

Hierarchical selection of variables in sparse high-dimensional regression

P. J. Bickel

Department of Statistics
University of California at Berkeley

Y. Ritov

Department of Statistics
The Hebrew University of Jerusalem

A.B. Tsybakov

Laboratoire de Statistique, CREST, Timbre J340
3, av.Pierre Larousse, 92240 Malakoff cedex, France
and Laboratoire de Probabilités et Modèles Aléatoires
Univeristé Pierre et Marie Curie

November 29, 2007

Abstract

We study a regression model with a huge number of interacting variables. We consider a specific approximation of the regression function under two assumptions: (i) there exists a sparse representation of the regression function in a suggested basis, (ii) there are no interactions outside of the set of the corresponding main effects. We suggest an hierarchical randomized search procedure for selection of variables and of their interactions. We show that given an initial estimator, an estimator with a similar prediction loss but with a smaller number of non-zero coordinates can be found.

1 Introduction

Suppose that we observe (Y_i, \mathbf{X}_i) , $i = 1, \dots, n$, an i.i.d. sample from the joint distribution of (Y, \mathbf{X}) , where $Y \in \mathcal{R}$, and $\mathbf{X} = (X_1, \dots, X_d) \in \mathcal{X}_1 \times \dots \times \mathcal{X}_d = \mathcal{X}$, with \mathcal{X}_j being some subsets of finite-dimensional Euclidean spaces. Our purpose is to estimate the regression function $f(\mathbf{X}) = E(Y|\mathbf{X})$

nonparametrically by constructing a suitable parametric approximation of this function, with data-dependent values of the parameters. We consider the situation where n is large, or even very large and the dimension d is also large. Without any assumptions, the problem is cursed by its dimensionality even when $\mathcal{X}_j = \mathcal{R}$ for all j . For example, a histogram approximation has $p = 3^{20} > 10^9$ parameters when the number of variables is $d = 20$, and the range of each is divided into the meager number of three histogram bins.

It is common now to consider models where the number of parameters p is much larger than the sample size n . The idea is that the effective dimension is defined not by the number of potential parameters p but by the (unknown) number of non-zero parameters that can be much smaller than n . Methods like thresholding in white noise model, cf. ? or ?, LASSO, LARS or Dantzig selector in regression, cf. ?, ?, ?, ?, are used, and it is proved that if the vector of estimated parameters is sparse (i.e., the number of non-zero parameters is relatively small) then the model can be estimated with reasonable accuracy, cf. ?; ?; ?; ?; ?; ?; ?; ?; ?. A direct selection of a small number of non-zero variables is relatively simple for the white noise model. There, each variable is processed separately, and the parameters can be ordered according to the likelihood that they are non-zero. The situation is more complicated in regression problems. Methods like LASSO and LARS yield numerically efficient ways to construct a sparse model, cf. ?; ?; ?; ?; ?; ?. However, they have their limits, and are not *numerically* feasible with too many parameters, as for instance in the simple example considered above.

Our aim is to propose a procedure that can work efficiently in such situations. We now outline its general scheme. Consider a collection of functions $(\psi_{i,j})_{i=1,\dots,d,j=0,1,\dots,L}$ where $\psi_{i,j} : \mathcal{X}_i \rightarrow \mathcal{R}$. For example, for fixed i this can be a part of a basis $(\psi_{i,j})_{j=0,1,\dots,L}$ for $L_2(\mathcal{X}_i)$. For simplicity, we take the same number L of basis functions for each variable. We assume that $\psi_{i,0} \equiv 1$. Consider an approximation f_β of regression function f given by:

$$f_\beta(\mathbf{X}) = \sum_{\mathbf{j} \in \{0,1,\dots,L\}^d} \beta_{\mathbf{j}} \prod_{i=1}^d \psi_{i,j_i}(X_i)$$

where $\mathbf{j} = (j_1, \dots, j_d)$ and $\beta_{\mathbf{j}}$ are unknown coefficients. Note that f_β is nothing but a specific model with interactions between variables, such that all the interactions are expressed by products of functions of a single variable. In fact, since $\psi_{i,0} \equiv 1$, the multi-indices \mathbf{j} with only one non-zero coefficient yield all the functions of a single variable, those with only two non-zero coefficients yield all the products of two such functions, etc. Clearly, this

covers the above histogram example, wavelet approximations and others.

The number of coefficients $\beta_{\mathbf{j}}$ in the model is $(L+1)^d$. The LASSO type estimator can deal with a large number of potential coefficients which grows exponentially in n . So, theoretically, we could throw all the factors into the LASSO algorithm and find a solution. But $p \sim L^d$ is typically a huge number. Although in the theory LASSO can handle that many variables, in practice, it becomes numerically infeasible. Therefore, a systematic search is needed.

Since there is no way to know in advance which factors are significant, we suggest a hierarchical selection: we build the model in a tree fashion. At each step of the iteration we apply a LASSO type algorithm to a collection of candidate functions, where we start with all functions of a single variable. Then, from the model selected by this algorithm we extract a sub-model which includes only K functions, for some predefined K . The next step of the iteration starts with the same candidate functions as its predecessor plus all the interactions between the K functions selected at the previous step.

Formally we consider the following hierarchical model selection method. For a set of functions \mathcal{F} with cardinality $|\mathcal{F}| \geq K$, let \mathcal{MS}_K be some procedure to select K functions out of \mathcal{F} . We denote by $\mathcal{MS}_K(\mathcal{F})$ the selected subset of \mathcal{F} , $|\mathcal{MS}_K(\mathcal{F})| = K$. Also, for a function $f : \mathcal{X} \rightarrow \mathcal{R}$, let $\mathbb{N}(f)$ be the minimal set of indices such that f is a function of $(X_i)_{i \in \mathbb{N}(f)}$ only. The procedure is defined as follows.

