Knot Selection for Regression Splines via the LASSO

M.R. Osborne B. Presnell B.A. Turlach
Centre for Mathematics and its Applications ACSys CRC & CMA
The Australian National University
Canberra ACT 0200
Australia

Key Words: Convex Programming, Dual Problem, Knot Selection, Regression Splines.

Abstract:

Tibshirani (1996) proposes the "Least Absolute Shrinkage and Selection Operator" (lasso) as a method for regression estimation which combines features of shrinkage and variable selection. In this paper we present an algorithm that allows efficient calculation of the lasso estimator. In particular our algorithm can also be used when the number of variables exceeds the number of observations. This algorithm is then applied to the problem of knot selection for regression splines.

1 Introduction

The performance of regression spline smoothing is governed by the choice of knots used in calculating the estimator, and much research effort has been devoted to the difficult problem of knot selection (see, e.g., Wand, 1997; Denison *et al.*, 1998).

Knot selection is not unlike variable selection in linear regression, for which Tibshirani (1996) proposes the least absolute shrinkage and selection operator. The lasso estimator is the solution of the constrained estimation problem

$$\underset{\beta\in\mathbb{R}^m}{\text{minimise}} \qquad f(\beta) = \frac{1}{2}(\mathbf{y}-\mathbf{X}\beta)^T(\mathbf{y}-\mathbf{X}\beta) \tag{1.1a}$$

$$\text{subject to} \qquad g(\beta) = t - \|\beta\|_1 \geq 0, \tag{1.1b}$$

where $\mathbf{y}$ is the vector of observations and $\mathbf{X}$ the design matrix. Thus the lasso solves the least-squares regression problem under a constraint on the l^1 norm of the vector of coefficient estimates.

Assume now that we have data $\{(x_i, y_i)\}_{i=1}^n$, where x_i is univariate and we assume without loss of generality that the x_i are sorted into increasing order. Then we may use the lasso to estimate the knot location for regression splines of order p by constructing the design matrix $\mathbf{X}$ with columns $(\mathbf{x}-x_2)_+^p, (\mathbf{x}-x_3)_+^p, \ldots, (\mathbf{x}-x_{n-1})_+^p, \mathbf{1}_n, (\mathbf{x}-x_1)^1, \ldots (\mathbf{x}-x_1)^p$. Note that this matrix has $n+p-1$ columns, so that the number of columns of $\mathbf{X}$ may exceed the number of rows.

Osborne *et al.* (1998) analyse (1.1) using convex programming methods and derive the dual problem for (1.1). Their development includes the case where $\mathbf{X}$ has more columns than rows for which they show that there exist solutions of (1.1) with at most n non-zero entries. They also show that for every solution β^* of (1.1) the following equation holds:

$$\mathbf{0} = -\mathbf{X}^T\mathbf{r}^* + \lambda\mathbf{v}, \tag{1.2}$$

where $\mathbf{r}^* = \mathbf{r}(\beta^*) = \mathbf{y} - \mathbf{X}\beta^*$, $\lambda > 0$, and $\mathbf{v} = (v_1, \ldots, v_m)^T$ satisfies $v_i = 1$ if $\beta_i^* > 0$, $v_i = -1$ if $\beta_i^* < 0$ and $-1 \leq v_i \leq 1$ if $\beta_i^* = 0$.

From these results Osborne *et al.* (1998) develop an efficient algorithm to calculate the lasso estimator which is particularly well suited for the case where $\mathbf{X}$ has more rows than columns. This algorithm is described in Section 2, together with a method of choosing t automatically in the knot selection problem. The performance of these algorithms is demonstrated in several examples in Section 3. In Section 4 we discuss how this methodology can be extended to other problems. Section 5 summarises the conclusions of the paper.

