

# Inference and Characterization of Planar Trajectories

Jerome

a thesis submitted for the degree of  
Doctor of Philosophy  
at the University of Otago, Dunedin,  
New Zealand.

October 21, 2016



# Contents

1.1	Introduction . . . . .	1
1.2	Tractor Spline . . . . .	3
1.2.1	Objective Function . . . . .	3
1.2.2	Basis Functions . . . . .	4
1.2.3	Solution to The Objective Function . . . . .	5
1.2.4	Adjusted Penalty Term and Parameter Function . . . . .	8
1.3	Parameter Selection and Cross Validation . . . . .	8
1.4	Simulation . . . . .	11
1.4.1	Numerical Examples . . . . .	11
1.4.2	Evaluation . . . . .	12
1.5	Conclusion and Discussion . . . . .	18
1.6	Introduction . . . . .	20
1.6.1	Spline . . . . .	20
1.6.2	Gaussian Process Regression . . . . .	21
1.6.3	The Smoothing Spline as Bayes Estimates . . . . .	22
1.7	A reproducing kernel on $\mathcal{C}_{p.w.}^2[0, 1]$ . . . . .	23
1.8	Computation of Polynomial Smoothing Splines . . . . .	24
1.9	Polynomial Smoothing Splines as Bayes Estimates . . . . .	25
1.10	Numeric Simulation of Smoothing Spline and GPR . . . . .	27
	<b>References</b>	<b>29</b>



# List of Tables

- 1.1 MSE. Mean square errors of different methods. The star sign (\*) marks the smallest error among these methods under the same level. The difference is not significant. . . . . 19
- 1.2 TMSE. True mean square errors of different methods. The star sign (\*) marks the smallest error among these methods under the same level. The proposed tractor spline returns the smallest TMSE among all the methods under the same level except for *Doppler* with SNR=7. The differences are significant. . . . . 19



# List of Figures

1.1	The two basis functions $N_{2k+1}$ and $N_{2k+2}$ on interval $[t_k, t_{k+2})$ . It is apparently that these basis functions are continuous on this interval and have continuous first derivatives. . . . .	6
1.2	Numerical example: <i>Blocks</i> . (a) The true velocity function. (b) Velocity with Gaussian noise at SNR=7. (c) Generated position function. (d) Position with Gaussian noise at SNR=7. (e) Reconstruction from Wavelet with sure threshold. (f) Reconstruction from Wavelet with BayesThresh approach. (g) Reconstruction by P-spline. (h) Reconstruction by tractor spline setting $\gamma = 0$ . (i) Reconstruction by tractor spline with normal penalty term. (j) Reconstruction by proposed tractor spline. . . . .	13
1.3	Numerical example: <i>Bumps</i> . (a) The true velocity function. (b) Velocity with Gaussian noise at SNR=7. (c) Generated position function. (d) Position with Gaussian noise at SNR=7. (e) Reconstruction from Wavelet with sure threshold. (f) Reconstruction from Wavelet with BayesThresh approach. (g) Reconstruction by P-spline. (h) Reconstruction by tractor spline setting $\gamma = 0$ . (i) Reconstruction by tractor spline with normal penalty term. (j) Reconstruction by proposed tractor spline. . . . .	14
1.4	Numerical example: <i>HeaviSine</i> . (a) The true velocity function. (b) Velocity with Gaussian noise at SNR=7. (c) Generated position function. (d) Position with Gaussian noise at SNR=7. (e) Reconstruction from Wavelet with sure threshold. (f) Reconstruction from Wavelet with BayesThresh approach. (g) Reconstruction by P-spline. (h) Reconstruction by tractor spline setting $\gamma = 0$ . (i) Reconstruction by tractor spline with normal penalty term. (j) Reconstruction by proposed tractor spline. . . . .	15
1.5	Numerical example: <i>Doppler</i> . (a) The true velocity function. (b) Velocity with Gaussian noise at SNR=7. (c) Generated position function. (d) Position with Gaussian noise at SNR=7. (e) Reconstruction from Wavelet with sure threshold. (f) Reconstruction from Wavelet with BayesThresh approach. (g) Reconstruction by P-spline. (h) Reconstruction by tractor spline setting $\gamma = 0$ . (i) Reconstruction by tractor spline with normal penalty term. (j) Reconstruction by proposed tractor spline. . . . .	16
1.6	Estimated penalty functions. Left side shows how the value of $\lambda(t)$ changes on the interval. Right side projects $\lambda(t)$ into reconstructions. The bigger the blacks dots present, the larger the penalty values are. . . . .	17

1.7	Estimated velocity functions by taking the first derivative of tractor spline. (a) Fitted <i>Blocks</i> . (b) Fitted <i>Bumps</i> . (c) Fitted <i>HeaviSine</i> . (d) Fitted <i>Doppler</i> . . . . .	18
1.8	(a) Comparing two methods under the same parameters $\lambda = 0.01$ and $\gamma = 0.1$ . In this graph, the blue line is reconstruction from tractor spline, the red line is the mean of Gaussian Process, which is the posterior $\mathbb{E}(\eta(x) \mathbf{Y}, \mathbf{V})$ . (b) The differences between two methods under the same parameters. . . . .	28



# Writing Something

I'm thinking, I should start writing from Spline method, without adaptive terms on  $\lambda$ . An advanced Cross Validation method is given. Then when I tried to use this method to construct trajectories (application on real data), there appears some issues. So I brought adaptive terms.

It's a batch case method. Trajectory is reconstructed from a batch of data and some issues appeared. The numeric simulations proved that this new method is better. However, some issues still exist in real data application (long-gap-curve).

Then I will introduce Gaussian Process Regression and how it could estimate Smoothing Spline. Based on some references and tractor spline, I will prove some relationships between Tractor Spline and GPR. this is a spin-off results.

These are all batch method. Even if researchers could do some online smoothing, the computation time cost a lot.

Why don't we do online estimation by using some efficiency method? Introduce filters and other methods (Probably Kalman filter, particle filter and other filter) and dynamic linear regression models. Then the proposed method, parameters estimation and something else. It's an online case.

Parameter estimation methods: Metropolis-Hastings methods, better than MLE because the latter doesn't give the distribution of the estimates.

Moreover, I need to figure out how these different methods working and the advantages and disadvantages they have.

## 1.1 Introduction

In a vehicular system, the simplest way of getting the trajectory of a moving-object is connecting position points by a sequence of lines (line-based trajectory representation) in a 3-dimensional or 4-dimensional space-time Agarwal *et al.* (2003). Due to the noises generated from observation units, one can use regression method to find the best fitting returning the smallest sum square errors among all the sequences. Consider a regression model  $y_i = f(t_i) + \epsilon_i$ , where  $a \leq t_1 < \dots < t_n \leq b$  and  $f \in C^2[a, b]$  is an unknown smooth function,  $(\epsilon_i)_{i=1}^n \sim N(0, \sigma^2)$  are random errors. In a classical parametric regression,  $f$  is assumed having the form  $f(x, \beta)$ , which is known up to the data estimated parameters  $\beta$  Kim and Gu (2004). When  $f(x, \beta)$  is linear in  $\beta$ , we will have a standard linear model. However, most of the natural moving objects return

smooth trajectories without any angles. Therefore, a spline method is used to construct such trajectories. A curved-base method uses a parametric cubic function  $P(t) = a_0 + a_1t + a_2t^2 + a_3t^3$  to obtain a spline that passes through any given sequence of joint position-velocity paired points  $(x_1, v_1), (x_2, v_2), \dots, (x_n, v_n)$  Yu *et al.* (2004). In Yang and Sukkarieh (2010), an efficient and analytical continuous curvature path-smoothing algorithm based on parametric cubic Bézier curves is proposed. It can fit ordered sequential points smoothly. In computer (or computerized) numerical control (CNC), Altintas and Erkorkmaz Erkorkmaz and Altintas (2001) presented a quintic spline trajectory generation algorithm connecting a series of reference knots that produces continuous position, velocity and acceleration profiles.

However, a parametric approach only captures features contained in the preconceived class of functions Yao *et al.* (2005) and increases model bias. To avoid this, an alternative approach called nonparametric model was invented. Rather than giving a specified parameter, it is desired to reconstruct  $f$  from the data  $y(t_i) \equiv y_i, i = 1, \dots, n$  Craven and Wahba (1978). Smoothing spline estimate of the  $f$  function appears as a solution to the following minimization problem: Find  $\hat{f} \in C^2[a, b]$  that minimizes the penalized residual sum of squares:

$$\text{RSS} = \sum_{j=1}^n (y_j - f(t_j))^2 + \lambda \int_a^b f''(t)^2 dt \quad (1.1)$$

for pre-specified value  $\lambda > 0$  Aydin and Tuzemen (2012). In equation (1.1), the first part is residual sum square and it penalizes the lack of fit. The second part is roughness penalty term weighted by a smoothing parameter  $\lambda$ , which varies from 0 to  $+\infty$  and establishes a trade-off between interpolation and a linear model. The motivation of roughness penalty term is from a formalization of a mechanical device: if a thin piece of flexible wood, called a spline, is bent to the shape of the graph  $g$ , then the leading term in the strain energy is proportional to  $\int f''^2$  Green and Silverman (1993). The cost of equation (1.1) is determined not only by its goodness-of-fit to the data quantified by the residual sum of squares, but also by its roughness Schwarz (2012). For a given  $\lambda$ , minimizing equation (1.1) will give the best compromise between smoothness and goodness-of-fit. Notice that the first term in equation (1.1) depends only on the values of  $f$  at knots  $t_i, i = 1, \dots, n$ . In the book, the authors show that the function that minimizes the roughness penalty for fixed values of  $f(t_i)$  is a cubic spline: an interpolation of points via a continuous piecewise cubic function, with continuous first and second derivatives Green and Silverman (1993). The continuity requirements uniquely determine the interpolating spline, except at the boundaries Sealton *et al.* (2005).

A conventional smoothing spline is controlled by one single parameter, which controls the smoothness of a spline on the whole domain. A natural extension is to allow the smoothing parameter to vary as a penalty function of the independent variable, adapting to the change of roughness in different domains Silverman (1985), Donoho *et al.* (1995). In this way, a new objective function is formulated in the form of

$$\sum_{j=1}^n (y_j - f(x_j))^2 + \int_T \lambda(t) f''(t)^2 dt, \quad (1.2)$$

by minimizing which, the best estimation  $\hat{f}$  can be found. This approach makes adaptive smoothing as a minimization problem with a new penalty term.

Similar to conventional smoothing spline problem, researchers are wondering how to choose the penalty function  $\lambda(t)$ . The fundamental idea of nonparametric smoothing is to let the data choose the amount of smoothness, which consequently decides the model complexity Gu (1998). Most of them focus on data driven criteria, such as cross validation (CV), generalized cross validation (GCV) Craven and Wahba (1978) and generalized maximum likelihood (GML) Wahba (1985). A new challenge is posed that the smoothing parameter becomes a function and is varying in domains. The structure of this penalty function controls the complexity on each domain and the whole final model. Liu and Guo proposed to approximate the penalty function with an indicator and extended the generalized likelihood to the adaptive smoothing spline Liu and Guo (2010).

In this paper, we propose an adaptive smoothing spline method based on Hermite Spline basis functions to get reconstruction of  $f$  and  $f'$  from noisy data  $\mathbf{y}$  and  $\mathbf{v}$ . Rather than only using residuals of  $f(t_i) - y_i$  term in the objective function in equation (1.2), we added the residuals of  $f'(t_i) - v_i$  as a new term containing a new parameter  $\gamma$ . In this way, the spline keeps a balance on both observed  $\mathbf{y}$  and  $\mathbf{v}$ . Using new generated basis functions, we reconstruct a smoothing spline on the whole interval  $[a, b]$ . Derived from the new objective function, an advanced cross validation formula of  $f(t)$  and  $f'(t)$  is given. This method can be used in either getting true signal from noisy data or moving-object database.

