CHAPTER 4

A NEW TWO-STAGE METHOD FOR NONPARAMETRIC REGRESSION WITH JUMP POINTS

C.Z. Wu[*], C.M. Liu**, K.L. Teo** and Q.X. Shao[***]

[*]*School of Mathematics and Computer Science,*
Chongqing Normal University, Chongqing, 40047, China.
E-mail:changzhiwu@yahoo.com
[**]*Department of Mathematics and Statistics,*
Curtin University of Technology, Perth, WA, 6102, Australia.
E-mail:jimmy_uwa@yahoo.com.tw and k.l.teo@curtin.edu.au
[***]*CSIRO Mathematical and Information Sciences,*
WA, 6104, Australia.
E-mail: Quanxi.Shao@csiro.au

In this chapter, a two-staged method is presented for nonparametric regression with jump points. After the rough location of all the possible jump points are identified using the existing efficient kernel method, a smoothing spline function is used to approximate each segment of the regression function. A time scaling transformation is derived so as to map the undecided jump points into fixed points. This approximation problem is formulated as an optimization problem which can be solved by many existing techniques. The method is applied to several examples. The results obtained show that the method is highly efficient.

4.1. Introduction

Statistical modelling generally assumes smoothness and continuity of the phenomena of interest. However, some phenomena may experience sudden or sharp change. For example, groundwater table may undergo drastic changes in very short periods of time [15] due to sudden changes in environment, such as land clearing. When we study the portfolio management, the amount of stocks of a particular investor can be viewed

as experiencing a jump when he/she purchases or sells his/her stocks. This sudden change is reflected as a jump in visual display. Without considering these jumps, we may make an serious error in drawing inference about the process under study. It is clearly important to estimate both the number of jumps and their locations and magnitudes.

Problems related to regression with jump points have been addressed and investigated for more than two decades. [11] proposed an algorithm to estimate regression functions when discontinuities are present. [20] used one-sided moving average to find the locations of jumps, [9] used Fourier analysis for jump detection. [8] made use of pairing pattern and linear filter to develop a discontinuity detector for the purpose of detecting the jumps and their magnitudes. [13] estimated the locations and magnitudes of the jumps by the boundary kernels. [19] used the kernel method to estimate both the locations and magnitudes of the jumps. [18] used wavelets to provide a useful procedure for the detection of the jump points and their magnitudes. [14] proposed a two-stage estimation scheme for the jump locations, where the asymptotic properties are also studied. Most of the estimators mentioned above are of the kernel type. The key idea is to investigate the difference between the estimators of the left- and right-hand side limits for the unknown regression functions. On this basis, the locations and magnitudes of the jumps can be estimated by using the maximization argument of the jump points estimation. However, the overall fittings obtained using these methods are not very satisfactory at around jump points and around end points.

At the same time, the spline method was also applied for detecting the jump points. [7] used linear splines to estimate discontinuities of the unknown regression functions. [12] applied the evolutionary algorithm to locate the optimal knots.

These methods are computationally expensive. In this chapter, a two-stage method is developed for detecting the number of jump points, their locations and magnitudes as well as finding the spline representations for the approximation of the unknown regression function with jump points. The following notations are used in this chapter. Let X be the response variable with respect to variable t. For a given set of observations $\{x_i, t_i\}$, $i = 1, 2, \cdots, n$, the regression function is written as:

$$x_i = m(t_i) + \varepsilon_i, \qquad i = 1, 2, \cdots, n \tag{4.1}$$

where m is an unknown function defined on $[0, T]$, $\{\varepsilon_i\}$, $i = 1, 2, \cdots, n$, are independent and identically distributed $N(0, \sigma^2)$ normal random variables. For each $i = 1, 2, \cdots, n$, $\{\varepsilon_i\}$ represents the variation of x_i around $m(t_i)$. Without loss of generality, we can let $T = 1$. Otherwise, a re-scaling of T can be used.