2 Algorithms

2.1 Calculating the lasso estimator

Osborne *et al.* (1998) propose an iterative algorithm that is based on a local linearisation of (1.1a) about the current iterate β. At any stage in the algorithm, let σ denote the set of indices of the nonzero components of β_i; that is, $\sigma = \{i : \beta_i \neq 0\}$. Let P be the permutation matrix that collects the nonzero components of β in the first $|\sigma|$ positions,

The third author wishes to acknowledge that this work was carried out within the Cooperative Research Centre for Advanced Computational Systems established under the Australian Government's Cooperative Research Centres Program.

i.e., $\beta = P^T \begin{pmatrix} \beta_\sigma \\ \mathbf{0} \end{pmatrix}$. Let θ_σ have entry 1 if the corresponding entry in β_σ is non-negative and -1 otherwise. At any step of the algorithm β_σ has to be feasible, i.e., $\|\beta\|_1 = \|\beta_\sigma\|_1 = \theta_\sigma^T \beta_\sigma \leq t$.

To obtain the next iterate from the current β, the following optimisation problem is solved:

$$\underset{\mathbf{h}}{\text{minimise}} \quad f(\beta + \mathbf{h}) \tag{2.1a}$$

$$\text{subject to} \quad \theta_\sigma^T(\beta_\sigma + \mathbf{h}_\sigma) \leq t \quad \text{and} \quad \mathbf{h} = P^T \begin{pmatrix} \mathbf{h}_\sigma \\ \mathbf{0} \end{pmatrix}. \tag{2.1b}$$

Let $\tilde{\beta} = \beta + \mathbf{h}$ be the solution of (2.1). If $\text{sign}(\tilde{\beta}_\sigma) = \theta_\sigma$ then $\tilde{\beta}$ is called *sign feasible*. If $\tilde{\beta}$ is not sign feasible, the current solution is updated as follows:

1. Move to the first new zero component in direction $\mathbf{h}$, i.e., find the smallest γ, $0 < \gamma < 1$ for which there exists some $k \in \sigma$ such that $0 = \beta_k + \gamma h_k$.

2. There are two possibilities: (1) setting $\theta_k = -\theta_k$, $\beta = \beta + \gamma\mathbf{h}$ and recomputing $\mathbf{h}$ by again solving (2.1) yields a descent direction that is consistent with the revised θ_σ, or, (2) update σ by deleting k, setting $\beta = \beta + \gamma\mathbf{h}$, resetting β_σ and θ_σ accordingly (they are still both feasible) and recomputing $\mathbf{h}$ by solving (2.1) again.

3. Iterate until a sign feasible $\tilde{\beta}$ is obtained.

If $\tilde{\beta}$ is sign feasible, then it can be tested for optimality by verifying (1.2). Calculate

$$\tilde{\mathbf{v}} = \mathbf{X}^T\tilde{\mathbf{r}}/\|\mathbf{X}^T\tilde{\mathbf{r}}\|_\infty = P^T \begin{pmatrix} \tilde{\mathbf{v}}_1 \\ \tilde{\mathbf{v}}_2 \end{pmatrix}.$$

If $|(\tilde{\mathbf{v}}_1)_i| = |\theta_i|$ for $i \in \sigma$ and $-1 \leq (\tilde{\mathbf{v}}_2)_i \leq 1$ for $i \notin \sigma$, then $\tilde{\beta}$ is a solution of (1.1). Otherwise, we proceed as follows:

1. Determine the most violated condition, i.e., find s such that $(\tilde{\mathbf{v}}_2)_s$ has maximal absolute value.

2. Update σ by adding s to it. β_σ is updated by appending a zero as last element and append $\text{sign}(\tilde{\mathbf{v}}_2)_s$ to θ_σ.

3. Solve (2.1) and iterate.

Remark 2.1. Solving (2.1): *Problem* (2.1) *is readily solved by calculating a QR factorisation of* $\mathbf{X}_\sigma$. *This factorisation can easily be updated and downdated whenever* σ *changes. If* $\mathbf{X}_\sigma$ *is augmented with the column* $\mathbf{y}$, *then the factorisation also yields the result of an* unconstrained *fit using the current set of knots (see Section 2.2).*

Remark 2.2. Starting the iteration: *The iteration can be started from* $\beta = \mathbf{0}$ *and* $\sigma = \emptyset$. *Starting from this end of the problem has two advantages:*

- *It builds up the optimal* σ *starting from a small base rather than by pruning a large one which would be ill-conditioned;*

- *It permits the computation to proceed while at the same time building up the factorisations mentioned in Remark 2.1.*