## 1.2 Tractor Spline

### 1.2.1 Objective Function

In a 2D curve nonparametric regression, consider  $n$  time points  $t_{1:n}$ , such that  $a \leq t_1, \dots, t_n \leq b$ . Let  $z_i = (x_i, y_i)$  and  $w_i = (u_i, v_i)$  for  $i = 1, \dots, n$ . We define a positive piecewise constant function  $\lambda(t)$  :

$$\lambda(t) = \lambda_i > 0, \quad (1.3)$$

where  $t_i \leq t < t_{i+1}$ ,  $t_0 = a$ ,  $t_{n+1} = b$ , that will control the curvature penalty of each interval. For a function  $f : [a, b] \rightarrow \mathbb{R}^2$  and  $\gamma > 0$ , define the objective function

$$J[f] = \frac{1}{n} \sum_{i=1}^n (f(t_i) - z_i)^2 + \frac{\gamma}{n} \sum_{i=1}^n (f'(t_i) - w_i)^2 + \sum_{i=0}^n \lambda_i \int_{t_i}^{t_{i+1}} f''(t)^2 dt, \quad (1.4)$$

where  $\gamma$  is a coefficient of the velocity information  $\mathbf{v}$  and it weights the residuals between  $\mathbf{f}'$  and  $\mathbf{v}$ , and  $\lambda(t)$  is the smoothing parameter function.

**Theorem 1.** *For  $n \geq 2$ , the objective function  $J[f]$  is minimized by a cubic spline that is linear outside the knots.*

The solution to the objective function (1.4) is called tractor spline.

In the following, we divide the 2D function  $f(x, y)$  into two sub functions  $f_x(t)$  on  $x$ -axis and  $f_y(t)$  on  $y$ -axis. Compared with other parameters, choosing time  $t$  to be the parameter has some advantages: 1. The expressions of all the constraints are simpler Zhang *et al.* (2013); 2. It can be simply applied from 2-dimension to 3-dimension by adding an extra  $z$ -axis.

### 1.2.2 Basis Functions

Suppose we have a time series sequence of observed dataset  $a = t_1 < t_2 < \dots < t_n = b$ . The function  $f(t)$  (stands for  $f_y(t)$ ) defined on this interval  $[t_1, t_n]$  is called tractor spline, if it is the solution to the objective function (1.4). Then it has the following property: on each interior interval  $(t_i, t_{i+1})$ ,  $i = 2, \dots, n-2$ ,  $f(t)$  is a cubic polynomial, but on interval  $(t_1, t_2)$  and  $(t_{n-1}, t_n)$  can be linear;  $f(t)$  fits together at each point  $t_i$  in such a way that  $f(t)$  itself and its first derivatives are continuous at each  $t_i$ ,  $i = 2, \dots, n-2$ .

Using Hermite interpolation on an arbitrary interval  $[t_i, t_{i+1}]$ , the cubic spline basis functions can be constructed as follows

$$h_{00}^{(i)}(t) = \begin{cases} 2\left(\frac{t-t_i}{t_{i+1}-t_i}\right)^3 - 3\left(\frac{t-t_i}{t_{i+1}-t_i}\right)^2 + 1 & t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}, \quad (1.5)$$

$$h_{10}^{(i)}(t) = \begin{cases} \frac{(t-t_i)^3}{(t_{i+1}-t_i)^2} - 2\frac{(t-t_i)^2}{t_{i+1}-t_i} + (t-t_i) & t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}, \quad (1.6)$$

$$h_{01}^{(i)}(t) = \begin{cases} -2\left(\frac{t-t_i}{t_{i+1}-t_i}\right)^3 + 3\left(\frac{t-t_i}{t_{i+1}-t_i}\right)^2 & t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}, \quad (1.7)$$

$$h_{11}^{(i)}(t) = \begin{cases} \frac{(t-t_i)^3}{(t_{i+1}-t_i)^2} - \frac{(t-t_i)^2}{t_{i+1}-t_i} & t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}. \quad (1.8)$$

Then a Hermite spline  $f^{(i)}(t)$  on interval  $[t_i, t_{i+1})$  with points  $p_i = \{y_i, v_i\}$  and  $p_{i+1} = \{y_{i+1}, v_{i+1}\}$  can be expressed as

$$f^{(i)}(t) = h_{00}^{(i)}(t)y_i + h_{10}^{(i)}(t)v_i + h_{01}^{(i)}(t)y_{i+1} + h_{11}^{(i)}(t)v_{i+1}. \quad (1.9)$$

To construct a tractor spline on the entire interval  $[t_1, t_n]$ , the new basis functions are defined in such way, that  $N_1 = h_{00}^{(1)}$ ,  $N_2 = h_{10}^{(1)}$ , and for all  $k = 1, 2, \dots, n-2$ ,

$$N_{2k+1} = \begin{cases} h_{01}^{(k)} + h_{00}^{(k+1)} & \text{if } t < t_n \\ 2\left(\frac{t-t_{n-1}}{t_n-t_{n-1}}\right)^3 - 3\left(\frac{t-t_{n-1}}{t_n-t_{n-1}}\right)^2 + 1 & \text{if } t = t_n \end{cases}, \quad (1.10)$$

$$N_{2k+2} = \begin{cases} h_{11}^{(k)} + h_{10}^{(k+1)} & \text{if } t < t_n \\ \frac{(t-t_{n-1})^3}{(t_n-t_{n-1})^2} - 2\frac{(t-t_{n-1})^2}{t_n-t_{n-1}} + (t-t_{n-1}) & \text{if } t = t_n \end{cases}, \quad (1.11)$$

and

$$N_{2n-1} = \begin{cases} h_{01}^{(n-1)} & \text{if } t < t_n \\ -2\left(\frac{t-t_{n-1}}{t_n-t_{n-1}}\right)^3 + 3\left(\frac{t-t_{n-1}}{t_n-t_{n-1}}\right)^2 & \text{if } t = t_n \end{cases}, \quad (1.12)$$

$$N_{2n} = \begin{cases} h_{11}^{(n-1)} & \text{if } t < t_n \\ \frac{(t-t_{n-1})^3}{(t_n-t_{n-1})^2} - \frac{(t-t_{n-1})^2}{t_n-t_{n-1}} & \text{if } t = t_n \end{cases}. \quad (1.13)$$

**Theorem 2.** On  $[t_1, t_n]$ , the functions  $N_1, \dots, N_{2n}$  provide a basis for the set of functions which are continuous, have continuous first derivatives and are cubic on each open interval  $(t_i, t_{i+1})$ , where  $i = 1, \dots, n-1$ .

As independent basis functions,  $N_1(t), \dots, N_{2n}(t)$  span a  $2n$  dimensional function space  $\mathbb{H}$ . For any  $f \in \mathbb{H}$ , it can be represented in the form of

$$f = \sum_{k=1}^{2n} \theta_k N_k(t), \quad (1.14)$$

where  $\{\theta_k\}_{k=1}^{2n}$  are parameters.

Figure (1.1) presents two basis functions on an arbitrary interval  $[t_k, t_{k+2})$  where they are continuous and differential. At the interior joint knot  $t_k$ , basis functions in the previous and following interval share the same position  $y_k$  and velocity  $v_k$ .

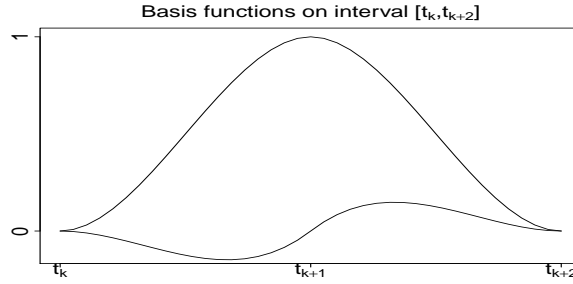


Figure 1.1: The two basis functions  $N_{2k+1}$  and  $N_{2k+2}$  on interval  $[t_k, t_{k+2})$ . It is apparently that these basis functions are continuous on this interval and have continuous first derivatives.

### 1.2.3 Solution to The Objective Function

Basis functions have been defined in the previous subsection, therefore the tractor spline  $f(t)$  on  $[a, b]$ , where  $a \leq t_1 < t_2 < \dots < t_{n-1} < t_n \leq b$ , can be found by minimizing objective function (1.4), which reduces to

$$\text{MSE}(\theta, \lambda, \gamma) = (\mathbf{y} - \mathbf{B}\theta)^\top (\mathbf{y} - \mathbf{B}\theta) + \gamma(\mathbf{v} - \mathbf{C}\theta)^\top (\mathbf{v} - \mathbf{C}\theta) + n\theta^\top \Omega_\lambda \theta, \quad (1.15)$$

where  $\{\mathbf{B}\}_{ij} = N_j(t_i)$ ,  $\{\mathbf{C}\}_{ij} = N'_j(t_i)$  and  $\{\Omega_{2n}^{(k)}\}_{jk} = \int_{t_k}^{t_{k+1}} \lambda_k N_j''(t) N_k''(t) dt$ . After substituting the series observation  $t_1, \dots, t_n$  into basis functions, we get  $N_1(t_1) = 1, N_1(t_2) = 0, \dots, N_{2k-1}(t_k) = 1, N_{2k}(t_k) = 0, \dots, N_{2n-1}(t_n) = 1, N_{2n}(t_n) = 0$ ; and into first derivative of basis functions, we get  $N'_1(t_1) = 0, N'_1(t_2) = 1, \dots, N'_{2k-1}(t_k) = 0, N'_{2k}(t_k) = 1, \dots, N'_{2n-1}(t_n) = 0, N'_{2n}(t_n) = 1$ . That means the matrices  $\mathbf{B}$  and  $\mathbf{C}$  in

MSE equation (1.15) are  $n \times 2n$  dimensional and the elements are

$$\mathbf{B} = \{B\}_{ij} = \begin{cases} 1, & j = 2i - 1 \\ 0, & \text{otherwise} \end{cases} \quad (1.16)$$

$$\mathbf{C} = \{C\}_{ij} = \begin{cases} 1, & j = 2i \\ 0, & \text{otherwise} \end{cases} \quad (1.17)$$

where  $i = 1, \dots, n$ . The  $k$ -th  $\Omega_{\lambda_k}^{(k)}$  is a  $2n \times 2n$  matrix and its details is in appendix. Then the penalty term is

$$\Omega_\lambda = \sum_{k=1}^{n-1} \Omega_{\lambda_k}^{(k)}, \quad (1.18)$$

which is a bandwidth four matrix.

The solution to (1.15) is easily seen to be

$$\hat{\theta} = (\mathbf{B}^\top \mathbf{B} + \gamma \mathbf{C}^\top \mathbf{C} + n\Omega_\lambda)^{-1} (\mathbf{B}^\top \mathbf{y} + \gamma \mathbf{C}^\top \mathbf{v}) \quad (1.19)$$

a generalized ridge regression. Then the fitted smoothing spline is given by

$$\hat{f}(t) = \sum_{i=1}^{2n} N_i(t) \hat{\theta}_i \quad (1.20)$$

A smoothing spline with parameters  $\lambda(t)$  and  $\gamma$  is an example of a linear smoother Trevor Hastie (2009). This is because the estimated parameters in equation (1.19) are a linear combination of  $y_i$  and  $v_i$ . Denote by  $\hat{\mathbf{f}}$  and  $\hat{\mathbf{f}}'$  the  $2n$  vector of fitted values  $\hat{f}(t_i)$  and  $\hat{f}'(t_i)$  at the training points  $t_i$ . Then

$$\begin{aligned} \hat{\mathbf{f}} &= \mathbf{B}(\mathbf{B}^\top \mathbf{B} + \gamma \mathbf{C}^\top \mathbf{C} + n\Omega_\lambda)^{-1} (\mathbf{B}^\top \mathbf{y} + \gamma \mathbf{C}^\top \mathbf{v}) \\ &\triangleq \mathbf{S}_{\lambda, \gamma} \mathbf{y} + \gamma \mathbf{T}_{\lambda, \gamma} \mathbf{v} \end{aligned} \quad (1.21)$$

$$\begin{aligned} \hat{\mathbf{f}}' &= \mathbf{C}(\mathbf{B}^\top \mathbf{B} + \gamma \mathbf{C}^\top \mathbf{C} + n\Omega_\lambda)^{-1} (\mathbf{B}^\top \mathbf{y} + \gamma \mathbf{C}^\top \mathbf{v}) \\ &\triangleq \mathbf{U}_{\lambda, \gamma} \mathbf{y} + \gamma \mathbf{V}_{\lambda, \gamma} \mathbf{v} \end{aligned} \quad (1.22)$$

The fitted  $\hat{\mathbf{f}}$  and  $\hat{\mathbf{f}}'$  are linear in  $\mathbf{y}$  and  $\mathbf{v}$ , and the finite linear operators  $\mathbf{S}_{\lambda, \gamma}$ ,  $\mathbf{T}_{\lambda, \gamma}$ ,  $\mathbf{U}_{\lambda, \gamma}$  and  $\mathbf{V}_{\lambda, \gamma}$  are known as the smoother matrices. One consequence of this linearity is that the recipe for producing  $\hat{\mathbf{f}}$  and  $\hat{\mathbf{f}}'$  from  $\mathbf{y}$  and  $\mathbf{v}$ , do not depend on  $\mathbf{y}$  and  $\mathbf{v}$  themselves;  $\mathbf{S}_{\lambda, \gamma}$ ,  $\mathbf{T}_{\lambda, \gamma}$ ,  $\mathbf{U}_{\lambda, \gamma}$  and  $\mathbf{V}_{\lambda, \gamma}$  depend only on  $t_i$ ,  $\lambda(t)$  and  $\gamma$ .

