Rodeo: Sparse Nonparametric Regression in High Dimensions

John Lafferty
School of Computer Science
Carnegie Mellon University

Larry Wasserman
Department of Statistics
Carnegie Mellon University

Abstract

We present a method for nonparametric regression that performs bandwidth selection and variable selection simultaneously. The approach is based on the technique of incrementally decreasing the bandwidth in directions where the gradient of the estimator with respect to bandwidth is large. When the unknown function satisfies a sparsity condition, our approach avoids the curse of dimensionality, achieving the optimal minimax rate of convergence, up to logarithmic factors, as if the relevant variables were known in advance. The method—called *rodeo* (regularization of derivative expectation operator)—conducts a sequence of hypothesis tests, and is easy to implement. A modified version that replaces hard with soft thresholding effectively solves a sequence of lasso problems.

1 Introduction

Estimating a high dimensional regression function is notoriously difficult due to the "curse of dimensionality." Minimax theory precisely characterizes the curse. Let $Y_i = m(X_i) + \epsilon_i$, $i = 1, \ldots, n$ where $X_i = (X_i(1), \ldots, X_i(d)) \in \mathbb{R}^d$ is a d-dimensional covariate, $m : \mathbb{R}^d \rightarrow \mathbb{R}$ is the unknown function to estimate, and $\epsilon_i \sim N(0, \sigma^2)$. Then if m is in $W_2(c)$, the d-dimensional Sobolev ball of order two and radius c, it is well known that

$$\liminf_{n \to \infty} n^{4/(4+d)} \inf_{\widehat{m}_n} \sup_{m \in W_2(c)} \mathcal{R}(\widehat{m}_n, m) > 0, \tag{1}$$

where $\mathcal{R}(\widehat{m}_n, m) = \mathbb{E}_m \int (\widehat{m}_n(x) - m(x))^2 \, dx$ is the risk of the estimate $\widehat{m}_n$ constructed on a sample of size n (Györfi et al. 2002). Thus, the best rate of convergence is $n^{-4/(4+d)}$, which is impractically slow if d is large.

However, for some applications it is reasonable to expect that the true function only depends on a small number of the total covariates. Suppose that m satisfies such a sparseness condition, so that $m(x) = m(x_R)$ where $x_R = (x_j : j \in R)$, $R \subset \{1, \ldots, d\}$ is a subset of the d covariates, of size $r = |R| \ll d$. We call $\{x_j\}_{j \in R}$ the *relevant variables*. Under this sparseness assumption we can hope to achieve the better minimax convergence rate of $n^{-4/(4+r)}$ if the r relevant variables can be isolated. Thus, we are faced with the problem of variable selection in nonparametric regression.

A large body of previous work has addressed this fundamental problem, which has led to a variety of methods to combat the curse of dimensionality. Many of these are based

on very clever, though often heuristic techniques. For additive models of the form $f(x) = \sum_j f_j(x_j)$, standard methods like stepwise selection, C_p and AIC can be used (Hastie et al. 2001). For spline models, Zhang et al. (2005) use likelihood basis pursuit, essentially the lasso adapted to the spline setting. CART (Breiman et al. 1984) and MARS (Friedman 1991) effectively perform variable selection as part of their function fitting. More recently, Li et al. (2005) use independence testing for variable selection and Bühlmann and Yu (2005) introduced a boosting approach. While these methods have met with varying degrees of empirical success, they can be challenging to implement and demanding computationally. Moreover, these methods are typically difficult to analyze theoretically, and so often come with no formal guarantees. Indeed, the theoretical analysis of sparse *parametric* estimators such as the lasso (Tibshirani 1996) is difficult, and only recently has significant progress been made on this front (Donoho 2004; Fu and Knight 2000).

