Title

Joint covariate selection and joint subspace selection for multiple classification problems

Permalink
https://escholarship.org/uc/item/3td3r1p6

Journal
Statistics and Computing, 20(2)

ISSN
1573-1375

Authors
Obozinski, Guillaume
Taskar, Ben
Jordan, Michael I.

Publication Date
2010-04-01

DOI
10.1007/s11222-008-9111-x

Peer reviewed

Joint covariate selection and joint subspace selection for multiple classification problems

Guillaume Obozinski · Ben Taskar · Michael I. Jordan

Received: 9 October 2007 / Accepted: 1 December 2008 / Published online: 14 January 2009

Abstract We address the problem of recovering a common set of covariates that are relevant simultaneously to several classification problems. By penalizing the sum of ℓ_2 norms of the blocks of coefficients associated with each covariate across different classification problems, similar sparsity patterns in all models are encouraged. To take computational advantage of the sparsity of solutions at high regularization levels, we propose a blockwise path-following scheme that approximately traces the regularization path. As the regularization coefficient decreases, the algorithm maintains and updates concurrently a growing set of covariates that are simultaneously active for all problems. We also show how to use random projections to extend this approach to the problem of *joint subspace selection*, where multiple predictors are found in a common low-dimensional subspace. We present theoretical results showing that this random projection approach converges to the solution yielded by trace-norm regularization. Finally, we present a variety of experimental results exploring joint covariate selection and joint subspace selection, comparing the path-following approach to competing algorithms in terms of prediction accuracy and running time.

Keywords Variable selection · Subspace selection · Lasso · Group Lasso · Regularization path · Supervised dimensionality reduction · Multitask learning · Block norm · Trace norm · Random projections

1 Introduction

The problem of covariate selection for regression and classification has been the focus of a substantial literature. As with many model selection problems, the problem is rendered difficult by the disparity between the large number of models to be considered and the comparatively small amount of data available to evaluate these models. One approach to the problem focuses on procedures that search within the exponentially-large set of all subsets of components of the covariate vector, using various heuristics such as *forward* or *backward selection* to limit the search (Draper and Smith 1998). Another approach treats the problem as a parameter estimation problem in which the shrinkage induced by a constraint on the ℓ_1 norm of the parameter vector yields estimates in which certain components are equal to zero (Tibshirani 1996; Fu and Knight 2000; Donoho 2004). A virtue of the former approach is that it focuses on the qualitative decision as to whether a covariate is relevant to the problem at hand, a decision which is conceptually distinct from parameter estimation. A virtue of the latter approach is its computational tractability.

In this paper, we focus on a problem setting in which these virtues appear to be better aligned than they are in general regression and classification problems. In particular, we focus on situations involving multiple, related data sets in which the same set of covariates are present in each data set

G. Obozinski (✉)
Department of Statistics, University of California at Berkeley, 367 Evans Hall, Berkeley, CA 94720-3860, USA
e-mail: gobo@stat.berkeley.edu

B. Taskar
Department of Computer and Information Science, University of Pennsylvania, 3330 Walnut Street, Philadelphia, PA 19104-6389, USA
e-mail: taskar@cis.upenn.edu

M.I. Jordan
Department of Statistics and Department of Electrical Engineering and Computer Science, University of California at Berkeley, 367 Evans Hall, Berkeley, CA 94720-3860, USA
e-mail: jordan@stat.berkeley.edu

but where the responses differ. In this multi-response setting it is natural to associate a notion of "relevance" to a covariate that is conceptually distinct from the numerical value of a parameter. For example, a particular covariate may appear with a positive coefficient in predicting one response variable and with a negative coefficient in predicting a different response. We would clearly want to judge such a covariate as being "relevant" to the overall class of prediction problems without making a commitment to a specific value of a parameter. In general we wish to "borrow strength" across multiple estimation problems in order to support a decision that a covariate is to be selected.

Our focus in this paper is the classification or discrimination problem. Consider, for example, the following pattern recognition problem that we consider later in Sect. 6. We assume that we are given a data set consisting of pixel-level or stroke-level representations of handwritten characters and we wish to classify a given character into one of a fixed set of classes. In this *optical character recognition* (OCR) problem, there are several thousand covariates, most of which are irrelevant to the classification decision of character identity. To support the choice of relevant covariates in this high-dimensional problem, we consider an extended version of the problem in which we assume that multiple data sets are available, one for each individual in a set of writers. We expect that even though the styles of individual writers may vary, there should be a common subset of image features (pixels, strokes) that form a shared set of useful covariates across writers.

As another example of our general setting, also discussed in Sect. 6, consider a DNA microarray analysis problem in which the covariates are levels of gene expression and the responses are phenotypes or cellular processes (Khan et al. 2001). Given the high-dimensional nature of microarray data sets, covariate selection is often essential both for scientific understanding and for effective prediction. Our proposal is to approach the covariate selection problem by considering multiple related phenotypes—e.g., related sets of cancers—and seeking to find covariates that are useful in predicting these multiple response variables.

Our approach to the simultaneous covariate selection problem is an adaptation of ℓ_1 shrinkage methods such as LASSO. Briefly, for each data set $\{(x_i^k, y_i^k) : i = 1, \dots, N_k\}$, where $k \in \{1, \dots, K\}$ indexes data sets, we fit a model involving a parameter vector $w^k \in \mathbb{R}^p$. View these vectors as rows of a $K \times p$ matrix W, and consider the jth column vector, w_j, of W. This vector consists of the set of parameters associated to the jth covariate across all classification problems. We now define a regularization term that is an ℓ_1 sum of the ℓ_2 norms of the covariate-specific parameter vectors w_j. Each of these ℓ_2 norms can be viewed as assessing the overall relevance of a particular covariate. The ℓ_1 sum then enforces a selection among covariates based on these norms.

This approach is a particular case of a general methodology in which block norms are used to define groupings of variables in regression and classification problems (Bach et al. 2004; Yuan and Lin 2006; Park and Hastie 2006; Meier et al. 2008; Kim et al. 2006; Zhao et al. 2008). However, the focus in this literature differs from ours in that it is concerned with grouping variables within a single regression or classification problem. For example, in a polynomial regression we may wish to group the linear, quadratic and cubic terms corresponding to a specific covariate and select these terms jointly. Similarly, in an ANOVA model we may wish to group the indicator variables corresponding to a specific factor. The block-norm approach to these problems is based on defining block norms involving hybrids of ℓ_1, ℓ_2 and ℓ_∞ norms as regularization terms.

Argyriou et al. (2008) have independently proposed the use of a block ℓ_1/ℓ_2 norm for covariate selection in the multiple-response setting. Moreover, they consider a more general framework in which the variables that are selected are linear combinations of the original covariates. We refer to this problem as *joint subspace selection*. Joint covariate selection is a special case in which the subspaces are restricted to be axis-parallel. Argyriou et al. show that the general subspace selection problem can be formulated as an optimization problem involving the trace norm.

Our contribution relative to Argyriou et al. is as follows. First, we note that the trace norm is difficult to optimize computationally (it yields a non-differentiable functional that is generally evaluated by the computation of a singular value decomposition at each step of a nonlinear optimization procedure Srebro et al. 2005b), and we thus focus on the special case of covariate selection, where it is not necessary to use the trace norm. For the case of covariate selection we show that it is possible to develop a simple homotopy-based approach that evaluates an entire regularization path efficiently (cf. Efron et al. 2004; Osborne et al. 2000). We present a theoretical result establishing the convergence of this homotopy-based method. Moreover, for the general case of joint subspace selection we show how random projections can be used to reduce the problem to covariate selection. Applying our homotopy method for joint covariate selection to the random projections, we obtain a computationally-efficient procedure for joint subspace selection. We also present a theoretical result showing that this approach approximates the solution obtained from the trace norm. Finally, we present several experiments on large-scale datasets that compare and contrast various methods for joint covariate selection and joint subspace selection.

The general problem of jointly estimating models from multiple, related data sets is often referred to as "transfer learning" or "multi-task learning" in the machine learning literature (Maurer 2006; Ben-David and Schuller-Borbely 2008; Argyriou et al. 2008; Jebara 2004; Evgeniou and Pontil 2004; Torralba et al. 2004; Ando and Zhang 2005). We adopt the following terminology from this literature: a *task* is defined to be a pairing of a set of covariate vectors and a specific component of a multiple response vector. We wish to find covariates and subspaces that are useful across multiple tasks.

The paper is organized as follows. In Sect. 2, we introduce the ℓ_1/ℓ_2 regularization scheme and the corresponding optimization problem. In Sect. 3 we discuss homotopy-based methods, and in Sect. 4 we propose a general scheme for following a piecewise smooth, nonlinear regularization path. We extend our algorithm to subspace selection in Sect. 5 and prove convergence to trace-norm regularization. In Sect. 6 we present an empirical evaluation of our joint feature selection algorithm, comparing to several competing block-norm optimizers. We also present an empirical evaluation and comparison of our extension to subspace selection. We conclude with a discussion in Sect. 7.

2 Joint regularization

We assume a group of K classification problems or "tasks" and a set of data samples $\{(x_i^k, y_i^k) \in \mathcal{X} \times \mathcal{Y},\ i = 1, \dots, N_k,\ k = 1, \dots, K\}$ where the superscript k indexes tasks and the subscript i indexes the i.i.d. observations for each task. We assume that the common covariate space $\mathcal{X}$ is $\mathbb{R}^p$ and the outcome space $\mathcal{Y}$ is $\{0, 1\}$.

Let $w^k \in \mathbb{R}^p$ parameterize a linear discriminant function for task k, and let $J^k(w^k \cdot x^k, y^k)$ be a loss function on example (x^k, y^k) for task k. Typical smooth loss functions for linear classification models include logistic and exponential loss. A standard approach to obtaining sparse estimates of the parameters w^k is to solve an ℓ_1-regularized empirical risk minimization problem:

$$\min_{w^k} \sum_{i=1}^{N_k} J^k(w^k \cdot x_i^k, y_i^k) + \lambda \|w^k\|_1,$$

where λ is a regularization coefficient. Solving an independent ℓ_1-regularized objective for each of these problems is equivalent to solving the global problem obtained by summing the objectives:

$$\min_{W} \sum_{k=1}^{K} \sum_{i=1}^{N_k} J^k(w^k \cdot x_i^k, y_i^k) + \lambda \sum_{k=1}^{K} \|w^k\|_1, \tag{1}$$

where $W = (w_j^k)_{k,j}$ is the matrix whose rows are the vectors w^k and whose columns are the vectors w_j of the coefficients associated with covariate j across classification tasks. Note that we have assumed that the regularization coefficient λ is the same across tasks. We refer to the regularization scheme in (1) as an ℓ_1/ℓ_1-regularization. Solving this optimization problem would lead to individual sparsity patterns for each w^k.

We focus instead on a regularization scheme that selects covariates jointly across tasks. We achieve this by encouraging several w_j to be zero. We thus propose to solve the problem

$$\min_{W} \sum_{k=1}^{K} \sum_{i=1}^{N_k} J^k(w^k \cdot x_i^k, y_i^k) + \lambda \sum_{j=1}^{p} \|w_j\|_2, \tag{2}$$

in which we penalize the ℓ_1 norm of the vector of ℓ_2 norms of the covariate-specific coefficient vectors. Note that this ℓ_1/ℓ_2-regularization scheme reduces to ℓ_1-regularization if the group is reduced to one task, and can thus be seen an extension of ℓ_1-regularization where instead of summing the absolute values of coefficients associated with covariates we sum the Euclidean norms of coefficient blocks.

The ℓ_2 norm is used here as a measure of magnitude and one could also generalize to ℓ_1/ℓ_p norms by considering ℓ_p norms for $1 \le p \le \infty$. The choice of p should depend on how much covariate sharing we wish to impose among classification problems, from none ($p = 1$) to full sharing ($p = \infty$). Indeed, increasing p corresponds to allowing better "group discounts" for sharing the same covariate, from $p = 1$, where the cost grows linearly with the number of classification problems that use a covariate, to $p = \infty$, where only the most demanding classification matters.

The shape of the unit "ball" of the ℓ_1/ℓ_2 norm is difficult to visualize. It clearly has corners that, in a manner analogous to the ℓ_1 norm, tend to produce sparse solutions. As shown in Fig. 1, one way to appreciate the effect of the ℓ_1/ℓ_2 norm is to consider a problem with two covariates and

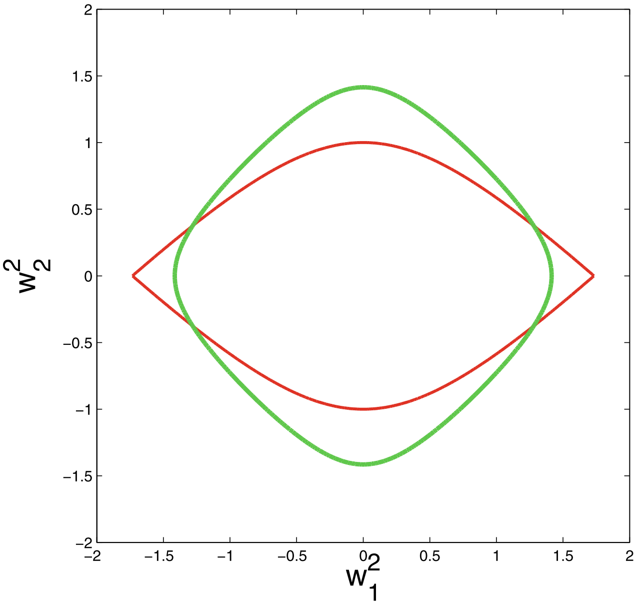


Fig. 1 (Color online) (*Left*) Norm ball induced on the coefficients (w_1^2, w_2^2) for task 2 as covariate coefficients for task 1 vary: *thin red contour* for $(w_1^1, w_2^1) = (0, 1)$ and *thick green contour* for $(w_1^1, w_2^1) = (0.5, 0.5)$

two tasks and to observe the ball of the norm induced on w^2 when w^1 varies under the constraint that $\|w^1\|_1 = 1$ in an ℓ_1/ℓ_2 ball of size 2 (which is the largest value of the ℓ_1/ℓ_2 norm if $\|w^1\|_1 = \|w^2\|_1 = 1$). If a covariate j has a non-zero coefficient in w^1 then the induced norm on w^2 is smooth around $w_j^2 = 0$. Otherwise, it has sharp corners, which encourages w_j^2 to be set to zero.

3 A path-following algorithm for joint covariate selection

In this section we present an algorithm for solving the ℓ_1/ℓ_2-regularized optimization problem presented in (2). One approach to solving such regularization problems is to repeatedly solve them on a grid of values of the regularization coefficient λ, if possible using "warm starts" to initialize the procedure for a given value of λ using the solution for a nearby value of λ. An alternative framework which can be more efficient computationally and can provide insight into the space of solutions is to attempt to follow the "regularization path" (the set of solutions for all values of λ). There are problems—including ℓ_1-regularized least-squares regression and the ℓ_1- and ℓ_2-regularized support vector machines—for which this path is piecewise linear and for which it is possible to follow the path exactly (Efron et al. 2004; Rosset and Zhu 2007). More generally, we can avail ourselves of path-following algorithms. Classical path-following algorithms involve traditional path-following a combination of *prediction steps* (along the tangent to the path) and *correction steps* (which correct for errors due to the first-order approximation of the prediction steps). These algorithms generally require the computation of the Hessian of the combined objective and thus are onerous computationally. However, in the case of ℓ_1 regularization it has been shown that the solution path can be approximated by computationally efficient variations of boosting and stagewise forward selection (Hastie et al. 2001; Zhao and Yu 2007).

