The Group-Lasso for Generalized Linear Models: Uniqueness of Solutions and Efficient Algorithms

Volker Roth VOLKER.ROTH@UNIBAS.CH
Computer Science Department, University of Basel, Bernoullistr. 16, CH-4056 Basel, Switzerland

Bernd Fischer BERND.FISCHER@INF.ETHZ.CH
Institute of Computational Science, ETH Zurich, Universitaetstrasse 6, CH-8092 Zurich, Switzerland

Abstract

The Group-Lasso method for finding important explanatory factors suffers from the potential non-uniqueness of solutions and also from high computational costs. We formulate conditions for the uniqueness of Group-Lasso solutions which lead to an easily implementable test procedure that allows us to identify all potentially active groups. These results are used to derive an efficient algorithm that can deal with input dimensions in the millions and can approximate the solution path efficiently. The derived methods are applied to large-scale learning problems where they exhibit excellent performance and where the testing procedure helps to avoid misinterpretations of the solutions.

1. Introduction

In many practical learning problems we are not only interested in low prediction errors but also in identifying important explanatory factors. These explanatory factors can often be represented as groups of input variables. Common examples are k-th order polynomial expansions of the inputs where the groups consist of products over combinations of variables up to degree k. Such expansions compute explicit mappings into feature spaces induced by polynomial kernel functions of the form $k(\boldsymbol{x}, \boldsymbol{y}) = (1 + \boldsymbol{x} \cdot \boldsymbol{y})^k$. Another popular example are categorical variables that are represented as groups of dummy variables.

A method for variable selection which has gained particular attention is the Lasso (Tibshirani, 1996) which exploits the idea of using ℓ_1-constraints in fitting problems. The Group-Lasso (Yuan & Lin, 2006) extends the former in the sense that it finds solutions that are sparse on the level of *groups* of variables, which makes this method a good candidate for situations described above. The Group-Lasso estimator, however, has several drawbacks: (i) in high-dimensional spaces, the solutions may not be unique. The potential existence of several solutions that involve different variables seriously hampers the interpretability of "identified" explanatory factors; (ii) existing algorithms can handle input dimensions up to thousands (Kim et al., 2006) or even several thousands (Meier et al., 2008), but in practical applications with high-order interactions or polynomial expansions these limits are easily exceeded; (iii) contrary to the standard Lasso, the solution path (i.e. the evolution of the individual group norms as a function of the constraint) is not piecewise linear, which precludes the application of efficient optimization methods like *least angle regression* (LARS) (Efron et al., 2004).

In this paper we address all these issues: (i) we derive conditions for the *completeness* and *uniqueness* of Group-Lasso estimates, where we call a solution *complete*, if it includes all groups that might be relevant in other solutions (meaning that we cannot have "overlooked" relevant groups). Based on these conditions we develop an easily implementable *test procedure*. If a solution is not complete, this procedure *identifies all other groups* that may be included in alternative solutions with identical costs. (ii) These results allow us to formulate a *highly efficient active-set algorithm* that can deal with input dimensions in the millions. (iii) The *solution path* can be approximated on a fixed grid of constraint values with almost no additional computational costs. Large-scale applications using both synthetic and real data illustrate the excellent performance of the developed concepts and algorithms. In particular, we demonstrate that the proposed completeness test successfully detects ambiguous solutions and thus avoids the misinterpretation of "identified" explanatory factors.

Appearing in *Proceedings of the 25^{th} International Conference on Machine Learning*, Helsinki, Finland, 2008.

2. Characterization of Group-Lasso Solutions for Generalized Linear Models

This section largely follows (Osborne et al., 2000), with the exception that here we address the *Group*-Lasso problem and a more general class of likelihood functions.

According to (McCullaghand & Nelder, 1983), a generalized linear model (GLM) consists of three elements:
(i) a random component $f(y;\mu)$ specifying the stochastic behavior of a response variable Y;
(ii) a systematic component $\eta = \boldsymbol{x}^\top\boldsymbol{\beta}$ specifying the variation in the response variable accounted for by known covariates $\boldsymbol{x}$; and
(iii) a link function $g(\mu) = \eta$ specifying the relationship between the random and systematic components.
The random component $f(y;\mu)$ is typically an exponential family distribution

$$f(y;\theta,\phi) = \exp(\phi^{-1}(y\theta - b(\theta)) + c(y,\phi)), \qquad (1)$$

with natural parameter θ, sufficient statistics y/ϕ, log partition function $b(\theta)/\phi$ and a scale parameter $\phi > 0$.

Note that in the model (1) the mean of the responses $\mu = E_\theta[y]$ is related to the natural parameter θ by $\mu = b'(\theta)$. The link function g can be any strictly monotone differentiable function. In the following, however, we will consider only *canonical* link functions for which $g(\mu) = \eta = \theta$. We will thus use the parametrization $f(y;\eta,\phi)$.

From a technical perspective, an important property of this framework is that $\log f(y;\eta,\phi)$ is strictly concave in η. This follows from the fact that the one-dimensional sufficient statistics y/ϕ is necessarily *minimal*, which implies that the log partition function $b(\eta)/\phi$ is strictly convex, see (Brown, 1986; Wainwright et al., 2005).

The standard linear regression model is a special case derived from the normal distribution with $\phi = \sigma^2$, the identity link $\eta = \mu$ and $b(\eta) = (1/2)\eta^2$. Other popular models include logistic regression (binomial distribution), Poisson regression for count data and gamma- (or exponential-, Weibull-) models for cost- or survival analysis.