If the lasso estimate is to be calculated for several values of t, *e.g. for an automatic choice of* t *as discussed in Section 2.2, then we can start for the smallest value of* t *with* $\beta = \mathbf{0}$ *and* $\sigma = \emptyset$. *For all further values of* t, *we take as starting point the solution for the previous smaller value.*

2.2 Automatic Choice of t

When using the lasso methodology to select knots for a regression spline, it is not clear that the constraint on the l^1 norm of the parameters should necessarily be retained once the knots have been determined. As demonstrated by the examples of the next section, the unconstrained regression spline using these knots may yield a much better fit to the data, particularly when the number of knots selected is relatively small. In addition, by using an implementation of the lasso algorithm based on a QR factorisation of $(\mathbf{X}_\sigma\ \mathbf{y})$, the unconstrained fit and its residual sum of squares are readily available.

This suggests that the lasso parameter t can be chosen in a convenient and automatic way by minimising a criterion, such as AIC, BIC, or FPE (Hocking, 1977; Eubank, 1988; Miller, 1990; Hjorth, 1994; Venables and Ripley, 1994), which penalises the residual sum of squares of the unconstrained fit according the number of knots retained, i.e., the number of nonzero β_i. Since the residual sum of squares of the unconstrained fit is immediately available from the QR decomposition, very little work is required beyond computing the lasso estimator for various values of t.

In our implementation of this approach, we start with $t = 0$ and use the algorithm described in Press *et al.* (1992) to bracket a minimum. Once a minimum is bracketed, we use Brent's algorithm (Brent, 1973; Press *et al.*, 1992) to find the minimum. This method is demonstrated in several examples in the next section.

Remark 2.3. *In our numerical work, we have observed that the lasso frequently chooses knots in close*

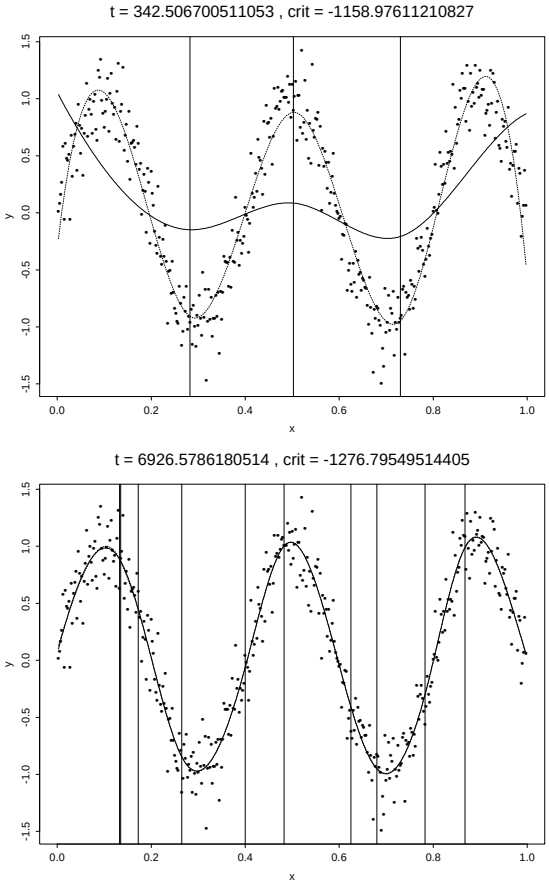


Figure 3.1: This figure shows the effect of the minimum bracketing routine. The solid line is the constrained fit, the dotted line is the unconstrained fit and knot locations are indicated by vertical lines.

proximity to one another. This problem can be alleviated by using the lasso "only" to select an initial set of knots from which the final set is chosen by best subset selection, backward deletion *or similar methods, again based on the residual sum of squares of the unconstrained fit.*

3 Numerical Properties

The minimum bracketing routine described in Press *et al.* (1992) depends on tuning parameters that have to be chosen by the user. Figure 3.1 shows that these parameters can have significant influence on the region in which a minimum is bracketed if t is chosen automatically as outlined in Section 2.2. Thus they also influence the final outcome of the procedure. The top panel in Figure 3.1 also demonstrates that the unconstrained fit may be excellent although the constrained fit approximates the data poorly.

