Automatica 48 (2012) 2107–2115

Contents lists available at SciVerse ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Variable selection via RIVAL (removing irrelevant variables amidst Lasso iterations) and its application to nuclear material detection[☆]

Paul Kump[a], Er-Wei Bai[a,d,1], Kung-sik Chan[b], Bill Eichinger[c], Kang Li[d]

[a] Department of Electrical and Computer Engineering, University of Iowa, Iowa City, IA 52242, United States
[b] Department of Statistics, University of Iowa, Iowa City, IA 52242, United States
[c] Department of Civil Engineering, University of Iowa, Iowa City, IA 52242, United States
[d] School of Electronics, Electrical Engineering and Computer Science, Queen's University, Belfast BT7 1NN, UK

ARTICLE INFO

Article history:
Received 22 April 2011
Received in revised form
16 February 2012
Accepted 20 April 2012
Available online 10 July 2012

Keywords:
Model selection
RIVAL
Positive Lasso
Nuclear material detection application

ABSTRACT

In many situations, the number of data points is fixed, and the asymptotic convergence results of popular model selection tools may not be useful. A new algorithm for model selection, RIVAL (removing irrelevant variables amidst Lasso iterations), is presented and shown to be particularly effective for a large but fixed number of data points. The algorithm is motivated by an application of nuclear material detection where all unknown parameters are to be non-negative. Thus, positive Lasso and its variants are analyzed. Then, RIVAL is proposed and is shown to have some desirable properties, namely the number of data points needed to have convergence is smaller than existing methods.

1. Introduction

This paper concerns a problem of parameter estimation and model selection. Let the system be represented by

$$y_n = X_n\beta^* + v_n \tag{1.1}$$

where $v_n = (v(1), \dots, v(n))^T \in \Re^n$ is an i.i.d. random noise sequence with zero mean and finite variance, $y_n = (y(1), \dots, y(n))^T \in \Re^n$ is the output sequence or the response sequence and $X_n \in \Re^{n\times p}$ is the regressor matrix. The regressors can be correlated or depend on the past values of the outputs. However in this paper, they are treated as given deterministic values.

The system (1.1) is assumed to be sparse, i.e., some of the unknown coefficients $\beta^* = (\beta_1^*, \dots, \beta_p^*)^T \in \Re^p$ are exactly zero corresponding to the regression vectors that are irrelevant to the output. Further, the unknown coefficients which are not zero are assumed to be positive. This assumption is motivated by our intended application of nuclear material detection which will be discussed in detail later. Though the assumption is non-trivial, it does have its place in a number of applications. For example, Frank and Heiser (2008) shows that Feature Network Models can be considered a linear regression problem with the positivity constraint. Another example is in the localization of DNA-binding proteins where each binding event generates a positive signal (Reiss, Facciotti, & Baliga, 2008). The authors of Chang, Hsu, and Huang (2010) use this positivity assumption for modeling non-stationary spatial data and apply it to finding NO_3 concentrations in US precipitation.

Without loss of generality by re-arranging indices, we assume

$$\beta^* = \begin{pmatrix} \beta_1^* \\ \vdots \\ \beta_d^* \\ 0 \\ \vdots \\ 0 \end{pmatrix} \qquad \beta_1^* > 0, \dots, \beta_d^* > 0, \tag{1.2}$$

$$\beta_{d+1}^* = \cdots = \beta_p^* = 0$$

$$X_n = \begin{pmatrix} X_{n,11} & X_{n,12} \\ X_{n,21} & X_{n,22} \end{pmatrix}$$

[☆] The work was supported in part by the DoE DE-FG52-09NA29364. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Martin Enqvist under the direction of Editor Torsten Söderström.

E-mail addresses: martymac13@gmail.com (P. Kump), er-wei-bai@uiowa.edu (E.-W. Bai), kungsik.chan@gmail.com (K.-s. Chan), william-eichinger@uiowa.edu (B. Eichinger), k.li@qub.ac.uk (K. Li).

[1] Tel.: +1 319 335 5949; fax: +319 335 6028.

0005-1098/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2012.06.051

where

$$X_{n,11} \in \Re^{d\times d}, \qquad X_{n,12} \in \Re^{d\times(p-d)}, \qquad X_{n,21} \in \Re^{(n-d)\times d}, \qquad X_{n,22} \in \Re^{(n-d)\times(p-d)}$$

for some unknown $0 < d < p$. Now, define the index set to be

$$A^* = \{j : \beta_j^* > 0\}. \tag{1.3}$$

Model selection or variable selection consists in identifying the index set A^* and removing those irrelevant variables that will simplify the model and determining which variables contribute to the output. This has applications in a number of areas including engineering. In identification, suppose an unknown system is represented by a linear combination of some basis functions $\phi(k)$'s that are known functions of the input and/or the output and some unknown coefficients β_j^*. It is usually the case that p has to be sufficiently large in order to approximate the unknown system well, resulting in many zero coefficients β_j^*'s. One such application is aircraft testing model selection (Kukreja, 2009; Kukreja, Lofberg, & Brenner, 2006). Another example is the detection of nuclear materials.

The ordinary least squares method does not solve the model selection problem. Though convergent, it gives non-zero estimates for all β_j's. Without knowing d, to determine how small is actually zero is a difficult task. A number of subset selection methods are available in the literature (Billings, Chen, & Korenberg, 1989; Chen, Billings, & Luo, 1989; Chen & Wigger, 1995; Kukreja, 2005; Li, Peng, & Bai, 2006; Lind & Ljung, 2005; Young, McKenna, & Bruun, 2001) including the forward step, backward step and combined forward and backward step methods. These methods are gradient-based algorithms; as such, they are suboptimal and often get trapped into a local minimum. Further, the methods are inefficient in terms of finding which β_j^* is exactly zero and which β_j^* is not (Efron, Hastie, Johnstone, & Tibshirani, 2004).

The second approach is a regularized or penalized approach like ridge regression (Hoerl & Kennard, 1970), bridge regression (Frank & Friedman, 1993), non-negative garrote (Breiman, 1995; Yuan & Lin, 2007) and Lasso (Tibshirani, 1996). Ridge regression is a convex optimization problem that, much like the ordinary least squares method, has a closed form solution allowing a very efficient computation. Ridge regression has the property that the amount of shrinkage toward zero increases with the magnitude of β_j. For large β_j^* the shrinkage is large, and for small β_j^* the shrinkage is basically non-existent. Thus, ridge regression often results in "uniformly" small but non-zero estimates that, much like ordinary least squares estimates, do not produce an interpretable index set. Bridge regression is not a convex problem and is expensive computationally. However, it was shown to have the ability of capturing the correct index set. This ability does not mean that bridge regression does not have strong opposition. Besides computational complexity, it was argued that the solution of bridge regression is not continuous and thus less favorable (Fan & Li, 2001; Zou, 2006). The non-negative garrote was shown to be more accurate and stable than some traditional subset selection methods.

