Regression Spline Fitting with Applications

C. G. Walker, M. J. O'Sullivan, Y. Zhu, S. Hsu
Department of Engineering Science
University of Auckland
New Zealand
cameron.walker@auckland.ac.nz

M. L. MacKenzie, C. R. Donovan
School of Mathematics and Staistics
University of St Andrews
Scotland

V. Rajagopal
Auckland Bioengineering Institute
University of Auckland
New Zealand

Abstract

Deciding how best to fit regression splines to data is a difficult non-linear optimization problem. In this paper we present a method for determining good fits using a spacially adaptive local smoothing algorithm (SALSA). We present results that show our method generates models that fit as well as those generated by techniques using smoothing splines, and discuss an application of our technique that enables the automatic landmarking of certain object boundaries.

Key words: Regression Splines, Shape Analysis, Landmarks, Clustering, PCA.

1 Introduction

Splines are a useful function-type to use in Regression when the relationship between a response and a set of covariates is not known *a priori*. Their benefits are well documented (see for example (DeBoor 1978; Schumaker 1993)). In particular, they share many of the nice mathematical properties of polynomials, without the global behaviour that can be problematic when fitting a regression model (often it is required that the change to a value in one region should not influence the fit in another region). The use of splines within the regression framework has been largely influenced by the work of Hastie and Tibshirani (Hastie and Tibshirani 1986; Hastie and Tibshirani 1990) on Generalized Additive Models (GAMs).

Approaches in spline-based regression for balancing fit to the signal (the mean) and fit to the noise (the data variation from the mean) include controlling the number and location of knots (Wold 1974) or including a penalty term in the fitness criteria (Eilers and Marx 1996; Wood 2000; Ruppert 2002). Recent work on the second approach has concentrated on implementating locally adaptive smoothing parameters (Ruppert and Carroll 2000; Baladandayuthapani, Mallick, and Carroll 2005; Crainiceanu et al. 2007; Krivobokova, Crainiceanu, and Kauermann 2008; Wood et al. 2008). Historically the first approach to regression spline fitting, adaptively placing knots, has involved a computer–intensive search. This includes stepwise forward and backward knot selection (TURBO (Friedman and Silverman 1989), MARS (Friedman 1991)), often with guided-search techniques included to reduce the solve time (SARS, (Zhou and Shen 2001)). Bayesian approaches (using the reversible-jump Metropolis-Hastings version of Markov Chain Monte Carlo simulation) have also been implemented in BARS (DiMatteo, Genovese, and Kass 2001; Behseta, Kass, and Wallstrom 2005; Behseta and Kass 2005) and cBARS (Kaufman, Venture, and Kass 2005).

In this paper we first describe a spatially-adaptive local smoothing algorithm (SALSA) (Walker et al. 2010), which automatically chooses the location and number of knots in the spline regression model. This heuristic includes local-search and a restricted forward/backward regression step that significantly reduces the number of models to be evaluated at each iteration, compared to the standard approach (Friedman and Silverman 1989). It performs as well as current alternatives in the literature on established benchmark functions.

Next we explain how this algorithm can be used to automatically determine landmarks on object boundaries, for use in shape analysis. Our approach is demonstrated in two examples. In the first we landmark animal tracks in the ocean (1-dimensional curves in 3-space). In the second we landmark the boundary of nuclei in cardiac cells (2-dimensional surfaces in 3-space).

The paper is set out as follows. In Section 2 we describe how SALSA works, and evaluate it's performance against standard benchmarks functions in the literature in section 3. In section 4 we describe how we have extended SALSA in a couple of applications to perform automatic landmarking, and give some concluding remarks in section 5.

