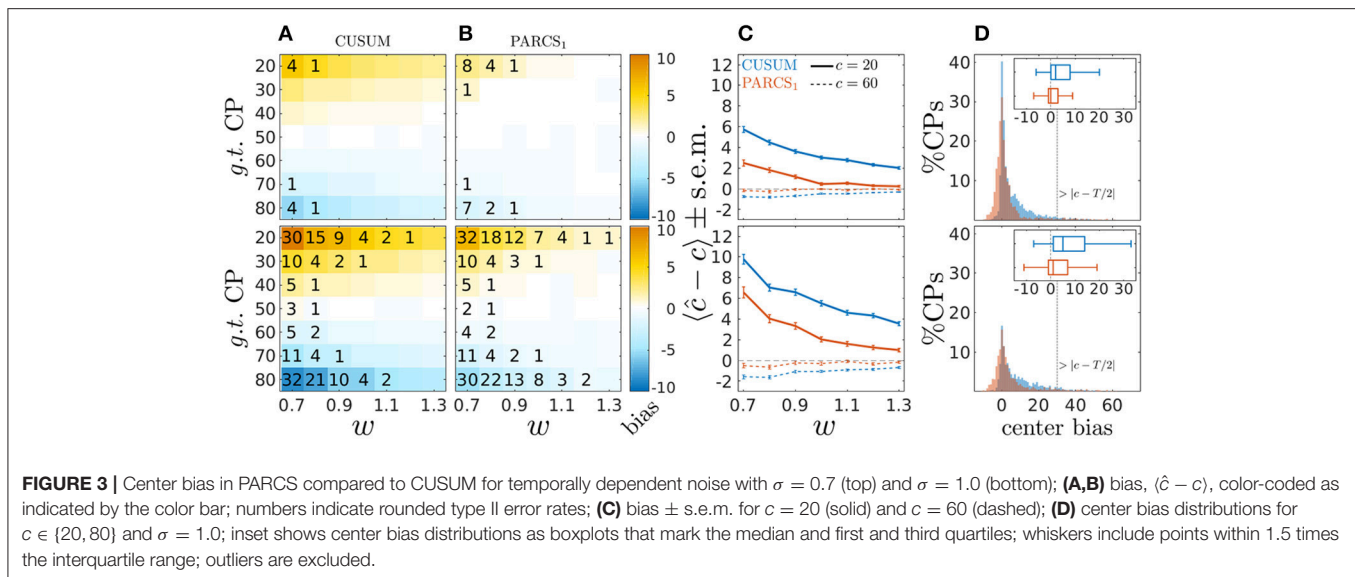


that the *receiver operating characteristic* (ROC) curve for PARCS, depicting the method’s false discovery rate against its power for different nominal α levels, consistently lies above that of CUSUM. For estimating the statistical power of each method, 1,000 white Gaussian noise realizations with one CP at a random location in the range $[20, 80]\%T$ (and $w = \sigma = 1.0$) are simulated and type II error rates at different nominal α levels are computed. This ROC analysis indicates that for every nominal α level for CUSUM, there exists at least one nominal α for PARCS such that PARCS has both higher power (fewer type II errors) and lower false detection rate (fewer type I errors), making it the preferable method. This point is explored in more detail later in the context of multiple CP detection in Section 3.2. For shorter time series, PARCS behaves similarly as for longer time series in our simulations, while for CUSUM type I error rates now start to fall below the nominal α level as well (**Figures 2F–H**). The area under the ROC curves become smaller for shorter time series

for both methods, but the ROC curve of PARCS consistently lies above that of CUSUM in those cases as well (**Figures 2J–L**).

Next, we examine the behavior of the tests with dependent noise. In the case of temporally dependent noise, an appropriate block size for the bootstrap procedure could be determined by inspecting the autocorrelation function of x_0 (see Equation 5 and **Algorithm 2**). One thousand noise realizations of length $T = 100$ are drawn from an order-2 MA process with coefficients $\kappa_1 = -0.5/\sigma$ and $\kappa_2 = 0.4/\sigma$. Since increasing noise variance in the temporally dependent case is not equivalent to decreasing the step parameter, we repeat the analysis with the same parameters from the temporally independent case above but varying the step parameter, $w \in \{0.7, 0.8, \dots, 1.3\}$ and considering two levels of Gaussian noise, $\sigma \in \{0.7, 1.0\}$.

Figure 3 shows results of the comparison for $\sigma = 0.7$ (top row) and $\sigma = 1.0$ (bottom row). Similar to the white noise case, the CUSUM method’s center bias increases for smaller



signal-to-noise ratio (smaller w), and PARCS, in comparison, consistently reduces center bias. For the same nominal α level, PARCS misses more of the true CPs than CUSUM for peripheral CPs when $\sigma = 0.7$ (top panels of **Figures 3A,B**), but the type II error rate of the two methods is more comparable in the high noise case, $\sigma = 1.0$ (bottom panels of **Figures 3A,B**), despite the PARCS method's more conservative behavior (far lower type I error rates) in this setting as well. A summary comparison between the two methods is shown in **Figure 3C** for two exemplary CP locations, $c \in \{20, 60\}$. Despite a significant reduction when using PARCS, both center bias distributions in the 62% of realizations with a CP identified by the two methods, and with the two hardest parameter settings ($c \in \{20, 80\}$, $\sigma = 1.0$ and $w = 0.7$) remain strongly skewed (bottom panels of **Figure 3D**).

So far, we compared PARCS to the CUSUM statistic with $\gamma = 0$. It is intuitive when developing PARCS to choose $\gamma = 0$ for the CUSUM transformation in Equation 2, since this directly corresponds to the numerical integral of the time series upon which the PARCS approach is based (but see Section 4). Besides, under certain conditions, the CUSUM method using the test statistic with $\gamma < 0.5$ is more sensitive than that with $\gamma = 0.5$ (Antoch et al., 1995). However, the CUSUM statistic with $\gamma = 0.5$ returns the maximum likelihood estimator of CP location in an AMOC scenario when noise in the step model of Equation 1 is i.i.d. and normally distributed, leading theoretically to the strongest center bias reduction under those conditions (Antoch and Hušková, 2001). We therefore also compare PARCS to this maximum likelihood CUSUM estimator here, henceforth referred to as CUSUM_{ML}.

Univariate time series of length $T = 100$ are simulated according to the step model in Equation 1 with different ground truth CP locations, $c \in \{20, 30, \dots, 80\}$. We consider only the scenario with largest white Gaussian noise variance, $\sigma = 1.0$, in this analysis, for which PARCS showed the largest center bias. A single CP was identified by using the CUSUM_{ML} method and estimating the PARCS₁ model, both followed by bootstrap

significance testing. Other parameters are as in the previous analyses above. CUSUM_{ML} results in a significant reduction in center bias compared to PARCS in three of the most peripheral ground truth CPs, $c \in \{20, 30, 80\}$, but does not eliminate it completely (see **Figure 4A**). For the same nominal α level, CUSUM_{ML} also shows lower type II error rates compared to PARCS for these CPs as reported in **Table 1**, but recall that PARCS has a far lower type I error rate than CUSUM for the same choice of nominal α (cf. **Figures 2E–H** and Kirch, 2007). The two methods are comparable in the quality of their detections for all other ground truth CP locations, $c \in \{40, \dots, 70\}$, with PARCS having a slight advantage.