Suppose in a traditional least squares fitting,  $\mathbf{B}_\xi$  is  $N \times M$  matrix of  $M$  cubic-spline basis functions evaluated at the  $N$  training points  $x_i$ , with knot sequence  $\xi$  and  $M \ll N$ . Then the vector of fitted spline values is given by

$$\hat{\mathbf{f}} = \mathbf{B}_\xi (\mathbf{B}_\xi^\top \mathbf{B}_\xi)^{-1} \mathbf{B}_\xi \mathbf{y} = \mathbf{H}_\xi \mathbf{y} \quad (1.23)$$

Here the linear operator  $\mathbf{H}_\xi$  is a symmetric, positive semidefinite matrices, and  $\mathbf{H}_\xi \mathbf{H}_\xi = \mathbf{H}_\xi$  (idempotent) Trevor Hastie (2009). In our case, it is easily seen that  $\mathbf{S}_{\lambda,\gamma}$ ,  $\mathbf{T}_{\lambda,\gamma}$ ,  $\mathbf{U}_{\lambda,\gamma}$  and  $\mathbf{V}_{\lambda,\gamma}$  are symmetric, positive semidefinite matrices as well. Additionally, by Cholesky decomposition

$$(\mathbf{B}^\top \mathbf{B} + \gamma \mathbf{C}^\top \mathbf{C} + n\Omega_\lambda)^{-1} = \mathbf{R}\mathbf{R}^\top, \quad (1.24)$$

it is easily to prove that  $\mathbf{T}_{\lambda,\gamma} = \mathbf{B}\mathbf{R}\mathbf{R}^\top \mathbf{C}^\top$  and  $\mathbf{U}_{\lambda,\gamma} = \mathbf{C}\mathbf{R}\mathbf{R}^\top \mathbf{B}^\top$ , then we will have  $\mathbf{T}_{\lambda,\gamma} = \mathbf{U}_{\lambda,\gamma}^\top$ . When  $\lambda = \gamma = 0$ , the matrix  $\mathbf{S}_{\lambda_0,\gamma_0} = \mathbf{B}(\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top$  is idempotent.

Additionally, by playing with  $\lambda(t)$  and  $\gamma$ , tractor spline has the following property:

- if  $\lambda(t)$  is a piecewise constant and  $\gamma \neq 0$ , then  $f$  and  $f'$  are continuous,  $f''$  is piecewise linear but not continuous at knots;
- if  $\lambda(t)$  is a piecewise constant and  $\gamma = 0$ , the same as above;
- if  $\lambda(t) = \lambda$  is a constant and  $\gamma \neq 0$ , the same as above;
- if  $\lambda(t) = \lambda$  is a constant and  $\gamma = 0$ , then  $f$ ,  $f'$  are continuous,  $f''$  is piecewise linear and continuous at knots.

## 1.2.4 Adjusted Penalty Term and Parameter Function

To get the reconstructed trajectory in a multi-dimensional space, one can use tractor spline to find the trajectory in each dimensions simultaneously can combine them together at the end. Additionally, sometimes the data is not recorded in equal space. Due to the property of Hermite spline, the combination of multi-dimensional reconstructions and non-equal space data will bring some issues. Imagining this situation that a vehicle is moving along the  $x$ -axis, its  $x$  position changes consequently, but its  $y$  position might stay the same. By fitting  $\mathbf{x}$  and  $\mathbf{u}$  on  $x$ -axis, the tractor spline  $f_x(t)$  will give us a best fit which returns smallest errors to the objective function. While with the same parameter  $\lambda(t)$  and  $\gamma$ ,  $f_y(t)$  will return a cubic curve. However, it should give us a straight line as we expected. Moreover, in some circumstances, the time mark increases but  $\mathbf{f}$  and  $\mathbf{f}'$  keep the same, or changes slightly. Facing this situation, the Hermite spline will return a wiggle in one dimension and a curve in two dimensions. To get a reliable reconstruction, we introduce an adjusted term  $\frac{(\Delta t_i)^\alpha}{(\Delta d_i)^\beta}$ , where  $\alpha \geq 0$  and  $\beta \geq 0$ , to the penalty function  $\lambda(t)$ , which means that the tractor spline should be penalized by its real difference of  $\Delta d_i$  and  $\Delta t_i$  between two points  $p_i$  and  $p_{i+1}$ . With this term, when  $\mathbf{u}$  goes down or equals to 0, it will make sure that the penalty function



will be large enough and return a straight line rather than a curve in each dimension of  $x$  and  $y$ . Because of the unit of the penalty term is  $m^2/t^3$ , to keep the same scale in the space,  $\alpha$  and  $\beta$  in the adjusted penalty term are chosen as 3 and 2. Then the final form of the penalty function is

$$\lambda(t) = \frac{(\Delta t_i)^3}{(\Delta d_i)^2} \lambda, \quad (1.25)$$

where  $t_i \leq t < t_{i+1}$ . Eventually in objective function there is one parameter  $\lambda$  controlling the curvature of tractor spline in different status, and another one parameter  $\gamma$  controlling the residuals of velocity.

### 1.3 Parameter Selection and Cross Validation

The problem of choosing the smoothing parameter is ubiquitous in curve estimation. And there are two different philosophical approaches to this question. The first one is to regard the free choice of smoothing parameter as an advantageous feature of the procedure. The other one is to find the parameter automatically by the data Green and Silverman (1993). We more prefer the latter one, use data to train our model and find the best parameters. The most well known method is cross-validation.

Assuming that the random errors has zero mean, the true regression curve  $f(t)$  has the property that, if an observation  $y$  is taken at a point  $t$ , the value  $f(t)$  is the best predictor of  $y$  in terms of returning a small value of  $(y - f(t))^2$ .

Now we focus on an observation  $y_i$  at point  $t_i$  as being a new observation by omitting it from the set of data, which are used to estimate  $\hat{f}$ . Denote by  $\hat{f}^{(-i)}(t, \lambda)$  the estimated function from the remaining data, where  $\lambda$  is the smoothing parameter. Then  $\hat{f}^{(-i)}(t, \lambda)$  minimizes

$$\frac{1}{n} \sum_{j \neq i} (y_j - f(t_j))^2 + \lambda \int f''^2 dt \quad (1.26)$$

and  $\lambda$  can be quantified by cross-validation score function

$$CV(\lambda) = \frac{1}{n} \sum_{i=1}^n \{y_i - \hat{f}^{(-i)}(t_i, \lambda)\}^2. \quad (1.27)$$

The basis idea of cross-validation is to choose the value of  $\lambda$  that minimizes  $CV(\lambda)$ .

An efficient way to calculate cross validation score is given by Green and Silverman (1993). Through the equation (1.23), we know that the value of the smoothing spline

$\hat{f}$  depend linearly on the data  $y_i$ . Define the matrix  $A(\lambda)$ , which is a map vector of observed values  $y_i$  to predicted values  $\hat{f}(t_i)$ . Then we have

$$\hat{\mathbf{f}} = A(\lambda)\mathbf{y} \quad (1.28)$$

and the following lemma.

**Lemma 1.** *The cross validation score satisfies*

$$CV(\lambda) = \frac{1}{n} \sum_{i=1}^n \left( \frac{y_i - \hat{f}(t_i)}{1 - A_{ii}(\lambda)} \right)^2 \quad (1.29)$$

where  $\hat{f}$  is the spline smoother calculated from the full data set  $\{(t_i, y_i)\}$  with smoothing paramter  $\lambda$ .

For a tractor spline and its MSE function, there are two parameters need to be estimated  $\lambda$  and  $\gamma$ . Then the objective function (1.26) becomes

$$\frac{1}{n} \sum_{j \neq i} (y_j - f(t_j))^2 + \frac{\gamma}{n} \sum_{j \neq i} (v_j - f'(t_j))^2 + \int \lambda(t) f''^2 dt, \quad (1.30)$$

and the cross-validation score function is

$$CV(\lambda, \gamma) = \frac{1}{n} \sum_{i=1}^n \{y_i - \hat{f}^{(-i)}(t_i, \lambda, \gamma)\}^2. \quad (1.31)$$

For a tractor spline, the parameter  $\hat{\theta} = (B^\top B + \gamma C^\top C + n\Omega_\lambda)^{-1}(B^\top \mathbf{y} + \gamma C^\top \mathbf{v})$  gives us

$$\begin{aligned} \hat{\mathbf{f}} &= B\hat{\theta} = B(B^\top B + \gamma C^\top C + n\Omega_\lambda)^{-1}B^\top \mathbf{y} + B(B^\top B + \gamma C^\top C + n\Omega_\lambda)^{-1}C^\top \mathbf{v} \\ &= S\mathbf{y} + \gamma T\mathbf{v}, \end{aligned} \quad (1.32)$$

$$\begin{aligned} \hat{\mathbf{f}}' &= C\hat{\theta} = C(B^\top B + \gamma C^\top C + n\Omega_\lambda)^{-1}B^\top \mathbf{y} + C(B^\top B + \gamma C^\top C + n\Omega_\lambda)^{-1}C^\top \mathbf{v} \\ &= U\mathbf{y} + \gamma V\mathbf{v}. \end{aligned} \quad (1.33)$$

From lemma 1, we can prove the following theorem.

**Theorem 3.** *The cross validation score of a tractor spline satisfies*

$$CV(\lambda, \gamma) = \frac{1}{n} \sum_{i=1}^n \left( \frac{\hat{f}(t_i) - y_i + \gamma \frac{T_{ii}}{1 - \gamma V_{ii}} (\hat{f}'(t_i) - v_i)}{1 - S_{ii} - \gamma \frac{T_{ii}}{1 - \gamma V_{ii}} U_{ii}} \right)^2 \quad (1.34)$$

where  $\hat{f}$  is the tractor spline smoother calculated from the full data set  $\{(t_i, y_i, v_i)\}$  with smoothing parameter  $\lambda$  and  $\gamma$ .

The proof of Theorem 3 follows immediately from a lemma, and gives an expression for the deleted residuals  $y_i - \hat{f}^{(-i)}(t_i)$  and  $v_i - \hat{f}'^{(-i)}(t_i)$  in terms of  $y_i - \hat{f}(t_i)$  and  $v_i - \hat{f}'(t_i)$  respectively.