Note that the amount of sparsity is controlled by the regularization coefficient λ. As λ ranges from 0 to ∞, the sparsity of solutions typically progresses through several levels (although this is not guaranteed in general). The approach that we present here exploits the high degree of sparsity for large values of λ.

Our approach is inspired by the stagewise Lasso algorithm of Zhao and Yu (2007). In their algorithm, the optimization is performed on a grid with step size ϵ and essentially reduces to a discrete problem that can be viewed as a simplex problem, where "forward" and "backward" steps are alternated. Our approach extends this methodology to the setting of blockwise norms by essentially combining stagewise Lasso with a classical correction step. We take advantage of sparsity so that this step can be implemented cheaply.

4 Active set and parameter updates

We begin our description of the path-following algorithm with a simple lemma that uses a subgradient calculation (equivalently, the Karush-Kuhn-Tucker (KKT) conditions) to show how the sparsity of the solution can lead to an efficient construction of the path. Let us denote the joint loss by $J(W) = \sum_{k=1}^K \sum_{i=1}^{N_k} J^k(w^k \cdot x_i^k, y_i^k)$.

Lemma 1 *If J is everywhere differentiable, then any solution W^* of the optimization problem in (2) is characterized by the following conditions*

$$\begin{aligned} &\textit{either} \quad w_j^* = 0, \qquad \|\nabla_{w_j} J(W^*)\|_2 \le \lambda \\ &\textit{or} \quad w_j^* \propto -\nabla_{w_j} J(W^*), \qquad \|\nabla_{w_j} J(W^*)\|_2 = \lambda, \end{aligned}$$

where $\nabla_{w_j} J(W)$ are partial gradients in each of the subspaces corresponding to covariate-specific parameter vectors.

Proof At an optimum, a subgradient of the objective function equals zero. This implies—given that the ℓ_1/ℓ_2-regularization term is separable for the column vectors w_j of W—that for all j, $\nabla_{w_j} J(W^*) + \lambda z_j^* = 0$ for $z_j^* \in \partial_{w_j} \|w_j\|_2$, where the latter denotes the subgradient of the Euclidean norm. Moreover, the subgradient of the Euclidean norm satisfies

$$\begin{cases} \partial_{w_j} \|w_j\|_2 = \frac{w_j}{\|w_j\|} & \text{if } w_j \neq 0, \\ \partial_{w_j} \|w_j\|_2 = \{z \in \mathbb{R}^K \mid \|z\|_2 \le 1\} & \text{otherwise,} \end{cases} \tag{3}$$

which proves the lemma. The subgradient equations can also be obtained by conic duality, in which case they result directly from the KKT conditions. □

In particular, only the "active" covariates—those for which the norm of the gradient vector is not strictly less than λ—participate in the solution. For these active covariates, $\frac{\lambda}{\|w_j^*\|} w_j^* = -\nabla_{w_j} J(W^*)$. (Note that if $\lambda \ge \lambda_0 = \max_j \|\nabla_{w_j} J(0)\|_2$ then the zero vector is a solution to our problem.)

These conditions suggest an algorithm which gradually decreases the regularization coefficient from λ_0 and populates an active set with inactive covariates as they start to violate subgradient conditions. In particular, we consider *approximate* subgradient conditions of the form:

$$\begin{aligned} &\text{either} \quad w_j = 0, \qquad \|\nabla_{w_j} J(W)\| < \lambda + \xi_0 \\ &\text{or} \quad \left\| \nabla_{w_j} J(W) + (\lambda - \xi) \frac{w_j}{\|w_j\|} \right\| \le \xi, \end{aligned} \tag{4}$$

where ξ and ξ_0 are slack parameters. These conditions are obtained by relaxing the constraints that there must exist a

Algorithm 1 Approximate block-Lasso path

```
Given $\epsilon$ and $\xi$,
while $\lambda^t > \lambda_{\min}$ do
    Set $j^* = \operatorname{argmax}_j \|\nabla_{w_j} J(W^t)\|$
    Update $w_{j^*}^{(t+1)} = w_{j^*}^{(t)} - \epsilon u^t$ with $u^t = \frac{\nabla_{w_{j^*}} J}{\|\nabla_{w_{j^*}} J\|}$
    $\lambda^{t+1} = \min\left(\lambda^t, \frac{J(W^t)-J(W^{t+1})}{\epsilon}\right)$
    Add $j^*$ to the active set
    Enforce (4) for covariates in the active set with $\xi_0 = \xi$.
end while
```

subgradient equal to zero, and asking instead that

$$\begin{cases} \text{For } j \text{ s.t. } w_j = 0, \\ \|\nabla_{w_j} J(W) + \lambda z_j\| \le \xi_0 \quad \text{for some } z_j \in \partial_{w_j} \|w_j\|_2, \\ \text{For } j \text{ s.t. } w_j \neq 0, \\ \|\nabla_{w_j} J(W) + (\lambda - \xi) z_j\| \le \xi \quad \text{for some } z_j \in \partial_{w_j} \|w_j\|_2. \end{cases}$$

The latter constraint ensures that, for any active covariate j, we have $\|\nabla_{w_j} J(W)\| \le \lambda$ and that the partial subgradient of the objective with respect to w_j is of norm at most 2ξ. Note that, on the other hand, if $\xi_0 > 0$, the previous inequality does not hold a priori for inactive covariates, so that a solution to (4) does not necessarily have the exact same active set as one satisfying conditions (3).

To obtain a path of solutions that satisfy these approximate subgradient conditions, consider Algorithm 1.

Algorithm 1 enforces explicitly the subgradient condition (4), with $\xi_0 = \xi$, on its active set. If J is twice continuously differentiable, and if the largest eigenvalue of its Hessian is bounded above by $\mu_{\max}$, Algorithm 1 actually also enforces (4) implicitly for the other variables with $\xi_0 = \frac{1}{2}\epsilon\mu_{\max}$. This crucial property is proved in Appendix A together with the next proposition, which shows that Algorithm 1 approximates the regularization path for the ℓ_1/ℓ_2 norm:

Proposition 1 *Let λ^t denote the value of the regularization parameter at the tth iteration, with initial value $\lambda^0 \ge \|\nabla_{w_{j^*}} J(0)\|$. Assuming J to be twice differentiable and strictly convex, for all η there exists $\epsilon > 0$ and $\xi > 0$ such that iterates W^t of Algorithm 1 obey $J(W^t) - J(W(\lambda^t)) \le \eta$ for every time step t such that $\lambda^{t+1} < \lambda^t$, where $W(\lambda^t)$ is the unique solution to (2). Moreover, the algorithm terminates (provided the active set is not pruned) in a finite number of iterations to a regularization coefficient no greater than any prespecified $\lambda_{\min} > 0$.*

It is also worth noting that it is possible to set $\xi_0 = 0$ and develop a stricter version of the algorithm that identifies the correct active set for each λ. We present this variant in Appendix B.

Since our algorithm does not appeal to global second-order information, it is quite scalable compared to standard homotopy algorithms such as LARS. This is particularly useful in the multi-task setting where problems can be relatively large, and where algorithms such as LARS become slow. Our algorithm samples the path regularly, on a scale that is determined automatically by the algorithm through the update rule for λ^t, and allows for several new covariates to enter the active set simultaneously. (Empirically we find that this scale is logarithmic.) The algorithm is obviously less efficient than LARS-type algorithms in long pieces of the path that are smooth, but we indicate in the following section how variants of the algorithm could address this. Finally, our algorithm applies to contexts in which LARS-type algorithms do not apply directly, and where the use of classical homotopy methods are precluded by non-differentiability.

In the following two subsections we further describe Algorithm 1, providing further details on the prediction step (the choice of u^t) and the correction step (the enforcement of (4) for covariates in the active set).

4.1 Prediction steps

The choice $u^t = \nabla_{w_{j^*}} J / \|\nabla_{w_{j^*}} J\|$ that we have specified for the prediction step is one possible option. It is also possible to take a global gradient descent step or more generally a step along a *gradient-related descent direction* (a direction such that $\liminf_t -u^t \cdot \frac{\nabla J(W^t)}{\|\nabla J(W^t)\|} > \delta > 0$) with an update rule for the regularization coefficient of the form: $\lambda^{t+1} = \min(\lambda^t, \frac{J(W^t)-J(W^{t+1})}{\|W^t - W^{t+1}\|_{\ell_1/\ell_2}})$. Indeed, the proof of Appendix A could easily be generalized to the case of steps of ℓ_1/ℓ_2 norm ϵ taken along a general descent direction. Note that only the iterates that conclude with a decrease of the regularization coefficient are guaranteed to be close to the path.

For simplicity, we have presented the algorithm as using a fixed step size ϵ, but in practice we recommend using an adaptive step size determined by a line search limited to the segment $(0, \epsilon]$. This allows us to explore the end of the path where the regularization coefficient becomes exponentially small. Lemma 3 in Appendix A considers this case.

If we understand the "active set" as the set of covariates with non-zero coefficients it is possible for a covariate to enter and later exit the set, which, a priori, would require pruning. The analysis of pruning is delicate and we do not consider it here. In practice, the case of parameters returning to zero appears to be rare—in our experiments typically at most two components return to zero per path. Thus, implementing a pruning step would not yield a significant speed-up of the algorithm.

4.2 Correction steps

We now turn to the correction step, in which the subgradient conditions in (4) are enforced on the active set. Note that these subgradient conditions are obtained directly from the optimization problem in (2), and thus any procedure that can be used to solve the latter optimization problem can be adapted for the correction step of our algorithm. In particular, we have chosen to implement this step via a block-wise quasi-Newton algorithm developed by Tseng and Yun (2008). This algorithm, which is applicable to general optimization problems with a separable conic-regularizer, has been used by Meier et al. (2008) to solve logistic regression with a block-norm regularization. Those authors show that Tseng and Yun's algorithm compares favorably with a number of alternatives, including projected gradient and path-following algorithms (Kim et al. 2006; Park and Hastie 2006). The algorithm is particularly appropriate for our correction step, because it maintains sparse solutions.

It is also possible to use Tseng and Yun's algorithm directly to solve the optimization problem in (2), solving the problem on a grid of values of the regularization coefficient. In Sect. 6, we compare this approach to our path-following approach (in which Tseng and Yun's algorithm is used in the inner loop as a correction step).

In the experimental section we also compare to an algorithm introduced by Argyriou et al. (2008). These authors introduce a quadratic regularizer parameterized by a diagonal positive semidefinite matrix Σ with bounded trace, and show that the ℓ_1/ℓ_2 norm is recovered by minimizing over Σ. They thus propose an alternating minimization scheme, where Σ and the parameters $w^{(t)}$ are optimized in turn. A weakness of this approach is that although the solution is sparse in both Σ and $w^{(t)}$, all the feasible solutions that are considered by the algorithm are non-sparse. This makes the algorithm undesirable as an implementation of our correction step. We do, however, evaluate the algorithm empirically as an alternative to our approach and to the direct usage of the Tseng and Yun algorithm.

5 Subspace selection

Covariate selection is a specific instance of the broader problem of dimensionality reduction of the covariate space. In this section, we consider an extension of our approach to the problem of selecting general subspaces (i.e., linear combinations of covariates). In particular, we consider situations in which a subspace that is useful across multiple tasks is not aligned with the original covariate coordinate system, such that the models are sparse in a rotated coordinate system.

The general problem of subspace selection in the context of a regression or classification problem is referred to as *sufficient dimension reduction*. There has been a large literature on sufficient dimension reduction (e.g., Chiaromonte and Cook 2002; Fukumizu et al. 2008; Li 1991), but the focus has been on univariate response variables. The extension to multiple response variables has been considered by Ando and Zhang (2005) and Argyriou et al. (2008). In this section we review these ideas and then present our proposal.

Ando and Zhang (2005) treat the multiple response problem by introducing a low-dimensional subspace of dimension h common to the response variables, defining the parameter vector w^k for the kth response as $w^k = U_h a^k + v^k$, where the columns of the matrix U_h form a basis of the common subspace and where v^k lies outside of the common subspace. They propose to regularize only the components v^k. This leads to the optimization problem:

$$\min_{v^k,a^k,U_h} \sum_{k=1}^{K}\left\{\sum_{i=1}^{N_k} J^k(w^k\cdot x_i^k, y_i^k) + \lambda\|v^k\|^2\right\}$$

$$\text{s.t.}\quad w^k = U_h a^k + v^k,$$

$$a^k \in \mathbb{R}^h, \qquad v^k \in \mathbb{R}^p,$$

$$U_h \in \mathbb{R}^{p\times h}, \qquad U_h^\top U_h = I_h.$$

They present an alternating optimization scheme that simultaneously estimates the parameter vectors w^k and the matrix U_h. The basis of the common space is shown to be the best approximation of rank h of the matrix of parameters $W = [w^1, \dots, w^K]$ and it can be obtained by a singular value decomposition of the latter.

Argyriou et al. (2008) consider a formulation in which the dimension h is not fixed a priori: a common basis U for subspaces of increasing sizes is considered and in this basis the matrix A of parameter coefficients is penalized by the ℓ_1/ℓ_2 norm. The optimization problem they consider is thus:

$$\min_{a^k,U} \sum_{k=1}^{K}\sum_{i=1}^{N_k} J^k(w^k\cdot x_i^k, y_i^k) + \lambda\|A\|^2_{\ell_1/\ell_2}$$

$$\text{s.t.}\quad w^k = Ua^k,$$

$$a^k \in \mathbb{R}^p, \qquad A = [a^1, \dots, a^K],$$

$$U \in \mathbb{R}^{p\times p}, \qquad U^\top U = I_p.$$

The authors show that this regularization scheme is equivalent to a regularization of the trace norm of the matrix of parameter vectors, where the trace norm is defined by $\|W\|_{\mathrm{tr}} = \mathrm{tr}(\sqrt{\mathrm{W}^\top \mathrm{W}})$. They showed that this regularization problem can be solved by an alternating minimization algorithm that involves iterating singular value decompositions.

More generally, when the dimension h is not known *a priori*, if the data interacts with parameters of the model linearly, as is the case for the two methods presented above, then, by duality, the selection of a *joint subspace* of low dimension is equivalent to choosing a low-rank parameter matrix. Rank constraints are non-convex, and thus various convex relaxations have been proposed to select matrices with low rank (Fazel et al. 2001, 2003). In particular, the trace norm, used by Argyriou et al. (2008), has been the focus of a recent theoretical literature Srebro et al. (2005a); Srebro and Shraibman (2005); Bach (2008); Recht et al. (2008). These authors have shown that trace-norm regularization retrieves a matrix with the optimal rank under appropriate technical conditions.

In this section we present a seemingly very different approach to the subspace selection problem in which we use random projections to reduce the problem to the covariate selection problem. We then solve the induced covariate selection problem using ℓ_1/ℓ_2-regularization. As it turns out, this approach is actually an indirect method for trace-norm regularization in disguise. Indeed, as we show in this section, as the number of random projections increases, the solution to the random projections problem converges to a solution to the trace-norm regularization problem Argyriou et al. (2008).