To form a segmented regression, we write $m(t)$ as:

$$m(t) = m_l(t), \quad \text{for } \tau_{l-1} < t \le \tau_l, \ l = 1, \cdots, N, \tag{4.2}$$

where $\tau_0 = 0$ and $\tau_N = T$, where τ_i, $i = 1, \cdots, N$, are jump points. This setting covers the case of the change of mean values (step changes) as a special case by choosing $m_l(t)$ as constants. Note that the form of the regression $m_l(t)$ is usually unknown, and hence are nonparametric. It is well-known that spline functions are effective for approximating nonparametric regressions if they are smooth. However, the smoothing spline does not work well directly for regressions with jump points, since the smoothing spline is rather sensitive to the location of jump points. Some difficulties in jump point estimation have been demonstrated in [6] for linear regression with only one jump point. Intuitively, the jump points should be identified before applying the smoothing spline to each of the individual segments defined by the jump points. This is the motivation behind our proposed two-stage method.

The proposed method consists of two steps. First, we should locate all potential locations of jump points. Then, we use a time scaling transformation to transform the potential jump points and spline knots into pre-fixed points. By doing this, the parameter estimation using least squares becomes an optimization problem, which can be solved by efficiently techniques, such as the sequential quadratic approximation optimization method. Finally, the modified Akaike's information criterion is used to determine which potential jump points are real jump points. We then obtain the final model.

This chapter is organized as follows. In section 4.2, we employ the kernel method proposed by [19] to locate locations of all the potential possible jump points to within respective observation points. Then, the

problem formulation is given in section 4.3. section 4.4 introduces a time scaling transform, which is motivated by the one proposed in [16]. This time scaling transform is then used to derive a new time scaling transform, which maps the undecided jump points and spline knots into fixed points. The obtained optimization problem is solved by the sequential quadratic approximation optimization technique in the Optimization Toolbox within the Matlab environment. In section 4.5, we use the modified Akaike's information criterion to determine the number and locations of jump points and therefore obtain the last model. In section 4.6, several numerical examples are presented and solved by using the proposed method. Some concluding remarks are given in section 4.7.

4.2. Estimation for Potential Jump Points

In statistics, the kernel method is an efficient and powerful statistical tool to detect jump points. Therefore, it is used in our two-stage method to find potential jumps. We assume that the following conditions are satisfied.

1) The number of the observation points is sufficiently large to detect all the possible jump points. That is, the number of jump points is very small in comparison with the number of observation points and the number of observation points between consecutive jump points is sufficiently large for spline fitting.
2) The jump points should be located in the interval $[\delta, 1-\delta]$, where δ is some small positive constant. That is, there is no jump point in the neighborhoods of the boundary points on which they become undetectable.
3) There is at most one jump point between t_i and t_{i+1}, $i = 0,1,\cdots,n,$ where $t_0 = 0$ and $t_{n+1} = 1$.
4) The regression errors ε_i, $i = 0,1,\cdots,n,$ are independent and identically distributed (*i.i.d*) random variables with mean 0 and variance $\sigma^2 < \infty$.

Let K be the kernel function with the bandwidth h. There are two popular methods to construct kernel estimators, depending on the choice of weights by either direct kernel evaluation or the convolution of the

kernel with a histogram obtained from the data. Each estimator has several important advantages and disadvantages. A thorough review on this subject can be found in the paper by [2]. [4] make use of the first method to construct an estimator to detect the vicinity of the jump points. In our two-stage method, the second method is used to identify rough positions of the jump points.