2 Details of SALSA

In this section we provide details of a spatially adaptive local smoothing algorithm (SALSA) for fitting regression splines to data. Our goal is to use a spline to approximate the mean of the response variable at each value (or combination of values) of the explanatory variable(s). This involves deciding the number and location of the knots, as well as the coefficients of the polynomial sections making up the spline. Determining each knot location adds a level of complexity - the minimax polynomial approximation problem can be modelled as a linear program, as opposed to the nonlinear mixed-integer program necessary for a regression spline. Although it is possible to position a knot anywhere in the domain when fitting a regression spline, we consider only data point locations as potential sites, which is standard in practice (see, for example, (Wold 1974; Hastie and Tibshirani 1990)). The fitness measure we use in this paper is the Bayesian Information Criterion (BIC), which

can be calculated for a model fit to n data points from the log-likelihood L and the number of parameters p by $-2 \times L + k \times p$, where $k = log(n)$. In this calculation both p and L are variable, depending on the number and location of the knots. The BIC balances improving the fit to the data against increasing the number of knots used.

SALSA iteratively determines regression splines that better fit the data using three steps. The first is a global exchange step, which enables the movement (addition) of a knot to (at) the worst fit data point in the domain. The second step is local, moving knot positions to neighbouring datapoints. The final step simplifies the model by removing knots from the regression spline.

SALSA:

Calculate s equally-spaced locations E between first and last data points
Initialize knots K with s data locations minimizing $\sum_{i=1}^{s} |K_i - E_i|$
Repeat
    Repeat Exchange step *While* (fit measure is improved)
    Repeat Improvement step *While* (fit measure is improved)
    If ($|K| >$ `minKnots`)
        Perform Simplification step
    End If
While (an improvement in fit measure is made by one of the above steps)

Figure 1: Pseudocode outlining the structure of SALSA

The structure of the algorithm is given in figure 1. We have included a number of parameters in the implementation of our algorithm to increase the user's ability to control basic characteristics of the final model.

Two of the parameters we include are:

`maxKnots`	maximum allowable number of knots;
`minKnots`	minimum allowable number of knots.

2.1 Algorithm Details

The specific details of each algorithm step are given in the remainder of this section.

Exhange Step. In the exchange step the location of the data point furthest from the fitted curve is identified, and regression splines are fit by shifting each existing internal knot from its current location to the position of this point. Where the spline contains less than `maxKnots` knots, a further model is fitted, retaining the current knot locations and including a further knot at this new location. All new models are evaluated, and, where an improvement is obtained, the current model is replaced with the best new model. This step is similar to the forward addition step described by Friedman and Silverman (Friedman and Silverman 1989), but restricting the new knot location to a single data point. This approach requires significantly fewer model evaluations per iteration than the standard forward regression approach. The exchange step has worked well on the benchmark functions we have used for testing, despite the restriction of considering only one new knot location.

Improvement Step. This is a local step, which considers relocating each knot, in turn, to each of its neighbouring data points (where possible). Where the best of these new models is better than the current model, the current model is updated accordingly.

Simplification Step. In this step, new models are obtained from the current model by removing each knot in turn and refitting. Where this results in a better fit, the current model is replaced with the best of these new regression splines. This step is just the standard backward deletion (Friedman and Silverman 1989), and is performed only if the regression spline includes more than `minKnots` knots.

3 Performance of SALSA on Benchmark Functions

In this section, we summarise SALSA's ability to fit to known benchmark functions that have low, moderate and high spatially-variable smoothness.

The performance of SALSA on benchmark functions of low and high spatially variable smoothness was evaluated using functions (Equation 1) proposed by Rupert and Carroll (Ruppert and Carroll 2000) with $n = 400$ equally spaced x's on [0,1] and a signal-to-noise (S/N) ratio of approximately 7 ($\sigma_\epsilon = 0.2$). We set $j = 3$ and $j = 6$ (Figure 2) to represent low and high spatial heterogeneity respectively. These benchmarks are subsequently referred to as the $RC_{j=3}$ and $RC_{j=6}$ functions.

$$f(x_i, j) = \sqrt{x(1-x)}\sin\left\{\frac{2\pi(1+2^{(9-4j)/5})}{x+2^{(9-4j)/5}}\right\} \tag{1}$$