An appealing aspect of this approach is that it avoids the direct optimization of the trace norm; this is desirable because it is difficult to optimize the trace norm directly.

We now describe the random projections method. Let Φ be a random $p \times d$ projection matrix whose columns are uniformly drawn from the unit sphere $\mathcal{S}^{p-1}$ in $\mathbb{R}^p$. Transform all of the covariate vectors via $z = \Phi^\top x$, where $x \in \mathbb{R}^p$ and $z \in \mathbb{R}^d$. In this new representation of the data, use ℓ_1/ℓ_2 regularization to perform joint covariate selection. The covariates selected in $\mathbb{R}^d$ correspond to a common relevant subset of directions in the original space. Intuitively, we would expect for this procedure to find projections that are useful across tasks, thus uncovering a common subspace linking the tasks.

The main advantage of our approximation is that it does not require singular value decomposition steps, which are at the core of the algorithms of Ando and Zhang (2005) and Argyriou et al. (2008). This makes the method potentially more scalable in spite of the fact that many random projections might be needed to obtain a good approximation.

We now present a theoretical result linking the random projection approach to trace-norm regularization. In particular, we show that sequences of solutions of the covariate selection problem based on random projections converge to a solution of the trace-norm regularization problem. Let $J(W) = \sum_{k=1}^K \sum_{i=1}^{N_k} J^k(w^k \cdot x_i^k, y_i^k)$ and note that we have $J(\Phi W) = \sum_{k=1}^K \sum_{i=1}^{N_k} J^k(w^k \cdot \Phi^\top x_i^k, y_i^k)$.

Proposition 2 *Let $\Phi_d \in \mathbb{R}^{p\times d}$ be a random projection matrix whose columns are uniformly drawn from the unit sphere $\mathcal{S}^{p-1}$ in $\mathbb{R}^p$ and let $W \in \mathbb{R}^{p\times K}$ and $\widetilde{W}_d \in \mathbb{R}^{d\times K}$ be parameter matrices. Consider the following two optimization problems*:

$$\min_W \quad J(W) + \lambda \|W\|_{\mathrm{tr}}^2 \tag{5}$$

$$\min_{\widetilde{W}_d} \quad J(\Phi_d \widetilde{W}_d) + \lambda \|\widetilde{W}_d\|_{\ell_1/\ell_2}^2. \tag{6}$$

If J is convex, continuous and lower bounded, then as the number of random projections d increases, the solutions $W_d^* = \Phi_d \widetilde{W}_d^*$ obtained from (6) form a sequence whose accumulation points are optimal solutions for (5) almost surely.

The proof of this proposition is presented in Appendix C. This result provides a clean link between the covariate selection approach based on random projections and trace-norm regularization. Given the existence of computationally-efficient algorithms for solving the covariate selection problem, we have reason to hope that this reduction will yield useful algorithms for solving the subspace selection problem. Of course, a weakness of the result is that it does not characterize the number of random projections needed to approximate the trace norm or to achieve comparable prediction performance. We thus turn to empirical evaluations to study the method further; see Sect. 6.5. Intuitively, one should use more random projections than the dimension of the space to generate sufficiently many directions so that any fixed direction is approximately in the span of a small number of random projections. Empirically we find that using 5 to 10 times p projections seems to work well.

5.1 Kernelized subspace selection

In this section we outline the form taken by our joint subspace selection algorithm when the ambient space is a (possibly infinite-dimensional) Reproducing Kernel Hilbert Space (RKHS). Our presentation will be brief, focusing on the essential theoretical concepts underlying the construction.

First, we note that a representer theorem has been established for spectral regularizers—a family which includes the trace norm—by Abernethy et al. (2008). When applied to the problem in (5), Theorem 3 of Abernethy et al. (2008) states that the columns w^{k*} of the optimal solution W^* belong to the span of the datapoints pooled from all tasks, which is a finite-dimensional space. Second, note that random directions in that space can be obtained by forming random linear combinations of the datapoints and renormalizing these combinations. Indeed, the sampling of standard Gaussian combinations of the datapoints corresponds to sampling points in the RKHS according to a Gaussian whose covariance is the empirical covariance matrix of the datapoints in the RKHS. If we denote by g the kernel function, then the projection of a datapoint x_i^k on the direction of a random point u_j is just $g(x_i^k, u_j)/\sqrt{g(u_j, u_j)}$. The representation of the data $(g(x_i^k, u_j)/\sqrt{g(u_j, u_j)})_{(k,i),j}$ by its projections on a set of random directions is therefore obtained by appropriately renormalizing random combinations of the columns of kernel matrix computed on all data points. We then apply Algorithm 1 on the transformed data. Finally, if needed, the kernel coefficients in the representer theorem can be obtained by an inversion of the matrix of random combinations.

A possible drawback of this construction is that for a large number of datapoints the dimensionality of the space may become very large and a large number of random directions may be needed to approximate directions in that space.

6 Experiments and applications

In this section we present experiments which aim to evaluate methods for solving the joint covariate selection and joint subspace selection problems. We first investigate simulated data sets in which the generative mechanism satisfies the assumptions underlying our model and analysis. We then turn to experiments with real data, focusing on optical handwritten character recognition. We also consider the case of multi-class classification. Finally, we turn to the joint subspace selection problem.

6.1 Experimental setup

In all experiments comparing the performance of different regularization schemes we study four setups:

- **Independent ℓ_1-regularization:** For each task an independent ℓ_1-regularized logistic regression is fitted. This is done by using Algorithm 1 specialized to the case of blocks of size one.
- **ℓ_1/ℓ_1-regularization:** The objective function is (1) with the logistic loss and tasks are thereby tied only by the regularization coefficient. The regularization path is obtained for all tasks simultaneously by Algorithm 1 with blocks of size one. Covariates enter the active set separately for the different tasks.
- **ℓ_1/ℓ_2-regularization:** The objective is (2) with the logistic loss. In this case the covariate selection processes are coupled by the regularization. The regularization path is obtained by Algorithm 1.
- **Pooled ℓ_1:** When the different classification tasks are very similar, it may make sense to consider merging the tasks into a single classification problem in which the positive examples and negative examples are pooled across tasks. In this case we fit a single logistic regression with ℓ_1-regularization.

For each of these schemes, we fit the regularization path using 3/4 of the data in the training set, retaining 1/4 of the data as a validation set to choose the regularization coefficient λ (as the maximizing value along the path). We then report results on a separate test set.

6.2 Synthetic data

We consider K binary classification tasks on a covariate space of dimension p. We generate data such that there exists a subset of $r \ll p$ covariates that defines a subspace $\mathcal{D}$ that discriminates between the two classes for each of the K classification tasks. In particular, a classification task is defined by a pair of Gaussian class-conditional densities where both class-conditional densities are Gaussian on $\mathcal{D}$, with the vector components in the remaining $p - r$ dimensions consisting of noise uniformly distributed on the interval [0, 1]. The covariance matrix for each class is drawn from an $r \times r$-dimensional Wishart distribution, $\mathcal{W}(r, r, Id)$, with r degrees of freedom. Pairs of classes are separated by a vector $\delta = \mu_1 - \mu_0$ constructed as follows: a random vector is drawn uniformly in $\{-1, 0, 1\}^r \backslash \{\mathbf{0}\}$ and then normalized so that the mean of the Mahalanobis distances for both covariance matrices is a fixed value $c = \frac{1}{2}\sqrt{\delta^\top \Sigma_0 \delta} + \frac{1}{2}\sqrt{\delta^\top \Sigma_1 \delta}$. We picked $c = 3$ in our experiments which corresponds to well-separated classes. Note that by construction, the coordinates of δ are non-zero only on a subset of the r common dimensions, so that the set of covariates that separates the classes is not exactly the same for each classification.

6.2.1 Comparison of regularization schemes

We first focus on the relative performance obtained with the different regularization schemes. The results averaged over ten replications are shown in Fig. 2, where we compare independent ℓ_1, ℓ_1/ℓ_1 and ℓ_1/ℓ_2-regularizations. The results indicate that the ℓ_1/ℓ_1 and independent ℓ_1-regularizations perform almost identically. This is not surprising because the essential difference between the behavior of these two regularizations is that the regularization coefficient is shared across tasks in the ℓ_1/ℓ_1 case, while a different value of the regularization can be chosen (via cross-validation) in the case of independent ℓ_1-regularizations. But the classification problems we generated are of equal difficulty, which means that the amount of regularization that is needed for each problem is presumably the same. On the other hand we see from Fig. 2 that the ℓ_1/ℓ_2-regularization yields systematically better results, with dramatic improvements for small training set sizes. Indeed, the error rate decreases initially much faster with the training size when the ℓ_1/ℓ_2-regularization is used in comparison to the other regularizations. As a consequence, the relative improvement is generally larger for small training sets. For large training set sizes all of the regularization schemes seem to yield the

Fig. 2 (Color online) Misclassification error represented as a function of the number n of samples used for training, in plots with increasing number of tasks (*from left to right*: $K = 2, 5, 10, 50$) and increasing number of total covariates (*from top to bottom*: $p = 100, 500, 1000, 5000$), for a fixed number $r = 20$ of informative covariates, and for three different algorithms based on either independent ℓ_1-regularization (*dotted red*), ℓ_1/ℓ_1-regularization (*green*) or ℓ_1/ℓ_2-regularization (*dashed blue*). Error bars at one standard deviation are estimated from 5 replicates for each curve. Note that the misclassification error decreases initially faster as function of the training size for ℓ_1/ℓ_2 than for the other regularizations. The relative improvement is more pronounced for larger number of tasks and larger ambient dimension

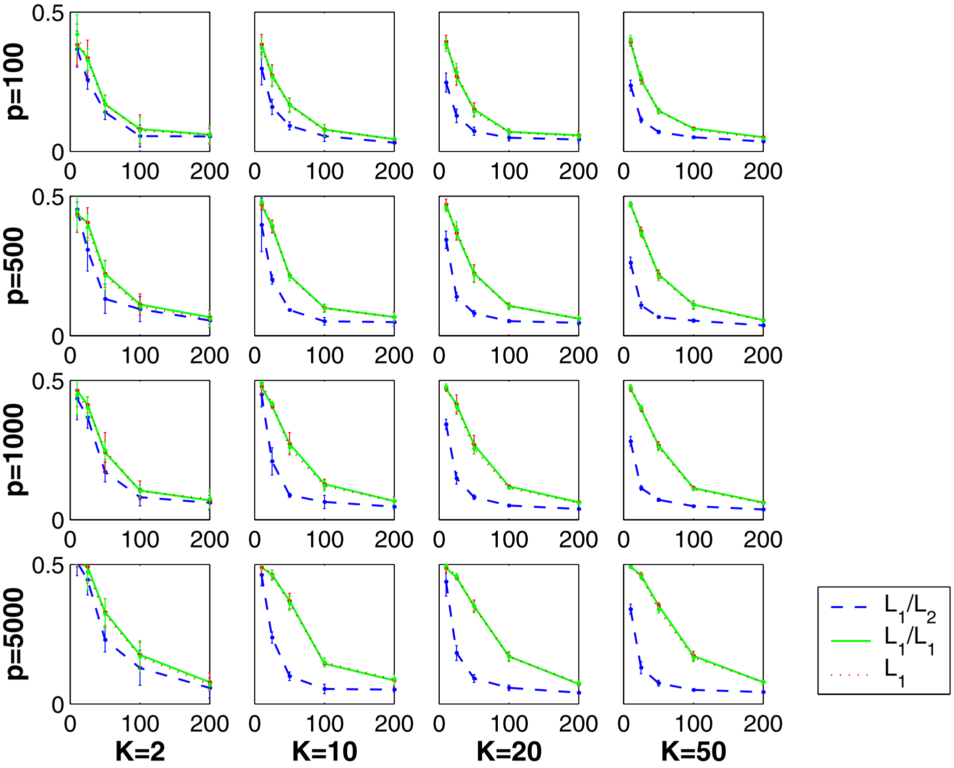
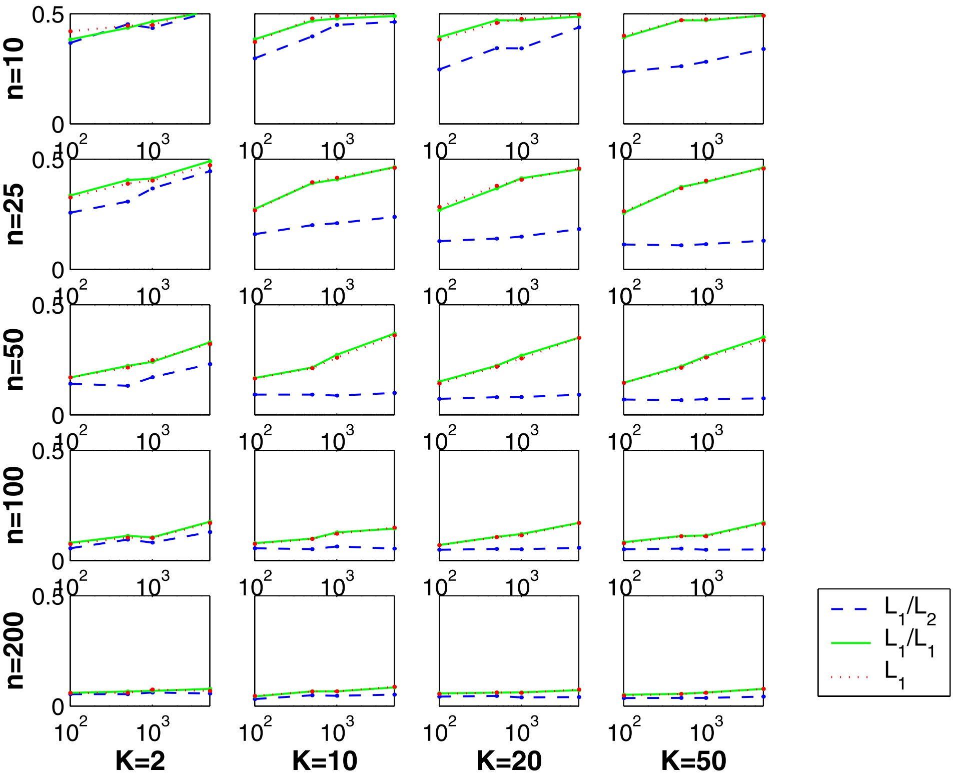


Fig. 3 (Color online) Average misclassification error represented as a function of the total number p of covariates (on a log scale), for a fixed number $r = 10$ of informative covariates, in plots with increasing number of tasks (*from left to right*: $K = 2, 5, 10, 50$) and increasing number of datapoints (*from top to bottom*: $n = 10, 25, 50, 100, 200$), and for three different algorithms based on either independent ℓ_1-regularization (*dotted red*), ℓ_1/ℓ_1-regularization (*green*) or ℓ_1/ℓ_2-regularization (*dashed blue*). The average is based on five replicates



same asymptotic value. The relative improvement is accentuated for larger number of tasks and for larger number of dimensions.

Figure 3 illustrates that ℓ_1/ℓ_2 is more robust to the number of noisy dimensions than the other regularizations, and suggests that the growth of the error is roughly linear with log p but that the slope decreases significantly with the number of tasks.

