Timo Similä and Jarkko Tikka. 2007. Input selection and shrinkage in multiresponse linear regression. Computational Statistics & Data Analysis, volume 52, number 1, pages 406-422.

© 2007 Elsevier Science

Reprinted with permission from Elsevier.

Available online at www.sciencedirect.com
ScienceDirect
Computational Statistics & Data Analysis 52 (2007) 406–422
COMPUTATIONAL STATISTICS & DATA ANALYSIS
www.elsevier.com/locate/csda

Input selection and shrinkage in multiresponse linear regression

Timo Similä*, Jarkko Tikka

Laboratory of Computer and Information Science, Helsinki University of Technology, P.O. Box 5400, FI-02015 HUT, Helsinki, Finland

Available online 17 February 2007

Abstract

The regression problem of modeling several response variables using the same set of input variables is considered. The model is linearly parameterized and the parameters are estimated by minimizing the error sum of squares subject to a sparsity constraint. The constraint has the effect of eliminating useless inputs and constraining the parameters of the remaining inputs in the model. Two algorithms for solving the resulting convex cone programming problem are proposed. The first algorithm gives a pointwise solution, while the second one computes the entire path of solutions as a function of the constraint parameter. Based on experiments with real data sets, the proposed method has a similar performance to existing methods. In simulation experiments, the proposed method is competitive both in terms of prediction accuracy and correctness of input selection. The advantages become more apparent when many correlated inputs are available for model construction.
© 2007 Elsevier B.V. All rights reserved.

Keywords: Subset selection; Variable selection; Constrained regression; Multivariate regression; Convex optimization; Cone programming

1. Introduction

Multiresponse regression is the task of estimating several response variables using a common set of input variables. There are two approaches to the problem. Either a separate model is built for each response variable, or a single model is used to estimate all the responses simultaneously. Breiman and Friedman (1997) and Srivastava and Solanky (2003) present simultaneous estimation techniques that have advantages over the separate model building, especially when the responses are correlated. Correlation among the responses is typical in many applications, for instance, in the field of chemometrics (Burnham et al., 1999). In this article, the focus is on linear simultaneous models.

Many input variables are usually available for model construction. However, some of the inputs may be weakly correlated with the responses and some others may be redundant in that they are highly correlated with the other inputs. A small number of observations compared to the number of inputs causes the problem of *overfitting*: the model fits well on training data but generalizes poorly. Highly correlated inputs cause the problem of *collinearity*: model interpretation is misleading as the importance of an input in the model can be compensated by another input. Traditional methods for meeting these problems are pure input selection (Sparks et al., 1985), regularization or shrinking (Breiman and Friedman, 1997; Srivastava and Solanky, 2003), and subspace methods (Abraham and Merola, 2005). Shrinking means that the regression coefficients are constrained such that the unimportant inputs tend to have coefficient values close

* Corresponding author. Tel.: +358 9 4513360; fax: +358 9 4513277.
E-mail addresses: timo.simila@hut.fi (T. Similä), tikka@cis.hut.fi (J. Tikka).

0167-9473/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.csda.2007.01.025

to zero. In the subspace approach the data are projected onto a smaller subspace in which the model is fitted. Input selection differs from the two other techniques as some of the inputs are completely left out of the model.

Practical benefits of input selection include aid in model interpretation, economic efficiency if measured inputs have costs, and computational efficiency due to simplicity of the model. Commonly used criteria for input selection are tests of statistical significance, information criteria, and prediction error (Bedrick and Tsai, 1994; Barrett and Gray, 1994; Sparks et al., 1985). These criteria only rank combinations of inputs and some greedy stepwise method is typically applied to find promising combinations. However, the greedy stepwise methods may fail to recognize important combinations of inputs, especially when the inputs are highly correlated (Derksen and Keselman, 1992). Better results can be obtained by incorporating shrinking in the selection strategy (Breiman, 1996; Similä and Tikka, 2006). Bayesian methods offer another approach (Brown et al., 2002), which is theoretically sound but may be a bit technical from a practical point of view. Recently, more straightforward methods have emerged in the statistical and signal processing communities, apparently through independent research efforts (Turlach et al., 2005; Cotter et al., 2005; Malioutov et al., 2005; Tropp, 2006). These methods either constrain or penalize the model fitting in a way that input selection and shrinking occur simultaneously. As a common denominator, the estimation is formulated as a single convex optimization problem. From now on, the family of this type of methods is called as simultaneous variable selection (SVS).

We consider a SVS method, which is used in the signal processing community (Cotter et al., 2005; Malioutov et al., 2005). The importance of an input in the model is measured by the 2-norm of the regression coefficients associated with the input, and that is why the method is denoted by L_2-SVS. The error sum of squares is minimized while constraining the sum of the importances over all the input variables. We also discuss a variant of SVS, where the ∞-norm is used instead of the 2-norm. L_∞-SVS is proposed by Turlach et al. (2005) in the statistical and Tropp (2006) in the signal processing community. The main contributions of this article are a formal analysis of the L_2-SVS problem and a numerical solver, which takes advantage of the structure of the problem. Furthermore, we present an efficient algorithm for following the path of solutions as a function of the constraint parameter. The existing SVS articles do not consider the solution path, although it is highly useful in practical problems, where the constraint parameter must be fixed by cross-validation or related techniques.

The rest of this article is organized as follows. In Section 2, we introduce the L_2-SVS estimate and position it with respect to related research. In Section 3, we derive the optimality conditions and propose algorithms for solving the L_2-SVS problem. Two types of comparisons are presented in Section 4. Firstly, several real world data sets are analyzed. Secondly, simulation experiments are carried out to explore the effect of collinearity among the input variables. Section 5 concludes the article.

2. L_2-SVS estimate and discussion of related work

Suppose that we have q response variables and m input variables from which we have n observations. The response data are denoted by an $n \times q$ matrix $\boldsymbol{Y}$ and the input data by an $n \times m$ matrix $\boldsymbol{X}$. All the variables are assumed to have zero means and similar scales. We focus on a linear model

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}^* + \boldsymbol{E}, \tag{1}$$

where $\boldsymbol{W}^*$ is an $m \times q$ matrix of regression coefficients and $\boldsymbol{E}$ is an $n \times q$ matrix of noise. Some rows of the matrix $\boldsymbol{W}^*$ are assumed to have only zero elements, which means that the corresponding input variables do not contribute to the response variables at all. In order to simplify the following notation, we further define

$$\boldsymbol{X} = [\boldsymbol{x}_1 \cdots \boldsymbol{x}_m] = [\underline{\boldsymbol{x}}_1 \cdots \underline{\boldsymbol{x}}_n]^{\mathrm{T}}, \quad \boldsymbol{Y} = [\boldsymbol{y}_1 \cdots \boldsymbol{y}_q] = [\underline{\boldsymbol{y}}_1 \cdots \underline{\boldsymbol{y}}_n]^{\mathrm{T}},$$
$$\boldsymbol{W} = [\boldsymbol{w}_1 \cdots \boldsymbol{w}_q] = [\underline{\boldsymbol{w}}_1 \cdots \underline{\boldsymbol{w}}_m]^{\mathrm{T}}, \quad \boldsymbol{E} = [\boldsymbol{e}_1 \cdots \boldsymbol{e}_q] = [\underline{\boldsymbol{e}}_1 \cdots \underline{\boldsymbol{e}}_n]^{\mathrm{T}},$$

where each bolded lower-case letter refers to a column vector.

We estimate $\boldsymbol{W}^*$ by minimizing the error sum of squares subject to a sparsity constraint, namely by solving the L_2-SVS problem

$$\underset{\boldsymbol{W}}{\text{minimize}}\ f(\boldsymbol{W}) \quad \text{subject to}\ g(\boldsymbol{W}) \leqslant r, \quad \text{where}\ f(\boldsymbol{W}) = \tfrac{1}{2}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W}\|_{\mathrm{F}}^2 \quad \text{and} \quad g(\boldsymbol{W}) = \sum_{j=1}^{m} \|\underline{\boldsymbol{w}}_j\|_2. \tag{2}$$

Here, the subscript F denotes the Frobenius norm, that is $\|\boldsymbol{B}\|_{\mathrm{F}}^2 = \sum_{ij} b_{ij}^2$. A large enough value of $r \geqslant 0$ has the consequence that the optimal $\boldsymbol{W}$ equals to the ordinary least squares (OLS) solution. Small values of r impose a row-sparse structure on $\boldsymbol{W}$, which means that only some of the inputs are effective in the estimate. The factor $\|\underline{\boldsymbol{w}}_j\|_2$ is a natural measure of the importance of the jth input in the model. The constraint takes a 1-norm of the factors, and this choice of the norm is known to encourage sparsity (Tibshirani, 1996). Thus, some of the factors $\|\underline{\boldsymbol{w}}_j\|_2$ can be exactly zero, which is not possible for standard regularization techniques including the multiresponse ridge regression (Brown and Zidek, 1980). Using the Lagrange multiplier theory and convexity of $f(\boldsymbol{W})$ and $g(\boldsymbol{W})$, it can be shown that (2) is equivalent to the minimization of the penalized loss function $f(\boldsymbol{W}) + \lambda g(\boldsymbol{W})$ in the sense that for any $\lambda \geqslant 0$ there exists a $r \geqslant 0$ such that the two problems share the same solution, and vice versa (Hiriart-Urruty and Lemaréchal, 1996). Some of the related methods are presented in the penalized form.

If we have $q = 1$, then (2) reduces to the LASSO problem (Tibshirani, 1996), which can be solved efficiently (Osborne et al., 2000; Efron et al., 2004). The case $q \geqslant 2$ is analyzed by Malioutov et al. (2005) who propose the use of second-order cone programming (SOCP) for parameter estimation. As a numerical solver, they apply the SeDuMi software package (Sturm, 1999). We use this approach as a benchmark for our implementation. Cotter et al. (2005) propose both forward selection algorithms and also methods for a direct minimization of penalized loss functions. Particularly, their regularized M-FOCUSS algorithm appears to be applicable to the penalized version of (2) for $q \geqslant 2$, but no rigorous proof of convergence is given. Turlach et al. (2005) discuss (2) for $q \geqslant 2$ and conclude that it cannot be handled effectively with currently available optimization techniques. Similä and Tikka (2006) propose the 2-norm multiresponse sparse regression (L_2-MRSR) algorithm, which is an extension of the least angle regression (LARS) algorithm (Efron et al., 2004) to many responses. This algorithm gives the whole path of solutions to (2) under an orthonormality assumption on $\boldsymbol{X}$, but it only approximates the path in a general case.

