Data-Driven Determination of the Number of Jumps in Regression Curves

Guanghui Wang and Changliang Zou
School of Statistics and Data Science, Nankai University
and
Peihua Qiu
Department of Biostatistics, University of Florida

April 20, 2021

Abstract

In nonparametric regression with jump discontinuities, one major challenge is to determine the number of jumps in a regression curve. Most existing methods to solve that problem are based on either a sequence of hypothesis tests or model selection, by introducing some extra tuning parameters that may not be easy to determine in practice. This paper aims to develop a data-driven new methodology for determining the number of jumps, using an order-preserved sample-splitting strategy together with a cross-validation-based criterion. Statistical consistency of the determined number of jumps by our proposed method is established. More interestingly, the proposed method allows us to move beyond just point estimation, and it can quantify uncertainty of the proposed estimate. The key idea behind our method is the construction of a series of statistics with marginal symmetry property and this property can be used for choosing a data-driven threshold to control the false discovery rate of our method. The proposed method is computationally efficient. Numerical experiments indicate that it has a reliable performance in finite-sample cases. A R package jra is developed to implement the proposed method.

Keywords: Cross-validation; False discovery rate; Jump discontinuity; Local linear smoothing; Nonparametric regression; Uniform convergence.

1 Introduction

Nonparametric regression analysis when the underlying regression curve has jumps, or simply one-dimensional (1-D) jump regression analysis (JRA), has received much attention in the literature because of its broad applications in internet traffic management, disease surveillance, environment monitoring, meteorology, and more. Conventional smoothing techniques for estimating continuous regression curves (Fan and Gijbels, 1996) cannot work well for 1-D JRA because the jump structure, which is often important for a specific application, would be blurred by them. For a comprehensive overview on 1-D JRA, see Chapter 3 in Qiu (2005). In general, 1-D JRA has two major goals. One is to estimate the jump part of the true regression curve, including the number, locations and magnitudes of jumps, and the other is to estimate the entire regression curve with the possible jumps preserved. These two problems are often referred to as *jump detection* and *jump-preserving curve estimation*, respectively. This paper focuses on jump detection.

Consider the following nonparametric regression model:

$$Y_i = f(t_i) + \varepsilon_i, \ \ i = 1, \ldots, n, \tag{1}$$

where $t_1 < \cdots < t_n$ are fixed design points in the interval $[0, 1]$, ε_i's are independent and identically distributed (i.i.d.) random errors with mean zero and variance σ^2, and $f(t)$ is the regression function. In this paper, $f(t)$ is assumed to have the following structure:

$$f(t) = f_{\rm C}(t) + \sum_{j=1}^{J^*} \delta_j^* I_{\{t \ge t_j^*\}}, \tag{2}$$

where $f_{\rm C}(t)$ is the continuity part of $f(t)$, and $\sum_{j=1}^{J^*} \delta_j^* I_{\{t \ge t_j^*\}}$ is the jump part with the jump locations $0 < t_1^* < \cdots < t_{J^*}^* < 1$, the jump magnitudes $\{\delta_j^*\}$, and the number of jumps J^*.

In (2), all quantities are unknown and need to be estimated from the observations $\{Y_i\}$.

For jump detection, most existing approaches start with a diagnostic statistic computed from observations in a local neighborhood of a given point, such as the difference between a right- and a left-sided local weighted average. Then, a large value of the diagnostic statistic would indicate a potential jump near the given point. Some examples of existing methods based on that idea can be found in Müller (1992), Loader (1996), Gijbels et al. (1999), Delgado and Hidalgo (2000) and Grégoire and Hamrouni (2002). In jump detection, determination of the number of jumps naturally plays a key role. In the literature, several methods have been proposed for that purpose. These existing methods estimate the number of jumps by either performing a sequence of hypothesis tests for existence of jumps at individual design points, or introducing a thresholding rule for determining the jump locations (Qiu, 1994; Wu and Chu, 1993a). Such methods are often difficult to use in practice since the convergence rates of the related test statistics are rather slow and/or some nuisance parameters need to be selected in advance. Recently, Xia and Qiu (2015) proposed a jump information criterion (JIC) from a model selection perspective. Specifically, the JIC consists of two terms: one measures the goodness-of-fit of a potential model estimate and the other penalizes for model complexity. Unlike the conventional Bayesian information criterion (BIC, Schwarz (1978)) in the model selection context, the penalty term in JIC incorporates the information about the jump part of $f(t)$ (cf., (2)). However, JIC involves tuning parameters that need to be specified in advance, and their optimal values may depend on the underlying model. Thus, there are no guidelines about their selection that are universally appropriate.

In this paper, we suggest a new method for estimating the number of jumps. Its novelty lies behind the fact that it is data-driven and no tuning parameters except a bandwidth are involved in the estimation. Guidelines on how to choose the bandwidth in a data-driven way are provided. Thus, it can be used in applications conveniently. The new method is based on the *Order-Preserved Splitting (OPS)* strategy that was proposed recently by Zou et al. (2020). By OPS, all observations are divided into two parts: one consists of observations at odd-indexed design points and the other consists of observations at even-indexed design points. Thus, the two parts have almost the same number of observations, and each part should contain information about the original jump structure of the regression model (1). Then, the number of jumps can be estimated by using a cross-validation (CV) criterion, in which the jump part of $f(t)$ is first estimated from one part of the observed data and then the estimate is validated using the other part. Under some mild conditions, it can be verified that the estimated number of jumps is statistically consistent. Numerical experiments will show that the proposed method has a reliable performance in a variety of simulated and real-data examples. As far as we know, the only related work is Müller and Stadtmüller (1999) which proposed to minimize a cross-validated residual sum of squares with respect to the number of jumps. However, they did not provide any theoretical results on the estimated number of jumps.

The JIC and CV methods discussed above focus mainly on consistent point estimation of the number of jumps J^*. They cannot avoid under- or over-estimation in finite sample cases. To overcome that limitation, we try to make another contribution in this paper by proposing an estimation procedure with the false discovery rate (FDR; Benjamini and

Hochberg (1995)) controlled properly. As mentioned earlier about the existing methods, to achieve consistent estimation of the number of jumps, they often impose some stringent conditions on the minimum jump magnitude so that the largest J^* local maximizers of their jump detection criteria can be computed for estimating the J^* true jump points. In many applications, a slightly conservative estimation of the number of jumps (i.e., the estimate of J^* is a bit larger than J^*) is often preferred because we do not want to miss any important jumps. In such cases, measures like FDR can help us quantify the *uncertainty* in the estimated jumps. However, it is not an easy task to control FDR when estimating the jump points, because this is equivalent to conducting a test using a statistic with complicated null distribution. To overcome this difficulty, we develop a simple but effective procedure that avoids using any asymptotic distributions while controlling FDR. With the help of OPS, our proposed method entails finding a series of statistics with the marginal symmetry property, by which the empirical distribution of the negative statistics can be used for approximating that of the positive ones. The new method is computationally efficient, and its ability in controlling FDR has an intuitive explanation. This FDR control procedure is data-driven, and only a bandwidth involved needs to be chosen. To specify the bandwidth properly, some practical guidelines are provided. Thus, the proposed FDR control procedure is convenient for practical use.

The remainder of this paper is organized as follows. In Section 2, we present our proposed method for estimating the number of jumps using the CV criterion. Section 3 describes our proposed procedure to determine a final set of the detected jump points with FDR properly controlled. Section 4 provides some details on their implementation, includ-

ing guidelines on how to select the bandwidth involved in both procedures. A number of simulated and real-data examples are presented in Section 5 regarding the numerical performance of the proposed methods. Finally, some remarks conclude the paper in Section 6. Proofs of theoretical results and some additional numerical examples are given in Supplementary Material.

2 Consistent estimation of the number of jumps

2.1 Proposed method

We suggest a data-driven method to estimate the number of jumps J^*. First, by applying the OPS strategy (Zou et al., 2020), all observed data are divided into two parts according to the parity of their time indices, being either odd (O) or even (E),

$$\mathcal{Z}^O := \{(Y_k^O, t_k^O) : Y_k^O = Y_{2k-1}, t_k^O = t_{2k-1}, k = 1, \ldots, m\},$$

$$\mathcal{Z}^E := \{(Y_k^E, t_k^E) : Y_k^E = Y_{2k}, t_k^E = t_{2k}, k = 1, \ldots, m\},$$

where for simplicity it has been assumed that the sample size $n = 2m$ is even. Then, one of these two parts is used as a training dataset for estimating the jump part of $f(t)$, and the other is for validation. Compared to the random sampling scheme to divide the observed data into two parts, this OPS strategy can preserve the underlying jump structure as much as possible in both $\mathcal{Z}^O$ and $\mathcal{Z}^E$.