Given an i.i.d. data sample $\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n\}$, $\boldsymbol{x}_i \in \mathbb{R}^d$, arranged as rows of the data matrix X, and a corresponding vector of responses $\boldsymbol{y} = (y_1,\ldots,y_n)^\top$, we will consider the problem of minimizing the negative log-likelihood

$$\begin{aligned} l(\boldsymbol{y},\boldsymbol{\eta},\boldsymbol{\phi}) &= -\sum_i \log f(y_i;\eta_i,\phi) \\ &= -\sum_i \phi^{-1}(y_i\eta_i - b(\eta_i)) + c(y_i,\phi). \end{aligned} \qquad (2)$$

We assume that the scale parameter is known, and for the sake of simplicity we assume $\phi = 1$. Since $\eta = x^T\beta$, the gradient of l can be viewed as a function in either $\boldsymbol{\eta}$ or $\boldsymbol{\beta}$:

$$\begin{aligned} &\nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta}) = -(\boldsymbol{y} - g^{-1}(\boldsymbol{\eta})), \\ &\nabla_{\boldsymbol{\beta}} l(\boldsymbol{\beta}) = -X^\top\nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta}) = -X^\top(\boldsymbol{y} - g^{-1}(X\boldsymbol{\beta})), \end{aligned} \qquad (3)$$

where $g^{-1}(\boldsymbol{\eta}) := (g^{-1}(\eta_1),\ldots,g^{-1}(\eta_n))^\top$. The corresponding Hessians are

$$H_{\boldsymbol{\eta}} = W, \quad H_{\boldsymbol{\beta}} = X^\top W X, \qquad (4)$$

where W is diagonal with elements $W_{ii} = (g^{-1})'(\eta_i) = 1/(g'(\mu_i)) = \mu'(\eta_i) = b''(\eta_i)$.

For the following derivation, it is convenient to partition X, $\boldsymbol{\beta}$ and $\boldsymbol{h} := \nabla_{\boldsymbol{\beta}} l$ into J subgroups: $X = (X_1,\ldots,X_j)$,

$$\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{\beta}_1 \\ \vdots \\ \boldsymbol{\beta}_J \end{pmatrix}, \quad \boldsymbol{h} = \begin{pmatrix} \boldsymbol{h}_1 \\ \vdots \\ \boldsymbol{h}_J \end{pmatrix} = \begin{pmatrix} X_1^\top\nabla_{\boldsymbol{\eta}} l \\ \vdots \\ X_J^\top\nabla_{\boldsymbol{\eta}} l \end{pmatrix}. \qquad (5)$$

As stated above, b is strictly convex in $\theta = \eta$, thus $b''(\eta_i) > 0$ which in turn implies that $H_{\boldsymbol{\eta}} \succ 0$ and $H_{\boldsymbol{\beta}} \succeq 0$. This means that l is a strictly convex function in $\boldsymbol{\eta}$. For general matrices X it is convex in $\boldsymbol{\beta}$, and it is strictly convex in $\boldsymbol{\beta}$ if X has full rank and $d \leq n$.

Given X and $\boldsymbol{y}$, the Group-Lasso minimizes the negative log-likelihood viewed as a function in $\boldsymbol{\beta}$ under a constraint on the sum of the ℓ_2-norms of the subvectors $\boldsymbol{\beta}_j$:

$$\text{minimize } l(\boldsymbol{\beta}) \quad \text{s.t.} \quad g(\boldsymbol{\beta}) \geq 0, \qquad (6)$$

$$\text{where} \quad g(\boldsymbol{\beta}) = \kappa - \textstyle\sum_{i=1}^J \|\boldsymbol{\beta}_j\|. \qquad (7)$$

Here $g(\boldsymbol{\beta})$ is implicitly a function of the fixed parameter κ.

Considering the *unconstrained* problem, the solution is not unique if the dimensionality exceeds n: every $\boldsymbol{\beta}^* = \boldsymbol{\beta}^0 + \boldsymbol{\xi}$ with $\boldsymbol{\xi}$ being an element of the null space $\boldsymbol{N}(X)$ is also a solution. By defining the unique value

$$\kappa_0 := \min_{\boldsymbol{\xi}\in\boldsymbol{N}(X)} \textstyle\sum_{i=1}^J \|\boldsymbol{\beta}_j^0 + \boldsymbol{\xi}_j\|, \qquad (8)$$

we will require that the constraint is active i.e. $\kappa < \kappa_0$. Note that the minimum κ_0 is unique, even though there might exist several vectors $\boldsymbol{\xi} \in \boldsymbol{N}(X)$ which attain this minimum. Enforcing the constraint to be active is essential for the following characterization of solutions. Although it might be infeasible to ensure this activeness by computing κ_0 and selecting κ accordingly, practical algorithms will not suffer from this problem: given a solution, we can always check if the constraint was active. If this was not the case, then the uniqueness question reduces to checking if $d \leq n$ (if X has full rank). In this case the solutions are usually not sparse, because the feature selection mechanism has been switched off. To produce a sparse solution,

one can then try smaller κ-values until the constraint is active. In section 3 we propose a more elegant solution to this problem in the form of an algorithm that approximates the solution path, i.e. the evolution of the group norms when relaxing the constraint. This algorithm can be initialized with an arbitrarily small constraint value κ^0 which typically ensures that the constraint is active in the first optimization step. Activeness of the constraint in the following steps can then be monitored by observing the decay of the Lagrange parameter when increasing κ, cf. Eq. (14) below.