The objective function in (2) can be transformed into a single response form

$$f(\boldsymbol{W}) = \tfrac{1}{2}\|\mathrm{vec}(\boldsymbol{Y}) - \mathrm{diag}(\boldsymbol{X}, \ldots, \boldsymbol{X})\,\mathrm{vec}(\boldsymbol{W})\|_2^2, \tag{3}$$

where $\mathrm{diag}(\cdot)$ produces a $qn \times qm$ block diagonal matrix and $\mathrm{vec}(\cdot)$ produces a long vector by stacking the columns of a matrix. Suppose that we form m groups, each of which contains the q regression coefficients associated with one of the columns of $\boldsymbol{X}$. Now the single response form (3) connects (2) to the grouped variable selection technique proposed by Bakin (1999). Yuan and Lin (2006) call the resulting estimate as the Group LASSO and propose algorithms for approximating the solution path. Particularly, their Group LARS algorithm applied to (2) equals to the L_2-MRSR algorithm. Kim et al. (2006) propose a gradient projection method for optimization and apply it to different loss functions with the Group LASSO constraint. Meier et al. (2006) focus on the Group LASSO estimate for logistic regression and propose a block coordinate gradient descent method for optimization. Park and Hastie (2006) study generalized linear models and propose a path following algorithm for the Group LASSO estimate. Lin and Zhang (2006) select components in the smoothing spline analysis of variance model in a way that resembles the Group LASSO type of penalization. Zhao et al. (2006) analyze overlapping groupings, different group-norms, and different ways of aggregating the norms such that the resulting penalty function evokes variable selection in a grouped fashion. Most of the grouped selection methods are directly applicable to (2), but they are unlikely to be efficient, if (3) is used without considering the special structure of our multiresponse regression problem.

The L_∞-SVS estimate (Turlach et al., 2005; Tropp, 2006) for $\boldsymbol{W}^*$ is given by the solution to the problem

$$\underset{\boldsymbol{W}}{\text{minimize}}\ f(\boldsymbol{W}) \quad \text{subject to} \quad \sum_{j=1}^{m} \|\underline{\boldsymbol{w}}_j\|_\infty \leqslant r. \tag{4}$$

This can be rewritten as a convex linearly constrained quadratic optimization problem and thereby it can be solved with primal-dual interior point methods (Boyd and Vandenberghe, 2004). Furthermore, it appears to be possible to compute the complete path of solutions to (4) (Turlach, 2005, 2006). This kind of an algorithm is presumably more efficient than the interior point methods. Another related work is by Zou and Yuan (2005) who apply the sum of ∞-norms penalty function to select groups of variables in a support vector machine for classification.

3. Formal analysis and numerical aspects of the L_2-SVS estimate

In this section, we derive the optimality conditions for the L_2-SVS problem (2) using a subdifferential calculus for convex nonsmooth optimization. Additionally, we give a condition, which guarantees the uniqueness of the solution. We also propose a convergent algorithm for solving (2) given a fixed value of r and then we extend this algorithm for following the path of solutions as a function of r.

3.1. Optimality conditions

Both $f(\boldsymbol{W})$ and $g(\boldsymbol{W})$ are convex functions, so any local minimizer of $f(\boldsymbol{W})$ on the feasible set is also a global minimizer. The Karush–Kuhn–Tucker (KKT) conditions are

$$-\nabla f(\boldsymbol{W}) \in \lambda \partial g(\boldsymbol{W}), \quad \lambda \geqslant 0, \quad \lambda(g(\boldsymbol{W}) - r) = 0. \tag{5}$$

These conditions are necessary and sufficient for optimality, because the weak Slater assumption holds (Hiriart-Urruty and Lemaréchal, 1996). This can be verified by $\boldsymbol{W} = \boldsymbol{0}_{m\times q}$, which is strictly feasible for $r > 0$ and it is the sole solution for $r = 0$. The objective is differentiable everywhere and the gradient is

$$\nabla f(\boldsymbol{W}) = -\mathrm{vec}(\boldsymbol{X}^{\mathrm{T}}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})). \tag{6}$$

The constraint is, however, nondifferentiable at the points, where $\underline{\boldsymbol{w}}_j = \boldsymbol{0}_{q\times 1}$ holds for some $j = 1, \ldots, m$. Since $g(\boldsymbol{W})$ is a positive combination of convex functions, its subdifferential can be factored as follows (Hiriart-Urruty and Lemaréchal, 1996):

$$\partial g(\boldsymbol{W}) = \sum_{j=1}^{m} \partial \|\underline{\boldsymbol{w}}_j\|_2. \tag{7}$$

The subdifferential of the jth factor is $\partial\|\underline{\boldsymbol{w}}_j\|_2 = \mathrm{vec}(\boldsymbol{u}_j \boldsymbol{d}_j^{\mathrm{T}})$, where the $m \times 1$ unit coordinate vector $\boldsymbol{u}_j$ and the set $\boldsymbol{d}_j \subset \mathbb{R}^{q\times 1}$ have the elements

$$u_{ji} = \begin{cases} 1, & i = j, \\ 0, & i \neq j, \end{cases} \qquad \boldsymbol{d}_j = \begin{cases} \underline{\boldsymbol{w}}_j / \|\underline{\boldsymbol{w}}_j\|_2, & \underline{\boldsymbol{w}}_j \neq \boldsymbol{0}_{q\times 1}, \\ \{\boldsymbol{d} \in \mathbb{R}^{q\times 1} : \|\boldsymbol{d}\|_2 \leqslant 1\}, & \underline{\boldsymbol{w}}_j = \boldsymbol{0}_{q\times 1}. \end{cases}$$

Members of the set $\partial g(\boldsymbol{W})$ are $qm \times 1$ subgradients.

The KKT conditions can be transformed into a more tractable form by substituting (6) and (7) into (5)

$$\begin{aligned} &(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})^{\mathrm{T}}\boldsymbol{x}_j = \lambda \underline{\boldsymbol{w}}_j / \|\underline{\boldsymbol{w}}_j\|_2, \quad j \in \{j : \underline{\boldsymbol{w}}_j \neq \boldsymbol{0}_{q\times 1}\}, \\ &\|(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})^{\mathrm{T}}\boldsymbol{x}_j\|_2 \leqslant \lambda, \quad j \in \{j : \underline{\boldsymbol{w}}_j = \boldsymbol{0}_{q\times 1}\}, \\ &\lambda \geqslant 0, \quad \lambda \left(\sum_{j=1}^{m} \|\underline{\boldsymbol{w}}_j\|_2 - r \right) = 0. \end{aligned}$$

Note that the ith entry of the vector $(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})^{\mathrm{T}}\boldsymbol{x}_j$ measures the correlation between the jth input and residuals associated with the ith response. The Lagrange multiplier λ, in turn, measures the maximum overall correlation in the 2-norm sense

$$\lambda = \max_{1 \leqslant j \leqslant m} \|(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})^{\mathrm{T}}\boldsymbol{x}_j\|_2. \tag{8}$$

Clearly, some of the inputs share the same value λ, while the others have weaker correlations. It is useful to group the indices of the maximally correlated inputs. So more formally, we define an active set

$$\mathscr{A} = \{j : \|(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})^{\mathrm{T}}\boldsymbol{x}_j\|_2 = \lambda\}. \tag{9}$$

If $\underline{\boldsymbol{w}}_j \neq \boldsymbol{0}_{q\times 1}$ holds, then we have $j \in \mathscr{A}$, but the converse does not necessarily apply. In the case of $\lambda = 0$, we have the result $\nabla f(\boldsymbol{W}) = \boldsymbol{0}_{qm\times 1}$ from (5) indicating that $\boldsymbol{W}$ is the OLS solution.

Since the feasible set is nonempty and compact for any finite $r \geqslant 0$, and $f(\boldsymbol{W})$ is continuous, the Extreme Value Theorem says that a solution to (2) exists. If we have $r = 0$, then the unique solution is obviously $\boldsymbol{W} = \boldsymbol{0}_{m\times q}$. The solution is also unique for all $r > 0$ if the matrix $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ is positive definite, because then we minimize a strictly convex function subject to a convex constraint. Even if $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ is not positive definite, for instance if $m > n$, the following theorem is still holding, see the proof in Appendix A.1.

Theorem 1. *If $\boldsymbol{W}$ solves the L_2-SVS problem* (2) *and $\boldsymbol{X}_{\mathscr{A}}^{\mathrm{T}}\boldsymbol{X}_{\mathscr{A}}$ is positive definite, then $\boldsymbol{W}$ is the unique solution.*

We use $\boldsymbol{X}_{\mathscr{A}} = [\cdots \boldsymbol{x}_j \cdots]_{j\in\mathscr{A}}$ in Theorem 1 to denote the $n \times |\mathscr{A}|$ matrix of data from the active inputs. The positive definiteness condition holds, if the columns of $\boldsymbol{X}_{\mathscr{A}}$ are linearly independent. In practical problems, this condition is nearly always satisfied when $|\mathscr{A}| < n$ holds.

3.2. Pointwise solutions

In order to solve (2) given a fixed value of r, we may transform the problem into an epigraph form. This is a different problem, but gives the same solutions

$$\underset{\boldsymbol{W},s_0,\ldots,s_m}{\text{minimize}}\ s_0 \quad \text{subject to} \quad \|\mathrm{vec}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})\|_2 \leqslant s_0, \quad \sum_{j=1}^{m} s_j \leqslant r, \quad \|\underline{\boldsymbol{w}}_j\|_2 \leqslant s_j, \quad j = 1,\ldots,m. \tag{10}$$

Note that $\frac{1}{2}s_0^2$ is an upper bound for $f(\boldsymbol{W})$. Minimizing s_0 is, however, equivalent to minimizing $f(\boldsymbol{W})$, because the square root is a monotone increasing transformation (Boyd and Vandenberghe, 2004). At the optimum, the bound s_0 is always tight and the bounds s_j for $j = 1,\ldots,m$ are tight assuming that r is small enough to impose an active constraint in the original problem (2). The epigraph formulation (10) is a SOCP problem. It has a linear objective, $m + 1$ second-order (Lorentz) cone constraints, and a single linear inequality constraint. The SOCP problems have been studied actively and primal-dual interior point methods have been developed to solve them efficiently (Alizadeh and Goldfarb, 2003). Several software packages are available for the SOCP problems. For instance, Malioutov et al. (2005) use the SeDuMi package (Sturm, 1999) to solve a penalized version of (10).