Without loss of generality, let us use $\mathcal{Z}^O$ as the training dataset for jump detection, and $\mathcal{Z}^E$ for validation. To detect jumps using $\mathcal{Z}^O$, we choose a local smoothing method similar to some existing methods introduced in Section 1. To be more specific, the local linear

kernel (LLK) estimation (Grégoire and Hamrouni, 2002; Loader, 1996; Xia and Qiu, 2015) will be adopted, because of its good properties of design adaptation and small boundary biases (Fan and Gijbels, 1996). Let $K(\cdot)$ be a kernel function with the support $[0,1]$ and $h>0$ be a bandwidth. Then, for a candidate jump location $t \in [h, 1-h]$, the corresponding jump magnitude can be estimated by the difference between a right- and a left-sided LLK estimate of $f(t)$ based on $\mathcal{Z}^O$, namely,

$$\widehat{\delta}^O(t) = \widehat{Y}_+^O(t) - \widehat{Y}_-^O(t), \tag{3}$$

where

$$\begin{aligned}
\widehat{Y}_+^O(t) &= \frac{\sum_{k=1}^m w_k^O\left(t; K_+\right) Y_k^O}{\sum_{k=1}^m w_k^O\left(t; K_+\right)}, \\
w_k^O\left(t; K_+\right) &= \left\{ s_2^O\left(t; K_+\right) - s_1^O\left(t; K_+\right)\left(t_k^O - t\right)\right\} K_+\left(\frac{t_k^O - t}{h}\right), \\
s_r^O\left(t; K_+\right) &= \sum_{k=1}^m (t_k^O - t)^r K_+\left(\frac{t_k^O - t}{h}\right), \ r = 0, 1, 2,
\end{aligned}$$

$K_+(u) = K(u) I_{\{u\in[0,1]\}}$, and $\widehat{Y}_-^O(t)$ is defined in the same way as $\widehat{Y}_+^O(t)$ except that all subscripts "+" should be replaced by "−" and that $K_-(u) = K(-u) I_{\{u\in[-1,0)\}}$ needs to be used. Intuitively, if t is close to a true jump point, then $|\widehat{\delta}^O(t)|$ would be relatively large. Consequently, the maximizer of $|\widehat{\delta}^O(t)|$ over $t \in [h, 1-h]$ can be used as the estimate of the most "significant" jump location, denoted as $\widehat{t}_1^O$. For practical implementation, we introduce the searching grid $\mathcal{G} = \{t_i,\ i = 1, \ldots, n\} \cap [h, 1-h]$. Then, $\widehat{t}_1^O = \arg\max_{t\in\mathcal{G}} |\widehat{\delta}^O(t)|$. Estimates of subsequent possible jumps can be found by

$$\widehat{t}_j^O = \underset{t\in\mathcal{G}\setminus\cup_{k=1}^{j-1}[\widehat{t}_k^O - \kappa h, \widehat{t}_k^O + \kappa h]}{\arg\max} |\widehat{\delta}^O(t)|,\ j = 2, 3, \ldots, \tag{4}$$

where $\kappa > 0$ is a constant that is set to be 1 in our numerical analysis. Obviously, the procedure (4) is essentially a "forward" searching procedure, and similar to the binary

segmentation algorithm in the change-point literature (Hawkins, 2001), while in each recursive step small neighborhoods of previously detected jumps have been removed to avoid false jump detection. It should be pointed out that the true jump points are estimated by discrete design points in (4) because the observed data cannot accurately specify a jump location between two consecutive design points, which is routinely done in the JRA literature (Loader, 1996).

In the above jump detection algorithm, the searching procedure (4) would be terminated with say $\tilde{J}_O$ ($\tilde{J}_O \leq \lceil 1/(\kappa h) \rceil$) estimated jumps by its definition or could be intentionally interrupted up to some pre-specified value for the number of candidate jumps say $\bar{J}$. Here $\lceil x \rceil$ represents the smallest integer not less than x. As a result, the number of the final detected jumps is $\bar{J}_O := \min\{\bar{J}, \tilde{J}_O\}$. To avoid missing any important jumps in the regression model, $\bar{J}$ should be chosen large intentionally, namely, $\bar{J} \geq J^*$. In practice, J^* should be much less than $1/(\kappa h)$, for a reasonably small bandwidth h. Thus, we can safely set $\bar{J} = \lceil 1/(\kappa h) \rceil$ in a specific application problem.

However, in such cases, some of the detected jumps could be false jumps and they need to be screened out. To this end, the test dataset $\mathcal{Z}^E$ can be used to validate the estimates. Let $\widehat{\delta}^E(t) = \widehat{Y}_+^E(t) - \widehat{Y}_-^E(t)$, where $\widehat{Y}_+^E(t)$ and $\widehat{Y}_-^E(t)$ are defined in the same way as $\widehat{Y}_+^O(t)$ and $\widehat{Y}_-^O(t)$, except that all superscripts "O" in their definitions need to be replaced by "E". For $1 \leq J \leq \bar{J}_O$, let $\tilde{\delta}^O(\widehat{t}_j^O) = \widehat{\delta}^O(\widehat{t}_j^O)$, for $j = 1, \ldots, J$, and $\tilde{\delta}^O(\widehat{t}_j^O) = 0$, for $j = J+1, \ldots, \bar{J}_O$. Then, $\{\tilde{\delta}^O(\widehat{t}_j^O)\}_{j=1}^{\bar{J}_O}$ represents our estimation of the jump part of $f(t)$ from the training dataset when the number of jumps is set to be J. To check whether our estimated jump part from the training dataset is reasonable, we can compare it with

$\{\widehat{\delta}^E(\widehat{t}_j^O)\}_{j=1}^{\bar{J}_O}$ which is computed from the test dataset, by considering

$$\begin{aligned}\mathcal{C}(J;\mathcal{Z}^O,\mathcal{Z}^E) &= \sum_{j=1}^{\bar{J}_O}\left\{\widehat{\delta}^E(\widehat{t}_j^O)-\tilde{\delta}^O(\widehat{t}_j^O)\right\}^2 \\ &= \sum_{j=1}^{J}\left\{\widehat{\delta}^E(\widehat{t}_j^O)-\widehat{\delta}^O(\widehat{t}_j^O)\right\}^2+\sum_{j=J+1}^{\bar{J}_O}\left\{\widehat{\delta}^E(\widehat{t}_j^O)\right\}^2. \end{aligned} \tag{5}$$

Intuitively, if for each $j=1,\ldots,J^*$, $\widehat{t}_j^O$ is closely located around $t^*_{k_j}$ for some $k_j\in\{1,\ldots,J^*\}$, then both $\widehat{\delta}^O(\widehat{t}_j^O)$ and $\widehat{\delta}^E(\widehat{t}_j^O)$ would approximate $\delta^*_{k_j}$ well. Our procedure could then pick all J^* jumps out in the first J^* steps if two successive jumps is not too close. Consequently, when $J<J^*$, the second term in (5) could be relatively large, compared to the first term, which helps prevent underfitting (if the last J^*-J jump magnitudes are not very small). When $J>J^*$, both terms would be small. But, according to the proof of Theorem 1, it turns out that $\widehat{\delta}^O(\widehat{t}_j^O)-\delta^*_{k_j}$ would be of higher order than $\widehat{\delta}^E(\widehat{t}_j^O)-\delta^*_{k_j}$ asymptotically, where we set $\delta^*_{k_j}=0$ for $j>J^*$. This can be explained by the fact that the former is a local maximization-induced statistic that has a lower bound of the order $\sqrt{\log n/(nh)}$ (cf., Lemma **??** in Supplementary Material) which is larger than the upper bound with the order $\sqrt{1/(nh)}$ of the latter term. Hence the first term in (5) would dominate the second term, and it can help avoid overfitting. In conclusion, $\mathcal{C}(J;\mathcal{Z}^O,\mathcal{Z}^E)$ is a reasonable criterion for determining the number of jumps J^*.