Under the assumption $l > -\infty$ a minimum of (6) is guaranteed to exist, since l is continuous and the region of feasible vectors β is compact. The assumption $l > -\infty$ simply means that the likelihood is finite ($f < +\infty$) for all parameter values θ which is usually satisfied for models of practical importance (see (Wedderburn, 1973) for a detailed discussion), and we will restrict our further analysis to models of this kind[1]. Since we assume that the constraint is active, any solution $\widehat{\boldsymbol{\beta}}$ will lie on the boundary of the constraint region. It is easily seen that $\sum_{j=1}^{J} \|\boldsymbol{\beta}_j\|$ is convex which implies that $g(\boldsymbol{\beta})$ is concave. Thus, the region of feasible values defined by $g(\boldsymbol{\beta}) \geq 0$ is convex. If $d \leq n$, the objective function l will be strictly convex if X has full rank, which additionally implies that the minimum is unique. In summary, we can state the following theorem:

Theorem 1. *If $\kappa < \kappa_0$ and X has maximum rank, then the following holds: (i) A solution $\widehat{\boldsymbol{\beta}}$ exists and $\sum_{i=1}^{J} \|\widehat{\boldsymbol{\beta}}_j\| = \kappa$ for any such solution. (ii) If $d \leq n$, the solution is unique.*

The Lagrangian for problem (6) reads

$$\mathcal{L}(\boldsymbol{\beta}, \lambda) = l(\boldsymbol{\beta}) - \lambda g(\boldsymbol{\beta}). \tag{9}$$

For a given $\lambda > 0$, $\mathcal{L}(\boldsymbol{\beta}, \lambda)$ is a convex function in $\boldsymbol{\beta}$. Under the assumption $l > -\infty$ a minimum is guaranteed to exist, since g goes to infinity if $\|\boldsymbol{\beta}\| \to \infty$.

The vector $\widehat{\boldsymbol{\beta}}$ minimizes $\mathcal{L}(\boldsymbol{\beta}, \lambda)$ iff the d-dimensional null-vector $\mathbf{0}_d$ is an element of the subdifferential $\partial_{\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{\beta}, \lambda)$. Let d_j denote the dimension of the j-th subvector $\boldsymbol{\beta}_j$ (i.e. the size of the j-th subgroup). The subdifferential is

$$\partial_{\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{\beta}, \lambda) = \nabla_{\boldsymbol{\beta}} l(\boldsymbol{\beta}) + \lambda \boldsymbol{v} = X^\top \nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta}) + \lambda \boldsymbol{v}, \tag{10}$$

with $\boldsymbol{v} = (\boldsymbol{v}_1, \dots \boldsymbol{v}_J)^\top$ defined by

$$\begin{aligned} \boldsymbol{v}_j &= \frac{\boldsymbol{\beta}_j}{\|\boldsymbol{\beta}_j\|}, \text{ if } \boldsymbol{\beta}_j \neq \mathbf{0}_{d_j} \text{ and} \\ \boldsymbol{v}_j &\in \{\boldsymbol{a} \in \mathbb{R}^{d_j} : \|\boldsymbol{a}\| \leq 1\}, \text{ else.} \end{aligned} \tag{11}$$

Thus, $\widehat{\boldsymbol{\beta}}$ is a minimizer for λ fixed iff

$$\mathbf{0}_d = X^\top \nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta})|_{\boldsymbol{\eta}=\widehat{\boldsymbol{\eta}}} + \lambda \boldsymbol{v} \qquad (\text{with } \widehat{\boldsymbol{\eta}} = X\widehat{\boldsymbol{\beta}}), \tag{12}$$

[1]Technically we require that the domain of l is $\mathbb{R}^d$, which implies that Slater's condition holds.

for some $\boldsymbol{v}$ of the form described above. Hence, for all j with $\widehat{\boldsymbol{\beta}}_j \neq \mathbf{0}_{d_j}$ it holds that

$$\|X_j^\top \nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta})|_{\boldsymbol{\eta}=\widehat{\boldsymbol{\eta}}}\| = \lambda. \tag{13}$$

For all other j with $\widehat{\boldsymbol{\beta}}_j = \mathbf{0}_{d_j}$ it holds that $\|X_j^\top \nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta})|_{\boldsymbol{\eta}=\widehat{\boldsymbol{\eta}}}\| \leq \lambda$ which implies

$$\lambda = \max_j \|X_j^\top \nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta})|_{\boldsymbol{\eta}=\widehat{\boldsymbol{\eta}}}\|. \tag{14}$$

Lemma 1. *Let $\widehat{\boldsymbol{\beta}}$ be a solution of (6). Let $\lambda = \lambda(\widehat{\boldsymbol{\beta}})$ be the associated Lagrangian multiplier. Then λ and $\widehat{\boldsymbol{h}} = \nabla_{\boldsymbol{\beta}} l(\boldsymbol{\beta})|_{\boldsymbol{\beta}=\widehat{\boldsymbol{\beta}}}$ are constant across all solutions $\widehat{\boldsymbol{\beta}}_{(i)}$ of (6).*

Proof. Since the value of the objective function $l(\boldsymbol{\eta}_{(i)}) = l_*$ is constant across all solutions and l is strictly convex in $\boldsymbol{\eta} = X\boldsymbol{\beta}$ and convex in $\boldsymbol{\beta}$, it follows that $\widehat{\boldsymbol{\eta}}$ must be constant across all solutions $\widehat{\boldsymbol{\beta}}_{(i)}$, which implies that $\nabla_{\boldsymbol{\beta}} l(\boldsymbol{\beta})|_{\boldsymbol{\beta}=\widehat{\boldsymbol{\beta}}} = X^\top \nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta})|_{\boldsymbol{\eta}=\widehat{\boldsymbol{\eta}}}$ is constant across all solutions. Uniqueness of λ follows now from (14). □

Theorem 2. *Let λ be the Lagrangian parameter associated with some (any) solution $\widehat{\boldsymbol{\beta}}$ of (6) and let $\widehat{\boldsymbol{h}}$ be the unique gradient vector at the optimum. Let $\mathcal{B} = \{j_1, \dots, j_p\}$ be the unique set of indices for which $\|\widehat{\boldsymbol{h}}_j\| = \lambda$. Then $\widehat{\boldsymbol{\beta}}_j = \mathbf{0}_{d_j}$ $\forall j \notin \mathcal{B}$ across all solutions $\widehat{\boldsymbol{\beta}}_{(i)}$ of (6).*