6.2.2 Comparison with other algorithms

In this section we report the results of comparisons with our implementations of the algorithms of Tseng and Yun (2008) (henceforth "TY") and the algorithm of Argyriou et al. (2008) (henceforth "AEP"). These algorithms are not path-following algorithms, and they must be evaluated on a grid of regularization coefficients. To enhance the speed of these algorithms, we implemented a "warm-start" technique in which the algorithm was run for decreasing values of the regularization coefficient and at each gridpoint the previous optimal solution was used as an initializer.

The choice of the grid values for λ is not easy to make a priori for these algorithms (which is an argument in favor of the use of path-following algorithms). Given that for $\lambda \geq \lambda_0 = \max_j \|\nabla_{w_j} J(0)\|_2$ the solution is the trivial null solution, we need only consider regularization coefficients smaller than λ_0. We found that using equally-spaced quantiles of the distribution of initial gradients was unsatisfactory—most gradients decrease significantly along the path and thus this approach does not explore far enough along the path. We instead noted that both Algorithms 1 and 2 tend to decrease the values of λ exponentially; thus we adopted the heuristic of selecting grid points for λ to be equally spaced on a logarithmic scale between λ_0 and $\lambda_0/500$.

For the TY algorithm and the AEP algorithm, we also studied a heuristic which consists of guessing the active set in advance based on the norms of gradients associated to each block. In particular, we only consider those covariates that have parameter vector with gradient in ℓ_2 norm larger than λ_t; we then solve the restricted optimization problem, check if additional covariates need to be included and, if so, iterate.

We first compare the TY algorithm and the AEP algorithm in terms of speed, using only four values of λ along the path to maximize computational efficiency. In the same experiment we also evaluate the active set heuristic. We use stabilization of performance on a test set as a stopping criterion. From the results are reported in Table 1 we see that the TY algorithm is significantly faster than the AEP algorithm.

Based on these results we retained only the TY algorithm in the comparison of grid search methods to our path-following algorithm (specifically, Algorithm 1). Using as a stopping criterion the attainment of an approximate subgradient condition on the active set, $\xi \leq \min\{10^{-3}, 0.01\lambda\}$, and using ten grid points for the TY algorithm, we compared the algorithms in prediction performance, sparsity of solutions and speed. We varied the number of tasks, the dimension of covariate space and the sample size.

Table 1 Comparisons of running times. TY I is a grid search based on the TY algorithm with a heuristic preselection of the active set. TY II is the same without preselection. AEP is our implementation of the AEP algorithm. Times were measured in seconds and were averaged over ten runs of the algorithm on different data sets. Some running times are not monotone in the size of the dataset, presumably because bigger data sets yield more strongly convex objectives

K	p	r	n	TY I	TY II	AEP
2	100	20	10	15	17	52
2	100	20	100	5	9	95
2	100	20	200	4	9	–
10	100	20	10	41	37	209
10	100	20	100	25	22	279
10	100	20	200	31	32	–
50	100	20	10	91	77	480
50	100	20	100	124	124	872
50	100	20	200	217	218	–
2	500	20	10	50	71	3486
2	500	20	100	22	45	6629
2	500	20	200	16	40	–
10	500	20	10	170	153	12818
10	500	20	100	102	83	22623
10	500	20	200	124	114	–
50	500	20	10	385	358	24171
50	500	20	100	437	403	–

Figure 4 presents the relative prediction error for the path-following algorithm and the TY algorithm (numbers less than one indicate smaller error for the path-following algorithm). We see that the performance achieved by the path-following approach tends to be better than that of the TY algorithm. Moreover, from Fig. 5 we see that the solutions obtained from path-following are significantly sparser than those obtained from the TY algorithm. Finally, Fig. 6 shows that the running times of the two algorithms as we have implemented them are comparable. Indeed, in the case of large values of the covariate dimension, the path-following algorithm is actually faster than the TY algorithm. Thus, in this case we are able to obtain the entire regularization path more quickly than its evaluation at a set of grid points via the TY algorithm.

We also compared Algorithm 1 with the stricter Algorithm 2 in Appendix B. We found (results not reported) that the prediction performance of the two algorithms is essentially identical. Algorithm 2 was slightly slower for larger number of datapoints, presumably because identifying exactly the active set for each regularization value increases the total number of function evaluations. However, this behavior was only observed for small numbers of tasks; for larger numbers of tasks the two algorithms were equally fast.

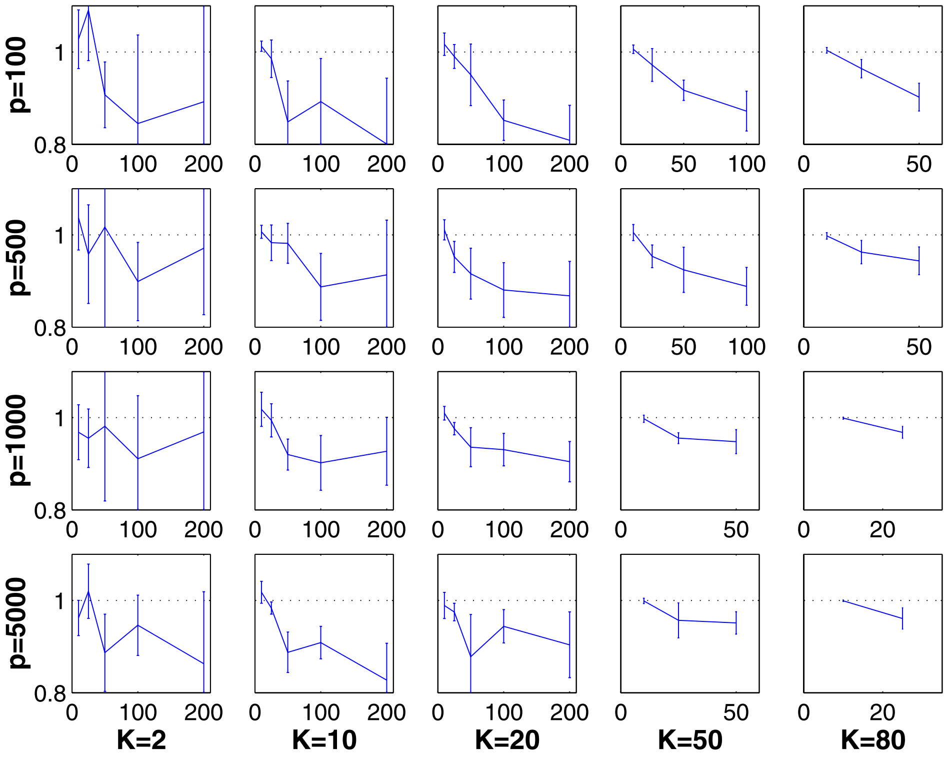


Fig. 4 Average of the ratio of the error rate on the test set for Algorithm 1 and the TY algorithm. These ratios are based on five replicates, and one standard deviation confidence intervals are indicated. Note that the average error rate of Algorithm 1 is almost always smaller than that of the TY algorithm. The improvement in error rate is typically significant for larger number of tasks and larger ambient dimension

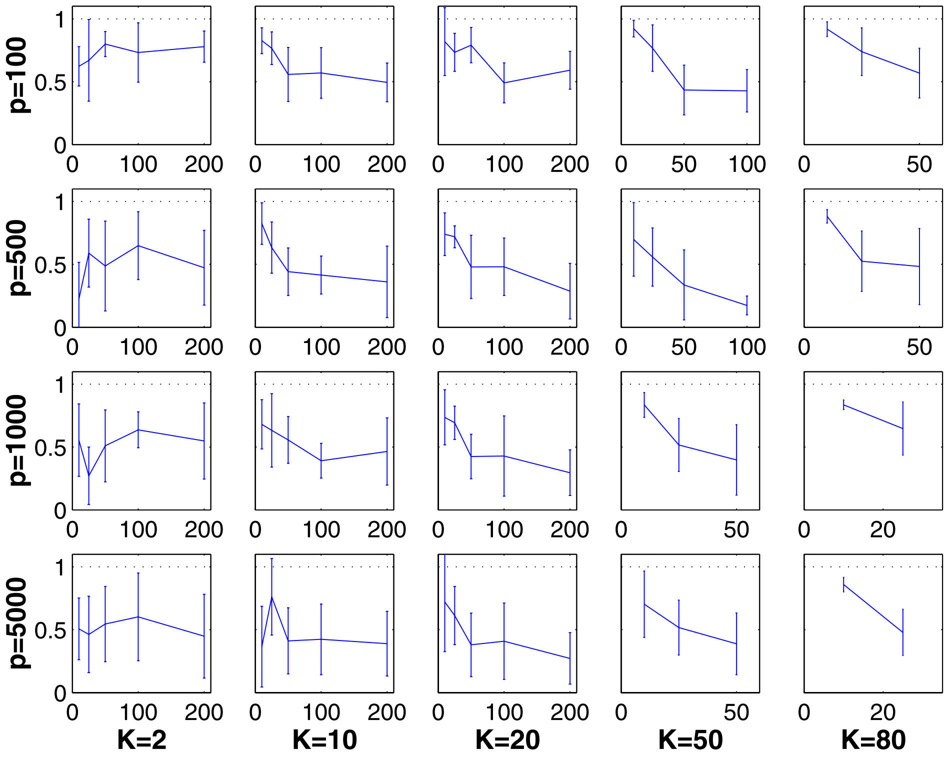


Fig. 5 Average of the ratio of the number of active covariates for Algorithm 1 and the same quantity for the TY algorithm. These ratios are based on five replicates, and one standard deviation confidence intervals are indicated. The models selected by Algorithm 1 are almost always sparser than those returned by the TY algorithm

6.2.3 Approximation of the path

To assess how well the path is approximated by Algorithm 1, we compared the solutions on the exact path with solutions obtained from the algorithm. We generated an instance of the synthetic data with $K = 5$ tasks and $r = 20$ discriminative dimensions out of $p = 100$, and a training set of size $n = 100$. We set the step size to $\epsilon = 0.02$ and we let

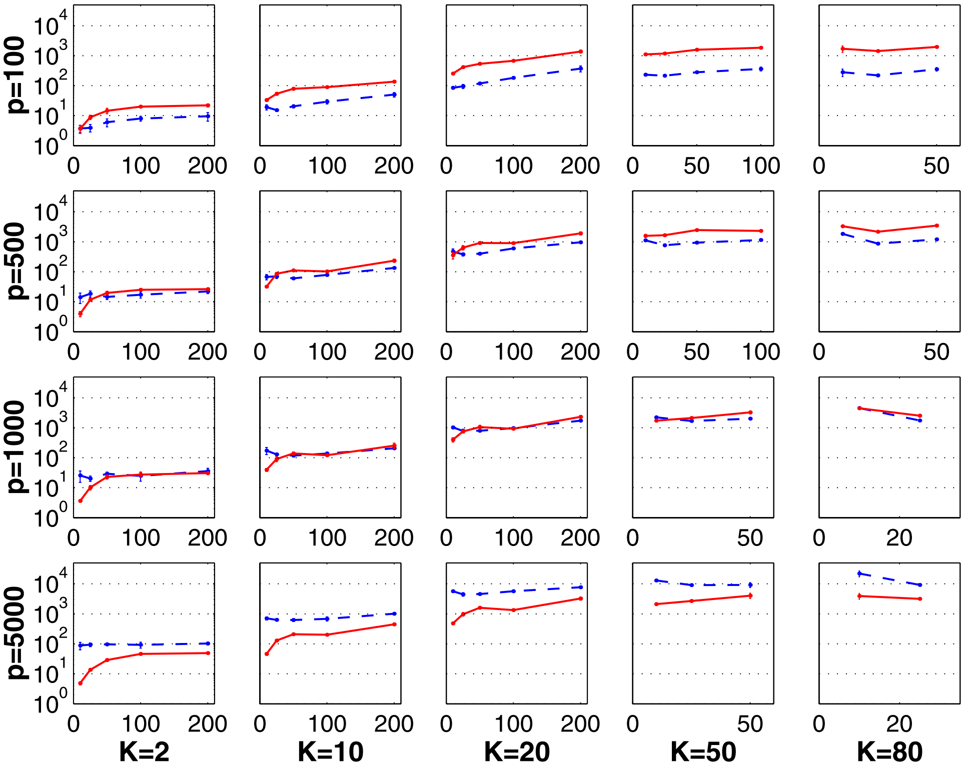


Fig. 6 (Color online) Running times for Algorithm 1 (*solid red curve*) and the TY algorithm (*dashed blue curve*)

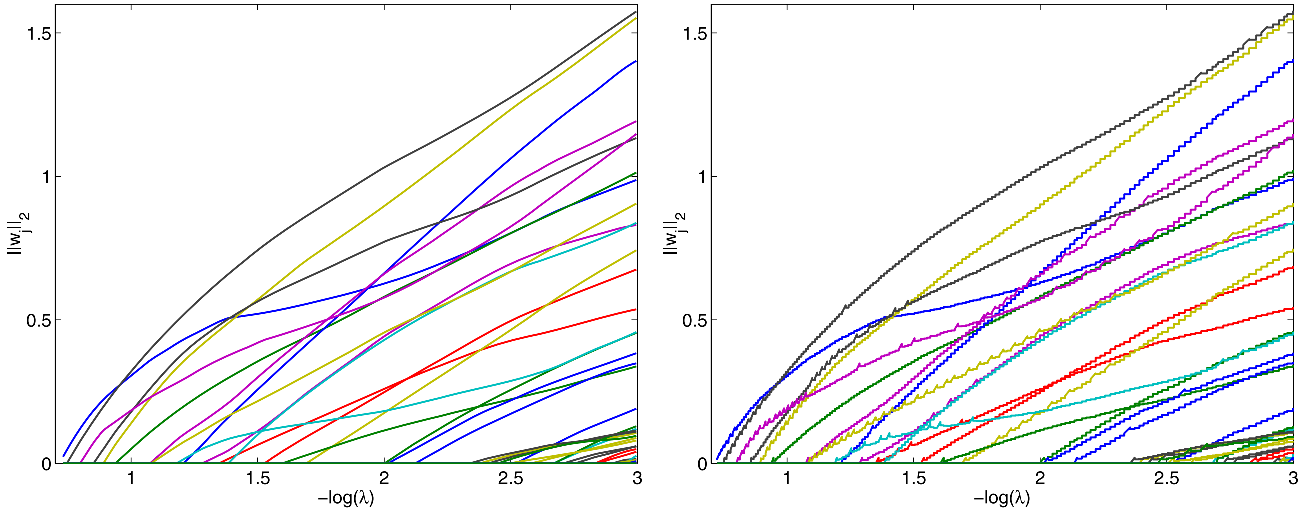


Fig. 7 Exact regularization path (*left*) and approximated path obtained with Algorithm 1 (*right*). In both plots, the relevance of each covariate, as measured by $\|w_j\|_2$, is plotted as a function of $-\log(\lambda)$, where λ is the regularization parameter

$\xi = \min(0.001, 0.1\lambda)$. Figure 7 illustrates the approximation of the regularization path obtained with algrefbblasso, where the value plotted for each covariate is the norm $\|w_j\|_2$ (intuitively, a measure of relevance of the covariate). As shown in the figure, the $\|w_j\|_2$ are well approximated. Similar results were obtained for each w_j^k individually (data not shown).