In practice, we can also define $\mathcal{C}(J;\mathcal{Z}^E,\mathcal{Z}^O)$ in a similar way as that for $\mathcal{C}(J;\mathcal{Z}^O,\mathcal{Z}^E)$, by switching the positions of $\mathcal{Z}_O$ and $\mathcal{Z}_E$. Thus, our cross-validated estimate of J^* is defined to be

$$\widehat{J}^*=\arg\min_{1\le J\le J_c}\{\mathcal{C}(J;\mathcal{Z}^O,\mathcal{Z}^E)+\mathcal{C}(J;\mathcal{Z}^E,\mathcal{Z}^O)\}, \tag{6}$$

where $J_c=\min\{\bar{J}_O,\bar{J}_E\}$ and $\bar{J}_E$ is the largest number of candidate jumps when applying

the searching procedure on $\mathcal{Z}^E$. This is in fact a specialized two-fold CV metric, where the data is split into two folds with each fold preserving the original jump structure as much as possible. The CV strategy is preferred here, because it can reduce variance of the resulting estimate as verified by our simulation studies. After J^* is estimated, the jump locations can then be estimated by applying the jump detection algorithm (4) to the entire dataset (i.e., $\{(Y_i, t_i), i = 1, \ldots, n\}$) using the knowledge that the number of jumps is $\widehat{J^*}$. Note that our procedure does not require the knowledge of the variance σ^2 or its estimate.

As a comparison, the JIC method proposed by Xia and Qiu (2015) estimates J^* via minimizing

$$\mathrm{JIC}(J) = n \log \left[n^{-1} \sum_{i=1}^{n} \left\{ Y_i - \widehat{f}(t_i; J) \right\}^2 \right] + P_n \sum_{j=1}^{J} |\widehat{\delta}(\widehat{t}_j)|^{-\nu},$$

where the estimated jump locations $\widehat{t}_1, \ldots, \widehat{t}_J$ are obtained as in (4) but from the entire dataset $\{(Y_i, t_i), i = 1, \ldots, n\}$, $\widehat{f}(t; J)$ is a LLK estimate of the underlying regression curve $f(t)$ based on the assumption that there are J jumps in $f(t)$, and $\widehat{\delta}(t)$ is the estimated jump magnitude at t obtained as in (3) but from the entire dataset. In the above JIC criterion, P_n is an adjustment factor for the penalty term $\sum_{j=1}^{J} |\widehat{\delta}(\widehat{t}_j)|^{-\nu}$ and $\nu \geq 0$ is a tuning parameter. It has been shown that P_n plays an important role in balancing the under- and over-estimation. Based on some study on its asymptotic properties, Xia and Qiu (2015) suggested choosing P_n to be a quantity in the order of $(nh/\log n)^{-\nu/2} nh$. For practical applications, one may set $P_n = C(nh/\log n)^{-\nu/2} nh$ for some constant $C > 0$ and an appropriate value of ν. However, simulation studies reveal that the optimal values of C and ν may vary from the underlying jump structure such as the number, locations and/or magnitudes of jumps. Consequently, there are no universal guidelines to specify it

before use. In our criterion (5), the first term plays a similar role to that of the second term in JIC(J). Both of them are used to avoid overestimation. As opposed to the JIC method, our CV-based objective function in (6) is *data-driven* and no nuisance parameters (except for a bandwidth) need to be selected in advance. Therefore, it is more convenient to use the proposed method in practice. This data-driven feature benefits from the use of sample-splitting and thus certain efficiency loss would be incurred in the exchange for the robustness in finite sample scenarios.

2.2 Theoretical justifications

Our proposed method for estimating the true number of jumps will be called *Cross-validation with Order-Preserved Splitting (COPS)* method in the remaining part of the paper. Its statistical consistency is established in Theorem 1.

Theorem 1. *Under Assumptions* **??**–**??** *in Section* **??** *of Supplementary Material, the COPS estimate is consistent in the sense that* $\Pr(\widehat{J}^* = J^*) \to 1$, *as* $n \to \infty$.

A technical discussion on the assumptions is deferred to Section **??** of Supplementary Material. Zou et al. (2020) showed that the CV-based procedure can achieve selection consistency under a class of parametric models and conjectured that this should also be valid for nonparametric regression models. Theorem 1 confirms this conjecture.

3 Jump detection with FDR under control

It has been shown above that the estimated number of jumps is consistent under some conditions. In fact, it can be further verified that the first J^* detected jump points $\widehat{t}_1^O, \ldots, \widehat{t}_{J^*}^O$ would each lie in a neighborhood of a true jump location t_j^*. In finite-sample cases, however, this property typically does not hold. Intuitively, if some δ_j^*'s are small, then they are very likely to be missed in the first J^* steps of the recursive jump detection algorithm but could be recovered in later steps. In such cases, the "entire solution path" $\{\widehat{t}_1^O, \ldots, \widehat{t}_J^O\}_{J=1}^{\bar{J}_O}$ would be overfitting, and $\{\widehat{t}_1^O, \ldots, \widehat{t}_J^O\}_{J=1}^{J^*}$ would not be a correct estimate. Accordingly, because one is often reluctant to miss any important jumps, we usually prefer an overfitting estimate with the number (or rate) of falsely identified jump points under control.

Here, we consider using the false discovery rate (FDR) which is a particularly useful tool to maintain the ability to reliably detect true signals without excessive false positive results (Benjamini and Hochberg, 1995). In the context of change-point detection, some related works on error rate control include Siegmund et al. (2011), Hao et al. (2013), Frick et al. (2014) and Li et al. (2016). To our best knowledge, there is no existing study on using FDR to determine the number of jumps in the nonparametric regression literature.

3.1 Definition of false discoveries

Before proceeding, it is necessary to discuss the definition of false discoveries. Let $\mathcal{T} = (\tau_1, \ldots, \tau_{\bar{J}})$ be a candidate jump set, for example, $\mathcal{T} = (\widehat{t}_1^O, \ldots, \widehat{t}_{\bar{J}_O}^O)$ obtained in Section 2. A selection procedure is then applied to $\mathcal{T}$ to determine which elements are "true" jumps. For a given element τ_k, even $\tau_k \neq t_j^*$, for any $j = 1, \ldots, J^*$, it can still be a good estimate if it

is close to one of t_j^*'s. Hence, we will use the concepts of "informative" and "uninformative" points to classify the points in $\mathcal{T}$, analogous to the "alternative" and "null" hypotheses in the context of multiple testing. It seems reasonable to define τ_k to be an informative point if

$$\min_{1\le j\le J^*} |\tau_k - t_j^*| \le \omega,$$

where ω is a user-specified threshold, for example, $\omega = h$. This notion provides immediate control over location accuracy and has been used in the literature of change-point analysis (Hao et al., 2013). But, it is inappropriate in the situation of overfitting that we are mainly concerned about. Imagining that there are more than one τ_k's close to a true change-point (their distances are all less than ω), all of them will be viewed as informative points. Also, choosing ω is somewhat subjective, which we always want to avoid.

To overcome the limitation mentioned above, we adopt the definition given by Li et al. (2016): τ_k is classified as an informative point if there is a true jump point lying in

$$\Big[(\tau_{k-1} + \tau_k)/2, (\tau_k + \tau_{k+1})/2\Big), \tag{7}$$

where $\tau_0 = 0$ and $\tau_{\bar{J}+1} = 1$; otherwise, it is an uninformative point. We denote by $\mathcal{I}_0$ and $\mathcal{I}_1$ the uninformative and informative sets, respectively. This notion is well defined since every candidate jump point is either informative or uninformative, but not both, and there is at most one informative point corresponding to each true jump point. The Eq. (7) is a weak definition of informative point in the sense that it neglects the location accuracy of τ_k's, but it is suitable for our current research problem which primarily focuses on determination of the number of jumps rather than estimating the locations.