For the kernel function K and the bandwidth h, the Gasser-Müller estimator is defined as:

$$\hat{m}(t)=\sum_{i=1}^{n} x_i \int_{s_{i-1}}^{s_i} K_h(t-\tau)d\tau, \quad \text{for } t\in(0,1), \tag{4.3}$$

where $K_h(\bullet)=h^{-1}K(\bullet/h)$, $s_n=0$, $s_i=(t_i+t_{i+1})/2$, $i=2,\cdots,n-1$, and $s_n=1$. Let $m_1(t)$ and $m_2(t)$ be two Gasser-Müller estimators obtained with the kernel functions K_1 and K_2, respectively, using the same bandwidth h. Let

$$J(t)=\hat{m}_1(t)-\hat{m}_2(t). \tag{4.4}$$

To continue, we need to analyze the value of $|J(t)|$.

If $m(t)$ has no jump point, then, under the usual regularity conditions, $m_1(t)$ and $m_2(t)$ are uniformly strongly consistent estimators of $m(t)$. Thus, the variation of $|J(t)|$ would not be obvious. If $m(t)$ has a jump point, then the value of $|J(t)|$ would have an obvious change around the jump point.

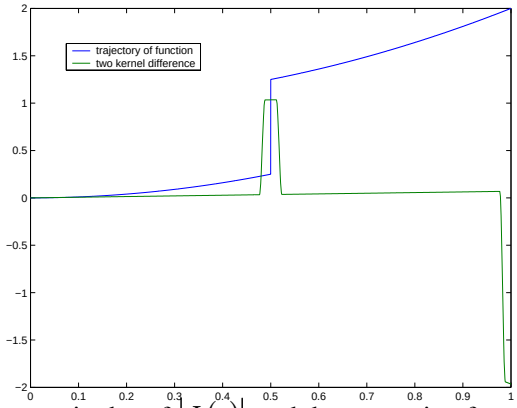

Fig. 4.1. The magnitudes of $|J(t)|$ and the regression function.

In Fig. 4.1., we give an example where $m(t)$ has a jump at $t=1/2$.

Write

$$m(t) = \varphi(t) + \psi(t), \tag{4.5}$$

where $\varphi(t)$ is the continuous part of $m(t)$, while $\psi(t)$ is a step function to characterize the jumps of $m(t)$. The magnitude of $|J(t)|$ corresponding to $\varphi(t)$ is of small order. However, the magnitude corresponding to $\psi(t)$ is symmetric and convex downward in the neighborhood of the jump point if the kernel functions K_1 and K_2 are chosen such that

$$K_1(t) = K_2(-t), \tag{4.6}$$

Furthermore, if K_1 and K_2 have compact support in $[-1,1]$, then the widths of the neighborhoods mentioned above are no more than $2h$. Based on the above discussions, we can give a numerical procedure to estimate rough locations (and hence, number) of the jump points.

Algorithm 4.1. (Kernel method to detect the potential jump points)

Step 1: Choose h such that $h = O(n^{-1/3})$.

Step 2: Choose a nonnegative function $K(t)$ with a compact support in [-1,1] and is such that $\int_0^1 K \neq \int_{-1}^0 K$.

Step 3: Let $K_1(t) = K(t)$ and $K_2(t) = K(-t)$. Calculate $J(t)$ by using (4.3) and (4.4), where h is given in Step 1.

Step 4: Find all the points, which correspond to local maxima of $|J(t)|$.

The points obtained by Algorithm 4.1 are considered as potential jump points due to Step 4. In the next section, we will use a time scaling transformation to find the accurate positions of these jump points.

4.3. Segmented Regression with Constraints

Suppose that $\tau_1, \tau_2, \cdots, \tau_{N-1}$, is a set of potential jump points (obtained from the kernel method crudely). Then, the interval $[0,1]$ has been partitioned into N subintervals $[\tau_{l-1}, \tau_l]$, $l = 1, \cdots, N$. For each

$l = 1, \cdots, N$, let the observation points contained in the subinterval $[\tau_{l-1}, \tau_l]$ be denoted by $t_{1,1}, \cdots, t_{1,N_1}$, and let $x_{1,1}, \cdots x_{1,N_1}$ be the corresponding observations. The regression function is denoted by $m_l(t)$ in the subinterval $(\tau_{l-1}, \tau_l]$ where $l = 1, \cdots, N$, $\tau_0 = 0$, and $\tau_N = T$. Now, we use the segmented regression to fit the segment $m_l(t)$, $l = 1, \cdots, N$.