In order to assess how well the two methods fare in the small sample size limit, and to characterize the convergence behavior of the bootstrap procedure in each method, we repeat the same analysis for shorter series lengths, $T \in \{50, 26\}$. Ground truth CPs are set to the same relative location within the time series as in the $T = 100$ simulations. As summarized in **Table 1**, detection rates deteriorate as series length decreases, as does the bias relative to series length (where the relative location within the series with respect to the periphery is more relevant than the absolute CP location; see **Figures 4B,C**). Especially for $T = 26$, PARCS performs mostly better than CUSUM_{ML}, giving higher detection rates (see **Table 1**) and smaller center bias (**Figure 4C**) in the majority of ground truth CPs, although it is still more conservative with near 0% type I error rate (given the bootstrap resolution; see **Figure 2G**). As we show next, this is a particularly important advantage of PARCS over the CUSUM-based methods when detecting multiple CPs, since CUSUM-based techniques rely on dissecting the time series into smaller segments in this case, reducing sample size at each iteration.

3.2. Detecting Multiple CPs in Univariate Data

For the scenario with multiple CPs, we assess the performance of PARCS in comparison to the CUSUM method with standard binary segmentation (Scott and Knott, 1974;

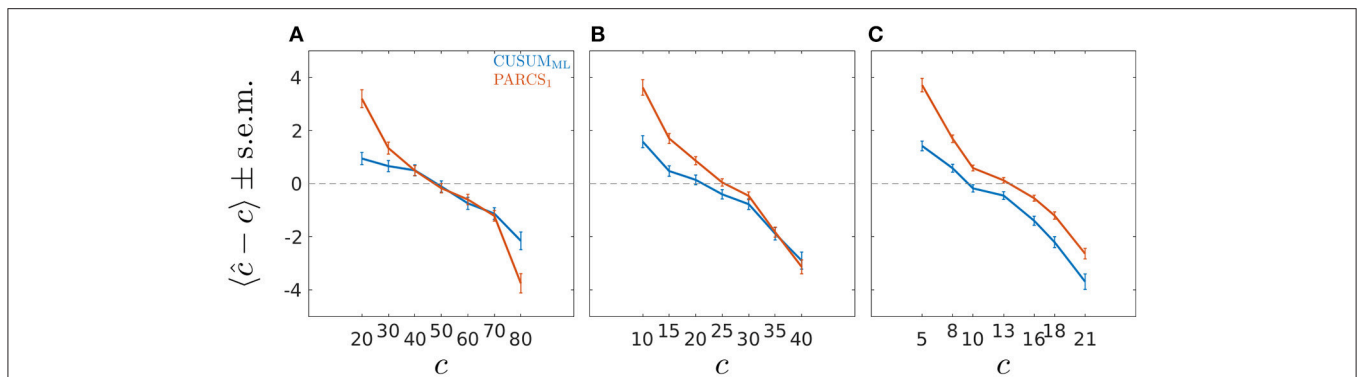


FIGURE 4 | Bias \pm s.e.m. in PARCS compared to CUSUM_{ML} with time series of length **(A)** $T = 100$, **(B)** $T = 50$, and **(C)** $T = 26$; noise is temporally independent with $\sigma = 1.0$.

Bai, 1997) for univariate data with white Gaussian noise. Standard binary segmentation is known to mislocate CPs in some scenarios, but modifications to the segmentation procedure have been proposed for solving this problem (Fryzlewicz, 2014). We show that PARCS provides an alternative approach. We then discuss a fundamental practical problem in statistical testing when using segmentation methods in general that is avoided by PARCS. Through comparison with standard binary segmentation, we illustrate conditions under which using such methods becomes infeasible. We then consider temporally dependent noise in univariate time series.

The *binary segmentation* method (Scott and Knott, 1974; Bai, 1997) for detecting multiple CPs proceeds as follows (pseudocode can be found in Fryzlewicz, 2014): If, according to a CUSUM test criterion, a CP \hat{c}_1 is detected over the full time series, the series is partitioned at \hat{c}_1 . The procedure is repeated on the resulting left and right segments, potentially returning two additional CPs, \hat{c}_{21} and \hat{c}_{22} , respectively. The procedure is terminated when no more CPs are detected after subsequent partitioning. Similar to the single CP case, we use bootstrap testing in deciding the significance of a CP at each stage. In the present context, we will refer to this CUSUM-based binary segmentation method simply by “CUSUM”.

Three different processes with white Gaussian noise, $\sigma = 1.0$, of the form given by Equation 6 and of length $T = 100$ are simulated for 1,000 realizations each. A baseline $b = 0$ and two CPs at time steps $c_1 = 20$ and $c_2 = 60$ are set in all three scenarios. Weights (w_1, w_2) are set to $(1, 2)$, $(2, -1)$, and $(2, 1)$ (see insets in top row of Figure 5). For the present comparison, binary segmentation is terminated after at most one partitioning, as this completely suffices to compare the methods (note that this allows CUSUM to detect up to three potential CPs, with only two present in the series). Similarly, the PARCS₃ model is estimated, and both methods use the corresponding permutation bootstrap test with $B = 10,000$ and $k = 1$. In order to compare type I and type II error rates between the two methods, we set nominal α levels to 0.05 and 0.30 for CUSUM and PARCS, respectively, which is expected to return about the same factual type I error

TABLE 1 | Type II error rates in PARCS compared to CUSUM_{ML} for different lengths of the time series; underline, method with higher detection rate; nominal α level, 0.05 for both methods.

T	Method	Type II error rate							
		$c = \text{round} ($	20	30	40	50	60	70	80
100	CUSUM _{ML}	<u>07</u>	<u>03</u>	02	01	01	<u>03</u>	<u>12</u>	
	PARCS	17	03	<u>01</u>	<u>01</u>	<u>01</u>	03	16	
50	CUSUM _{ML}	<u>30</u>	<u>20</u>	16	13	16	26	<u>37</u>	
	PARCS	44	22	<u>12</u>	<u>08</u>	<u>10</u>	<u>19</u>	41	
26	CUSUM _{ML}	<u>53</u>	<u>38</u>	35	32	38	44	59	
	PARCS	68	41	<u>32</u>	<u>24</u>	<u>29</u>	<u>37</u>	<u>58</u>	

rates of 5% for both methods in series of length $T \in [26, 100]$, according to Figures 2E–G.

A first look at Figure 5 suggests that both CUSUM and PARCS detect CPs that are close to the ground truth. A more detailed comparison with respect to type I and type II error rates and the quality of CP detections is provided in Table 2. The quality of detections using accuracy scores is defined as the correct detection rate within a $\pm 5\%T$ range from the ground truth CP location, adjusted for type I errors by an additive term of $-\hat{\alpha}/M$, where $\hat{\alpha}$ is the factual (empirical) α level. This way, the accuracy score is an overall performance measure that takes into account both type I and type II errors, and how far off the detected CP is from the true one.

The first scenario (left panels in Figure 5) has the hardest parameter setting, since c_1 is both more peripheral and smaller in magnitude than c_2 . CUSUM first detects the easier CP, followed by detecting c_1 at the left hand segment as \hat{c}_{21} , but with a lower accuracy score as confirmed in Table 2. Similarly, PARCS returns c_2 and c_1 as the first and second rank CPs, respectively, with accuracy scores higher than those of CUSUM. The relatively low accuracy scores at detecting c_1 in both