(i) Set $\mathcal{F}_0 = \cup_{i=1}^d \{\psi_{i,1}, \dots, \psi_{i,L}\}$.

(ii) For $m = 1, 2, \dots$, let

$$\mathcal{F}_m = \mathcal{F}_{m-1} \cup \{fg : f, g \in \mathcal{MS}_K(\mathcal{F}_{m-1}), \mathbb{N}(f) \cap \mathbb{N}(g) = \emptyset\}.$$

(iii) Continue until convergence is declared. The output of the algorithm is the set of functions $\mathcal{MS}_K(\mathcal{F}_m)$ for some m .

This search procedure is valid under the dictum of no interaction outside of the set of the corresponding main effects: a term is included only if it is a function of one variable or it is a product of two other included terms. If this is not a valid assumption one can enrich the search at each step to cover all the coefficients $\beta_{\mathbf{j}}$ of the model. However, this would be cumbersome.

Note that $|\mathcal{F}_m| \leq K^2 + |\mathcal{F}_{m-1}| \leq mK^2 + |\mathcal{F}_0| = mK^2 + Ld$. Thus, the set \mathcal{F}_m is not excessively large. At every step of the procedure we keep for selection all the functions of a single variable, along with not too many interaction terms. In other words, functions of a single variable are treated

as privileged contributors. On the contrary, interactions are considered with a suspicion increasing as their multiplicity grows: they cannot be candidates for inclusion unless their “ancestors” were included at all the previous steps.

The final number of selected effects is K by construction. We should choose K to be much smaller than n if we want to fit our final model in the framework of the classical regression theory.

One can split the sample in two parts and do model selection and estimation separately. Theoretically, the rate of convergence of the LASSO type procedures suffers very little when the procedures are applied only to a sub-sample of the observations, as long as the sub-sample size n_{MS} used for model selection is such that n_{MS}/n converges slowly to 0. We can therefore, first use a sub-sample of size n_{MS} to select, according to (i)–(iii), a set of K terms that we include in the model. The second stage will use the rest of the sample and estimate via, e.g., standard least-square method the regression coefficients of the K selected terms.

This paper has two goals. The first one, as described already, is suggesting a method to build highly complex models in a hierarchical fashion. The second purpose is arguing that a reasonable way to do model selection is a two stage procedure. The first stage can be based on the LASSO, which is an efficient way to obtain sparse representation of a regression model. We argue, however, by a way of example in Section 2, that using solely the LASSO can be an non-optimal procedure for model selection. Therefore, in Section 3 we introduce the second stage of selection, such that a model of a desired size is obtained at the end. At this stage we suggest to use either randomized methods or the standard backward procedure. We prove prediction error bounds for two randomized methods of pruning the result of the LASSO stage. Finally, in Section 4 we consider two examples that combine the ideas presented in this paper.

2 Model selection: an example

The above hierarchical method depends on a model selection procedure \mathcal{MS}_K that we need to determine. For high-dimensional case that we are dealing with, LASSO is known to be an efficient model selection tool: it is shown that under general conditions the set of non-zero coefficients of LASSO estimator coincides with the true set of non-zero coefficients in linear regression, with probability converging to 1 as $n \rightarrow \infty$ (see, e.g., ?; ?). However, these results depend on strong assumptions that essentially rule off anything close to multicollinearity. These conditions are often violated in

practice when there are many variables representing a plentitude of highly related one to another demographic and physical measurements of the same subject. They are also violated in a common statistical learning setup where the variables of the analysis are values of different functions of one real variable (e.g., different step functions). Note that for our procedure we do not need to retain all the non-zero coefficients but just to extract the K “most important” ones. To achieve this, we first tried to tune the LASSO in some natural way. However, this approach failed.

We start with an example. We use this example to argue that although the LASSO does select a small model (i.e., typically many of the coordinates of the LASSO estimator are 0), it does a poor job in selecting the relevant variables. A naive approach for model selection when the constraint applies to the number of non-zero coefficients, is to relax the LASSO algorithm until it yields a solution with the right number of variables. We believe that this is a wrong approach. The LASSO is geared for L_1 constraints and not for L_0 ones. We suggest another procedure in which we run the LASSO until it yields a model more complex than wished, but not too complex, so that a standard model selection technique like backward selection can be used. This was the method considered in ? to argue that there are model selection methods which are persistent under general conditions.

We first recall the basic definition of LASSO. Consider the linear regression model

$$\mathbf{y} = \mathbf{Z}\beta_0 + \varepsilon$$

where $\mathbf{y} = (Y_1, \dots, Y_n)' \in \mathcal{R}^n$ is the vector of observed responses, $\mathbf{Z} \in \mathcal{R}^{n \times p}$ is the design matrix, $\beta_0 \in \mathcal{R}^p$ is an unknown parameter and $\varepsilon = (\xi_1, \dots, \xi_n)' \in \mathcal{R}^n$ is a noise. The LASSO estimator $\hat{\beta}_L$ of β_0 is defined as a solution of the minimization problem

$$\min_{\beta: \|\beta\|_1 \leq T} \|\mathbf{y} - \mathbf{Z}\beta\|^2 \tag{1}$$

where $T > 0$ is a tuning parameter, $\|\beta\|_1$ is the ℓ_1 -norm of β and $\|\cdot\|$ is the empirical norm associated to the sample of size n :

$$\|\mathbf{y}\|^2 = n^{-1} \sum_{i=1}^n Y_i^2.$$

This is the formulation of the LASSO as given in ?. Another formulation, given below in (8), is that of minimization of the sum of squares with L_1 penalty. Clearly, (1) is equivalent to (8) with some constant r dependent on T and on the data, by the Lagrange argument. The standard LARS-like

algorithm of ?, which is the algorithm we used, is based on gradual relaxation of the constraint T of equation (1), and solves therefore simultaneously both problems. The focus of this paper is the selection of a model of a given size. Hence we apply the LARS algorithm until we get for the first time a model of a prescribed size.