**Lemma 2.** For fixed  $\lambda, \gamma$  and  $i$ , denote  $\mathbf{f}^{(-i)}$  by the vector with components  $f_j^{(-i)} = \hat{f}^{(-i)}(t_j, \lambda, \gamma)$ ,  $\mathbf{f}'^{(-i)}$  by the vector with components  $f_j'^{(-i)} = \hat{f}'^{(-i)}(t_j, \lambda, \gamma)$ , and define vectors  $\mathbf{y}^*$  and  $\mathbf{v}^*$  by

$$\begin{cases} y_j^* = y_j & j \neq i \\ y_i^* = \hat{f}^{(-i)}(t_i) & \text{otherwise} \end{cases}, \quad (1.35)$$

$$\begin{cases} v_j^* = v_j & j \neq i \\ v_i^* = \hat{f}'^{(-i)}(t_i) & \text{otherwise} \end{cases}. \quad (1.36)$$

Then

$$\hat{\mathbf{f}}^{(-i)} = S\mathbf{y}^* + \gamma T\mathbf{v}^* \quad (1.37)$$

$$\hat{\mathbf{f}}'^{(-i)} = U\mathbf{y}^* + \gamma V\mathbf{v}^* \quad (1.38)$$

## 1.4 Simulation

### 1.4.1 Numerical Examples

In this section, we examine the visual quality of the proposed method with four functions: Blocks, Bumps, HeaviSine and Doppler, which have been used in Donoho and Johnstone (1994), Donoho and Johnstone (1995) and Abramovich *et al.* (1998) because of their caricature features in imaging, spectroscopy and other scientific signal processing. However it is unfair for tractor spline fitting "jump" position in Blocks and Bumps function, because it fits position and velocity simultaneously and these points imply infinite first derivative in original functions, which are impossible for vehicles or individuals. In terms of this issue, we treat these functions as velocity, and use noise free points to generate accurate position data, then add noises back to them.

For calculating consideration, we use  $n = 1024$  Nason (2010). Because all noises are randomly generated, for convenience of reinitialization and repetition of comparing, we set random seed as 2016. The noises are independent Gaussian distribution  $\epsilon \sim N(0, 1)$  and signal-to-noise ratio (SNR) is 7. These data are treated as velocity (first derivative). By setting initial position  $y_0 = 0$ , acceleration  $a_0 = 0$  and using the following formula

to calculate position

$$y_{i+1} = y_i + (v_i + v_{i+1}) \frac{t_{i+1} - t_i}{2}, \quad (1.39)$$

we can easily generate position data. Then we add some noises, which are independent Gaussian distribution  $\epsilon \sim N(0, 1)$  and SNR is 7. For wavelet reconstruction, we use the threshold policy of "sure" and "BayesThresh" with levels  $j = 4, \dots, 9$ . Penalized B-spline is added in comparison. For tractor spline we have two parameters  $\lambda$  and  $\gamma$ . To evaluate the performance of the velocity term in objective function (1.4) and the adjusted penalty term in (1.25), the parameter  $\gamma$  is set as 0 in one reconstruction of tractor spline, whose objective function and solution become

$$J[f]_{\gamma=0} = \frac{1}{n} \sum_{i=1}^n (f(t_i) - y_i)^2 + \sum_{i=1}^{n-1} \lambda_i \int_{t_i}^{t_{i+1}} f''(t)^2 dt, \quad (1.40)$$

and

$$\hat{\theta}_{\gamma=0} = (\mathbf{B}^\top \mathbf{B} + n\Omega_\lambda)^{-1} \mathbf{B}^\top \mathbf{y} \quad (1.41)$$

and the adjusted penalty term in (1.25) was removed from another reconstruction, noted as "tractor spline without APT". Figure (1.2) to (1.5) display the original (velocity), generated position, wavelet with two different threshold methods, P-spline and three kinds of tractor spline fitted functions. The parameters  $\lambda$  and  $\gamma$  of a tractor spline are automatically selected from formula (1.34) by *optim()* function in *R*.

By comparing, we can see that all these methods can rebuild up the skeleton of generated trajectory. Wavelets(sure) method have more wiggles in interior interval than Wavelet(BayesThresh), and the latter one becomes fluctuation near boundary knots. P-spline gives much smoother fitting than wavelets, but the drawback is losing more specific details. Tractor spline without velocity might give us less fitting, as can be seen from Blocks and Bumps where there should be a straight line. Tractor spline without adjusted penalty term could also get over fitting when the direction changes more frequently than normal, although it catches specific feature in HeaviSine. The proposed tractor spline performs much better than other methods and returns the near-true trajectory reconstruction.

Figure 1.6 shows the estimated penalty function

$$\lambda(t) = \frac{(\Delta t)^3}{(\Delta d)^2} \lambda. \quad (1.42)$$

The left column illustrates the value of penalty function on different intervals and the right column is the projection on position. Bigger black dots present larger penalty values. It can be seen that  $\lambda(t)$  adapts to the smoothness pattern of position and will

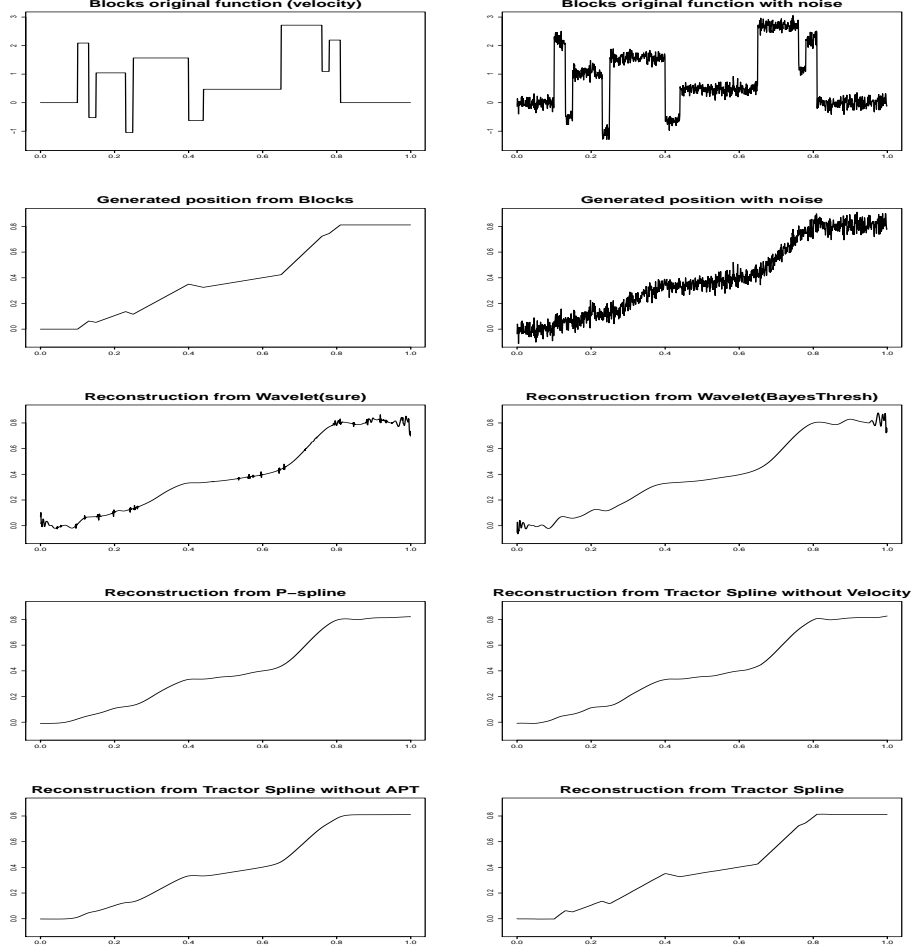


Figure 1.2: Numerical example: *Blocks*. (a) The true velocity function. (b) Velocity with Gaussian noise at SNR=7. (c) Generated position function. (d) Position with Gaussian noise at SNR=7. (e) Reconstruction from Wavelet with sure threshold. (f) Reconstruction from Wavelet with BayesThresh approach. (g) Reconstruction by P-spline. (h) Reconstruction by tractor spline setting  $\gamma = 0$ . (i) Reconstruction by tractor spline with normal penalty term. (j) Reconstruction by proposed tractor spline.

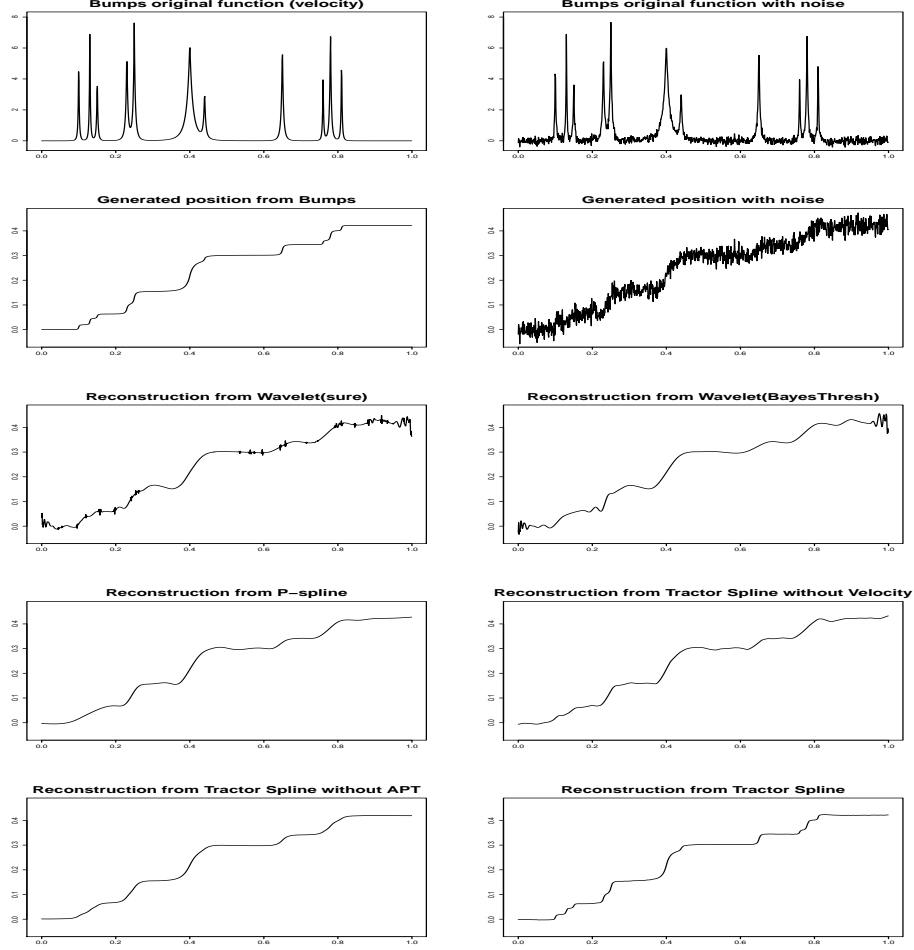


Figure 1.3: Numerical example: *Bumps*. (a) The true velocity function. (b) Velocity with Gaussian noise at SNR=7. (c) Generated position function. (d) Position with Gaussian noise at SNR=7. (e) Reconstruction from Wavelet with sure threshold. (f) Reconstruction from Wavelet with BayesThresh approach. (g) Reconstruction by P-spline. (h) Reconstruction by tractor spline setting  $\gamma = 0$ . (i) Reconstruction by tractor spline with normal penalty term. (j) Reconstruction by proposed tractor spline.

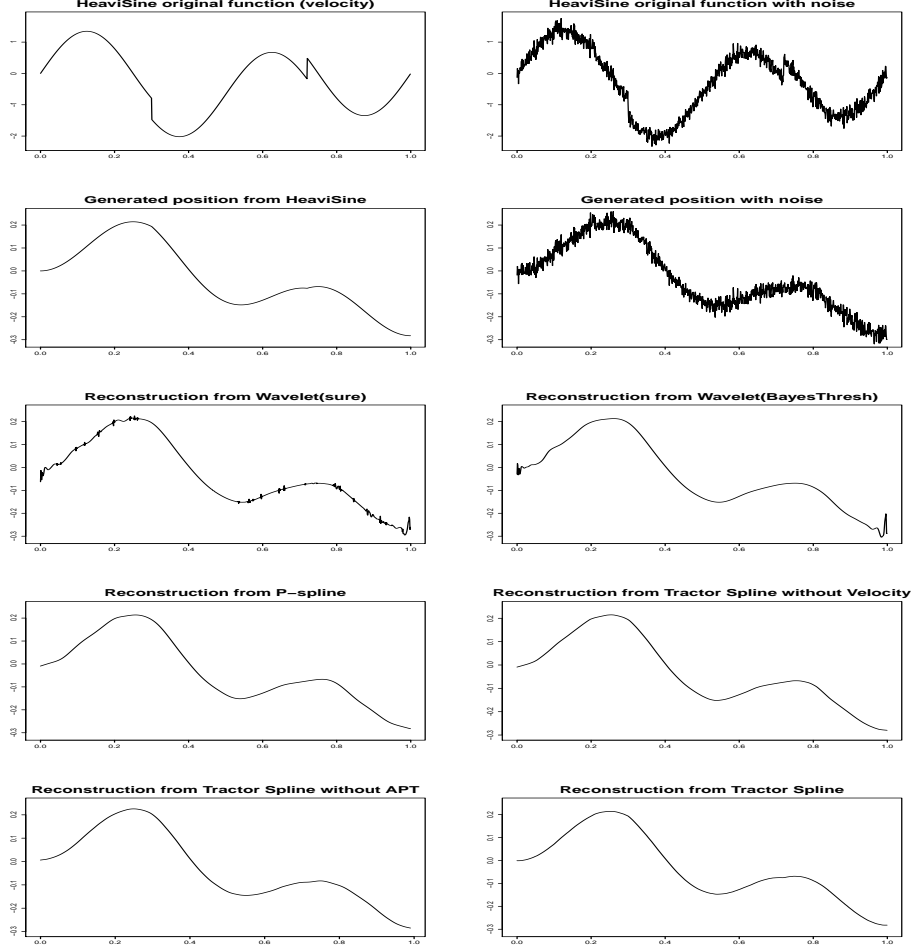


Figure 1.4: Numerical example: *HeaviSine*. (a) The true velocity function. (b) Velocity with Gaussian noise at SNR=7. (c) Generated position function. (d) Position with Gaussian noise at SNR=7. (e) Reconstruction from Wavelet with sure threshold. (f) Reconstruction from Wavelet with BayesThresh approach. (g) Reconstruction by P-spline. (h) Reconstruction by tractor spline setting  $\gamma = 0$ . (i) Reconstruction by tractor spline with normal penalty term. (j) Reconstruction by proposed tractor spline.