The most widely used approach to curve fitting is least squares. If we place no restrictions on the residual sum of squares, this method is in fact an intoporlant which may be caused a rapid fluctuation. To avoid this, we incorporate a smooth penalty in the cost function. In this chapter, we will introduce the cubic spline for fitting the segment $m_l(t)$.

A general cubic spline basis is defined as

$$\left\{1, t, t^2, t^3, (t-\theta_1)_+^3, \cdots, (t-\theta_K)_+^3\right\}. \tag{4.7}$$

where $(t-\theta_k)_+ = \max(0, t-\theta_k)$, θ_k, $k = 1, \cdots, K$, are the knot points. Since the smoothness is penalized in the cost, we need the estimator $\hat{m}_l(t)$ of $m_l(t)$ is continuously differentiable at the knot θ_k, $k = 1, \cdots, K$. Write

$$\hat{m}_l(t) = a_{l,1} + \sum_{k=1}^{3} a_{l,k+1} t^k + \sum_{k=1}^{i_l} a_{l,k+4} (t-\theta_{l,k})_+^3, \tag{4.8}$$

where $\theta_{l,1}, \cdots, \theta_{l,i_l}$, are the pre-fixed knots contained in the *l*-th segment, $a_{l,1}, \cdots, a_{l,l+4}$, are the coefficients. Since $(t-\theta_k)_+^3$ is twice continuously differentiable, there is no any need restrictions on $m_l(t)$. Define the cost functional as the following:

$$J(\boldsymbol{\tau}, \mathbf{a}) = \sum_{l=1}^{N} \sum_{i=1}^{N_l} \left(\hat{m}_l(t_{l,i}) - x_{l,i}\right)^2 + \lambda \sum_{i=1}^{N} \int_{\tau_{i-1}}^{\tau_i} \left(\hat{m}_i''(t)\right)^2 dt \tag{4.9}$$

where $\boldsymbol{\tau} = [\tau_1, \cdots, \tau_{N-1}]^T$, $\mathbf{a} = \left[a_{1,1}, \cdots, a_{1,i_1+4}, \cdots, a_{N,i_N+4}\right]^T$, λ is the smoothness parameter.

Under the least squares method with the smoothness penalty, our objective is to find $(\boldsymbol{\tau}, \mathbf{a})$ such that (4.9) is minimized subject to the following constraints

$$t_{l,N_l} \le \tau_l \le t_{l+1,1}. \tag{4.10}$$

For this optimization problem, the estimates of the jump locations and the optimal regression coefficients are obtained simultaneously. Let this problem be referred to as Problem (P).

For the penalized parameter λ, we can choose it by the generalized cross-validation method [3]. However, here we will choose it interactively for simplicity.

To solve Problem (P), we need to compute the cost. We note that the cost (4.9) is composed of two parts. Since the first part is only related to the coefficient vector $\mathbf{a}$, it is easily computed. The second part is the sum of some integrals with the integral limit related to the jump points. Thus, to compute it with its corresponding gradient is hard. To overcome this difficulty, we just introduce a time scaling transformation. By this transformation, the jump points and the spline knots are all mapped into some pre-fixed points.

