Statistica Sinica **16**(2006), 471-494

BOOSTING FOR HIGH-MULTIVARIATE RESPONSES IN HIGH-DIMENSIONAL LINEAR REGRESSION

Roman Werner Lutz and Peter Bühlmann

ETH Zürich, Switzerland

Abstract: We propose a boosting method, multivariate L_2Boosting, for multivariate linear regression based on some squared error loss for multivariate data. It can be applied to multivariate linear regression with continuous responses and to vector autoregressive time series. We prove, for i.i.d. as well as time series data, that multivariate L_2Boosting can consistently recover sparse high-dimensional multivariate linear functions, even when the number of predictor variables p_n and the dimension of the response q_n grow almost exponentially with sample size n, $p_n = q_n = O(\exp(Cn^{1-\xi}))$ $(0 < \xi < 1,\ 0 < C < \infty)$, but the ℓ_1-norm of the true underlying function is finite. Our theory seems to be among the first to address the issue of large dimension of the response variable; the relevance of such settings is briefly outlined. We also identify empirically some cases where our multivariate L_2Boosting is better than multiple application of univariate methods to single response components, thus demonstrating that the multivariate approach can be very useful.

Key words and phrases: High-multivariate high-dimensional linear regression, L_2Boosting, vector AR time series.

1. Introduction

Boosting was originally proposed as an ensemble scheme for classification (AdaBoost by Freund and Schapire (1996)), and has attracted a lot of attention both in the machine learning and statistics literature, mainly because of its success as an excellent prediction method in numerous examples. The pioneering work by Breiman (1998, 1999) demonstrated that the AdaBoost ensemble method can be represented as a gradient descent approximation in function space, see also Friedman, Hastie and Tibshirani (2000). This has opened new possibilities for better understanding, and new versions of boosting. In particular, such gradient descent methods can be applied to different loss functions, each yielding another boosting algorithm. L_2Boosting which uses the squared error loss (L_2-loss) has been demonstrated to be a powerful method for univariate regression (Friedman (2001), Bühlmann and Yu (2003) and Bühlmann (2004)).

We propose here a boosting method with some squared error loss (Gaussian negative log-likelihood) for multivariate data, called multivariate L_2Boosting. We restrict ourselves to linear models (linear basis expansions). They can be very high-dimensional in terms of the response or predictor dimension, and we allow for seemingly unrelated regressions (SUR; Zellner (1962, 1963)) where each response may have another design matrix (other predictor variables). The SUR model is more general than the multivariate setting where each covariate has an influence on all response variables. Our multivariate L_2Boosting takes potential correlations among the components of the multivariate error-noise into consideration, that is, we account for the fact that the responses are still exhibiting conditional dependence given all the predictor variables. We prove that our boosting method is able to consistently recover sparse high-dimensional multivariate functions, even when the number of predictor variables p_n and the dimension of the response q_n grow almost exponentially with sample size n, i.e., $p_n = q_n = O(\exp(Cn^{1-\xi}))$ $(0 < \xi < 1,\ 0 < C < \infty)$. The mathematical arguments extend the analysis for boosting for high-dimensional univariate regression (Bühlmann (2004)). Our theory seems to be among the first for the setting of large dimension of the response (for its practical relevance, see below).

We also demonstrate the use of our multivariate L_2Boosting for multivariate, q_n-dimensional time series $\{\mathbf{x}_{(t)}\}_{t\in\{1,\ldots,n\}}$, where q_n can grow as fast as any polynomial in the sample size n. We prove a consistency result for stationary, linear processes which are representable as a sparse vector autoregressive model of order ∞.

From a theoretical perspective it is interesting how far we can go with dimensionality when the true underlying structure is sparse. From a practical point of view, there are many applications nowadays with large predictor dimension p, notably a broad variety of data mining problems belong to this setting. There are also some applications where q is very large. We mention multi-category classification with a huge number of categories − in Kriegel, Kroger, Pryakhin and Schubert (2004), the categories are subsets of functions from gene ontology (see also Remark 1 in Section 4). Another application is briefly outlined in Section 4.1. In the context of time series, some graphical modelling for many stochastic processes falls into our setting of q-dimensional linear time series, e.g the partial correlation graph (cf., Dahlhaus and Eichler (2003)).

Besides presenting some theory, we also empirically identify some cases where our multivariate L_2Boosting is better than methods based on individual estimation − we compare with individual univariate L_2Boosting and with another L_2Boosting method in a multivariate regression model where every predictor variable either influences all or none of the response components. Some data sets are analyzed as well.

2. Multivariate Linear Regression

We consider the multivariate linear regression model with n observations of a q-dimensional response and a p-dimensional predictor (for more detailed information, see for example Timm (2002)). In matrix notation,

$$\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{E}, \tag{2.1}$$

with $\mathbf{Y} \in \mathbb{R}^{n\times q}$, $\mathbf{X} \in \mathbb{R}^{n\times p}$, $\mathbf{B} \in \mathbb{R}^{p\times q}$ and $\mathbf{E} \in \mathbb{R}^{n\times q}$. We denote by $\mathbf{y}_{(i)}$ the response of the ith sample point (row-vector of $\mathbf{Y}$), and by $\mathbf{y}_k$ the kth response-variable for all sample points (column-vector of $\mathbf{Y}$). For each $\mathbf{y}_k$ ($k = 1, \ldots, q$) we have a univariate regression model with the predictor matrix $\mathbf{X}$ and the coefficient vector $\mathbf{b}_k$. For the row-vectors $\mathbf{e}_{(i)}$ ($i = 1, \ldots, n$) of the error matrix, we assume $\mathbf{e}_{(i)}$ i.i.d., $E[\mathbf{e}_{(i)}] = \mathbf{0}$ and $\mathrm{Cov}(\mathbf{e}_{(i)}) = \mathbf{\Sigma}$. Additionally, we assume w.l.o.g. that all covariates and responses are centered to have mean zero, so we need not worry about intercepts.

The ordinary least squares estimator (OLS) of $\mathbf{B}$ is given by, assuming $\mathbf{X}$ is of full rank p,

$$\hat{\mathbf{B}}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}, \tag{2.2}$$

and is nothing else than the OLS's of the q univariate regressions. In particular, it is independent of $\mathbf{\Sigma}$.

To test whether a covariate has a significant influence on the multivariate response we can use Wilk's Λ, which is derived from the likelihood ratio test. For an overall test with null-hypothesis $H_0 : \mathbf{B} = \mathbf{0}$ we compare the empirical covariance matrix of the residuals to the one from the responses:

$$\Lambda = \frac{|(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}_{OLS})^T(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}_{OLS})|}{|\mathbf{Y}^T\mathbf{Y}|},$$

where $|\cdot|$ denotes the determinant of a matrix. We reject the null hypothesis H_0 if Λ is smaller than a critical value.

2.1. Forward stepwise variable selection

As with univariate regression, we can define a multivariate forward stepwise variable selection algorithm in a straightforward manner: start with the empty model and add in each step the most significant covariate according to Wilk's Λ. Notice that in each step the entries of a whole row $\mathbf{b}_{(j)}$ of $\mathbf{B}$ are changed from zero to non-zero by using OLS on the reduced space of all included covariates. Therefore, this approach is not suited for the SUR model where a covariate may only have an effect on some but not all components of the response.

3. L_2 Boosting for Multivariate Linear Regression

For constructing a boosting algorithm, we define a loss function and a base procedure (simple fitting method). The latter is usually called a “weak learner” in the machine learning community: it is an estimator which is repeatedly used in boosting.

3.1. The loss function

Regarding the loss function, we use the negative Gaussian log-likelihood as a starting point:

$$-l(\mathbf{B},\boldsymbol{\Sigma}) = -\log((2\Pi)^{\frac{nq}{2}}|\boldsymbol{\Sigma}|^{\frac{n}{2}}) + \frac{1}{2}\sum_{i=1}^{n}(\mathbf{y}_{(i)}^T - \mathbf{x}_{(i)}^T\mathbf{B})\boldsymbol{\Sigma}^{-1}(\mathbf{y}_{(i)}^T - \mathbf{x}_{(i)}^T\mathbf{B})^T.$$

The maximum likelihood estimator of $\mathbf{B}$ coincides with the OLS solution in (2.2) and is therefore independent of $\boldsymbol{\Sigma}$. The covariance matrix $\boldsymbol{\Sigma}$ only becomes relevant in the seemingly unrelated regressions (SUR) model when there are covariates that influence only a few components of the response.

Because $\boldsymbol{\Sigma}$ is usually unknown, we use the loss function

$$L(\mathbf{B}) = \frac{1}{2}\sum_{i=1}^{n}(\mathbf{y}_{(i)}^T - \mathbf{x}_{(i)}^T\mathbf{B})\boldsymbol{\Gamma}^{-1}(\mathbf{y}_{(i)}^T - \mathbf{x}_{(i)}^T\mathbf{B})^T, \qquad (3.1)$$

where $\boldsymbol{\Gamma}$ is the implementing covariance matrix. We may use for it an estimate of $\boldsymbol{\Sigma}$ (e.g., from another model-fit such as univariate boosting for each response separately) or we can choose something simpler, e.g., $\boldsymbol{\Gamma} = \mathbf{I}$ (in particular if q is large, see also Remark 2). The choice for $\boldsymbol{\Gamma}$ will show up again in our Theorem 1 in Section 4 (and Theorem 2 in Section 5); there it becomes clear that $\boldsymbol{\Gamma} = \mathbf{I}$ can be a reasonable choice.