Formulation (10) is suitable only for relatively low-dimensional problems, because it does not take enough advantage of the structure of the problem. For instance, it is necessary to store q copies of the matrix $\boldsymbol{X}$. Furthermore, general-purpose software packages do not offer efficient ways to solve (2) for a sequence of values of r, which is our main focus in Section 3.3. This motivates considering the problem

$$\underset{\boldsymbol{W},\boldsymbol{s}}{\text{minimize}}\ f(\boldsymbol{W}) \quad \text{subject to} \quad \sum_{j=1}^{m} s_j \leqslant r, \quad \|\underline{\boldsymbol{w}}_j\|_2 \leqslant s_j, \quad j = 1,\ldots,m, \tag{11}$$

in detail, which is yet another equivalent form for (2). The solution to (11) can be approximated by solving a logarithmic barrier reformulation

$$\underset{\boldsymbol{W},\boldsymbol{s}}{\text{minimize}}\ F_\mu(\boldsymbol{W},\boldsymbol{s}), \quad \text{where } \mu > 0 \quad \text{and}$$
$$F_\mu(\boldsymbol{W},\boldsymbol{s}) = f(\boldsymbol{W}) - \frac{\mu}{m}\sum_{j=1}^{m}\log(s_j^2 - \|\underline{\boldsymbol{w}}_j\|_2^2) - \mu\log\left(r - \sum_{j=1}^{m} s_j\right). \tag{12}$$

The objective here is convex and differentiable, so the necessary and sufficient conditions for optimality are

$$\frac{\partial F_\mu(\boldsymbol{W},\boldsymbol{s})}{\partial \underline{\boldsymbol{w}}_j} = -(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})^{\mathrm{T}}\boldsymbol{x}_j + \eta_j\underline{\boldsymbol{w}}_j = \boldsymbol{0}_{q\times 1}, \quad j = 1,\ldots,m,$$
$$\frac{\partial F_\mu(\boldsymbol{W},\boldsymbol{s})}{\partial s_j} = \lambda' - \eta_j s_j = 0, \quad j = 1,\ldots,m. \tag{13}$$

Algorithm 1 (*Barrier method*).

Given strictly feasible starting point $(\boldsymbol{W}, \boldsymbol{s})$, tolerance $\varepsilon_{\min} > 0$, initial value $\mu := \mu_{\max} > 0$, final value $\mu_{\min} \in (0, \mu_{\max}]$, and factor $\alpha \in (0, 1)$.
repeat
$\mathrm{vec}([\widehat{\boldsymbol{W}}|\widehat{\boldsymbol{s}}]) = -\nabla^2 F_\mu(\boldsymbol{W}, \boldsymbol{s})^{-1} \nabla F_\mu(\boldsymbol{W}, \boldsymbol{s})$
$\varepsilon = -\nabla F_\mu(\boldsymbol{W}, \boldsymbol{s})^{\mathrm{T}} \mathrm{vec}([\widehat{\boldsymbol{W}}|\widehat{\boldsymbol{s}}])$
quit if $\varepsilon < \varepsilon_{\min}$ and $\mu = \mu_{\min}$
$\gamma = \max\{\gamma \leqslant 1 : \sum_{j=1}^m (s_j + \gamma \widehat{s}_j) < r,\ s_j + \gamma \widehat{s}_j > \|\underline{\boldsymbol{w}}_j + \gamma \widehat{\underline{\boldsymbol{w}}}_j\|_2,\ \forall j\}$
$\boldsymbol{W} := \boldsymbol{W} + \gamma \widehat{\boldsymbol{W}}, \quad \boldsymbol{s} := \boldsymbol{s} + \gamma \widehat{\boldsymbol{s}}, \quad \mu := \max\{\alpha\mu,\ \mu_{\min}\}$

In addition, we must have $s_j > \|\underline{\boldsymbol{w}}_j\|_2$ and $\sum_{j=1}^m s_j < r$ to ensure that the arguments of the logarithms are positive in (12). For simplicity, we use the notations $\lambda' = \mu/(r - \sum_{j=1}^m s_j)$ and $\eta_j = 2\mu/(m(s_j^2 - \|\underline{\boldsymbol{w}}_j\|_2^2))$ in (13). The parameter λ' is an estimate of λ, which is defined in (8).

As the parameter μ decreases, the approximation becomes more accurate. Boyd and Vandenberghe (2004) present a theoretical framework for analyzing the accuracy of logarithmic barrier reformulations. In order to follow this framework, suppose that $\boldsymbol{W}$ and $\boldsymbol{s}$ solve (12). Moreover, λ' is defined by $\boldsymbol{s}$, and η_j is defined by $\underline{\boldsymbol{w}}_j$ and s_j. Then $(-\eta_j \underline{\boldsymbol{w}}_j,\ \eta_j s_j)$ is a dual feasible pair of Lagrange multipliers associated with the jth cone constraint in (11), and λ' is a dual feasible Lagrange multiplier associated with the linear inequality constraint in (11). As an important consequence, the dual function of (11) can be evaluated and we get a duality gap 3μ. This means that $f(\boldsymbol{W})$ evaluated at the solution to (12) differs at most by 3μ from the optimal value.

A standard technique for computing numerical solutions is the barrier method (Boyd and Vandenberghe, 2004). Algorithm 1 sketches the method for solving (11). For a sequence of decreasing values of μ, Newton steps are applied to minimize $F_\mu(\boldsymbol{W}, \boldsymbol{s})$. We take only a single Newton step per each of the early values of μ, but it is also possible to take several ones. As μ reaches the value $\mu_{\min}$, the parameter $\varepsilon_{\min}$ guarantees that the Newton steps are applied until convergence. We measure convergence by the squared Newton decrement. Convergence is rapid in general, and quadratic near the solution (Boyd and Vandenberghe, 2004). The step length rule returns a unit length whenever this is possible, but it always keeps the next iterate within the interior of the feasible region. The rule can be easily applied by solving a single linear and m quadratic equations of γ. Appendix A.2 instructs how to compute the Newton step efficiently. Although we have $(q + 1)m$ parameters to be optimized, inverting the Hessian matrix of $F_\mu(\boldsymbol{W}, \boldsymbol{s})$ means essentially working with the inverses of two $m \times m$ matrices. Moreover, we need to store only single copies of the matrices $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ and $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Y}$.

3.3. Solution path

The L_2-SVS problem (2) has a single free parameter r, which controls the amount of shrinkage that is applied to the estimate. Since this parameter needs to be tuned in practical problems, we focus now on solving (2) efficiently for a sequence of values of r. In the case of $q = 1$, the LASSO estimate (Tibshirani, 1996) solves (2). It is well known by the work of Osborne et al. (2000) and Efron et al. (2004) that the solution $\boldsymbol{W}$ is then a piecewise linear function of the constraint parameter r. If the inputs are orthonormal, or in other words $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ is an identity matrix, the solution path is again piecewise linear for any $q \geqslant 2$ (Similä and Tikka, 2006; Yuan and Lin, 2006). Piecewise linearity enables efficient and fast algorithmic solutions. Unfortunately, the solution path is nonlinear for a general $\boldsymbol{X}$ and $q \geqslant 2$. Thus, the design of an exact incremental path following algorithm is substantially more difficult. We present an approximate method in this section.

Suppose, without loss of generality, that $\lambda^{[0]} := \|\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{x}_k\|_2 > \|\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{x}_j\|_2$ holds for all $j \neq k$. Then the active set is a singleton $\mathscr{A} = \{k\}$ given the solution $\boldsymbol{W}(0) = \boldsymbol{0}_{m \times q}$ according to (9). It can be shown that the solution path continues as follows:

$$\lambda(r) = \lambda^{[0]} - r\|\boldsymbol{x}_k\|_2^2 \quad \text{and} \quad \underline{\boldsymbol{w}}_j(r) = \begin{cases} (r/\lambda^{[0]})\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{x}_k, & j = k, \\ \boldsymbol{0}_{q \times 1}, & j \neq k \end{cases} \tag{14}$$

Algorithm 2 (*Path following method*).

Given sequence $r^{[2]} < \cdots < r^{[t_{\max}]}$ and parameters $\mu_{\min}$, $\varepsilon_{\min}$, α as presented in Algorithm 1.
Compute $\lambda^{[1]} := \lambda(r^{[1]})$ and $\boldsymbol{W}^{[1]} := \boldsymbol{W}(r^{[1]})$ according to (14) and initialize
$s_j^{[1]} := \|\underline{\boldsymbol{w}}_j^{[1]}\|_2 + 10^{-4}$ for $j = 1, \ldots, m$. Set $t = 1$ and define $\mathscr{A}$ according to (9).
while $t < t_{\max}$

$$\widehat{v}_j = \frac{2s_j^{[t]}}{m((s_j^{[t]})^2 - \|\underline{\boldsymbol{w}}_j^{[t]}\|_2^2)}, \quad v_j = \frac{\widehat{v}_j r^{[t+1]}}{1 + \sum_{j\in\mathscr{A}} \widehat{v}_j s_j^{[t]}}, \quad j \in \mathscr{A},$$

$\mu_{\max} = \lambda^{[t]}(r^{[t+1]} - \sum_{j\in\mathscr{A}} v_j s_j^{[t]})$.
Run Algorithm 1 in the $\mathscr{A}$-subspace such that $v_j \underline{\boldsymbol{w}}_j^{[t]}$, $v_j s_j^{[t]}$, $j \in \mathscr{A}$ is the starting point at $r^{[t+1]}$ in order to get the solution $\boldsymbol{W}_{\mathscr{A}}^{[t+1]}$, $\boldsymbol{s}_{\mathscr{A}}^{[t+1]}$. Compute $\lambda^{[t+1]}$ according to (8).
$\mathscr{A} = \{j : \|(\boldsymbol{Y} - \boldsymbol{X}_{\mathscr{A}}\boldsymbol{W}_{\mathscr{A}}^{[t+1]})^{\mathrm{T}}\boldsymbol{x}_j\|_2 > \lambda^{[t+1]} - 0.1\lambda^{[1]}\}$
$t := t + 1$

if $0 \leqslant r \leqslant r^{[1]}$. The upper bound $r^{[1]}$ is the smallest positive value such that some new index joins the set $\mathscr{A}$. This value equals the point where the curves $\lambda(r)$ and $\|(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W}(r))^{\mathrm{T}}\boldsymbol{x}_j\|_2$ intersect for some $j \neq k$ for the first time. Appendix A.3 instructs how to compute $r^{[1]}$ analytically. Note that the initial linear step (14) follows the solution path exactly.