Example 2.1 We consider a linear regression model with 100 i.i.d. observations of (Y, Z_1, \dots, Z_{150}) where the predictors (Z_1, \dots, Z_{150}) are i.i.d. standard normal, the response variable is $Y = \sum_{j=1}^{150} \beta_j Z_j + \xi = \sum_{j=1}^{10} \frac{10}{25+j^2} Z_j + \xi$, and the measurement error is $\xi \sim N(0, \sigma^2)$, $\sigma = 0.1$.

Note that we have more variables than observations but most of the β_j are zero.

Figure 1a presents the regularization path, i.e. the values of the coefficients of $\hat{\beta}_L$ as a function of T in (1). The vertical dashed lines indicate the values of the T for which the number of non-zero coefficients of $\hat{\beta}_L$ is for the time larger than the mark value (multiple values of 5). The legend on the right gives the value of the 20 coefficients with the highest values (sorted by the absolute value of the coefficient).

Figure 1b presents a similar situation. In fact, the only difference is that the correlation between any two Z_i 's is now 0.5. Again, the 10 most important variables are those with non-zero true values.

Suppose we knew in advance that there are exactly 10 non-zero coefficients. It could be assumed that LASSO can be used, stopped when it first finds 10 non-zero coefficients (this corresponds to $T \approx 0.5$ in Figure 1b). However, if that was the algorithm, then only three coefficients with non-zero true value, β_3 , β_8 , and β_{10} , were included together with some 7 unrelated variables. For $T \approx 2$ the 10 largest coefficients do correspond to the 10 relevant variables, but along with them many unrelated variables are still selected (8 variables in Figure 1b), and moreover this particular choice of T cannot be known in advance if we deal with real data.

3 Randomized selection

The approach to design the model selector \mathcal{MS}_K that we believe should be used is the one applied in the examples of Section 4. It acts as follows: run the LASSO for a large model which is strictly larger than the model we want to consider, yet small enough so that standard methods for selecting a good subset of the variables can be implemented. Then run one of such methods, with given subset size K : in the examples of Section 4 we use the standard

backward selection procedure. We do not have a mathematical proof which is directly relevant to such a method. We can prove, however, the validity of an inferior backward method which is based on random selection (with appropriate weights) of the variable to be dropped at each stage. We bound the increase in the sum of squares of the randomized method. The same bounds are applied necessarily to the standard backward selection.

Suppose that we have an arbitrary estimator $\tilde{\beta}$ with values in \mathcal{R}^p , not necessarily the LASSO estimator. We may think, for example, of any estimator of parameter β_0 in the linear model of Section 2, but our argument is not restricted to that case. We now propose a randomized estimator $\hat{\beta}$ such that:

- (A) the prediction risk of $\hat{\beta}$ is on the average not too far from that of $\tilde{\beta}$,
- (B) $\hat{\beta}$ has at most K non-zero components,
- (C) large in absolute value components of $\hat{\beta}$ coincide with those of $\tilde{\beta}$.

Definition of the randomization distribution. Let \mathcal{I} be the set of non-zero coordinates of the vector $\tilde{\beta} = (\tilde{\beta}_1, \dots, \tilde{\beta}_p)$. We suppose that its cardinality $\tilde{K} = |\mathcal{I}| \geq 2$. Introduce the values

$$p_i = \min\{1, c(\tilde{K} - 1)|\tilde{\beta}_i|/\|\tilde{\beta}\|_1\}, \quad i \in \mathcal{I},$$

where $c \geq 1$ is a solution of $\sum_{i \in \mathcal{I}} p_i = \tilde{K} - 1$. Such c exists since the function

$$t \mapsto \bar{p}_i(t) \equiv \min\{1, t(\tilde{K} - 1)|\tilde{\beta}_i|/\|\tilde{\beta}\|_1\}$$

is continuous and non-decreasing, $\lim_{t \rightarrow \infty} \sum_{i \in \mathcal{I}} \bar{p}_i(t) = \tilde{K}$ and $\sum_{i \in \mathcal{I}} \bar{p}_i(1) \leq \tilde{K} - 1$. From $\sum_{i \in \mathcal{I}} p_i = \tilde{K} - 1$ we get

$$\sum_{i \in \mathcal{I}} (1 - p_i) = 1, \tag{2}$$

so that the collection $\{1 - p_i\}_{i \in \mathcal{I}}$ defines a probability distribution on \mathcal{I} that we denote by P^* . Note that there exists a p_i not equal to 1 (otherwise we have $\sum_{i \in \mathcal{I}} p_i = \tilde{K}$), in particular, we have always $p_i < 1$ for the index i that corresponds to the smallest in absolute value $\tilde{\beta}_i$. On the other hand, $p_i > 0$ since $\tilde{\beta}_i \neq 0$ for $i \in \mathcal{I}$. Therefore, $0 < p_i < 1$ for at least two indices i corresponding to the two smallest in absolute values coordinates of $\tilde{\beta}$.