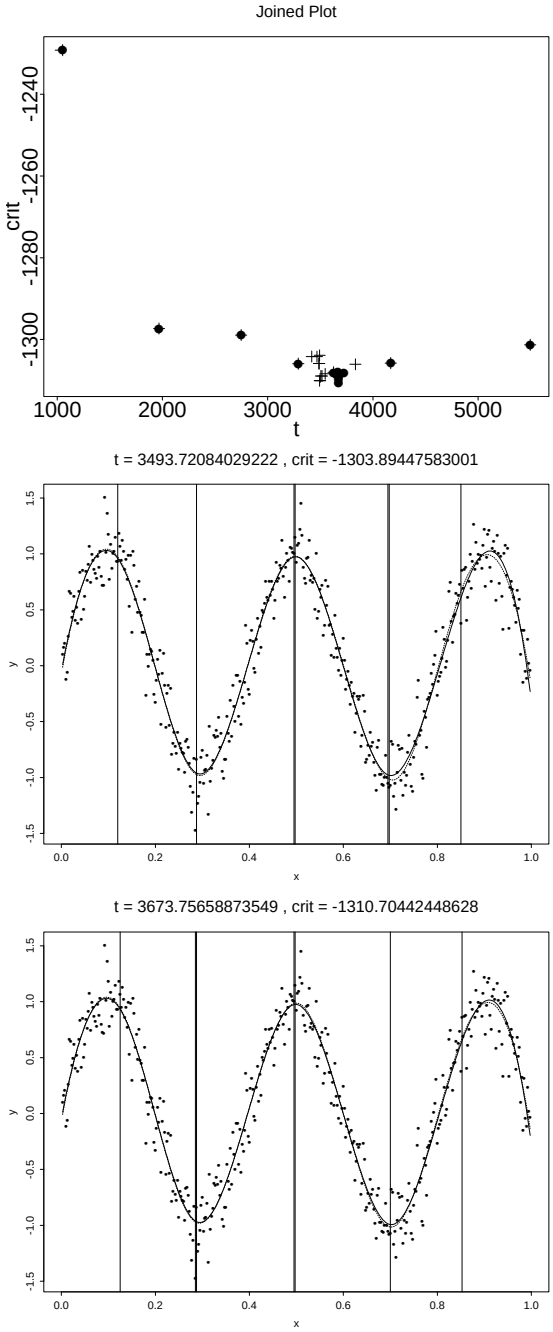
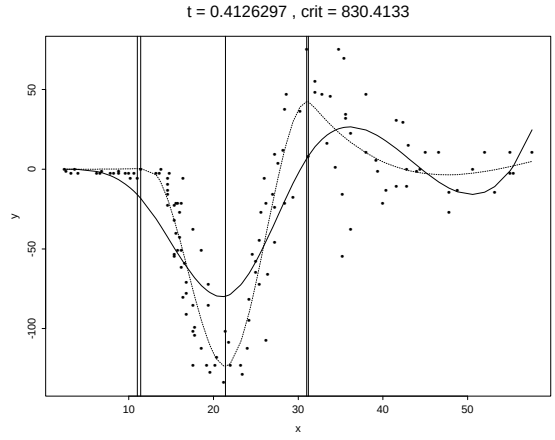
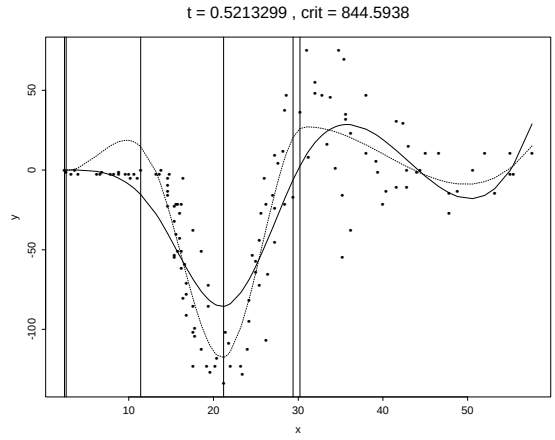


Figure 3.2: This figure demonstrates the roughness of the criterion function that we try to minimise.