4.4. A Time Scaling Transformation Method

We suppose that $\tau_1,\cdots,\tau_N$, are N variable times in the time interval $[0,T]$. A time scaling transformation is introduced such that the variable times $\tau_1,\cdots,\tau_N$, are transformed to pre-fixed times $\xi_1,\cdots,\xi_N$, in the new time scale. This transformation was known as the enhancing transform, which was introduced by [10] to overcome the numerical difficulties in the computation of some optimal control problems. It is defined by

$$\frac{dt}{ds} = v(s), \tag{4.11}$$

with initial condition

$$t(0) = 0,\ \frac{dt}{ds} = v(s), \tag{4.12}$$

where $v(s)$ satisfies the following conditions:

- $v(s) \ge 0$ for all s;
- $v(s)$ is a piecewise constant on the interval $(\xi_{i-1}, \xi_i]$;

$$\int_{\xi_{i-1}}^{\xi_i} v(s)\,ds = \tau_i - \tau_{i-1}\,, \tag{4.13}$$

where $\tau_0 = 0$, $\xi_0 = 0$.

Now we apply this transformation to our problem such that the jump points $\tau_1, \cdots, \tau_{N-1}$, are mapped into the fixed points $1, \cdots, N-1$, and the spline knots $\theta_{l,1}, \cdots, \theta_{l,i_l}$, are mapped into $l+1/(i_l+1), \cdots, l+i_l/(i_l+1)$, $l = 1, \cdots, N$. To achieve it, we just choose

$$v(s) = \sum_{k=1}^{N}\sum_{j=1}^{i_k+1} \xi_{k,j}\chi_{\left[k+(j-1)/(i_{k+1}),k+j/(i_{k+1})\right)}$$

where χ_I is the indicator function, *i.e.*,

$$\chi_I(s) = \begin{cases} 1, & \textit{if} \quad s \in I, \\ 0, & \textit{otherwise.} \end{cases}$$

$\xi_{k,j}$, $k = 1, \cdots, N$, $j = 1, \cdots, i_k$, satisfy the following conditions:

$$\xi_{k,j} \geq 0; k = 1, \cdots, N,\ j = 1, \cdots, i_k; \tag{4.14}$$

$$\sum_{k=1}^{N}\sum_{j=1}^{i_k+1} \xi_{k,j} = T; \tag{4.15}$$

$$\xi_{k,j} = (i_k+1)\left(\theta_{k,j} - \theta_{k,j-1}\right), k = 1, \cdots, N,\ j = 2, \cdots, i_k; \tag{4.16}$$

$$\frac{1}{i_k+1}\xi_{k,i_k+1} + \frac{1}{i_{k+1}+1}\xi_{k+1,1} = \theta_{k+1,1} - \theta_{k,i_k}, k = 1, \cdots, N. \tag{4.17}$$

By this transformation, we have

$$\sum_{i=1}^{N}\int_{\tau_{i-1}}^{\tau_i} \left(m_i''(t)\right)^2 dt = \sum_{i=1}^{N}\int_{i-1}^{i} \left(m_i''(t)\right)^2 v(s)\,ds. \tag{4.18}$$

Let

$$\overline{J}(\boldsymbol{\xi}, \mathbf{a}) = \sum_{l=1}^{N}\sum_{i=1}^{N_l}\left(\hat{m}_l(t_{l,i}) - x_{l,i}\right)^2 + \lambda\sum_{l=1}^{N}\int_{l-1}^{l}\left(m_l''(t(s))\right)^2 v(s)\,ds \tag{4.19}$$

where $\boldsymbol{\xi} = \left[\xi_{1,1}, \cdots, \xi_{1,i_1+1}, \cdots, \xi_{N,i_N+1}\right]^T$. Problem ($\hat{P}$) is stated as the following.

Problem ($\hat{P}$). $\min \overline{J}(\boldsymbol{\xi}, \mathbf{a})$ subject to (4.14), (4.15), (4.16) and (4.17).

Clearly, we have the following theorem.

Theorem 4.1. Problem (P) is equivalent to Problem ($\hat{P}$) in the sense that $(\boldsymbol{\tau}, \mathbf{a})$ is the optimal solution of Problem (P) if and only if $(\boldsymbol{\xi}, \mathbf{a})$ is an optimal solution of Problem (P). Furthermore, they have the same optimal cost.