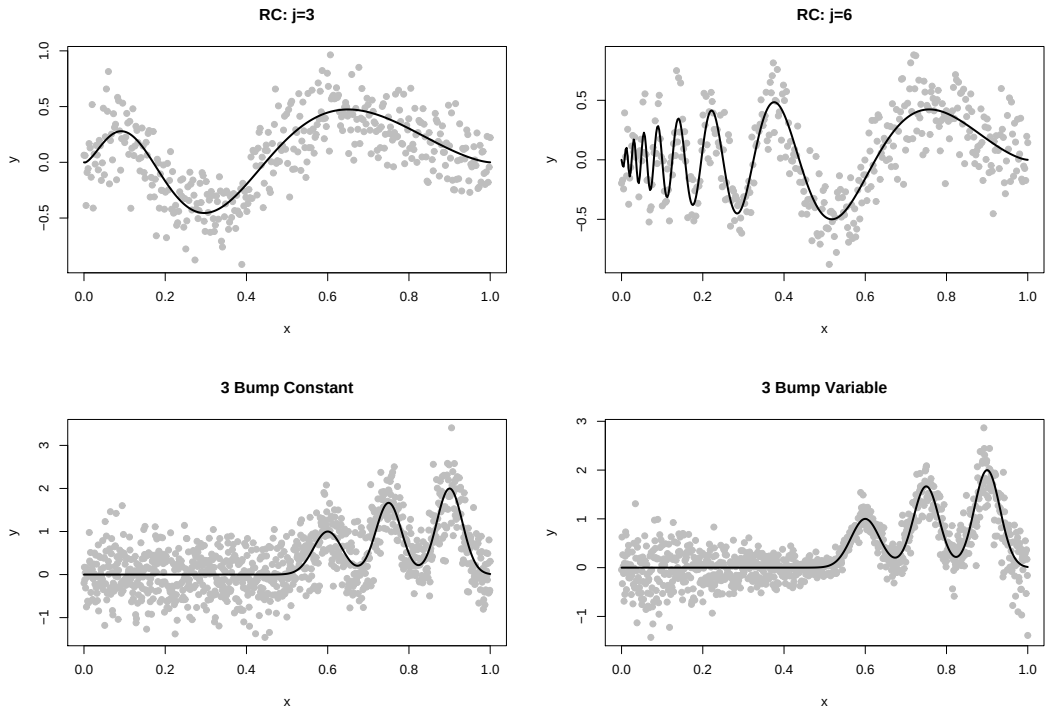

Figure 2: $RC_{j=3}$, $RC_{j=6}$, Three-Bump Constant and Three-Bump Variable data sets. The original function is a solid line, and the simulated data are the points.

The performance of SALSA for moderate spatial heterogeneity was examined using the 'three bump' function (Equation 2, Figure 2). These data sets were generated

using equally spaced x's on [0,1] ($n = 1000$) and with error variance $\sigma_\epsilon = 0.5$. We refer to these functions as the 'Three-Bump Constant' and 'Three-Bump Variable' models.

$$f(x_i) = \exp\{-400(x-0.6)^2\} + \frac{5}{3}\exp\{-500(x-0.75)^2\} + 2\exp\{-500(x-0.9)^2\} \quad (2)$$

3.1 Algorithm performance

We generated 100 data sets for each benchmark function and ran the algorithm under two schemes: one with a fixed number of knots; one where the number of knots can vary. The variable-knot scheme included knot addition, deletion and relocation while the fixed-knot scheme only allowed knot relocation (achieved by setting `maxKnots` to be equal to `minKnots`). We used the B-spline basis (DeBoor 1978) in fitting our model. We permitted both quadratic and cubic bases, using the Bayesian Information Criterion (BIC) to select the degree of the spline and the number and location of knots.

The Average Squared Error (ASE, Equation 3) was used to measure the fidelity of the fitted model to the underlying function for each data set and the mean of these ASE values (MASE) across the 100 data sets was calculated.

$$ASE = \frac{1}{n}\sum_{j=1}^{n}\Big(f(x_i) - \hat{f}(x_i)\Big)^2 \quad (3)$$

Table 1: Mean Average Square Error (MASE) using the three algorithm settings.