Proof. A solution with $\widehat{\boldsymbol{\beta}}_j \neq \mathbf{0}_{d_j}$ for at least one $j \notin \mathcal{B}$ would contradict (13). □

Assume that an algorithm has found a solution $\widehat{\boldsymbol{\beta}}$ of (6) with the set of "active" groups $\mathcal{A} := \{j : \widehat{\boldsymbol{\beta}}_j \neq \mathbf{0}\}$. If $\mathcal{A} = \mathcal{B} = \{j : \|\widehat{\boldsymbol{h}}_j\| = \lambda\}$, then there cannot exist any other solution with an active set $\mathcal{A}'$ with $|\mathcal{A}'| > |\mathcal{A}|$. Thus, $\mathcal{A} = \mathcal{B}$ implies that all relevant groups are contained in the solution $\widehat{\boldsymbol{\beta}}$. Otherwise, the additional elements in $\mathcal{B}$ which are not contained in $\mathcal{A}$ define all possible groups that potentially become active in alternative solutions.

Note that $\mathcal{A} = \mathcal{B}$ guarantees that we cannot have "overlooked" relevant groups, which is typically sufficient in practical applications. We will call such a solution *complete*. However, $\mathcal{A}$ might still contain redundant groups, and we might be additionally interested if we have found a *unique* (an thus minimal) set $\mathcal{A}$. The following theorem characterizes a simple test for uniqueness under a further rank assumption of the data matrix X.

Theorem 3. *Assume that every $n \times n$ submatrix of X has full rank. Let $\mathcal{A}$ be the active set corresponding to some solution $\widehat{\boldsymbol{\beta}}$ of (6) and let $X_{\mathcal{A}}$ be the $n \times s$ submatrix of X composed of all active groups. Assume further that $\mathcal{A}$ is complete, i.e. $\mathcal{A} = \mathcal{B}$. Then, if $s \leq n$, $\widehat{\boldsymbol{\beta}}$ is the unique solution of (6).*

Proof. Since the set $\mathcal{B}$ is unique, the assumption $\mathcal{A} = \mathcal{B}$ implies that the search for the optimal solution can be restricted to the space $\mathbb{S} = \mathbb{R}^s$. If $s \leq n$, $X_\mathcal{A}$ must have full rank by assumption. Thus, $l(\beta_\mathbb{S})$ is a strictly convex function on $\mathbb{S}$ which is minimized over the convex constraint set. Thus, $\widehat{\boldsymbol{\beta}}_\mathbb{S}$ is the unique minimizer on $\mathbb{S}$. Since all other $\widehat{\boldsymbol{\beta}}_{j:j\notin\mathcal{A}}$ must be zero, $\widehat{\boldsymbol{\beta}}$ is unique on the whole space. □

In practice, it might be difficult to guarantee the rank condition in the above theorem. Note, however, that for a given set $\mathcal{A}$ and associated matrix $X_\mathcal{A}$ it is sufficient to check if *rank*$(X_\mathcal{A}) = s$ via SVD or QR-decomposition.

3. An Efficient Active-Set Algorithm

The characterization of optimal solutions presented above is now used to build a highly efficient algorithm, which is a straight-forward generalization of the subset algorithm for the standard Lasso problem presented in (Osborne et al., 2000). Similar ideas for the standard Lasso have also been introduced in (Shevade & Keerthi, 2003). The algorithm starts with only one active group. The selection of further active groups (or their removal) is guided by observing Lagrangian violations. Testing for completeness of the active set will then identify all groups that could have nonzero coefficients in alternative solutions.

A: Initialize set $\mathcal{A} = \{j_0\}$, $\boldsymbol{\beta}_{j_0}$ arbitrary with $\|\boldsymbol{\beta}_{j_0}\| = \kappa$.
B: Optimize over the current active set $\mathcal{A}$. Define set $\mathcal{A}^+ = \{j \in \mathcal{A} : \|\boldsymbol{\beta}_j\| > 0\}$ (some $\boldsymbol{\beta}_j$ could have vanished during optimization). Define $\lambda = \max_{j\in\mathcal{A}^+} \|\boldsymbol{h}_j\|$. Adjust the active set $\mathcal{A} = \mathcal{A}^+$.
C: Lagrangian violation. $\forall j \notin \mathcal{A}$, check if $\|\boldsymbol{h}_j\| \leq \lambda$. If this is the case, we have found a global solution. Otherwise, include the group with the largest violation to $\mathcal{A}$ and go to **B**.
D: Completeness and uniqueness. $\forall j \notin \mathcal{A}$, check if $\|\boldsymbol{h}_j\| = \lambda$. If so, there might exist other solutions with identical costs that include these groups in the active set. Otherwise, the active set is *complete* in the sense that it contains all relevant groups. If X_a has full rank $s \leq n$, *uniqueness* can be checked additionally via theorem 3. Note that step **D** requires (almost) no additional computations, since it is a by-product of step **C**.

The above algorithm is easily extended to practical optimization routines in which we stop the fitting process at a predefined tolerance level: testing for "completeness within a ϵ-range" ($|\|\boldsymbol{h}_j\| - \lambda| < \epsilon$ in **D** with ϵ being the maximum deviation of gradient norms from λ in the active set) will then identify all potentially active groups in alternative solutions with costs close to the actual costs.