4.5. Model Selection

Note that the kernel method in section 4.2 tries to detect all the potential jump points. The spline approximation method outlined in section 4.3 can be used in conjunction with the time transform introduced in section 4.4 to any possible combinations of potential jump points. Suppose that the number p is the candidate of jump points. Then, the number of jump points p is required to be chosen appropriately. Several methods for choosing it have been suggested in the literature. We propose to use the Akaike Information Criterion (AIC) [1] for the purpose of model selection. Tong [17] applied AIC for model selection in a threshold autoregression. Hurvich and Tsai [5] derived a modified Akaike information criterion AIC_c in small samples and claimed that the AIC_c dramatically reduces the bias and improves model selection. The AIC_c penalizes the RSS by a function of the number of free parameters and is given by

$$AIC_c = \ln \frac{RSS}{N} + \frac{N+p}{N+p-2}, \tag{4.20}$$

where N is the number of observations, p is the number of free parameters in the model and RSS is the first part of the cost (4.9). Note that AIC tends to overestimate the number of parameters, thus, we will use BIC as a criterion instead of AIC when the number of observation points is large enough, where

$$BIC = \ln \frac{RSS}{N} + \frac{p \ln N}{N}. \tag{4.21}$$

In our simulation, for those examples, where the data produced from a mathematical function, BIC will be used as the criterion as the data can be collected as much as we need. For the real data, we will use AIC_c as the criterion since the observation points are limited.

Algorithm 4.2.

1. Set $\boldsymbol{\tau} = \left[\tau_1, \cdots, \tau_{N-1}\right]^T$, which is a set of candidate jump points, Γ is a set to store the solution and set it empty, $AIC_c = 1000.$
2. Choosing candidate jump points $\left\{\tau_i\right\}_{i=1}^{m} \subset \boldsymbol{\tau}$ and for given spline knots θ, use the time scaling transformation (4.11) to solve ($\hat{P}$) and evaluate the corresponding AIC_c (or BIC), Let Γ' denotes the current solution which is the set of the jump points and the regression coefficients.
3. If $AIC_c^{'} \le AIC_c$, set $\Gamma = \Gamma'$, $AIC_c = AIC_c^{'}$, and go to Step 2. If there is no possible candidate jump points, stop.

4.6. Numerical Example

To assess the performance of our proposed method, some numerical examples are presented here.

For the simulated study, the data were generated by the following equation:

$$x_i = m\left(t_i\right) + \varepsilon_i, \tag{4.22}$$

with $t_i = i/n$ and ε_i sampled randomly from a normal distribution with standard deviation σ_ε. In the next numerical examples, we choose $\lambda = 0.1$.

Example 4.1. Let $n = 200$, $\sigma_\varepsilon = 0.2$, and the data is produced from the function:

$$m\left(t\right) = 2 - 2\left|t - 0.26\right|^{1/5} 1\left(t \le 0.26\right) - 2\left|t - 0.26\right|^{3/5} 1\left(t > 0.26\right) + 1\left(t \ge 0.78\right). \tag{4.23}$$

First, we use Algorithm 4.1 to detect the rough locations of the jump points. We rearrange the observation points with the given candidate jump points. Let the spline knots vector be [0.1, 0.15, 0.3, 0.5, 0.6]. We use BIC as criterion and the obtained results are that there are two jump points: 0.25225, 0.78178. The obtained BIC is -4.1056. The result is depicted in Fig. 4.2. We do the same simulation for 100 times. The

results are depicted in Fig. 4.3.

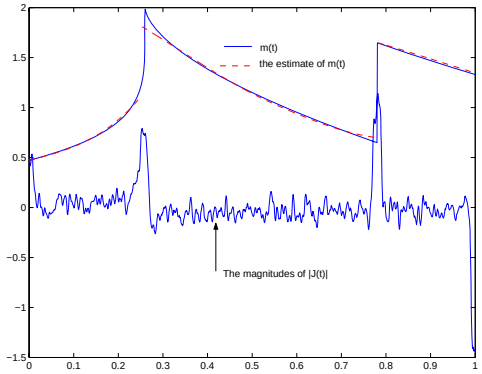

Fig. 4.2. Example 4.1. with $\lambda = 0.01$.