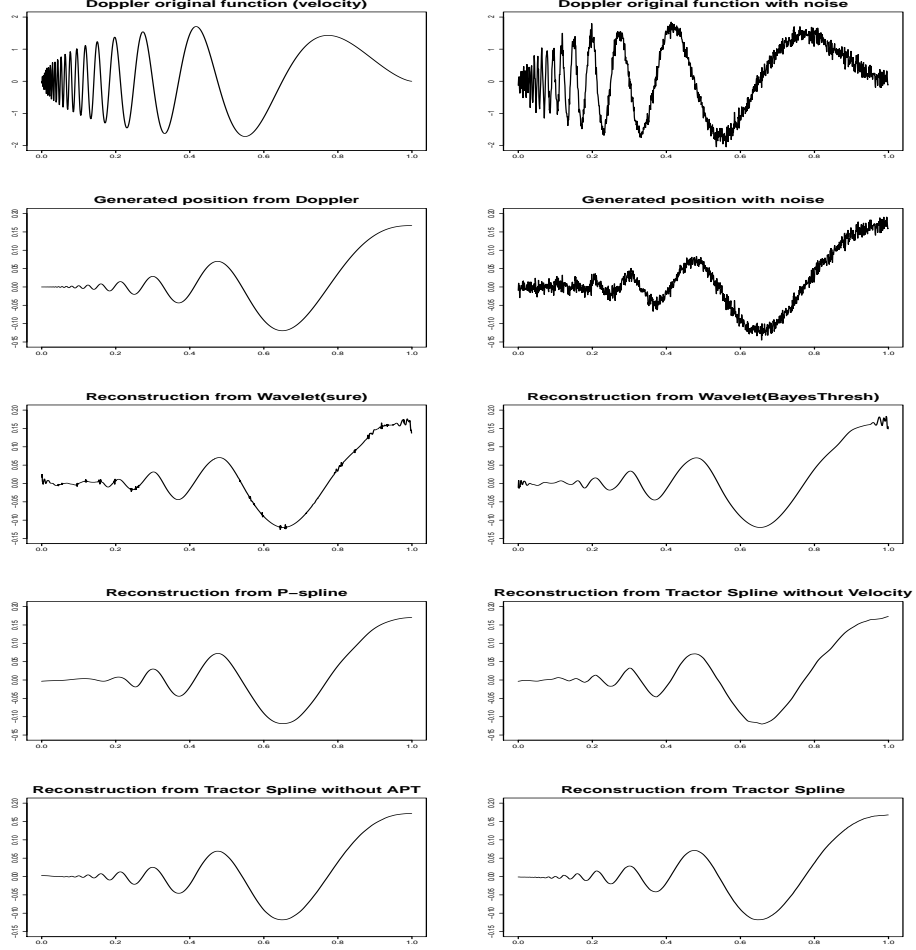


Figure 1.5: Numerical example: *Doppler*. (a) The true velocity function. (b) Velocity with Gaussian noise at SNR=7. (c) Generated position function. (d) Position with Gaussian noise at SNR=7. (e) Reconstruction from Wavelet with sure threshold. (f) Reconstruction from Wavelet with BayesThresh approach. (g) Reconstruction by P-spline. (h) Reconstruction by tractor spline setting  $\gamma = 0$ . (i) Reconstruction by tractor spline with normal penalty term. (j) Reconstruction by proposed tractor spline.



be large where a long time gap may occur. The details of how this penalty function works will be explained in next subsection.

Figure 1.7 demonstrates the estimated velocity functions. By taking the first derivative of fitted tractor spline, it is easy to get the original four velocity functions. The fitting of velocity is not as smooth as that in position, because we only care about the smoothness of position rather than velocity in our cross-validation formula (3). However, velocity information does help us reconstructing trajectory.

## 1.4.2 Evaluation

To examine the performance of tractor spline, we conducted an evaluation by comparing the mean square errors and true mean square errors, which are respectively calculated in

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{f}_{\lambda, \gamma}(t_i))^2, \quad (1.43)$$

$$\text{TMSE} = \frac{1}{n} \sum_{i=1}^n (f(t_i) - \hat{f}_{\lambda, \gamma}(t_i))^2. \quad (1.44)$$

The results are shown in table 1.1 and 1.2. All of these methods have good performances in fitting noisy data. The differences of mean square error between these methods are not significant, as can be seen from table 1.1. The proposed method is not the best among these simulations according to MSE. However, from table 1.2, tractor spline returns the smallest true mean square errors. The difference is significant, that means the reconstruction from tractor spline is closer to the true trajectory.

## 1.5 Conclusion and Discussion

In this paper, we proposed a tractor spline model with first derivative and adaptive penalty terms to reconstruct trajectory. This method performs better when we know  $\mathbf{z}$  and  $\mathbf{w}$  information than other methods. Additionally, the reconstruction of a tractor spline contains  $4 \times (n - 1)$  parameters if we have  $n$  knots. By adding  $2 \times (n - 2)$  constraints, the original function and its first derivative are continuous on each interior knots, the degrees of freedom will be  $4 \times (n - 1) - 2 \times (n - 2) = 2n$ . Because there are  $n$  location and  $n$  velocity data, thus we don't need to specify more parameters or add more constraints on the model. Although the mean square error of tractor spline is not the smallest comparing with other methods, the true mean square error is the smallest

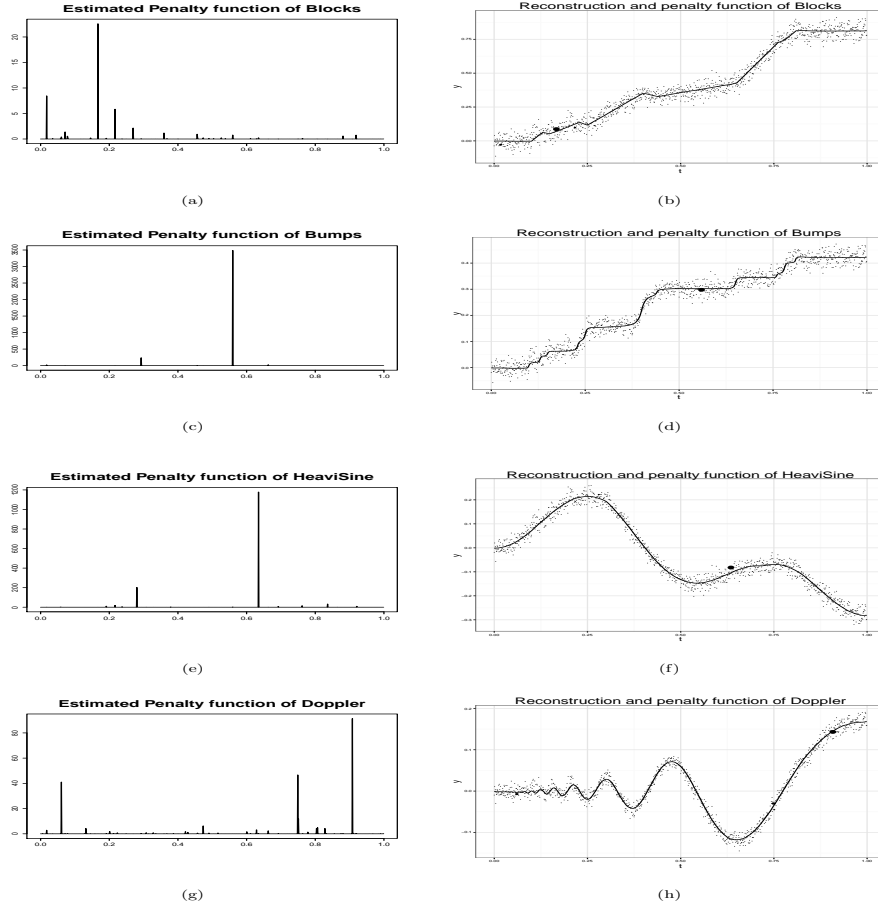


Figure 1.6: Estimated penalty functions. Left side shows how the value of  $\lambda(t)$  changes on the interval. Right side projects  $\lambda(t)$  into reconstructions. The bigger the blacks dots present, the larger the penalty values are.

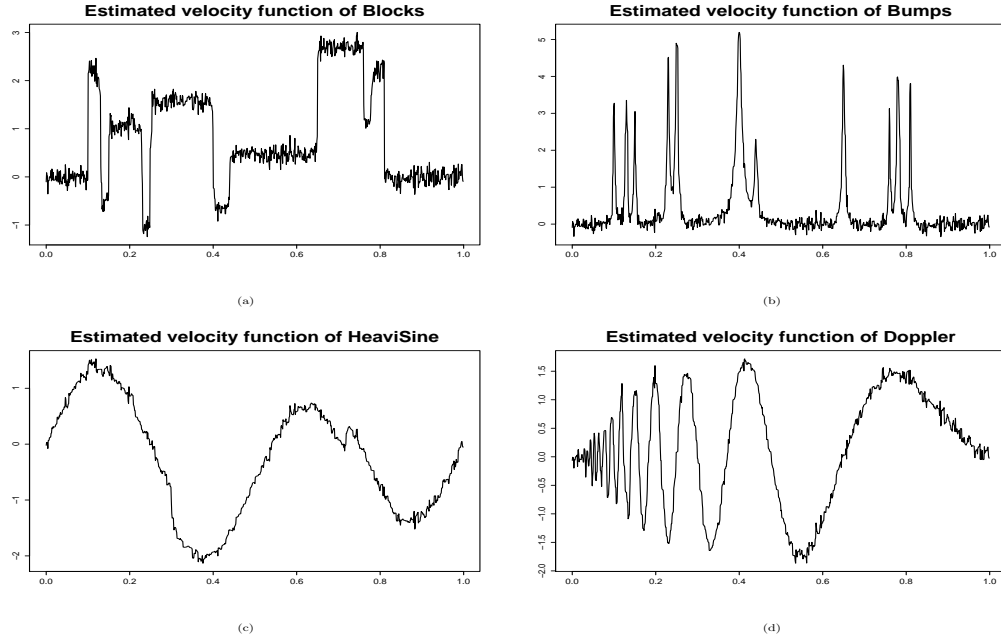


Figure 1.7: Estimated velocity functions by taking the first derivative of tractor spline. (a) Fitted *Blocks*. (b) Fitted *Bumps*. (c) Fitted *HeaviSine*. (d) Fitted *Doppler*.

Table 1.1: MSE. Mean square errors of different methods. The star sign (\*) marks the smallest error among these methods under the same level. The difference is not significant.