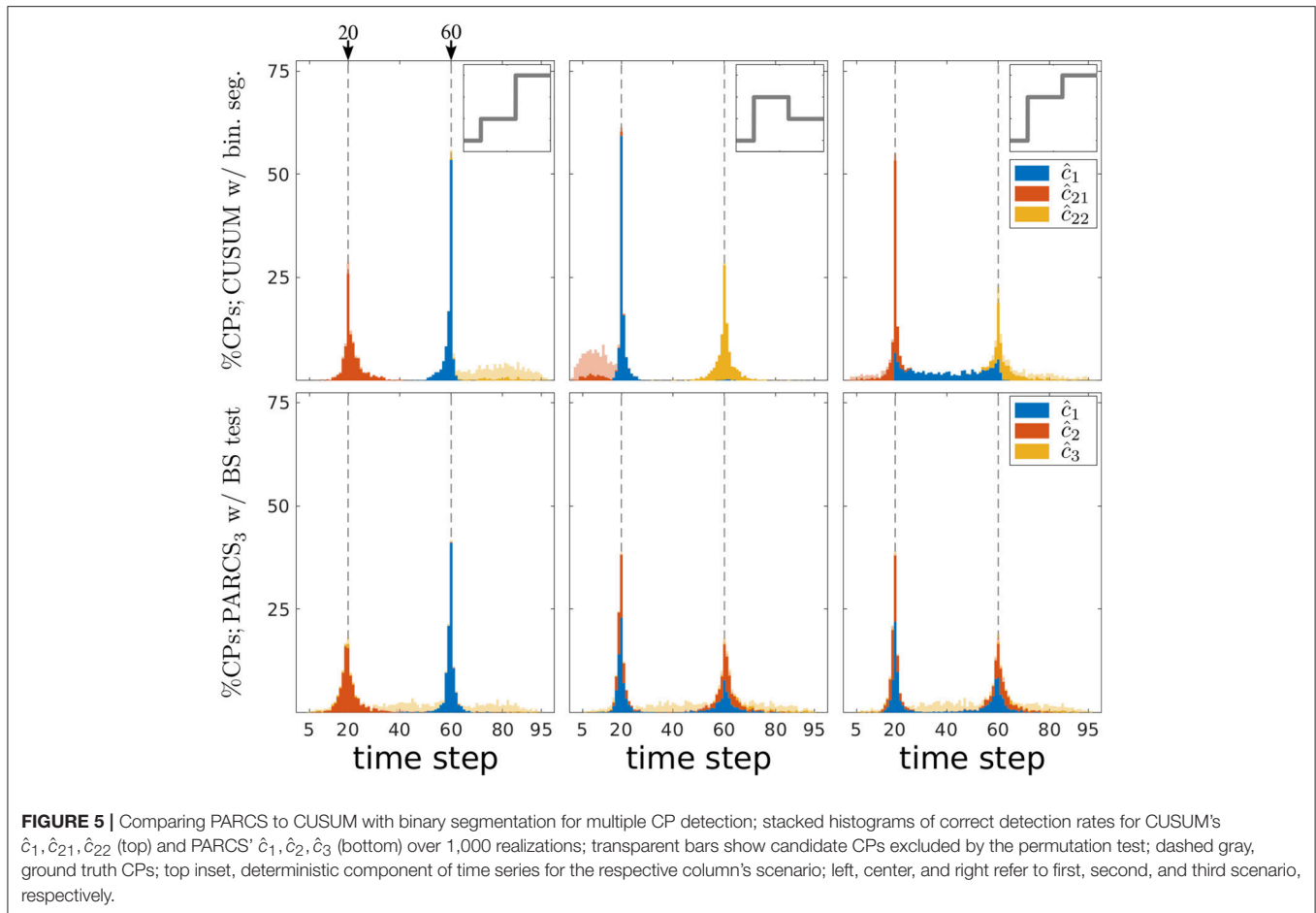


FIGURE 5 | Comparing PARCS to CUSUM with binary segmentation for multiple CP detection; stacked histograms of correct detection rates for CUSUM’s $\hat{c}_1, \hat{c}_{21}, \hat{c}_{22}$ (top) and PARCS’ $\hat{c}_1, \hat{c}_2, \hat{c}_3$ (bottom) over 1,000 realizations; transparent bars show candidate CPs excluded by the permutation test; dashed gray, ground truth CPs; top inset, deterministic component of time series for the respective column’s scenario; left, center, and right refer to first, second, and third scenario, respectively.

TABLE 2 | Comparing PARCS to CUSUM with binary segmentation for multiple CP detection for different lengths of the time series; error rates and accuracy scores are rounded; triplet, scenarios 1 / 2 / 3; underline, method with lower error rate or higher accuracy score; nominal α levels, 0.05 and 0.30 for CUSUM and PARCS, respectively.

T	Method	Error rate		Accuracy score	
		type I	type II	c = 20%T	c = 60%T
100	CUSUM	10 / 14 / 41	08 / 02 / 01	72 / 92 / 77	91 / 78 / 45
	PARCS	<u>02 / 03 / 02</u>	<u>04 / 00 / 01</u>	<u>80 / 96 / 95</u>	<u>96 / 74 / 76</u>
50	CUSUM	12 / 18 / 19	27 / 21 / 11	34 / 56 / 61	74 / 38 / 16
	PARCS	<u>04 / 04 / 03</u>	<u>13 / 02 / 05</u>	<u>51 / 82 / 82</u>	<u>85 / 52 / 52</u>
26	CUSUM	12 / 22 / 18	41 / 44 / 16	28 / 44 / 71	77 / 26 / 25
	PARCS	<u>06 / 07 / 06</u>	<u>24 / 09 / 10</u>	<u>37 / 75 / 76</u>	<u>79 / 47 / 47</u>

methods are due to its peripheral location and small magnitude. Both type I and type II error rates are markedly lower in PARCS.

The second and third scenarios (center and right panels in **Figure 5**, respectively) are easier in terms of ground truth parameter settings, since the more peripheral CP c_1 has the

larger magnitude. In the second scenario (center panels in **Figure 5**), the two methods are comparable with regards to their overall accuracy scores as defined above (many detections lie outside the $\pm 5\%T$ accuracy score range, especially for c_2), but PARCS has the lower type I and type II error rates. CUSUM has a higher rate of false discoveries than in the first scenario. Its first detection is the higher magnitude CP, which comes with a lower accuracy than PARCS, followed by a detection at the right hand segment with a higher accuracy than PARCS. The relatively low accuracy rates for detecting c_2 in both methods are due to its small magnitude. The third scenario (right panels in **Figure 5**) is an example of a setting in which standard binary segmentation may fail in correctly allocating CPs (see Fryzlewicz, 2014, for a binary segmentation approach that solves this problem). While the performance of PARCS remains about the same as in the second scenario, CUSUM’s first detection diverges from either of the two ground truth CPs in a large number of realizations (see top-right panel in **Figure 5**). The large type I error rate (more than three times that in the first and second scenarios) markedly reduces the accuracy scores, which are substantially lower than for PARCS for both ground truth CPs (see **Table 2**). Another factor behind the high type I error rates of any iterative procedure including binary segmentation is that the same CP could be detected

again at later iterations when an earlier detection is slightly biased.

We now compare the impact of shorter series on the performance of binary segmentation methods and PARCS. We consider 1,000 noise realizations from each of the three scenarios with $T \in \{50, 26\}$. Two ground truth CPs are set to the same relative location within the time series as for $T = 100$. As seen in **Table 2**, both type I and type II error rates increase in both methods with the decrease in series length, with the exception of type I error rates for CUSUM in the third scenario. PARCS is consistently the superior method, having both higher statistical power and less false discoveries than CUSUM. While accuracy scores predictably decrease with shorter series length, comparison between the two methods remains qualitatively similar to the $T = 100$ case in the first and third scenario, while there is a marked change in the second scenario: While CUSUM is slightly superior in accurately detecting c_2 for $T = 100$, PARCS progressively surpasses CUSUM in accuracy as series length decreases. This behavior is a result of deterioration in the power of the bootstrap test statistic for shorter series. Not only does the overall sample size decrease, but CUSUM in the second scenario with $T = 26$ is tasked after a potential first detection of c_1 with bootstrapping the CUSUM test statistic on segments as short as 4 or 5 time steps only, which is statistically infeasible, either when using bootstraps or approximate parametric tests (Olshen et al., 2004; Fryzlewicz, 2014; Cho and Fryzlewicz, 2015). We stress that this is a fundamental drawback to any method that relies on partitioning, and is not specific to standard binary segmentation.