Definition of the randomized selection procedure. Choose i^* from \mathcal{I} at random according to distribution P^* : $P^*(i^* = i) = 1 - p_i$, $i \in \mathcal{I}$. We

suppose that the random variable i^* is independent of the data \mathbf{y} . Define a randomized estimator $\beta^* = (\beta_1^*, \dots, \beta_p^*)$ where $\beta_{i^*}^* = 0$, $\beta_i^* = \tilde{\beta}_i/p_i$ for $i \in \mathcal{I} \setminus \{i^*\}$, and $\beta_i^* = 0$ for $i \notin \mathcal{I}$. In words, we set to zero one coordinate of $\tilde{\beta}$ chosen at random, and the other coordinates are either increased in absolute value or left intact. We will see that on the average we do not lose much in prediction quality by dropping a single coordinate in this way.

We then perform the same randomization process taking β^* as initial estimator and taking randomization independently of the one used on the first step. We thus drop one more coordinate, etc. Continuing iteratively after $\tilde{K} - K$ steps we are left with the estimator which has exactly the prescribed number K of non-zero coordinates. We denote this final randomized estimator by $\hat{\beta}$. This is the one we are interested in.

Denote by \mathbf{E}^* the expectation operator with respect to the overall randomization measure which is the product of randomization measures over the $\tilde{K} - K$ iterations.

Theorem 3.1 *Let $\mathbf{Z} \in \mathcal{R}^{n \times p}$ be a given matrix. Suppose that the diagonal elements of the corresponding Gram matrix $\mathbf{Z}'\mathbf{Z}/n$ are equal to 1, and let $\tilde{\beta}$ be any estimator with $\tilde{K} \geq 3$ non-zero components. Then the randomized estimator $\hat{\beta}$ having at most $K < \tilde{K}$ non-zero coordinates has the following properties.*

(i) *For any vector $\mathbf{f} \in \mathcal{R}^n$,*

$$\mathbf{E}^* \|\mathbf{f} - \mathbf{Z}\hat{\beta}\|^2 \leq \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \|\tilde{\beta}\|_1^2 \left(\frac{1}{K-1} - \frac{1}{\tilde{K}-1} \right).$$

(ii) *Let $\tilde{\beta}_{(j)}$ be the coordinates of $\tilde{\beta}$ ordered by absolute value: $|\tilde{\beta}_{(1)}| \geq |\tilde{\beta}_{(2)}| \geq \dots \geq |\tilde{\beta}_{(p)}|$. Suppose that $|\tilde{\beta}_{(k)}| > \|\tilde{\beta}\|_1/(\tilde{K}-1)$ for some k . Then the estimator $\hat{\beta}$ coincides with $\tilde{\beta}$ in the k largest coordinates: $\hat{\beta}_{(j)} = \tilde{\beta}_{(j)}$, $j = 1, \dots, k$.*

(iii) *Suppose that $|\tilde{\beta}_{(k+1)}| = 0$ and $|\tilde{\beta}_{(k)}| > \|\tilde{\beta}\|_1/(\tilde{K}-1)$ for some k . Then $\hat{\beta}$ keeps all the non-zero coordinates of $\tilde{\beta}$.*

Proof. It is easy to see that $\mathbf{E}^*(\beta_i^*) = \tilde{\beta}_i$ for all i and, for any vector $\mathbf{f} \in \mathcal{R}^n$,

$$\begin{aligned} \mathbf{E}^* \|\mathbf{f} - \mathbf{Z}\beta^*\|^2 &= \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \frac{1}{n} \text{trace}(\mathbf{Z}'\mathbf{Z}\Sigma^*) \\ &= \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \frac{1}{n} \sum_{i=1}^n \mathbf{z}_i' \Sigma^* \mathbf{z}_i \\ &\leq \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \sum_{j=1}^p \tilde{\beta}_j^2 \frac{1-p_j}{p_j} \end{aligned} \quad (3)$$

where \mathbf{z}_i are the rows of matrix \mathbf{Z} and $\Sigma^* = \mathbf{E}^*[(\beta^* - \tilde{\beta})(\beta^* - \tilde{\beta})']$ is the randomization covariance matrix. We used here that Σ^* is of the form

$$\Sigma^* = \text{diag} \left(\tilde{\beta}_j^2 \frac{1-p_j}{p_j} \right) - (B\tilde{\beta})(B\tilde{\beta})' \quad \text{with} \quad B = \text{diag} \left(\frac{1-p_i}{p_i} \right),$$

and the diagonal elements of $\mathbf{Z}'\mathbf{Z}/n$ are equal to 1, by assumption of the theorem.

Recall that $c \geq 1$, and therefore $|\tilde{\beta}_j| \geq \|\tilde{\beta}\|_1/(\tilde{K} - 1)$ implies $p_j = 1$. Hence,

$$\begin{aligned} \sum_{j \in \mathcal{I}} \tilde{\beta}_j^2 \frac{1-p_j}{p_j} &= \sum_{0 < |\tilde{\beta}_j| < \|\tilde{\beta}\|_1/(\tilde{K}-1)} \tilde{\beta}_j^2 \frac{1-p_j}{p_j} \\ &\leq \frac{\|\tilde{\beta}\|_1}{c(\tilde{K}-1)} \sum_{0 < |\tilde{\beta}_j| < \|\tilde{\beta}\|_1/(\tilde{K}-1)} |\tilde{\beta}_j| (1-p_j) \\ &\leq \frac{\|\tilde{\beta}\|_1^2}{(\tilde{K}-1)^2} \sum_{j \in \mathcal{I}} (1-p_j) \\ &= \frac{\|\tilde{\beta}\|_1^2}{(\tilde{K}-1)^2} \end{aligned} \quad (4)$$

where we used (2). Thus, the randomized estimator β^* with at most $\tilde{K} - 1$ non-zero components satisfies

$$\mathbf{E}^* \|\mathbf{f} - \mathbf{Z}\beta^*\|^2 \leq \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \frac{\|\tilde{\beta}\|_1^2}{(\tilde{K}-1)^2}. \quad (5)$$