The celebrated Lasso estimates the parameters by minimizing

$$J_1(\beta) = \min_{\beta} \|y_n - X_n\beta\|^2 + \lambda(n)\sum_{j=1}^{p} |\beta_j|$$

where $\lambda(n)$ is a pre-specified weight, and $\|\cdot\|$ is the Euclidean norm. We use the notation $\lambda(n)$ to explicitly show the dependence of λ on the number of data n. The original Lasso is actually in a slightly different form, $\min \|y_n - X_n\beta\|^2$ subject to $\sum_{j=1}^{p} |\beta_j| \leq t$ for a given $t > 0$, though these two forms are equivalent. Lasso is very successful in a wide range of applications when the number of data points n is large, and it can be solved efficiently by the LARS algorithm (Efron et al., 2004). Regarding our application, the performance of the Lasso can be much improved by incorporating the *a priori* information that $\beta_j^* \geq 0$ directly into the minimization, resulting in the positive Lasso (Efron et al., 2004):

$$J_1(\beta) = \min_{\beta\geq 0} \|y_n - X_n\beta\|^2 + \lambda(n)\sum_{j=1}^{p} \beta_j,$$

where $\lambda(n)$ is the positive regularization parameter and $\beta \geq 0$ stands for $\beta_i \geq 0, i = 1, 2, \ldots, p$. The positive Lasso can be modified further to

$$J_1(\beta) = \min_{\beta\geq 0} \|y_n - X_n\beta\|^2 + \lambda(n)\sum_{j=1}^{p} w_j\beta_j, \tag{1.4}$$

where

$$0 \leq w = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{pmatrix} = \begin{pmatrix} w_{11} \\ \vdots \\ w_{1d} \\ w_{21} \\ \vdots \\ w_{2(p-d)} \end{pmatrix} \in \Re^p$$

is a non-negative weighting vector. The second term in (1.4) is the penalty. Introducing the weighting vector in the penalty term allows for more flexibility in the minimization. The idea is to shrink the coefficients towards zero as $\lambda(n)$ increases and hopefully, β_j's are shrunk exactly to zero for those j's that $\beta_j^* = 0$. Therefore, the selection of $\lambda(n)$ is critical to the performance of positive Lasso. Once the optimal $\lambda(n)$ is chosen (discussed in Section 4), the solution can be computed with a modified LARS (Efron et al., 2004) algorithm or the algorithm provided in Section 2. Of course, an optimal $\lambda(n)$ does not necessarily imply that positive Lasso will find the correct model. Model selection error is inherent to Lasso (Bai, Chan, Eichinger, & Kump, 2011), but we look to use Lasso's simplicity to build on and hopefully reduce model selection errors.

The goal of model selection is to identify the index set $A^* = \{j : \beta_j^* > 0\}$ and parameter vector β^* based on y_n and X_n, to remove those superfluous variables $\beta_{d+1}^*, \ldots, \beta_p^*$ and the corresponding regressor vectors $\begin{pmatrix} X_{n,12} \\ X_{n,22} \end{pmatrix}$ and then to build a model

$$\hat{y}_n = \begin{pmatrix} X_{n,11} \\ X_{n,21} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_d \end{pmatrix}$$

where $\hat{y}_n$ is the predicted output and β_j's, $j = 1, \ldots, d$, are the estimates of β_j^*.

The desirable properties of model selection are (Fan & Li, 2001; Zou, 2006)

(1)

$$\text{Prob}\{A = A^*\} \to 1, \quad \text{as } n \to \infty \tag{1.5}$$

where A is the estimate of A^* for a given data set y_n and X_n,

(2)

$$\text{Prob}\{\beta_j = \beta_j^*\} \to 1, \quad \text{as } n \to \infty,\ j = 1, \ldots, d. \tag{1.6}$$

In other words, those positive β_j^*'s and zero β_j^*'s are correctly identified.

It is true that the ordinary least squares estimate has the properties of (1.5) and (1.6) at $n = \infty$. In practice, the number of data points never reaches infinity and the ordinary least squares method is unlikely to produce zero estimates. Lasso and its variants have the potential ability to produce zero estimates with only a finite number of data points—sometimes a *very large* number of data points.

In this paper, we analyze the positive Lasso (1.4) and provide conditions for it to correctly solve the model selection problem.

This analysis paves the way for adaptive positive Lasso, for which we provide an algorithm. The major contribution of this paper, however, is a new algorithm, which we call RIVAL (Removing Irrelevant Variables Amidst Lasso iterations), that can solve the model selection problem with fewer data points than is required by existing methods including adaptive Lasso, adaptive positive Lasso and non-negative garrote. We provide theoretical proofs and numerical simulations to support our claim.

The rest of this article is organized as follows. In Section 2, a very simple but efficient numerical algorithm is proposed to solve the positive Lasso problem along with the convergence results. In Section 3, we provide the necessary and sufficient conditions for the parameter consistency and set consistency of the positive Lasso. We then define the adaptive positive Lasso and give the conditions for it to have set and parameter consistency. In Section 4, RIVAL is proposed with a focus on large but fixed data points. We give two numerical examples, one with highly correlated regressors and one with orthogonal regressors, and demonstrate that RIVAL works well for both examples whereas none of the existing methods work well for both examples. We apply RIVAL to the detection of nuclear material in Section 5. Some final remarks are provided in Section 6. All the proofs are provided in the Appendix.

2. Positive Lasso algorithm

It is known that the positive Lasso can be numerically calculated by modifying LARS (Efron et al., 2004). The resultant algorithm is very efficient but sensitive in terms of finding the right variables if the regressors are highly correlated which is the case in our nuclear material detection problem. If two regressors are highly correlated, the algorithm may find a wrong regressor to proceed in the presence of small noise (see examples in Section 4.2). To avoid this problem, we propose an approach to solve the problem which is a coordinate ascent type of algorithm well known in the literature.

Let the initial estimate be $\beta(0) \geq 0$ and let

$$Q = X_n^T X_n = \begin{pmatrix} q_{11} & \cdots & q_{1p} \\ \vdots & \ddots & \vdots \\ q_{p1} & \cdots & q_{pp} \end{pmatrix},$$

$$C^T = \left(C_1, \dots, C_p\right) = y_n^T X_n - \frac{1}{2}\lambda(n) w^T.$$

```
Set $k = 0$.
Step 1. At each $k$, let $\gamma(k) = \beta(k) = \begin{pmatrix} \beta_1(k) \\ \vdots \\ \beta_p(k) \end{pmatrix}$. Set $l = 1$.
    Step 1.1. For each $l$, calculate
        $\gamma_l = \max\left(\frac{C_l - \sum_{j \neq l} q_{lj}\gamma_j(k)}{q_{ll}}, 0\right).$
    Step 1.2. Let $\gamma_l(k) = \gamma_l$. If $l < p$, set $l = l + 1$ and go to Step 1.1 If $l = p$, go to Step 2.
Step 2. Set $\beta(k+1) = \gamma(k)$, $k = k + 1$ and go to Step 1. (This step may be modified to add the stopping criterion. For instance, terminate the iteration if $\|\beta(k+1) - \beta(k)\| / \|\beta(k)\|$ is smaller than a prescribed smaller number.)
```

It is clear that $\beta(k)$ is a convergent sequence and converges to a local minimum that is also the global minimum. Even when two regressors are highly correlated, the algorithm performs well. In the two dimensional case, two highly correlated regressors make the ellipses in Fig. 1 extremely narrow, and noise jitters their centers and has very little effect on the algorithm selecting the correct variables. Summarizing, we have

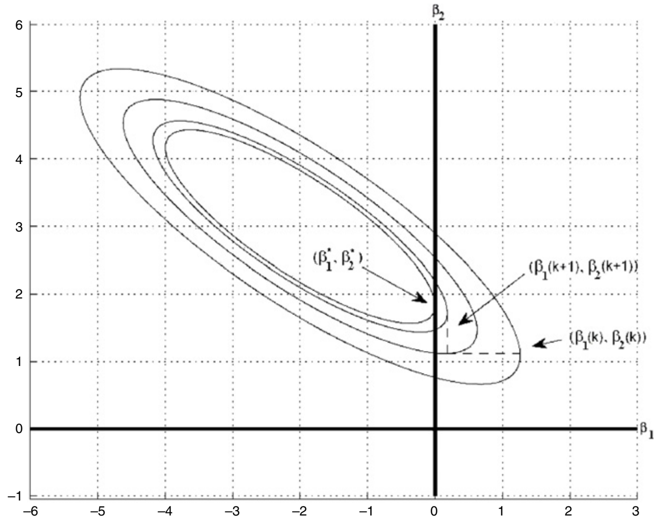


Fig. 1. Illustration of the positive Lasso algorithm in two dimensions.