The limitations of binary segmentation methods become more obvious when noise is temporally dependent. For instance, given a time series of length $T = 100$ with parameters as in the second scenario and an order-2 MA noise process, blocks of size $k \simeq 3$ are required for proper block-permutation. If CUSUM first detected $c_1 = 20$ accurately, $\hat{c}_1 = 20$, the left hand segment would be only 20 time steps long. This allows for only 7 blocks, yielding 5040 possible permutations. This number drops to 720 permutations had \hat{c}_1 been detected only 2 time steps further to the left, which makes it hard to approximate the EDF of the CUSUM statistic reliably. In addition, specifying the block size first requires estimating the MA process order by approximating the H_0 -conform time series (see **Algorithm 2**), but potential CPs are not known *a priori*, due to the recursive nature of binary segmentation methods.

Given these considerations, we focus on PARCS only as we now move over to the case of detecting multiple CPs in series with temporally dependent noise. 1,000 noise realizations of length $T = 100$ are drawn from an order-2 MA process with $\sigma = 0.7$ and coefficients $\kappa_1 = -0.5/\sigma$ and $\kappa_2 = 0.4/\sigma$. Other parameters are as in the previous analysis. In **Figure 6**, we show distributions of correct detections for the second scenario only (with the other two scenarios qualitatively comparable to their counterparts in **Figure 5**), but error and accuracy rates on these scenarios are reported as well. After removing the three step changes following PARCS₃ CP detection, the majority of residual time series (70% as shown in bottom panel in **Figure 6A**) had an autocorrelation that cuts off at the correct order of the

ground truth MA(2) noise process, i.e., with $\text{acorr}(\mathbf{x}_0; 2)$ being the last coefficient that lies outside the 95% confidence bounds (dashed lines; top panel in **Figure 6A**; results for the first and third scenarios are comparable). Block-permutation bootstrap testing with nominal $\alpha = 0.05$ and $B = 10,000$ is carried out on these series with blocks of size 3 and, for other time series, according to the estimated order in **Figure 6A** (with an upper bound of 10 on block size). Exactly two CPs are detected in more than 99.5% of realizations in all scenarios. **Figure 6B** shows that the distribution of correct detections for the second scenario is largely concentrated around the ground truth CPs. Accuracy scores in each scenario are, respectively, 96, 99, and 99% for c_1 and 99, 89, and 89% for c_2 . Note also the oscillation in the example realization in **Figure 6C**, which results from dependent noise with a negative MA coefficient κ_1 .

3.3. Detecting Multiple CPs in Multivariate Data

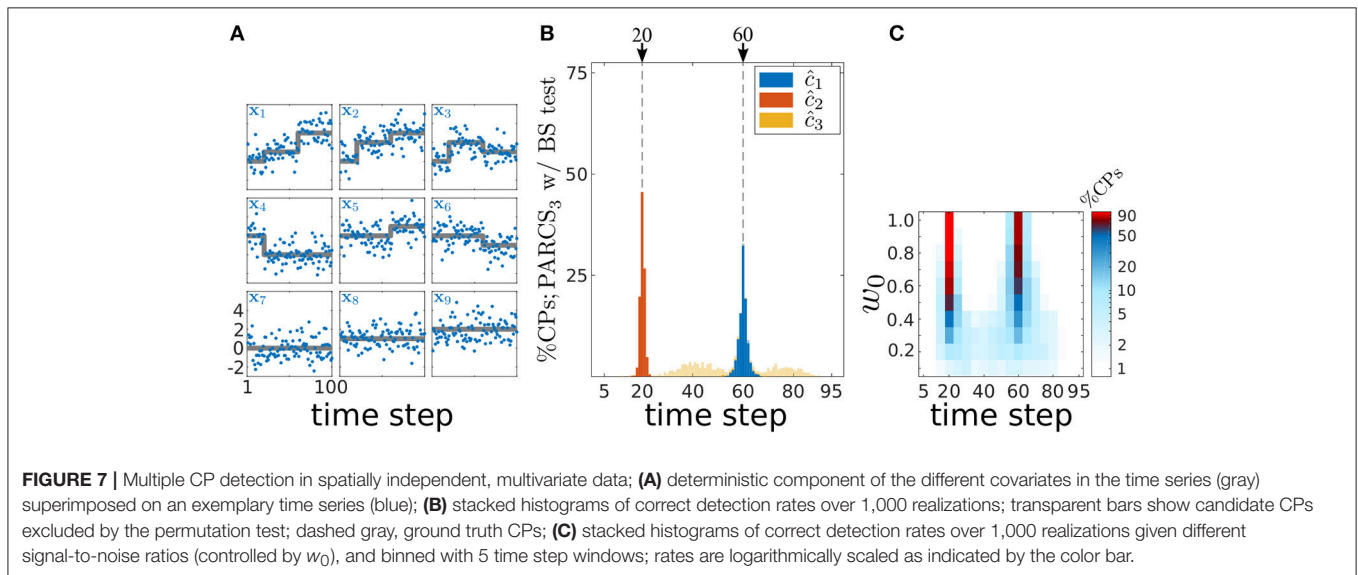
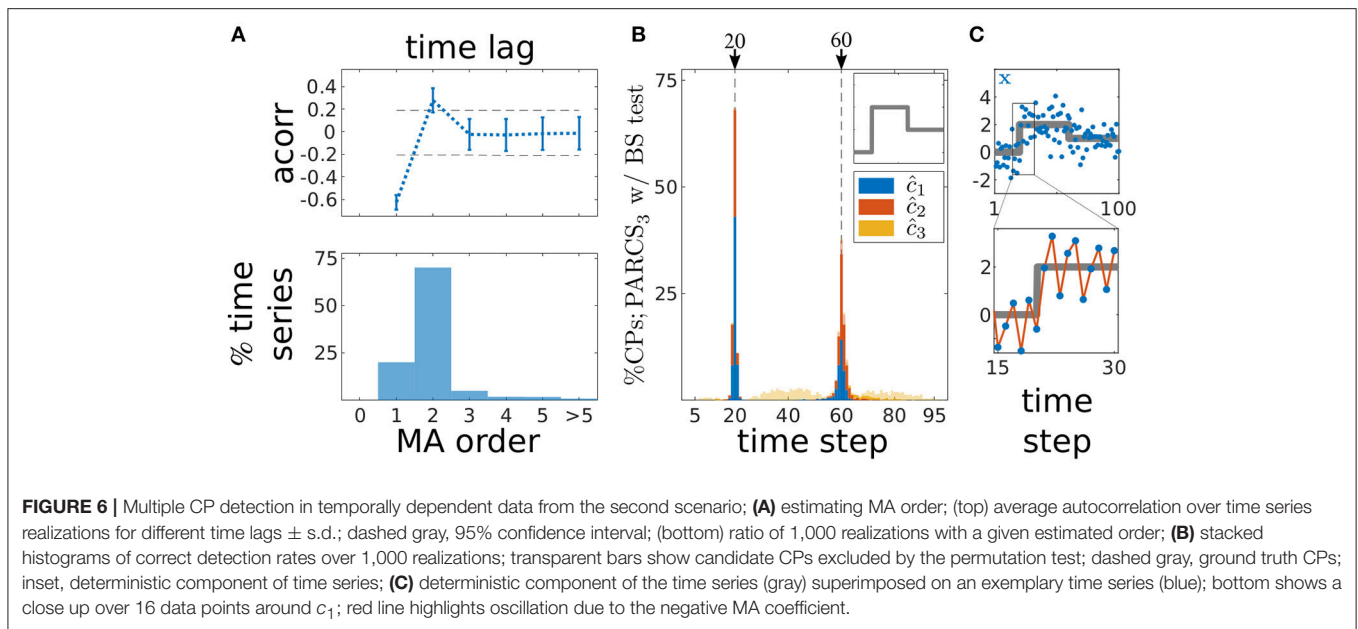
The PARCS method's ability to detect multiple CPs in spatially independent, multivariate time series is demonstrated in **Figure 7** on 1,000 realizations of length $T = 100$ with $N = 9$ covariates and white Gaussian noise, $\sigma = 1.0$. Parameters are set to $\mathbf{b} = (0, 0, 0, 2, 2, 2, 0, 1, 2)$, $c_1 = 20$, $\mathbf{w}_1 = w_0 \cdot (1, 2, 2, -2, 0, 0, 0, 0, 0)$, $c_2 = 60$ and $\mathbf{w}_2 = w_0 \cdot (2, 1, -1, 0, 1, -1, 0, 0, 0)$. The scaling parameter w_0 controls signal-to-noise ratio in the time series and is initially set to 1.0. Given these parameter values, the two CPs are not represented in all covariates of the time series, as exemplified in **Figure 7A**, rendering CPs harder to detect from the averaged univariate time series (with steps differing in sign across the covariates partially canceling each other, resulting in small weights, $\langle \mathbf{w}_1 \rangle = 3/9$ and $\langle \mathbf{w}_2 \rangle = 2/9$, for the resulting univariate time series).