Note also that β^* has the same ℓ_1 norm as the initial estimator $\tilde{\beta}$:

$$\|\beta^*\|_1 = \|\tilde{\beta}\|_1 \quad (6)$$

In fact, the definition of β^* yields

$$\begin{aligned}
\|\beta^*\|_1 - \|\tilde{\beta}\|_1 &= \left(\sum_{j \in \mathcal{I}} \frac{|\tilde{\beta}_j|}{p_j} - \frac{|\tilde{\beta}_{i^*}|}{p_{i^*}} \right) - \sum_{j \in \mathcal{I}} |\tilde{\beta}_j| \\
&= \frac{\|\tilde{\beta}\|_1}{c(\tilde{K} - 1)} \sum_{p_j < 1} \left(1 - c(\tilde{K} - 1) \frac{|\tilde{\beta}_j|}{\|\tilde{\beta}\|_1} \right) - \frac{\|\tilde{\beta}\|_1}{c(\tilde{K} - 1)} \\
&= \frac{\|\tilde{\beta}\|_1}{c(\tilde{K} - 1)} \sum_{j \in \mathcal{I}} (1 - p_j) - \frac{\|\tilde{\beta}\|_1}{c(\tilde{K} - 1)} \\
&= 0,
\end{aligned}$$

in view of 2 .

Using (5) and (6) and continuing by induction we get that the final randomized estimator $\hat{\beta}$ satisfies

$$\begin{aligned}
\mathbb{E}^* \|\mathbf{f} - \mathbf{Z}\hat{\beta}\|^2 &\leq \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \sum_{j=1}^{\tilde{K}-K} \frac{\|\tilde{\beta}\|_1^2}{(\tilde{K} - j)^2} \\
&\leq \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \|\tilde{\beta}\|_1^2 \left(\frac{1}{K-1} - \frac{1}{\tilde{K}-1} \right).
\end{aligned}$$

This proves part (i) of the theorem. Part (ii) follows easily from the definition of our procedure, since $p_j = 1$ for all the indices j corresponding to $\tilde{\beta}_{(1)}, \dots, \tilde{\beta}_{(k)}$ and the ℓ_1 norm of the estimator is preserved on every step of the iterations. The same argument holds for part (iii) of the theorem. \square

Consider now the linear model of Section 2 . Let $\tilde{\beta}$ be an estimator of parameter β_0 . Using Theorem 3.1 with $\mathbf{f} = \mathbf{Z}\beta_0$ we get the following bound on the prediction loss of the randomized estimator $\hat{\beta}$:

$$\mathbb{E}^* \|\mathbf{Z}(\hat{\beta} - \beta_0)\|^2 \leq \|\mathbf{Z}(\tilde{\beta} - \beta_0)\|^2 + \|\tilde{\beta}\|_1^2 \left(\frac{1}{K-1} - \frac{1}{\tilde{K}-1} \right). \quad (7)$$

We see that if K is large enough and the norm $\|\tilde{\beta}\|_1^2$ is bounded, the difference between the losses of $\tilde{\beta}$ and $\hat{\beta}$ is on the average not too large. For $\tilde{\beta} = \hat{\beta}_L$ we can replace $\|\tilde{\beta}\|_1^2$ by T^2 in (7).

As $\tilde{\beta}$ we may also consider another LASSO type estimator which is somewhat different from $\hat{\beta}_L$ described in Section 2 :

$$\tilde{\beta} = \arg \min_{\beta \in \mathcal{R}^p} \{ \|\mathbf{y} - \mathbf{Z}\beta\|^2 + r\|\beta\|_1 \}, \quad (8)$$

where $r = A\sqrt{(\log p)/n}$ with some constant $A > 0$ large enough. As shown in ?, for this estimator, as well as for the associated Dantzig selector, under general conditions on the design matrix \mathbf{Z} the ℓ_1 norm satisfies $\|\tilde{\beta}\|_1^2 = \|\beta_0\|_1^2 + o_p(s\sqrt{(\log p)/n})$ where s is the number of non-zero components of β_0 . Thus, if β_0 is sparse and has a moderate ℓ_1 norm, the bound (7) can be rather accurate.

Furthermore, Theorem 3.1 can be readily applied to nonparametric regression model

$$\mathbf{y} = \mathbf{f} + \varepsilon$$

where $\mathbf{f} = (f(\mathbf{X}_1), \dots, f(\mathbf{X}_n))'$ and f is an unknown regression function. In this case $\mathbf{Z}\beta = f_\beta(\mathbf{X})$ is an approximation of $f(\mathbf{X})$, for example as the one discussed in the Introduction. Then, taking as $\tilde{\beta}$ either the LASSO estimator (8) or the associated Dantzig selector we get immediately sparsity oracle inequalities for prediction loss of the corresponding randomized estimator $\hat{\beta}$ that mimic (to within the residual term $O(\|\tilde{\beta}\|_1^2/K)$) those obtained for the LASSO in ?; ? and for the Dantzig selector in ?.

It is interesting to compare our procedure with the randomization device usually referred to as the ‘‘Maurey argument’’. It is implemented as a tool to prove approximation results over convex classes of functions ?. Maurey’s randomization has been used in statistics in connection to convex aggregation ?, pages 192–193 (K -concentrated aggregation), and ?, Lemma B.1.