3.2. The componentwise linear least squares base procedure

We specify the base procedure that will be repeatedly used in boosting. Given is the design matrix $\mathbf{X}$ and a pseudo-response matrix $\mathbf{R} \in \mathbb{R}^{n\times q}$ (which is not necessarily equal to $\mathbf{Y}$).

We focus exclusively on what we call the componentwise linear least squares base learner. It fits the linear least squares regression with one selected covariate (column of $\mathbf{X}$) and one selected pseudo-response (column of $\mathbf{R}$) so that the loss function in (3.1), with $\mathbf{R}$ instead of $\mathbf{Y}$, is reduced most. Thus, the base procedure fits one selected matrix element of $\mathbf{B}$:

$$\begin{aligned}(\hat{s}\hat{t}) &= \underset{1\le j\le p, 1\le k\le q}{\operatorname{argmin}} \{L(\mathbf{B}); B_{jk} = \hat{\beta}_{jk}, B_{uv} = 0 \ (uv \neq jk)\} \\ &= \underset{1\le j\le p, 1\le k\le q}{\operatorname{argmax}} \frac{\left(\sum_{v=1}^{q}\mathbf{r}_v^T\mathbf{x}_j\Gamma_{vk}^{-1}\right)^2}{\mathbf{x}_j^T\mathbf{x}_j\Gamma_{kk}^{-1}},\end{aligned}$$

$$\hat{\beta}_{jk} = \frac{\sum_{v=1}^{q} \mathbf{r}_v^T \mathbf{x}_j \Gamma_{vk}^{-1}}{\mathbf{x}_j^T \mathbf{x}_j \Gamma_{kk}^{-1}},$$
$$\hat{B}_{\hat{s}\hat{t}} = \hat{\beta}_{\hat{s}\hat{t}}, \quad \hat{B}_{jk} = 0, \ (jk) \neq (\hat{s}\hat{t}). \tag{3.2}$$

Corresponding to the parameter estimate, there is a function estimate $\hat{\mathbf{g}}(\cdot) : \mathbb{R}^p \to \mathbb{R}^q$ defined as follows. For $\mathbf{x} = (x_1, \ldots, x_p)$,

$$(\hat{\mathbf{g}})_\ell(\mathbf{x}) = \begin{cases} \hat{\beta}_{\hat{s}\hat{t}} x_{\hat{s}} & \text{if } \ell = \hat{t}, \\ 0 & \text{if } \ell \neq \hat{t}, \end{cases} \quad \ell = 1, \ldots, q.$$

From (3.2) we see that the coefficient $\hat{\beta}_{jk}$ is not only influenced by the kth response but also by other response-components, depending on the partial correlations of the errors (via $\mathbf{\Gamma}^{-1}$ if $\mathbf{\Gamma}$ is a reasonable estimate of $\mathbf{\Sigma}$) and by the correlations of the other response-components with the jth covariate (i.e., $\mathbf{r}_v^T \mathbf{x}_j$).

3.3. The boosting algorithm

The base learner is fitted many times to different pseudo-responses $\mathbf{R}$ and the function estimates are added up as described by the algorithm below. We build the multivariate regression function $\hat{\mathbf{f}} : \mathbb{R}^p \to \mathbb{R}^q$ step by step, where $\hat{\mathbf{f}}(\mathbf{x}) = \hat{\mathbf{B}}^T \mathbf{x}$.

Multivariate L_2 Boosting with componentwise linear least squares

Step 1 (initialization): $\hat{f}_k^{(0)}(\cdot) \equiv 0, \ k = 1, \ldots, q$. Set $m = 1$.
Step 2: Compute the current residuals $\mathbf{r}_{(i)}^{(m)} = \mathbf{y}_{(i)} - \hat{\mathbf{f}}^{(m-1)}(\mathbf{x}_{(i)})$ $(i = 1, \ldots, n)$ and fit the base learner to them as in (3.2). The fit is denoted by $\hat{\mathbf{g}}^{(m)}(\cdot)$. Update $\hat{\mathbf{f}}^{(m)}(\cdot) = \hat{\mathbf{f}}^{(m-1)}(\cdot) + \hat{\mathbf{g}}^{(m)}(\cdot)$.
Step 3 (iteration): Increase the iteration index m by one and go back to Step 2 until a stopping iteration m_{stop} is met.

Multivariate L_2Boosting is thus iteratively fitting of residuals where, in each step, we change only one entry of $\mathbf{B}$. Also, every iteration m corresponds to an estimate $\hat{\mathbf{B}}^{(m)}$ with $\hat{\mathbf{f}}^{(m)}(\mathbf{x}) = (\hat{\mathbf{B}}^{(m)})^T \mathbf{x}$. The estimate $\hat{\mathbf{f}}^{(m_{stop})}(\cdot)$ is an estimator of the multivariate regression function $E[\mathbf{y}|\mathbf{x} = \cdot]$.

It is often better to use some shrinkage in *Step* 2; this has been first recognized by Friedman (2001), and there are also some supporting theoretical arguments for it (Efron, Hastie, Johnstone and Tibshirani (2004), Bühlmann and Yu (2005)). We modify *Step* 2 to: $\hat{\mathbf{f}}^{(m)}(\cdot) = \hat{\mathbf{f}}^{(m-1)}(\cdot) + \nu \cdot \hat{\mathbf{g}}^{(m)}(\cdot)$, with $\nu < 1$, for example $\nu = 0.1$. We then need more iterations but often achieve better out-of-sample predictions. The boosting algorithm does depend on ν, but its choice is surprisingly insensitive as long as it is taken to be "small". On the other hand, the number of boosting iterations m_{stop} is a much more crucial tuning parameter.

The computational complexity of the multivariate L_2Boosting algorithm for m iterations is $O(npqm)$ if $\mathbf{\Gamma}$ is diagonal and $O(npq^2m)$ for arbitrary $\mathbf{\Gamma}$.

3.4. Stopping the boosting iterations with the corrected AIC

The number of iterations m_{stop} can be estimated by cross validation, a separate validation set or by an internal AIC criterion. We pursue the latter because of its computational attractiveness.

First we recall the definition of AIC for the multivariate linear regression model. For $d \leq p$ covariates in a sub-model M_d,

$$AIC(M_d) = \log(|\hat{\mathbf{\Sigma}}(M_d)|) + \frac{2qd}{n},$$

where $\hat{\mathbf{\Sigma}}(M_d)$ is the MLE of the error covariance-matrix. Note that we have a total of $q \cdot d$ parameters. In small samples, the corrected AIC (Hurvich and Tsai (1989) and Bedrick and Tsai (1994)) is often a better model selection tool:

$$AIC_c(M_d) = \log(|\hat{\mathbf{\Sigma}}(M_d)|) + \frac{q(n+d)}{n-d-q-1}.$$

To apply AIC or AIC_c for boosting we have to determine the number of parameters or degrees of freedom of boosting as a function of the number of iterations. Clearly, the degrees of freedom of boosting increase as the number of iterations grow, but this increase is heavily sub-linear (Bühlmann and Yu (2003)).

We first consider the hat-operator of the base learner in (3.2), mapping $\mathbf{Y}$ to $\hat{\mathbf{Y}} = \mathbf{X}\hat{\mathbf{B}}$. After having selected the jth predictor and kth component of the response, fitting is a linear operation which can be represented by a hat-matrix. In the multivariate case, we stack the q responses $\mathbf{y}_1, \ldots, \mathbf{y}_q$ end-to-end in a vector of length nq (written as $vec(\mathbf{Y})$). The hat-matrix is then of dimension $nq \times nq$ and, with the jth predictor and the kth response selected in the base learner, it is of the form

$$\mathbf{H}^{(jk)} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \vdots & \vdots & & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{H}^j \frac{\Gamma^{-1}_{k1}}{\Gamma^{-1}_{kk}} & \mathbf{H}^j \frac{\Gamma^{-1}_{k2}}{\Gamma^{-1}_{kk}} & \ldots & \mathbf{H}^j \frac{\Gamma^{-1}_{kq}}{\Gamma^{-1}_{kk}} \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \vdots & \vdots & & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \end{pmatrix} \longleftarrow k\text{th row},$$

where each entry is a $n \times n$ matrix, the non-zero matrix-entries are at row k, and $\mathbf{H}^j = \mathbf{x}_j\mathbf{x}_j^T/\mathbf{x}_j^T\mathbf{x}_j$ is the hat-matrix of the univariate componentwise linear learner using the jth predictor variable.

Due to the nature of iterative fitting of residuals, the hat-matrix of multivariate L_2Boosting after m iterations is then (c.f., Bühlmann and Yu (2003) and Bühlmann (2004))

$$\mathbf{K}_m = \mathbf{I} - (\mathbf{I} - \nu\mathbf{H}^{(\hat{s}_m\hat{t}_m)})(\mathbf{I} - \nu\mathbf{H}^{(\hat{s}_{m-1}\hat{t}_{m-1})})\dots(\mathbf{I} - \nu\mathbf{H}^{(\hat{s}_1\hat{t}_1)}).$$

Here, $(\hat{s}_m\hat{t}_m)$ denote the selected covariate and response-component from the base learner in (3.2) in boosting iteration m. The computation of the hat-matrix has a complexity of $O(n^2p + n^3q^2m)$ and is not feasible if n (and/or q) is large.