The minimization in step **B** can be performed efficiently by the projected gradient method introduced in (Kim et al., 2006), which is applicable for all continuous convex cost functions. Finding the projection is typically the computational bottleneck in methods of this kind. For our special case, however, the projection can be found very efficiently. We refer the reader to (Kim et al., 2006) for details.
Iterate:
B1: Gradient. At time $t-1$, set $\boldsymbol{b} = \boldsymbol{\beta}^{t-1} - s\nabla_{\boldsymbol{\beta}} l(\boldsymbol{\beta}^{t-1})$ and $\mathcal{A}^+ = \mathcal{A}$, where s is a step-size parameter.
B2: Projection. For all $j \in \mathcal{A}^+$ define $M_j := \|\boldsymbol{b}_j\| + (\kappa - \sum_j \|\boldsymbol{b}_j\|)/|\mathcal{A}^+|$. If $M_j \geq 0\, \forall j \in \mathcal{A}^+$, go to **B3**. Else update the active set $\mathcal{A}^+ = \{j : M_j > 0\}$ and repeat **B2**.
B3: New solution. For all $j \in \mathcal{A}^+$ set $\boldsymbol{\beta}_j^t = \boldsymbol{b}_j M_j / \|\boldsymbol{b}_j\|$. For all other $j \in \mathcal{A}, j \notin \mathcal{A}^+$ set $\boldsymbol{\beta}_j^t = 0$.

Note that during the whole algorithm, access to the full set of variables is only necessary in steps **C** and **D**, which are outside the core optimization routine. Thus, in large-scale applications where not all groups can be hold in the main memory, we still have a rather efficient method, even if we have to access external storage in steps **C/D**.

Computing the Solution Path. Contrary to the standard Lasso, the Group-Lasso does not exhibit a piecewise linear solution path. Algorithms like LARS (Efron et al., 2004) are therefore not applicable. Despite this problem, we can still approximate the solution path on a grid of constraint values with almost no additional costs: starting with a very small $\kappa^{(0)}$ (which will result in a small active set), we iteratively relax the constraint, resulting in a series of increasing values $\kappa^{(i)}$. Note that at the i-th step, the previous solution $\boldsymbol{\beta}(\kappa^{(i-1)})$ is a feasible initial estimate since $\kappa^{(i)} > \kappa^{(i-1)}$. Typically only few further iterations are needed to find $\boldsymbol{\beta}(\kappa^{(i)})$. Completeness/uniqueness can be tested efficiently at every step i. In practical applications we observed that the stepwise approximation of the solution path up to some final $\kappa^{(f)}$ is usually *faster* than directly computing the solution for $\kappa^{(f)}$, probably because the stepwise procedure allows the use of larger stepsizes.

4. Applications

As a first application example we use synthetic data generated by a script that has been used in the context of the NIPS'03 feature selection workshop (follow the link "dataset description" on the workshop webpage www.clopinet.com/isabelle/Projects/NIPS2003/#challenge). We reproduced the XOR example explained in the above cited document: there are two classes, each of which is composed of two Gaussian clusters. Two "useful" features are drawn from $N(0,1)$ for each cluster. Some covariance is added by multiplying by a random matrix. The clusters are placed in an XOR configuration and 3200 "useless" features are added, drawn from $N(0,1)$. All the features are shifted and rescaled randomly. Random noise is then added according to $N(0, 0.1)$. Finally, 1% of the labels are ran-

domly flipped. We construct a training set of size 2000 and a test set of size 6000.

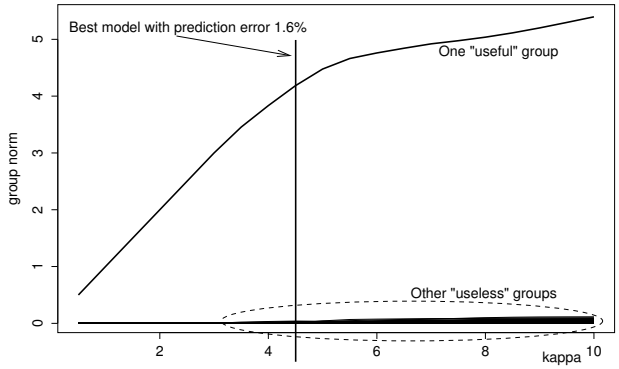

Figure 1. Solution path for the XOR problem with 3200 noise dimensions. The norm of the one "useful" group grows steeply when the constraint is relaxed. What appears as a horizontal "thickening" line is an overlay of 275 "useless" groups.

Without feature selection, prediction becomes very difficult: a SVM with RBF kernel achieves 42% test error (on the subset of the two "useful" features the error decreases to 1.5%). Moreover, simple feature selection methods like correlation-based scoring fail badly on these data.

We expand the dataset in a polynomial basis of degree 2, i.e. each pair of features (a, b) is mapped to a 5-dimensional vector $(a, b, a \cdot b, a^2, b^2)$. Given the 3202 features (2 + 3200 "useless"), this expansion yields $\approx 5 \cdot 10^6$ groups of size 5, each of which contains 5 quadratic interactions. We are, thus, working in a $\approx 2.5 \cdot 10^7$-dimensional space. Since the expanded feature set cannot be hold in the main memory, we only store the original dataset and recompute the expansions on demand. Despite this computational overhead, our active set algorithm allows us to optimize the Group-Lasso functional very efficiently, see also Figure 2. Since we are dealing with a classification problem, we choose the logistic model from the GLM family. Figure 1 shows the solution path for the logistic Group-Lasso when relaxing κ in 20 steps. Note that in the first iterations the algorithm was able to determine the one "useful" group of variables. The norm of the corresponding weight vector increases almost linearly until $\kappa \approx 4.5$, where the minimum error rate of 1.6% on the test set is obtained.