6.3 Writer-specific character recognition

In this section, we investigate an application to the problem of the optical character recognition (OCR) of handwritten characters. Consider the problem of discriminating between pairs of letters for different writers. The simplest approach is to pool all the letters from all writers and build a global

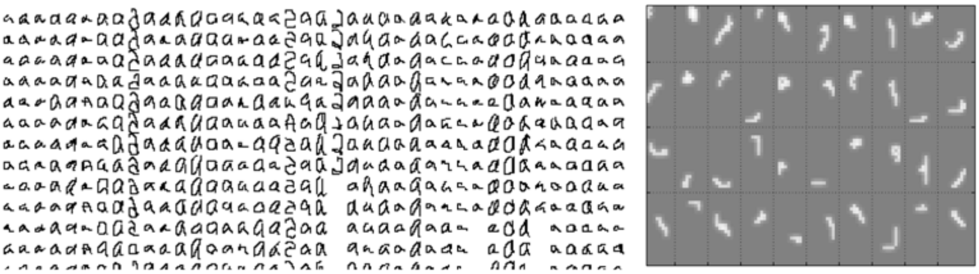


Fig. 8 (*Left*) The letter *a* written by 40 different people. (*Right*) Strokes extracted from the data

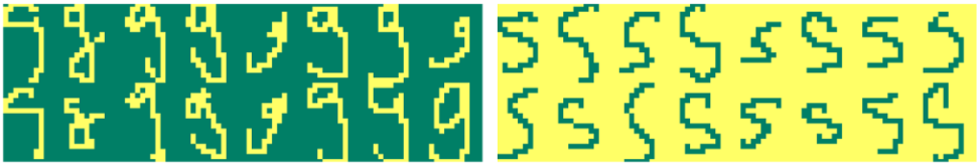


Fig. 9 Samples of the letters *s* and *g* for one writer

classifier for each pair; this may be justifiable if we obtain only a few examples of each letter per writer, but large numbers of different writers. Another naive method is to learn a classifier for each writer independently. We compare these naive methods to our ℓ_1/ℓ_2 regularization method.

6.3.1 Data

We used letters from a handwritten words data gathered by Rob Kassel at the MIT Spoken Language Systems Group.[1] This data set contains samples from more than 180 different writers (see Fig. 8, left, for examples). For each writer, however, the number of examples of each letter is rather small: between 4 and 30 depending on the letter. As shown in Fig. 9, the letters are originally represented as 8 × 16 binary pixel images.

6.3.2 Covariates: pixels and strokes

The basic covariates we use are the 8 × 16 binary pixels. Since individual pixels are often uninformative, we also used a simple, ad hoc procedure to generate combinations of contiguous pixels ("strokes") that appeared in the images.

To produce a stroke, we select a random image and a random filled pixel and follow a biased random walk on the filled pixels of the image. We use an second-order Gaussian Markov model of strokes in which the velocity varies slowly to bias for low-curvature lines and generated walks of length two, four and six pixels. To produce realistically thick strokes we then include the pixels of the letters that are neighbors of the stroke. The obtained stroke are finally smoothed by convolution with a simple kernel combining only neighboring pixels. For a new letter, the covariate associated with a stroke is the scalar obtained as the dot product between the image of the letter and the image of the stroke both considered as vectors in $\mathbb{R}^{8\times 16}$. To construct a set of strokes for the task of discriminating between two letters we extracted 500 strokes in the training set from letters of each of these two types and 100 strokes from other letter types as well. The total number of strokes we generated in each of our experiments was on the order of a thousand. The strokes selected by our algorithm for the *g* vs. *s* classification are shown in Fig. 8(right).

6.3.3 Setup

We built binary classifiers that discriminate between pairs of letters. Specifically we concentrated on the pairs of letters that are difficult to distinguish when written by hand. We compared the four discriminative methods presented at the beginning of Sect. 6.1. For the pooled ℓ_1 scheme, the writers are ignored and all the letters of both classes to be discriminated are pooled. For all other schemes, a separate model is fitted for each writer with either an independent ℓ_1 regularization or a ℓ_1/ℓ_1 or ℓ_1/ℓ_2 joint regularization.

6.3.4 Results

We fitted classification models for discriminating nine pairs of letters for 40 different writers according to the four schemes presented in Sect. 6.3. We conducted experiments with the two types of covariate sets proposed (pixels and strokes). The error rates of the classifiers obtained are reported in Table 2.

For the pixel covariates, the ℓ_1/ℓ_2-regularization method improves significantly on pooling and on the other regularization methods. Indeed, it improves in all cases except one, with an improvement over ℓ_1-regularization that is greater than 50% in many cases.

For the stroke covariates the improvement due to the ℓ_1/ℓ_2-regularization is less pronounced. There is a clear improvement over pooling and over ℓ_1/ℓ_1; on the other hand, ℓ_1 and ℓ_1/ℓ_2-regularizations perform comparably.

[1] Available at www.seas.upenn.edu/~taskar/ocr/.

Table 2 Average 0–1 loss on the test set, for covariate selection (left) and subspace selection (right), in the case of pixel features or stroke features, for the four schemes proposed. The bold font indicates the best-performing scheme among ℓ_1/ℓ_2, ℓ_1/ℓ_1, independent (id.) ℓ_1 or pooled ℓ_1, for a fixed type of covariate. The boxed entry indicates conditions in which performing subspace selection led to an improvement of the average 0–1 loss over the covariate selection, with the same type of covariate

Task	Covariate selection								Subspace selection							
	Strokes: error(%)				Pixels: error (%)				Strokes: error(%)				Pixels: error (%)			
	ℓ_1/ℓ_2	ℓ_1/ℓ_1	id.ℓ_1	Pool	ℓ_1/ℓ_2	ℓ_1/ℓ_1	id.ℓ_1	Pool	ℓ_1/ℓ_2	ℓ_1/ℓ_1	id.ℓ_1	Pool	ℓ_1/ℓ_2	ℓ_1/ℓ_1	id.ℓ_1	Pool
c/e	**2.5**	3.0	3.3	3.0	**4.0**	8.5	9.0	4.5	$\boxed{\mathbf{2.0}}$	3.5	3.3	2.5	$\boxed{\mathbf{3.5}}$	7.8	10.3	4.5
g/y	8.4	11.3	**8.1**	17.8	**11.4**	16.1	17.2	18.6	10.3	10.3	**9.3**	16.9	11.6	$\boxed{\mathbf{9.7}}$	10.9	21.4
g/s	3.3	3.8	**3.0**	10.7	**4.4**	10.0	10.3	6.9	3.8	4.0	$\boxed{\mathbf{2.5}}$	12.0	4.7	6.7	5.0	6.4
m/n	4.4	4.4	**3.6**	4.7	**2.5**	6.3	6.9	4.1	4.1	5.8	$\boxed{\mathbf{3.6}}$	5.3	$\boxed{\mathbf{1.9}}$	2.8	4.1	–
a/g	**1.4**	2.8	2.2	2.8	**1.3**	3.6	4.1	3.6	**0.8**	1.6	1.3	2.5	$\boxed{\mathbf{0.8}}$	1.7	1.4	3.9
i/j	**8.9**	9.5	9.5	11.5	12.0	14.0	14.0	**11.3**	**9.2**	9.8	11.1	11.3	$\boxed{\mathbf{10.3}}$	12.7	13.5	11.5
a/o	**2.0**	2.9	2.3	3.8	**2.8**	4.8	5.2	4.2	2.7	2.7	$\boxed{\mathbf{1.9}}$	4.3	$\boxed{\mathbf{2.1}}$	3.1	3.5	4.2
f/t	**4.0**	5.0	6.0	8.1	**5.0**	6.7	6.1	8.2	5.8	**4.1**	5.5	7.5	**6.4**	11.1	9.6	7.1
h/n	**0.9**	1.6	1.9	3.4	**3.2**	14.3	18.6	5.0	0.9	0.6	$\boxed{\mathbf{0.3}}$	3.7	$\boxed{\mathbf{1.8}}$	3.6	5.0	5.0

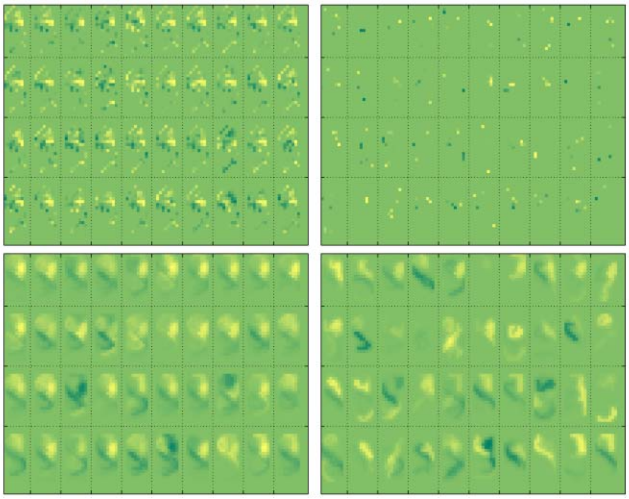


Fig. 10 (Color online) Plots of the discriminative masks learned for the classification of g vs s under ℓ_1/ℓ_2 regularization (*left*) and independent ℓ_1 regularization (*right*), based on either pixel covariates (*top*) or stroke covariates (*bottom*). Intuitively these masks should resemble a yellow letter g to which is subtracted a letter s which therefore appears by contrast in darker green. The better masks capture the (*yellow*) closure of the circle in g and the (*dark green*) diagonal stroke of s as discriminative features of these letters

Our interpretation of these results is that classifiers based on the weaker features (pixels) benefit more from the sharing among tasks than those based on the stronger features (strokes). As support for this interpretation, consider Fig. 10, where we represent the "discriminative mask" learned, i.e. a pixel image with colors ranging from yellow to dark green corresponding to individual parameter values, representing the whole vector of parameters w^k learned for each of the 40 writers. The top two rectangles contain the parameters for the pixel covariates, with the results from ℓ_1/ℓ_2-regularization on the left and the results from independent ℓ_1-regularization on the right. It is clear that the sharing induced by the ℓ_1/ℓ_2-regularization has yielded parameters that are more discriminative in this case. On the other hand, in the case of stroke covariates (the lower two rectangles), we see that the parameters induced by independent ℓ_1 are already quite discriminative; thus, there appears to be less to gain from shrinkage among tasks in this case. Note also (from Table 2) that the overall error rate from the classifiers based on pixels is significantly higher than that of the classifiers based on strokes. Finally, for this problem pooling does not perform well presumably because the inter-writer variance of the letters is large compared to the inter-class variance.

Another advantage of the ℓ_1/ℓ_2-regularization is that it yields a more compact representation than the other methods (with the exception of pooling). This is particularly noticeable for the stroke representation where fewer than 50 features are typically retained for the ℓ_1/ℓ_2-regularization versus three to five times as many for the other regularization schemes.

6.4 Multi-class classification

Multi-class classification can be viewed as a multiple response problem in which a set of responses share a set of covariates. This is certainly an appropriate perspective if the multi-class classification problem is approached (as is often done) by fitting a set of binary classifiers, but it is also appropriate if a single multi-class classifier is fit by a single "polychotomous" logistic regression. In either case, it may be useful to find covariates that are useful across the set of discriminations. Our ℓ_1/ℓ_2-regularization applies directly to this setting; indeed, the methodology that we have presented thus far makes no reference to the fact that the loss function is a sum of losses across tasks. We can thus replace this loss function with any joint loss function (e.g., the polychotomous logistic loss). In the remainder of this section we investigate the use of ℓ_1/ℓ_2-regularization in two multi-class classification domains.

6.4.1 *Digit classification*

We conducted a multi-class classification experiment using the "multi-feature digit" data set from the University of California Irvine repository (van Breukelen et al. 1998). This data set of 2000 entries contains 200 examples of each of the 10 digits. The data are represented by 649 covariates of different types (76 Fourier coefficients, 216 profile correlations, 64 Karhunen-Loève coefficients, 240 pixel averages in 2×3 windows, 47 Zernike moments and 6 morphological features). We compared models based on polychotomous logistic regression fitted with ℓ_1/ℓ_2 and ℓ_1/ℓ_1-regularizations and the classification obtained by combining individually regularized logistic regressions (using the ℓ_1 norm). To focus on the data-poor regime in which regularization methods would appear to be of most value, we used only 1/10 of the data to fit the model and retained the rest for testing. We replicated the experiment ten times.

Our results indicate that ℓ_1/ℓ_2-regularization is clearly superior for this problem compared to the other regularization methods. The average error rate obtained was 2.9% ($\hat{\sigma} = 0.24\%$) for ℓ_1/ℓ_2, versus 4.2% ($\hat{\sigma} = 0.65\%$) for ℓ_1/ℓ_1 and 4.1% ($\hat{\sigma} = 0.65\%$) for separate binary classifications.

6.4.2 *Classification of cancers*

The diagnosis of complex diseases such as cancer can be assisted by genomic information provided by expression microarrays; specifically, microarrays allow us to identify genes that are differentially expressed in different cell lineages or at different stages of a cancer. This is interesting because the relationship between gene expression patterns and the illness is more direct than that of somatic symptoms, but it is also difficult because of the large number of genes and the high levels of noise present in the data. We used the ℓ_1/ℓ_2, ℓ_1/ℓ_1 and independent ℓ_1-regularizations to differentiate four types of skin cancers (studied by Khan et al. 2001) based on gene expression data.