The solution path is piecewise smooth but nonlinear for $r \geqslant r^{[1]}$. At first glance, all that is needed is to follow each piece and detect the kinks where the active set changes. Such techniques are considered, for instance, by Rakowska et al. (1991) and Rosset (2005). However, it is often difficult to decide when a new input should enter to or an old one leave from the model due to numerical reasons. A more stable approach has been presented by Bach et al. (2005) in the context of learning multiple kernels. The original problem is approximated by a logarithmic barrier reformulation and the duality gap is held fixed. The next solution is predicted and Newton steps are applied to correct the prediction. This kind of an estimate for the path is smooth everywhere, but it still enables to restrict computations to a subset of parameters. Our path following method, sketched in Algorithm 2, is based on this technique.

We prefer to approximate the solution path at a pre-fixed sequence of points $r^{[2]} < \cdots < r^{[t_{\max}]}$, because it simplifies cross-validation and other analysis of the model. If the sequence is dense enough, the current solution is a good starting point for the next one, but this may not be the case when a loose sequence is used. In general, it is not reasonable to keep the duality gap parameter μ fixed to a small value unless the starting point is good (Boyd and Vandenberghe, 2004). Fortunately, the current solution at $r^{[t]}$ can be improved by a simple scaling procedure. We consider factors $v_j > 0$ such that $v_j \underline{\boldsymbol{w}}_j^{[t]}$, $v_j s_j^{[t]}$, $\forall j$ represents the starting point for Algorithm 1 at $r^{[t+1]}$. Now the condition $\lambda' = \eta_j s_j$ from (13) has the form

$$\frac{\mu_{\max}}{r^{[t+1]} - \sum_{j=1}^m v_j s_j^{[t]}} = \frac{2\mu_{\max} v_j s_j^{[t]}}{m((v_j s_j^{[t]})^2 - \|v_j \underline{\boldsymbol{w}}_j^{[t]}\|_2^2)}, \quad j = 1, \ldots, m,$$

which determines the factors v_j uniquely. Assuming that $\lambda^{[t+1]}$ is close to $\lambda^{[t]}$, we can determine the value

$$\mu_{\max} = \lambda^{[t]} \left(r^{[t+1]} - \sum_{j=1}^m v_j s_j^{[t]} \right).$$

This is a safe overestimate, because we have $r^{[t+1]} > r^{[t]}$ and this implies very likely that $\lambda^{[t+1]} < \lambda^{[t]}$ holds. The procedure works well even for quite loose sequences. The denser the sequence the less iterations of Algorithm 1 is needed per a single value of r. By using the terminology in the literature of numerical analysis (Allgower and Georg, 1993), we can say that the scaling corresponds to a predictor step and Algorithm 1 performs corrector steps. When the corrector steps have converged, the solution is no more than $3\mu_{\min}$ suboptimal.

Computational savings can be achieved by working in a subspace of the parameters. Note that the set $\mathscr{A}$ in Algorithm 2 includes not only the purely active indices according to (9), but also some other potential candidates due to the safety margin $0.1\lambda^{[1]}$. This ensures that we do not miss any change in the purely active set. On the other hand, $\|\underline{\boldsymbol{w}}_j^{[t]}\|_2$ and

$s_j^{[t]}$ are always close to zero if j is nonactive. One could also consider a relative safety margin, where the norm of the correlation vector is compared to a factor of the form $0.9\lambda^{[t+1]}$. However, the constant safety margin works well for the whole range of values, but the relative margin may become unstable for small values of λ.

It is rather difficult to evaluate the total running time complexity of Algorithm 2, because the required number of Newton steps is problem dependent and the density of the sequence $r^{[t]}$ has an influence. It is also hard to predict how the number of elements of the set $\mathscr{A}$ changes along the path. To illustrate the computational burden, we report the empirical running times for the experiments in Section 4.

4. Experiments

This section consists of experiments on real and simulated data sets. We compare three different methods, namely L_2-MRSR (Similä and Tikka, 2006), L_2-SVS (Eq. (2)), and L_∞-SVS (Eq. (4)). Motivated by the discussion of Turlach et al. (2005), we also compute so-called subset OLS solutions for the both SVS methods. Once the L_p-SVS problem has been solved, we identify the subset of effective indices $\{j : \|\underline{\boldsymbol{w}}_j\|_p > 10^{-3}\}$ and compute the OLS solution using only the corresponding inputs. This estimate is denoted by L_p-SVS & OLS. The L_2-SVS estimates are computed using Algorithm 2. The duality gap parameter is decreased by the factor $\alpha = 0.7$ after each Newton step until the parameter reaches the final value $\mu_{\min} = 10^{-3}$. Iteration is stopped when the squared Newton decrement goes below the value $\varepsilon_{\min} = 10^{-5}$. We implemented Algorithm 2 in MATLAB and the results are calculated on a 2.2 GHz AMD Opteron processor. It is worth noting that our code is potentially not fully optimized in terms of running time efficiency. This may favor the SeDuMi package, which is used as a benchmark.

4.1. Experiments on real data

The methods are applied to five real data sets. The first set is Tobacco data, given by Anderson and Bancroft (1952, p. 205). The data are measurements from different chemical components of tobacco leaves. There are $n=25$ observations of $q = 3$ responses and $m = 6$ inputs. The solution paths of the methods are illustrated in Fig. 1. L_2-MRSR produces a piecewise linear path as a function of a correlation criterion, which is identical with the definition of λ in (8). A new input variable enters the model at each breakpoint. The path is evaluated at 500 values of λ, which are linearly equally spaced in the range $[0, \lambda^{[0]}]$, but we present the path as a function of $g(\boldsymbol{W})$ to make it more comparable with L_2-SVS in Fig. 1. The paths of L_2-SVS and L_∞-SVS are evaluated at 500 linearly equally spaced points of r in the range

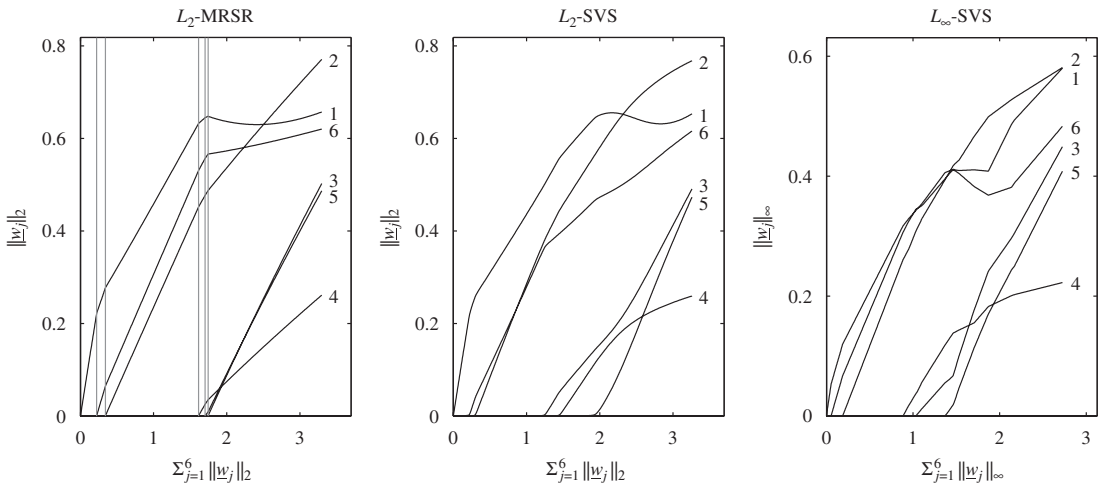

Fig. 1. Solution paths of the importance factors for Tobacco data. Vertical lines in the leftmost subfigure indicate the breakpoints of the L_2-MRSR algorithm. All paths end to the OLS solution.

$[0, r_{\mathrm{OLS}}]$. The value r_{OLS} corresponds to the constraint evaluated at the OLS solution. The first three inputs are selected in the order $\boldsymbol{x}_1, \boldsymbol{x}_6$, and $\boldsymbol{x}_2$ by all the methods. These inputs are obviously the most important and the same thing is also noticed, for instance, by Sparks et al. (1985) and Bedrick and Tsai (1994).

The second set is Chemometrics data, given by Srivastava (2002, pp. 15–17). The decimal point appears to be missing from the 37th observation of the 15th input variable. We replaced the value 19 203 with 1.9203. The data are from a simulation of a low density tubular polyethylene reactor. The set includes $n = 56$ observations of $m = 22$ inputs and $q = 6$ responses. Following Breiman and Friedman (1997), we log-transformed the responses, because they are skewed to the right. The third set is called Chemical reaction data, and it is given by Rencher (2002, p. 340). There are $n = 19$ observations of $q = 3$ responses and three inputs from a planned chemical reaction experiment. In this case, a quadratic model with all the terms x_j, x_j^2, and $x_i x_j$ is fitted, which gives the total number of inputs $m = 9$. The fourth set is Macroeconomic data, given by Reinsel and Velu (1998, p. 233). It is a 10-dimensional time-series from the United Kingdom with quarterly measurements. The data contain $n = 36$ observations of $q = 5$ responses and five inputs. The quadratic model is also used in this case, which gives the total number of inputs $m = 20$. The time-dependency of the observations is ignored. In the cases of Chemometrics, Chemical reaction, and Macroeconomic data the solution paths of L_2-MRSR are evaluated at 500 points of λ, which are logarithmically equally spaced in the range $[0.01, \lambda^{[0]}]$. L_2-SVS and L_∞-SVS are evaluated at 500 logarithmically equally spaced points of the parameter r in the range $[0.01, r_{\mathrm{OLS}}]$.