The trace of $\mathbf{K}_m$ gives the number of degrees of freedom. For AIC_c we need the degrees of freedom (number of equivalent parameters) per response variable; thus, we divide the total number of degrees of freedom by q to get the average number of degrees of freedom per response. For AIC and AIC_c for multivariate L_2Boosting, as functions of the number of iterations m, we get

$$AIC(m) = \log(|\hat{\mathbf{\Sigma}}(m)|) + \frac{2\cdot\mathrm{trace}(\mathbf{K}_m)}{n},$$
$$AIC_c(m) = \log(|\hat{\mathbf{\Sigma}}(m)|) + \frac{q(n + \frac{\mathrm{trace}(\mathbf{K}_m)}{q})}{n - \frac{\mathrm{trace}(\mathbf{K}_m)}{q} - q - 1},$$

where $\hat{\mathbf{\Sigma}}(m) = n^{-1}\sum_{i=1}^n(\mathbf{y}_{(i)} - \hat{\mathbf{f}}^{(m)}(\mathbf{x}_{(i)}))(\mathbf{y}_{(i)} - \hat{\mathbf{f}}^{(m)}(\mathbf{x}_{(i)}))^T$. The number of boosting iterations is chosen to minimize AIC or AIC_c, respectively:

$$\hat{m}_{stop} = \underset{0\le m<M}{\mathrm{argmin}}\ AIC_c(m),$$

where M is a pre-specified large, upper bound for the candidate number of boosting iterations.

3.5. L_2 Boosting with whole rows of B

Multivariate L_2Boosting changes only one entry of $\mathbf{B}$ at each step. This might be suboptimal if we believe that a covariate either has some influence on all response-components or no influence at all. It may then be better at each step to update a whole row of $\mathbf{B}$. This can also be done with a L_2Boosting-type algorithm, which we call "row-boosting": we select in each step the covariate which gives the best multivariate fit to the current residuals (according to Wilk's Λ) and add it to the multivariate function estimate. This algorithm is more closely related to multivariate forward variable selection, see Section 2.1, with the difference that we do not adjust the coefficients of the covariates already included in the model.

4. Consistency of Multivariate L_2Boosting

We present a consistency result for multivariate L_2Boosting in linear regression where the number of predictors and the dimension of the response are allowed to grow very fast as the sample size increases. Consider the model

$$
\begin{aligned}
&\mathbf{y}_{(i)} = \mathbf{f}(\mathbf{x}_{(i)}) + \mathbf{e}_{(i)},\ i = 1, \ldots, n,\quad \mathbf{y}_{(i)}, \mathbf{e}_{(i)} \in \mathbb{R}^{q_n},\quad \mathbf{x}_{(i)} \in \mathbb{R}^{p_n},\\
&\mathbf{f}(\mathbf{x}) = \mathbf{B}^T\mathbf{x},\quad \mathbf{B} \in \mathbb{R}^{p_n \times q_n},\\
&\mathbf{x}_{(i)} \text{ i.i.d. and } \mathbf{e}_{(i)} \text{ i.i.d., independent of } \{\mathbf{x}_{(i)}; 1 \le i \le n\}\\
&\text{with } E[\mathbf{e}_{(i)}] = \mathbf{0} \text{ and } \mathrm{Cov}(\mathbf{e}_{(i)}) = \mathbf{\Sigma}.
\end{aligned}
\tag{4.1}
$$

Because p_n and q_n are allowed to grow with n, the predictors and the responses depend on n. We ignore this notationally most of the time. To identify the magnitude of B_{jk} we assume $E|x_{(1)j}|^2 = 1,\ j = 1, \ldots, p_n$.

We make the following assumptions.

(A1) The dimension of the predictor and the response in (4.1) satisfies $p_n = O(\exp(Cn^{1-\xi})),\ q_n = O(\exp(Cn^{1-\xi}))\ (n \to \infty)$, for some $0 < \xi < 1, 0 < C < \infty$.

(A2) $\sup_{n \in \mathbb{N}} \sum_{j=1}^{p_n} \sum_{k=1}^{q} |B_{jk,n}| < \infty$.

(A3) For the implementing $\mathbf{\Gamma}$ in 3.1,
$\sup_{n \in \mathbb{N}, 1 \le k \le q_n} \sum_{\ell=1}^{q_n} |\Gamma^{-1}_{k\ell,n}| < \infty, \quad \inf_{n \in \mathbb{N}, 1 \le k \le q_n} \Gamma^{-1}_{kk,n} > 0.$

(A4) $\sup_{1 \le j \le p_n} \|x_{(1)j}\|_\infty < \infty$, where $\|x\|_\infty = \sup_{\omega \in \Omega} |x(\omega)|$ (Ω denotes the underlying probability space).

(A5) $\sup_{1 \le k \le q_n} E|e_{(1)k}|^s < \infty$ for some $s > 2/\xi$, with ξ from (A1).

Assumption (A1) allows for very large predictor and response dimensions relative to the sample size n. Assumption (A2) is a l_1-norm sparseness condition for the underlying multivariate regression function $\mathbf{f}(\cdot)$. If q_n grows with sample size it seems quite restrictive, but we describe a potential application in Section 4.1 (second example) where (A2) could be reasonable even if q_n grows. Assumption (A3) is a sparseness condition on $\mathbf{\Gamma}^{-1}$ that holds when choosing $\mathbf{\Gamma} = \mathbf{I}$ (or other reasonable diagonal matrices). Assumption (A4) and (A5) are the same as in Bühlmann (2004); (A4) can be relaxed at the price of a polynomial growth $O(n^\delta)$ $(0 < \delta < \infty)$ in (A1) and assuming sufficiently high-order moments, cf., Section 5.

Theorem 1. *Suppose* (A1)−(A5) *hold at* (4.1). *Then the multivariate L_2Boosting estimate* $\hat{\mathbf{f}}^{(m_n)}$ *with the componentwise linear learner from* (3.2) *satisfies: for* *some sequence* $(m_n)_{n\in\mathbb{N}}$ *with* $m_n \to \infty$ $(n \to \infty)$ *sufficiently slowly,*

$$E_{\mathbf{x}}\left[\left(\hat{\mathbf{f}}^{(m_n)}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\right)^T \mathbf{\Gamma}^{-1}\left(\hat{\mathbf{f}}^{(m_n)}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\right)\right] = o_p(1) \ (n \to \infty),$$

where $\mathbf{x}$ *denotes a new observation, independent of and with the same distribution as the* $\mathbf{x}_{(i)}$, $i = 1, \ldots, n$.

A proof is given in Section 9. Theorem 1 says that multivariate L_2Boosting recovers the true sparse regression function even if the dimensions of the predictor and response grow almost exponentially with sample size n.

Remark 1. We can also use the multivariate L_2Boosting for multi-category classification with q categories labeled $1, \ldots, q$. This can be encoded with a multivariate q-dimensional response $\mathbf{y} = (y_1, \ldots, y_q)$, where

$$y_j = \begin{cases} 1 & \text{if the category-label } = j, \\ 0 & \text{if the category-label } \neq j. \end{cases}$$

Assume that the data $(\mathbf{x}_{(1)}, \mathbf{y}_{(1)}), \ldots, (\mathbf{x}_{(n)}, \mathbf{y}_{(n)})$ are independent and identically distributed, that the conditional probabilities $\pi_j(\mathbf{x}) = P[y_j = 1|\mathbf{x}]$ are linear in $\mathbf{x}$. If (A1)$-$(A4) hold, then multivariate L_2Boosting is consistent e.g., with $\mathbf{\Gamma} = \mathbf{I}$: $\sum_{j=1}^{q} E_{\mathbf{x}}[(\hat{\pi}_j^{(m_n)}(\mathbf{x}) - \pi_j(\mathbf{x}))^2] = o_P(1)$.

Remark 1 is a consequence of Theorem 1. Note that for binary classification, we typically encode the problem by a univariate response. Multi-category problems could also be represented with a $q-1$-dimensional response, but this would require tagging a particular label as the complement of all others; we typically want to avoid such arbitrariness.

4.1. Two potential applications with large response dimension q

One problem is classification (see Remark 1) of biological objects such as genes or proteins into *subsets* of various functional categories, e.g., in Gene Ontology (GO) (cf., Kriegel et al. (2004)). Because biological objects typically belong to many functional categories, the labels for classification are subsets of functional categories, resulting in a large value of q (and p is large here as well).

Another application is to screening for associations of q candidate random variables $\mathbf{y}_1, \ldots, \mathbf{y}_q$ with a system of p target variables $\mathbf{x}_1, \ldots, \mathbf{x}_p$. This occurs in Wille et al. (2004) when screening expressions of $q = 795$ genes which exhibit some potential associations to the expressions of $p = 39$ genes from two biosynthesis pathways in Arabidopsis thaliana. We would like to know whether the partial correlation $\mathrm{Parcor}(\mathbf{y}_k, \mathbf{x}_j | \{\mathbf{x}_u; u \in \{1, \ldots, p\} \setminus j)$ is zero or not, for all $1 \leq k \leq q,\ 1 \leq j \leq p$. This is equivalent to checking in the linear regressions

$$\mathbf{y}_k = B_{jk}\mathbf{x}_j + \sum_{1 \leq u \leq p, u \neq j} B_{uk}\mathbf{x}_u + \mathbf{e}_k$$

whether $B_{jk} = 0$ or not. One could imagine that only a few of the q candidate variables $\mathbf{y}_1, \ldots, \mathbf{y}_q$ have something to do with the p target variables $\mathbf{x}_1, \ldots, \mathbf{x}_p$ (i.e., there are many k's where $B_{jk} \equiv 0$ for all j), and that existing relations between the candidate and target variables are sparse in terms of the corresponding regression coefficients, i.e., (A2) could be a reasonable assumption.