We found that all three of these regularization schemes performed as well in terms of predictive performance as the best-performing methods studied by Khan et al. (2001) and Wu (2005). However, ℓ_1/ℓ_2-regularization achieved this result with a smaller set of non-zero parameters than the other methods: there were 57, 81 and 85 contributing genes to the classifier based on ℓ_1/ℓ_2, ℓ_1/ℓ_1 and independent ℓ_1, respectively. This small gene signature is obviously of importance in the biological setting, where simpler/cheaper tests are desirable and where predictively-important genes may be prioritized for further study. Note also that the parameter values obtained from ℓ_1/ℓ_2-regularization were different qualitatively from those obtained via the other regularizations (see Fig. 11). We found that a striking feature of the sparsity pattern obtained from ℓ_1/ℓ_2 was that sev-

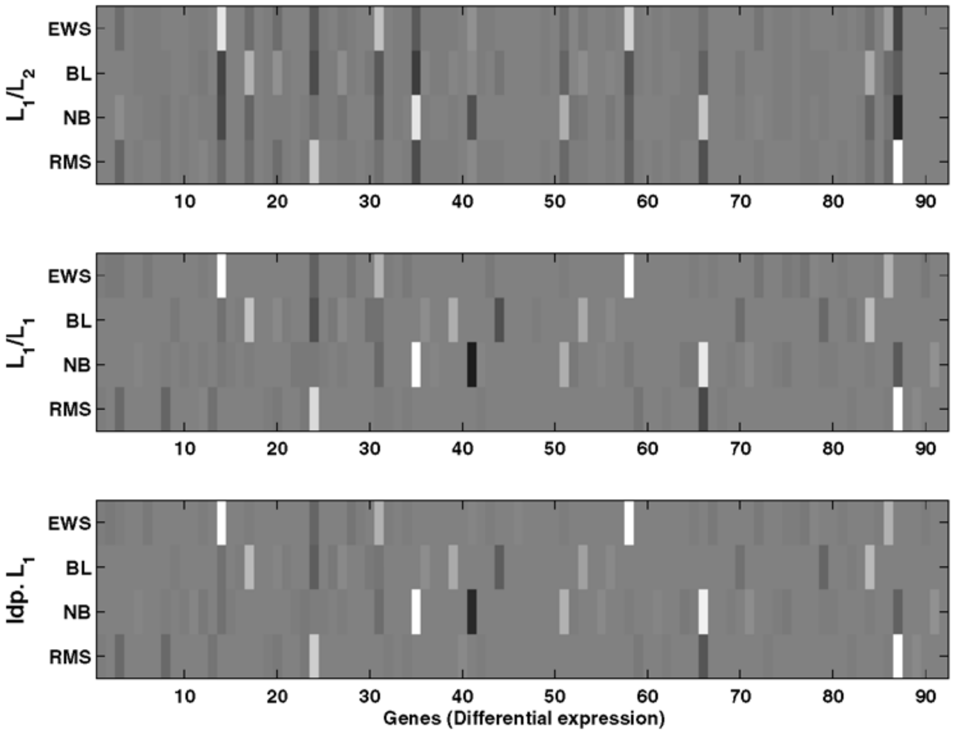
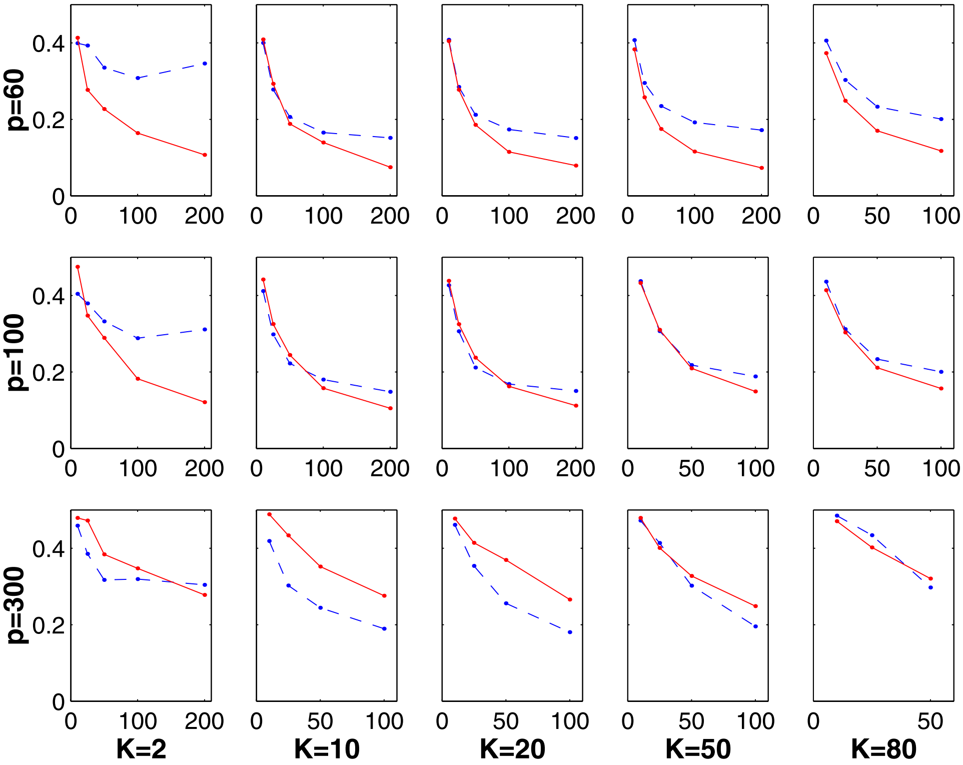


Fig. 11 Matrix of parameters obtained from three regularization methods. The ℓ_1/ℓ_2, ℓ_1/ℓ_1 and independent ℓ_1 regularizations use 57, 81 and 85 (respectively) contributing genes to classify four cancer types: EWS, BL, NB, RMS. Note that the ℓ_1/ℓ_2 regularization has an interesting "mikado" pattern (i.e., with alternating, contrasted coefficients columnwise) indicating that a given feature has important opposite effects in the classification of two classes that it discriminates well

Fig. 12 (Color online) Prediction errors of Algorithm 1 (*solid red curve*) combined with random projections and the algorithm of Argyriou et al. (2008) (*dashed blue curve*)



eral genes used by the other regularizations were eliminated because if the expression of a gene is indicative of a cancer type, then that covariate is encouraged to be also more discriminative for the other cancers. This might be an efficient way to eliminate competing correlated predictors.

6.5 Experiments on subspace selection

In this section we present an experimental evaluation of our approach to subspace selection based on random projections. We compare this approach to the alternating minimization algorithm of Argyriou et al. (2008), both in terms of speed and performance. The non-differentiability of the trace norm underlying the latter algorithm creates difficulties; these were addressed by Argyriou et al. (2008) using a numerical smoothing method. We also found that smoothing was necessary for this algorithm; moreover, we found that it was somewhat difficult to calibrate the amount of smoothing. When the smoothing was significantly large to avoid numerical difficulties, the resulting solutions tended to have a spectrum of singular values that was quite different from those of the original problem.

In a first set of experiments we returned to the artificial data described in Sect. 6.2, where we defined a 20-dimensional subspace that discriminates the pairs of classes in all tasks. For the random projections method, we used $5p$ random projections where p is the dimension of the covariate space. (Recall that these projections serve as a transformed set of coordinates to which we apply Algorithm 1.)

We report the results of the comparison in Fig. 12, where we report prediction errors, and Fig. 13, where we report running times. We see from Fig. 12 that the two methods yield comparable prediction errors, with each method outperforming the other method in a certain regime. From Fig. 13 we see that our random projections method is generally faster than the other algorithm, particularly so for high-dimensional covariate spaces. However, in the high-dimensional spaces our method was less accurate than that of Argyriou et al. Presumably this could be mitigated by choosing a larger number of random projections; however, we currently lack a theoretical basis for choosing the proper tradeoff between accuracy and efficiency in terms of the number of projections.

Finally, we report results on subspace selection using random projections in the OCR domain. We conducted an experiment that was identical to the previous OCR experiment, but in which 500 random projections were used to transform the pixel covariates into a new covariate space. Similarly, in the case of the strokes covariates we used 3000 projections. In both cases this yielded roughly four times as many projections as there were dimensions of the original covariate space. The results of this experiment are shown in Table 2. We see that the subspace selection yields an improvement

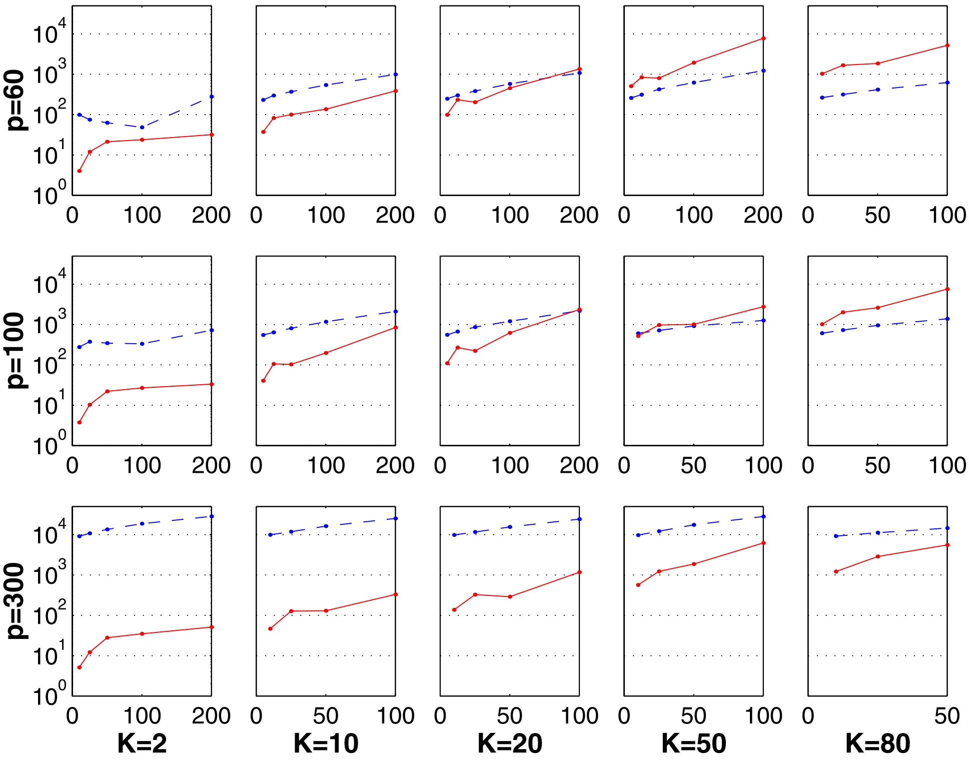


Fig. 13 (Color online) Running times of Algorithm 1 combined with random projections (*solid red curve*) and the algorithm of Argyriou et al. (2008) (*dashed blue curve*)

over the earlier covariate selection results in the case of the pixel covariates.

7 Discussion

We have considered a regularization scheme for joint covariate selection in grouped classification, where several classification models are fitted simultaneously and make simultaneous choices for relevant covariates. We have developed a path-following algorithm for solving this problem and assessed its performance in both artificial and real datasets compared to ℓ_1 and ℓ_2-regularizations. We have also developed an extension of this approach to the subspace selection problem.

We should emphasize that although classification has been the focus of our presentation, the approach is generic and applies immediately to problems based on other smooth loss functions, including least squares regression and more broadly generalized linear models. More generally, any norm that induces sparse solutions can benefit from a similar approach.

We should also point out that, even though we have used our proposed regularization scheme to fit parameters for all classifiers simultaneously, it is also possible to use this regularization scheme in a sequential fashion, where new tasks are encouraged to share the same sparsity pattern as previous classifiers. In this case, tasks are presented one after another and, in the ℓ_1/ℓ_2-regularization, parameters of previously fitted models are fixed and only the parameters for the new task are fit. A computational advantage of this approach is that it does not require retaining the previously fitted parameters in memory; rather, one only needs to keep the previously defined *relevance* of each covariate as measured by the ℓ_2 norm of parameters associated to that covariate across tasks.

There are several open theoretical questions associated with this work. First, it is of great interest to consider the *recovery problem* for ℓ_1/ℓ_2-regularization; in particular, assuming that a sparse set of covariates are relevant across multiple tasks, what are the conditions under which this set can be recovered asymptotically? Also, our empirical results suggest that the ℓ_1/ℓ_2-regularization is particularly useful for high-variance covariates (cf. the pixel features in the OCR problem) and in cases where the amount of data for each classification task is limited. It would be useful to attempt to characterize these tradeoffs theoretically.

Acknowledgements We would like to thank Simon Lacoste-Julien for helpful discussions. We also wish to acknowledge support from DARPA IPTO Contract FA8750-05-2-0249 and a grant from Google Research. Michael Jordan was supported by an appointment as a Miller Research Professor in the Miller Institute for Basic Research in Science.

Appendix A: Proof of Proposition 1

In this Appendix we prove Proposition 1, showing that the path-following algorithm that we have presented progresses steadily along the path and guaranteeing that the latter is well approximated.

The proof proceeds via a sequence of lemmas. Lemma 3 justifies the update rule $\lambda^{t+1} = \min(\lambda^t, \epsilon^{-1}[J(W^t) - J(W^{t+1})])$ by showing that it ensures that each time the regularization coefficient λ^t is updated, the solution satisfies approximate subgradient conditions and is thus, by Lemma 2, reasonably close to the path. The algorithm is designed to move along the path smoothly in parameter space, by taking a bounded step. Lemmas 4 and 5 establish that the progression is steady in terms of λ^t and that the algorithm terminates after a finite number of steps. More precisely, Lemma 4 shows that the regularization decreases by at least a constant amount $\epsilon\mu_{\min}$ at almost each iteration and therefore becomes smaller than $\epsilon\mu_{\min}/2$ after a finite number of steps. Lemma 5 establishes additionally that even the part of the path corresponding to small values of the regularization can be reached efficiently after a finite number of steps if a bounded line search method is used to determine the step size of the descent steps on J.

All lemmas assume that J is convex, continuously twice differentiable ($\mathcal{C}^2$) with a non-singular Hessian and that, as a consequence, the spectrum of its Hessian is uniformly bounded above and below respectively by $\mu_{\max}$ and $\mu_{\min}$ on some fixed compact set. Lemmas 4 and 5 assume that Algorithm 1 is used without pruning the active set A (i.e., once a point is inserted in A it stays in A). For a function F, we denote by $\partial F(x)$ the set of subgradients of the function at x and $\partial_j F(x)$ the set of subgradients in the jth subspace.

Lemma 2 *Let T be any convex function, and $G(x) = \lambda T(x) + J(x)$. Then let $g \in \partial G(x)$ be a subgradient of G at x and x^* the unique minimum of G, then*

$$\|x^* - x\| \le 2\frac{\|g\|}{\mu_{\min}}.$$

Proof This is an extension of a standard result in optimization (Boyd and Vandenberghe 2004, pp. 459–460). Combining a Taylor expansion of J with a convexity inequality for the norm we get that there exists ξ such that

$$J(x^*) \ge J(x) + \nabla J(x)(x^* - x) + \frac{1}{2}(x^* - x)^\top H(\xi)(x^* - x)$$

$$T(x^*) \ge T(x) + t^\top (x^* - x) \quad \text{with } t \in \partial T(x).$$

Thus, with $g = \lambda t + \nabla J(x)$, there exists ξ such that

$$\exists \xi, \quad G(x^*) \ge G(x) + g^\top (x^* - x) + \frac{1}{2}(x^* - x)^\top H(\xi)(x^* - x),$$

$$0 \ge G(x^*) - G(x) \ge g^\top (x^* - x) + \frac{1}{2}\mu_{\min}\|x^* - x\|^2,$$

$$\frac{1}{2}\mu_{\min}\|x^* - x\|^2 \le \|g\|\|x^* - x\|,$$

which yields the desired result. □

Lemma 3 *Let ξ_0 in (4) satisfy $\xi_0 \ge \frac{1}{2}\epsilon\mu_{\max}$. Then for all t such that $\lambda^{t+1} < \lambda^t$ the approximate subgradient conditions hold just before the gradient step at iteration t; as a consequence $\|W^t - W(\lambda^t)\| \le \sqrt{p}\frac{2\xi_0}{\mu_{\min}}$ and $J(W^t) - J(W(\lambda^t)) \le p\frac{2\xi_0^2}{\mu_{\min}}$ where $W(\lambda^t)$ is the optimal solution of (2) for the regularization coefficient λ^t.*

Proof The approximate subgradient conditions (4) are explicitly enforced by the algorithm in the active set. Using the fact that we performed a descent step on the steepest partial gradient we have:

$$J(W^{t+1}) - J(W^t) = -\epsilon\|\nabla_{w_{j^*}} J(W^t)\| + \frac{1}{2}\epsilon^2 u^{t\top}\nabla^2 J(\widetilde{W}^t)u^t, \tag{7}$$

with $u^t = \frac{\nabla_{w_{j^*}} J(W^t)}{\|\nabla_{w_{j^*}} J(W^t)\|}$ and $\widetilde{W}^t$ on the segment joining W^t and W^{t+1}. Now if $\lambda^{t+1} < \lambda^t$, then given the update rule, it has to be the case that $\frac{1}{\epsilon}(J(W^t) - J(W^{t+1})) < \lambda^t$. As a consequence, and using (7), we have that $\forall j \notin A,\ w_j = 0$ and

$$\|\nabla_{w_j} J(W^t)\| \le \|\nabla_{w_{j^*}} J(W^t)\| \le \frac{1}{\epsilon}(J(W^t) - J(W^{t+1})) + \frac{1}{2}\epsilon\mu_{\max} \le \lambda^t + \xi_0.$$