Theorem 2.1. *Consider the positive Lasso for given $\lambda(n) > 0$, $w > 0$, y_n and X_n. Assume $Q = X_n^T X_n > 0$. Then the sequence $\beta(k)$ generated by the above algorithm converges to the solution of the positive Lasso.*

3. Consistency of positive Lasso, adaptive positive Lasso

Define C_n as

$$\begin{aligned} \alpha_2 I \geq \frac{1}{n} X_n^T X_n &= C_n \\ &= \begin{pmatrix} c_n(1,1) & c_n(1,2) \\ c_n^T(1,2) & c_n(2,2) \end{pmatrix} \geq \alpha_1 I > 0 \end{aligned}$$

where the α_i's are independent of n, and define

$$a_n = c_n^T(1,2) c_n^{-1}(1,1) \begin{pmatrix} w_{11} \\ \vdots \\ w_{1d} \end{pmatrix} \in \Re^{p-d}.$$

Then, in an analysis similar to that of Zhao and Yu (2006) for Lasso, it can be shown that when $\frac{n}{\lambda(n)} \to \infty$ and $\frac{\lambda(n)}{\sqrt{n}} \to \infty$ as $n \to \infty$, a sufficient condition for the positive Lasso to have set consistency (1.5) and parameter consistency (1.6) is

$$a_{nj} \leq w_{2j} - \epsilon_n, \quad j = 1, 2, \dots, p - d \tag{3.7}$$

and a necessary condition is

$$a_{nj} \leq w_{2j} + O_p\left(\frac{\sqrt{n}}{\lambda(n)}\right), \quad j = 1, 2, \dots, p - d$$

where $\epsilon_n > 0$ is a positive sequence satisfying

$$\epsilon_n \to 0, \qquad \epsilon_n \Big/ \left(\frac{\lambda(n)}{n}\right) \to \infty, \qquad \epsilon_n \Big/ \left(\frac{\sqrt{n}}{\lambda(n)}\right) \to \infty$$

as $n \to \infty$.

The sufficient condition $a_{nj} \leq w_{2j} - \epsilon_n$ is similar to the one derived in Zhao and Yu (2006) but is tighter and more general, especially as it applies to positive Lasso. For instance, the condition $|a_{nj}| \leq 1 - \eta$ for some $\eta > 0$ derived in Zhao and Yu (2006) applies to the flat weights $w_1 = \cdots = w_p = 1$, and the condition derived here $a_{nj} \leq w_{2j} - \epsilon_n$ applies to arbitrary weights.

The positive Lasso is convex and efficient computationally. However, whether it can capture the index set A^* depends on the (almost necessary and) sufficient condition (3.7). Observe that a_n depends on the data X_n and the weights w_{1j}. In general, the condition is not satisfied. Thus, the question is if the positive Lasso can be modified to ensure the correct detection of the index set A^*, at least for large n. It is interesting to observe that the condition depends on the choice of the weights $w_{1j} > 0$ and $w_{2j} > 0$:

$$c_n^T(1,2)c_n^{-1}(1,1)\begin{pmatrix} w_{11} \\ \vdots \\ w_{1d} \end{pmatrix} \leq \begin{pmatrix} w_{21} \\ \vdots \\ w_{2(p-d)} \end{pmatrix} - \epsilon_n$$

that is automatically satisfied if the weights were chosen in such a way that w_{1j}'s are small and w_{2j}'s are large enough, giving rise to the right index set A^*. The problem is of course that we do not know which $\beta_j^* = 0$ and which $\beta_j^* > 0$. To overcome this difficulty, we may use the idea of data dependent weights as in the adaptive Lasso. Based on the discussion, we propose an adaptive positive Lasso, the algorithm for which is similar to that of the adaptive Lasso: instead of solving the regular Lasso, solve the positive Lasso as in (1.4). Then, by a similar argument as in Zou (2006) for adaptive Lasso, we have the following result:

Theorem 3.1. *Consider the adaptive positive Lasso and assume* $\frac{n}{\lambda(n)} \to \infty$ *and* $\frac{\lambda(n)}{\sqrt{n}} \to \infty$ *as* $n \to \infty$. *Then, in probability as* $n \to \infty$, $A \to A^*$ *and* $\beta \to \beta^*$.

4. RIVAL

In the previous section, the consistencies of the positive Lasso and adaptive positive Lasso have been established. All results are asymptotical, i.e., $A \to A^*$ and $\beta \to \beta^*$ as $n \to \infty$. However in many applications, the number of data points n is large but finite which is particularly true in our nuclear material detection application. It would be nice to have some results that could apply to a large but fixed n. For a fixed n, we want to generate a sequence of weights $w(k) > 0$'s such that if $\beta(k)$ is the solution of the positive Lasso for the given weight $w(k)$, $0 < \eta_1 \leq \beta_i(k) \leq \eta_2 < \infty, i = 1, 2, \ldots, d$ and $\beta_i(k) = 0, i = d+1, \ldots, p$ as $k \to \infty$. Note the iteration of $\beta(k)$ is with respect to a fixed n. This is completely different from the positive Lasso or the adaptive positive Lasso discussed previously where only when a new observation is made or $n \to n+1$, a new β based on the $n+1$ data points is recalculated.

It is easy to see that if (1.5) is satisfied, i.e., the set A^* can be correctly identified, we can achieve (1.6) by applying the ordinary least squares method by trimming off the variables $\beta_j, j = d+1, \ldots, p$ and the corresponding regression vectors $\binom{X_{n,12}}{X_{n,22}}$. Therefore, we will focus on the goal (1.5).

4.1. Description of the RIVAL algorithm

Consider the positive Lasso as described in (1.2)–(1.4) for given λ, n, y_n and X_n. Assume $\frac{1}{n}X_n^TX_n > 0$. Let $0 \leq q(k) \leq 1$ be a sequence satisfying $\sum_{k=1}^{\infty} q(k) = \infty$.

Step 1. Let $\bar{w}(1) = (\bar{w}_1(1), \ldots, \bar{w}_p(1))^T = (1, 1, \ldots, 1)^T$, $w_j(1) = \xi_j\bar{w}_j(1), j = 1, \ldots, p$ and set $k = 1$, where ξ_j is the norm of the jth column of the matrix $\frac{1}{n}X_n^TX_n$.

Step 2. Apply the positive Lasso as in (1.4) and denote $\beta(k) = (\beta_1(k), \beta_2(k), \ldots, \beta_p(k))^T$ the solution of the positive Lasso with respect to the weight vector $w(k)$.

Step 3. If $\beta_j(k) = 0$, set $\beta_j(k+i) = 0$, for all $i \geq 0$. Remove β_j and w_j from β and w respectively,

$$\begin{pmatrix} \beta_1 \\ \vdots \\ \beta_{j-1} \\ \beta_j \\ \beta_{j+1} \\ \vdots \\ \beta_p \end{pmatrix} \to \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_{j-1} \\ \beta_{j+1} \\ \vdots \\ \beta_p \end{pmatrix}, \quad \begin{pmatrix} w_1 \\ \vdots \\ w_{j-1} \\ w_j \\ w_{j+1} \\ \vdots \\ w_p \end{pmatrix} \to \begin{pmatrix} w_1 \\ \vdots \\ w_{j-1} \\ w_{j+1} \\ \vdots \\ w_p \end{pmatrix}.$$

Also remove the corresponding jth column from X_n so the dimension of the optimization is reduced by one. If $\beta_j(k) > 0$, let

$$\bar{w}_j(k+1) = q(k) \cdot \frac{1}{\beta_j(k)} + (1 - q(k))\bar{w}_j(k),$$
$$w_j(k+1) = \xi_j\bar{w}_j(k+1).$$

Repeat the process for all $j = 1, 2, \ldots, p$.