In this paper we present a new approach to sparse nonparametric function estimation that is both computationally simple and amenable to theoretical analysis. We call the general framework *rodeo*, for regularization of derivative expectation operator. It is based on the idea that bandwidth and variable selection can be simultaneously performed by computing the infinitesimal change in a nonparametric estimator as a function of the smoothing parameters, and then thresholding these derivatives to effectively get a sparse estimate. As a simple version of this principle we use hard thresholding, effectively carrying out a sequence of hypothesis tests. A modified version that replaces testing with soft thresholding effectively solves a sequence of lasso problems. The potential appeal of this approach is that it can be based on relatively simple and theoretically well understood nonparametric techniques such as local linear smoothing, leading to methods that are simple to implement and can be used in high dimensional problems. Moreover, we show that the rodeo can achieve near optimal minimax rates of convergence, and therefore circumvents the curse of dimensionality when the true function is indeed sparse. When applied in one dimension, our method yields a locally optimal bandwidth. We present experiments on both synthetic and real data that demonstrate the effectiveness of the new approach.

2 Rodeo: The Main Idea

The key idea in our approach is as follows. Fix a point x and let $\widehat{m}_h(x)$ denote an estimator of $m(x)$ based on a vector of smoothing parameters $h = (h_1, \ldots, h_d)$. If c is a scalar, then we write $h = c$ to mean $h = (c, \ldots, c)$. Let $M(h) = \mathbb{E}(\widehat{m}_h(x))$ denote the mean of $\widehat{m}_h(x)$. For now, assume that x_i is one of the observed data points and that $\widehat{m}_0(x) = Y_i$. In that case, $m(x) = M(0) = \mathbb{E}(Y_i)$. If $P = (h(t) : 0 \leq t \leq 1)$ is a smooth path through the set of smoothing parameters with $h(0) = 0$ and $h(1) = 1$ (or any other fixed, large bandwidth) then

$$m(x) = M(0) = M(1) - \int_0^1 \frac{dM(h(s))}{ds} \, ds = M(1) - \int_0^1 \langle D(s), \dot{h}(s) \rangle ds$$

where $D(h) = \nabla M(h) = \left(\frac{\partial M}{\partial h_j}, \ldots, \frac{\partial M}{\partial h_j} \right)^T$ is the gradient of $M(h)$ and $\dot{h}(s) = \frac{dh(s)}{ds}$ is the derivative of $h(s)$ along the path. A biased, low variance estimator of $M(1)$ is $\widehat{m}_1(x)$. An unbiased estimator of $D(h)$ is

$$Z(h) = \left(\frac{\partial \widehat{m}_h(x)}{\partial h_1}, \ldots, \frac{\partial \widehat{m}_h(x)}{\partial h_d} \right)^T. \tag{2}$$

The naive estimator

$$\widehat{m}(x) = \widehat{m}_1(x) - \int_0^1 \langle Z(s), \dot{h}(s) \rangle ds \tag{3}$$
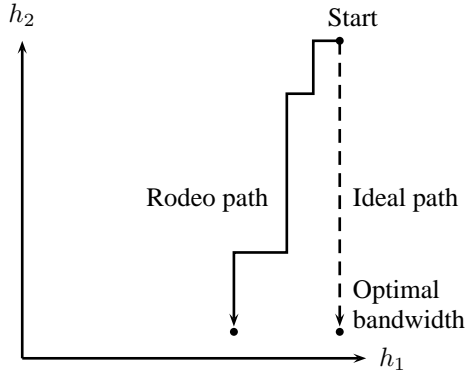
Figure 1: The bandwidths for the relevant variables (h_2) are shrunk, while the bandwidths for the irrelevant variables (h_1) are kept relatively large. The simplest rodeo algorithm shrinks the bandwidths in discrete steps $1, \beta, \beta^2, \ldots$ for some $0 < \beta < 1$.

is identically equal to $\widehat{m}_0(x) = Y_i$, which has poor risk since the variance of $Z(h)$ is large for small h. However, our sparsity assumption on m suggests that there should be paths for which $D(h)$ is also sparse. Along such a path, we replace $Z(h)$ with an estimator $\widehat{D}(h)$ that makes use of the sparsity assumption. Our estimate of $m(x)$ is then

$$\widetilde{m}(x) = \widehat{m}_1(x) - \int_0^1 \big\langle \widehat{D}(s), \dot{h}(s) \big\rangle ds \,. \tag{4}$$

To implement this idea we need to do two things: (i) we need to find a sparse path and (ii) we need to take advantage of this sparseness when estimating D along that path.