Remark 2. Using an estimate of $\mathbf{\Sigma}$ for $\mathbf{\Gamma}$ may result in a poor fit when q is large relative to n. In this case we may choose something simpler, e.g., $\mathrm{diag}(\hat{\mathbf{\Sigma}})$ or $\mathbf{I}$ (this is only reasonable when the responses are standardized) or a convex-combination $\gamma\hat{\mathbf{\Sigma}} + (1 - \gamma)\mathrm{diag}(\hat{\mathbf{\Sigma}})$ with $0 < \gamma < 1$ small. If $\mathbf{\Gamma}$ is diagonal, multivariate L_2Boosting fits q independent univariate linear regressions. For each response it produces the same sequence of selected covariates as univariate L_2Boosting. The only difference in the multivariate method is that it mixes the individual sequences of selected covariates for the different responses, and uses only one stopping iteration. From a theoretical point of view, the multivariate method (even for $\mathbf{\Gamma} = \mathbf{I}$) allows one to derive consistency for growing q.

5. Multivariate L_2 Boosting for vector AR processes

The boosting method from Section 3 can be used for vector autoregressive processes (VAR, see for example Lütkepohl (1993))

$$\mathbf{x}_{(t)} = \sum_{j=1}^{p} \mathbf{A_j}\mathbf{x}_{(t-j)} + \mathbf{e}_{(t)}, \quad t \in \mathbb{Z}, \tag{5.1}$$

where $\mathbf{x}_{(t)} \in \mathbb{R}^q$ is the q-dimensional observation at time t, $\mathbf{A_j} \in \mathbb{R}^{q \times q}$ and $\mathbf{e}_{(t)} \in \mathbb{R}^q$ i.i.d. with $E[\mathbf{e}_{(t)}] = \mathbf{0}$ and $\mathrm{Cov}(\mathbf{e}_{(t)}) = \mathbf{\Sigma}$. The model is stationary and causal if all roots of $\det(\mathbf{I} - \sum_{j=1}^{p} \mathbf{A_j} z^j)$ $(z \in \mathbb{C})$ are greater than one in absolute value.

For observations $\mathbf{x}_{(t)}$ $(t = 1, \ldots, n)$, (5.1) can be written as a multivariate regression model as at (2.1), with $\mathbf{Y} = [\mathbf{x}_{(p+1)}, \ldots, \mathbf{x}_{(n)}]^T \in \mathbb{R}^{(n-p) \times q}$, $\mathbf{B} = [\mathbf{A_1}, \ldots, \mathbf{A_p}]^T \in \mathbb{R}^{qp \times q}$ and $\mathbf{X} \in \mathbb{R}^{(n-p) \times qp}$ the corresponding design matrix.

The consistency result from Theorem 1 carries over to the time series case. We assume that the data is generated from the q_n-dimensional VAR(∞) model

$$\mathbf{x}_{(t)} = \sum_{j=1}^{\infty} \mathbf{A_j}\mathbf{x}_{(t-j)} + \mathbf{e}_{(t)}, \quad t \in \mathbb{Z}, \tag{5.2}$$

with $\mathbf{e}_{(t)} \in \mathbb{R}^{q_n}$ i.i.d., $E[\mathbf{e}_{(t)}] = \mathbf{0}$, $\mathrm{Cov}(\mathbf{e}_{(t)}) = \mathbf{\Sigma}$, and $\mathbf{e}_{(t)}$ independent of $\{\mathbf{x}_{(s)};\ s < t\}$. Again, we ignore notationally that the model and its terms depend on n due to the growing dimension q_n. Appropriate assumptions now are the following.

(B1) $\{\mathbf{x}_{(t)}\}_{t\in\mathbb{Z}}$ in (5.2) is strictly stationary and α-mixing with mixing coefficients $\alpha_n(\cdot)$.

(B2) The dimension satisfies: $q = q_n = O(n^{\delta})$ for some $0 < \delta < \infty$.

(B3) $\sup_{n\in\mathbb{N}} \sum_{j=1}^{\infty} \sum_{k,v=1}^{q_n} |A_{kv;j,n}| < \infty, \quad A_{kv;j,n} = (\mathbf{A}_{\mathbf{j},n})_{kv}$.

(B4) The mixing coefficients and moments are such that, for some $s \in \mathbb{N}$ with $s > 2(1+\delta) - 2$ (δ as in (B2)) and $\gamma > 0$,

$$\sum_{k=1}^{\infty} (k+1)^{s-1} \alpha_n(k)^{\gamma/(2s+\gamma)} < \infty,$$
$$\sup_{1\le k\le q_n, n\in\mathbb{N}} E|x_{(t)k}|^{4s+2\gamma} < \infty, \qquad \sup_{1\le k\le q_n, n\in\mathbb{N}} E|e_{(t)k}|^{2s+\gamma} < \infty.$$

Theorem 2. *Suppose* (B1)$-$(B4) *and* (A3) *hold at* (5.2). *Consider multivariate L_2Boosting with componentwise linear least squares* (*as in Section* 3) *using* $p = p_n$ *lagged variables* (*as in* (5.1)) *with* $p_n \to \infty$, $p_n = O(n^{1-\kappa})$ $(n \to \infty)$, *where* $2(1+\delta)/(s+2) < \kappa < 1$. *Then, the conclusion of Theorem* 1 *holds with* $\mathbf{f}(\mathbf{x}) = \sum_{j=1}^{\infty} \mathbf{A}_{\mathbf{j}} \mathbf{x}_{(t-j)}$, $\hat{\mathbf{f}}^{(m_n)}(\mathbf{x}) = \sum_{j=1}^{p_n} \hat{\mathbf{A}}_{\mathbf{j}}^{(m_n)} \mathbf{x}_{(t-j)}$ *and* $\mathbf{x}$ *a new realization from* (5.2), *independent of the training data.*

A proof is given in Section 9. Note that if in (B4) the mixing coefficients decay exponentially and all moments exist, i.e., for a suitably regular Gaussian VAR(p) of finite order, Theorem 2 holds for arbitrarily large δ in (B2) and arbitrarily small $\kappa > 0$, implying $p_n = O(n^{1-\kappa})$ is allowed to grow almost as fast as n.

6. Simulation Study

In this Section we compare multivariate L_2Boosting (MB) to individual L_2Boosting (IB, univariate L_2Boosting for each response alone; cf., Bühlmann (2004)), row-boosting (RB, see Section 3.5), and multivariate forward stepwise variable selection (MFS, see Section 2.1) on simulated data sets.

6.1. Design

The sample size is always $n = 50$ and the number of responses is $q = 5$. We take two numbers of covariates ($p = 10$ and $p = 30$) and two proportions of non-zero entries of $\mathbf{B}$ ($p_{eff} = 0.2$ and $p_{eff} = 0.5$, where $p_{eff} = 0.2$ means that 20% of the entries of $\mathbf{B}$ are non-zero).

The covariates are generated according to a multivariate normal distribution with covariance matrix $\mathbf{V}$, $\mathbf{x}_{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{V})$, with $V_{kv} = 0.9^{|k-v|}$. The value 0.9 seems to be pretty high but, when having $p = 30$ covariates, the average correlation between the covariates is only 0.42. Smaller values lead to similar results among the boosting methods, only MFS performs then a bit better.

For the true coefficient-matrix $\mathbf{B}$ we take two different types, characterized by the non-zero entries. For the first type, we arbitrarily choose the $q \cdot p \cdot p_{eff}$ non-zero entries of $\mathbf{B}$ with the only constraint that each response must depend on at least one covariate. We call this type "**B** arbitrary" (this is the case of seemingly unrelated regressions). For the other type, we randomly choose $p \cdot p_{eff}$ rows of $\mathbf{B}$ and take entries of the full rows unequal to zero ("**B** row-complete"). The non-zero entries of $\mathbf{B}$ are, for both types, i.i.d. $\sim \mathcal{N}(0, 1)$.

The errors are again generated according to a multivariate normal distribution with covariance-matrix $\mathbf{\Sigma}$, $\mathbf{e}_{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$. The diagonal elements of $\mathbf{\Sigma}$ are constructed to give individual signal-to-noise ratios of 0.71, 0.84, 1.00, 1.19, 1.41. The off-diagonal elements of $\mathbf{\Sigma}$ are chosen so that $\mathrm{Cor}(\mathbf{e}_k, \mathbf{e}_v) = \rho^{|k-v|}$, with ρ taking the values 0, 0.6, and 0.9. All responses are standardized to unit variance to make them comparable.

The design of this simulation comprises two types of **B**-matrices, three values for the correlations between the errors, two values for the number of predictors and two values for the number of effective predictors. A complete factorial design over all these levels gives rise to 24 settings. Each setting is replicated 100 times and the different methods are applied.

To select the number of boosting iterations or the number of steps in MFS we use either a validation set of size 50 or AIC_c. For all boosting methods we choose the shrinkage factor $\nu = 0.1$. For the covariance-matrix $\mathbf{\Gamma}$ in MB we use the empirical covariance-matrix of the residuals $\mathbf{r}_{(i)}^{IB}$ of the IB: $\mathbf{\Gamma} = \hat{\mathbf{\Sigma}} = n^{-1} \sum_{i=1}^{n} \mathbf{r}_{(i)}^{IB} (\mathbf{r}_{(i)}^{IB})^T$.