Consider a procedure to select a subset of detected jumps from $\mathcal{T}$, and the selected

subset is denoted as $\mathcal{S} = (\tau_{k_1}, \ldots, \tau_{k_{|S|}})$. A false discovery is made by $\mathcal{S}$ if $\tau_k \in \mathcal{I}_0 \cap \mathcal{S}$. Then, the FDP associated with $\mathcal{S}$ is defined to be

$$\mathrm{FDP}(\mathcal{S}) = \frac{\#\{k : \tau_k \in \mathcal{I}_0 \cap \mathcal{S}\}}{|\mathcal{S}| \vee 1},$$

and the FDR is accordingly defined as the expectation of $\mathrm{FDP}(\mathcal{S})$.

3.2 Method construction

Let $\mathcal{T} = (\widehat{t}_1^O, \ldots, \widehat{t}_{\bar{J}}^O)$. The "likelihood" of $\widehat{t}_k^O$ being a jump point can be measured by

$$W_k = nh\widehat{\delta}^O(\widehat{t}_k^O)\widehat{\delta}^E(\widehat{t}_k^O),\ k = 1, \ldots, \bar{J}.$$

Clearly, W_k is likely to be large for most informative points regardless of the signs of jumps, and small for most uninformative points. Note that W_k is (asymptotically) symmetric with mean zero for any $\widehat{t}_k^O \in \mathcal{I}_0$ due to the asymptotic normality of $\widehat{\delta}^E(\widehat{t}_k^O)$ by the central limit theorem and the independence between $\mathcal{Z}_O$ and $\mathcal{Z}_E$. Thus, we can choose a threshold $L > 0$ by setting

$$L = \inf\left\{s > 0 : \frac{1 + \#\{k : W_k \le -s\}}{\#\{k : W_k \ge s\} \vee 1} \le \alpha\right\}, \tag{8}$$

and identify $\widehat{t}_k^O$ as an informative point if $W_k \ge L$, where α is the target FDR level. If the set on the right-hand-side of (8) is empty, then we simply set $L = +\infty$. The selected set of informative jumps is denoted as $\mathcal{S}(L)$.

Note that the set $\{k : W_k \le -s, \widehat{t}_k^O \in \mathcal{I}_1\}$ is often very small when the jump magnitudes are not too small. Thus, $\#\{k : W_k \le -s\}$ is a good approximation to $\#\{k : W_k \le -s, \widehat{t}_k^O \in \mathcal{I}_0\}$ which is a good approximation to $\#\{k : W_k \ge s, \widehat{t}_k^O \in \mathcal{I}_0\}$ due to the marginal symmetry

of W_k for $\widehat{t}_k^O \in \mathcal{I}_0$. This implies that the fraction in (8) should be a good estimate of the proportion of false discoveries. Since we use the empirical distribution of the negative statistics to approximate that of the positive ones, we call our method *Symmetry-based OPS* (SOPS) selection procedure. Benefiting from the joint use of W_k and L, the SOPS procedure is data-driven and its implementation does not depend on any asymptotic or simulated distributions for procedure calibration.

The definition of L in (8) is similar to that in the knockoff framework introduced by Barber and Candès (2015) for parametric regression modeling. The knockoff procedure operates by constructing copies of each of the covariates with certain knowledge of the covariates or responses. However, in the current problems, such copies are not directly available. Instead, our sample-splitting strategy, in conjunction with the proposed statistic W_k, fulfills our goal to control FDR.

3.3 Theoretical justifications

Assumption 1 (*Strong signals*)**.** The jump magnitudes $\{\delta_j^*\}$ satisfy the conditions that $\max_{1\le j\le J^*}|\delta_j^*| = O(1)$ and, as $n\to\infty$, $\beta_n\to\infty$, where $\beta_n = |\mathcal{M}_\delta|$ and $\mathcal{M}_\delta = \{1\le j\le J^* : \left(|\delta_j^*|/\sigma\right)/\sqrt{\log n/(nh)}\to\infty\}$.

Theorem 2. *Under Assumptions* **??**–**??** *in Section* **??** *of Supplementary Material and the condition that* $\bar{J}(\log n)^{3/2}/\sqrt{nh}\to 0$ *as* $n\to\infty$*, we have* $\limsup_{n\to\infty}\mathrm{FDR}(L)\le\alpha$*, if either of the following two conditions holds:*

(i) With probability tending to one, W_k*'s are independent of each other for* $\widehat{t}_k^O\in\mathcal{I}_0$*;*

(ii) Assumption 1 holds.

Theorem 2 implies that the SOPS jump selection procedure can control the FDR level asymptotically under some mild conditions. In this theorem, $\bar{J}$ is allowed to increase with n (so is J^*), as long as $\bar{J}(\log n)^{3/2}/\sqrt{nh} \to 0$. In cases when W_k's are independent (e.g., when $\kappa > 2$ in (4)), we do not need any other conditions on the jump magnitudes, such as Assumption **??** (see Section **??** of Supplementary Material) required in Theorem 1. In more general cases when W_k's are correlated, Assumption 1 is needed, which implies that the number of jumps with identifiable jump magnitudes is not too small as $n \to \infty$. As opposed to Assumption **??**, here we allow a certain amount of jumps that are small and may not be distinguishable from random noise. In such situations, some uninformative points would be included in $\mathcal{T}$ before some informative ones, and consequently some existing methods, such as the JIC and COPS, would produce "inconsistent" jump detection results. In contrast, the SOPS method, which usually yields overestimation, can guarantee that the rate of uninformative points included in $\mathcal{S}(L)$ is under control. If in addition Assumption **??** holds, the SOPS method is capable of selecting all informative points in the sense that $\lim_{n\to\infty} \Pr\{\mathcal{S}(L) \supseteq \mathcal{I}_1\} = 1$ (see Section **??** of Supplementary Material).

4 Implementation

4.1 Computation

Both the proposed COPS and SOPS methods rely on evaluations of the LLK estimates at the grid set $\mathcal{G}$, which would require $O(n^2h)$ operations for a naive implementation. This can be improved to $O(n)$ operations by using for example the updating strategy discussed

in Fan and Marron (1994), although we will not pursue it here. Then, finding all candidate jumps could be achieved by an efficient sorting algorithm with the computing complexity of $O(n \log n)$. In the model selection stage to determine the number of jumps, both methods only need to do a few additional evaluations with the computing complexity of at most $O(h^{-1} \times nh) = O(n)$ about the LLK estimates computed from the validation dataset at the selected jumps. Hence, both COPS and SOPS methods require at most $O(n^2 h)$ operations in total.

4.2 Guidelines on bandwidth selection

Bandwidth selection plays a key role in JRA, and different choices for the bandwidth often lead to different conclusions. If the goal is to accurately recover the underlying regression function, Wu and Chu (1993b) proposed selecting the bandwidth by minimizing a cross-validated residual sum of squares in all subintervals segmented by the detected jumps, and Spokoiny (1998) suggested a pointwise adaptive strategy that searched for a maximal bandwidth for fitting the regression function. If the target is for jump detection, then a small value of the bandwidth should be used as it could capture the data structure around the jumps, although the selected bandwidth should not be too small to avoid artificial peaks in the related jump detection criterion. Gijbels and Goderniaux (2004) proposed selecting the bandwidth by maximizing the probability of accurately localizing a jump via bootstrap. However, the bootstrap approximation is not computationally efficient even in cases with just one jump, because the procedure requires first refitting the regression model based on the detected jump with the bandwidth selected by a CV method and then generating the

bootstrap replications for obtaining a sequence of bootstrap estimates of the jumps.