The key observation is that if x_j is irrelevant, then we expect that changing the bandwidth h_j for that variable should cause only a small change in the estimator $\widehat{m}_h(x)$. Conversely, if x_j is relevant, then we expect that changing the bandwidth h_j for that variable should cause a large change in the estimator. Thus, $Z_j = \partial \widehat{m}_h(x)/\partial h_j$ should discriminate between relevant and irrelevant covariates. To simplify the procedure, we can replace the continuum of bandwidths with a discrete set where each $h_j \in \mathcal{B} = \{h_0, \beta h_0, \beta^2 h_0, \ldots\}$ for some $0 < \beta < 1$. Moreover, we can proceed in a greedy fashion by estimating $D(h)$ sequentially with $h_j \in \mathcal{B}$ and setting $\widehat{D}_j(h) = 0$ when $h_j < \widehat{h}_j$, where $\widehat{h}_j$ is the first h such that $|Z_j(h)| < \lambda_j(h)$ for some threshold λ_j. This greedy version, coupled with the hard threshold estimator, yields $\widetilde{m}(x) = \widehat{m}_{\widehat{h}}(x)$. A conceptual illustration of the idea is shown in Figure 1. This idea can be implemented using a greedy algorithm, coupled with the hard threshold estimator, to yield a bandwidth selection procedure based on testing.

This approach to bandwidth selection is similar to that of Lepski et al. (1997), which uses a more refined test leads to estimators that achieve good spatial adaptation over large function classes. Our approach is also similar to a method of Ruppert (1997) that uses a sequence of decreasing bandwidths and then estimates the optimal bandwidth by estimating the mean squared error as a function of bandwidth. Our greedy approach tests whether an infinitesimal change in the bandwidth from its current setting leads to a significant change in the estimate, and is more easily extended to a practical method in higher dimensions. Related work of Hristache et al. (2001) focuses on variable selection in multi-index models rather than on bandwidth estimation.

3 Rodeo using Local Linear Regression

We now present the multivariate case in detail, using local linear smoothing as the basic method since it is known to have many good properties. Let $x = (x(1), \ldots, x(d))$ be some point at which we want to estimate m. Let $\widehat{m}_H(x)$ denote the local linear estimator of

$m(x)$ using bandwidth matrix H. Thus,

$$\widehat{m}_H(x) = e_1^T (X_x^T W_x X_x)^{-1} X_x^T W_x Y, \qquad X_x = \begin{pmatrix} 1 & (X_1 - x)^T \\ \vdots & \vdots \\ 1 & (X_n - x)^T \end{pmatrix} \tag{5}$$

where $e_1 = (1, 0, \ldots, 0)^T$, and W_x is the diagonal matrix with (i, i) element $K_H(X_i - x)$ and $K_H(u) = |H|^{-1} K(H^{-1} u)$. The estimator $\widehat{m}_H$ can be written as $\widehat{m}_H(x) = \sum_{i=1}^n G(X_i, x, h) Y_i$ where

$$G(u, x, h) = e_1^T (X_x^T W_x X_x)^{-1} \begin{pmatrix} 1 \\ (u - x)^T \end{pmatrix} K_H(u - x) \tag{6}$$

is called the *effective kernel*. We assume that the covariates are random with sampling density $f(x)$, and make the same assumptions as Ruppert and Wand (1994) in their analysis of the bias and variance of local linear regression. In particular, (i) the kernel K has compact support with zero odd moments and $\int u u^\top K(u)\, du = \nu_2(K) I$ and (ii) the sampling density $f(x)$ is continuously differentiable and strictly positive. In the version of the algorithm that follows, we take K to be a product kernel and H to be diagonal with elements $h = (h_1, \ldots, h_d)$.

Our method is based on the statistic

$$Z_j = \frac{\partial \widehat{m}_h(x)}{\partial h_j} = \sum_{i=1}^n G_j(X_i, x, h) Y_i \tag{7}$$

where $G_j(u, x, h) = \frac{\partial G(u, x, h)}{\partial h_j}$. Straightforward calculations show that

$$Z_j = \frac{\partial \widehat{m}_h(x)}{\partial h_j} = = e_1^\top (X_x^\top W_x X_x)^{-1} X_x^\top \frac{\partial W_x}{\partial h_j} (Y - X_x \widehat{\alpha}) \tag{8}$$