The fifth set is Sugar data, which is supplementary material for the work by Brown et al. (2002) and it can be obtained from the website of the publisher of their article. The response data correspond to a mixture of $q = 3$ sugars at different levels in aqueous solution. The input data correspond to the near-infrared absorbance spectra (second differenced), which are recorded from $m = 700$ wavelengths. We use $n = 125$ observations for model selection and separate $n = 21$ observations for model assessment, which agrees with the splitting reported by Brown et al. (2002). L_2-MRSR is evaluated at 500 logarithmically equally spaced points in the range $[0.1, \lambda^{[0]}]$, L_2-SVS at 500 linearly equally spaced points in the range $[0, 5]$, and L_∞-SVS at 500 linearly equally spaced points in the range $[0, 3]$. The inputs are highly correlated. For instance, 26% of the absolute correlations exceed the value 0.9 and the average is 0.6. This has the effect that the active set strategy in Algorithm 2 is rather conservative, since hundreds of extra indices may be in the active set. As a remedy, the active set for Sugar data contains the indices of the current nonzero rows of $\boldsymbol{W}$ supplemented with 60 extra indices, which have the highest ranks of $\|(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})^{\mathrm{T}}\boldsymbol{x}_j\|_2$ in the descending order.

All the responses and inputs were normalized to zero mean and unit variance before the analysis to make the results more comparable. Prediction accuracy of the methods is estimated by the average squared leave-one-out cross-validation error (CVE). The minimum CVE values are summarized in Table 1 and the average numbers of selected inputs in Table 2. All the subset methods are more accurate than the full OLS solution. In the case of Chemometrics data, the L_p-SVS estimates perform identically and they are the most precise. Their CVE value of 0.18 is slightly smaller than the values reported by Lutz and Bühlmann (2006) and clearly smaller than those by Breiman and Friedman (1997) and Srivastava and Solanky (2003). Shrinking without selection seems to work well for Chemometrics data. L_2-SVS performs best for Chemical reaction data and the CVE value of 0.46 is slightly smaller than the errors reported by Lutz and Bühlmann (2006). The subset OLS estimates are the best ones for Tobacco data and these estimates use the inputs $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_6$, which confirms the earlier analysis of their importance. L_∞-SVS is equally accurate, but it uses all the six inputs. In the case of Macroeconomic data, L_∞-SVS & OLS is the best choice, since it has the lowest CVE value with the

Table 1
Average errors with standard deviations for real data sets

	Chemomet.	Chem. react.	Tobacco	Macroecon.	Sugar$^{(1)}$	Sugar$^{(2)}$	Sugar$^{(3)}$
L_2-MRSR	0.22 (0.38)	0.51 (0.64)	0.45 (0.34)	0.24 (0.25)	0.12 (0.25)	0.45 (0.30)	0.43 (0.33)
L_2-SVS	0.18 (0.31)	0.46 (0.54)	0.43 (0.35)	0.22 (0.19)	0.10 (0.26)	0.41 (0.32)	0.40 (0.32)
L_2-SVS & OLS	0.22 (0.43)	0.52 (0.80)	0.41 (0.32)	0.24 (0.19)	0.09 (0.22)	0.40 (0.33)	0.23 (0.24)
L_∞-SVS	0.18 (0.31)	0.54 (0.69)	0.41 (0.32)	0.24 (0.21)	0.10 (0.26)	0.47 (0.35)	0.44 (0.36)
L_∞-SVS & OLS	0.23 (0.47)	0.50 (0.79)	0.41 (0.31)	0.21 (0.21)	0.10 (0.23)	0.53 (0.47)	0.32 (0.27)
Full OLS	0.41 (1.14)	1.34 (3.45)	0.48 (0.34)	0.50 (0.60)	—	—	—

Minimum CVE error (columns 2–6), test error of the minimum CVE model (7th column), the lowest possible test error (8th column). Division by 16 (variance of all the responses) converts the errors for Sugar data to normalized units. Errors for the other data sets are already in normalized units.

Table 2
Selection results for real data sets

	Chemomet.	Chem. react.	Tobacco	Macroecon.	Sugar(1)	Sugar(2)	Sugar(3)
L_2-MRSR	21.8 (0.4)	6.2 (0.6)	6.0 (0.0)	9.5 (0.7)	40.0 (4.2)	43	46
L_2-SVS	21.9 (0.4)	7.7 (0.7)	6.0 (0.0)	20.0 (0.0)	36.0 (1.4)	35	35
L_2-SVS & OLS	20.0 (0.3)	5.2 (0.4)	3.0 (0.2)	10.9 (0.9)	13.0 (1.0)	13	24
L_∞-SVS	21.9 (0.3)	8.3 (0.7)	5.7 (0.5)	18.3 (1.1)	45.7 (1.8)	46	52
L_∞-SVS & OLS	18.3 (0.6)	5.2 (0.4)	3.0 (0.0)	7.0 (0.4)	15.1 (0.7)	15	25
Full OLS	22.0 (0.0)	9.0 (0.0)	6.0 (0.0)	20.0 (0.0)	—	—	—

Average number of inputs with standard deviation in the minimum CVE model during cross-validation (columns 2–6), number of inputs in the model used for test prediction (columns 7–8).

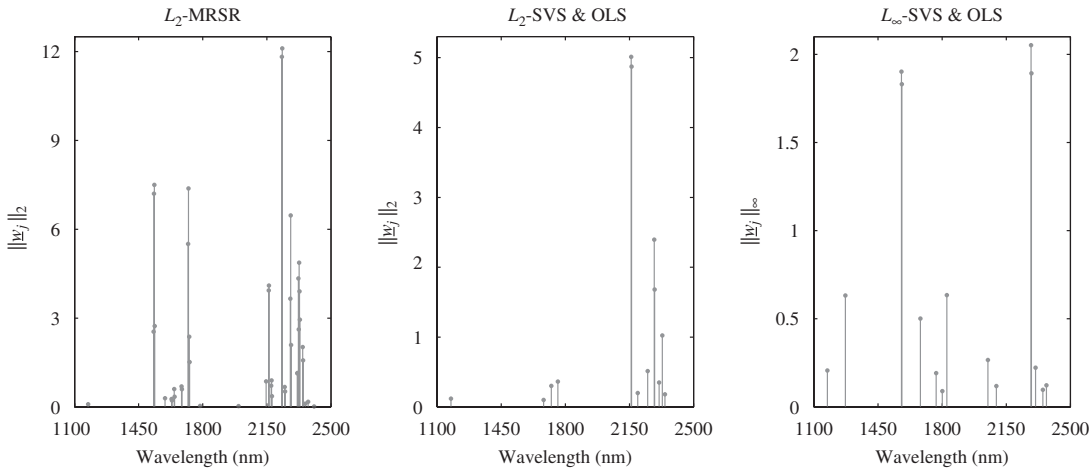

Fig. 2. Importance factors of the selected wavelengths in the minimum CVE models for Sugar data.

least number of inputs. However, Lutz and Bühlmann (2006) report a CVE value of 0.19, which is smaller than any of our errors. The cross-validation results should be interpreted with care, because they base on estimates of the true prediction error and the standard deviations of the estimates are quite large. However, the standard deviations of input selection are small compared to the average values, which means that the methods selected inputs consistently in the cross-validation repetitions.

In the case of Sugar data, cross-validation is used for model selection and prediction accuracy is evaluated on the separate test set. L_2-SVS & OLS is the most accurate with the test error of 0.40 and it uses the least number of inputs. Most of the thirteen selected inputs correspond to the wavelengths, which are in the range 2050–2350 nanometers as shown in Fig. 2. According to Brown et al. (2002), the three sugars actually show spectral differences in this range. The Bayes model average predictor of Brown et al. (2002) has a test error of 0.29 and uses only six inputs, and the highest probability model of the same authors has a test error of 0.34 with twelve inputs. These errors are smaller than the ones of our minimum CVE models. However, L_2-SVS & OLS would have a test error of 0.23, if the model selection was more successful.

It is very fast to compute the L_2-SVS path at all the above-mentioned 500 values of r. The average running times are 2, 1, 1, 2, and 30 s for Chemometrics, Chemical reaction, Tobacco, Macroeconomic, and Sugar data using our implementation. For comparison, solving (10) at the same values of r using the SeDuMi software package takes on average 80, 30, 30, 60 s, and more than 3 h for Chemometrics, Chemical reaction, Tobacco, Macroeconomic, and Sugar data. It should be noted that SeDuMi decreases the duality gap below our limit, but this does not have any effect on the results. Even if the difference in accuracy is taken into account, our implementation is faster.

4.2. Experiments on simulated data

The methods are compared to each other according to prediction accuracy and correctness of input selection using simulated data. The collinearity of input variables has a strong effect on linear models and this experiment illustrates the effect. Data are generated according to (1), where the number of observations, responses, and inputs are always $n = 50$, $q = 5$, and $m = 100$, respectively. Similä and Tikka (2006) sample data in the same way.

The input data are generated according to an m-dimensional normal distribution with zero mean and covariance $\boldsymbol{\Sigma}_x$,

$$\underline{\boldsymbol{x}}_i \sim \mathrm{N}(\boldsymbol{0}_{m\times 1}, \boldsymbol{\Sigma}_x), \quad \text{where } [\boldsymbol{\Sigma}_x]_{ij} = \sigma_x^{|i-j|}.$$

The collinearity among the input variables can be controlled by the parameter σ_x. We consider the values $\sigma_x = 0$, 0.5, and 0.9. The correlation between all the inputs is zero with $\sigma_x = 0$. A few inputs have medium correlation with $\sigma_x = 0.5$ and the average is 0.02. The average correlation is 0.16 with $\sigma_x = 0.9$, but then there are also some strongly correlated inputs. Errors are distributed according to a q-dimensional normal distribution with zero mean and covariance $\boldsymbol{\Sigma}_e$,

$$\underline{\boldsymbol{e}}_i \sim \mathrm{N}(\boldsymbol{0}_{q\times 1}, \boldsymbol{\Sigma}_e), \quad \text{where } [\boldsymbol{\Sigma}_e]_{ij} = 0.2^2 \cdot 0.6^{|i-j|}.$$

The errors have an equal standard deviation 0.2 and there are correlations between the errors of different responses due to the construction.