Testing both the completeness and the uniqueness of the active set gives a positive result, which guarantees that at this constraint value there are no alternative solutions. Further increasing κ leads to the selection of additional groups with spurious weights. The model obtained for $\kappa = 10$ uses 275 groups which include "useless" features and have norms < 0.2. Solutions for $\kappa > 5$ appear to be lacking completeness: our test identified a steeply increasing number of other groups that may also become active. Given that the "useless" variables are randomly drawn from a normal distribution, the observed lack of completeness might be caused by the limited numerical precision in the optimization routine: for models with $\kappa < 7$ we could indeed show by increasing the numerical precision that the solutions are complete, however at the price of drastically increasing computational costs. For larger models, however, we were not able to find complete solutions within any reasonable time limits. This result nicely shows that lacking completeness of Group-Lasso solutions is indeed a relevant issue in real-world applications which are necessarily computed with limited numerical precision. Besides the theoretical properties of our completeness test, this test might thus be also a valuable *practical* tool to detect possible ambiguities that are caused by numerical problems.

To compare the efficiency of our active set algorithm with related approaches, we measured the time needed to compute ten steps of the solution path ($\kappa = 1, 2, \ldots, 10$) for different numbers of "useless" groups. Figure 2 shows the observed computation times of three different methods: (1): the *blockwise sparse* method (Kim et al., 2006), (2): the *block coordinate* method by (Meier et al., 2008), (3): our algorithm. The comparison with method 1 was straight forward, since the same implementation was used (note that by dropping the active set selection mechanism, our method simply reduces to method (1)). In order to guarantee a fair comparison with method (2) for which we used the R-package `grplasso`, a few modifications were necessary: we first trained our method on the data and recorded the sequence of Lagrange parameters $\lambda_1, \ldots, \lambda_{10}$ corresponding to the sequence of constraints $\kappa = 1, 2, \ldots, 10$, since the `grplasso` package needs the Lagrange parameters on input. We also recorded the achieved log-likelihood at each step. We then trained method (2) on the dataset and adjusted its tolerance parameters as to (roughly) reproduce the recorded sequence of log-likelihoods. The double logarithmic scale in Figure 2 should make the interpretation of the plot rather insensitive against performance differences caused by using different implementations, since such differences are expected to produce additive shifts without changing the slopes.

For input instances that could be hold in the main memory, the log-log plot shows a relatively steep increase for the models (1) and (2), whereas method (3) increases linearly with a moderate slope. For the "out-of-core" models, we recomputed the groups whenever necessary (step **C/D** in our algorithm). We again see an almost linear increase of costs up to models including $\approx 10^6$ groups. Three observations seem to be important: (i) the slope of the curve for method (3) in the "out-of-core" regime does not even exceed the slope of the corresponding curve for method (2) at the end of the "cached" region; (ii) when fixing the costs at the level of method (2) at the end of the "cached" region, method (3) was able to solve instances which are larger by

at least $1 - 1.5$ orders of magnitude; (iii) comparison with method (1) shows that the active set formalism leads to a speed-up of several orders of magnitudes.

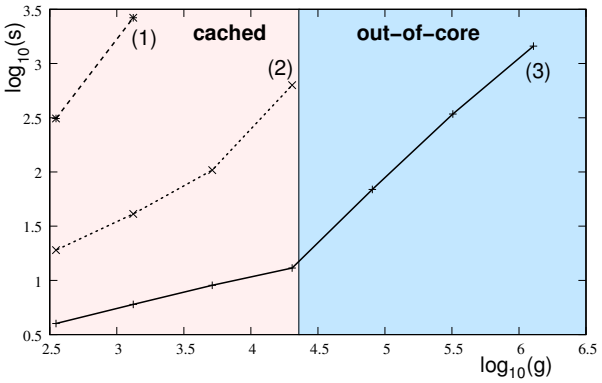

Figure 2. Log-log plot of computation time (y-axis, in seconds) for the XOR problem with logistic loss as a function of the number g of groups (x-axis). The three different methods are: 1:(Kim et al., 2006), 2:(Meier et al., 2008), 3: our algorithm.

Splice Site Detection. The prediction of splice sites has an important role in gene finding algorithms. Splice sites are the regions between coding (exons) and non-coding (introns) DNA segments. The $5'$ end of an intron is called a donor splice site and the $3'$ end an acceptor splice site. The *MEMset Donor* dataset (http://genes.mit.edu/burgelab/maxent/ssdata/) consists of a training set of 8415 true and 179438 false human donor sites. An additional test set contains 4208 true and 89717 "false" (or *decoy*) donor sites. A sequence of a real splice site is modeled within a window that consists of the last 3 bases of the exon and the first 6 bases of the intron. Decoy splice sites also match the consensus sequence at position zero and one. Removing this consensus "GT" results in sequences of length 7, i.e. sequences of 7 factors with 4 levels $\{A, C, G, T\}$, see (Yeo & Burge, 2004) for details. The goal of this experiment is to overcome the restriction to marginal probabilities (main effects) in the widely used *Sequence-Logo* approach (see Figure 4) by exploring all possible interactions up to order 4.

Following (Meier et al., 2008), the original training dataset is used to build a balanced training dataset and an unbalanced validation set which exhibits the same true/false ratio as the test set. The data are represented as a collection all factor interactions up to degree 4. Every interaction is encoded using dummy variables and treated as a group, leading to 120 groups of sizes varying between 4 (main effects) and 4^5 (4th order interactions). In total, we are working in a 33068-dimensional feature space. This dataset has also been analyzed in (Meier et al., 2008) with the Group-Lasso, but only up to 2nd order interactions.

To correct for the unbalancedness of the classes, the validation set is used to choose the best threshold τ on the classifier output. It is further used to select κ. The performance is measured in terms of the maximum correlation coefficient $\rho_{\max}$ between predicted and true labels.