where $\widehat{\alpha} = (X_x^\top W_x X_x)^{-1} X_x^\top W_x Y$ is the coefficient vector for the local linear fit. Note that the factor $|H|^{-1} = \prod_{i=1}^d 1/h_i$ in the kernel cancels in the expression for $\widehat{m}$, and therefore we can ignore it in our calculation of Z_j. Assuming a product kernel we have

$$W_x = \text{diag}\left(\prod_{j=1}^d K((X_{1j} - x_j)/h_j), \ldots, \prod_{j=1}^d K((X_{nj} - x_j)/h_j) \right) \tag{9}$$

and $\partial W_x / \partial h_j = W_x D_j$ where

$$D_j = \text{diag}\left(\frac{\partial \log K((X_{1j} - x_j)/h_j)}{\partial h_j}, \ldots, \frac{\partial \log K((X_{nj} - x_j)/h_j)}{\partial h_j} \right) \tag{10}$$

and thus $Z_j = e_1^\top (X_x^\top W_x X_x)^{-1} X_x^\top W_x D_j (Y - X_x \widehat{\alpha})$. For example, with the Gaussian kernel $K(u) = \exp(-u^2/2)$ we have $D_j = \frac{1}{h_j^3} \text{diag}\left((X_{1j} - x_j)^2, \ldots, (X_{nj} - x_j)^2 \right)$.

Let

$$\mu_j \equiv \mu_j(h) = \mathbb{E}(Z_j | X_1, \ldots, X_n) = \sum_{i=1}^n G_j(X_i, x, h) m(X_i) \tag{11}$$

$$s_j^2 \equiv s_j^2(h) = \mathbb{V}(Z_j | X_1, \ldots, X_n) = \sigma^2 \sum_{i=1}^n G_j(X_i, x, h)^2. \tag{12}$$

Then the hard thresholding version of the rodeo algorithm is given in Figure 2.

The algorithm requires that we insert an estimate $\widehat{\sigma}$ of σ in (12). One estimate of σ can be obtained by generalizing a method of Rice (1984). For $i < \ell$, let $d_{i\ell} = \|X_i - X_\ell\|$. Fix an integer J and let $\mathcal{E}$ denote the set of pairs (i, ℓ) corresponding to the J smallest values of $d_{i\ell}$. Now define $\widehat{\sigma}^2 = \frac{1}{2J} \sum_{i,\ell \in \mathcal{E}} (Y_i - Y_\ell)^2$. Then $\mathbb{E}(\widehat{\sigma}^2) = \sigma^2 + \text{bias}$ where

Rodeo: Hard thresholding version

1. *Select* parameter $0 < \beta < 1$ and initial bandwidth h_0 slowly decreasing to zero, with $h_0 = \Omega \left(1/\sqrt{\log \log n} \right)$. Let $c_n = \Omega(1)$ be a sequence satisfying $dc_n = \Omega(\log n)$.

2. *Initialize* the bandwidths, and activate all covariates:
   (a) $h_j = h_0, j = 1, 2, \ldots, d$.
   (b) $\mathcal{A} = \{1, 2, \ldots, d\}$

3. *While $\mathcal{A}$ is nonempty*, do for each $j \in \mathcal{A}$:
   (a) Compute the estimated derivative expectation: Z_j (equation 7) and s_j (equation 12).
   (b) Compute the threshold $\lambda_j = s_j \sqrt{2 \log(dc_n)}$.
   (c) If $|Z_j| \geq \lambda_j$, then set $h_j \leftarrow \beta h_j$, otherwise remove j from $\mathcal{A}$.

4. *Output* bandwidths $h^\star = (h_1, \ldots, h_d)$ and estimator $\widetilde{m}(x) = \widehat{m}_{h^\star}(x)$.

Figure 2: The hard thresholding version of the rodeo, which can be applied using the derivatives Z_j of any nonparametric smoother.

bias $\leq D \sup_x \sum_{j \in R} \left| \frac{\partial f(x)}{\partial x_j} \right|$ with $D = \max_{i,\ell \in \mathcal{E}} \|X_i - X_\ell\|$. There is a bias-variance tradeoff: large J makes $\widehat{\sigma}^2$ positively biased, and small J makes $\widehat{\sigma}^2$ highly variable. Note however that the bias is mitigated by sparsity (small r). This is the estimator used in our examples.