The actual matrix of regression coefficients $\boldsymbol{W}^*$ is set to have a row-sparse structure by selecting 20 rows out of the total number of 100 rows randomly. The values of the coefficients in the selected rows are independently normally distributed with zero mean and unit variance. The other 80 rows are filled with zeros. The columns $\boldsymbol{w}_i^*$ of the matrix $\boldsymbol{W}^*$ are scaled after sampling as follows

$$\boldsymbol{w}_i^* := \boldsymbol{w}_i^* \Big/ \sqrt{\boldsymbol{w}_i^{*\mathrm{T}} \boldsymbol{\Sigma}_x \boldsymbol{w}_i^*}.$$

This scaling ensures that the responses $\boldsymbol{y}_i$ are on the same scale. They are also correlated due to the construction. Since the responses are comparable, the mean of the individual mean squared errors

$$\mathrm{MSE} = \frac{1}{q}\sum_{i=1}^{q} (\boldsymbol{w}_i^* - \boldsymbol{w}_i)^{\mathrm{T}} \boldsymbol{\Sigma}_x (\boldsymbol{w}_i^* - \boldsymbol{w}_i),$$

can be used as an accuracy measure for different methods (Breiman and Friedman, 1997).

For each value of σ_x, the generation of data is replicated 500 times. The path of L_2-SVS is evaluated at 300 linearly equally spaced points of r in the range [0, 15]. L_∞-SVS is evaluated at 300 linearly equally spaced points of r in the range [0, 11]. The L_p-SVS & OLS estimates can only be calculated at those points, where L_p-SVS has selected at most 50 inputs, since we have only 50 observations. The largest such a value of r varies slightly in the replicated data sets. In the case of L_2-MRSR, the breakpoints are calculated first. After that, we find the estimates between the breakpoints such that L_2-MRSR is evaluated at the same points as L_2-SVS. The maximum number of inputs is also 50 in the L_2-MRSR models.

The two leftmost columns in Fig. 3 show the average MSEs for all the methods. L_2-SVS competes favorably with the other methods. The minimum MSE is roughly the same for L_2-SVS regardless of the value of σ_x. L_2-MRSR has the same prediction accuracy as L_2-SVS when we have $\sigma_x = 0$ and 0.5, but it is worse in the case of $\sigma_x = 0.9$. The most accurate L_∞-SVS and L_∞-SVS & OLS models are always worse than the corresponding 2-norm models. L_2-SVS has the highest prediction accuracy and it is also the least sensitive to the degree of collinearity among the inputs.

The rightmost column in Fig. 3 shows the average number of selected inputs and the average number of correctly selected inputs. The minimum MSE is achieved using approximately 60 inputs in the L_2-SVS models, 80 inputs in the L_∞-SVS models, and 40 inputs in the L_2-MRSR models. This happens with all the values of σ_x. Nearly all the correct inputs are selected, but there are clearly too many false ones. However, shrinking of the regression coefficients prevents from overfitting. L_2-MRSR and L_2-SVS choose inputs equally correctly with $\sigma_x = 0$ and 0.5. In the case of $\sigma_x = 0.9$, L_2-SVS is better. In addition, L_2-SVS is consistently more correct than L_∞-SVS, and of course, this also applies to the subset OLS models.

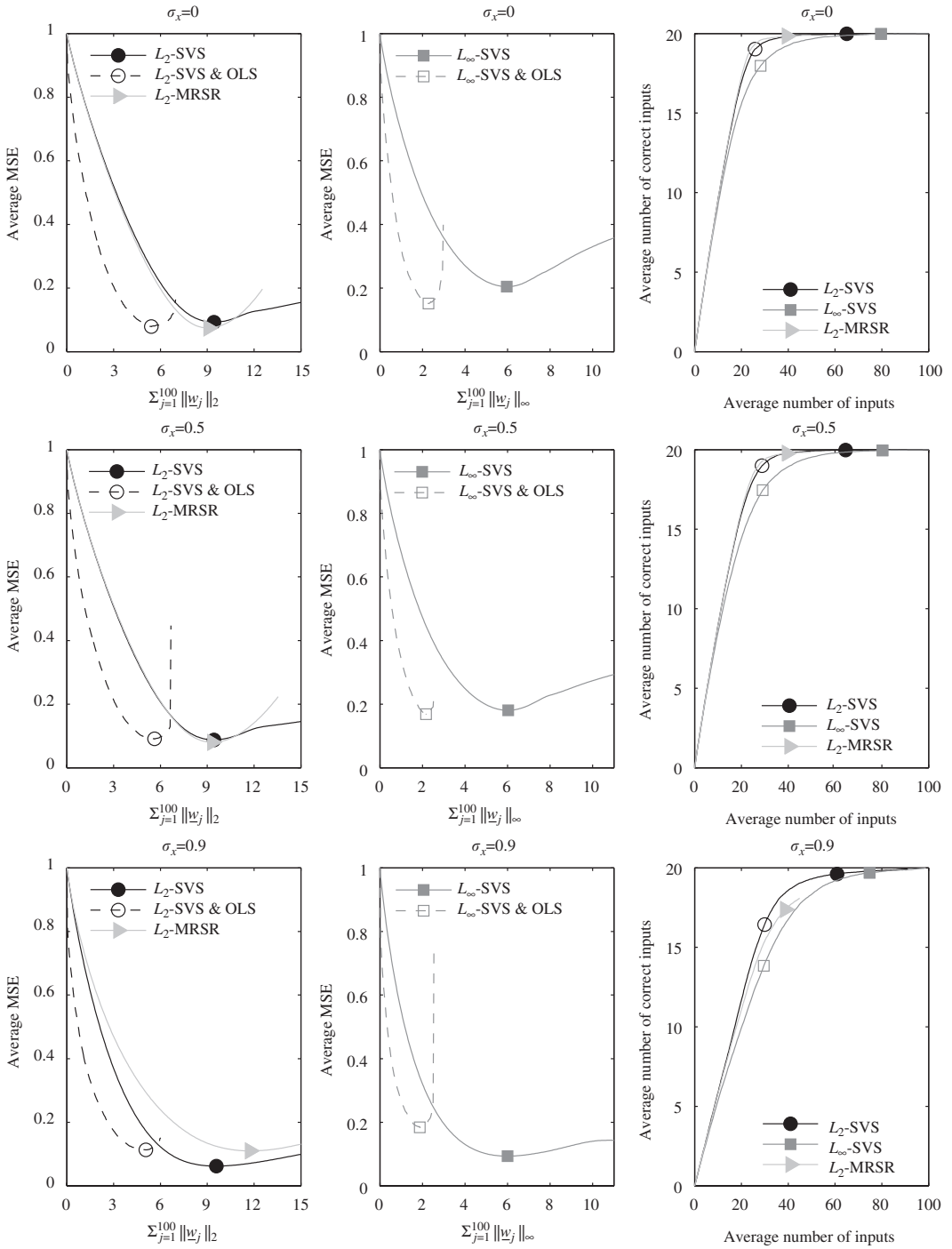

Fig. 3. Averages calculated from 500 replicates of simulated data. Markers indicate the positions of minimum average MSE in each subfigure. Points in the dashed curves are obtained by computing the average MSE of the subset OLS solutions and keeping the horizontal position fixed.

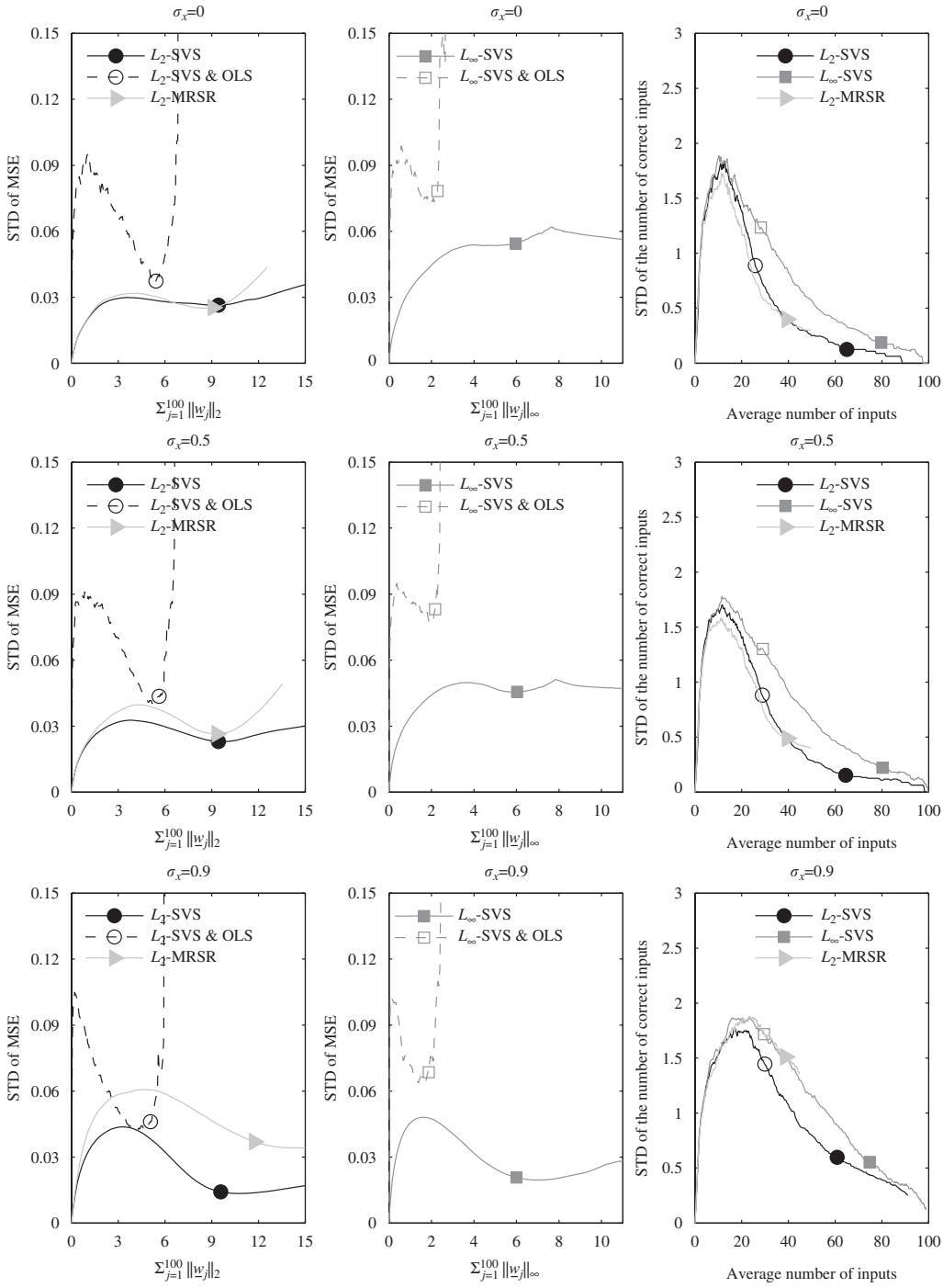

Fig. 4. Standard deviations (STD) calculated from 500 replicates of simulated data. Markers indicate the positions of minimum average MSE in each subfigure. Points in the dashed curves are obtained by computing the STD of the subset OLS solutions and keeping the horizontal position fixed.