The bracketing of a minimum is, however, only part of the problem, since it appears that the criterion function to be minimised is typically very rough. In connection with this, we note also that there is an error in the description of the minimisation algorithm in Brent (1973). These points are illustrated in Figure 3.2. The middle panel of the figure shows





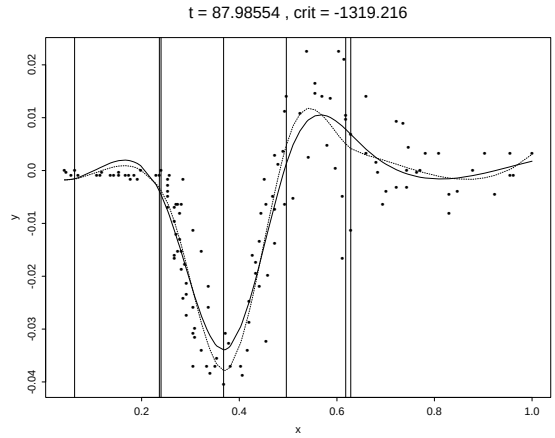
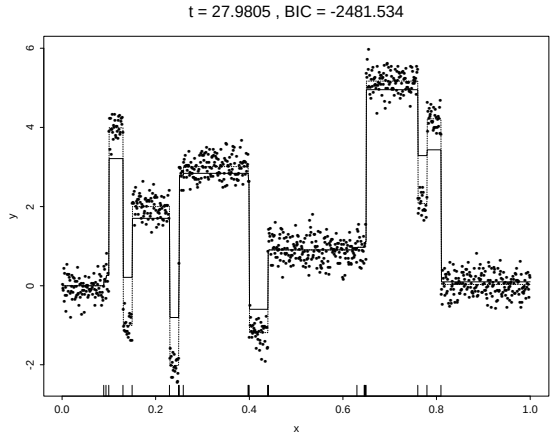
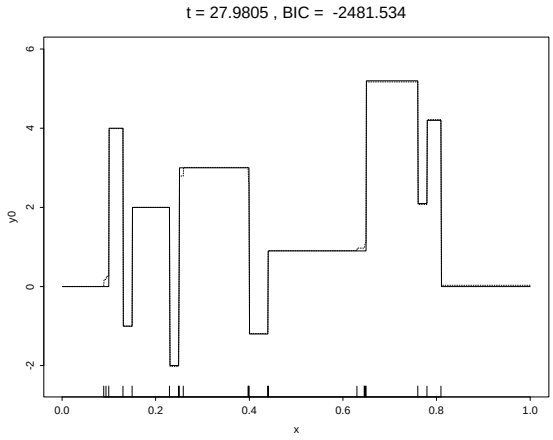


Figure 3.3: Example of a fit using the motorcycle data (Silverman, 1985). The first two figures show the effect of the tuning parameters that have to be chosen in the minimum bracketing routine. The third figure shows the influence scaling has on the method. Vertical lines indicate knot locations. The solid line is the constrained fit, the dotted line the unconstrained fit. Here, AIC was used.





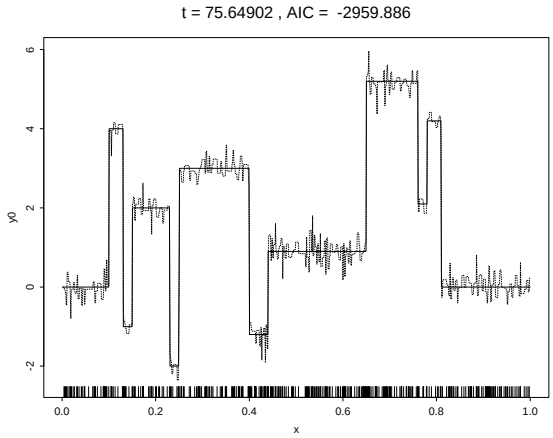


Figure 3.4: Example using the "blocks" function (Donoho and Johnstone, 1994, 1995) and $p = 0$. The first figure shows the simulated data ($n = 1024$) as dots, the solid line is the constrained fit and the dotted line is the unconstrained fit. The second figure shows the true curve together with the unconstrained fit. The "rug" shows knot locations. t was chosen automatically using BIC. The third figure shows the resulting fit if AIC is used.