4 Analysis

In this section we present some results on the properties of the resulting estimator. Formally, we use a triangular array approach so that $f(x)$, $m(x)$, d and r can all change as n changes. For convenience of notation we assume that the covariates are numbered such that the relevant variables x_j correspond to $1 \leq j \leq r$, and the irrelevant variables to $j > r$. To begin, we state the following technical lemmas on the mean and variance of Z_j.

Lemma 4.1. *Suppose that K is a product kernel with bandwidth vector $h = (h_1, \ldots, h_d)$. If the sampling density f is uniform, then $\mu_j = 0$ for all $j \in R^c$. More generally, assuming that r is bounded, we have the following when $h_j \to 0$: If $j \in R^c$ the derivative of the bias is*

$$\mu_j = \frac{\partial}{\partial h_j} \mathbb{E}[\widehat{m}_H(x) - m(x)] = -\text{tr}\left(H_R \mathcal{H}_R \right) \nu_2^2 \left(\nabla_j \log f(x) \right)^2 h_j + o_P(h_j) \quad (13)$$

where the Hessian of $m(x)$ is $\mathcal{H} = \begin{pmatrix} \mathcal{H}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}$ and $H_R = \text{diag}(h_1^2, \ldots, h_r^2)$. For $j \in R$ we have

$$\mu_j = \frac{\partial}{\partial h_j} \mathbb{E}[\widehat{m}_H(x) - m(x)] = h_j \nu_2 m_{jj}(x) + o_P(h_j). \quad (14)$$

Lemma 4.2. *Let $C = \left(\frac{\sigma^2 R(K)}{4m(x)} \right)$ where $R(K) = \int K(u)^2 \, du$. Then, if $h_j = o(1)$,*

$$s_j^2 = \mathbb{V}\text{ar}(Z_j | X_1, \ldots, X_n) = \frac{C}{nh_j^2} \left(\prod_{k=1}^{d} \frac{1}{h_k} \right) \left(1 + o_P(1) \right). \quad (15)$$

These lemmas parallel the calculations of Ruppert and Wand (1994) except for the difference that the irrelevant variables have different leading terms in the expansions than relevant variables.

Our main theoretical result characterizes the asymptotic running time, selected bandwidths, and risk of the algorithm. In order to get a practical algorithm, we need to make assumptions on the functions m and f.

(A1) For some constant $k > 0$, each $j > r$ satisfies

$$\nabla_j \log f(x) = O\left(\frac{\log^k n}{n^{1/4}}\right) \tag{16}$$

(A2) For each $j \leq r$,

$$m_{jj}(x) \neq 0 \,. \tag{17}$$

Explanation of the Assumptions. To give the intuition behind these assumptions, recall from Lemma 4.1 that

$$\mu_j = \begin{cases} A_j h_j + o_P(h_j) & j \leq r \\ B_j h_j + o_P(h_j) & j > r \end{cases} \tag{18}$$

where

$$A_j = \nu_2 m_{jj}(x), \quad B_j = -\text{tr}(H\mathcal{H})\nu_2^2(\nabla_j \log f(x))^2. \tag{19}$$

Moreover, $\mu_j = 0$ when the sampling density f is uniform or the data are on a regular grid. Consider assumption (A1). If f is uniform then this assumption is automatically satisfied since then $\mu_j(s) = 0$ for $j > r$. More generally, μ_j is approximately proportional to $(\nabla_j \log f(x))^2$ for $j > r$ which implies that $|\mu_j| \approx 0$ for irrelevant variables if f is sufficiently smooth in the variable x_j. Hence, assumption (A1) can be interpreted as requiring that f is sufficiently smooth in the irrelevant dimensions.

Now consider assumption (A2). Equation (18) ensures that μ_j is proportional to $h_j|m_{jj}(x)|$ for small h_j. Since we take the initial bandwidth h_0 to be decreasingly slowly with n, (A2) implies that $|\mu_j(h)| \geq ch_j|m_{jj}(x)|$ for some constant $c > 0$, for sufficiently large n.

In the following we write $Y_n = \widetilde{O}_P(a_n)$ to mean that $Y_n = O_P(b_n a_n)$ where b_n is logarithmic in n; similarly, $a_n = \widetilde{\Omega}(b_n)$ if $a_n = \Omega(b_n c_n)$ where c_n is logarithmic in n.