In this paper, we provide some guidelines on selecting a globally effective bandwidth for both of the proposed procedures COPS and SOPS. These guidelines are data-driven, except that a sequence of candidate values for the bandwidth, denoted as $\mathcal{H}$, needs to be specified in advance. To be more specific, let $\mathcal{H} = \{h = h_{\max} a^j : h \geq h_{\min}, j = 0, 1, 2, \ldots\}$, where $0 < h_{\min} < h_{\max}$ are the minimum and maximum candidate values and $0 < a < 1$ is a constant. Then, for selecting a global bandwidth for the COPS procedure, we suggest using a three-fold OPS strategy described below. Let

$$\mathcal{Z}^{(s)} := \{(Y_k^{(s)}, t_k^{(s)}) : Y_k^{(s)} = Y_{\mathcal{I}_{sk}}, t_k^{(s)} = t_{\mathcal{I}_{sk}}, k = 1, \ldots, m\}, \ s = 1, 2, 3,$$

where $\mathcal{I}_{1k} \in \{3k-2, 3k-1, 3k\}$, $\mathcal{I}_{2k} \in \{3k-2, 3k-1, 3k\} \backslash \mathcal{I}_{1k}$ and $\mathcal{I}_{3k} \in \{3k-2, 3k-1, 3k\} \backslash \{\mathcal{I}_{1k} \cup \mathcal{I}_{2k}\}$, for $k = 1, \ldots, m$. Without loss of generality, assume that $n = 3m$. Then, $\mathcal{Z}^{(1)}$ will be used to estimate the jump part of the regression function, and $\mathcal{Z}^{(2)}$ will be used to select the number of jumps, similarly to the roles played by $\mathcal{Z}^O$ and $\mathcal{Z}^E$ in the original COPS procedure. The remaining dataset $\mathcal{Z}^{(3)}$ will be used for bandwidth selection.

To proceed, for a specific candidate value $h \in \mathcal{H}$, a sequence of jumps can be determined by applying the COPS to $\mathcal{Z}^{(1)}$ and $\mathcal{Z}^{(2)}$. Then, we can obtain an estimate of the underlying regression function based on $\mathcal{Z}^{(1)}$, denoted as $\widehat{f}^{(12)}(t)$. This estimate can be used to predict the response values at $t_k^{(3)}$'s in $\mathcal{Z}^{(3)}$. By comparing these predicted responses with $Y_k^{(3)}$'s, we can define the residual sum of squares $\text{RSS}(h) = \sum_{k=1}^m \left\{Y_k^{(3)} - \widehat{f}^{(12)}(t_k^{(3)})\right\}^2$. A bandwidth that minimizes $\text{RSS}(h)$ should be good for estimating the regression function. This strategy has certain similarity to the leave-one-out CV method used for bandwidth selection in Wu and Chu (1993b), although the OPS strategy is used here. Asymptotically speaking, the

estimated number of jumps based on $\mathcal{Z}^{(1)}$ and $\mathcal{Z}^{(2)}$ should still be consistent for any given bandwidth that meets the assumptions in Theorem 1. But, it is still an open problem whether the bandwidth that minimizes $\mathrm{RSS}(h)$ defined above can keep the consistency. This theoretical issue will be studied in our future research.

For practical purposes, in order to improve the stability of bandwidth selection and reduce the power loss due to sample-splitting, we suggest a multiple-splitting strategy by randomizing the construction of $\mathcal{I}_{sk}$, for $s = 1, 2, 3$. More specifically, let us randomly select three consecutive observations without replacement from the original observed data $\{(Y_i, t_i), i = 1, 2, \ldots, n\}$, and randomly assign one and only one of them to each of $\mathcal{Z}^{(1)}$, $\mathcal{Z}^{(2)}$ and $\mathcal{Z}^{(3)}$. This random-selection process continues until each of $\mathcal{Z}^{(1)}$, $\mathcal{Z}^{(2)}$ and $\mathcal{Z}^{(3)}$ has m observations, and one random sample-splitting is then completed. Assume that a total of U random sample-splittings have been completed. Then, we can obtain U sets of residual sum of squares, denoted as $\{\mathrm{RSS}^u(h), h \in \mathcal{H}\}$, for $u = 1, \ldots, U$, each of which is computed from one randomly splitted samples. Let $\hat{h}_u = \arg\min_{h \in \mathcal{H}} \mathrm{RSS}^u(h)$, for $u = 1, \ldots, U$. Then, we select h to be $\widehat{h} = \max_{u=1,\ldots,U} \hat{h}_u$, which returns the most parsimonious model.

For the SOPS procedures, we suggest first applying the bandwidth selection method for the COPS procedure to the observed data, and then using the selected bandwidth value for the original SOPS procedure. Our simulation studies show that the performance of the proposed bandwidth-adaptive COPS and SOPS procedures is satisfactory, and not sensitive to the specification of $(h_{\min}, h_{\max}, a)$. In practice, we recommend using $(h_{\min}, h_{\max}, a) = (0.01, 0.4, 0.8)$ to allow a wide range of candidates for the bandwidth. The number of random sample-splitting is set to be $U = 20$ in all our numerical studies.

To facilitate the implementation of both procedures, we have developed a R package jra, which integrated R and C++ codes using the package Rcpp to speed up the computation. A well-documented vignette to illustrate its usage is provided together with the R package, both of which will be published online in the journal web page.

5 Numerical Studies

In this section, we investigate the finite-sample numerical performance of the proposed COPS and SOPS methods in a wide range of simulated examples. All performance measures presented in this section are obtained based on 1,000 replicated simulations. The following two different JRA models are considered. In both models, we set $t_i \overset{\text{iid}}{\sim} U(0,1)$ and $\varepsilon_i = \sigma\tilde{\varepsilon}_i$, for $i = 1, \ldots, n$, where $\tilde{\varepsilon}_i$'s are i.i.d. with $\mathbb{E}(\tilde{\varepsilon}_i) = 0$ and $\mathrm{Var}(\tilde{\varepsilon}_i) = 1$.

- (Model I) Two jumps exist at locations 0.3 and 0.7 with magnitudes $\delta_j^* = \mu$, for $j = 1, 2$. The continuity part of the regression function is set to be

$$f_{\mathrm{C}}(t) = \begin{cases} -3t + 2, & 0 \le t \le 0.3, \\ -3t + 2 - \sin\{\pi(t - 0.3)/0.2\}, & 0.3 < t \le 0.7, \\ t/2 - 0.45, & 0.7 < t \le 1. \end{cases}$$

  This model with $\mu = 1$ was considered in Xia and Qiu (2015).

- (Model II) The second model allows the number of jumps J^* to grow with n by the formula $J^* = \lfloor (\log n)^{\zeta} \rfloor$, where ζ is a parameter and $\lfloor x \rfloor$ represents the largest integer that is not greater than x. The jump locations are set to be $t_j^* = j/(J^* + 1) + U(-1, 1) \times n^{-3/5}$ and the jump magnitudes are $\delta_j^* = \mu$ with probability 0.7 and

$\delta_j^* = -\mu$ with probability 0.3, for $j = 1, \ldots, J^*$. The continuity part is $f_C(t) = \sin\{4\pi(t+0.2)\}/(t+0.2)$.

5.1 Estimate the number of true jumps

We first evaluate the performance of the proposed COPS method for estimating the number of true jumps J^*. Because this method is based on the LLK smoothing, the JIC method proposed by Xia and Qiu (2015), which is also based on the LLK smoothing, is used as a benchmark. As mentioned earlier, the choice of P_n is crucial for that method. To investigate the impact of P_n on the performance of that method, we fix $\nu = 1$ and consider the following three values for P_n: $\{nh^2(\log n)^2\}^{1/2}$, $(nh\log n)^{1/2}$ and $(nh)^{1/2}\log n$, to represent small, moderate and large penalty terms, respectively, as discussed in Xia and Qiu (2015). The corresponding JIC methods are denoted as JIC-S, JIC-M and JIC-L. In the COPS method, we use $K(u) = 1.5(1-u^2)$, for $0 \le u \le 1$. In the JIC methods, the two-sided kernel function $K_{-+}(u) = 0.75(1-u^2)$, for $-1 \le u \le 1$, is used.

To make a fair comparison among different methods and evaluate the impact of the bandwidth on their performance, each method will use a sequence of exponentially decreasing values in $\mathcal{H}$ specified in Section 4.2 for the bandwidth. The COPS method with the data-adaptive bandwidth is also considered, and it is denoted as COPS*, in which $K_{-+}(u)$ is used for estimating the regression function.

Example I: We consider Model I with $n = 1,000$, $\mu = 1$ and $\tilde{\varepsilon}_i \overset{\text{iid}}{\sim} N(0,1)$. The noise level σ is one of $\{0.4, 0.3\}$ to represent a relatively low and a relatively high level of the signal-to-noise ratio (SNR), respectively.