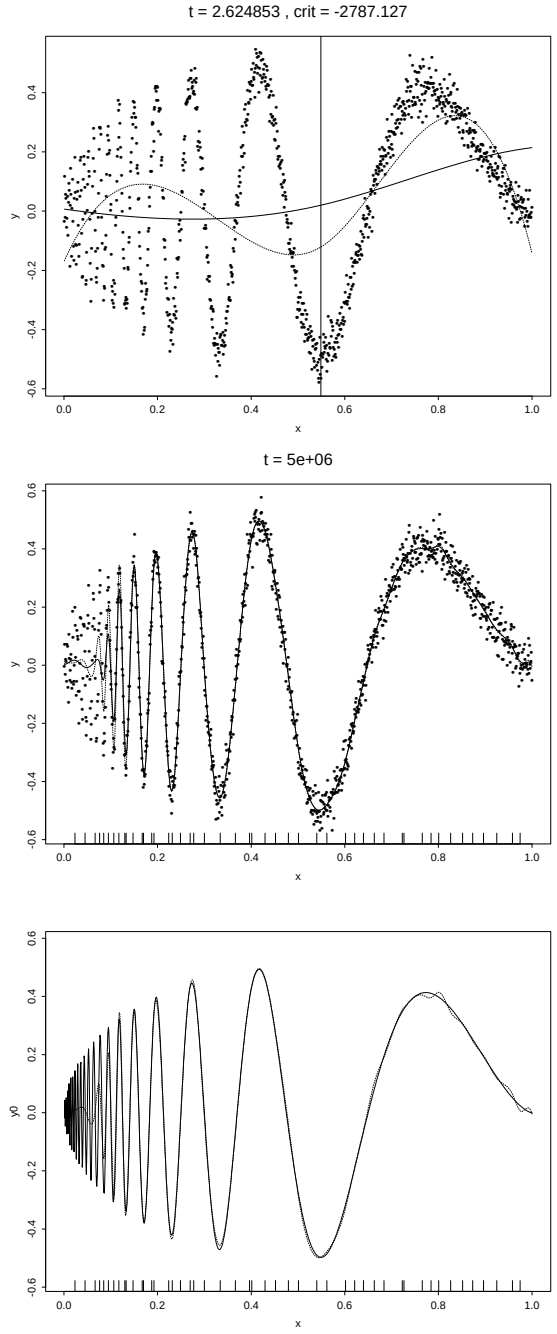


Figure 3.5: Example using the "doppler" function (Donoho and Johnstone, 1994, 1995). The first figure shows the result when t is chosen automatically (using AIC). For the second figure we used $t = 5 \times 10^6$. The simulated data ($n = 1024$) is shown as dots, the solid line is the constrained fit and the dotted line is the unconstrained fit. The third figure shows the true curve together with the unconstrained fit. The "rug" shows knot locations.

the result of the minimisation process if Brent's algorithm is used without correcting the error mentioned above, while the bottom panel shows the result obtained using the corrected algorithm. The top panel shows the values of t considered during these two minimisation processes and the corresponding criterion values (AIC in this example); dots denote the values calculated by the corrected algorithm and crosses those evaluated by the uncorrected version. The roughness of the criterion as a function of t is evident here.

Figure 3.3 shows again that selecting the "correct" tuning parameters for the minimum bracketing routine can be important. Here we applied the algorithm of Section 2.2 to the "motorcycle data" (Silverman, 1985). The parameters in the minimum bracketing routine were changed between the top panel and the middle panel. The bottom panel uses the same parameters for the minimum bracketing routine as the top panel but the data was rescaled so that the observed x_i lie between zero and one. This demonstrates that the result is also sensitive to the scaling of the data.

In the above examples, the order of the regression spline used is $p = 3$. There is some evidence that if the function is piecewise linear and $p = 0$ is used, then our current approach of choosing t automatically may overfit rather seriously, i.e., far too many knots are selected. This was the case if AIC was used as the criterion for selecting t in estimating the "blocks" function of Donoho and Johnstone (1994, 1995). Using BIC in this example on the other hand seemed to work well. This is illustrated in Figure 3.4.

Finally, the "Doppler" example (Donoho and Johnstone, 1994, 1995) demonstrates that with our current implementation, the automatic selection of t can fail by bracketing a minimum "too early". This is shown in the top panel of Figure 3.5. However, if t is chosen by the user, then reasonable fits can be obtained.