Theorem 4.3. *Suppose assumptions (A1) and (A2) hold. In addition, suppose that $d_{\min} = \min_{j \leq r} |m_{jj}(x)| = \widetilde{\Omega}(1)$ and $d_{\max} = \max_{j \leq r} |m_{jj}(x)| = \widetilde{O}(1)$. Then the number of iterations T_n until the rodeo stops satisfies*

$$\mathbb{P}\left(\frac{1}{4+r} \log_{1/\beta}(na_n) \leq T_n \leq \frac{1}{4+r} \log_{1/\beta}(nb_n)\right) \longrightarrow 1 \tag{20}$$

where $a_n = \widetilde{\Omega}(1)$ and $b_n = \widetilde{O}(1)$. Moreover, the algorithm outputs bandwidths $h^\star$ that satisfy

$$\mathbb{P}\left(h_j^\star \geq \frac{1}{\log^k n} \text{ for all } j > r\right) \longrightarrow 1 \tag{21}$$

and

$$\mathbb{P}\left(h_0(nb_n)^{-1/(4+r)} \leq h_j^\star \leq h_0(na_n)^{-1/(4+r)} \text{ for all } j \leq r\right) \longrightarrow 1 \,. \tag{22}$$

Corollary 4.4. *Under the conditions of Theorem 4.3, the risk $\mathcal{R}(h^\star)$ of the rodeo estimator satisfies*

$$\mathcal{R}(h^\star) = \widetilde{O}_P\left(n^{-4/(4+r)}\right). \tag{23}$$

In the one-dimensional case, this result shows that the algorithm recovers the locally optimal bandwidth, giving an adaptive estimator, and in general attains the optimal (up to logarithmic factors) minimax rate of convergence.

The proofs of these results are given in the full version of the paper.

5 Some Examples and Extensions

Figure 3 illustrates the rodeo on synthetic and real data. The left plot shows the bandwidths obtained on a synthetic dataset with $n = 500$ points of dimension $d = 20$. The covariates are generated as $x_i \sim \text{Uniform}(0, 1)$, the true function is $m(x) = 2(x_1+1)^2 + 2\sin(10x_2)$, and $\sigma = 1$. The results are averaged over 50 randomly generated data sets; note that the displayed bandwidth paths are not monotonic because of this averaging. The plot shows how the bandwidths of the relevant variables shrink toward zero, while the bandwidths of the irrelevant variables remain large. Simulations on other synthetic data sets, not included here, are similar and indicate that the algorithm's performance is consistent with our theoretical analysis.

The framework introduced here has many possible generalizations. While we have focused on estimation of m locally at a point x, the idea can be extended to carry out global bandwidth and variable selection by averaging over multiple evaluation points $x_1, \ldots, x_k$. These could be points interest for estimation, could be randomly chosen, or could be taken to be identical to the observed X_is. In addition, it is possible to consider more general paths, for example using soft thresholding or changing only the bandwidth corresponding to the largest $|Z_j|/\lambda_j$.

Such a version of the rodeo can be seen as a nonparametric counterpart to least angle regression (LARS) (Efron et al. 2004), a refinement of forward stagewise regression in which one adds the covariate most correlated with the residuals of the current fit, in small, incremental steps. Note first that Z_j is essentially the correlation between the Y_is and the $G_j(X_i, x, h)$s (the change in the effective kernel). Reducing the bandwidth is like adding in more of that variable. Suppose now that we make the following modifications to the rodeo: (i) change the bandwidths one at a time, based on the largest $Z_j^* = |Z_j|/\lambda_j$, (ii) reduce the bandwidth continuously, rather than in discrete steps, until the largest Z_j^* is equal to the next largest. Figure 3 (right) shows the result of running this greedy version of the rodeo on the diabetes dataset used to illustrate LARS. The algorithm averages Z_j^* over a randomly chosen set of $k = 100$ data points. The resulting variable ordering is seen to be very similar to, but different from, the ordering obtained from the parametric LARS fit.

Acknowledgments

We thank the reviewers for their helpful comments. Research supported in part by NSF grants IIS-0312814, IIS-0427206, and DMS-0104016, and NIH grants R01-CA54852-07 and MH57881.