This shows the first part of the lemma. As we argue now, these approximate subgradient conditions imply that there exists a subgradient of our regularized objective of size at most $\sqrt{p}\xi_0$, which by Lemma 2 implies the result. Indeed for every covariate j such that $w_j \neq 0$, given the form of the approximate subgradient conditions (4) that we maintain, we have $\|\nabla_{w_j} J(W)\| \le (\lambda - \xi) + \xi = \lambda$; then for every covariate such that $w_j = 0$, since the subgradient set of $\lambda\| \cdot \|_2$ at 0 is the Euclidean ball of radius λ, given that $\|\nabla_{w_j} J(W)\| \le \lambda + \xi_0$, one can choose a subgradient of the ℓ_2 norm such that the corresponding partial subgradient of the regularized objective with respect to w_j is of norm less than ξ_0. Since the subgradient of the norms can be chosen independently in each subspace, we have a subgradient $g = (g_1, \ldots, g_p)$ such that $\max_j \|g_j\| \le \xi_0$ and therefore $\|g\| \le \sqrt{p}\xi_0$. Finally, the inequality in the proposition for the gap in empirical risk J results from the convexity inequality $J(W^t) - J(W(\lambda^t)) \le -g^\top(W^t - W(\lambda^t)) \le \|g\|\|W^t - W(\lambda^t)\| \le p\frac{2\xi_0^2}{\mu_{\min}}$. □

Lemma 4 *If we use steps of fixed size ϵ, after a finite number of steps λ^t becomes smaller than $\frac{1}{2}\epsilon\mu_{\min}$.*

Proof Except for a number of iterations bounded by p, at the beginning of each iteration of the algorithm, we have $\|\nabla_{w_{j_t^*}} J(W^t)\| \le \lambda^t$. Indeed, any active covariate j satisfies $\|\nabla_{w_j} J(W^t)\| \le \lambda^t$ after the approximate subgradient conditions are enforced at the end of the previous iteration, and if some inactive covariate has a gradient larger than λ^t then the largest gets incorporated in the active set, which can only happen once for every covariate if there is no pruning. For all steps t such that $\|\nabla_{w_{j_t^*}} J(W^t)\| \le \lambda^t$, if the step taken is $\epsilon\, u^t$ with u^t a unit vector in subspace j, then, using again (7) with a lower bound on the Hessian term, the update of the regularization satisfies

$$\begin{aligned}\lambda^{t+1} &= \frac{J(W^t) - J(W^{t+1})}{\epsilon}\\ &\le \|\nabla_{w_{j_t^*}} J(W^t)\| - \frac{\epsilon}{2}\mu_{\min}\\ &\le \lambda^t - \frac{\epsilon}{2}\mu_{\min}.\end{aligned}$$

So if steps of fixed size ϵ are used, then, after a finite number of steps λ^t becomes smaller than $\frac{1}{2}\epsilon\mu_{\min}$. □

Lemma 5 *If, given the direction $u^t = \frac{\nabla_{w_{j_t^*}} J(W^t)}{\|\nabla_{w_{j_t^*}} J(W^t)\|}$, we choose a step size $\epsilon_t \le \epsilon$ which maximizes the decrease $J(W^t) - J(W^{t+1})$, then $\lim_t \lambda^t \le 2\xi$.*

Proof The beginning of the previous argument is still valid and so there exists t_0 such that $\forall t > t_0$, $\lambda^{t+1} \le \lambda^t - \frac{1}{2}\epsilon_t\mu_{\min}$. So ϵ_t converges to 0. In particular, there exists t_1 such that $\forall t > t_1$, $\epsilon_t < \epsilon$. But if $\epsilon_t < \epsilon$, using a Taylor expansion at W^{t+1},

$$\begin{aligned}J(W^t) - J(W^{t+1}) &\le \epsilon_t \nabla_{w_{j_t^*}} J(W^{t+1}) \cdot u^t + \frac{1}{2}\epsilon_t^2\mu_{\max}\\ &= \frac{1}{2}\epsilon_t^2\mu_{\max},\end{aligned} \tag{8}$$

the last equality being due to the fact that the minimizer is in the interior of $[0, \epsilon]$. Using Taylor expansion (7) we have the inequality $J(W^t) - J(W^{t+1}) \ge \epsilon_t\|\nabla_{w_{j_t^*}} J(W^t)\| - \frac{\epsilon_t^2}{2}\mu_{\max}$. Given that we maintain the approximate subgradient conditions (4) the inequality $\lambda^t - 2\xi \le \|\nabla_{w_{j_t^*}} J(W^t)\|$ holds and, combining these two inequalities with Taylor expansion at W^{t+1} above, we finally get $\lambda^t - 2\xi \le \|\nabla_{w_{j_t^*}} J(W^t)\| \le \epsilon_t\mu_{\max} \xrightarrow[t]{} 0$. □

Appendix B: A stricter algorithm

The following algorithm maintains the constraints in (4) for decreasing values of λ with $\xi_0 = 0$, updating the regularization coefficient only if none of the inactive covariates violates the approximate subgradient conditions at the end of the previous iteration.

Algorithm 2 Maintain approximate subgradient conditions

```
while $\lambda^t > \lambda_{\min}$ do
    Set $j^* = \operatorname{argmax}_j \|\nabla_{w_j} J(W^t)\|$
    Update $w_{j^*}^{(t+1)} = w_{j^*}^{(t)} - \epsilon u^t$ with $u^t = \frac{\nabla_{w_{j^*}} J}{\|\nabla_{w_{j^*}} J\|}$
    if $\|\nabla_{w_{j^*}} J(W^t)\| > \lambda^t$ then
        $\lambda^{t+1} = \lambda^t$
    else
        $\lambda^{t+1} = \min(\lambda^t, \frac{J(W^t) - J(W^{t+1})}{\epsilon})$
    end if
    Add $j^*$ to the active set
    Enforce (4) only for covariates of the active set
end while
```

The correctness of the algorithm results from the fact that the regularization coefficient is unchanged when the subgradient conditions of (4) are not enforced and the fact that the algorithm terminates. Up to minor changes, Lemmas 4 and 5 in Appendix A that prove the termination of Algorithm 1 also apply to Algorithm 2.

Appendix C: Random projections, ℓ_1/ℓ_2 norm and trace norm

The essential connection between the trace norm and the ℓ_1/ℓ_2 norm is that the trace norm is the minimal ℓ_1/ℓ_2 norm over all possible orthonormal bases (cf. Argyriou et al. 2008): for $X \in \mathbb{R}^{p \times K}$,

$$\|X\|_{\mathrm{tr}} \overset{(*)}{=} \min_{U \in \mathcal{O}^p} \|UX\|_{\ell_1/\ell_2}.$$

Combining ℓ_1/ℓ_2-regularization with random projections of the data can be viewed intuitively as replacing the optimal U in the above expression by a rectangular matrix with random unit-length columns. The relation between the two norms is easier to understand via their "quadratic over linear" formulations which we review in the next lemma.

Lemma 6 *It is a common feature of the ℓ_1, ℓ_1/ℓ_2 and trace norms that they are each related to a "quadratic over linear" formulation where the variable for the linear part σ (or Σ) is constrained to lie in some truncated cone. The following relations hold*:

$$\|y\|_1^2 = \inf_{\sigma_i > 0,\, \sum_i \sigma_i \le 1} \sum_i \frac{y_i^2}{\sigma_i}.$$

If x_i is the ith row of $X \in \mathbb{R}^{p\times K}$, then

$$\|X\|_{\ell_1/\ell_2}^2 = \Big(\sum_i \|x_i\|_2\Big)^2 = \inf_{\sigma_i>0,\, \sum_i \sigma_i \le 1} \sum_i \frac{\|x_i\|_2^2}{\sigma_i}$$
$$= \inf_{\Sigma = \mathrm{diag}(\sigma),\, \sigma_i > 0,\, \mathrm{tr}(\Sigma)\le 1} \mathrm{tr}(X\Sigma^{-1}X^\top).$$

If $(\lambda_i)_{1\le i\le p}$ is the set of singular values of X and $\Lambda = \mathrm{diag}(\lambda)$ then

$$\|X\|_{\mathrm{tr}}^2 = \|\lambda\|_1^2 = \|\Lambda\|_{\ell_1/\ell_2}^2 \overset{(*)}{=} \min_{U\in\mathcal{O}^p} \|UX\|_{\ell_1/\ell_2}^2$$
$$= \inf_{U\in\mathcal{O}^p,\, \Sigma=\mathrm{diag}(\sigma),\, \sigma_i>0,\, \mathrm{tr}(\Sigma)\le 1} \mathrm{tr}(X^\top U^\top \Sigma^{-1} U X)$$
$$= \inf_{D\succ 0,\, \mathrm{tr}(D)\le 1} \mathrm{tr}(X^\top D^{-1} X),$$

where $\mathcal{O}^p \subset \mathbb{R}^{p\times p}$ is the set of orthonormal matrices.

Proof Except for $(*)$ which is proven by Argyriou et al. (2008) all identities stem from the identity for the ℓ_1 norm which can be verified by straightforward minimization. □

To formulate optimization problems that involve the above-mentioned norms, it is convenient to replace all the infima by minima (i.e., the infima are attained). This is possible if the constraint set is closed on the part of the boundary of the set where the objective function does not diverge, and if all inverses are extended by continuity by their Moore-Penrose pseudoinverses. The appropriate partial closure can be obtained replacing $\sigma > 0$ (resp. $D \succ 0$) by $\sigma \ge 0$ (resp. $D \succeq 0$), and imposing $(\sigma_i = 0) \Rightarrow (y_i = 0)$ (resp. $\mathcal{I}m(X) \subseteq \mathcal{I}m(D)$) where $\mathcal{I}m(X)$ is the range of X. The set $\{(X, D) | \mathcal{I}m(X) \subseteq \mathcal{I}m(D), D \succ 0\}$ is a convex set as we argue in Lemma 7.

Lemma 7 *The set $\mathcal{X} = \{(X, D) \mid \mathcal{I}m(X) \subseteq \mathcal{I}m(D), D \succ 0\}$ is convex.*

Proof The set is obviously stable under multiplication by a scalar. Moreover if $(X_1, D_1) \in \mathcal{X}$ and $(X_2, D_2) \in \mathcal{X}$, then $\mathcal{I}m(X_1 + X_2) \subseteq \mathcal{I}m(X_1) + \mathcal{I}m(X_2) \subseteq \mathcal{I}m(D_1) + \mathcal{I}m(D_2)$, where the sum of two vector spaces denotes their span. The convexity of $\mathcal{X}$ is therefore proved if we show that, for p.s.d. matrices $\mathcal{I}m(D_1) + \mathcal{I}m(D_2) = \mathcal{I}m(D_1 + D_2)$. Indeed, for p.s.d. matrices D_1 and D_2, $\mathcal{I}m(D_1 + D_2)^\perp = \mathcal{K}er(D_1 + D_2)$, which is clear if the matrix $D_1 + D_2$ is considered in its orthonormal basis of eigenvectors. Then $\mathcal{K}er(D_1 + D_2) = \mathcal{K}er(D_1) \cap \mathcal{K}er(D_2)$, because $x^\top(D_1 + D_2)x = 0 \Leftrightarrow (x^\top D_1 x = 0 \,\&\, x^\top D_2 x = 0)$. Finally, $\mathcal{K}er(D_1) \cap \mathcal{K}er(D_2) \subseteq (\mathcal{I}m(D_1) + \mathcal{I}m(D_2))^\perp$. This yields $\mathcal{I}m(D_1) + \mathcal{I}m(D_2) \subseteq \mathcal{I}m(D_1 + D_2)$ and since the other inclusion holds trivially, this proves the result. □

Using the above, we have the following corollary to Lemma 6:

Corollary 1 *For a matrix $A \in \mathbb{R}^{p\times K}$ define J as $J(A) = \sum_{k=1}^K \sum_{i=1}^{N_k} J^k(a^k \cdot x_i^k, y_i^k)$. The two following optimization problems are equivalent*:

$$\min_A \quad \|A\|_{\mathrm{tr}}^2 + \frac{1}{\lambda} J(A) \tag{9a}$$

$$\begin{aligned} \min_{A,D} \quad & A^\top D^+ A + \frac{1}{\lambda} J(A) \\ \text{s.t.} \quad & D \succeq 0,\ \mathrm{tr}(D) \le 1 \\ & \mathcal{I}m(A) \subseteq \mathcal{I}m(D) \end{aligned} \tag{9b}$$

where D^+ is the Moore-Penrose pseudoinverse of D and $\mathcal{I}m(D)$ is the range of D.