The two leftmost columns in Fig. 4 show the standard deviations of MSEs for all the methods. L_2-SVS is the most stable method and its stability is rather independent of the value of σ_x. The deviation of L_2-MRSR is similar to L_2-SVS with $\sigma_x = 0$, but it is otherwise larger. L_2-SVS has consistently a smaller deviation than L_∞-SVS. Interestingly, the deviation of L_∞-SVS decreases as the value of σ_x increases. The L_p-SVS & OLS models have clearly higher deviations than the L_p-SVS models. However, the 2-norm subset OLS models are more stable than the corresponding ∞-norm models.

The rightmost column in Fig. 4 shows the standard deviation of the number of correct inputs. L_2-SVS is also the most stable in terms of input selection. L_2-SVS has a smaller deviation than L_2-MRSR with $\sigma_x = 0.9$, and there is otherwise no significant difference. Observe also that the deviation of correctly selected inputs in the L_2-SVS model is always smaller than the deviation in the corresponding L_∞-SVS model. Again, this also applies to the subset OLS models.

The average running times for computing the L_2-SVS path at all the 300 values of r are the following. It takes 8, 9, and 20 s for $\sigma_x = 0$, 0.5, and 0.9 using our implementation. For comparison, solving (10) at the same values of r using the SeDuMi software package takes on average 790, 810, and 890 s for $\sigma_x = 0$, 0.5, and 0.9. Our implementation is again faster.

5. Conclusions

We have investigated the problem of selecting a subset of input variables, which is useful for modeling several response variables simultaneously. The presented L_2-SVS estimate is given by the solution to a convex optimization problem, where the error sum of squares is minimized while constraining a sum of 2-norms. Each norm concerns the regression coefficients associated with a certain input variable in the model. We have given the optimality conditions and derived another condition, which guarantees the uniqueness of the solution. Furthermore, we have proposed an algorithm for computing the L_2-SVS estimate given a fixed value of the constraint parameter, and even more importantly, a variant of the algorithm for computing the entire path of estimates as a function of this parameter. Our path following algorithm is fast and it enables solving even large problems.

We have applied L_2-SVS to several real data sets and carried out simulation experiments. Results for the real data sets do not show major differences between L_2-SVS and other subset estimates. However, all the studied subset estimates are more accurate and stable than the full OLS estimate. Our simulation experiments indicate that L_2-SVS is very competitive in terms of prediction accuracy and correctness of selection when many input variables are available for model construction. L_2-SVS is consistently more precise and stable than L_∞-SVS and it outperforms L_2-MRSR, especially when the input variables are highly correlated.

Acknowledgments

The authors thank Dr. Petteri Pajunen for helpful discussions. The comments of the editors and referees led to many improvements in the manuscript. This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. Timo Similä acknowledges the support by ComMIT graduate school, Nokia Foundation, and Technological Foundation. Jarkko Tikka acknowledges the support by Emil Aaltonen Foundation. This publication only reflects the authors' views.

Appendix A

A.1. Proofs

Proof of Theorem 1. Suppose that $\boldsymbol{W}^1$ solves (2) and $\boldsymbol{X}_{\mathscr{A}^1}^{\mathrm{T}}\boldsymbol{X}_{\mathscr{A}^1}$ is positive definite. Assume, to the contrary, that there exists $\boldsymbol{W}^0 \neq \boldsymbol{W}^1$, which also solves (2) with the minimum $f(\boldsymbol{W}^0) = f(\boldsymbol{W}^1)$. Since $f(\boldsymbol{W})$ and $g(\boldsymbol{W})$ are convex, the set of solutions is a closed convex subset of feasible points (Hiriart-Urruty and Lemaréchal, 1996). Thus, $\boldsymbol{W}^\alpha = \alpha\boldsymbol{W}^1 + (1-\alpha)\boldsymbol{W}^0$ must also be a solution with the minimum $f(\boldsymbol{W}^\alpha) = f(\boldsymbol{W}^1)$ for all $\alpha \in [0, 1]$. (i) Suppose that $\mathscr{A}^0 = \mathscr{A}^1$ holds, so $\boldsymbol{W}^\alpha$ can have nonzero entries only on the rows indexed by $\mathscr{A}^1$. Define $\widehat{f}(\boldsymbol{W}_{\mathscr{A}^1}) = \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{X}_{\mathscr{A}^1}\boldsymbol{W}_{\mathscr{A}^1}\|_{\mathrm{F}}^2$ and observe that the values $\widehat{f}(\boldsymbol{W}^\alpha_{\mathscr{A}^1}) = f(\boldsymbol{W}^\alpha) = f(\boldsymbol{W}^1)$ are constant for all $\alpha \in [0, 1]$. This contradicts the fact that $\widehat{f}(\boldsymbol{W}_{\mathscr{A}^1})$ is

strictly convex due to positive definiteness of $\boldsymbol{X}_{\mathscr{A}^1}^{\mathrm{T}}\boldsymbol{X}_{\mathscr{A}^1}$. (ii) Suppose, in turn, that $\mathscr{A}^0 \neq \mathscr{A}^1$ holds. Then we have $f(\boldsymbol{W}^\alpha) = \frac{1}{2}\|\mathrm{vec}(\boldsymbol{Y}) - \mathrm{vec}(\boldsymbol{X}\boldsymbol{W}^\alpha)\|_2^2 = f(\boldsymbol{W}^1)$ for all $\alpha \in [0, 1]$ and thereby an equal distance between the point $\mathrm{vec}(\boldsymbol{Y})$ and all points in the line segment $\{\mathrm{vec}(\boldsymbol{X}\boldsymbol{W}^\alpha) : \alpha \in [0, 1]\}$ in $\mathbb{R}^{qn\times 1}$. This is possible only if the line segment reduces to a single point and we have $\boldsymbol{X}\boldsymbol{W}^0 = \boldsymbol{X}\boldsymbol{W}^1$, which implies $(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W}^0)^{\mathrm{T}}\boldsymbol{x}_j = (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W}^1)^{\mathrm{T}}\boldsymbol{x}_j$ for all $j = 1, \ldots, m$. But then, we must have $\mathscr{A}^0 = \mathscr{A}^1$ according to (9), which contradicts the assumption $\mathscr{A}^0 \neq \mathscr{A}^1$. Steps (i) and (ii) together prove Theorem 1. □

A.2. Newton step

The gradient vector is

$$\nabla F_\mu(\boldsymbol{W}, \boldsymbol{s}) = \mathrm{vec}([\boldsymbol{G}_{\boldsymbol{w}}|\boldsymbol{g}_{\boldsymbol{s}}]),$$

where

$$\boldsymbol{G}_{\boldsymbol{w}} = -\boldsymbol{X}^{\mathrm{T}}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W}) + \mathrm{diag}(\boldsymbol{\eta})\boldsymbol{W}, \quad \boldsymbol{g}_{\boldsymbol{s}} = \lambda'\mathbf{1}_{m\times 1} - \mathrm{diag}(\boldsymbol{\eta})\boldsymbol{s},$$
$$\lambda' = \mu \Big/ \left(r - \sum_{j=1}^{m} s_j\right), \quad \eta_j = 2\mu/(m(s_j^2 - \|\underline{\boldsymbol{w}}_j\|_2^2)).$$

The Hessian matrix has the form

$$\nabla^2 F_\mu(\boldsymbol{W}, \boldsymbol{s}) = \boldsymbol{D} + \boldsymbol{M}\,\mathrm{diag}(\boldsymbol{v})\boldsymbol{M}^{\mathrm{T}},$$

where

$$\boldsymbol{D} = \mathrm{diag}(\boldsymbol{H}_{\boldsymbol{w}}, \ldots, \boldsymbol{H}_{\boldsymbol{w}}, \boldsymbol{H}_{\boldsymbol{s}}), \quad v_j = 4\mu/(m(s_j^2 - \|\underline{\boldsymbol{w}}_j\|_2^2)^2),$$
$$\boldsymbol{H}_{\boldsymbol{w}} = \boldsymbol{X}^{\mathrm{T}}\boldsymbol{X} + \mathrm{diag}(\boldsymbol{\eta}), \quad \boldsymbol{H}_{\boldsymbol{s}} = \phi\mathbf{1}_{m\times m} - \mathrm{diag}(\boldsymbol{\eta}), \quad \phi = \mu \Big/ \left(r - \sum_{j=1}^{m} s_j\right)^2,$$
$$\boldsymbol{M}^{\mathrm{T}} = [\mathrm{diag}(\boldsymbol{w}_1)|\cdots|\mathrm{diag}(\boldsymbol{w}_q)|\mathrm{diag}(-\boldsymbol{s})].$$

According to the Matrix Inversion Lemma (Boyd and Vandenberghe, 2004), we may express the inverse of the Hessian as follows:

$$\nabla^2 F_\mu(\boldsymbol{W}, \boldsymbol{s})^{-1} = \boldsymbol{D}^{-1} - \boldsymbol{D}^{-1}\boldsymbol{M}\boldsymbol{P}^{-1}\boldsymbol{M}^{\mathrm{T}}\boldsymbol{D}^{-1},$$

where $\boldsymbol{P} = \mathrm{diag}(\widehat{\boldsymbol{v}}) + \boldsymbol{M}^{\mathrm{T}}\boldsymbol{D}^{-1}\boldsymbol{M}$ and $\widehat{v}_j = 1/v_j$. Since $\boldsymbol{D}$ is block diagonal, it has an inverse matrix of the form $\boldsymbol{D}^{-1} = \mathrm{diag}(\boldsymbol{H}_{\boldsymbol{w}}^{-1}, \ldots, \boldsymbol{H}_{\boldsymbol{w}}^{-1}, \boldsymbol{H}_{\boldsymbol{s}}^{-1})$. The inverse of the block $\boldsymbol{H}_{\boldsymbol{s}}$ is simply

$$\boldsymbol{H}_{\boldsymbol{s}}^{-1} = \left(\sum_{j=1}^{m}\widehat{\eta}_j - 1/\phi\right)^{-1}\widehat{\boldsymbol{\eta}}\widehat{\boldsymbol{\eta}}^{\mathrm{T}} - \mathrm{diag}(\widehat{\boldsymbol{\eta}}),$$

where hats denote componentwise inverses, that is $\widehat{\eta}_j = 1/\eta_j$.