Function	Fixed-knot	Variable-knot
$RC_{j=3}$ ($\sigma_\epsilon = 0.2$)	0.00110	0.00127
$RC_{j=6}$ ($\sigma_\epsilon = 0.2$)	0.00452	0.00465
Three-Bump Constant	0.00467	0.00534
Three-Bump Variable	0.00284	0.00332

Under the fixed-knot scheme, SALSA performed at least as well as current spatially adaptive methods for functions with low spatial heterogeneity (Table 1). Our results for the $RC_{j=3}$ function (Equation 1) were very similar to Ruppert and Carroll (Ruppert and Carroll 2000) and Crainiceanu et al. (Crainiceanu et al. 2007), which reported values of 0.0011 and 0.0012 respectively. In contrast, under the variable-knot scheme SALSA returned a MASE that was about 13% larger than when used under the fixed-knot alternative.

Regardless of scheme, SALSA outperformed adaptive alternatives for functions with high spatial heterogeneity. For instance, under the fixed-knot scheme the algorithm returned a MASE for the $RC_{j=6}$ function that was 26% smaller than the algorithms described in (Ruppert and Carroll 2000) and (Crainiceanu et al. 2007) (these reported MASE values of 0.0061 and 0.0065 values respectively). Under the variable-knot scheme our algorithm gave a MASE score 24% smaller than these alternatives.

Under the fixed-knot scheme SALSA also gave smaller MASE values for the Three-Bump Constant function, returning a MASE between 16% and 25% smaller than those reported in (Ruppert and Carroll 2000) (MASE=0.0054), (Crainiceanu et al. 2007) (MASE=0.0055) and (Baladandayuthapani, Mallick, and Carroll 2005)

(MASE=0.0061). Under the variable-knot scheme SALSA gave closer MASE scores to these alternatives.

4 Applications

In this section we describe how to extend the original version of SALSA to fit splines to 1-dimensional curves in 3-dimensional space, and apply this new version of SALSA to automatically landmark sea mammal dives and cell nucleus boundaries.

4.1 SALSA for landmarking curves parametrized by time

Each sea mammal dive is a 1-dimensional curve in 3-dimensional space, so it is necessary to extend the algorithm to accommodate this sort of curve. This is done by parametrizing the x, y and z co-ordinates of the dive by a fourth parameter t (which can be thought of as time in this application, although for an arbitrary closed or open curve this approach will also work). Each of these curves, $x(t)$, $y(t)$ and $z(t)$, is then fitted using the SALSA algorithm, with the regression splines used for the 3 parametric curves constrained to have knots at the same values of t. To fit 3 curves at once the fitness criterion needs to be updated to incorporate a fitness measure from each curve. It is possible to use the residual sums of squares from the 3 paramteric curves to compute a BIC value for the single curve in 3D, but we have found the sum of the BIC values for each paramteric curve to be an effective criterion. The exchange step of SALSA is also updated to identify the value of t that yields the largest magnitude residual across all 3 parametric curves. An example of the approach for the sea mammal dive shown in figure 4 is presented in figure 3.