6.2. Performance measure

In simulations we can measure how close the prediction for an additional observation comes to the true value. For the kth response, the mean squared prediction error is given by

$$MSPE_k = \int \left(\mathbf{x}^T(\mathbf{b}_k - \hat{\mathbf{b}}_k)\right)^2 dP(\mathbf{x}) = (\mathbf{b}_k - \hat{\mathbf{b}}_k)^T \mathbf{V} (\mathbf{b}_k - \hat{\mathbf{b}}_k).$$

Our performance measure is the mean of the individual $MSPE$'s

$$MSPE = q^{-1} \sum_{k=1}^{q} MSPE_k.$$

This is reasonable, because we have standardized the responses.

6.3. Results

The results are summarized in Table 6.1 and Figure 6.1. We give the mean of the $MSPE$ of the 100 replicates (multiplied by 1,000) for each method and setting. Additionally, paired sample Wilcoxon tests are performed that compare for each setting the best method to the other three methods. A p-value below $1e - 9$ is set to zero. The iterations are stopped with a validation set.

Table 6.1. Mean squared prediction error $MSPE$, multiplied by 1,000, of multivariate forward stepwise variable selection (MFS), row-boosting (RB), individual L_2Boosting (IB) and multivariate L_2Boosting (MB) averaged over 100 replicates. The best method for each setting is in bold face. P-values of the paired sample Wilcoxon tests that compare for each setting the best method to the other three methods, are also given.

				MSPE				Wilcoxon p-value			
B	ρ	p	p_{eff}	MFS	RB	IB	MB	MFS	RB	IB	MB
arbitr.	0.0	10	0.2	84	63	**50**	51	0	0		1e−1
arbitr.	0.0	10	0.5	96	71	**66**	67	0	8e−5		9e−2
arbitr.	0.0	30	0.2	176	125	**112**	116	0	1e−9		5e−3
arbitr.	0.0	30	0.5	216	132	**130**	135	0	1e−1		1e−3
arbitr.	0.6	10	0.2	73	60	50	**44**	0	0	2e−6	
arbitr.	0.6	10	0.5	93	71	67	**62**	0	8e−8	3e−4	
arbitr.	0.6	30	0.2	164	116	109	**100**	0	0	4e−6	
arbitr.	0.6	30	0.5	203	126	127	**117**	0	6e−6	5e−7	
arbitr.	0.9	10	0.2	62	53	49	**33**	0	0	0	
arbitr.	0.9	10	0.5	93	71	68	**51**	0	0	0	
arbitr.	0.9	30	0.2	149	107	110	**72**	0	0	0	
arbitr.	0.9	30	0.5	183	115	126	**85**	0	0	0	
row-c.	0.0	10	0.2	**26**	41	48	50		0	0	0
row-c.	0.0	10	0.5	70	**66**	67	71	6e−2		2e−1	4e−6
row-c.	0.0	30	0.2	123	**105**	118	121	1e−4		1e−9	0
row-c.	0.0	30	0.5	203	**132**	136	139	0		7e−2	1e−5
row-c.	0.6	10	0.2	**25**	38	49	50		1e−9	0	0
row-c.	0.6	10	0.5	64	**60**	64	63	2e−2		6e−4	7e−2
row-c.	0.6	30	0.2	109	**101**	120	110	3e−2		0	3e−6
row-c.	0.6	30	0.5	186	**128**	137	129	0		1e−5	8e−1
row-c.	0.9	10	0.2	**31**	33	50	45		6e−2	6e−9	2e−5
row-c.	0.9	10	0.5	62	54	63	**48**	5e−5	2e−1	1e−9	
row-c.	0.9	30	0.2	**88**	88	120	89		7e−1	0	5e−1
row-c.	0.9	30	0.5	179	120	137	**102**	0	4e−7	0	

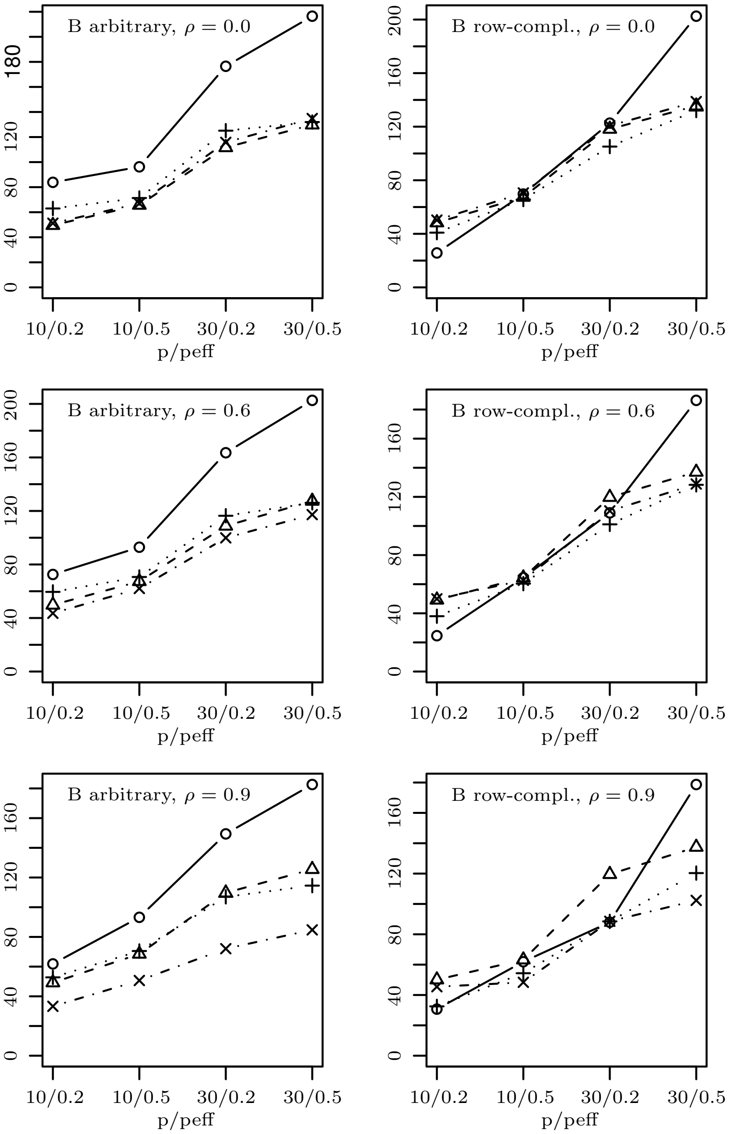

Figure 6.1. Mean squared prediction error MSPE, multiplied by 1,000, of multivariate forward stepwise variable selection ($\circ$), row-boosting ($+$), individual L_2Boosting ($\triangle$) and multivariate L_2Boosting ($\times$).

For $\rho = 0$, multivariate L_2Boosting is a few percent worse than individual L_2Boosting. But for $\rho = 0.6$ and $\rho = 0.9$, MB performs significantly better than IB and the gain can be up to a factor of 1.5 (for less correlated predictors the gain is even bigger). Thus, MB is able to exploit the additional information in a multivariate response.

As expected, MB and IB perform well when $\mathbf{B}$ is arbitrary and RB performs well when $\mathbf{B}$ is row-complete. MFS gives good results only in the easier settings, especially with $\mathbf{B}$ row-complete, $p = 10$ and $p_{eff} = 0.2$. It is interesting to see that MB performs best in the case when $\mathbf{B}$ is row-complete, $\rho = 0.9$ and $p_{eff} = 0.5$, even though the setting favors methods which work with whole rows of $\mathbf{B}$.

The given results come about with stopping by a validation set. Stopping methods which only use the training data (like AIC_c) lead on average to worse results because they use much less information. Therefore we can use the validation set stopping as a benchmark to assess the performance of AIC_c stopping: MB is 6.3% worse (median over all 24 settings) when we use AIC_c instead of the validation set, RB is 3.5% worse, MFS 10.2% worse and IB 25.0% worse. AIC_c stopping works relatively better for the multivariate methods (MB, RB and also MFS) than for IB. A possible explanation is that MB and RB have to be stopped only once and not q times. This gives less variability in the final boosting estimate and makes it easier to stop at a good point. Note that for IB, it is desirable, or even essential, to allow for individual stopping iterations because we often need varying complexities for modelling the different response variables.

7. Data

We have analyzed the following data sets.

Chemical reaction data (Box and Youle (1955) and Rencher (2002)). This is a planned experiment involving a chemical reaction with three input (predictor) variables (temperature, concentration, time) and three output (response) variables (percentage of unchanged starting material, percentage converted to the desired product, percentage of unwanted by-product). We fit a quadratic model including the first order interactions (product of the predictor variables). This gives a total of nine covariates.

Macroeconomic data (Klein, Ball, Hazlewood and Vandome (1961) and Reinsel and Velu (1998)). This is a 10-dimensional time series from the United Kingdom from 1948−1956 with quarterly measurements. Five terms are taken as predictor variables (total labor force, weekly wage rates, price index of imports, price index of exports, price index of consumption) and five terms are taken as response variables (industrial production, consumption, unemployment, total imports, total exports). We ignore the time-dependency of the observations and again fit a quadratic model with first order interactions.