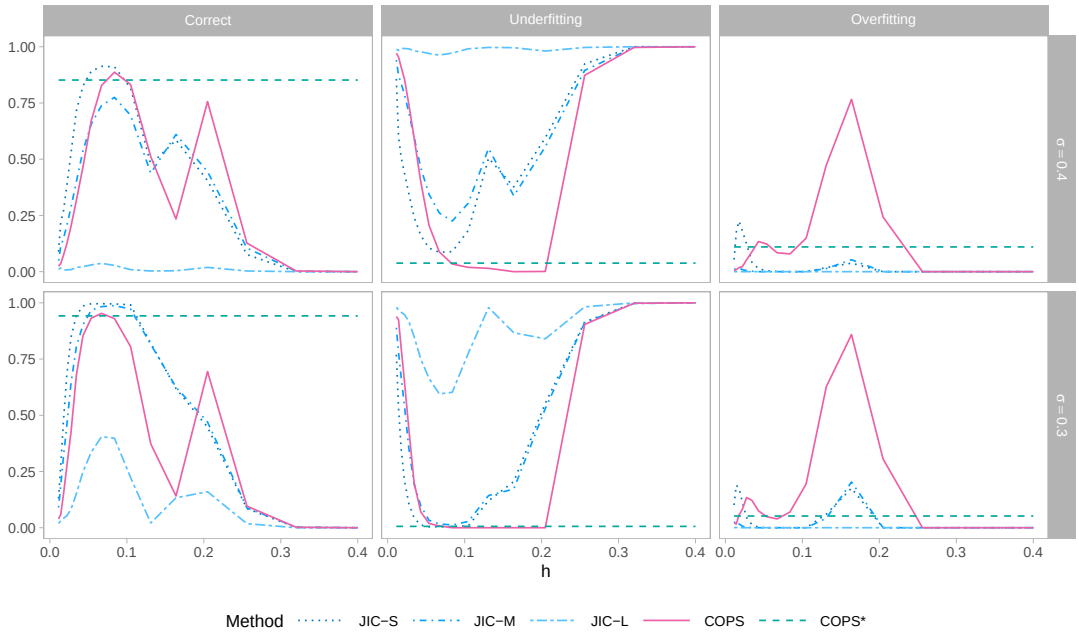

Figure 1: Probabilities of correct, under- and over-estimation of the number of true jumps by different methods in Example I when the bandwidth h changes.

Figure 1 presents the probabilities of correct, under- and over-estimation of the number of true jumps by different methods when the bandwidth h changes. From the plots in the figure, we can first observe that the impact of h on the performance of the JIC and COPS methods is quite significant. This is not surprising since bandwidth selection is important for all local smoothing methods. From the plots, it can be seen that when h is chosen too small, the model would be highly underestimated by the JIC and COPS methods because the variance of the LLK estimates are large in such cases, and consequently only the jumps with very large magnitudes can be detected by these methods. As h increases, the LLK estimates are getting more accurate and the probability of correctly identifying the number of jumps by the JIC and COPS methods would increase as well. However, when h exceeds a certain level, the probability of correctly identifying the number of jumps by these methods

declines significantly as the model tends to be overestimated. An interesting phenomenon is that the ability of correctly identifying the number of jumps by these methods seems to be revived for some large h values (e.g., near 0.16 for the JIC method when $\sigma = 0.4$). In fact, this is not the whole story if we also check the estimation precision of the detected jumps presented in Figures **??**–**??** in Supplementary Material. In such cases, it can be seen that although the number of jumps can be estimated relatively well, the estimated jump locations would differ substantially from the true jump locations. When h further increases, we get a highly underestimated model once again by these methods, since only a few (here, it is just one) candidate values are available when detecting the jumps by (4). As a comparison, the COPS method with the data-adaptive bandwidth performs well in this example. In summary, the bandwidth should not be chosen too small or too large for both the JIC and COPS methods, since such a bandwidth could result in underestimation of the number of jumps. Therefore, proper selection of the bandwidth is indeed important.

It can also be seen form Figure 1 that the performance of the JIC method indeed depends heavily on the choice of P_n. Xia and Qiu (2015) suggested using JIC-M with $P_n = (nh \log n)^{1/2}$ in practice, while we observe in Example I that JIC-S with a smaller penalty can perform better than JIC-M in many cases considered. For the JIC method, it is difficult to specify a universally applicable adjust factor P_n, since its optimal value may depend on the underlying jump regression model. In contrast, the proposed COPS method is more convenient to use since there are no tuning parameters other than the bandwidth that need to be selected in advance. Taking the bandwidth selection into consideration, the COPS method is at least comparable to, and sometimes better than the JIC method

(more evidence will be revealed in Example II), if both methods are well tuned. The COPS method in conjunction with the proposed data-adaptive bandwidth selection strategy, i.e., the COPS*, can reduce the impact of the bandwidth to a certain degree and perform robustly under different SNR levels.

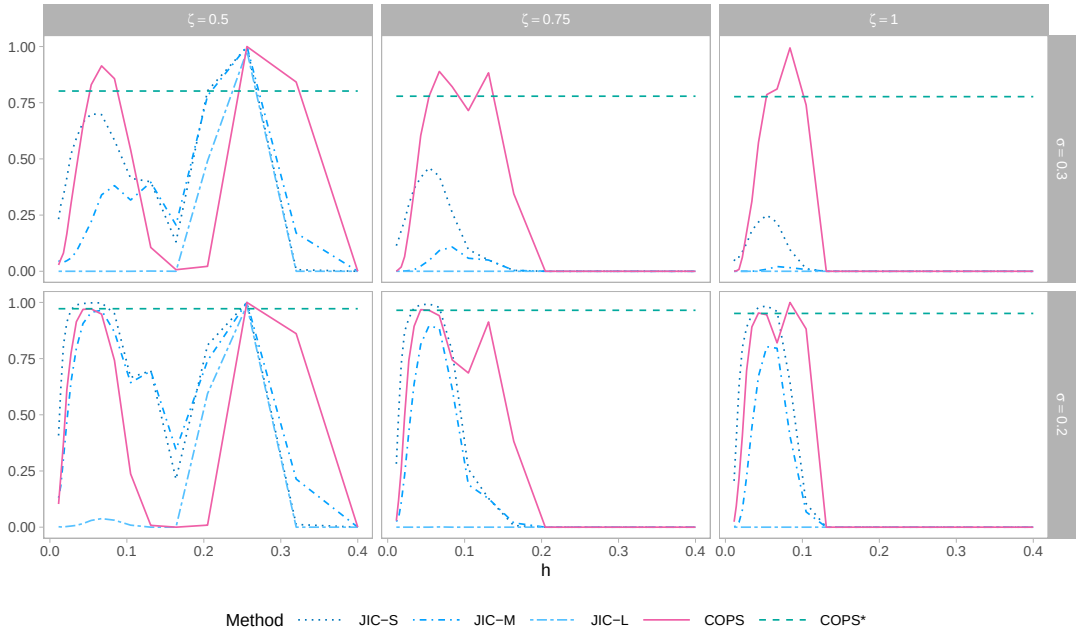

Figure 2: Probabilities of correct estimation of the number of true jumps by different methods in Example II when the bandwidth h changes. Different plots correspond to different combinations of the number of jumps and the SNR.

Example II: We consider Model II with $n = 2,000$, $\zeta \in \{0.5, 0.75, 1\}$, $\mu = 0.6$ and $\tilde{\varepsilon}_i$ being generated from the standardized t_5 distribution. The noise level σ can change between 0.3 and 0.2 to represent two different levels of SNR.

Figure 2 shows the probabilities of correct estimation of the number of true jumps by different methods when the bandwidth h changes, under different combinations of the number of jumps and the SNR. First, note that when $\zeta = 0.5$, the most favorable bandwidth

for some methods is around the value of 0.25, which should be excluded from our consideration, however, since the precision of the estimated jump locations is unsatisfactory in that case (cf., the results in Figure **??** of the Supplementary Material). Similar phenomenon has been observed in Example I. After taking this into account, it can be seen from the plots that the COPS method outperforms all three versions of the JIC method by a quite large margin if each method selects its most favorable bandwidth in the relatively low SNR scenario (i.e., $\sigma = 0.3$). In the relatively high SNR scenario (i.e., $\sigma = 0.2$), the performance of the COPS method is comparable to that of JIC-S and JIC-M, and much better than that of JIC-L. As a comparison, the COPS* method performs well in all cases considered.