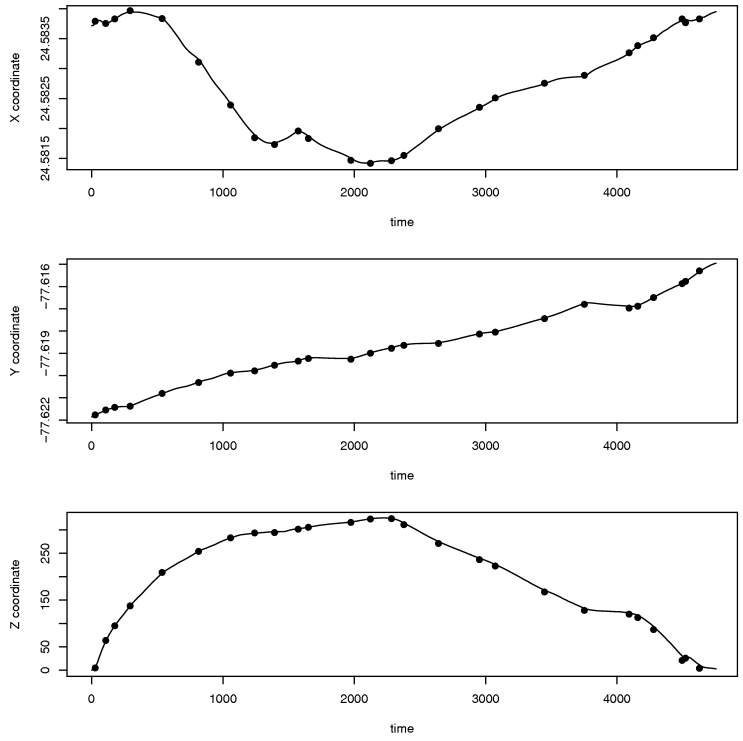

Figure 3: Parametrization of a sea mammal dive - x, y, and z corordinate given in terms of time. A fit is performed using SALSA with degree 1 splines. The solution places 26 knots (shown here as dots), which can be used as landmarks for the original curve in 3D.

The fitted values of the degree d regression splines for $x(t)$, $y(t)$, and $z(t)$ give the coordinates of a degree d spline for the original curve in 3 dimensions. Figure 4 shows a solution for a particular sea mammal dive. The grey curve is the original sea mammal dive, and the black curve is the fitted regression spline, with the black dots showing the location of the knots.

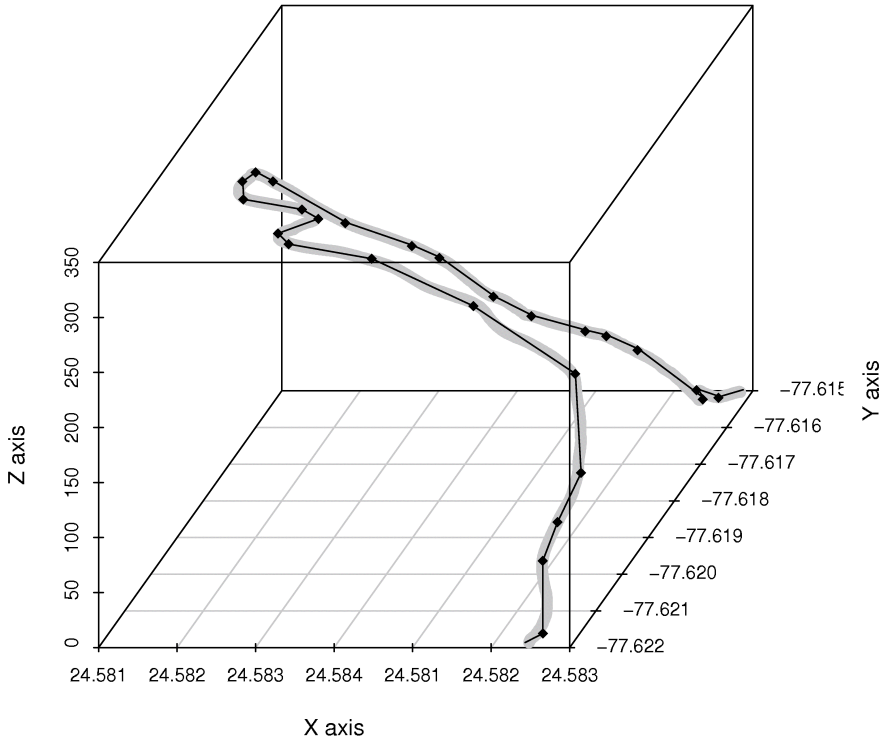

Figure 4: A sea mammal dive (in grey) and the 1d spline fitted using SALSA (in black). The knots for the spline (shown as black dots) can be used as landmarks for a PCA analysis.