The following two lemmas prove Proposition 2:

Lemma 8 *We consider a general learning problem with a loss function $J(A) = \sum_{k=1}^K \sum_{i=1}^{N_k} J^k(a^k \cdot x_i^k, y_i^k)$ depending on products of the parameter matrix $A \in \mathbb{R}^{p\times K}$ with K task-specific data matrices $X_1, \ldots, X_K$ where $X_k \in \mathbb{R}^{N_k \times p}$. Let $\Phi \in \mathbb{R}^{p\times d}$ be a random projection matrix whose columns are uniformly drawn from the unit sphere $\mathcal{S}^p$ in $\mathbb{R}^p$ and let $W \in \mathbb{R}^{d\times K}$ be another parameter matrix. The two following optimization problems are equivalent*:

$$\begin{aligned} \min_A \quad & \|W\|_{\ell_1/\ell_2}^2 + \frac{1}{\lambda} J(A) \\ \text{s.t.} \quad & A = \Phi W \end{aligned} \tag{10a}$$

$$\begin{aligned} \min_{A,D,\Sigma} \quad & A^\top D^+ A + \frac{1}{\lambda} J(A) \\ \text{s.t.} \quad & D \succeq 0,\ \mathrm{tr}(D) \le 1 \\ & \mathcal{I}m(A) \subseteq \mathcal{I}m(D) \\ & D = \Phi\Sigma\Phi^\top, \qquad \Sigma = \mathrm{diag}(\sigma), \\ & \sigma \in \mathbb{R}_+^d, \qquad \mathbf{1}^\top \sigma \le 1. \end{aligned} \tag{10b}$$

Proof We denote by $\Phi^+ = \Phi^\top(\Phi\Phi^\top)^+$ the Moore-Penrose pseudoinverse of Φ. If $\Phi W = A$ we can rewrite $W = \Phi^+ A + H$ with $H \in \mathbb{R}^{d\times K}$ such that $\Phi H = 0$. We consider first

$$\begin{aligned} \min_{H} \quad & \|\Phi^{+}A + H\|_{\ell_1/\ell_2} \\ \text{s.t.} \quad & \Phi H = 0 \end{aligned}$$

or equivalently

$$\begin{aligned} \min_{\sigma,H}\max_{\Lambda} \quad & \operatorname{tr}((\Phi^{+}A + H)^{\top}\Sigma^{+}(\Phi^{+}A + H)) + \operatorname{tr}(\Lambda^{\top}\Phi H) \\ \text{s.t.} \quad & \Sigma = \operatorname{diag}(\sigma), \\ & \sigma \in \mathbb{R}^{d}_{+}, \qquad \mathbf{1}^{\top}\sigma \leq 1, \\ & \mathcal{I}m(\Phi^{+}A + H) \subseteq \mathcal{I}m(\Sigma). \end{aligned} \tag{11}$$

For any fixed A and σ the problem is convex in H and strictly feasible so we can minimize with respect to H before maximizing in Λ. Setting H as follows: $H^{*} = -\Phi^{+}A - \Sigma\Phi^{\top}\Lambda$, the range inclusion constraint is satisfied and the partial gradient of the objective with respect to H is equal to zero. We solve for the Lagrange multipliers Λ^{*} by enforcing the equality constraints: $\Phi H^{*} = 0 = -\Phi\Phi^{+}A - \Phi\Sigma\Phi^{\top}\Lambda^{*}$ which yields $H^{*} = -\Phi^{+}A + \Sigma\Phi^{\top}(\Phi\Sigma\Phi^{\top})^{+}\Phi\Phi^{+}A$. But then, using the identities $BB^{+}B = B$ and $B^{+}BB^{+} = B^{+}$ for the pseudoinverse,

$$\begin{aligned} &(\Phi^{+}A + H^{*})^{\top}\Sigma^{+}(\Phi^{+}A + H^{*}) \\ &\quad = A^{\top}\Phi\Phi^{+}(\Phi\Sigma\Phi^{\top})^{+}\Phi\Sigma\Sigma^{+}\Sigma\Phi^{\top}(\Phi\Sigma\Phi^{\top})^{+}\Phi\Phi^{+}A \\ &\quad = A^{\top}\Phi\Phi^{+}(\Phi\Sigma\Phi^{\top})^{+}\Phi\Phi^{+}A. \end{aligned}$$

We can finally transform (11) into

$$\begin{aligned} \min_{W,A,H,\Sigma} \quad & W^{\top}\Sigma^{+}W + \frac{1}{\lambda}J(A) \\ \text{s.t.} \quad & W = \Phi^{+}A + H,\ \Phi H = 0, \\ & \mathcal{I}m(A) \subseteq \mathcal{I}m(\Phi W), \\ & \mathcal{I}m(W) \subseteq \mathcal{I}m(\Sigma),\ \Sigma = \operatorname{diag}(\sigma), \\ & \sigma \in \mathbb{R}^{d}_{+},\ \mathbf{1}^{\top}\sigma \leq 1. \end{aligned}$$

Then eliminate W and H from the previous equations to get:

$$\begin{aligned} \min_{A,\Sigma} \quad & \operatorname{tr}(A^{\top}\Phi\Phi^{+}(\Phi\Sigma\Phi^{\top})^{+}\Phi\Phi^{+}A) + \frac{1}{\lambda}J(A) \\ \text{s.t.} \quad & \mathcal{I}m(A) \subseteq \mathcal{I}m(\Phi\Sigma),\ \Sigma = \operatorname{diag}(\sigma), \\ & \sigma \in \mathbb{R}^{d}_{+}, \qquad \mathbf{1}^{\top}\sigma \leq 1. \end{aligned}$$

If we then assume that $d \geq p$, then $\Phi\Phi^{+}$ is almost surely the identity matrix, because Φ is almost surely of full column rank and therefore so is $\Phi\Phi^{+}$. Letting $D = \Phi\Sigma\Phi^{\top}$, D is positive semi-definite since Σ is; moreover $\operatorname{tr}(\Phi\Sigma\Phi^{\top}) = \sum_{i=1}^{d}\sigma_i\|\phi^{i}\|^{2}$ where ϕ^{i} is the ith column of Φ but by assumption $\|\phi^{i}\| = 1$ so that $\operatorname{tr}(D) = \operatorname{tr}(\Sigma)$. Taking into account these identities, we obtain the equivalence to (10b). □

Lemma 9 *If J is convex, continuous and lower bounded, then as the number of random projections d increases, the solutions $A^{*}_{d} = \Phi_d W^{*}_{d}$ obtained from (10b) form a sequence whose accumulation points are almost surely optimal solutions for (9a).*

Proof For problem (9b), denote by $G(D, A)$ its objective function, Ω its constraint set, and (D^{*}, A^{*}) an optimal solution. Problem (10b) has the same objective function, constraint set Ω_d and we denote an optimal solution by (D^{*}_{d}, A^{*}_{d}). We first show that as $d \to \infty$, with high probability, there exists a full rank matrix D_d such that $(D_d, A^{*}) \in \Omega_d$ and D_d is close to D^{*} in Frobenius norm.

Given Φ as in (10b), for any $D \succeq 0$, $\operatorname{tr}(D) \leq 1$, we can approximate D by a matrix of the form $\Phi\Sigma\Phi^{\top}$ with Σ a diagonal matrix such that $\operatorname{tr}(\Sigma) \leq 1$ as follows: write $D = V\widetilde{\Sigma}V^{\top}$, where $\widetilde{\Sigma}$ is diagonal and V a matrix of eigenvectors of D and approximate it with $\widetilde{\Phi}\widetilde{\Sigma}\widetilde{\Phi}^{\top}$ where $\widetilde{\Phi}$ is the matrix formed of p distinct columns of Φ where each approximation is the best to a column of V in the sense that $\|V - \widetilde{\Phi}\|_F$ is small. Then, since $\operatorname{tr}(\widetilde{\Sigma}) \leq 1$, $\widetilde{\Phi}\widetilde{\Sigma}\widetilde{\Phi}^{\top}$ can be rewritten as $\Phi\Sigma\Phi^{\top}$ for some Σ with $\operatorname{tr}(\Sigma) \leq 1$, and we have

$$\begin{aligned} &\|D - \widetilde{\Phi}\widetilde{\Sigma}\widetilde{\Phi}^{\top}\|_F \\ &\quad = \|V\widetilde{\Sigma}^{\frac{1}{2}}(\widetilde{\Sigma}^{\frac{1}{2}}V^{\top} - \widetilde{\Sigma}^{\frac{1}{2}}\widetilde{\Phi}^{\top}) \\ &\qquad + (V\widetilde{\Sigma}^{\frac{1}{2}} - \widetilde{\Phi}\widetilde{\Sigma}^{\frac{1}{2}})\widetilde{\Sigma}^{\frac{1}{2}}\widetilde{\Phi}^{\top}\|_F \\ &\quad \leq (\|D^{\frac{1}{2}}\|_F + \|\widetilde{\Phi}\widetilde{\Sigma}^{\frac{1}{2}}\|_F)\|(V - \widetilde{\Phi})\widetilde{\Sigma}^{\frac{1}{2}}\|_F \\ &\quad \leq (\operatorname{tr}(D) + \operatorname{tr}(\widetilde{\Phi}\widetilde{\Sigma}\widetilde{\Phi}^{\top}))\|V - \widetilde{\Phi}\|_F \operatorname{tr}(\widetilde{\Sigma}) \\ &\quad \leq 2\|V - \widetilde{\Phi}\|_F, \end{aligned}$$

where we used first that the Frobenius norm satisfies the inequality $\|AB\|_F \leq \|A\|_F\|B\|_F$, next, the fact that for a p.s.d. matrix $\|A^{\frac{1}{2}}\|_F = \operatorname{tr}(A)$, further that $\operatorname{tr}(\widetilde{\Phi}\widetilde{\Sigma}\widetilde{\Phi}^{\top}) = \operatorname{tr}(\widetilde{\Sigma})$ since $\widetilde{\Phi}$ has unit norm columns, and finally that the traces of D and $\widetilde{\Sigma}$ are smaller or equal to 1.

To approximate D^{*} with a full-rank matrix, note first that it can be approximated arbitrarily closely by a full rank matrix D' in the p.s.d. cone and the latter can be approximated by $D_d = \widetilde{\Phi}\widetilde{\Sigma}\widetilde{\Phi}^{\top}$. For a full rank matrix D_d, we have trivially that $\mathcal{I}m(A^{*}) \subseteq \mathcal{I}m(D_d)$ and therefore we have $(D_d, A^{*}) \in \Omega_d$.

By the previous result, as $d \to \infty$, with high probability there exists $(D_d, A^{*}) \in \Omega_d$, such that $\|D^{*} - D_d\|_F \leq \epsilon$. But then, by continuity of J and the trace norm, for all $\eta > 0$, there exists ϵ such that, if $\|D^{*} - D_d\|_F \leq \epsilon$, then $G(D_d, A^{*}) \leq G(D^{*}, A^{*}) + \eta$. As a consequence, with high probability, if (D^{*}_{d}, A^{*}_{d}) is an optimal solution of (10b), we have a fortiori $G(D^{*}_{d}, A^{*}_{d}) \leq G(D^{*}, A^{*}) + \eta$. This proves that $G(D^{*}_{d}, A^{*}_{d})$ converges in probability to $G(D^{*}, A^{*})$ as $d \to \infty$. Denoting by $\widetilde{G}$ the objective function of (9a), we have that $\widetilde{G}(A_d^*)$ converges in probability to $\widetilde{G}(A^*)$. However, since for all ω, the sequence $\widetilde{G}(A_d^*(\omega))$ is monotonically decreasing, the convergence to $\widetilde{G}(A^*)$ is in fact almost sure. But since J is lower bounded and the trace norm is coercive, so is $\widetilde{G}$ and its sublevel sets are thus compact; as a consequence (A_d^*) is deterministically bounded and, almost surely, all converging subsequences of (A_d^*) converge to a minimum of $\widetilde{G}$. □

The construction in this lemma, although sufficient to prove the almost sure convergence, seems too pessimistic to obtain a reasonable idea of the rate of convergence. Indeed it is a quite strong requirement to ask that each of the eigenvectors of D be approximated by an individual column of Φ and D could possibly be well approximated without requiring that this property holds.

References

Abernethy, J., Bach, F., Evgeniou, T., Vert, J.-P.: (2008). A new approach to collaborative filtering: Operator estimation with spectral regularization. Technical report, Computer Science Division, University of California at Berkeley

Ando, R., Zhang, T.: A framework for learning predictive structures from multiple tasks and unlabeled data. J. Mach. Learn. Res. **6**, 1817–1853 (2005)

Argyriou, A., Evgeniou, T., Pontil, M.: Convex multi-task feature learning. Mach. Learn. (2008)

Bach, F.: Consistency of trace norm minimization. J. Mach. Learn. Res. **9**, 1019–1048 (2008)

Bach, F., Lanckriet, G., Jordan, M.I.: Multiple kernel learning, conic duality, and the SMO algorithm. In: Proceedings of the Twenty-first International Conference on Machine Learning. Morgan Kaufmann Publishers, San Francisco (2004)

Ben-David, S., Schuller-Borbely, R.: A notion of task relatedness yielding provable multiple-task learning guarantees. Mach. Learn. **73**, 273–287 (2008)

Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)

Chiaromonte, F., Cook, R.D.: Sufficient dimension reduction and graphics in regression. Ann. Inst. Stat. Math. **54**(4), 768–795 (2002)

Donoho, D.: For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution. Technical Report 2004–10, Statistics Department, Stanford University (2004)

Draper, N.R., Smith, H.: Applied Regression Analysis. Wiley–Interscience, New York (1998)

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

Evgeniou, T., Pontil, M.: Regularized multi-task learning. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 109–117 (2004)

Fazel, M., Hindi, H., Boyd, S.: A rank minimization heuristic with application to minimum order system approximation. In: Proceedings of the American Control Conference, vol. 6, pp. 4734–4739 (2001)

Fazel, M., Hindi, H., Boyd, S.: Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. In: Proceedings of the American Control Conference, vol. 3, pp. 2156–2162 (2003)

Fu, W., Knight, K.: Asymptotics for lasso-type estimators. Ann. Stat. **28**, 1356–1378 (2000)

Fukumizu, K., Bach, F.R., Jordan, M.I.: Kernel dimension reduction in regression. Ann. Stat. (2008, to appear)

Hastie, T., Tibshirani, R., Friedman, J.: Elements of Statistical Learning. Springer, Berlin (2001)

Jebara, T.: Multi-task feature and kernel selection for SVMs. In: Proceedings of the International Conference on Machine Learning. Morgan Kaufmann, San Francisco (2004)

Khan, J., Wei, J., Ringnér, M., et al.: Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat. Med. **7**, 673–679 (2001)

Kim, Y., Kim, J., Kim, Y.: Blockwise sparse regression. Stat. Sin. **16**(2), 375–390 (2006)

Li, K.C.: Sliced inverse linear regression for dimension reduction. J. Am. Stat. Assoc. **86**, 316–342 (1991)

Maurer, A.: Bounds for linear multi-task learning. J. Mach. Learn. Res. **7**, 117–139 (2006)

Meier, L., van de Geer, S., Bühlmann, P.: The group lasso for logistic regression. J. R. Stat. Soc. Ser. B **70**(1), 53–71 (2008)

Osborne, M.R., Presnell, B., Turlach, B.A.: A new approach to variable selection in least squares problems. IMA J. Numer. Anal. **20**(3), 389–403 (2000)

Park, M.Y., Hastie, T.: Regularization path algorithms for detecting gene interactions. Technical Report 2006-13, Department of Statistics, Stanford University (2006)

Recht, B., Xu, W., Hassibi, B.: Necessary and sufficient conditions for success of the nuclear norm heuristic for rank minimization. Technical report, California Institute of Technology (2008)

Rosset, S., Zhu, J.: Piecewise linear regularized solution paths. Ann. Stat. **35**(3), 1012–1030 (2007)

Srebro, N., Shraibman, A.: Rank, Trace-Norm and Max-Norm, vol. 3559, pp. 545–560. Springer, New York (2005).

Srebro, N., Alon, N., Jaakkola, T.: Generalization error bounds for collaborative prediction with low-rank matrices. In: Advances in Neural Information Processing Systems. MIT Press, Cambridge (2005a)

Srebro, N., Rennie, J., Jaakkola, T.S.: Maximum-margin matrix factorization. In: Advances in Neural Information Processing. MIT Press, Cambridge (2005b)

Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. B **58**(1), 267–288 (1996)

Torralba, A., Murphy, K.P., Freeman, W.T.: Sharing features: efficient boosting procedures for multiclass object detection. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), pp. 762–769. IEEE Computer Society, Washington (2004)

Tseng, P., Yun, S.: A coordinate gradient descent method for linearly constrained smooth optimization and support vector machines training. Comput. Optim. Appl. (2008, to appear)

van Breukelen, M., Duin, R.P.W., Tax, D.M.J., den Hartog, J.E.: Handwritten digit recognition by combined classifiers. Kybernetika **34**(4), 381–386 (1998)

Wu, B.: Differential gene expression detection and sample classification using penalized linear regression models. Bioinformatics **22**(5), 472–476 (2005)

Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. J. R. Stat. Soc. B **1**(68), 49–67 (2006)

Zhao, P., Yu, B.: Stagewise lasso. J. Mach. Learn. Res. **8**, 2701–2726 (2007)

Zhao, P., Rocha, G., Yu, B.: Grouped and hierarchical model selection through composite absolute penalties. Ann. Stat. (2008, to appear)