Now it is straightforward to compute the Newton step $-\nabla^2 F_\mu^{-1}\nabla F_\mu$. The computationally most demanding task is solving the systems of linear equations associated with the $m \times m$ matrices $\boldsymbol{P}$ and $\boldsymbol{H}_{\boldsymbol{w}}$. The matrix $\boldsymbol{P}$ is tricky as the values between the rows may differ considerably. This complicates the computation of the vector $\boldsymbol{a} = \boldsymbol{P}^{-1}\boldsymbol{b}$, where $\boldsymbol{b} = \boldsymbol{M}^{\mathrm{T}}\boldsymbol{D}^{-1}\nabla F_\mu$. Numerical stability can be increased by solving the system $\widetilde{\boldsymbol{P}}\widetilde{\boldsymbol{a}} = \widetilde{\boldsymbol{b}}$, where

$$\widetilde{\boldsymbol{P}} = \mathrm{diag}(\widetilde{\boldsymbol{p}})\boldsymbol{P}\,\mathrm{diag}(\widetilde{\boldsymbol{p}}), \quad \widetilde{\boldsymbol{b}} = \mathrm{diag}(\widetilde{\boldsymbol{p}})\boldsymbol{b}, \quad \widetilde{p}_j = 1 \Big/ \sqrt{|p_{jj}|}.$$

Finally, we have $\boldsymbol{a} = \mathrm{diag}(\widetilde{\boldsymbol{p}})\widetilde{\boldsymbol{a}}$.

A.3. Length of the linear step

It can be shown that the curves $\lambda(r)$ and $\|(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W}(r))^{\mathrm{T}}\boldsymbol{x}_j\|_2$ intersect at a unique point r_j on the interval $(0, \lambda^{[0]}/\|\boldsymbol{x}_k\|_2^2]$ for all $j \neq k$ (Similä and Tikka, 2006). The value $r^{[1]}$ corresponds to the smallest such a point. According to (14), the following equivalence holds:

$$\|(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W}(r))^{\mathrm{T}}\boldsymbol{x}_j\|_2^2 = \lambda(r)^2 \quad \Leftrightarrow \quad a_j r^2 - 2b_j r + c_j = 0,$$

where

$$a_j = (\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_k)^2 - (\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_j)^2, \quad b_j = \lambda^{[0]}\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_k - (\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_j/\lambda^{[0]})\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{Y}\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{x}_j, \; c_j = (\lambda^{[0]})^2 - \|\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{x}_j\|_2^2.$$

Now r_j denotes the smallest positive root of the above polynomial and so we have

$$r^{[1]} = \min_{j \neq k} r_j, \quad \text{where } r_j = \min{}^{+}\left(b_j \pm \sqrt{b_j^2 - a_j c_j}\right)/a_j.$$

The expression $\min^{+}$ indicates that the minimum is taken only over the positive terms.

References

Abraham, B., Merola, G., 2005. Dimensionality reduction approach to multivariate prediction. Comput. Statist. Data Anal. 48 (1), 5–16.

Alizadeh, A., Goldfarb, D., 2003. Second-order cone programming. Math. Programming Ser. B 95 (1), 3–51.

Allgower, E.L., Georg, K., 1993. Continuation and path following. Acta Numer. 2, 1–64.

Anderson, R.L., Bancroft, T.A., 1952. Statistical Theory in Research. McGraw-Hill Book Company, New York.

Bach, F.R., Thibaux, R., Jordan, M.I., 2005. Computing regularization paths for learning multiple kernels. In: Saul, L.K., Weiss, Y., Bottou, L. (Eds.), Advances in Neural Information Processing Systems, vol. 17. MIT Press, Cambridge, MA, pp. 73–80.

Bakin, S., 1999. Adaptive regression and model selection in data mining problems. Ph.D. Thesis, Australian National University.

Barrett, B.E., Gray, J.B., 1994. A computational framework for variable selection in multivariate regression. Statist. Comput. 4 (3), 203–212.

Bedrick, E.J., Tsai, C.-L., 1994. Model selection for multivariate regression in small samples. Biometrics 50 (1), 226–231.

Boyd, S., Vandenberghe, L., 2004. Convex Optimization. Cambridge University Press, Cambridge, MA.

Breiman, L., 1996. Heuristics of instability and stabilization in model selection. Ann. Statist. 24 (6), 2350–2383.

Breiman, L., Friedman, J.H., 1997. Predicting multivariate responses in multiple linear regression. J. Roy. Statist. Soc. Ser. B 59 (1), 3–54.

Brown, P.J., Zidek, J.V., 1980. Adaptive multivariate ridge regression. Ann. Statist. 8 (1), 64–74.

Brown, P.J., Vannucci, M., Fearn, T., 2002. Bayes model averaging with selection of regressors. J. Roy. Statist. Soc. Ser. B 64 (3), 519–536.

Burnham, A.J., MacGregor, J.F., Viveros, R., 1999. Latent variable multivariate regression modeling. Chemometrics Intell. Lab. Systems 48 (2), 167–180.

Cotter, S.F., Rao, B.D., Engan, K., Kreutz-Delgado, K., 2005. Sparse solutions to linear inverse problems with multiple measurement vectors. IEEE Trans. Signal Process. 53 (7), 2477–2488.

Derksen, S., Keselman, H.J., 1992. Backward, forward and stepwise automated subset selection algorithms: frequency of obtaining authentic and noise variables. British J. Math. Statist. Psych. 45, 265–282.

Efron, B., Hastie, T., Johnstone, I., Tibshirani, R., 2004. Least angle regression. Ann. Statist. 32 (2), 407–499.

Hiriart-Urruty, J.-B., Lemaréchal, C., 1996. Convex Analysis and Minimization Algorithms I. Springer, Berlin.

Kim, Y., Kim, J., Kim, Y., 2006. Blockwise sparse regression. Statist. Sinica 16 (2), 375–390.

Lin, Y., Zhang, H.H., 2006. Component selection and smoothing in multivariate nonparametric regression. Ann. Statist. 34 (5), 2272–2297.

Lutz, R.W., Bühlmann, P., 2006. Boosting for high-multivariate responses in high-dimensional linear regression. Statist. Sinica 16 (2), 471–494.

Malioutov, D., Çetin, M., Willsky, A.S., 2005. A sparse signal reconstruction perspective for source localization with sensor arrays. IEEE Trans. Signal Process. 53 (8), 3010–3022.

Meier, L., van de Geer, S., Bühlmann, P., 2006. The group lasso for logistic regression. Technical Report, Eidgenössische Technische Hochschule.

Osborne, M.R., Presnell, B., Turlach, B.A., 2000. A new approach to variable selection in least squares problems. IMA J. Numer. Anal. 20 (3), 389–404.

Park, M.Y., Hastie, T., 2006. Regularization path algorithms for detecting gene interactions. Technical Report, Stanford University.

Rakowska, J., Haftka, R.T., Watson, L.T., 1991. An active set algorithm for tracing parametrized optima. Structural Optim. 3 (1), 29–44.

Reinsel, G.C., Velu, R.P., 1998. Multivariate Reduced-Rank Regression, Theory and Applications. Springer, New York.

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

Rosset, S., 2005. Following curved regularized optimization solution paths. In: Saul, L.K., Weiss, Y., Bottou, L. (Eds.), Advances in Neural Information Processing Systems, vol. 17. MIT Press, Cambridge, MA, pp. 1153–1160.

Similä, T., Tikka, J., 2006. Common subset selection of inputs in multiresponse regression. In: IEEE International Joint Conference on Neural Networks, Vancouver, Canada, pp. 1908–1915.

Sparks, R.S., Zucchini, W., Coutsourides, D., 1985. On variable selection in multivariate regression. Comm. Statist. Theory Methods 14 (7), 1569–1587.

Srivastava, M.S., 2002. Methods of Multivariate Statistics. Wiley, New York.
Srivastava, M.S., Solanky, T.K.S., 2003. Predicting multivariate response in linear regression model. Comm. Statist. Simulation Comput. 32 (2), 389–409.
Sturm, J.F., 1999. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optim. Methods Softw. 11–12, 625–653.
Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. J. Roy. Statist. Soc. Ser. B 58 (1), 267–288.
Tropp, J.A., 2006. Algorithms for simultaneous sparse approximation. Part II: convex relaxation. Signal Process. 86 (3), 589–602.
Turlach, B.A., 2005. On homotopy algorithms in statistics. In: Symposium on Optimisation and Data Analysis. Canberra, Australia, Keynote Talk, ⟨http://www.maths.uwa.edu.au/∼berwin/⟩.
Turlach, B.A., 2006. Simultaneous variable selection. In: Joint Statistical Meetings, Seattle, USA, Invited Talk, ⟨http://www.maths.uwa.edu.au/∼berwin/⟩.
Turlach, B.A., Venables, W.N., Wright, S.J., 2005. Simultaneous variable selection. Technometrics 47 (3), 349–363.
Yuan, M., Lin, Y., 2006. Model selection and estimation in regression with grouped variables. J. Roy. Statist. Soc. Ser. B 68 (1), 49–67.
Zhao, P., Rocha, G., Yu, B., 2006. Grouped and hierarchical model selection through composite absolute penalties. Technical Report, University of California.
Zou, H., Yuan, M., 2005. The F_∞-norm support vector machine. Technical Report, Georgia Institute of Technology.