Chemometrics data (Skagerberg, MacGregor and Kiparissides (1992) and Breiman and Friedman (1997)). This is a simulation of a low density tubular polyethylene reactor. There are 22 predictor variables (20 reactor temperatures, wall temperature of the reactor, feed rate of the reactor) and six responses (number-average molecular weight, weight-average molecular weight, frequency of long chain branching, frequency of short chain branching, content of vinyl groups, content of vinylidene groups). Because the responses are skewed, they are all log-transformed.

Arabidopsis thaliana data (Wille et al. (2004)). This is a microarray experiment. There are 795 genes (the responses) that may show some association to 39 genes (the predictors) from two biosynthesis pathways in A. thaliana. All variables are log transformed.

All responses are standardized to unit variance to make them comparable. The predictive accuracy of each method is estimated by leave-one-out cross-validation (for the A. thaliana data set we used 5-fold cross-validation):

$$MSPE_{CV} = q^{-1}\sum_{k=1}^{q} n^{-1}\sum_{i=1}^{n}(y_{(i)k} - \hat{f}_k^{(-i)}(\mathbf{x}_{(i)}))^2.$$

Note that we compare the prediction with the observation, the latter being an unbiased rough estimate for the true unknown function $\mathbf{f}$. Therefore the prediction accuracy contains also the error variances, which makes it harder to see clear differences between the methods.

The data sets are summarized in Table 7.2 and the results are given in Table 7.3. We use 5-fold cross validation and AIC_c to stop the iteration. The implementing $\mathbf{\Gamma}$ in MB is the empirical covariance-matrix $\hat{\mathbf{\Sigma}}$ of the residuals from IB (for the A. thaliana data set we used the diagonal matrix diag$(\hat{\mathbf{\Sigma}})$).

MFS performs worst, but there is no overall best boosting method. As mentioned already in Section 6.3, it seems easier to stop the iteration for MB and RB than for IB. Therefore, cross-validation stopping and AIC_c stopping differ only slightly for MB and RB.

For IB, stopping by AIC_c works much better than using cross validation in two examples. The mean squared prediction error of 0.208 for the chemometrics data is quite good compared to the numbers published in Breiman and Friedman (1997). We remark here that we only have rounded data (taken from Skagerberg et al. (1992)) and therefore we get slightly different prediction errors (e.g., for OLS: 0.411 instead of 0.431 in Breiman and Friedman (1997)).

Table 7.2. Summary of the analyzed data sets: sample size (n), number of predictors (p), number of responses (q), and average absolute empirical correlation between the responses (aac).

Data set	n	p	q	aac
Chemical reaction	19	9	3	0.56
Macroeconomic	36	20	5	0.71
Chemometrics	56	22	6	0.48
A. thaliana	118	39	795	0.21

Table 7.3. Leave-one-out (5-fold for A. thaliana) cross-validated mean squared prediction error $MSPE_{CV}$ for four data sets. Iteration stopped either by 5-fold cross validation or AIC_c. AIC_c-stopping for A. thaliana is not easily accomplished (see Section 3.4) and the computation of Wilk's Λ (MFS and RB) is only possible if $n > q$.

Data set	OLS	MFS		RB		IB		MB	
		CV	AIC_c	CV	AIC_c	CV	AIC_c	CV	AIC_c
Chemical	1.343	1.261	0.616	0.532	0.500	0.744	0.527	0.488	0.479
Macroe.	0.499	0.209	0.224	0.193	0.197	0.194	0.195	0.202	0.204
Chemom.	0.411	0.360	0.386	0.253	0.262	0.260	0.208	0.259	0.263
A. thaliana	0.753					0.559	0.556	0.551	

8. Conclusions

We propose a multivariate L_2Boosting method for multivariate linear models. The multivariate L_2Boosting inherits the good properties from its univariate counterpart: it does variable selection and shrinkage. Our multivariate L_2Boosting method is suitable for a variety of different situations: (i) multivariate linear regression, with or without seemingly unrelated regressions (SUR), and with covariates which can be arbitrarily correlated; (ii) for multivariate vector autoregressive time series. The method is particularly powerful if the predictor dimension, or the dimension of the response, is large relative to the sample size.

Our multivariate L_2Boosting takes potential correlations among the components of the multivariate error-noise into account. It is therefore very different from OLS and other methods which work on individual responses only. Correlation among the errors can arise from various sources – for example via an unobservable covariate which influences the responses in the same way.

We prove here, for i.i.d. data as well as for time series, that multivariate L_2Boosting can consistently recover sparse, very high-multivariate and very high-dimensional linear functions. In high response-dimensions, a non-trivial issue is how to control the estimation error over all multivariate components simultaneously. Our theory seems to be among the first which actually addresses such questions.

An important question in multivariate regression is whether "jointness" pays off: is the multivariate method better than q estimates from a univariate method? Our simulation study shows that multivariate L_2Boosting outperforms individual univariate L_2Boosting by a substantial amount when the errors are correlated, and is almost as good when the errors are independent. In data examples, we were not able to see a clear difference (which may be masked by substantial noise variance). This has been found in other work, e.g., Brooks and Stone (1994).

9. Proofs

9.1. Proof of Theorem 1

The proof of Theorem 1 is similar to that in Bühlmann (2004), where the univariate case is discussed. We define an appropriate Hilbert space and dictionary of basis functions; then, it is sufficient to prove Lemma 1 from Bühlmann (2004) for the setting of multivariate L_2Boosting.

A population version

The L_2Boosting algorithm has a population version known as "matching pursuit" (Mallat and Zhang (1993)) or "weak greedy algorithm" (Temlyakov (2000)).

Consider the Hilbert space $L_2(P) = \{\mathbf{f} : \mathbb{R}^{p_n} \to \mathbb{R}^{q_n}; \|\mathbf{f}\|^2 = \langle \mathbf{f}, \mathbf{f} \rangle < \infty\}$ with inner product $\langle \mathbf{f}, \mathbf{g} \rangle = \int \mathbf{f}(\mathbf{x})^T \mathbf{\Gamma}^{-1} \mathbf{g}(\mathbf{x}) dP(\mathbf{x})$. Here, the probability measure P is generating the predictor $\mathbf{x}$ in (4.1). To be precise, the probability measure $P = P_n$ and the function $\mathbf{f} = \mathbf{f}_n$ depend on n, but we often ignore this notationally (a uniform bound in (9.4) will be a key result to deal with sequences of Hilbert spaces).

Denote the components of $\mathbf{x} = (x_1, \ldots, x_{p_n})$ viewed as scalars or as 1-dimensional functions from $\mathbb{R}^{p_n} \to \mathbb{R}$, by $g_j(\mathbf{x}) = x_j$, and denote the components of $\mathbf{x} = (x_1, \ldots, x_{p_n})$ viewed as q_n-dimensional vectors or as functions from $\mathbb{R}^{p_n} \to \mathbb{R}^{q_n}$ with only component k different from zero, by

$$(\mathbf{g}_{(j,k)})_l(\mathbf{x}) = \begin{cases} x_j, & \text{if } l = k, \\ 0, & \text{if } l \neq k. \end{cases}$$

For notational simplicity, assume that $\|\mathbf{g}_{(j,k)}\| = \int x_j^2 \Gamma_{kk}^{-1} dP(x_j) = \Gamma_{kk}^{-1} = 1$ for all k (this simplifies e.g., the formula (9.2)); the proof for non-equal Γ_{kk}^{-1} would work analogously using the second assumption in (A3).

Define the following sequence of remainder functions, called matching pursuit or the weak greedy algorithm:

$$\begin{aligned} &R^0\mathbf{f} = \mathbf{f}, \\ &R^m\mathbf{f} = R^{m-1}\mathbf{f} - \left\langle R^{m-1}\mathbf{f}, \mathbf{g}_{(s_m,t_m)} \right\rangle \mathbf{g}_{(s_m,t_m)}, \ m = 1, 2, \ldots, \end{aligned} \tag{9.1}$$

where (s_m, t_m) would be ideally chosen as

$$(s_m, t_m) = \underset{1 \le j \le p_n; 1 \le k \le q_n}{\operatorname{argmax}} |\left\langle R^{m-1}\mathbf{f}, \mathbf{g}_{(j,k)} \right\rangle|.$$

The choice functions (s_m, t_m) are often infeasible to realize in practice, because we have finite samples. A weaker criterion is: for every m (under consideration), choose any (s_m, t_m) which satisfies, for some $0 < d \le 1$,

$$|\left\langle R^{m-1}\mathbf{f}, \mathbf{g}_{(s_m,t_m)} \right\rangle| \ge d \cdot \sup_{1 \le j \le p_n; 1 \le k \le q_n} |\left\langle R^{m-1}\mathbf{f}, \mathbf{g}_{(j,k)} \right\rangle|. \tag{9.2}$$

Of course, the sequence $R^m\mathbf{f} = R^{m,s,t}\mathbf{f}$ depends on $(s_1, t_1), (s_2, t_2), \ldots, (s_m, t_m)$ as to how we actually make the choice in (9.2). Again, we ignore this notationally.

It easily follows that

$$\mathbf{f} = \sum_{j=0}^{m-1} \left\langle R^j\mathbf{f}, \mathbf{g}_{(s_{j+1},t_{j+1})} \right\rangle \mathbf{g}_{(s_{j+1},t_{j+1})} + R^m\mathbf{f}.$$

Temlyakov (2000) gives a uniform bound for the algorithm in (9.1) with (9.2).