Step 4. Set $k = k+1$ and go back to Step 2. The dimension could be reduced if some of $\beta_j = 0$ at Step 3. (This step may be modified to add the stopping criterion using the standard one in the numerical analysis, e.g., the iteration stops if $\|\beta(k+1) - \beta(k)\|/\|\beta(k)\|$ is smaller than the prescribed threshold.)

Theorem 4.1. *Consider the RIVAL algorithm. Define* $a = \frac{n}{\lambda(n)}$, $\hat{d} = (\hat{d}_1, \ldots, \hat{d}_p) = \frac{2\nu_n^TX_n}{\sqrt{n}}\frac{\sqrt{n}}{\lambda(n)}$, $c_j = \hat{d}_j/\xi_j$, *and* $b_j = \beta_j^* + \frac{\hat{d}_j}{2a\xi_j}$. *Assume* $n/\lambda(n), \lambda(n) \to \infty$ *as* $n \to \infty$. *Further assume* X_n *is orthogonal, i.e.,* $\frac{1}{n}X_n^TX_n = \begin{pmatrix} \xi_1 & 0 & \cdots & 0 \\ 0 & \xi_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \xi_p \end{pmatrix} > 0$. *Then, there is an integer* $n_0 > 0$ *such that for any* $n \geq n_0$, *there exists an integer* $k_0 > 0$ *such that when*

$$b_j > \delta_j = \frac{b_j - \sqrt{b_j^2 - 2/a}}{2} > 0, \qquad 1 \leq 2ab_j - \frac{1}{b_j - \delta_j}, \quad j = 1, \ldots, d \tag{4.8}$$

and

$$a \geq \frac{(c_j + \delta)^2}{8} \quad \text{for any } \delta > 0,\ j = d+1, \ldots, p \tag{4.9}$$

the sequence $\beta(k)$ *generated by the above RIVAL satisfies*

$$0 < \eta_1 \leq \beta_1(k), \ldots, \beta_d(k) \leq \eta_2 < \infty, \quad \forall k \geq k_0$$
$$\beta_{d+1}(k) = \beta_{d+2}(k) = \cdots = \beta_p(k) = 0, \quad \forall k \geq k_0$$

or equivalently

$$A(k) = A^* \quad \forall k \geq k_0$$

where $A(k)$ *is the estimated index set at stage* k.

The above results are asymptotic. For a finite n, there is no way to choose a λ that could guarantee $A = A^*$.

Also in the RIVAL algorithm, the increasing rate of $\lambda = \lambda(n)$ as n gets larger is specified which guarantees the set consistency asymptotically. In applications, however, n is fixed and how to choose an optimal λ becomes an issue. Recall the idea of RIVAL is that by a properly chosen λ for a fixed n, a "correct" sequence of weights can be generated that identifies and eliminates irrelevant variables and the corresponding regression vectors at each iteration (hence, the acronym Removing Irrelevant Variables Amidst Lasso iterations). Too small λ would not be useful for identification of irrelevant predictors and too greedy λ would likely mis-identify important variables. To this end, the optimal λ_{optimal} can be determined by some information based criterion. In this paper, the well known and extensively studied AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion) (Shao, 1997) are adopted. To emphasize the dependence of A and λ, we write A as $A(\lambda)$ and β_{ALS} as $\beta_{A(\lambda)LS}$. Further, let $q(\lambda)$ be the number of non-zero components of β_{ALS} and X_{An} denote X_n after the corresponding columns are removed according to A. The optimal λ_{optimal} is defined as

$$\lambda_{\text{optimal}} = \arg\min_{\lambda}\left\{n \ln \frac{\|y_n - X_{An}\beta_{A(\lambda)LS}\|^2}{n} + \alpha q(\lambda)\right\}$$

where $\alpha = 2$ and $\alpha = \ln(n)$ represent AIC or BIC respectively.

The idea is that with a number of candidate λ's, solve the positive Lasso for each λ, and generate the corresponding subset of regressors and trim off the irrelevant regressors from X_n and β_j's from the estimate. Then, the ordinary least squares estimate is computed and used to determine the estimate's AIC/BIC value. The candidate λ which minimizes the AIC/BIC value is the optimal λ to be used with RIVAL. In actual implementation, LARS of Efron et al. (2004) provides a path of Lasso solutions and thus efficiently solves the optimal λ based on AIC/BIC.

4.2. Discussion and examples

We make a few comments about RIVAL.

- The idea of RIVAL is that the weights $w_j(k)$'s corresponding to non-zero coefficients $\beta_j^*, j = 1, \ldots, d$ are uniformly bounded for all k and the weights $w_j(k)$ corresponding to zero coefficients $\beta_j^*, j = d+1, \ldots, p$ grow monotonically until the corresponding estimate $\beta_j = 0$.
- The initial estimate $\beta(1)$ is generated by the uniform weights $w_i(1) = 1, i = 1, \ldots, p$. To increase the algorithm's speed, the inverse of the least squares could be used: $w_i(1) = |1/\beta_{iLS}|$ if n is large.
- At least in theory, RIVAL has the perfect ability to find the index set if $\frac{1}{n}X_n^T X_n$ is diagonal and positive and n is large enough to satisfy conditions (4.8) and (4.9). Simulations seem to suggest that the results also hold for a much larger class, including when matrix $X_n^T X_n$ is diagonally dominated.
- Clearly, if $A = A^*$, we can apply the ordinary least squares estimate $\beta_{ALS} = (\beta_{A1}, \ldots, \beta_{Ad})'$ to estimate $(\beta_i^*, \ldots, \beta_d^*)'$ by trimming off zero coefficients $\beta_j = 0, j = d+1, \ldots, p$ identified and the corresponding regression vectors. The subscript ALS indicates that the estimate β_{ALS} is A dependent that is a result of RIVAL. If n is allowed to be large, the convergence of β_{ALS} to $(\beta_i^*, \ldots, \beta_d^*)'$ is guaranteed by the least squares estimate properties and the convergence of A to A^*.

The idea of RIVAL is similar to non-negative garrote and adaptive Lasso: weights are adjusted based on a convergent sequence of estimates. All three have asymptotical set consistency with proper choices of λ as n gets larger. However for a fixed and finite n, the performances are different. Because of the self-adjusting ability for a fixed n which neither adaptive Lasso nor non-negative garrote has, RIVAL seems to work better. Though it is hard to quantify the improvement theoretically, several large simulations seem to show that. We give two examples here:

Example 1.

$$y_n = \begin{pmatrix} 10 & 20 \\ 10 & 21 \end{pmatrix} \begin{pmatrix} 10 \\ 0 \end{pmatrix} + v_n, \quad v_n \sim \mathcal{N}(0, 15^2 \cdot I).$$

Example 2.

$$y_n = \begin{pmatrix} 10 & -30 \\ 10 & 30 \end{pmatrix} \begin{pmatrix} 10 \\ 0 \end{pmatrix} + v_n, \quad v_n \sim \mathcal{N}(0, 15^2 \cdot I).$$

Both are two dimensional for a small $n = 2$, that makes set consistency a tough task. In Example 1, the two regressors are highly correlated while in Example 2, the two are orthogonal. In both cases, the noise components are i.i.d. normal with zero mean and variance 15^2.

To guarantee the asymptotical consistency, λ's were chosen according to their theoretically specified values. $\lambda = n^{1/3} = 2^{1/3}$ for the adaptive Lasso according to (Zou, 2006) and $\lambda = 3\sqrt{\log(n)/n} = 3\sqrt{\log(2)/2}$ for the non-negative garrote according to (Yuan & Lin, 2007). For RIVAL $\lambda = n^{1/3} = 2^{1/3}$ was also chosen.