Based on the numerical results in this subsection, we have the following suggestions. i) The JIC method can be used in applications when one has a prior information that the underlying SNR is relatively large and when its adjustment factor P_n can be chosen properly. ii) In cases when such prior information is unavailable, the COPS method with the proposed data-adaptive bandwidth selection strategy should be a good choice.

5.2 Select the detected jumps by controlling FDR

In this subsection, we investigate the numerical performance of the proposed SOPS method for selecting the detected jumps with FDR under a proper control.

For comparison purposes, the standard Benjamini-Hochberg (BH) FDR control method that uses the entire sample is considered here. To use the BH method, we need to specify the null distribution of $\widehat{\delta}(\widehat{t}_j)$ in cases when there are no jumps in the regression function. This is quite complicated because $\widehat{\delta}(\widehat{t}_j)$ is a local maximum of a sequence of two-sample

t-statistics and its distribution usually converges rather slowly. To address this issue, we consider using simulations to obtain an approximate distribution. Specifically, we generate data with $f = 0$ and $\tilde{\epsilon}_i \overset{\text{iid}}{\sim} N(0,1)$. The nuisance parameter σ in that method is estimated via $\hat{\sigma}^2 = \{2(n-1)\}^{-1} \sum_{i=1}^{n-1} (y_{i+1} - y_i)^2$ (Rice, 1984), which is an appropriate choice if the number and magnitudes of the jumps are not too large. The target FDR level is then fixed at $\alpha = 20\%$. The performance of the COPS, BH and SOPS methods are compared in terms of the actual FDR, TPR (i.e., the proportion of true jumps among all detected ones) and the coverage probability (CP, i.e., the proportion of true jumps that are detected).

Similar to the previous subsection, each method will use a sequence of exponentially decreasing values for the bandwidth. Thus, we can see how the bandwidth affects the performance of each method, and make a fair comparison among different methods. The bandwidth-adaptive SOPS method is also considered here, and it is denoted as SOPS*.

Example III: We consider Model II with $n = 2,000$, $J^* = 7$ and $\mu = 0.6$. The value of σ is chosen to be one of $\{0.5, 0.3\}$, and the following two noise distributions are considered: (i) $\varepsilon_i \overset{\text{iid}}{\sim} N(0,1)$, and (ii) $\{\varepsilon_i\}$ are iid with the standardized t_3 distribution.

Figure 3 shows the FDR and TPR values of different methods when the bandwidth h changes, under different combinations of the noise level and the noise distribution considered in Example III. It can be seen from the figure that the impact of h on FDR is not significant for the proposed SOPS method, and that the impact on its TPR is much more significant. Let us study the results in cases with the $N(0,1)$ noise distribution. In such cases, if h is chosen too small, both FDR and TPR are very small for the BH and SOPS methods, since they rarely report "positives" (i.e., detected jumps). This fact is also true

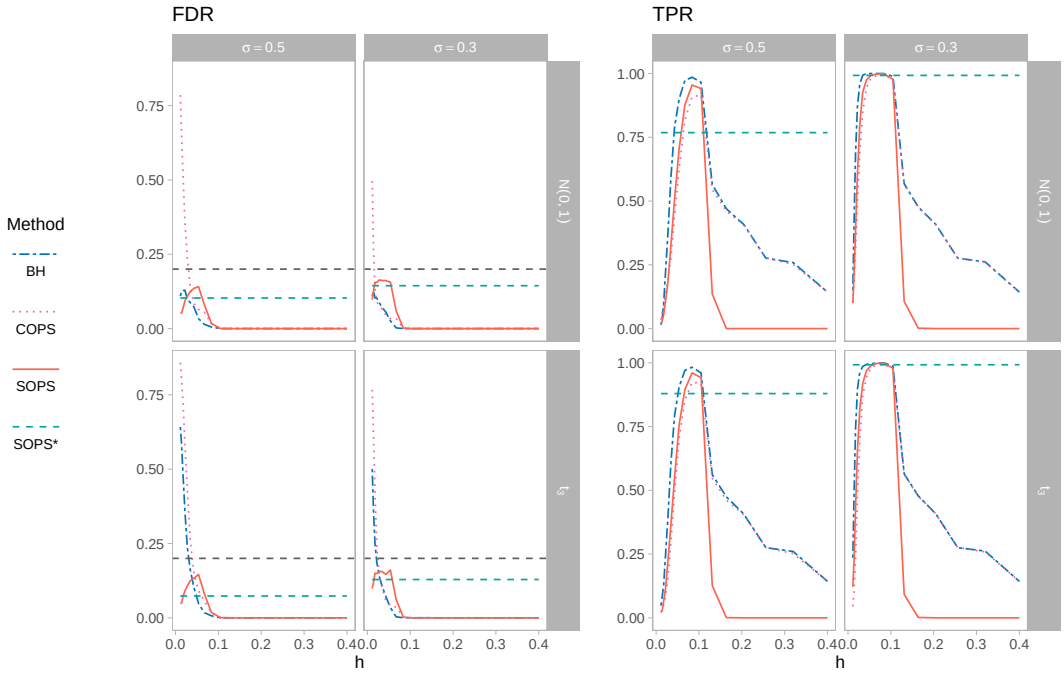

Figure 3: FDR and TPR values of different methods when the bandwidth h changes, under different combinations of the noise level and the noise distribution considered in Example III.

for the COPS method, although its FDR is unnecessarily large in those cases because that method can always detect at least one jump, which could be a false jump. If h is chosen too large, then the COPS and BH methods could detect some "true positives" (informative points) without detecting any "false positives" (uninformative points). However, the SOPS method is less tolerant to large bandwidth values and it tends to report no positives. As a comparison, the bandwidth-adaptive SOPS method, i.e., SOPS*, performs well in all cases considered. Its FDR values are close to the nominal level, and its TPR values are comparable to the best TPR values of the other methods.

Figure **??** in Supplementary Material shows the coverage probability (CP) against the bandwidth for different methods in the cases considered above. Results in that figure show

that the BH and SOPS methods have larger CP values than those of the COPS method. The CP for the SOPS* method is reasonably good in all cases considered, although it sacrifices some efficiency for such a robustness porperty, as shown in Figure 3.

5.3 A real-data example

In this part, we apply the proposed methods to a financial time series data to demonstrate their use in a real-data setup. The dataset includes weekly open price of the Dow Jones Industrial Average over the period 02/04/1985–03/29/2021. The data can be downloaded from the Yahoo Finance. For this dataset, we are interested in detecting possible jumps in the underlying regression function of the open prices of the Dow Jones Industrial Average.

Figure **??** in Supplementary Material shows the estimated numbers of jumps against the value of the bandwidth for the three versions of JIC, the COPS and SOPS methods considered in Sections 5.1–5.2. For the SOPS procedure, we use $\alpha = 20\%$. It can be seen that the detected numbers of jumps are close to each other among all methods considered for a given bandwidth, and for a given method the estimated number of jumps decreases significantly as the bandwidth increases.

Next, we focus on the results of the COPS* and SOPS* methods, which are the bandwidth-adaptive versions of the proposed COPS and SOPS procedures, respectively. The dates corresponding to the detected jumps are annotated in Figure 4 by small triangles and dots located around the horizontal line of $y = 20,000$, where the results of the two methods are comparable in this setting. To save some space, the specific dates of the detected jumps and the corresponding jump magnitudes are listed in Table **??** in Supplemen-