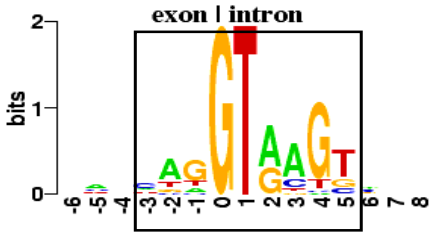

Figure 3. Sequence Logo representation of the human $5'$ splice site. The consensus "GT" appears at positions $0, 1$. The overall height of the stack of symbols at a certain position indicates the sequence conservation at that position, while the height of symbols within the stack indicates the relative frequency of each nucleic acid, see (Crooks et al., 2004). We model the splice sites in a window over positions $[-3, 5]$.

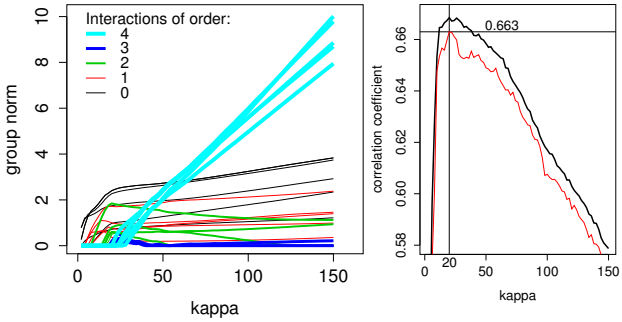

Figure 4. Left: solution path for donor splice site prediction. Color and thickness of curves indicate different orders of interactions. Right: Correlation coefficient as a function of κ. Bold curve: correlation on the validation set that is used for model selection (the thin vertical line indicates the chosen model). Thin curve: correlation on the separate test set.

From the correlation curve in Figure 4 we conclude that the inclusion of interactions of order three and greater does not improve the predictive performance and produces some pronounced overfitting effects. The model with the highest correlation coefficient ($\kappa = 20$) contains 36 groups: all 7 main effects, 21 1st-order interactions and 8 2nd-order interactions. Among the top-scoring groups we find the main effects at positions -1, 2 and 4, the interactions at positions $(4 : 5)$, $(-2 : -1)$ and $(2 : 3)$ and the triplet $(-3 : -2 : -1)$, which all share the property that they exclusively contain exon positions (or intron positions, respectively). One might conclude that long-range interactions between the preceding exon and the starting intron are of minor importance for splice site recognition. The completeness test reveals, however, that the solution with

36 groups is not complete, and that a complete model for $\kappa = 20$ additionally contains the four interactions $(-1:4)$, $(-2:5)$, $(-1:3:4)$ and $(-3:-1:2:5)$, all of which combine exon and intron positions. This is a nice example where the completeness test gives rise to query an initial hypothesis (about the weak exon-intron dependencies) which seems to be plausible from observing the Group-Lasso solution. It should be noticed that the obtained correlation coefficient of $\rho_{\max} = 0.663$ compares favorably with the result in the original paper (Yeo & Burge, 2004) ($\rho_{\max} = 0.659$), which has been viewed as among the best methods for short motive modeling.

The next experiment shows a situation where the completeness test indicates that the interpretability of the Group-Lasso might be generally complicated if relatively complex models are required. The problem is again the discrimination between true and "false" splice sites, this time, however, at the $3'$ end. Compared to the $5'$ situation, $3'$ (acceptor) splice site motives are less concentrated around the consensus nucleotide pair ("AG" at positions -2,-1 in Figure 5), which requires the use of larger windows. We trained the logistic Group-Lasso model on all interactions up to order 4 using windows of length 21. In total, we have 27896 groups which span a $22,458,100$-dimensional feature space. Despite this huge dimensionality, our active set algorithm was able to compute the solution path up to $\kappa = 150$ within roughly 20 hours. From the correlation curve in Figure 6 we conclude that in this example, the inclusion of 3rd- and 4th-order interactions does indeed increase the predictive performance. The optimal model at $\kappa = 66$ contains 386 groups. Among the 10 highest-scoring groups are the main effects at positions -3, -5 and 0, the 1st-order interactions $(-9:-8)$, $(-11:-10)$, $(-11:-9)$ and $(-12:-11)$, the triplet $(-6:-5:-3)$, the 3rd-order interaction $(-14:-9:0:1)$ and the 4th-order interaction $(-10:-8:-6:-3:2)$. The latter might be of particular interest, since it couples the position 2 which appears to be non-informative in the Sequence-Logo representation (Fig. 5) with positions at the end of the intron. This observation nicely emphasizes the strength of a model that is capable of exploring high-order dependencies among the positions.

A closer look at the results of the completeness tests in Figure 7 shows, however, that probably all solutions with $\kappa > 40$ are rather difficult to interpret, since a steeply increasing number of groups must be added to obtain complete models. This means that care should be taken when it comes to interpreting specific groups occurring in particular solutions (as we have done above). Since most of the models are not complete, it might well be that other groups not contained in a particular solution might be of high importance or even "substitute" identified groups. For the 4th-order interaction $(-10:-8:-6:-3:2)$ in the optimal solution with $\kappa = 66$ it might well be that there exist other groups that can take over the role of this interaction. Even though the high score of this group might indicate that a complete substitution is not very likely, the "discovery" of the coupling between position 2 and intron positions should not be accepted unquestioningly.

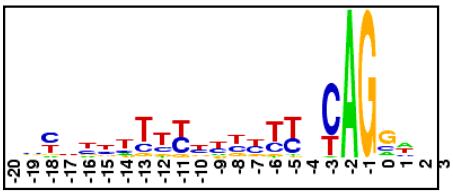

Figure 5. Sequence Logo representation of the human $3'$ splice site. The consensus "AG" appears at positions $-2, -1$. We use a window over positions $[-20, 2]$.