Following PARCS₃ CP detection, augmented by a bootstrap test with nominal $\alpha = 0.05$, $B = 10,000$ and $k = 1$, exactly two CPs are detected in 99.9% of realizations. Accuracy scores are 99.8 and 98% for c_1 and c_2 , respectively. The lower variance in c_1 detections, as seen in **Figure 7B**, is due to the higher average absolute weight $\langle |\mathbf{w}_1| \rangle = 7/9$ compared to $\langle |\mathbf{w}_2| \rangle = 6/9$.

We then test the method's performance for smaller signal-to-noise ratios with $w_0 \in \{1.0, 0.9, \dots, 0.1\}$. **Figure 7C** shows that correct detection rates within a $\pm 2\%T$ range from the ground truth CPs for different values of w_0 remain above 50%, even for magnitudes as small as $w_0 = 0.5$. These rates are a result of PARCS leveraging CP information from multiple covariates simultaneously, rather than depleting the signal through averaging.

3.4. Detecting Neural Events That Reflect Learning

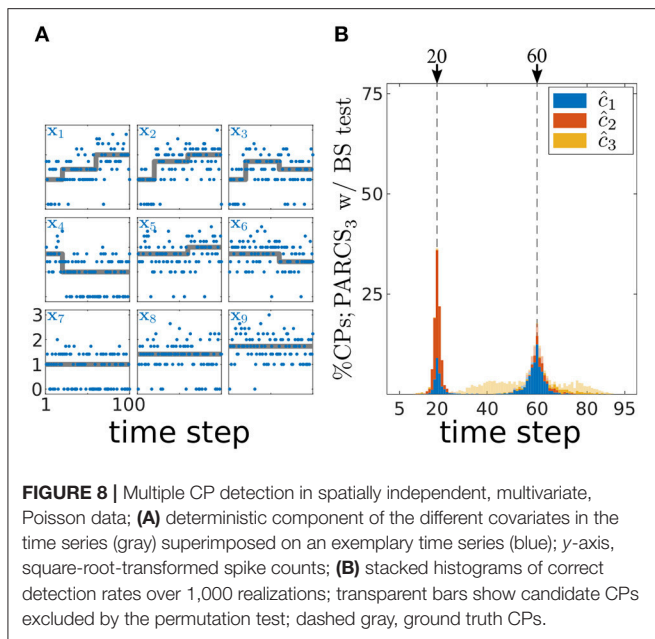
A previous study by one of the authors and colleagues exemplifies the practical value of change point detection in neuroscience (Durstewitz et al., 2010). These authors demonstrated that acquiring a new behavioral rule in rats is accompanied by sudden jumps in behavioral performance, which in turn is reflected in the activity of neural units recorded simultaneously in the medial



prefrontal cortex (mPFC). In the current section, we revisit part of these data to showcase PARCS in a real data scenario.

Before moving to the demonstration, it is important to note that the data in question are not normally distributed and potentially include linear trends (Durstewitz et al., 2010) not accounted for by the step models in Equations 1, 6, and 9. As such, some preprocessing may be necessary for a statistical analysis that is more consistent with the step model assumptions (this may include detrending and potentially some mild smoothing with Gaussian kernels; see Durstewitz et al., 2010). However, to keep the present demonstration simple, PARCS was applied directly to the data with minimal preprocessing, which only involves square-root-transforming the neural count data for bringing them closer to a Gaussian distribution and stabilizing the variance (Kihlberg et al., 1972).

In order to show that PARCS can still return reasonable CP estimates under these non-Gaussian conditions, we first test its performance on simulated spike count data, before applying it to the empirical data. We simulate 1,000 realizations of length $T = 100$ with $N = 9$ covariates according to a Poisson process. Parameters are set to $\mathbf{b} = (1, 1, 1, 3, 3, 3, 1, 2, 1)$, $c_1 = 20$, $\mathbf{w}_1 = (1, 2, 2, -2, 0, 0, 0, 0, 0)$, $c_2 = 60$ and $\mathbf{w}_2 = (2, 1, -1, 0, 1, -1, 0, 0, 0)$. This choice of parameters results in average firing rates that are comparable in their means to the white Gaussian noise case (cf. Figures 7A, 8A) and to the low firing rates often observed in mPFC neurons. One obvious diversion from Gaussian assumptions in the case of Poisson noise is that the variance is not constant anymore, but is equal to the means within each of the segments separated by true CPs. Following square-root-transforming the data and PARCS₃ CP



detection, augmented by a bootstrap test with nominal $\alpha = 0.05$, $B = 10,000$ and $k = 1$, exactly two CPs are detected in 92% of realizations. Accuracy scores are 98 and 70% for c_1 and c_2 , respectively. The lower accuracy scores compared to the white Gaussian noise scenario are due to the lower signal-to-noise ratios, resulting from the increase in noise variance with firing rates (cf. **Figures 7B, 8B**). Nevertheless, these results sufficiently justify the use of PARCS in the present context.

We now turn to the experimentally obtained dataset. Six animals were trained on a two-choice deterministic operant rule switching task which proceeds as follows: At the beginning of the session, the animal follows a previously acquired behavioral rule whereby it responds to a visual cue by a lever press for attaining a reward (visual rule). Unknown to the animal, reward contingencies are switched after 20 trials to a novel spatial rule, in which attaining the reward requires pressing a certain baited lever (right or left), regardless of the visual cue. The session is terminated when the animal reaches a preset criterion that indicates that the new rule behavior has been learnt. In addition to the binary behavioral data of lever presses over trials, spike counts emitted by mPFC units during the 3 seconds following cue onset were collected through single unit recording techniques. Neural and behavioral data from one animal are shown in **Figures 9A,B**, respectively. Trials corresponding to the steady state visual and spatial rule (first and last 20 trials, respectively) are not considered in the analysis.

Time series with one or two significant CPs were described in the original study by Durstewitz et al. (2010), so PARCS₂ models are estimated for each animal for both the multivariate neural data (multiple response PARCS model; **Figure 9A**) and the univariate behavioral data (**Figure 9B**), in addition to PARCS₁ models for the behavioral data as summarized in **Figure 9C**. As shown in **Figure 9B**, one neural CP in that exemplary animal matches its behavioral counterpart. The second neural

CP, while not as close to its behavioral counterpart, is only 7 trials apart, and the two are highly correlated across animals, as shown in **Figure 9C**, concurring with the original findings of Durstewitz et al. (2010). Besides the significant correlation, the corresponding black linear regression line lies very close to the diagonal, indicating that neural and behavioral CPs are not only correlated, but are almost equal. Moreover, those authors report that data from many animals contain at most a single CP (also note the low weight of one of the CPs as estimated from one of the animals using PARCS₂). Comparing the PARCS₁ behavioral CP to its neural counterpart (blue circles in **Figure 9C**) shows that correlation remains high and significant. A sample size of $s > 5$ is usually recommended for evaluating the significance of a correlation. However, the corresponding linear regression line is also close to the diagonal, in further support for the reliability of this result, and in agreement with the original results despite different procedures: For the neural data in the original study, CUSUM-based detection was performed on a multivariate discrimination statistic defined across the whole neural population, while here, the model was determined directly from the multiple spike count data.