Table 1
Successful rates for finding the correct index set. The top one is for Example 1 and the bottom one for Example 2 (RG = ridge, NNG = non-negative garrote, AL = adaptive Lasso).

Example 1	RIVAL	LARS	NNG	AL	RG
	0.9714	0.0002	0.0017	0.0008	0

Example 2	RIVAL	LARS	NNG	AL
	0.9994	0	0	0.0415

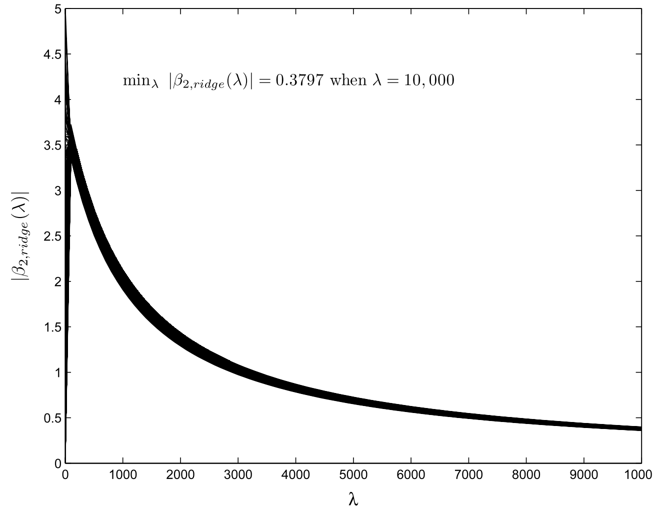


Fig. 2. Trajectories of β_2 provided by ridge regression as functions of λ.

For Example 1, 10,000 Monte Carlo simulations were carried out for ridge regression (λ varied from 0 to 10,000), Lasso implemented by LARS, adaptive Lasso, non-negative garrote and RIVAL. The top table in Table 1 shows the successful rates of the five methods in the task of finding the correct index set.

Clearly, adaptive Lasso did not work well and only had 8 successes in 10,000 tries. Non-negative garrote also did not work well and had 17 successes. The reason is the same: $n = 2$ is too small for asymptotic convergence to take effect. On the other hand, since RIVAL possesses the ability to adjust its weights for a fixed n, it performed well and was successful 9714 times in 10,000 tries. Ridge regression did not work well as expected and failed in every try for any finite λ. Fig. 2 shows the trajectories of the estimate β_2 of $\beta_2^* = 0$ for 1000 tries. In each try, λ varies from 0 (corresponds to the least squares estimate) to 10,000 (extremely high penalty on the magnitude of the estimate). Unless $\lambda = \infty$, ridge regression fails. LARS did not fare well either as expected because the two regressors are highly correlated. In 9998 of 10,000 simulations, LARS begins with a wrong regressor. Fig. 3 shows a path of Lasso solutions for Example 1 generated by LARS where it starts with the second regressor and in fact the second regressor does not contribute to the output. Even if LARS could estimate the model dimension correctly, it would pick the second regressor and not the first. In fact, it was shown in Leng, Lin, and Wahba (2006) that LARS and its modifications (like Lasso) fail to be consistent even for orthogonal designs. To this end, Example 2 is used to test how the three methods work in the orthogonal regressors case. 10,000 Monte Carlo simulations were generated with the same choices of λ's. The bottom table in Table 1 shows the results. Again, non-negative garrote and adaptive Lasso did not work because of a small n but RIVAL worked almost perfectly. This demonstrates the advantage of RIVAL, e.g., it achieves the set consistency for a smaller number n of the data points.

We comment that in the previous two examples, $n = 2$ is small and extreme. However, even with a modest or large n, a similar conclusion holds. This will be made clear in the next section.

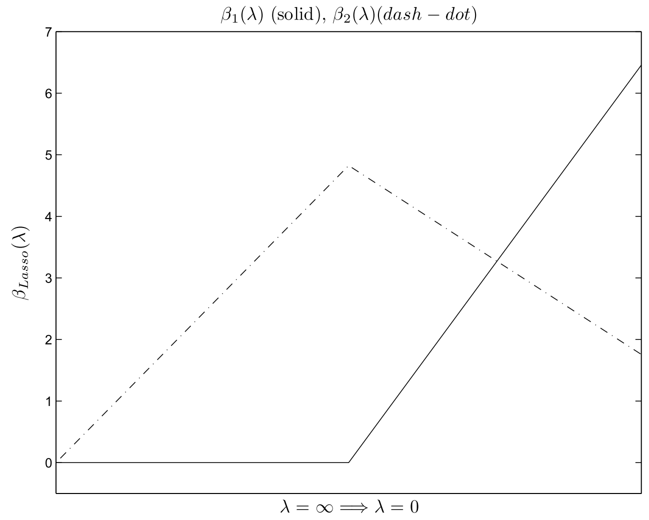


Fig. 3. A path of Lasso solutions by LARS.

5. Application to nuclear material detection

Detection of nuclear material is important for national security. The emission of gamma rays from a radioactive nuclear source is a random variable. For a given range of energies (keV or detector channels), the number of gamma ray counts registered by a detector per unit time and β^* units of source material follows a Poisson distribution (Killian & Hartwell, 2000),

$$\frac{(x\beta^*)^k}{k!}e^{-x\beta^*}$$

where x primarily depends on the radioactive decay of the source and the characteristics of the detector material. $\beta^* = 0$ indicates that the source is absent. In the presence of p radioactive sources including the background, it is reasonable to assume that the contribution of each individual one is statistically independent. Since the sum of independent Poisson random variables is still Poisson, the total gamma ray counts at the jth energy level or the jth channel is still Poisson, $\frac{(\sum_{j=1}^p x_{ij}\beta_j^*)^k}{k!}e^{-\sum_{j=1}^p x_{ij}\beta_j^*}$, $i = 1, \dots, n$, where n is the total number of detector channels and $(x_{1j}, x_{2j}, \dots, x_{nj})^T$ is the spectrum of the jth radioactive nuclear source. From the properties of the Poisson distribution, both the mean and variance of the gamma ray counts at the ith channel are $(x_{i1}\beta_1^* + x_{i2}\beta_2^* + \cdots + x_{ip}\beta_p^*)$. Define

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \dots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}, \qquad \beta^* = \begin{pmatrix} \beta_1^* \\ \vdots \\ \beta_p^* \end{pmatrix} \tag{5.10}$$

and the received gamma ray counts at channel i be $y(i)$. Then, we have

$$\begin{pmatrix} y(1) \\ \vdots \\ y(n) \end{pmatrix} = X\beta^* + v$$

where v is a random vector describing the difference between the actual received gamma ray counts and its statistical average. In the case of nuclear materials, where the aforementioned Poisson distribution has a relatively large mean, a Poisson distribution can be approximated by a Gaussian distribution. Therefore, it is reasonable to assume that v is Gaussian with zero mean. On the other hand, since the variance at each channel is different, each component of v does not have the same variance. To reduce the effect of non-homogenous variance at different channels, we normalize the equation by

$$\underbrace{\begin{pmatrix} \frac{1}{\sqrt{y(1)}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\sqrt{y(n)}} \end{pmatrix} \begin{pmatrix} y(1) \\ \vdots \\ y(n) \end{pmatrix}}_{y_n}$$

$$= \underbrace{\begin{pmatrix} \frac{1}{\sqrt{y(1)}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\sqrt{y(n)}} \end{pmatrix} X\,\beta^*}_{X_n}$$

$$+ \underbrace{\begin{pmatrix} \frac{1}{\sqrt{y(1)}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\sqrt{y(n)}} \end{pmatrix} v}_{v_n}$$

which is exactly (1.1).