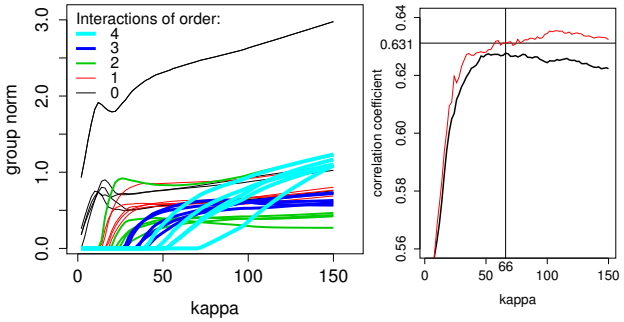

Figure 6. Left: solution path for $3'$ splice site prediction. The upper most curve represents the most important position -3 (last position of the intron). Right: Correlation coefficient as a function of κ. Bold curve: correlation on the validation set that is used for model selection. Thin curve: correlation on the separate test set.

5. Conclusion

The completeness- and uniqueness test presented here overcomes a severe problem of the Group-Lasso estimator for generalized linear models (GLM). Since in many practical applications the dimensionality exceeds the sample size, we cannot *a priori* assume that the active set of groups is unique, which somehow contradicts our goal of identifying important factors. Our testing procedure has the advantage that it identifies all groups that are potential candidates for the active set. Even if a solution is not complete, this latter property still allows us to explicitly list (and potentially investigate) the set of all *candidate* groups.

We have presented a highly efficient active-set algorithm that can handle extremely high-dimensional input spaces which typically arise when investigating high-order factor interactions or when using polynomial basis expansions. Our theoretical characterization of solutions is used to check both optimality and completeness/uniqueness.

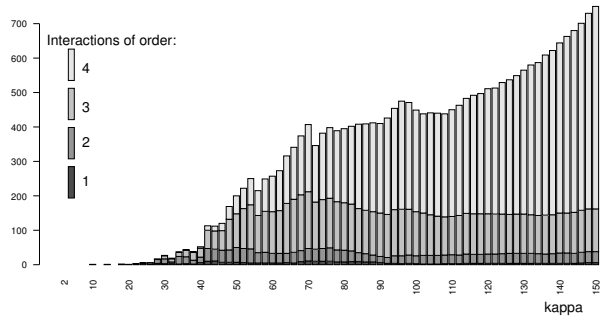

Figure 7. Acceptor splice site prediction: groups that must be included in the logistic Group-Lasso estimates to obtain complete models (gray values represent different orders of interactions).

The experiment on synthetic data in XOR configuration with additional noise features showed that the methods and concepts presented here can be successfully applied to problems with millions of groups. We demonstrated that non-completeness of solutions is indeed an important issue in real-world applications where round-off errors are unavoidable. Without any additional computational costs, the proposed completeness/uniqueness test easily detects such situations and additionally identifies all groups that must be included to achieve a complete model.

The splice-site prediction example confirmed these observations in a real-world context, where the inclusion of high-order factor interactions helps to increases the predictive performance but also leads to incomplete and, thus, potentially ambiguous solutions. The active set algorithm was able to approximate the solution path of the logistic Group-Lasso for feature-space dimensions up to $\approx 2 \cdot 10^7$ within a reasonable time, and the completeness test helped to avoid mis- or over-interpretations of identified interactions between the nucleotide positions. In particular for the $5'$ (donor-) splicing sites, we could show that the completeness test avoids a potentially severe misinterpretation regarding the independence of exon and intron positions.

While in the application examples we have focused on logistic classification problems, both the characterization of solutions and the algorithms proposed are valid for the much richer class of GLMs. Notable extensions include models for counting processes (e.g. Poisson or log-linear models). Details of such models for the analysis of sparse contingency tables in the spirit of the work in (Dahinden et al., 2007) will appear elsewhere. A C++ implementation of the active set algorithm with completeness test is available from the authors on request.

In the broad perspective – and in the light of recent theoretical results on the algorithmic complexity of feature selection (Nilsson et al., 2007) – one might conclude that feature selection can be simpler than previously thought.

References

Brown, L. D. (1986). *Fundamentals of statistical exponential families: with applications in statistical decision theory*. Hayworth, CA, USA: Institute of Mathematical Statistics.

Crooks, G., Hon, G., Chandonia, J., & Brenner, S. (2004). Weblogo: A sequence logo generator. *Genome Research*, *14*.

Dahinden, C., Parmigiani, G., Emerick, M., & Bühlmann, P. (2007). Penalized likelihood for sparse contingency tables with an application to full-length cDNA libraries. *BMC Bioinformatics*, *8*, 476.

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

Kim, Y., Kim, J., & Kim, Y. (2006). Blockwise sparse regression. *Statistica Sinica*, *16*, 375–390.

McCullaghand, P., & Nelder, J. (1983). *Generalized linear models*. Chapman & Hall.

Meier, L., van de Geer, S., & Bühlmann, P. (2008). The Group Lasso for Logistic Regression. *J. Roy. Stat. Soc. B*, *70*, 53–71.

Nilsson, R., Peña, J., Björkegren, J., & Tegnér, J. (2007). Consistent feature selection for pattern recognition in polynomial time. *JMLR*, *8*, 589–612.

Osborne, M., Presnell, B., & Turlach, B. (2000). On the LASSO and its dual. *J. Comp. and Graphical Statistics*, *9*, 319–337.

Shevade, K., & Keerthi, S. (2003). A simple and efficient algorithm for gene selection using sparse logistic regression. *Bioinformatics*, *19*, 2246–2253.

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

Wainwright, M., Jaakkola, T., & Willsky, A. (2005). A new class of upper bounds on the log partition function. *IEEE Trans. Information Theory*, *51*.

Wedderburn, R. W. M. (1973). On the existence and uniqueness of the maximum likelihood estimates for certain generalized linear models. *Biometrika*, *63*, 27–32.

Yeo, G., & Burge, C. (2004). Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *J. Comp. Biology*, *11*, 377–394.

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