MSE ( $10^{-4}$ )	SNR	TS	$TS_{\gamma=0}$	$TS_{APT=0}$	P-spline	Wavelet(sure)	Wavelet(Bayes)
<i>Blocks</i>	7	16.53	15.99	16.69	16.14	*15.39	16.68
<i>Blocks</i>	3	89.79	*87.64	89.94	88.27	98.35	90.24
<i>Bumps</i>	7	4.40	4.19	4.55	4.33	*4.18	4.59
<i>Bumps</i>	3	23.93	*23.19	24.10	23.55	26.23	23.74
<i>HeaviSine</i>	7	4.16	4.01	4.16	4.02	*3.79	4.19
<i>HeaviSine</i>	3	22.63	*22.19	22.65	22.02	23.53	22.07
<i>Doppler</i>	7	1.15	*1.07	1.10	1.15	*1.07	1.13
<i>Doppler</i>	3	6.27	*5.94	6.28	6.05	6.85	6.29

Table 1.2: TMSE. True mean square errors of different methods. The star sign (\*) marks the smallest error among these methods under the same level. The proposed tractor spline returns the smallest TMSE among all the methods under the same level except for *Doppler* with SNR=7. The differences are significant.

TMSE ( $10^{-6}$ )	SNR	TS	$TS_{\gamma=0}$	$TS_{APT=0}$	P-spline	Wavelet(sure)	Wavelet(Bayes)
<i>Blocks</i>	7	*1.75	54.25	28.68	54.76	201.02	182.12
<i>Blocks</i>	3	*16.44	152.5	30.76	171.59	1138.08	712.36
<i>Bumps</i>	7	*1.64	23.44	21.10	24.21	71.71	69.26
<i>Bumps</i>	3	*8.51	77.78	37.12	77.52	330.77	238.79
<i>HeaviSine</i>	7	*1.53	7.80	1.56	9.54	55.37	44.88
<i>HeaviSine</i>	3	*8.21	33.56	8.49	34.26	240.72	110.49
<i>Doppler</i>	7	1.51	6.67	*1.08	8.26	14.87	12.01
<i>Doppler</i>	3	*8.10	22.14	8.25	19.95	81.48	50.33

most of the time. It means that the reconstruction is closer to the truth. In cross validation, we only focus on the errors of  $f$  ignoring that in  $f'$ . So the reconstruction of  $f'$  is not as smooth as that in  $f$ , which does not affect trajectory reconstruction. A drawback of tractor spline is the cost in finding local minimal parameters, where the cross validation algorithm returns a smaller score. So we optimized our code to make it runs as faster as it can.

## 1.6 Introduction

### 1.6.1 Spline

In interpolation and curve fitting, piecewise linear approximation may not have the practical significance of cubic spline, or even higher order, approximation. These "broken lines" are neither very smooth nor very efficient approximation. Researchers can go to piecewise polynomial approximation with higher order pieces De Boor *et al.* (1978), which is called spline method. A spline is a numeric function that is piecewise-defined by polynomial functions, and which possesses a high degree of smoothness at the places where the polynomial pieces connect (known as knots) Judd (1998)Chen (2009). Suppose we are given observed data  $t_1, t_2, \dots, t_n$  on interval  $[0, 1]$ , satisfying  $0 \leq t_1 < t_2 < \dots < t_n \leq 1$ . A piecewise polynomial function  $f(t)$  can be obtained by

dividing the interval into contiguous intervals  $(t_1, t_2), \dots, (t_{n-1}, t_n)$  and represented by a separate polynomial in each interval. For any continuous  $f \in \mathbb{C}^{(m)}[0, 1]$ , it can be represented in a linear combination of basis functions  $h_m(t)$ , just as every vector in a vector space can be represented as a linear combination of basis vectors. So we have

$$f(t) = \sum_{m=1}^M \beta_m h_m(t), \quad (1.45)$$

where  $\beta_m$  are coefficients Ellis *et al.* (2009).

Suppose we were attempt to fit a model of  $f(t)$  by least squares without any restrictions, the best fitting  $\hat{f}(t)$  would go through every given data to reduce sum of squares to zero. Most of the time, the results are unsatisfactory as explanations of the given data. The roughness penalty approach is introduced to quantify the notion of a rapidly fluctuating curve and then to pose the estimation problem in a way that makes explicit the necessary compromise between varying rapidly fluctuation and slowly trend in curve estimation Green and Silverman (1993). By adding a penalty term  $\int_0^1 (f^{(m)}(t))^2 dt$ , the curve estimate  $\hat{f}(t)$  exists and is unique over all spline functions  $f(t)$  with  $m - 1$  continuous derivatives fitting observed data in the space  $\mathbb{C}^{(m)}[0, 1]$ , and it can be found by minimizing the following penalized mean residual sum squares

$$\text{MSE}(f, \lambda) = \frac{1}{n} \sum_{i=1}^n (f(t_i) - y_i)^2 + \lambda \int_0^1 (f^{(m)}(t))^2 dt, \quad (1.46)$$

where  $\lambda$  is a fixed smoothing parameter,  $(t_i, y_i)$ ,  $i = 1, \dots, n$  are observed data and  $0 \leq t_1 < t_2 < \dots < t_n \leq 1$ . In equation (1.46), the smoothing parameter  $\lambda$  controls the trade-off between over-fitting and bias. Smoothing spline provides a powerful tool to estimate nonparametric functions Hastie and Tibshirani (1990).

### 1.6.2 Gaussian Process Regression

A Gaussian Process is a collection of random variables, any finite number of which have a joint Gaussian distribution Rasmussen and Williams (2006). It is fully defined by its mean  $m(t)$  and covariance  $K(s, t)$  functions as

$$m(t) = \mathbb{E}[f(t)] \quad (1.47)$$

$$K(s, t) = \mathbb{E}[(f(s) - m(s))(f(t) - m(t))], \quad (1.48)$$

where  $s$  and  $t$  are two variables, and a function  $f$  distributed as such is denoted in form of

$$f \sim GP(m(t), K(s, t)). \quad (1.49)$$

Usually the mean function is assumed to be zero everywhere.

Given a set of input variables  $\mathbf{T}$  for function  $f(t)$  and the output  $\mathbf{y} = f(\mathbf{T}) + \varepsilon$  with independent identically distributed Gaussian noise  $\varepsilon$  with variance  $\sigma_n^2$ , we can use the above definition to predict the value of the function  $f_* = f(t_*)$  at a particular input  $t_*$ . As the noisy observations becoming

$$\text{cov}(y_p, y_q) = K(t_p, t_q) + \sigma_n^2 \delta_{pq} \quad (1.50)$$

where  $\delta_{pq}$  is a Kronecker delta which is one iff  $p = q$  and zero otherwise, the joint distribution of the observed outputs  $\mathbf{y}$  and the estimated output  $f_*$  according to prior is

$$\begin{bmatrix} \mathbf{y} \\ f_* \end{bmatrix} \sim N \left( 0, \begin{bmatrix} K(\mathbf{T}, \mathbf{T}) + \sigma_n^2 I & K(\mathbf{T}, t_*) \\ K(t_*, \mathbf{T}) & K(t_*, t_*) \end{bmatrix} \right). \quad (1.51)$$

The posterior distribution over the predicted value is obtained by conditioning on the observed data

$$f_* | \mathbf{y}, \mathbf{T}, t_* \sim N(\bar{f}_*, \text{cov}(f_*)) \quad (1.52)$$

where

$$\bar{f}_* = \mathbb{E}[f_* | \mathbf{y}, \mathbf{T}, t_*] = K(t_*, \mathbf{T})[K(\mathbf{T}, \mathbf{T}) + \sigma_n^2 I]^{-1} \mathbf{y}, \quad (1.53)$$

$$\text{cov}(f_*) = K(t_*, t_*) - K(t_*, \mathbf{T})[K(\mathbf{T}, \mathbf{T}) + \sigma_n^2 I]^{-1} K(\mathbf{T}, t_*). \quad (1.54)$$

### 1.6.3 The Smoothing Spline as Bayes Estimates

It is possible to interpret the smoothing spline regression estimator as a Bayesian estimate when the mean function  $r(\cdot)$  is given an improper prior distribution. Berline and Thomas-Agnan (2011) Wahba (1990)

Consider the model

$$y_i = f(t_i) + \varepsilon_i, \quad (1.55)$$

where  $i = 1, \dots, n$ ,  $\varepsilon_i$  are i.i.d. Gaussian distributed noise with variance  $\sigma^2$ . Assume  $f \in \mathbb{H}^{(m)}[0, 1]$ , where

$$\mathbb{H}^{(m)}[0, 1] = \{f : f^{(\nu)} \text{ absolutely continuous}, \nu = 0, \dots, m-1, \int_0^1 (f^{(m)}(t))^2 dt < \infty\}. \quad (1.56)$$

A smoothing spline  $\hat{f}_\lambda$  is the minimizer of objective function (1.46) Wang (1998). Equipped with an appropriate inner product

$$\langle f, g \rangle = \sum_{\nu=0}^{m-1} f^{(\nu)}(0)g^{(\nu)}(0) + \int_0^1 f^{(m)}g^{(m)}dt, \quad (1.57)$$

the space  $\mathbb{H}^{(m)}[0, 1]$  becomes a reproducing kernel Hilbert space.

Let  $\phi_\nu(t) = \frac{t^{\nu-1}}{(\nu-1)!}$  where  $\nu = 1, \dots, m$  and  $R_1(s, t) = \int_0^1 \frac{(s-u)_+^{m-1}}{(m-1)!} \frac{(t-u)_+^{m-1}}{(m-1)!} du$ . Denote  $S = \{\phi_\nu(t_i)\}_{n \times m}$  where  $i = 1, \dots, n, \nu = 1, \dots, m$  and  $Q = \{R_1(t_i, t_j)\}_{n \times n}$  where  $i = 1, \dots, n, j = 1, \dots, n$ . Kimeldorf and Wahba (1971) and Kimeldorf and Wahba (1970) proved that  $\hat{f}_\lambda$  has the form

$$\hat{f}(t) = \sum_{\nu=1}^m d_\nu \phi_\nu(t) + \sum_{i=1}^n c_i R_1(t, t_i). \quad (1.58)$$

By denoting  $M = Q + n\lambda I$ , Gu (2013) found that the coefficients will be given by

$$\mathbf{c} = (M^{-1} - M^{-1}S(S^\top M^{-1}S)^{-1}S^\top M^{-1})\mathbf{Y}, \quad (1.59)$$

$$\mathbf{d} = (S^\top M^{-1}S)^{-1}S^\top M^{-1}\mathbf{Y}. \quad (1.60)$$

## 1.7 A reproducing kernel on $\mathcal{C}_{p.w.}^2[0, 1]$

The minimizer  $\eta(x)$  of

$$\frac{1}{n} \sum_{i=1}^n (Y_i - \eta(x_i))^2 + \frac{\gamma}{n} \sum_{i=1}^n (v_i - \eta'(x_i))^2 + \lambda \int_0^1 \eta''^2 dx \quad (1.61)$$

in the space  $\mathcal{C}_{p.w.}^2[0, 1] = \{f : f, f' \text{ are continuous and } f'' \text{ is piecewise continuous on } [0, 1]\}$  is a tractor spline. Equipped with an appropriate inner product

$$(f, g) = f(0)g(0) + f'(0)g'(0) + \int_0^1 f''g'' dx, \quad (1.62)$$

the space  $\mathcal{C}_{p.w.}^2[0, 1]$  is made a reproducing kernel Hilbert space. In fact, the representer  $R_x(\cdot)$  is

$$R_x(y) = 1 + xy + \int_0^1 (x-u)_+(y-u)_+ du. \quad (1.63)$$

It can be seen that  $R_x(0) = 1$ ,  $R'_x(0) = x$ , and  $R''_x(y) = (x-y)_+$ .

The two terms of the reproducing kernel  $R(x, y) = R_x(y) = R_0(x, y) + R_1(x, y)$ , where

$$R_0(x, y) = 1 + xy \quad (1.64)$$

$$R_1(x, y) = \int_0^1 (x-u)_+(y-u)_+ du \quad (1.65)$$

are both non-negative definite themselves.