The Maurey randomization can be also applied to our setting. Define the estimator $\hat{\beta}_M$ as follows:

- (i) choose $K < \tilde{K}$; draw independently at random K coordinates from \mathcal{I} with the probability distribution $\{|\tilde{\beta}_i|/\|\tilde{\beta}\|_1\}_{i \in \mathcal{I}}$,
- (ii) set the j th coordinate of $\hat{\beta}_M$ equal to

$$\hat{\beta}_{Mj} = \begin{cases} \|\tilde{\beta}\|_1 k_j / K & \text{if } \tilde{\beta}_j > 0, \\ -\|\tilde{\beta}\|_1 k_j / K & \text{if } \tilde{\beta}_j < 0, \\ 0 & \text{if } j \notin \mathcal{I} \end{cases}$$

where $k_j \leq K$ is the number of times the j th coordinate is selected at step (i).

Note that, in general, none of the non-zero coordinates of $\hat{\beta}_M$ is equal to the corresponding coordinate of the initial estimator $\tilde{\beta}$. The prediction risk of $\hat{\beta}_M$ is on the average not too far from that of $\tilde{\beta}$ as the next theorem states.

Theorem 3.2 *Under the assumptions of Theorem 3.1 the randomized estimator $\widehat{\beta}_M$ with at most $K < \widetilde{K}$ non-zero coordinates satisfies*

$$\mathbf{E}^* \|\mathbf{f} - \mathbf{Z}\widehat{\beta}_M\|^2 \leq \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \frac{\|\tilde{\beta}\|_1^2}{K}. \quad (9)$$

Proof. Let η_1, \dots, η_K be i.i.d. random variables taking values in \mathcal{I} with the probability distribution $\{|\tilde{\beta}_i|/\|\tilde{\beta}\|_1\}_{i \in \mathcal{I}}$. We have $k_j = \sum_{s=1}^K I(\eta_s = j)$ where $I(\cdot)$ is the indicator function. It is easy to see that $\mathbf{E}^*(\widehat{\beta}_{Mj}) = \beta_j$ and the randomization covariance matrix $\Sigma^* = \mathbf{E}^*[(\widehat{\beta}_M - \tilde{\beta})(\widehat{\beta}_M - \tilde{\beta})']$ has the form

$$\Sigma^* = \frac{\|\tilde{\beta}\|_1}{K} \text{diag}|\tilde{\beta}_i| - \frac{1}{K} |\tilde{\beta}\| |\tilde{\beta}|' \quad (10)$$

where $|\tilde{\beta}|$ is the vector of absolute values $|\tilde{\beta}_i|$. Acting as in (3) and using (10) we get

$$\begin{aligned} \mathbf{E}^* \|\mathbf{f} - \mathbf{Z}\widehat{\beta}_M\|^2 &= \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \frac{1}{n} \sum_{i=1}^n \mathbf{z}_i' \Sigma^* \mathbf{z}_i \\ &\leq \|\mathbf{f} - \mathbf{Z}\tilde{\beta}\|^2 + \frac{\|\tilde{\beta}\|_1}{K} \sum_{j \in \mathcal{I}} |\tilde{\beta}_j| \end{aligned}$$

which yields the result. \square

The residual term in (9) is of the same order of magnitude $O(\|\tilde{\beta}\|_1^2/K)$ as the one that we obtained in Theorem 3.1. In summary, $\widehat{\beta}_M$ does achieve the properties (A) and (B) mentioned at the beginning of this section, but not the property (C): it does not preserve the largest coefficients of $\tilde{\beta}$.

Finally, note that applying (5) with $\mathbf{f} = \mathbf{y}$ we get an inequality that links the residual sums of squares (RSS) of β^* and $\tilde{\beta}$:

$$\mathbf{E}^* \|\mathbf{y} - \mathbf{Z}\beta^*\|^2 \leq \|\mathbf{y} - \mathbf{Z}\tilde{\beta}\|^2 + \frac{\|\tilde{\beta}\|_1^2}{(\widetilde{K} - 1)^2}. \quad (11)$$

The left hand side of (11) is bounded from below by the minimum of the RSS over all the vectors β with exactly $\widetilde{K} - 1$ non-zero entries among the \widetilde{K} possible positions where the entries of the initial estimator $\tilde{\beta}$ are non-zero. Hence, the minimizer β^{**} of the residual sums of squares $\|\mathbf{y} - \mathbf{Z}\beta\|^2$ over all such β is an estimator whose RSS does not exceed the right hand side of (11). Note that β^{**} is obtained from $\tilde{\beta}$ by dropping the coordinate which

has the smallest contribution to R^2 . Iterating such a procedure $\tilde{K} - K$ times we get nothing but a standard backward selection. This is exactly what we apply in Section 4 . However, the estimator obtained by this non-randomized procedure has neither of the properties stated in Theorem 3.1 since we have only a control of the RSS but not necessarily of the prediction loss, and the ℓ_1 norm of the estimators is not preserved from step to step, on the difference from our randomized procedure.

4 Examples

We consider here two examples of application of our method. The first one deals with simulated data.