If the function $\mathbf{f}$ is representable as

$$\mathbf{f}(\mathbf{x}) = \sum_{j,k} B_{jk}\mathbf{g}_{(j,k)}(\mathbf{x}), \ \sum_{j,k} |B_{jk}| \le D < \infty, \tag{9.3}$$

which is true by (A2), then

$$\|R^m\mathbf{f}\| \le D(1 + md^2)^{-d/(2(2+d))}, \ 0 < d \le 1 \text{ as in (9.2).} \tag{9.4}$$

To make the point clear, this bound holds also for sequences $R^m\mathbf{f} = R^{m,s,t,n}\mathbf{f}$ which depend on the choice function (s, t) in (9.2) and on the sample size n (since $\mathbf{x} \sim P$ depends on n and also the function of interest $\mathbf{f}$); all we have to assume is the condition (9.3).

A sample version

The multivariate L_2boosting algorithm can be represented analogously to (9.1). We introduce the following notation:

$$\langle \mathbf{f}, \mathbf{g} \rangle_{(n)} = n^{-1} \sum_{i=1}^{n} \mathbf{f}^T(\mathbf{x}_{(i)})\mathbf{\Gamma}^{-1}\mathbf{g}(\mathbf{x}_{(i)}) \text{ and } \|\mathbf{f}\|^2_{(n)} = \langle \mathbf{f}, \mathbf{f} \rangle_{(n)}$$

for functions $\mathbf{f}, \mathbf{g} : \mathbb{R}^{p_n} \to \mathbb{R}^{q_n}$. As before, we denote by $\mathbf{Y} = (\mathbf{y}_{(1)}, \ldots, \mathbf{y}_{(n)})^T$ the matrix of response variables.

Define

$$\hat{R}^1_n\mathbf{f} = \mathbf{f} - \left\langle \mathbf{Y}, \mathbf{g}_{(\hat{s}_1,\hat{t}_1)} \right\rangle_{(n)} \mathbf{g}_{(\hat{s}_1,\hat{t}_1)},$$
$$\hat{R}^m_n\mathbf{f} = \hat{R}^{m-1}_n\mathbf{f} - \left\langle \hat{R}^{m-1}_n\mathbf{f}, \mathbf{g}_{(\hat{s}_m,\hat{t}_m)} \right\rangle_{(n)} \mathbf{g}_{(\hat{s}_m,\hat{t}_m)}, \ m = 2, 3, \ldots,$$

where

$$(\hat{s}_1, \hat{t}_1) = \underset{1 \le j \le p_n; 1 \le k \le q_n}{\operatorname{argmax}} \left| \left\langle \mathbf{Y}, \mathbf{g}_{(j,k)} \right\rangle_{(n)} \right|,$$
$$(\hat{s}_m, \hat{t}_m) = \underset{1 \le j \le p_n; 1 \le k \le q_n}{\operatorname{argmax}} \left| \left\langle \hat{R}^{m-1}_n\mathbf{f}, \mathbf{g}_{(j,k)} \right\rangle_{(n)} \right|, \ m = 2, 3, \ldots.$$

With some abuse of notation, we denote by $\hat{R}_n^{m-1}\mathbf{f}$ and $\mathbf{g}_{(\hat{s}_m,\hat{t}_m)}$ either functions from $\mathbb{R}^{p_n} \to \mathbb{R}^{q_n}$ or $n \times q_n$ matrices evaluated at the observed predictors. We emphasize here the dependence of $\hat{R}_n^m$ on n since finite-sample estimates $\langle \hat{R}_n^{m-1}\mathbf{f}, \mathbf{g}_{(j,k)}\rangle_{(n)}$ are involved. We also assume without loss of generality (but simplifying the notation) that $\|\mathbf{g}_{(j,k)}\|_{(n)} \equiv 1$ for all j,k and n (note that we have already assumed w.l.o.g. that $\|\mathbf{g}_{(j,k)}\| \equiv 1$ for all j,k). Then, the formulae above are the same as in (3.2) (because $\|\mathbf{g}_{(j,k)}\|_{(n)} = \mathbf{x}_j^T\mathbf{x}_j\Gamma_{kk}^{-1}$). Hence, $\hat{R}_n^m\mathbf{f} = \mathbf{f} - \hat{\mathbf{f}}^{(m)}$.

For analyzing $\|\hat{R}_n^m\mathbf{f}\| = E_{\mathbf{x}}|(\hat{\mathbf{f}}^{(m_n)}(\mathbf{x}) - \mathbf{f}(\mathbf{x}))^T\mathbf{\Gamma}^{-1}(\hat{\mathbf{f}}^{(m_n)}(\mathbf{x}) - \mathbf{f}(\mathbf{x}))|$, which is the quantity in the assertion of Theorem 1, we need some uniform laws of large numbers, as discussed below.

Uniform laws of large numbers

Lemma 1. *Under* (A1)$-$(A5), *with* $0 < \xi < 1$ *as in* (A1),
(i) $\sup_{1\le j,u\le p_n;1\le k,v\le q_n} |\langle \mathbf{g}_{(j,k)}, \mathbf{g}_{(u,v)}\rangle_{(n)} - \langle \mathbf{g}_{(j,k)}, \mathbf{g}_{(u,v)}\rangle| = \zeta_{n,1} = O_P(n^{-\xi/2})$,
(ii) $\sup_{1\le j\le p_n;1\le k\le q_n} |\langle \mathbf{g}_{(j,k)}, \mathbf{E}\rangle_{(n)}| = \zeta_{n,2} = O_P(n^{-\xi/2})$,
(iii) $\sup_{1\le j\le p_n;1\le k\le q_n} |\langle \mathbf{g}_{(j,k)}, \mathbf{f}\rangle_{(n)} - \langle \mathbf{g}_{(j,k)}, \mathbf{f}\rangle| = \zeta_{n,3} = O_P(n^{-\xi/2})$,
(iv) $\sup_{1\le j\le p_n;1\le k\le q_n} |\langle \mathbf{g}_{(j,k)}, \mathbf{Y}\rangle_{(n)} - \langle \mathbf{g}_{(j,k)}, \mathbf{Y}\rangle| = \zeta_{n,4} = O_P(n^{-\xi/2})$.

Proof. For (i), note that

$$\begin{aligned}
&\sup_{j,u,k,v}\left|\left\langle \mathbf{g}_{(j,k)}, \mathbf{g}_{(u,v)}\right\rangle_{(n)} - \left\langle \mathbf{g}_{(j,k)}, \mathbf{g}_{(u,v)}\right\rangle\right| = \\
&= \sup_{j,u,k,v}\left|n^{-1}\sum_{i=1}^n \mathbf{g}_{(j,k)}^T(\mathbf{x}_{(i)})\mathbf{\Gamma}^{-1}\mathbf{g}_{(u,v)}(\mathbf{x}_{(i)}) - E\left[\mathbf{g}_{(j,k)}^T(\mathbf{x}_{(i)})\mathbf{\Gamma}^{-1}\mathbf{g}_{(u,v)}(\mathbf{x}_{(i)})\right]\right| = \\
&= \sup_{j,u,k,v}\left|n^{-1}\sum_{i=1}^n x_{(i)j}\Gamma_{kv}^{-1}x_{(i)u} - E\left[x_{(1)j}\Gamma_{kv}^{-1}x_{(1)u}\right]\right| \\
&= \sup_{j,u,k,v}\left|\Gamma_{kv}^{-1}\right|\left|n^{-1}\sum_{i=1}^n g_j(\mathbf{x}_{(i)})g_u(\mathbf{x}_{(i)}) - E\left[g_j(\mathbf{x}_{(1)})g_u(\mathbf{x}_{(1)})\right]\right| \\
&\le \sup_{k,v}\left|\Gamma_{kv}^{-1}\right| \cdot O_P(n^{-\frac{\xi}{2}}) = O_P(n^{-\frac{\xi}{2}}).
\end{aligned}$$

We have used here that $\sup_{j,u}|n^{-1}\sum_{i=1}^n g_j(\mathbf{x}_{(i)})g_u(\mathbf{x}_{(i)}) - E[g_j(\mathbf{x}_{(1)})g_u(\mathbf{x}_{(1)})]| = O_P(n^{-\xi/2})$ (Bühlmann (2004)), and also the first assumption in (A3).

For (ii), we write

$$\left\langle \mathbf{g}_{(j,k)}, \mathbf{E}\right\rangle_{(n)} = \sum_{v=1}^{q_n} n^{-1}\sum_{i=1}^n x_{(i)j}\Gamma_{kv}^{-1}e_{(i)v} = n^{-1}\sum_{i=1}^n g_j(\mathbf{x}_{(i)})Q_i(k), \qquad (9.5)$$

where $Q_i(k) = \sum_{v=1}^{q_n}\Gamma_{kv}^{-1}e_{(i)v}$.

Note that $Q_i(k)$ is independent from $\mathbf{X}$, $E[Q_i(k)] = 0$ for all i, k and

$$\sup_k E|Q_i(k)|^s \leq \sup_k \Big(\sum_{v=1}^{q_n} |\Gamma_{kv}^{-1}| (E|e_{(1)v}|^s)^{1/s} \Big)^s < \infty, \tag{9.6}$$

using (A3) and (A5). The form in (9.5), with the moment property in (9.6), is the same as in Lemma 1 (ii) from Bühlmann (2004).

For (iii), note that

$$\sup_{j,k} \Big| \big\langle \mathbf{g}_{(j,k)}, \mathbf{f} \big\rangle_{(n)} - \big\langle \mathbf{g}_{(j,k)}, \mathbf{f} \big\rangle \Big| \leq \sum_{u,v} |B_{uv,n}| \cdot \zeta_{n,1} = O_P(n^{-\xi/2})$$

using (A2) and the bound from (i).