The problem for nuclear material detection is that very often detectors cannot reasonably be expected to be close to the sources, the exposure time is short and there may be a wide range of materials between the source and the detector. In these cases, the detectors generally used have large volume (to increase sensitivity) and poor resolution. The signals from these detectors will be weak and difficult to separate from the background radiation or from the signatures of commonly used radioactive materials. In such cases, the traditional peak based detection methods fail to work (Jarman, Daly, Anderson, & Wahl, 2003; Killian & Hartwell, 2000).

Consider a semi-real experiment where a germanium type of detector with $n = 1024$ channels is used and the background is a real measurement which consists primarily of trace amounts of radiation from local sources and cosmic rays. Then, gamma ray counts of 11 isotopes as shown in Table 2 were synthetically generated according to nuclear physics posted on the web-site of the National Institute of Standards and Technology or equivalently by the handbook of Killian and Hartwell (2000). The background was considered as the 12th isotope in simulation. Recall that to detect if the contribution of an isotope is significant or not, t and/or f statistics could also be used. In fact, we did use t and f statistics and found the results were unsatisfactory. The reason is that the data point n is large but not large enough. This motivated us to use Lasso based algorithms and propose a new one.

In simulation, the coefficients β_{12}^* of the background and β_9^* of I131 are set to be 1 and the coefficient β_1^* for Pu239 is equal to 0.2. The reason is to hide Pu239 under the shadow of I131 to make the detection problem non-trivial. All other β_i^*'s are zero. The signal to noise (background) ratio is calculated by summing up the energy at all frequencies by varying α,

$$\text{SNR}(\alpha) = 10 \cdot \log \frac{\sum_{i=1}^{1024}\{\alpha(0.2\text{Pu}239(i) + \text{I}131(i))\}}{\sum_{i=1}^{1024}\text{Background}(i)}.$$

The detection problem is to find which β_i^* are non-zero or equivalently which isotopes are present. Fig. 4 shows the result of traditional peak based detection methods at SNR $= -10$ dB. The top diagram in Fig. 4 shows the peaks detected (circles) when

Table 2
Isotopes involved in testing (BG = background).

Isotope	1	2	3	4	5	6	7	8	9	10	11	12
	Pu239	Ga67	Cs137	U235	K40	Na22	Ba133	Ce139	I131	Co57	Co60	BG

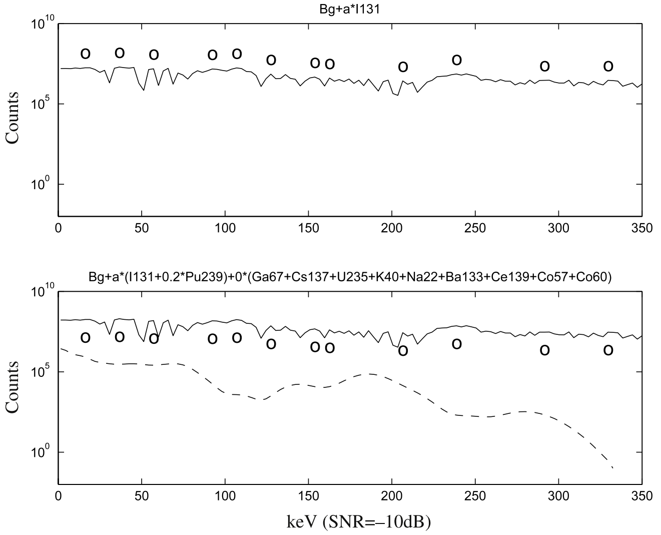


Fig. 4. Top: Peaks detected (circles) when Pu239 is absent. Bottom: Peaks detected (circles) with Pu239 present. The dashed line is Pu239. The peaks detected with and without Pu239 are the same.

Table 3
False positive and false negative rates of the RIVAL algorithm (AIC and BIC).

SNR (dB)	False positive rate	False negative rate
−10	0.0000	0.0000
−20	0.0005	0.0013
−30	0.0010	0.0015
−32	0.0017	0.0065

Pu239 is absent, and the bottom diagram shows the peaks detected (circles) with Pu239 present. The peaks detected with and without Pu239 remain the same. This implies the traditional methods completely fail to find Pu239 at SNR $= -10$ dB and some new detection methods have to be developed.

Since β_i^* cannot be negative, and for our problem $n = 1024$ is relatively large but fixed, we propose that RIVAL be applied here. The results of the simulation with $q(k) \equiv 1$ are shown in Table 3 for both AIC and BIC. In fact, the results of AIC and BIC are the same. Numerically, $\beta_i(k)$ is considered to be zero if $\beta_i(k) < 0.00009$ and the corresponding ith isotope is considered to be absent. The results are the averages of 500 Monte Carlo runs. The definitions of the false negative rate and the false positive rate are

False negative: isotope is present and the algorithm fails to find it,
False positive: isotope is absent and the algorithm falsely identifies it.

For comparison, the results of the standard Lasso, non-negative garrote, and adaptive Lasso are shown in Table 4.

Clearly, RIVAL works well even in the very low SNR $= -30$ dB level. On the other hand, the traditional peak based methods even fail at SNR $= -10$ dB, and also the standard Lasso, non-negative garrote, and adaptive Lasso do not work well even at -10 dB. For adaptive Lasso and non-negative garrote, the results corresponding to two choices of λ are presented. One is according to its asymptotic value for convergence when $n \to \infty$ and the other is based on AIC/BIC for the fixed $n = 1024$, whichever provides a better result.

Table 4
Top: false positive rate of the standard Lasso at −10 dB. Bottom: error rates for adaptive Lasso (AL) and the non-negative garrote (NNG) with regularization parameters chosen according to theoretical values. Numbers in parentheses are error rates when AIC/BIC chooses the regularization parameters.

	False positive rate
Lasso (AIC), −10 dB	0.6352
Lasso (BIC), −10 dB	0.7341

Method	SNR (dB)	False positive rate	False negative rate
AL	−10	0.0624 (0.0164)	0.0000 (0.0000)
	−20	0.0771 (0.0482)	0.1387 (0.1353)
NNG	−10	0.0000 (0.0893)	0.6667 (0.0000)
	−20	0.0000 (0.0649)	0.6667 (0.136)

6. Concluding remarks

Positive Lasso is a constrained optimization method to solve model selection problems when the statisticians or scientists know *a priori* that the unknown parameters are non-negative. In this paper, the conditions for consistency of both the positive Lasso and adaptive positive Lasso have been analyzed by extending the irrepresentable conditions (Knight & Fu, 2000) of the standard Lasso. Further, a new algorithm called RIVAL, which is related to positive Lasso, has been developed and is particularly useful for applications where the number of data is large but fixed. Theoretically, convergence of RIVAL is established, and it is shown that RIVAL can be more reliable in model selection than the positive lasso method upon which it is based. RIVAL is applied to weak nuclear signal detection to confirm that it is a valuable tool in practice.

Acknowledgment

We would like to thank Dr. Kun Chen of Kansas State University for reading the manuscript and providing useful suggestions.