Example 4.1 We considered a sample of size 250 from (Y, X_1, \dots, X_{10}) , where X_1, \dots, X_{10} are i.i.d. standard uniform, $Y = \beta_1 \mathbf{1}(\frac{1}{8} < X_1 \leq \frac{1}{4}) + \beta_2 \mathbf{1}(\frac{1}{8} < X_2 \leq \frac{1}{2}) \mathbf{1}(\frac{1}{8} < X_3 \leq \frac{3}{8}) \mathbf{1}(\frac{1}{8} \leq X_4 \leq \frac{5}{8}) + \varepsilon$, where $\mathbf{1}(\cdot)$ denotes the indicator function and ε is normal with mean 0 and variance such that the population R^2 is 0.9. The coefficients β_1 and β_2 were selected so that the standard deviation of the second term was three times that of the first.

We followed the hierarchical method (i)–(iii) of the Introduction. Our initial set \mathcal{F}_0 was a collection of $L = 32$ step functions for each of the ten variables ($d = 10$). The jump points of the step functions were equally spaced on the unit interval. The cardinality of \mathcal{F}_0 was 279 (after taking care of multicollinearity). At each step we run the LASSO path until $\tilde{K} = 40$ variables were selected, from which we selected $K = 20$ variables by the standard backward procedure. Then the model was enlarged by including interaction terms, and the iterations were continued until there was no increase in R^2 .

The first step (with single effects only) ended with $R^2 = 0.4678$, and the correlation of the predicted value of Y with the true one was 0.4885. The second iteration (two way interactions) ended with $R^2 = 0.6303$ and correlation with the truth of 0.6115. The third (three and four ways interactions were added) ended with $R^2 = 0.7166$ and correlation of 0.5234 with the truth. The process stopped after the fifth step. The final predictor had correlation of 0.5300 with the true predictor.

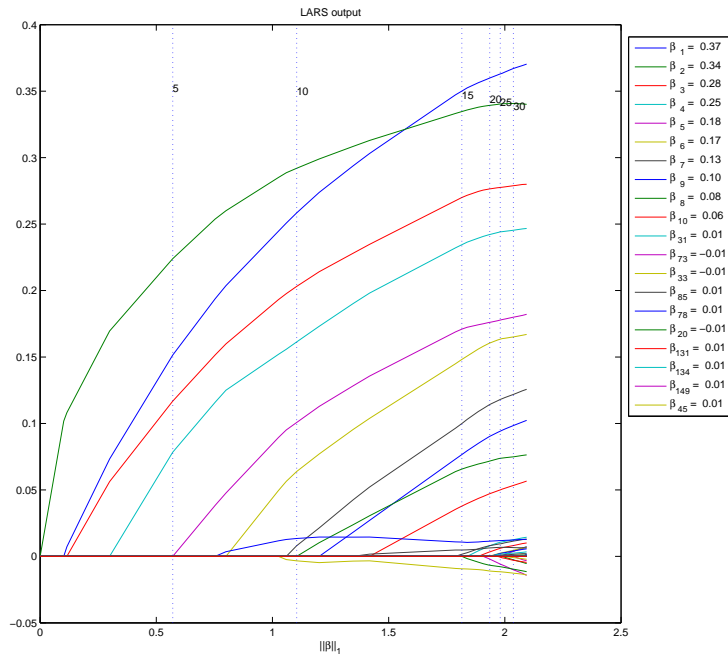
The LASSO regularization path for the final (fifth) iteration is presented in Figure 2 . The list of 20 terms included in the model is given in the legend where i_k denotes the the k th step function of variable i . The operator \times denotes interaction of variables. We can observe that the first 12 selected

terms are functions of variables 1 to 4 that are in the true model. Some of the 20 terms depend also on two other variables (8 and 10) that do not belong to the true model.

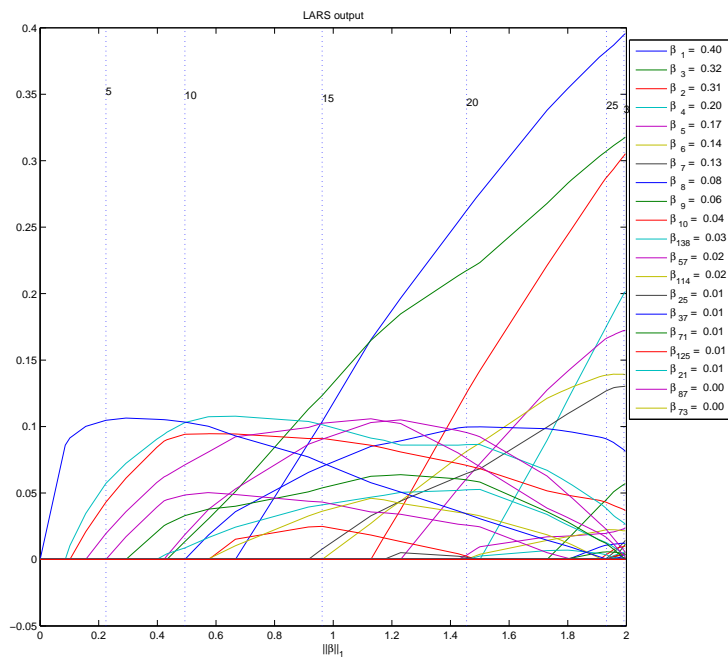
Example 4.2 (The Abalone Data) The abalone data set, taken from <ftp://ftp.ics.uci.edu/pub/machine-learning-databases/abalone/>, gives the age of abalone (as determined by cutting the shell and counting the number of rings) and some physical measurements (sex, length, diameter, height, whole weight, weight of meat, gut weight, and shell weight after being dried). The data was described initially by Nash, et al in 1994. We selected at random 3500 data points as a training set. The 677 remaining points were left as a test bed for cross-validation.