Finally, (iv) follows immediately from (ii) and (iii).

The rest of the proof is the same as in Bühlmann (2004). We only have to replace the basis functions g_j by our double indexed basis functions $\mathbf{g}_{(j,k)}$.

9.2. Proof of Theorem 2

As we have seen from the proof of Theorem 1, a substantial part of the analysis can be borrowed from Bühlmann (2004); we only need to reconsider uniform laws of large numbers, as in Lemma 1, but for dependent data. This can be done by invoking the following result.

Lemma 2. *Consider sequences $\{Z_{t,n}\}_{t \in \mathbb{Z}}$, $n \in \mathbb{N}$, that are strictly stationary and α-mixing, with mixing coefficients $\alpha_{Z,n}(\cdot)$. Assume that $E[Z_{t,n}] = 0$ for all $n \in \mathbb{N}$, $\sup_{n \in \mathbb{N}} E|Z_{t,n}|^{2s+\gamma} < \infty$ for some $s \in \mathbb{N}$, $\gamma > 0$, and the mixing coefficients satisfy for, some constants $0 < C_1, C_2 < \infty$,*

$$\sum_{k=0}^{\infty} (k+1)^{s-1} \alpha_{Z,n}(k)^{\gamma/(4s+\gamma)} < C_1 p_n^s + C_2,$$

where $s \in \mathbb{N}$ is linked to the moments of $Z_{t,n}$ as above. Then,

$$E \Big| n^{-1} \sum_{t=1}^{n} Z_{t,n} \Big|^{2s} = O(p_n^s n^{-s}) \ (n \to \infty).$$

Proof. The reasoning can be done analogously to the proof of Theorem 1 in Yokoyama (1980).

The only part of the proof of Theorem 1 which needs to be changed is Lemma 1. A version of Lemma 1 also holds for stationary VAR(∞) processes;

the predictor variables at time t are the p_n lagged q_n-dimensional variables $\mathbf{x}_{(t-1)}, \ldots, \mathbf{x}_{(t-p)}$ and the response variable is the current $\mathbf{x}_{(t)}$.

Instead of exponential inequalities we first invoke Markov's inequality and then Lemma 2. For example, for the analogue of Lemma 1 (i), note that

$$\begin{aligned} &P\Big[|(n-p_n)^{-1}\sum_{t=p_n+1}^{n} x_{(t-j)k}x_{(t-u)v} - E[x_{(t-j)k}x_{(t-u)v}]| > \varepsilon\Big] \\ &\le \varepsilon^{-2s}\ E\Big|(n-p_n)^{-1}\sum_{t=p_n+1}^{n} x_{(t-j)k}x_{(t-u)v} - E[x_{(t-j)k}x_{(t-u)v}]\Big|^{2s}. \qquad (9.7) \end{aligned}$$

We now observe that $Z_{t,n} = x_{(t-j)k}x_{(t-u)v} - E[x_{(t-j)k}x_{(t-u)v}]$ is still stationary and α-mixing with coefficients that satisfy the requirement from Lemma 2. Due to different lags j and u, the mixing coefficients of $Z_{t,n}$ usually don't decay for the first $|j-u|$ lags (therefore the special construction with $C_1 p_n^s + C_2$ in Lemma 2. Invoking Lemma 2 for the right hand side of (9.7), we get

$$\begin{aligned} &P\Big[|(n-p_n)^{-1}\sum_{t=p_n+1}^{n} x_{(t-j)k}x_{(t-u)v} - E[x_{(t-j)k}x_{(t-u)v}]| > \varepsilon\Big] \\ &\le \varepsilon^{-2s} O(p_n^s n^{-s}) = O(n^{-s\kappa}), \end{aligned}$$

since $p_n = O(n^{1-\kappa})$ by assumption. For the supremum over the different lags and components we then get

$$\begin{aligned} &P\Big[\sup_{1\le j,u\le p_n, 1\le k,v\le q_n}\Big|(n-p_n)^{-1}\sum_{t=p_n+1}^{n} x_{(t-j)k}x_{(t-u)v} - E[x_{(t-j)k}x_{(t-u)v}]\Big| > \varepsilon\Big] \\ &= O(p_n^2 q_n^2 n^{-s\kappa}) = O(n^{2(1+\delta)-(s+2)\kappa}). \end{aligned}$$

Hence, since $\kappa > 2(1+\delta)/(s+2)$, we have proved that there exists a $c > 0$ such that

$$\sup_{1\le j,u\le p_n, 1\le k,v\le q_n}\Big|(n-p_n)^{-1}\sum_{t=p_n+1}^{n} x_{(t-j)k}x_{(t-u)v} - E[x_{(t-j)k}x_{(t-u)v}]\Big| = O_P(n^{-c}).$$

The version of Lemma 1 (ii) follows analogously; the versions of Lemma 1 (iii) and (iv) can be proved exactly as in Lemma 1.

Acknowledgement

We thank the guest editors Yi Lin and Xiaotong Shen for their invitation to contribute to this special issue.

References

Bedrick, E. J. and Tsai, C.-L. (1994). Model selection for multivariate regression in small samples. *Biometrics* **50**, 226-231.

Box, G. and Youle, P. (1955). The exploration of response surfaces: an example of the link between the fitted surface and the basic mechanism of the system. *Biometrics* **11**, 287-323.

Breiman, L. (1998). Arcing classifier. *Ann. Statist.* **26**, 801-824.

Breiman, L. (1999). Prediction games and arcing algorithms. *Neural Comput.* **11**, 1493-1517.

Breiman, L. and Friedman, J. H. (1997). Predicting multivariate responses in multiple linear regression (Disc: p37-54). *J. Roy. Statist. Soc. Ser. B* **59**, 3-37.

Brooks, R. and Stone, M. (1994). Joint continuum regression for multiple predictands. *J Amer. Statist. Assoc.* **89**, 1374-1377.

Bühlmann, P. (2006). Boosting for high-dimensional linear models. *Ann. Statist.* **34**, to appear.

Bühlmann, P. and Yu, B. (2003). Boosting with the L2-loss: Regression and classification. *J Amer. Statist. Assoc.* **98**, 324-339.

Bühlmann, P. and Yu, B. (2005). Boosting, model selection, lasso and nonnegative garrote. Technical Report 127, Seminar für Statistik ETH Zürich.

Dahlhaus, R. and Eichler, M. (2003). Causality and graphical models for time series. *In Highly Structured Stochastic Systems* (Edited by P. Green, N. Hjort and S. Richardson). Oxford University Press.

Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004). Least angle regression. *Ann. Statist.* **32**, 407-451.

Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. *Proceedings of the Thirteenth International Conference on Machine Learning*, 148-156.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Ann. Statist.* **29**, 1189-1232.

Friedman, J., Hastie, T. and Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting. *Ann. Statist.* **28**, 337-373.

Hurvich, C. M. and Tsai, C.-L. (1989). Regression and time series model selection in small samples. *Biometrika* **76**, 297-307.

Klein, L., Ball, R., Hazlewood, A. and Vandome, P. (1961). *An Economic Model of the United Kingdom.* Blackwell, Oxford.

Kriegel, H.-P., Kroger, P., Pryakhin, A. and Schubert, M. (2004). Using support vector machines for classifying large sets of multi-represented objects. *Proc. 4th SIAM Int. Conf. on Data Mining*, 102-114.

Lütkepohl, H. (1993). *Introduction to Multiple Time Series Analysis.* 2nd edition. Springer-Verlag, New York.

Mallat, S. and Zhang, Z. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Processing* **41**, 3397-3415.

Reinsel, G. C. and Velu, R. P. (1998). *Multivariate Reduced-rank Regression: Theory and Applications.* Springer-Verlag, New York.

Rencher, A. C. (2002). *Methods of Multivariate Analysis.* Wiley, New York.

Skagerberg, B., MacGregor, J. and Kiparissides, C. (1992). Multivariate data analysis applied to low-density polyethylene reactors. *Chemometr. Intell. Lab. Syst.* **14**, 341-356.

Temlyakov, V. (2000). Weak greedy algorithms. *Adv. Comput. Math.* **12**, 213-227.

Timm, N. H. (2002). *Applied Multivariate Analysis.* Springer-Verlag, New York.

Wille, A., Zimmermann, P., Vranova, E., Fürholz, A., Laule, O., Bleuler, S., Hennig, L., Prelic, A., von Rohr, P., Thiele, L., Zitzler, E., Gruissem, W. and Bühlmann, P. (2004). Sparse graphical gaussian modeling of the isoprenoid gene network in arabidopsis thaliana. *Genome Biology* **5**, 1-13.

Yokoyama, R. (1980). Moment bounds for stationary mixing sequences. *Z. Wahrscheinlichkeitstheorie verw. Gebiete* **52**, 45-75.

Zellner, A. (1962). An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. *J. Amer. Statist. Assoc.* **57**,348-368.

Zellner, A. (1963). Estimators for seemingly unrelated regression equations: Some exact finite sample results (corr: V67, p255). *J. Amer. Statist. Assoc.* **58**, 977-992.

Seminar für Statistik, ETH Zurich, CH-8092 Zurich, Switzerland.

E-mail: lutz@stat.math.ethz.ch

Seminar für Statistik, ETH Zurich, CH-8092 Zurich, Switzerland.

E-mail: buhlmann@stat.math.ethz.ch

(Received April 2005; accepted August 2005)