4. DISCUSSION

In the current article, we introduced PARCS, a method for detecting multiple step changes, or CPs, in potentially multivariate, temporally dependent data, supported by a bootstrap-based nonparametric test. We also showed that PARCS substantially reduces center bias in estimating CPs compared to the most basic specification of the CUSUM method, and presented conditions under which it compares to or outperforms the maximum-likelihood CUSUM statistic. Furthermore, we demonstrated that PARCS may achieve higher sensitivity (statistical power) than CUSUM-based methods while at the same time having lower type I errors in multiple CP scenarios, mainly because PARCS can make use of the full time series while CUSUM-based methods rely on segmenting the time series for detecting multiple CPs. We finally confirmed previous results pertaining to the acquisition of a new behavioral rule and the role of the medial prefrontal cortex in this process.

As already apparent from some of our simulation studies, the basic PARCS method as introduced here leaves room for improvement. In the presence of a single CP, we showed that PARCS strongly reduces the amount of bias toward the center that results from the direct application of the most basic form of the CUSUM locator statistic. Theoretically-grounded modifications to the CUSUM transformation that reduce this amount of bias rely on down-weighting more centrally-located points (Kirch, 2007). As shown with PARCS, this problem is not quite as severe. Nevertheless, since PARCS approximates the CUSUM transformation using a regression model, similar down-weighting could be incorporated into the PARCS procedure as well by using *weighted least squares* instead of regular least squares (Hastie et al., 2009), which is a straightforward amendment. Furthermore, the PARCS method currently requires a liberal guess of the number M of CPs in advance, followed

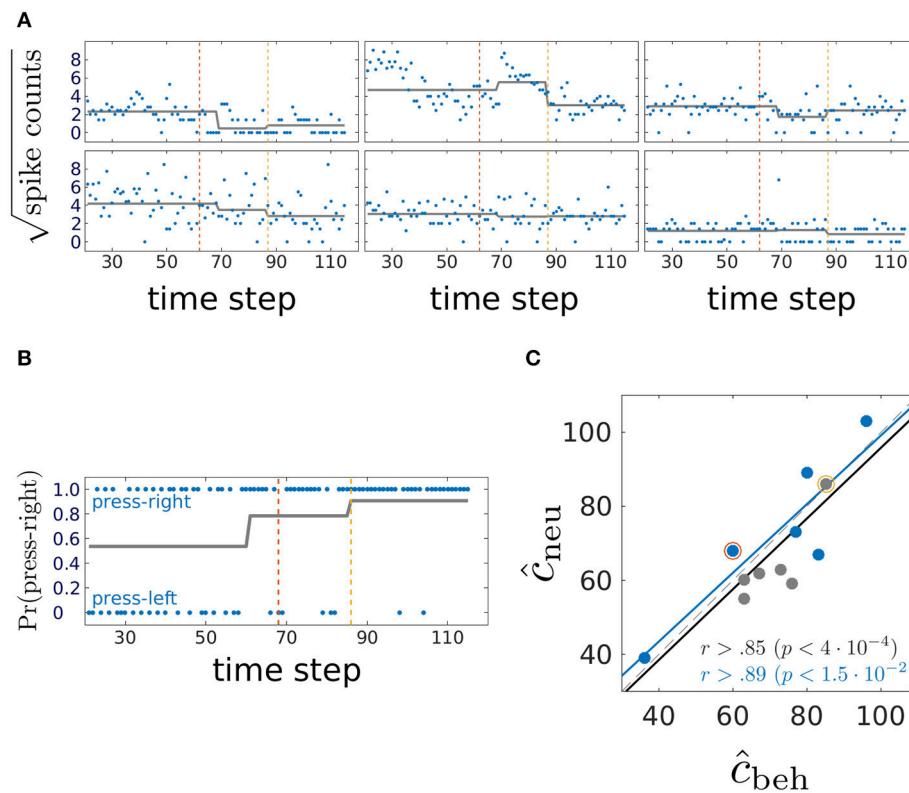


FIGURE 9 | Comparing behavioral and mPFC neural CPs; **(A)** blue, square-root-transformed spike count data in the three seconds following cue onset from 6 representative mPFC units of one rat; gray, mean as estimated by inverting the neural multiple response PARCS₂ model. Note potential CP in top-center unit which was not detected by PARCS₂ since it did not contribute strongly to population-wide CPs; dashed lines, behavioral CPs from the same animal; **(B)** blue, lever press at each trial; this animal is rewarded for pressing the right lever during the spatial rule; gray, probability of pressing right lever as estimated by inverting the behavioral PARCS₂ model; dashed lines, neural CPs from the same animal (see **A**); **(C)** relating behavioral and neural CPs; blue, behavioral CPs with higher weight; r , correlation coefficients as computed over all 12 data points (black) and over those where behavioral CPs have the higher weight (blue); p -values, significance levels of corresponding r ; black and blue lines, respective least-square linear regression fits to the two sets of data points; red and yellow circles, neural and behavioral CP pairs from the exemplary animal in **(A,B)**, respectively.

by refinements through nonparametric bootstrap testing. It is desirable, however, especially when no prior information on M is available, to have statistical tests as termination criteria for the forward and backward stages. In adaptive regression spline methods (Friedman and Silverman, 1989; Friedman, 1991; Stone et al., 1997), there is strong empirical evidence (Hinkley, 1969, 1971b) backed by theoretical results (Feder, 1975) that the difference in residual mean-square-error between two nested models that differ in one additional knot is well approximated, albeit conservatively, by a scaled χ^2 statistic on 4 degrees of freedom (Friedman, 1991). This led to one nonparametric termination recipe that is based on generalized cross-validation (Craven and Wahba, 1978). Another approach is to infer the piecewise linear regression model with the aid of a parametric test for specifying the number and location of knots, without recourse to iterative procedures (Liu et al., 1997). Unfortunately, neither approach is directly applicable to PARCS, since they both require assumptions that are not met in the CUSUM-transformed time series. The CUSUM transformation of the time series is a nonstationary ARMA(1, q) process. Deriving

reasonable generalized cross-validation (Craven and Wahba, 1978; Friedman and Silverman, 1989; Friedman, 1991), F-ratio (Hastie et al., 2009; Durstewitz, 2017) or parametric (Liu et al., 1997) test statistics require currently unknown corrections to those tests which account for nonstationarity and the particular form of the ARMA model underlying the CUSUM-transformed data.