Appendix A

Proof of Theorem 4.1. Minimizing

$$J_1 = (y_n - X_n\beta)^T(y_n - X_n\beta) + \lambda(n)\sum_{j=1}^{p} w_j\beta_j, \quad \beta_j \geq 0$$

is equivalent to minimizing

$$J_2 = -2n\beta^{*T}C_n\beta - 2v_n^T X_n\beta + n\beta^T C_n\beta + \lambda(n)\sum_{j=1}^{p} w_j\beta_j$$

which is equivalent to minimizing

$$J_3 = -2\frac{n}{\lambda(n)}\beta^{*T}\begin{pmatrix}\xi_1 & 0 & \cdots & 0\\ 0 & \xi_2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & \cdots & 0 & \xi_p\end{pmatrix}\beta + \frac{2v_n^T X_n}{\lambda(n)}\beta^*$$
$$-\frac{2v_n^T X_n}{\lambda(n)}\beta + \frac{n}{\lambda(n)}\beta^T\begin{pmatrix}\xi_1 & 0 & \cdots & 0\\ 0 & \xi_2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & \cdots & 0 & \xi_p\end{pmatrix}\beta$$

$$+\frac{n}{\lambda(n)}\beta^{*T}\begin{pmatrix}\xi_1 & 0 & \cdots & 0\\ 0 & \xi_2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & \cdots & 0 & \xi_p\end{pmatrix}\beta^* + \sum_{j=1}^{p} w_j\beta_j$$

$$= a(\beta^*-\beta)^T\begin{pmatrix}\xi_1 & 0 & \cdots & 0\\ 0 & \xi_2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & \cdots & 0 & \xi_p\end{pmatrix}(\beta^*-\beta)$$

$$+\hat{d}(\beta^*-\beta) + \sum_{j=1}^{p} w_j\beta_j$$

where $a = \frac{n}{\lambda(n)}, \hat{d} = \frac{2v_n^T X_n}{\lambda(n)} = \frac{2v_n^T X_n}{\sqrt{n}}\frac{\sqrt{n}}{\lambda(n)} = (\hat{d}_1, \hat{d}_2, \dots, \hat{d}_p)$.

Therefore, minimizing J_3 is achieved by minimizing J_4 and J_5 separately, where

$$J_4 = \sum_{j=1}^{d}\left(a\beta_j^2 - 2ab_j\beta_j + \frac{w_j}{\xi_j}\beta_j\right), \quad b_j = \beta_j^* + \frac{1}{2a\xi_j}\hat{d}_j$$

where $\beta_j \geq 0$, and

$$J_5 = \sum_{j=d+1}^{p}\left[a\beta_j^2 - c_j\beta_j + \frac{w_j}{\xi_j}\beta_j\right], \quad \beta_j \geq 0.$$

Since $a = \frac{n}{\lambda(n)}, \hat{d} = \frac{2v_n^T X_n}{\sqrt{n}}\frac{\sqrt{n}}{\lambda(n)}, c_j = \frac{\hat{d}}{\xi_j}, b_j = \beta_j^* + \frac{1}{2a\xi_j}\hat{d}_j$, there always exists a constant $\delta > 0$ for large enough n so that the conditions of Lemmas A.1 and A.2 are simultaneously satisfied

$$a \geq \frac{(c+\delta)^2}{8}, \qquad b > \delta = \frac{b-\sqrt{b^2-2/a}}{2} > 0,$$
$$1 \leq 2ab - \frac{1}{b-\delta}.$$

Then, the conclusions follow from these two lemmas. This concludes the proof. □

Lemma A.1 (*Zero Identification*)**.** *Consider a sequence of scalar minimization problems*

$$J = \min_{\beta(k)\geq 0}\{a\beta^2(k) - c\beta(k) + w(k)\beta(k)\}, \quad a > 0,\ w(1) = 1.$$

Assume there exists a constant $\delta > 0$ such that

$$a \geq \frac{(c+\delta)^2}{8}.$$

Let $\beta(k)$ be the solution of J for the given $w(k)$. Construct $\beta(k+1)$ as follows. If $\beta(k) = 0$, set $\beta(k+i) = 0$ for $i \geq 0$ and stop the algorithm. If $\beta(k) > 0$, let $\bar{w}(k) = 1/\beta(k)$ and

$$w(k+1) = q(k)\bar{w}(k) + (1-q(k))w(k),$$

where $0 \leq q(k) \leq 1$ is a sequence satisfying $\sum_{k=1}^{\infty} q(k) = \infty$. Denote $\beta(k+1)$ the solution of J for given $w(k+1)$. Then, there exists a finite integer $k_0 \geq 1$ such that the sequence generated above satisfies

$$\beta(k) = 0, \quad \forall k \geq k_0.$$

Proof. If $1 = w(1) \geq c$, the minimum $\beta(1) = 0$ and this implies $k_0 = 1$ and $\beta(k) = 0, k \geq k_0 = 1$. If $c > w(1)$, the minimum $\beta(1)$ is achieved at some $\beta(k) > 0$. The first order necessary condition

$$\frac{\partial J}{\partial \beta} = 2a\beta - (c-w) = 0$$

implies

$$\beta(1) = \frac{c-w(1)}{2a} > 0 \rightarrow \bar{w}(1) = \frac{2a}{c-w(1)}.$$

From the hypothesis, we have

$$2a - \frac{(c-\delta)^2}{4} \geq \delta c \rightarrow \left(w(1) - \frac{(c-\delta)}{2}\right)^2 - \frac{(c-\delta)^2}{4} + 2a \geq \delta c.$$

Thus,

$$\frac{2a - cw(1) + w^2(1)}{c-w(1)} \geq \delta \quad \text{or} \quad \bar{w}(1) \geq w(1) + \delta$$

and

$$\begin{aligned} w(2) &= q(1)\bar{w}(1) + (1-q(1))w(1)\\ &\geq q(1)w(1) + q(1)\delta + (1-q(1))w(1)\\ &= w(1) + q(1)\delta. \end{aligned}$$

By induction, if $w(k) < c$, $w(k+1) \geq w(k) + q(k)\delta \geq w(1) + \sum_{i=1}^{k} q(i)\cdot\delta$ or equivalently, there is an integer $k_0 > 0$ such that $w(k_0) > c$ and the corresponding solution of J is $\beta(k_0) = 0$ and $\beta(k_0+i) = 0, i \geq 0$. This completes the proof. □

Lemma A.2 (*Non-Zero Identification*)**.** *Consider a sequence of scalar minimization problems*

$$J = \min_{\beta(k)\geq 0}\{a\beta^2(k) - 2ab\beta(k) + w(k)\beta(k)\},$$
$$a > 0,\ b > 0,\ w(1) = 1.$$

Assume there exists a constant $\delta > 0$ such that

$$b > \delta = \frac{b-\sqrt{b^2-2/a}}{2} > 0$$

and

$$1 = w(1) \leq 2ab - \frac{1}{b-\delta}.$$

Let $\beta(k)$ be the solution of J for the given $w(k)$. Construct $\beta(k+1)$ as follows. If $\beta(k) = 0$, set $\beta(k+i) = 0$ for $i \geq 0$ and stop the algorithm. If $\beta(k) > 0$, let $\bar{w}(k) = 1/\beta(k)$ and

$$w(k+1) = q(k)\bar{w}(k) + (1-q(k))w(k),$$

where $0 \leq q(k) \leq 1$ is a sequence satisfying $\sum_{k=1}^{\infty} q(k) = \infty$. Denote $\beta(k+1)$ the solution of J for given $w(k+1)$. Then, the sequence $\beta(k)$ is uniformly bounded from above and from below

$$0 < \eta_1 \leq \beta(k) \leq \eta_2 < \infty, \quad \forall k.$$

Proof. The idea of the proof is to show that the $w(k)$'s are bounded. $b > 0$ and $w(1) < 2ab$ imply that the minimum is achieved at some $\beta(1) > 0$. The first order necessary condition

$$\frac{\partial J}{\partial \beta} = 2a\beta - 2ab + w = 0$$

implies

$$\beta(1) = b - \frac{w(1)}{2a} > 0 \rightarrow \bar{w}(1) = \frac{1}{b - \frac{w(1)}{2a}} > 0.$$

Further, $w(1) \leq 2ab - \frac{1}{b-\delta}$ leads to

$$(b-\delta)w(1) \leq 2ab(b-\delta) - 1 \rightarrow \frac{\bar{w}(1)}{2a} - b = \frac{1}{2ab - w(1)} - b \leq -\delta$$

or

$$\bar{w}(1) \leq 2ab - 2a\delta.$$

On the other hand, from the definition of δ, it is easily verified that

$$\begin{aligned}\delta^2 - \delta b + 1/(2a) = 0 \to 2a\delta \\ = 1/(b - \delta).\end{aligned}$$