4 Further Development

As the above examples indicate more work on automatic selection of t is necessary, especially if we assume that the function is piecewise linear ($p = 0$) or has a high spatial variance (doppler function).

An obvious extension for this methodology is the generalisation to a multivariate setting. Here the question is how to choose a suitable set of basis functions. But also for the univariate case we intend to investigate how the methodology presented in this paper works with other basis functions, e.g. B-splines with varying location and scale (Bakin

et al., 1997).

To allow for different orders p at the same time (Denison *et al.*, 1998) is also an interesting challenge. The truncated power basis with $p = 0$ has a different scale than the truncated power basis with $p = 3$ and Figure 3.3 shows that the lasso approach to knot selection is sensitive to scaling. In this situation it is probably necessary to group all parameters belonging to truncated power basis functions with the same power together and use separate l^1 penalties for each group.

Finally a generalisation to additive models

$$\mathrm{E}[Y|\mathbf{X} = \mathbf{x}] = \sum_j f_j(x_j)$$

should be possible. Each f_j could be estimated by a regression spline. Again the question arises whether one single l^1 penalty for all basis function should be used (scaling problems?) or a separate penalty for each f_j.

5 Conclusions

The least absolute shrinkage and selection operator seems to be a powerful subset (variable) selection method. In the context of knot selection for regression splines it cannot be used "naively" as it often selects knots in close proximity. However, there is evidence that it is an excellent tool to choose an initial set of knots that should be considered. From this set the final set of knots may then be determined by best subset selection, backward elimination or similar techniques.

References

Bakin, S., Hegland, M. and Osborne, M.R. (1997). Can MARS be improved with B-splines?, *in* B.J. Noye, M.D. Teubner and A.W. Gill (eds), *Computational Techniques and Applications: CTAC97*, World Scientific, Singapore, pp. 75–82.

Brent, R.P. (1973). *Algorithms for Minimization without Derivatives*, Series in Automatic Computation, Prentice-Hall, Englewood Cliffs, New Jersey.

Denison, D.G.T., Mallick, B.K. and Smith, A.F.M. (1998). Automatic bayesian curve fitting, *Journal of the Royal Statistical Society, Series B* **60**(2): 333–350.

Donoho, D.L. and Johnstone, I.M. (1994). Ideal spatial adaptation via wavelet shrinkage, *Biometrika* **81**(3): 425–455.

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.

Eubank, R.L. (1988). *Smoothing Splines and Nonparametric Regression*, Marcel Dekker, New York and Basel.

Hjorth, J.S.U. (1994). *Computer Intensive Statistical Methods: Validation, Model Selection and Bootstrap*, Chapman and Hall, London.

Hocking, R.R. (1977). Selection of the best subset of regression variables, *in* K. Enslein, A. Ralston and H.S. Wilf (eds), *Statistical Methods for Digital Computers*, Vol. 3 of *Mathematical Methods for Digital computers*, John Wiley & Sons, New York, chapter 3, pp. 39–57.

Miller, A.J. (1990). *Subset Selection in Regression*, Vol. 40 of *Monographs on Statistics and Applied Probability*, Chapman and Hall, London.

Osborne, M.R., Presnell, B. and Turlach, B.A. (1998). A note on the least absolute shrinkage and selection operator. Unpublished manuscript.

Press, W., Flannery, B., Teukolsky, S. and Vetterling, W. (1992). *Numerical Recipes in C: The Art of Scientific Computing*, 2 edn, Cambridge University Press, Cambridge.

Silverman, B.W. (1985). Some aspects of the spline smoothing approach to non-parametric regression curve fitting (with discussion), *Journal of the Royal Statistical Society, Series B* **47**(1): 1–50.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso, *Journal of the Royal Statistical Society, Series B* **58**(1): 267–288.

Venables, W.N. and Ripley, B.D. (1994). *Modern Applied Statistics with S-Plus*, Statistics and Computing, 1 edn, Springer-Verlag, New York.

Wand, M.P. (1997). A comparison of regression spline smoothing procedures. Unpublished manuscript.
URL: *http://www.biostat.harvard.edu/~mwand/-rsppapps.gz*