References

L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and regression trees.* Wadsworth Publishing Co Inc, 1984.

P. Bühlmann and B. Yu. Boosting, model selection, lasso and nonnegative garrote. Technical report, Berkeley, 2005.
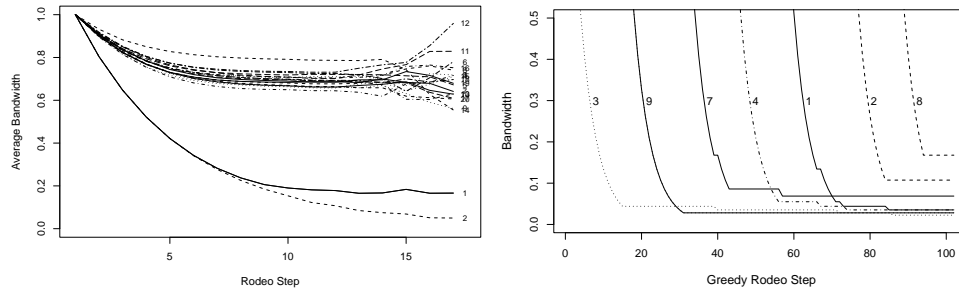
Figure 3: Left: Average bandwidth output by the rodeo for a function with $r = 2$ relevant variables in $d = 20$ dimensions ($n = 500$, with 50 trials). Covariates are generated as $x_i \sim \text{Uniform}(0, 1)$, the true function is $m(x) = 2(x_1 + 1)^3 + 2\sin(10x_2)$, and $\sigma = 1$, fit at the test point $x = (\frac{1}{2}, \dots, \frac{1}{2})$. The variance is greater for large step sizes since the rodeo runs that long for fewer data sets. Right: Greedy rodeo on the diabetes data, used to illustrate LARS (Efron et al. 2004). A set of $k = 100$ of the total $n = 442$ points were sampled ($d = 10$), and the bandwidth for the variable with largest average $|Z_j|/\lambda_j$ was reduced in each step. The variables were selected in the order 3 (body mass index), 9 (serum), 7 (serum), 4 (blood pressure), 1 (age), 2 (sex), 8 (serum), 5 (serum), 10 (serum), 6 (serum). The parametric LARS algorithm adds variables in the order 3, 9, 4, 7, 2, 10, 5, 8, 6, 1. One notable difference is in the position of the age variable.

D. Donoho. For most large underdetermined systems of equations, the minimal ℓ^1-norm near-solution approximates the sparest near-solution. Technical report, Stanford, 2004.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of Statistics*, 32:407–499, 2004.

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

W. Fu and K. Knight. Asymptotics for lasso type estimators. *The Annals of Statistics*, 28:1356–1378, 2000.

L. Györfi, M. Kohler, A. Krzyżak, and H. Walk. *A Distribution-Free Theory of Nonparametric Regression*. Springer-Verlag, 2002.

T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, 2001.

M. Hristache, A. Juditsky, J. Polzehl, and V. Spokoiny. Structure adaptive approach for dimension reduction. *Ann. Statist.*, 29:1537–1566, 2001.

O. V. Lepski, E. Mammen, and V. G. Spokoiny. Optimal spatial adaptation to inhomogeneous smoothness: An approach based on kernel estimates with variable bandwidth selectors. *The Annals of Statistics*, 25:929–947, 1997.

L. Li, R. D. Cook, and C. Nachsteim. Model-free variable selection. *J. R. Statist. Soc. B.*, 67:285–299, 2005.

J. Rice. Bandwidth choice for nonparametric regression. *The Annals of Statistics*, 12:1215–1230, 1984.

D. Ruppert. Empirical-bias bandwidths for local polynomial nonparametric regression and density estimation. *Journal of the American Statistical Association*, 92:1049–1062, 1997.

D. Ruppert and M. P. Wand. Multivariate locally weighted least squares regression. *The Annals of Statistics*, 22:1346–1370, 1994.

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

H. Zhang, G. Wahba, Y. Lin, M. Voelker, R. K. Ferris, and B. Klein. Variable selection and model building via likelihood basis pursuit. *J. of the Amer. Stat. Assoc.*, 99(467):659–672, 2005.