We used as a basic function of the univariate variable the ramp function $(x - a)\mathbf{1}(x > a)$. The range of the variables was initially normalized to the unit interval, and we considered all break points a on the grid with spacing $1/32$. However, after dropping all transformed variables which are in the linear span of those already found, we were left with only 17 variables. We applied the procedure with LASSO which ends with at most $\tilde{K} = 60$ variables, from which at most $K = 30$ were selected by backward regression.

The first stage of the algorithm ends with $R^2 = 0.5586$ (since we started with 17 terms and we were ready to leave up to 30 terms, nothing was gained in this stage). The second stage, with all possible main effects and two-way interactions, dealt already with 70 variables and finished with only slightly higher R^2 (0.5968). The algorithm stopped after the fifth iteration. This iteration started with 2670 terms, and ended with $R^2 = 0.5779$. The correlation of the prediction with the observed age of the test sample was 0.5051. The result of the last stage is given in Figure 3 . It can be seen that the term with the largest coefficient is that of the whole weight. Then come 3 terms involving the meat weight, and its interaction with the length. The shell weight which was most important when no interaction terms were allowed, became not important when the interactions were added.



(a)



15
(b)

Figure 1: Selecting variables. Coefficients vs. L_1

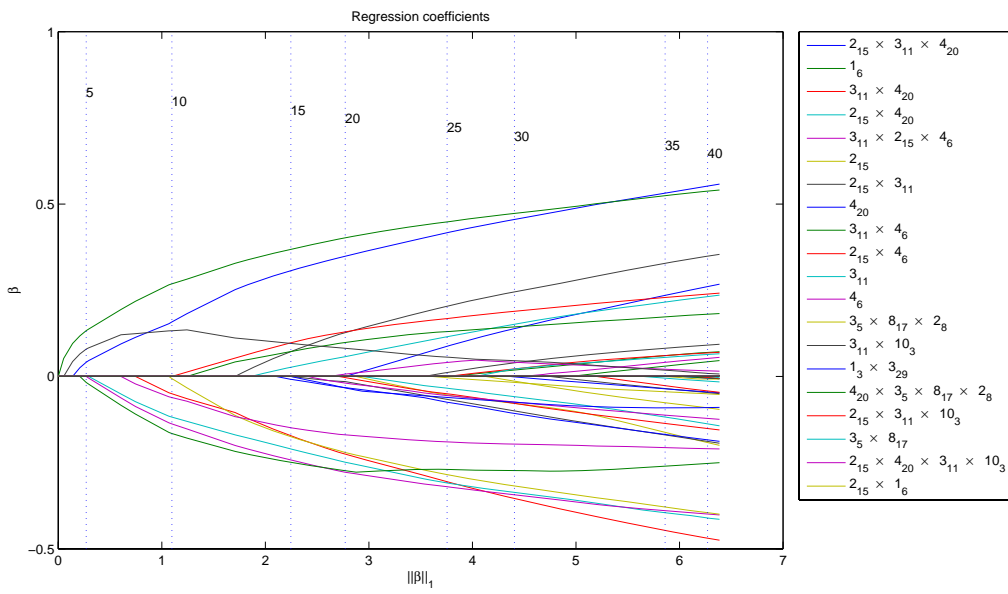


Figure 2: The final path of the LASSO algorithm for the simulation of Example 4.1 .

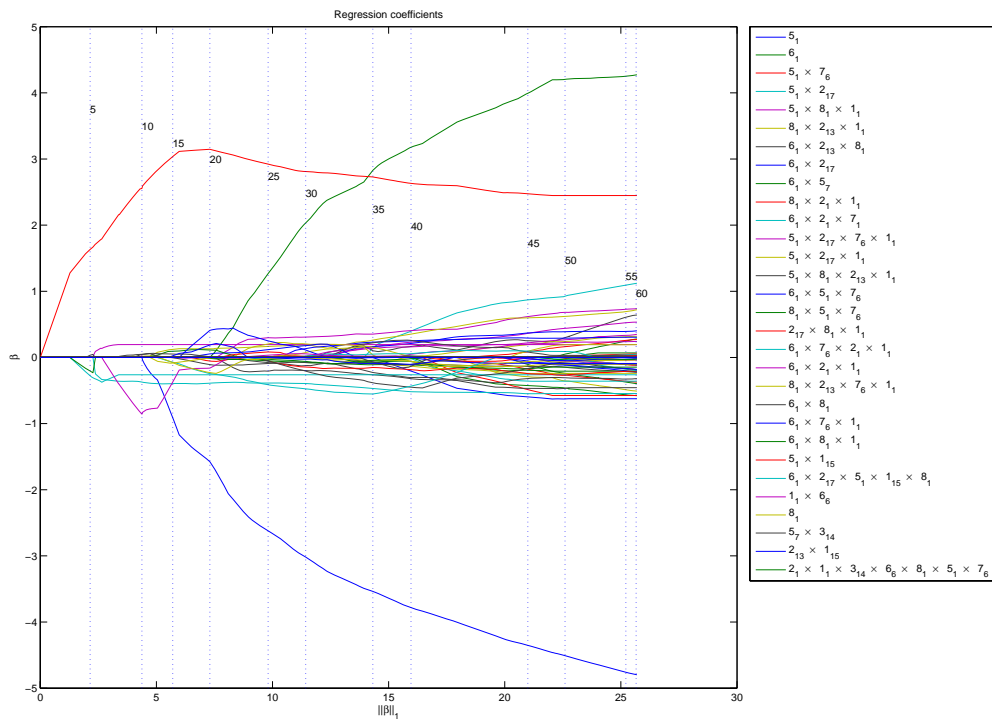


Figure 3: The final path of the LASSO algorithm for the abalone data set.