**Theorem 4.** *If the reproducing kernel  $R$  of a space  $\mathcal{H}$  on domain  $X$  can be decomposed into  $R = R_0 + R_1$ , where  $R_0$  and  $R_1$  are both non-negative definite,  $R_0(x, \cdot), R_1(x, \cdot) \in \mathcal{H}$ ,  $\forall x \in X$ , and  $(R_0(x, \cdot), R_1(y, \cdot)) = 0$ ,  $\forall x, y \in X$ , then the spaces  $\mathcal{H}_0$  and  $\mathcal{H}_1$  corresponding respectively to  $R_0$  and  $R_1$  form a tensor sum decomposition of  $\mathcal{H}$ . Conversely, if  $R_0$  and  $R_1$  are both non-negative definite and  $\mathcal{H}_0 \cap \mathcal{H}_1 = \{0\}$ , then  $\mathcal{H} = \mathcal{H}_0 \oplus \mathcal{H}_1$  has a reproducing kernel  $R = R_0 + R_1$ .*

According to Theorem 1,  $R_0$  can correspond the space of polynomials  $\mathcal{H}_0 = \{f : f'' = 0\}$  with an inner product  $(f, g)_0 = f(0)g(0) + f'(0)g'(0)$ , and  $R_1$  corresponds the orthogonal complement of  $\mathcal{H}_0$

$$\mathcal{H}_1 = \{f : f(0) = 0, f'(0) = 0, \int_0^1 f''^2 dx < \infty\}$$

with inner product  $(f, g)_1 = \int_0^1 f''g'' dx$ . Thus,  $\mathcal{H}_0$  and  $\mathcal{H}_1$  are two subspaces of the  $\mathcal{C}_{p.w.}^2[0, 1]$ , and the reproducing kernel is  $R_x(\cdot) = R_0(x, \cdot) + R_1(x, \cdot)$ .

Define a new notation  $\dot{R}(x, y) = \frac{\partial R}{\partial x}(x, y) = \frac{\partial R_0}{\partial x}(x, y) + \frac{\partial R_1}{\partial x}(x, y) = y + \int_0^x (y-u)_+ du$ . Obviously  $\dot{R}_x(y) \in \mathcal{C}_{p.w.}^2[0, 1]$ . Additionally, we have  $\dot{R}_x(0) = 0$ ,  $\dot{R}'_x(0) = \frac{\partial \dot{R}_x}{\partial y}(0) = 1$ , and  $\dot{R}''_x(y) = \begin{cases} 0 & x \leq y \\ 1 & x > y \end{cases}$ . Then, for any  $f \in \mathcal{C}_{p.w.}^2[0, 1]$ , it gives us

$$(\dot{R}_x, f) = \dot{R}_x(0)f(0) + \dot{R}'_x(0)f'(0) + \int_0^1 \dot{R}''_x f'' du = f'(0) + \int_0^y f'' du = f'(y).$$

It can be seen that the first term  $\dot{R}_0 = y \in \mathcal{H}_0$ , and the space spanned by the second term  $\dot{R}_1 = \int_0^x (y-u)_+ du$ , denoted as  $\dot{\mathcal{H}}$ , is not in  $\mathcal{H}_1$ , but  $\dot{\mathcal{H}} \cap \mathcal{H}_1 \neq \emptyset$ . Then we have a new space  $\mathcal{H}_* = \dot{\mathcal{H}} \cup \mathcal{H}_1$ . Thus the two new sub spaces in  $\mathcal{C}_{p.w.}^2[0, 1]$  are  $\mathcal{H}_0$  and  $\mathcal{H}_*$ .

## 1.8 Computation of Polynomial Smoothing Splines

Given the sample points  $x_j, j = 1, \dots, n$  in equation (1.61) and noting that the space

$$\mathcal{A} = \{f : f = \sum_{j=1}^n \alpha_j R_1(x_j, \cdot) + \sum_{j=1}^n \beta_j \dot{R}_1(x_j, \cdot)\} \quad (1.66)$$

is a closed linear subspace of  $\mathcal{H}_*$ . Then  $\eta \in \mathcal{C}_{p.w.}^2[0, 1]$  can be written as

$$\eta(x) = d_1 + d_2 x + \sum_{j=1}^n c_j R_1(x_j, x) + \sum_{i=j}^n b_j \dot{R}_1(x_j, \cdot) + \rho(x) \quad (1.67)$$

where  $\mathbf{d}, \mathbf{c}$  and  $\mathbf{b}$  are coefficients, and  $\rho(x) \in \mathcal{H}_* \ominus \mathcal{A}$ .



The equation (1.61) can be written as

$$\begin{aligned} & \frac{1}{n} \sum_{i=1}^n \left( Y_i - d_1 - d_2 x - \sum_{j=1}^n c_j R_1(x_j, x_i) - \sum_{j=1}^n b_j \dot{R}_1(x_j, x_i) - \rho(x_i) \right)^2 \\ & \frac{\gamma}{n} \sum_{i=1}^n \left( V_i - d_2 - \sum_{j=1}^n c_j R'_1(x_j, x_i) - \sum_{j=1}^n b_j \dot{R}'_1(x_j, x_i) - \rho'(x_i) \right)^2 \\ & + \lambda \int_0^1 \left( \sum_{j=1}^n c_j R''_1(x_j, x) + \sum_{j=1}^n c_j \dot{R}''_1(x_j, x) + \rho''(x) \right)^2 dx \end{aligned}$$

By orthogonality,  $\rho(x_i) = (R_1(x_i, \cdot), \rho) = 0$ ,  $\rho'(x_i) = (\dot{R}_1(x_i, \cdot), \rho') = 0$ ,  $i = 1, \dots, n$ . Denoting by

$$\begin{aligned} S &= \{S_{ij}\}_{n \times 2} = \begin{bmatrix} 1 & x_i \end{bmatrix}, \quad Q = \{Q_{ij}\}_{n \times n} = R_1(x_j, x_i), \quad P = \{P_{ij}\}_{n \times n} = \dot{R}_1(x_j, x_i), \\ S' &= \{S'_{ij}\}_{n \times 2} = \begin{bmatrix} 0 & 1 \end{bmatrix}, \quad Q' = \{Q'_{ij}\}_{n \times n} = R'_1(x_j, x_i), \quad P' = \{P'_{ij}\}_{n \times n} = \dot{R}'_1(x_j, x_i). \end{aligned}$$

and noting that  $\int_0^1 R''_1(x_i, x) R''_1(x_j, x) dx = R_1(x_i, x_j)$ ,  $\int_0^1 R''_1(x_i, x) \dot{R}''_1(x_j, x) dx = \int_0^v (x_i - x) dx = \dot{R}_1(x_j, x_i)$ , and  $\int_0^1 \dot{R}''_1(x_i, x) \dot{R}''_1(x_j, x) dx = \int_0^v 1 dx = \dot{R}'_1(x_i, x_j)$ , where  $v = \min(x_i, x_j)$ , the above equation can be written as

$$\begin{aligned} & (\mathbf{Y} - S\mathbf{d} - Q\mathbf{c} - P\mathbf{b})^\top (\mathbf{Y} - S\mathbf{d} - Q\mathbf{c} - P\mathbf{b}) + \gamma (\mathbf{V} - S'\mathbf{d} - Q'\mathbf{c} - P'\mathbf{b})^\top (\mathbf{V} - S'\mathbf{d} - Q'\mathbf{c} - P'\mathbf{b}) \\ & + n\lambda (\mathbf{c}^\top Q\mathbf{c} + 2\mathbf{c}^\top P\mathbf{b} + \mathbf{b}^\top P'\mathbf{b}) + n\lambda(\rho, \rho). \end{aligned} \quad (1.68)$$

Note that  $\rho$  only appears in the third term in (1.68), which is minimised at  $\rho = 0$ . Hence, a polynomial smoothing spline resides in the space  $\mathcal{H}_0 \oplus \mathcal{A}$  of finite dimension. Then the solution to (1.61) could be computed via minimization of the first two terms in (1.68) with respect to  $\mathbf{d}$ ,  $\mathbf{c}$  and  $\mathbf{b}$ .

## 1.9 Polynomial Smoothing Splines as Bayes Estimates

Now in the model  $Y = \eta(x) + \epsilon$  and  $V = \eta'(x) + \frac{\epsilon}{\gamma}$  where  $\epsilon \sim N(0, \sigma^2)$ , according to equation (1.67), for  $\eta(x) \in \mathcal{C}_{p.w.}^2[0, 1]$  and  $x \in X$ , we have

$$\eta(x) = (d_1 + d_2 x) + \sum_{i=1}^n c_i R_1(x_i, x) + \sum_{i=1}^n b_i \dot{R}_1(x_i, x). \quad (1.69)$$

The covariance functions for  $Y, V$  and  $\eta, \eta'$  are

$$\begin{aligned}
\mathbb{E}(\eta(x)\eta(y)) &= \tau^2 R_0(x, y) + \beta R_1(x, y) & \mathbb{E}(\eta(x)\eta'(y)) &= \tau^2 R'_0(x, y) + \beta R'_1(x, y) \\
\mathbb{E}(\eta'(x)\eta(y)) &= \tau^2 \dot{R}_0(x, y) + \beta \dot{R}_1(x, y) & \mathbb{E}(\eta'(x)\eta'(y)) &= \tau^2 \dot{R}'_0(x, y) + \beta \dot{R}'_1(x, y) \\
\mathbb{E}(y_i, y_j) &= \tau^2 R_0(x_i, x_j) + \beta R_1(x_i, x_j) & \mathbb{E}(v_i, v_j) &= \tau^2 \dot{R}'_0(x_i, x_j) + \beta \dot{R}'_1(x_i, x_j) \\
&+ \sigma^2 \delta_{ij} & &+ \frac{\sigma^2}{\gamma} \delta_{ij} \\
\mathbb{E}(v_i, y_j) &= \tau^2 \dot{R}_0(x_i, x_j) + \beta \dot{R}_1(x_i, x_j) & \mathbb{E}(y_i, v_j) &= \tau^2 R'_0(x_i, x_j) + \beta R'_1(x_i, x_j) \\
\mathbb{E}(y_i, \eta(x)) &= \tau^2 R_0(x_i, x) + \beta R_1(x_i, x) & \mathbb{E}(y_i, \eta'(x)) &= \tau^2 R'_0(x_i, x) + \beta R'_1(x_i, x) \\
\mathbb{E}(v_i, \eta(x)) &= \tau^2 \dot{R}_0(x_i, x) + \beta \dot{R}_1(x_i, x) & \mathbb{E}(v_i, \eta'(x)) &= \tau^2 \dot{R}'_0(x_i, x) + \beta \dot{R}'_1(x_i, x)
\end{aligned}$$

where  $R(x, y)$  is taken from (1.63).

Observing  $Y_i \sim N(\eta(x_i), \sigma^2)$  and  $V_i \sim N(\eta(x_i), \frac{\sigma^2}{\gamma})$ , the joint distribution of  $Y, V$  and  $\eta(x)$  is normal with mean zero and a covariance matrix can be found. The posterior mean of  $\eta(x)$  is

$$\begin{aligned}
\mathbb{E}(\eta|Y, V) &= \begin{bmatrix} \text{cov}(Y, \eta) & \text{cov}(\eta, V) \end{bmatrix} \begin{bmatrix} \text{var}(Y) & \text{cov}(Y, V) \\ \text{cov}(V, Y) & \text{var}(V) \end{bmatrix}^{-1} \begin{bmatrix} Y \\ V \end{bmatrix} \\
&= \begin{bmatrix} \tau^2 \phi^\top S^\top + \beta \xi^\top & \tau^2 \phi^\top S'^\top + \beta \psi^\top \end{bmatrix} \begin{bmatrix} \tau^2 S S^\top + \beta Q + \sigma^2 I & \tau^2 S S'^\top + \beta P \\ \tau^2 S' S^\top + \beta Q' & \tau^2 S' S'^\top + \beta P' + \frac{\sigma^2}{\gamma} I \end{bmatrix}^{-1} \begin{bmatrix} Y \\ V \end{bmatrix} \\
&= \begin{bmatrix} \rho \phi^\top S^\top + \xi^\top & \rho \phi^\top S'^\top + \psi^\top \end{bmatrix} \begin{bmatrix} \rho S S^\top + Q + n\lambda I & \rho S S'^\top + P \\ \rho S' S^\top + Q' & \rho S' S'^\top + P' + \frac{n\lambda}{\gamma} I \end{bmatrix}^{-1} \begin{bmatrix} Y \\ V \end{bmatrix} \\
&= (\phi^\top \rho \begin{bmatrix} S \\ S' \end{bmatrix}^\top + \begin{bmatrix} \xi^\top & \psi^\top \end{bmatrix}) \left( \rho \begin{bmatrix} S \\ S' \end{bmatrix}^\top \begin{bmatrix} S \\ S' \end{bmatrix} + \begin{bmatrix} Q + n\lambda I & P \\ Q' & P' + \frac{n\lambda}{\gamma} I \end{bmatrix} \right)^{-1} \begin{bmatrix} Y \\ V \end{bmatrix} \\
&\triangleq \phi^\top \rho T^\top (\rho T^\top T + M)^{-1} \begin{bmatrix} Y \\ V \end{bmatrix} + \begin{bmatrix} \xi^\top & \psi^\top \end{bmatrix} (\rho T^\top T + M)^{-1} \begin{bmatrix} Y \\ V \end{bmatrix}
\end{aligned} \tag{1.70}$$

where  $\phi$  is  $2 \times 1$  matrix with entry 1 and  $x$ ,  $\xi$  is  $n \times 1$  matrix with  $i$ th entry  $R(x_i, x)$  and  $\psi$  is  $n \times 1$  matrix with  $i$ th entry  $\dot{R}(x_i, x)$ ,  $\rho = \tau^2/\beta$  and  $n\lambda = \sigma^2/\beta$ .