The knot points of the regression spline fitted by the parametric SALSA can be used as landmarks for an analysis of the dive shape. SALSA aims to fit the model with the lowest BIC, so it is extremely unlikely that all dives being compared will generate the same number of landmarks (a requirement for the subsequent PCA). To deal with this we do an initial fit for all dives to determine the maximum number of knots K fitted across all the resulting regression splines. SALSA allows the user to define the maximum number (**max**) and minimum number (**min**) of knots to be permitted when fitting a spline. By computing a second fit for all dives with both **max** and **min** set to K we are able to produce a comparable set of landmarks across all the dives. This approach ensures we have chosen the minimum number of landmarks for all dives while still capturing sufficient shape information to satisfy the BIC criterion. For some dives we will capture more shape information than is required, but for our application this is not a problem - one could implement other approaches if desired, such as fixing the number of knots to be the minimum or average across all the fits from the first application of SALSA.

4.2 SALSA for slicing and landmarking 3d objects

To landmark the boundary of a 3-dimensional object, such as the nucleus of a cardiac cell, we have adapted SALSA further. The nucleus is aligned along a "major" axis

and then the boundary segmented by intersecting n planes with the object boundary. The first plane is chosen to include the major axis (and an orthoganol "minor" axis), and each subsequent plane is determined from its predecessor by rotating $\frac{180}{n}$ degrees clockwise around the major axis. This process results in $2n$ equally spaced curves on the object boundary, running from the top of the object (with reference to the minor axis) to the bottom. We use our further adapted version of SALSA to fit degree 1 splines to these $2n$ curves simultaneously, each with knots at the same k heights. Note: it is possible that the object shape may be such that a given boundary curve has more than one point at a given domain value (height) – a situation not possible when a curve is parametrized by time. Thus, although each spline will have knots positioned at the same k heights, a given spline may have more than k knots, and hence the $2n$ splines need not all have the same number of knots. This process is performed to identify k heights at which to take 2-dimensional cross-sections of the object. These heights should capture the shape well (althought the equal spacing of the boundary segments means the choice will not be "optimal"). The approach described in subsection 4.1 is then used to landmark the boundaries of the k 2-dimensional cross-sections. An example of landmarked nucleus is shown in figure 5.

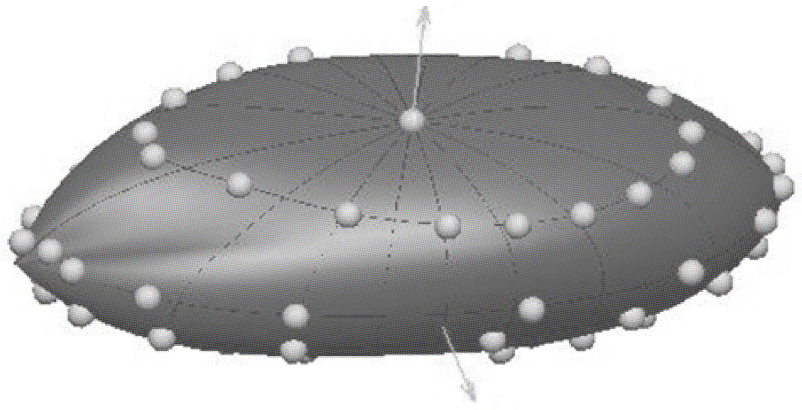

Figure 5: A nucleus from a cardiac cell with landmarks determined using SALSA. The algorithm determined 5 heights at which to slice the nucleus, and fitted splines with 16 knots to each cross-sectional slice.

5 Conclusion

In this paper we have described an algorithm for fitting regression splines to data, and described how extensions of this algorithm can be used to automatically generate landmarks for object boundaries, or 1-dimensional curves (such as animal tracks) in 3-dimensions. In the applications discussed landmarking was the initial step in a more comprehensive study. For the analysis of animal tracks a Principle Components Analysis (PCA) was carried out on the landmarks for a sample of animal dives, and clustering analysis performed on the PCA scores to cluster the dives into types. Of particular interest was identifying whether ensonification affected animal behaviour. For the analysis of cardiac cell nuclei PCA was also used to identify the main modes of variation across a sample of nuclei. The longer term goal is to determine whether there is a difference in nucleus shape for cells from healthy and diabetic tissue.