If we choose λ=10, then the obtained result is depicted in Fig. 4.4. From Fig. 4.4, we can see that if λ is enough, then the fitting becomes linear fitting. Thus, the parameter λ controls the gradient change rate of the splines.

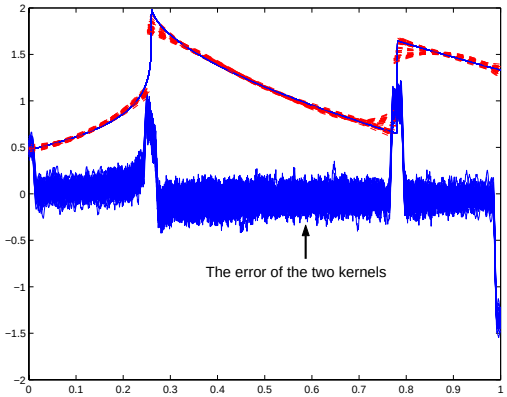

Fig. 4.3. Example 4.1. with 100 times simulation.

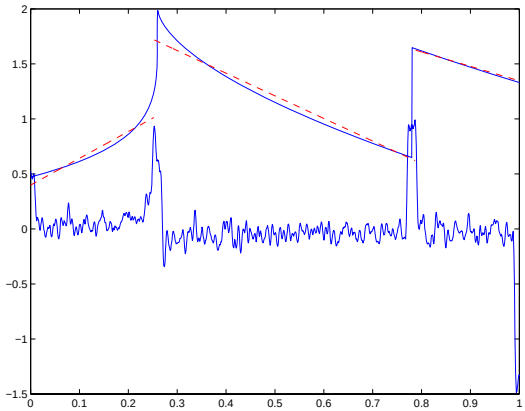

Fig. 4.4. Example 4.1. with $\lambda = 10$.

Example 4.2. We now apply our method to a real example, which contains the water levels (depths) in boreholes monitored irregularly over time. The data are taken from the database of Agriculture of Western Australia and have been analyzed by [15]. There are 49 observations. The observation points as well as its jump estimator by Algorithm 4.1 is depicted in Fig. 4.5. We can see that the kernel method cannot present an good jump estimator since the observation points is too little from the figure. Thus, we have to consider all the potential jump points and use AIC_c as criterion to choose the best model. In the process of fitting, we suppose that all of the jumps are positive since the level of the groundwater does not experience drop instantly. We re-scaling the time to the new interval [0,1]. The last obtained model has 3 jump points: 0.1050,0.2727,0.4554, $AIC_c = 4.3148$. The obtained results are depicted in Fig. 4.6.

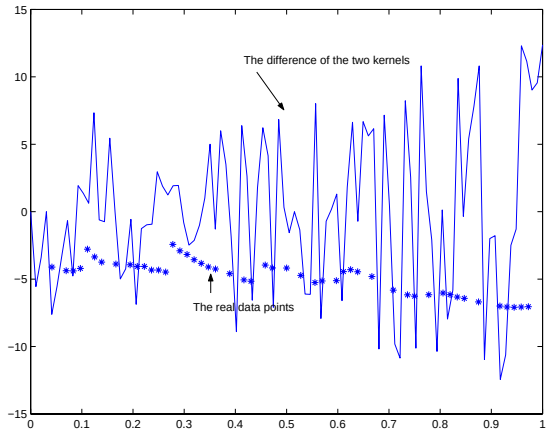

Fig. 4.5. Observation points and jump estimator of Example 4.2.