When multiple CPs are present in the data, PARCS can outperform standard binary segmentation (Scott and Knott, 1974; Bai, 1997). Other segmentation methods also solve the problem of mislocating CPs inherent in the standard procedure (Olshen et al., 2004; Fryzlewicz, 2014; Cho and Fryzlewicz, 2015). Wild binary segmentation (WBS; Fryzlewicz, 2014), for instance, relies on sampling local CUSUM transformations of randomly chosen segments of the time series. The candidate CP with the largest value among sampled CUSUM curves is returned to be tested against a criterion, followed by binary segmentation. WBS is preferable to PARCS in that its test statistic and termination criterion when noise is independent are backed up by rigorous theory, and may be the favorable method when

segments are large enough for the test statistic to converge. If series of only limited length are available, however, WBS may run into similar problems as standard binary segmentation for CUSUM, since each detection is still followed by partitioning the data further. WBS also, to the best of our knowledge, currently lacks a thorough analysis on the behavior of its test statistic for dependent data. It is tempting to speculate on the potential for a hybrid approach that capitalizes on the desirable features of both methods. Computational demands arise in WBS from the need to choose segment range parameters by sampling few thousand CUSUM curves to which PARCS may offer an easy and efficient workaround: Fryzlewicz (2014) demonstrated that the optimal WBS segment choice is the segment bounded by the two CPs closest to the target CP from each side. PARCS could thus provide an informed selection of boundaries by returning candidate CPs in the data and use these to demarcate segments, rather than random sampling as in WBS.

In dealing with multivariate data, recent methods tackled the computational demands of having a large number of covariates and sparse CP representations (Cho and Fryzlewicz, 2015; Wang and Samworth, 2018). These methods rely on low dimensional projections of the multivariate CUSUM curve that preserve the CPs and follow this projection by a binary segmentation method. Since PARCS for multivariate time series is also based on the CUSUM transformation, it is straightforward to leverage the computational savings provided by such projection methods in reducing the dimensionality of the PARCS input, while avoiding the drawbacks of binary segmentation methods. This may offer a route for extending PARCS to the important case of multivariate CP detection in mutually dependent time series with spatial dependence, a configuration which these projection methods also consider (Cho and Fryzlewicz, 2015; Wang and Samworth, 2018). Alternatively, nondiagonal covariance structure in multivariate series may be accounted for by extending the PARCS formulation to the multivariate regression spline realm (Friedman, 1991; Stone et al., 1997).

Finally, when analyzing the neural and behavioral data during the rule switching task, we mentioned that data may also contain trends that are not accounted for by step change time series models (Durstewitz et al., 2010). Caution must be made when analysing real data using CP detection methods in that these methods, PARCS included, assume a step change model

underlying the generation of the data and hence may attempt to approximate trends and other nonstationary features by a series of step changes, a point made more explicit by Fryzlewicz (2014) (Durstewitz et al., 2010, therefore removed trends around candidate CPs first). Hence, to avoid wrong conclusions with respect to the source and type of nonstationarity in experimental time series, it may be necessary to either augment change point detection by adequate preprocessing (Durstewitz et al., 2010) or to generalize time series models for CP detection to include other forms of nonstationarity.

DATA AVAILABILITY STATEMENT

The simulated data supporting the conclusions of this manuscript will be made available by the authors, without undue reservation, to any qualified researcher. MATLAB code for PARCS, together with demos, is freely available at <https://github.com/htoutounji/PARCS>. The experimental data analyzed in this study were obtained from Dr. Jeremy Seamans. Any requests to access these datasets should be directed to jeremy.seamans@ubc.ca.

AUTHOR CONTRIBUTIONS

HT and DD conceived study, interpreted results, revised and finalized manuscript. HT developed and implemented methods, carried out simulations, analyzed data, prepared figures, wrote manuscript.

FUNDING

This research was funded by grants to DD by the German Research Foundation (DFG) (SPP1665, DU 354/8-2) and through the German Ministry for Education and Research (BMBF) via the e:Med framework (01ZX1311A & 01ZX 1314E).

ACKNOWLEDGMENTS

The authors thank Dr. Georgia Koppe and Dr. Eleonora Russo for discussions and Dr. Jeremy Seamans for providing the neural and behavioral data. This manuscript is posted to the preprint server arXiv, with no restrictions on full copyright and reuse rights, under the identifier arXiv:1802.03627.

REFERENCES

- Aminikhanghahi, S., and Cook, D. J. (2017). A survey of methods for time series change point detection. *Knowl. Inf. Syst.* 51, 339–367. doi: 10.1007/s10115-016-0987-z
- Antoch, J., and Hušková, M. (2001). Permutation tests in change point analysis. *Stat. Probab. Lett.* 53, 37–46. doi: 10.1016/S0167-7152(01)00009-8
- Antoch, J., Hušková, M., and Prášková, Z. (1997). Effect of dependence on statistics for determination of change. *J. Stat. Plan. Inference* 60, 291–310. doi: 10.1016/S0378-3758(96)00138-3
- Antoch, J., Hušková, M., and Veraverbeke, N. (1995). Change-point problem and bootstrap. *J. Nonparametr. Statist.* 5, 123–144. doi: 10.1080/10485259508832639
- Aston, J. A. D., and Kirch, C. (2012). Evaluating stationarity via change-point alternatives with applications to fMRI data. *Ann. Appl. Stat.* 6, 1906–1948. doi: 10.1214/12-AOAS565
- Bai, J. (1997). Estimating multiple breaks one at a time. *Econ. Theory* 13, 315–352. doi: 10.1017/S0266466600005831
- Basseville, M. (1988). Detecting changes in signals and systems—a survey. *Automatica* 24, 309–326. doi: 10.1016/0005-1098(88)90073-8