**Lemma 3.** Suppose  $M$  is symmetric and nonsingular and  $S$  is of full column rank.

$$\begin{aligned}
\lim_{\rho \rightarrow \infty} (\rho T T^\top + M)^{-1} &= M^{-1} - M^{-1} T (T^\top M^{-1} T)^{-1} T^\top M^{-1}, \\
\lim_{\rho \rightarrow \infty} \rho T^\top (\rho T T^\top + M)^{-1} &= (T^\top M^{-1} T)^{-1} T^\top M^{-1}.
\end{aligned}$$

Setting  $\rho \rightarrow \infty$  in equation (1.70) and applying Lemma 3, the posterior mean  $\mathbb{E}(\eta(x)|\mathbf{Y}, \mathbf{V})$  is of the form  $\eta = \phi^\top \mathbf{d} + \xi^\top \mathbf{c} + \psi^\top \mathbf{b}$ , with the coefficients given by

$$\mathbf{d} = (T^\top M^{-1} T)^{-1} T^\top M^{-1} \begin{bmatrix} Y \\ V \end{bmatrix},$$

$$\begin{bmatrix} \mathbf{c} \\ \mathbf{b} \end{bmatrix} = (M^{-1} - M^{-1} T (T^\top M^{-1} T)^{-1} T^\top M^{-1}) \begin{bmatrix} Y \\ V \end{bmatrix},$$

where  $T = \begin{bmatrix} S \\ S' \end{bmatrix}$  and  $M = \begin{bmatrix} Q + n\lambda I & P \\ Q' & P' + \frac{n\lambda}{\gamma} I \end{bmatrix}$ .

It is easy to verify that  $d, c, b$  are the solutions to

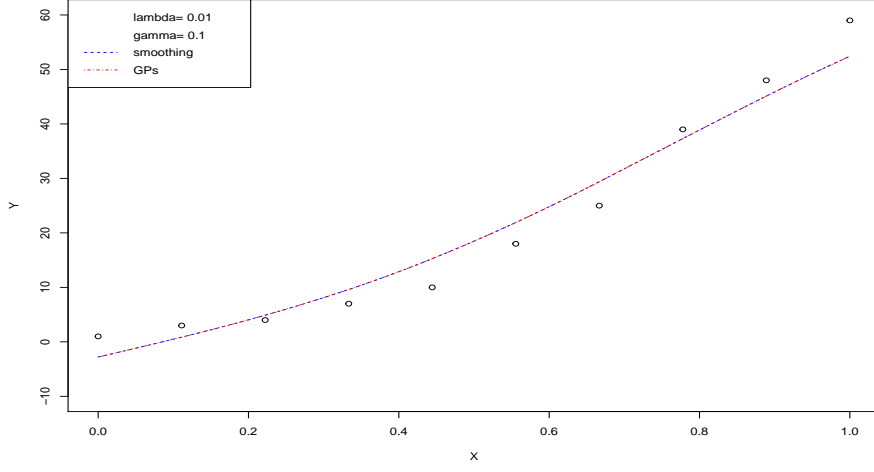
$$\begin{cases} S^\top (Sd + Qc + Pb - Y) + \gamma S'^\top (S'd + P^\top c + S'^\top P'b - V) = 0, \\ Q(Sd + (Q + n\lambda I)c + Pb - Y) + P(\gamma S'd + \gamma P^\top c + (\gamma P' + n\lambda I)b - \gamma V) = 0, \\ P^\top (Sd + (Q + n\lambda I)c + Pb - Y) + P'(\gamma S'b + P^\top c + (\gamma P' + n\lambda I)b - \gamma V) = 0. \end{cases}$$

## 1.10 Numeric Simulation of Smoothing Spline and GPR

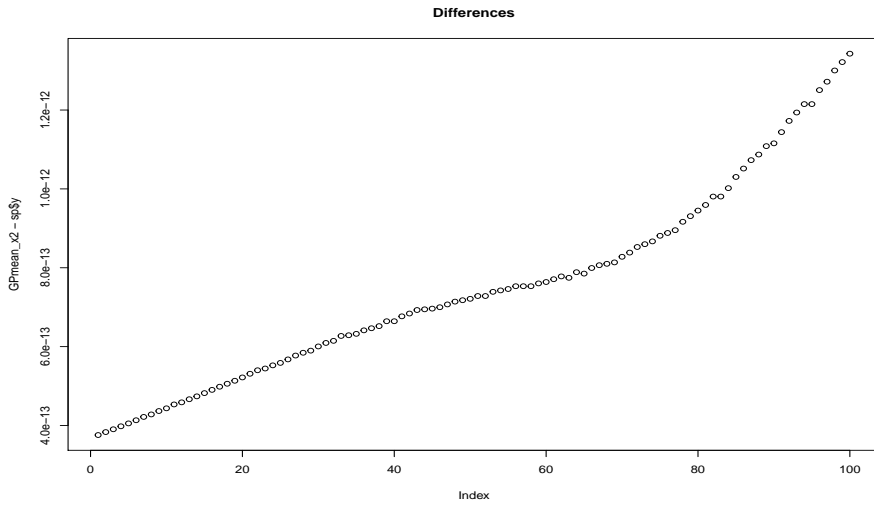
Given  $\mathbf{X}$  a length-10 sequence from 0 to 1,

$$\mathbf{Y} = [1, 3, 4, 7, 10, 18, 25, 39, 48, 59],$$

$$v_i = \begin{cases} \frac{y_{i+1} - y_i}{x_{i+1} - x_i} & \text{if } 1 \leq i \leq 9 \\ 0 & \text{if } i = 10 \end{cases}. \text{ A simulated result is in figure 1.}$$



(a)



(b)

Figure 1.8: (a) Comparing two methods under the same parameters  $\lambda = 0.01$  and  $\gamma = 0.1$ . In this graph, the blue line is reconstruction from tractor spline, the red line is the mean of Gaussian Process, which is the posterior  $\mathbb{E}(\eta(x)|\mathbf{Y}, \mathbf{V})$ . (b) The differences between two methods under the same parameters.

# References

- Abramovich, F., Sapatinas, T., and Silverman, B. W. (1998). Wavelet thresholding via a Bayesian approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(4), 725–749.
- Agarwal, P. K., Arge, L., and Erickson, J. (2003). Indexing moving points. *Journal of Computer and System Sciences*, 66(1), 207–243.
- Aydin, D. and Tuzemen, M. S. (2012). Smoothing parameter selection problem in nonparametric regression based on smoothing spline: A simulation study. *Journal of Applied Sciences*, 12(7), 636.
- Berlinet, A. and Thomas-Agnan, C. (2011). *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media.
- Chen, W.-K. (2009). *Feedback, nonlinear, and distributed circuits*. CRC Press.
- Craven, P. and Wahba, G. (1978). Smoothing noisy data with spline functions. *Numerische Mathematik*, 31(4), 377–403.
- De Boor, C., De Boor, C., Mathématicien, E.-U., De Boor, C., and De Boor, C. (1978). *A practical guide to splines*, Volume 27. Springer-Verlag New York.
- Donoho, D. L. and Johnstone, I. M. (1995). Adapting to unknown smoothness via wavelet shrinkage. *Journal of the american statistical association*, 90(432), 1200–1224.
- Donoho, D. L., Johnstone, I. M., Kerkycharian, G., and Picard, D. (1995). Wavelet shrinkage: asymptopia? *Journal of the Royal Statistical Society. Series B (Methodological)*, 301–369.
- Donoho, D. L. and Johnstone, J. M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3), 425–455.

- Ellis, D., Sommerlade, E., and Reid, I. (2009). Modelling pedestrian trajectory patterns with gaussian processes. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, 1229–1234. IEEE.
- Erkorkmaz, K. and Altintas, Y. (2001). High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation. *International Journal of machine tools and manufacture*, 41(9), 1323–1345.
- Green, P. J. and Silverman, B. W. (1993). *Nonparametric regression and generalized linear models: a roughness penalty approach*. CRC Press.
- Gu, C. (1998). Model indexing and smoothing parameter selection in nonparametric function estimation. *Statistica Sinica*, 607–623.
- Gu, C. (2013). *Smoothing Spline ANOVA Models Second Edition*. Springer New York Heidelberg Dordrecht London.
- Hastie, T. J. and Tibshirani, R. J. (1990). *Generalized additive models*, Volume 43. CRC Press.
- Judd, K. L. (1998). *Numerical methods in economics*. MIT press.
- Kim, Y.-J. and Gu, C. (2004). Smoothing spline Gaussian regression: more scalable computation via efficient approximation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(2), 337–356.
- Kimeldorf, G. and Wahba, G. (1971). Some results on Tchebycheffian spline functions. *Journal of mathematical analysis and applications*, 33(1), 82–95.
- Kimeldorf, G. S. and Wahba, G. (1970). A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41(2), 495–502.
- Liu, Z. and Guo, W. (2010). Data driven adaptive spline smoothing. *Statistica Sinica*, 1143–1163.
- Nason, G. (2010). *Wavelet methods in statistics with R*. Springer Science & Business Media.
- Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.

- Schwarz, K.-P. (2012). *Geodesy Beyond 2000: The Challenges of the First Decade, IAG General Assembly Birmingham, July 19–30, 1999*, Volume 121. Springer Science & Business Media.
- Sealfon, C., Verde, L., and Jimenez, R. (2005). Smoothing spline primordial power spectrum reconstruction. *Physical Review D*, 72(10), 103520.
- Silverman, B. W. (1985). Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society. Series B (Methodological)*, 1–52.
- Trevor Hastie, Robert Tibshirani, J. F. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second Edition*. Springer-Verlag.
- Wahba, G. (1985). A comparison of GCV and GML for choosing the smoothing parameter in the generalized spline smoothing problem. *The Annals of Statistics*, 1378–1402.
- Wahba, G. (1990). *Spline models for observational data*, Volume 59. Siam.
- Wang, Y. (1998). Smoothing spline models with correlated random errors. *Journal of the American Statistical Association*, 93(441), 341–348.
- Yang, K. and Sukkarieh, S. (2010). An analytical continuous-curvature path-smoothing algorithm. *Robotics, IEEE Transactions on*, 26(3), 561–568.
- Yao, F., Müller, H.-G., Wang, J.-L., *et al.* (2005). Functional linear regression analysis for longitudinal data. *The Annals of Statistics*, 33(6), 2873–2903.
- Yu, B., Kim, S. H., Bailey, T., and Gamboa, R. (2004). Curve-based representation of moving object trajectories. In *Database Engineering and Applications Symposium, 2004. IDEAS’04. Proceedings. International*, 419–425. IEEE.
- Zhang, K., Guo, J.-X., and Gao, X.-S. (2013). Cubic spline trajectory generation with axis jerk and tracking error constraints. *International Journal of Precision Engineering and Manufacturing*, 14(7), 1141–1146.