Hence,

$$0 < w(1) \leq 2ab - \frac{1}{b - \delta} \to 0 < \bar{w}(1) \leq 2ab - \frac{1}{b - \delta}$$

and this implies

$$0 < w(2) = q(1)\bar{w}(1) + (1 - q(1))w(1) \leq 2ab - \frac{1}{b - \delta}.$$

By induction, we have for all $k \geq 1$,

$$0 < w(k) \leq 2ab - \frac{1}{b - \delta}$$

and

$$\beta(k) = b - \frac{w(k)}{2a} \geq \delta > 0.$$

This shows that $\beta(k)$ is bounded away from zero. The upper bound can be derived easily,

$$\beta(k) \leq b + |w(k)/(2a)| \leq 2b - \delta < \infty, \quad \forall k.$$

This finishes the proof. □

References

Bai, E., Chan, K., Eichinger, W., & Kump, P. (2011). Detection of radionuclides from weak and poorly resolved spectra using Lasso and subsampling techniques. *Radiation Measurements*, *46*, 1138–1146.

Billings, S. A., Chen, S., & Korenberg, M. J. (1989). Identification of MIMO non-linear systems using a forward-regression orthogonal estimator. *International Journal of Control*, *49*, 2157–2189.

Breiman, L. (1995). Better subset regression using the nonnegative garrote. *Technometrics*, *37*, 373–384.

Chang, Y. M., Hsu, N. J., & Huang, H. C. (2010). Semiparametric estimation of selection for nonstationary spatial covariance functions. *Journal of Computational and Graphical Statistics*, *19*, 117–139.

Chen, S., Billings, S. A., & Luo, W. (1989). Orthogonal least squares methods and their application to non-linear system identification. *International Journal of Control*, *50*, 1873–1896.

Chen, S., & Wigger, J. (1995). Fast orthogonal least squares algorithm for efficient subset model selection. *IEEE Transactions on Signal Processing*, *43*, 1713–1715.

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

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, *96*, 1348–1360.

Frank, I. E., & Friedman, J. H. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, *35*, 109–148.

Frank, L. E., & Heiser, W. (2008). Feature selection in feature network models: finding predictive subsets of features with the positive lasso. *British Journal of Mathematical and Statistical Psychology*, *61*, 1–27.

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, *12*, 55–67.

Jarman, K., Daly, D., Anderson, K., & Wahl, K. (2003). A new approach to automated peak detection. *Chemometrics and Intelligent Laboratory Systems*, *69*, 61–76.

Killian, E., & Hartwell, J. (2000). PCGAP: users guide and algorithm description. Idaho national engineering and environmental laboratory bechtel BWXT idaho, LLC, INEEL/EXT-2000-00908.

Knight, K., & Fu, W. (2000). Asymptotic for Lasso-type estimators. *The Annals of Statistics*, *28*, 1356–1378.

Kukreja, S. L. (2009). Application of a least absolute shrinkage and selection operator to aeroelastic flight test data. *International Journal of Control*, *82*(12), 2284–2292.

Kukreja, S. L. (2005). A suboptimal bootstrap method for structure detection of non-linear output-error models with application to human ankle dynamics. *International Journal of Control*, *78*(12), 937–948.

Kukreja, S. L., Lofberg, J., & Brenner, M. J. (2006). A least absolute shrinkage and selection operator for nonlinear system identification. In *Proc of 14th IFAC Symposium on System Identification*, Newcastle, Australia, pp. 814–819.

Leng, C., Lin, Y., & Wahba, G. (2006). A note on the lasso and related procedures in model selection. *Statistica Sinica*, *16*, 1273–1284.

Li, K., Peng, J., & Bai, E. W. (2006). A two stage algorithm for identification of nonlinear system. *Automatica*, *42*, 1189–1197.

Lind, I., & Ljung, L. (2005). Regressor selection with the analysis of variance method. *Automatica*, *41*(4), 693–700.

Reiss, D., Facciotti, M., & Baliga, N. (2008). Model based deconvolution of genome-wide DNA binding. *Bioinformatics*, *3*, 396–403.

Shao, J. (1997). An asymptotic theory for linear model selection. *Statistica Sinica*, *7*, 221–264.

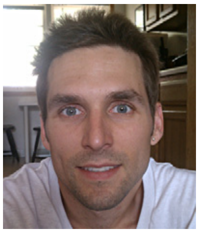
Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Series B*, *58*, 267–288.

Young, P. C., McKenna, P., & Bruun, J. (2001). Identification of nonlinear stochastic systems by state dependent parameter estimation. *International Journal of Control*, *74*(18), 1837–1957.

Yuan, M., & Lin, Y. (2007). On the non-negative garrote estimator. *Journal of the Royal Statistical Society. Series B*, *69*, 143–161.

Zhao, P., & Yu, B. (2006). On model selection consistency of Lasso. *Journal of Machine Learning Research*, *7*, 2541–2563.

Zou, H. (2006). The adaptive Lasso and its oracle properties. *Journal of the American Statistical Association*, *101*, 1418–1429.



Paul Kump received his M.S. degree in electrical engineering from the University of Iowa in 2008. He is currently working towards the Ph.D. degree in electrical engineering from the University of Iowa with applications in model selection and nuclear material detection. His research interests include signal processing, which is motivated by his love for music.

Er-Wei Bai was educated in Fudan University, and Shanghai Jiaotong University, both in Shanghai, China, and the University of California at Berkeley. Dr. Bai is Professor of Electrical and Computer Engineering and Radiology at the University of Iowa where he teaches and conducts research in identification, control, signal processing and their applications in engineering and life sciences. He also is the World Class Research Professor (in System Identification), School of Electronics, Electrical Engineering and Computer Science, Queen's University, Belfast, UK.

Dr. Bai is an IEEE Fellow and a recipient of the President's Award for Teaching Excellence and the Board of Regents Award for Faculty Excellence.

Kung-sik Chan received his education from the Chinese University of Hong Kong, and Princeton University. Dr. Chan is Professor of Statistics at the University of Iowa where he teaches and conducts research in chaos and time series analysis.

Dr. Chan is a Fellow of the American Statistical Association and the Institute of Mathematical Statistics, and an elected member of the International Statistical Institute.

Bill Eichinger graduated from the US Military Academy, West Point, the Air Force Institute of Technology and the University of California at Davis. Dr. Eichinger is a Professor of Civil and Environmental Engineering at the University of Iowa where he teaches and conducts research in hydrology and surface–atmosphere interactions. He is also a Professor at the University of Nova Gorica, Slovenia.

Dr. Eichinger is a recipient of the M.L. Huit Award for Teaching Excellence, the Allen Prize from the Optical Society of America, and an R&D 100 award.

Kang Li received the Ph.D. degree on control theory and applications from Shanghai Jiaotong University, China, in 1995. He is currently a Professor of Intelligent Systems and Control at the School of Electronics, Electrical Engineering and Computer Science, Queen's University, Belfast, UK, where he teaches and conducts research in control engineering. His research interests covers nonlinear system modeling, identification and control, and bio-inspired computational intelligence, with recent applications to power systems and polymer extrusion. He has also extended his research to bioinformatics and systems biology with applications on food safety and healthcare. He has published over 160 papers in the above areas. Dr. Li serves on the editorial board as an associate editor or member of the editorial board for Neurocomputing, the Transactions of the Institute of Measurement & Control, Cognitive Computation and International Journal of Modelling, Identification and Control. Dr. Li is a senior member of the IEEE, a Fellow of the Higher Education Academy, UK, and a member of the IFAC Technical Committee on Computational Intelligence in Control.