In both cases landmarking was performed by fitting degree 1 splines using extensions of SALSA (Walker et al. 2010). In all applications the number of landmarks could be determined by the algorithm, given a measure of fit to optimize (for example, the BIC). Alternatively, if only a certain number of landmarks are wanted, parameters could be set to obtain the best fit using only that number of knots in the resulting regression splines.

References

Baladandayuthapani, V., B. K. Mallick, and R. J. Carroll. 2005. "Spatially Adaptive Bayesian Penalized Regression Splines (P-splines)." *Journal of Computational and Graphical Statistics* 14 (2): 378–394.

Behseta, S., and R. E. Kass. 2005. "Testing equality of two functions using BARS." *Statistics in Medicine* 24:3523–3534.

Behseta, S., R. E. Kass, and G. L. Wallstrom. 2005. "Hierachical models for assessing variability among functions." *Biometrika* 92:419–434.

Crainiceanu, C. M., D. Ruppert, R. J. Carroll, A. Joshi, and B. Goodner. 2007. "Spatially Adaptive Bayesian Penalized Splines With Heteroscedastic Errors." *Journal of Computational and Graphical Statistics* 16 (2): 265–288.

DeBoor, C. 1978. *A Practical Guide to Splines*. New York: Springer-Verlag.

DiMatteo, I., C. R. Genovese, and R. E. Kass. 2001. "Bayesian curve-fitting with free-knot splines." *Biometrika* 88:1–67.

Eilers, P. H. C., and B. D. Marx. 1996. "Flexible smoothing with -splines and penalties." *Statistical Science* 11 (2): 115–121.

Friedman, J. H. 1991. "Multivariate adaptive regression splines." *The Annals of Statistics* 19:1–67.

Friedman, Jerome H., and Bernard W. Silverman. 1989. "Flexible Parsimonious Smoothing and Additive Modeling." *Technometrics* 31 (1): 3–29.

Hastie, T., and R. Tibshirani. 1986. "Generalized Additive Models." *Statistical Science* 1 (3): 297–310.

———. 1990. *Generalized Additive Models*. New York: Chapman & Hall.

Kaufman, C. G., V. Venture, and R. E. Kass. 2005. "Spline-based non-parametric regression for periodic functions and its application to directional tuning of neurons." *Statistics in Medicine* 24:2255–2265.

Krivobokova, T., C. M. Crainiceanu, and G. Kauermann. 2008. "Fast adaptive penalized splines." *Journal of Computational and Graphical Statistics* 17:1–20.

Ruppert, D. 2002. "Selecting the Number of Knots for Penalized Splines." *Journal of Computational and Graphical Statistics* 11 (4): 735–757.

Ruppert, D., and R. J. Carroll. 2000. "Spatially-Adaptive Penalties For Spline Fitting." *Australian and New Zealand Journal of Statistics* 42 (2): 205–223.

Schumaker, Larry L. 1993. *Spline Functions: Basic Theory*. Malabar, Fla.: Krieger.

Walker, C. G., M. L. Mackenzie, C. R. Donovan, and M. J. O'Sullivan. 2010. "SALSA a spatially adaptive local smoothing algorithm." *Journal of Statistical Computation and Simulation* http://www.informaworld.com/10.1080/00949650903229041:1–13.

Wold, Svante. 1974. "Spline Functions in Data Analysis." *Technometrics* 16 (1): 1–11.

Wood, S. A., R. C. Kohn, R. Cottet, W. Jiang, and M. Tanner. 2008. "Locally adaptive nonparametric binary regression." *Journal of Computational and Graphical Statistics* 17:352–372.

Wood, S.N. 2000. "Modelling and Smoothing Parameter Estimation with Multiple Quadratic Penalties." *J. R. Statist. Soc. B* 62 (2): 413–428.

Zhou, S., and X. Shen. 2001. "Spatially Adaptive Regression Splines and Accurate Knot Selection Schemes." *Journal of the American Statistical Association* 96 (453): 247–259.