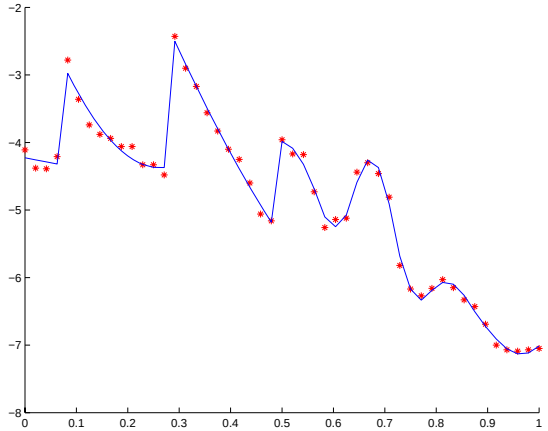

Fig. 4.6. The fitting result of Example 4.2.

4.7. Conclusion

In this chapter, a new two stage method has been proposed for solving spline regression problem with jump points. First, we detect the rough locations of jump points based on the kernel method. Then, we introduce a time scaling transformation reformulate our regression problem as a nonlinear optimization problem which is easily to be solved. Also, some numerical results are presented and which are shown that our proposed method is efficiency.

References

[1] H. Akaike, A new look at the statistical model identification, *IEEE Transaction on Automatic Control* **19**, 716-723, (1974).

[2] C. K. Chu and J. S. Marron, Choosing a kernel regression estimators, *Statistical Science* **6**, 404-419, (1991).

[3] P. Craven, and G. Wahba, Smoothing noisy data with spline functions, *Numerical Mathematics* **31**, 377-403, (1979).

[4] I. Gigbels, H. Peter and A. Kneip, On the estimation of jump points in smooth curves, *Annales Institue of Statistics and Mathematics* **51**, 231-251, (1999).

[5] M. Hurvich, and C. L. Tsai, Bias of the corrected AIC criterion for underfitted regression and time series models, *Biometrika* **78**, 499-509, (1991).

[6] S. Julious, Inference and estimation in a change-point regression problem, *Journal of Royal Statistical Society Series A* **50**, 51-61, (2001).

[7] J. Y. Koo, Spline estimation of discontinuous regression functions, *Journal of Computational and Graphical Statistics* **6**, 266-284, (1997).

[8] D. Lee, Coping with discontinuities in computer vision: their detection, classification, and measurement, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **12**, 321-344, (1990).

[9] F. Lombard, Detecting change points by Fourier analysis, *Technomerics* **30**, 305-310, (1988).

[10] H. W. J. Lee, K. L. Teo, V. Rehbock and L. S. Jennings, Control parameterization enhancing transform technique for time optimal control problems. *Dynamic Systems and Applications* **6**, 243-261, (1997).

[11] A. McDonald and A. L. Owen, Smoothing with split linear fits, *Technometrics* **28**, 195-208, (1986).

[12] S. Miyata and X. Shen, Adaptive free-knot splines, *Journal of Computational and Graphical Statistics* **12**, 197-213, (2003).

[13] H. G. Müller, Change-points in nonparametric regression analysis, *Annals Statistics* **20**, 737-761, (1992).

[14] H. G. Müller and K. Song, Two-stage change-point estimators in smooth regression models, *Statistics and Probability Letters* **34**, 323-335, (1997).

[15] Q. Shao and N. Campbell, Modelling trends in groundwater levels by segmented regression with constraints. *Australian and New Zealand Journal of Statistics* **44**, 129-141, (2002).

[16] K. L. Teo, L. S. Jennings, H. W. J. Lee and V. Rehbock, The control parametrization enhancing transform for constrained optimal control problems, *Journal of Australia Mathematical Society B* **40**, 314-335, (1999).

[17] H. Tong, Threshold autoregression, limit cycles and cyclical data, with discussions, *Journal of Royal Statistical Society B* **42**, 245-292, (1980).

[18] Y. Wang, Jump and sharp cusp selection by wavelets, *Biometrika* **82**, 385-397, (1995).