- Bhattacharya, P. K. (1994). Some aspects of change-point analysis. *Lect. Notes Monogr. Ser.* 23, 28–56. doi: 10.1214/lnms/1215463112
- Brown, R. L., Durbin, J., and Evans, J. M. (1975). Techniques for testing the constancy of regression relationships over time. *J. R. Stat. Soc. B Stat. Methodol.* 37, 149–192.
- Chen, J., and Gupta, A. K. (2012). *Parametric Statistical Change Point Analysis: With Applications to Genetics, Medicine, and Finance*. Basel: Springer Science+Business Media, LLC.
- Chernoff, H., and Zacks, S. (1964). Estimating the current mean of a normal distribution which is subjected to changes in time. *Ann. Math. Stat.* 35, 999–1018.
- Cho, H., and Fryzlewicz, P. (2015). Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *J. R. Stat. Soc. Series B Stat. Methodol.* 77, 475–507.
- Craven, P., and Wahba, G. (1978). Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. *Numer. Math.* 31, 377–403. doi: 10.1007/BF01404567
- Csörgő, M., and Horváth, L. (1997). *Limit Theorems in Change-point Analysis*, Vol. 18. New York, NY: John Wiley & Sons Inc.
- Davison, A. C., and Hinkley, D. V. (1997). *Bootstrap Methods and Their Application*. Cambridge Series in Statistical and Probabilistic Mathematics. New York, NY: Cambridge University Press.
- Dumbgen, L. (1991). The asymptotic behavior of some nonparametric change-point estimators. *Ann. Stat.* 19, 1471–1495. doi: 10.1214/aos/1176348257
- Durstewitz, D. (2017). *Advanced Data Analysis in Neuroscience: Integrating Statistical and Computational Models*. Bernstein Series in Computational Neuroscience. Cham: Springer International Publishing.
- Durstewitz, D., Vittoz, N. M., Floresco, S. B., and Seamans, J. K. (2010). Abrupt transitions between prefrontal neural ensemble states accompany behavioral transitions during rule learning. *Neuron* 66, 438–448. doi: 10.1016/j.neuron.2010.03.029
- Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *Ann. Stat.* 32, 407–499. doi: 10.1214/009053604000000067
- Elsner, J. B., Niu, X., and Jagger, T. H. (2004). Detecting shifts in hurricane rates using a Markov chain Monte Carlo approach. *J. Clim.* 17, 2652–2666. doi: 10.1175/1520-0442(2004)017<2652:DSIHUR>2.0.CO;2
- Fan, J., and Yao, Q. (2003). *Nonlinear Time Series: Nonparametric and Parametric Methods*. New York, NY: Springer.
- Fan, Z., Dror, R. O., Mildorf, T. J., Piana, S., and Shaw, D. E. (2015). Identifying localized changes in large systems: change-point detection for biomolecular simulations. *Proc. Natl. Acad. Sci. U.S.A.* 112, 7454–7459. doi: 10.1073/pnas.1415846112
- Feder, P. I. (1975). The log likelihood ratio in segmented regression. *Ann. Stat.* 3, 84–97. doi: 10.1214/aos/1176343000
- Friedman, J. H. (1991). Multivariate adaptive regression splines. *Ann. Stat.* 19, 1–67. doi: 10.1214/aos/1176347963
- Friedman, J. H., and Silverman, B. W. (1989). Flexible parsimonious smoothing and additive modeling. *Technometrics* 31, 3–21. doi: 10.1080/00401706.1989.10488470
- Fryzlewicz, P. (2014). Wild binary segmentation for multiple change-point detection. *Ann. Stat.* 42, 2243–2281. doi: 10.1214/14-AOS1245
- Gärtner, M., Duvarci, S., Roeper, J., and Schneider, G. (2017). Detecting joint pausiness in parallel spike trains. *J. Neurosci. Methods* 285, 69–81. doi: 10.1016/j.jneumeth.2017.05.008
- Gombay, E., and Horváth, L. (1996). On the rate of approximations for maximum likelihood tests in change-point models. *J. Multivar. Anal.* 56, 120–152. doi: 10.1006/jmva.1996.0007
- Hamilton, J. D. (1994). *Time Series Analysis*. Princeton, NJ: Princeton University Press.
- Hanks, T. D., and Summerfield, C. (2017). Perceptual decision making in rodents, monkeys, and humans. *Neuron* 93, 15–31. doi: 10.1016/j.neuron.2016.12.003
- Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edn*. New York, NY: Springer.
- Hinkley, D. V. (1969). Inference about the intersection in two-phase regression. *Biometrika* 56, 495–504. doi: 10.1093/biomet/56.3.495
- Hinkley, D. V. (1971a). Inference about the change-point from cumulative sum tests. *Biometrika* 58, 509–523. doi: 10.2307/2334386
- Hinkley, D. V. (1971b). Inference in two-phase regression. *J. Am. Stat. Ass.* 66, 736–743. doi: 10.2307/2284220
- Horváth, L. (1997). Detection of changes in linear sequences. *Ann. Inst. Stat. Math.* 49, 271–283. doi: 10.1023/A:1003110912735
- Hušková, M. (2004). “Permutation principle and bootstrap in change point analysis,” in *Asymptotic Methods in Stochastics: Festschrift for Miklós Csörgő*, Vol. 44 of *Fields Institute Communications*, eds L. Horváth and B. Szyszkowicz (Providence, RI: American Mathematical Society), 273–291. doi: 10.1090/fic/044/15
- Hušková, M., and Slabý, A. (2001). Permutation tests for multiple changes. *Kybernetika* 37, 605–622.
- Jirak, M. (2012). Change-point analysis in increasing dimension. *J. Multivar. Anal.* 111, 136–159. doi: 10.1016/j.jmva.2012.05.007
- Kendall, M. G., and Stuart, A. (1983). *The Advanced Theory of Statistics*, Vol. 3. London: Griffin.
- Kihlberg, J., Herson, J., and Schotz, W. (1972). Square root transformation revisited. *Appl. Statist.* 20, 76–81. doi: 10.2307/2346609
- Kirch, C. (2007). Block permutation principles for the change analysis of dependent data. *J. Stat. Plan. Inference* 137, 2453–2474. doi: 10.1016/j.jspi.2006.09.026
- Latimer, K. W., Yates, J. L., Meister, M. L. R., Huk, A. C., and Pillow, J. W. (2015). Single-trial spike trains in parietal cortex reveal discrete steps during decision-making. *Science* 349, 184–187. doi: 10.1126/science.aaa4056
- Liu, J., Wu, S., and Zidek, J. V. (1997). On segmented multivariate regression. *Stat. Sin.* 7, 497–525.
- Lombard, F., and Hart, J. (1994). The analysis of change-point data with dependent errors. *Lect. Notes Monogr. Ser.* 23, 194–209. doi: 10.1214/lnms/1215463125
- Matteson, D. S., and James, N. A. (2014). A nonparametric approach for multiple change point analysis of multivariate data. *J. Am. Stat. Assoc.* 109, 334–345. doi: 10.1080/01621459.2013.849605
- Olshen, A. B., Venkatraman, E., Lucito, R., and Wigler, M. (2004). Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics* 5, 557–572. doi: 10.1093/biostatistics/kxh008
- Page, E. (1954). Continuous inspection schemes. *Biometrika* 41, 100–115. doi: 10.1093/biomet/41.1-2.100
- Paillard, D. (1998). The timing of pleistocene glaciations from a simple multiple-state climate model. *Nature* 391, 378–381. doi: 10.1038/34891
- Picard, D. (1985). Testing and estimating change-points in time series. *Adv. Appl. Probab.* 17, 841–867. doi: 10.2307/1427090
- Powell, N. J., and Redish, A. D. (2016). Representational changes of latent strategies in rat medial prefrontal cortex precede changes in behaviour. *Nat. Commun.* 7:12830. doi: 10.1038/ncomms12830
- Quandt, R. E. (1958). The estimation of the parameters of a linear regression system obeying two separate regimes. *J. Am. Stat. Assoc.* 53, 873–880. doi: 10.1080/01621459.1958.10501484
- Roitman, J. D., and Shadlen, M. N. (2002). Response of neurons in the lateral intraparietal area during a combined visual discrimination reaction time task. *J. Neurosci.* 22, 9475–9489. doi: 10.1523/JNEUROSCI.22-21-09475.2002
- Scott, A. J., and Knott, M. (1974). A cluster analysis method for grouping means in the analysis of variance. *Biometrics* 30, 507–512. doi: 10.2307/2529204
- Shah, S. P., Lam, W. L., Ng, R. T., and Murphy, K. P. (2007). Modeling recurrent DNA copy number alterations in array CGH data. *Bioinformatics* 23, i450–i458. doi: 10.1093/bioinformatics/btm221
- Shumway, R. H., and Stoffer, D. S. (2010). *Time Series Analysis and Its Applications: With R Examples, 3rd Edn*. New York, NY: Springer.
- Smith, A. C., Frank, L. M., Wirth, S., Yanike, M., Hu, D., Kubota, Y., et al. (2004). Dynamic analysis of learning in behavioral experiments. *J. Neurosci.* 24, 447–461. doi: 10.1523/JNEUROSCI.2908-03.2004
- Smith, P. L. (1982). *Curve fitting and modeling with splines using statistical variable selection techniques*. Nasa report 166034. Hampton, VA: Langley Research Center.
- Stock, J. H., and Watson, M. W. (2014). Estimating turning points using large data sets. *J. Econom.* 178, 368–381. doi: 10.1016/j.jeconom.2013.08.034

- Stone, C. J., Hansen, M. H., Kooperberg, C., and Truong, Y. K. (1997). Polynomial splines and their tensor products in extended linear modeling: 1994 Wald memorial lecture. *Ann. Stat.* 25, 1371–1470. doi: 10.1214/aos/1031594728
- Strogatz, S. H. (2001). *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. Cambridge, MA: Perseus Books Publishing.
- Vert, J.-P., and Bleakley, K. (2010). “Fast detection of multiple change-points shared by many signals using group LARS,” in *Advances in Neural Information Processing Systems 23*, eds J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta (Red Hook, NY: Curran Associates, Inc.), 2343–2351.
- Wang, T., and Samworth, R. J. (2018). High dimensional change point estimation via sparse projection. *J. R. Stat. Soc. Series B Stat. Methodol.* 80, 57–83. doi: 10.1111/rssb.12243

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The reviewer JI and handling editor declared their shared affiliation, at the time of the review.

Copyright © 2018 Toutounji and Durstewitz. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.