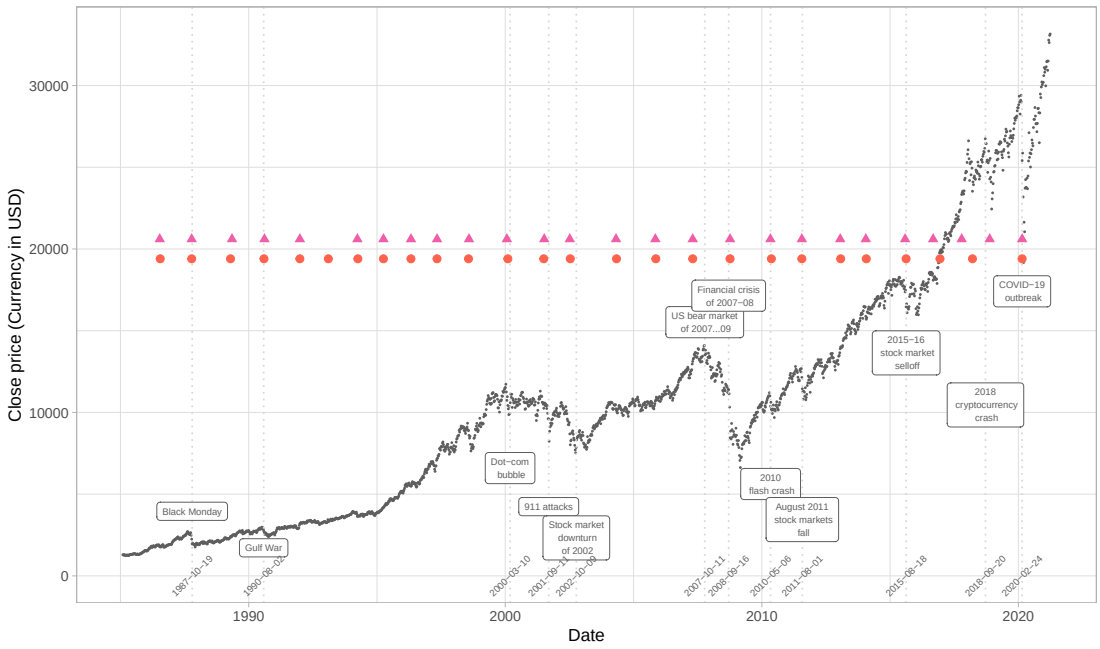

Figure 4: The original time series data is plotted as grey dots. Vertical dotted lines represent some notorious time points of stock market crashes or bear markets as indicated by the labeled names. Jumps detected by the proposed methods are also pointed around the line $y = 20,000$ (▲ : COPS*; ● : SOPS*).

tary Material. Figure 4 also presents the original time series data, and some notorious time points of stock market crashes or bear markets (cf. https://en.wikipedia.org/wiki/List_of_stock_market_crashes_and_bear_markets) that are indicated by the vertical dotted lines. It can be seen that some of the detected jumps are close to the dates of such stock market crashes. In addition, there are three detected jumps with positive jump magnitudes, which indicate a bull market at these three time points.

The data considered here are collected discretely by weeks. It should be interesting to study whether our methods, after some necessary modifications, could also be used for detecting jumps in continuously sampled financial time series data. See related discussions

in Bajgrowicz et al. (2016); Scaillet et al. (2020).

6 Concluding Remarks

Determination of the number of jumps is an important but challenging problem for jump regression analysis. With the help of the order-preserved sample-splitting strategy, this paper introduces two data-driven methods (i.e., COPS and SOPS) for solving this problem. The COPS method is based on the cross-validation idea and capable to provide a consistent estimate of the number of jumps under some regularity conditions, including the one that the jump magnitudes are not too small. When this condition on jump magnitudes is questionable in a specific application, the SOPS method can be considered, which often results in a slightly overfitting model without excessive false positives. The SOPS method is shown to control the FDR well; thus, it could serve as a useful alternative to the COPS method for practical use.

There are still some issues related to the proposed methods that need to be addressed in the future research. For instance, it is of great importance to theoretically investigate whether the proposed data-adaptive strategy for bandwidth selection could yield consistent estimate of the number of jumps or achieve the FDR control property. Also, both COPS and SOPS methods assume that the observed data are independent at different time points. It is necessary to generalize the proposed methods to cases with autocorrelated data. Last but not the least, the current paper focuses on 1-D cases. We expect that the proposed methods in this paper can be generalized to 2-D cases to detect jump location curves, which certainly warrants much future research.

Supplementary Material

The supplementary material contains a list of assumptions and proofs of Theorems 1 and 2, along with some additional numerical results. The jra package that implements the proposed COPS and SOPS methods can be found in the attached file jra_0.0.0.9000.tar.gz. A demonstration example for illustrating the use of the jra package is also included in that file.

Acknowledgments

The authors are grateful to the Editor, the Associate Editor and two referees for their insightful comments which have greatly improved the quality of the article.

References

Bajgrowicz, P., Scaillet, O., and Treccani, A. (2016), "Jumps in High-Frequency Data: Spurious Detections, Dynamics, and News," *Management Science*, 62, 2198–2217.

Barber, R. F., and Candès, E. J. (2015), "Controlling the false discovery rate via knockoffs," *Ann. Statist.*, 43, 2055–2085.

Benjamini, Y., and Hochberg, Y. (1995), "Controlling the false discovery rate: a practical and powerful approach to multiple testing," *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, 57, 289–300.

Delgado, M. A., and Hidalgo, J. (2000), "Nonparametric inference on structural breaks," *Journal of Econometrics*, 96, 113–144.

Fan, J., and Gijbels, I. (1996), *Local polynomial modelling and its applications*, vol. 66 of *Monographs on Statistics and Applied Probability*, Chapman & Hall.

Fan, J., and Marron, J. S. (1994), "Fast Implementations of Nonparametric Curve Estimators," *Journal of Computational and Graphical Statistics*, 3, 35.

Frick, K., Munk, A., and Sieling, H. (2014), "Multiscale change point inference," *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, 76, 495–580.

Gijbels, I., and Goderniaux, A.-C. (2004), "Bandwidth Selection for Changepoint Estimation in Nonparametric Regression," *Technometrics*, 46, 76–86.

Gijbels, I., Hall, P., and Kneip, A. (1999), "On the estimation of jump points in smooth curves," *Ann. Inst. Statist. Math.*, 51, 231–251.

Grégoire, G., and Hamrouni, Z. (2002), "Change point estimation by local linear smoothing," *J. Multivariate Anal.*, 83, 56–83.

Hao, N., Niu, Y. S., and Zhang, H. (2013), "Multiple change-point detection via a screening and ranking algorithm," *Statist. Sinica*, 23, 1553–1572.

Hawkins, D. M. (2001), "Fitting multiple change-point models to data," *Comput. Statist. Data Anal.*, 37, 323–341.

Li, H., Munk, A., and Sieling, H. (2016), "FDR-control in multiscale change-point segmentation," *Electron. J. Stat.*, 10, 918–959.

Loader, C. R. (1996), "Change point estimation using nonparametric regression," *Ann. Statist.*, 24, 1667–1678.

Müller, H.-G. (1992), "Change-points in nonparametric regression analysis," *Ann. Statist.*, 20, 737–761.

Müller, H.-G., and Stadtmüller, U. (1999), "Discontinuous versus smooth regression," *Ann. Statist.*, 27, 299–337.

Qiu, P. (1994), "Estimation of the number of jumps of the jump regression functions," *Comm. Statist. Theory Methods*, 23, 2141–2155.

— (2005), *Image processing and jump regression analysis*, Wiley Series in Probability and Statistics, John Wiley & Sons.

Rice, J. (1984), "Bandwidth choice for nonparametric regression," *Ann. Statist.*, 12, 1215–1230.

Scaillet, O., Treccani, A., and Trevisan, C. (2020), "High-Frequency Jump Analysis of the Bitcoin Market," *Journal of Financial Econometrics*, 18, 209–232.

Schwarz, G. (1978), "Estimating the dimension of a model," *Ann. Statist.*, 6, 461–464.

Siegmund, D. O., Zhang, N. R., and Yakir, B. (2011), "False discovery rate for scanning statistics," *Biometrika*, 98, 979–985.

Spokoiny, V. G. (1998), "Estimation of a function with discontinuities via local polynomial fit with an adaptive window choice," *The Annals of Statistics*, 26, 1356–1378.

Wu, J. S., and Chu, C. K. (1993a), "Kernel-type estimators of jump points and values of a regression function," *Ann. Statist.*, 21, 1545–1566.

Wu, J. S.— (1993b), "Nonparametric function estimation and bandwidth selection for discontinuous regression functions," *Statist. Sinica*, 3, 557–576.

Xia, Z., and Qiu, P. (2015), "Jump information criterion for statistical inference in estimating discontinuous curves," *Biometrika*, 102, 397–408.

Zou, C., Wang, G., and Li, R. (2020), "Consistent selection of the number of change-points via sample-splitting," *Ann. Statist.*, 48, 413–439.