Inferring the Group Lasso Structure via Bilevel Optimization

Jordan Frecon[1] **Saverio Salzo**[1] **Massimiliano Pontil**[1]

Abstract

We present a continuous bilevel optimization problem for inferring the Group Lasso structure. It relies on an approximation where the lower level problem is replaced by a smooth dual forward-backward scheme with Bregman distances. Theoretical guarantees regarding its convergence to the exact problem are also provided.

1. Introduction

Many classes of datasets have shown to exhibit a sparse representation when expressed as a linear combination of suitable dictionary elements. This has led over the past decades to the development of sparsity inducing norms and regularizers to unveil structure in the data. But there might also be a rich structure beyond the sparsity patterns, which is widely referred to as structured sparsity (Jenatton et al., 2011; Micchelli et al., 2013). In that sense, a lot of work has been devoted to encode *a priori* structure of the data in possibly overlapping groups (Jacob et al., 2009).

In the present paper, we restrict our study to the popular Group Lasso problem (Yuan & Lin, 2006). Given an observation $y \in \mathbb{R}^N$ and a regression matrix $D \in \mathbb{R}^{N\times P}$, the Group Lasso problem amounts in finding

$$\hat{x}(\theta) \in \underset{x\in\mathbb{R}^P}{\operatorname{argmin}} \frac{1}{2}\|y - Dx\|^2 + \lambda \sum_{l=1}^{L} \|x_{\mathcal{G}_l}\|_2, \tag{1}$$

for some regularization parameter $\lambda > 0$ and a non-overlapping group structure, i.e., an unordered partition of the features in L groups $\{\mathcal{G}_1, \ldots, \mathcal{G}_L\}$ such that $\cup_{l=1}^{L}\mathcal{G}_l = \{1, \ldots, P\}$ and $(\forall l \neq l'), \mathcal{G}_l \cap \mathcal{G}_{l'} = \emptyset$.

However, in many applications, we might have a large number of features whose group-structure $\{\mathcal{G}_1, \ldots, \mathcal{G}_L\}$ may be (partially) unknown. In addition, L itself might not be known. Nonetheless, the prior knowledge of the groups is crucial as it would lead to a lower prediction error (Lounici et al., 2011). To the best of our knowledge, only a few approaches have been devoted to inferring the groups under some hypothesis (Hernández-Lobato & Hernández-Lobato, 2013; Shervashidze & Bach, 2015).

Contributions and outline. The main novelties of this paper rely on i) the formulation of the problem of inferring groups as a continuous bilevel optimization problem and ii) a sound approximation scheme. This contribution is presented in Section 2. Moreover, iii) a new algorithmic solution based on an upper stochastic gradient descent and a lower dual forward-backward scheme with Bregman distances is devised in Section 3. The well-behavior and performance of the proposed approach are assessed on synthetic data in Section 4. Finally, conclusions and perspectives are drawn in Section 5.

Notations. Let $\mathcal{X}$ be an Euclidian space. $\Gamma_0(\mathcal{X})$ denotes the space of functions $h\colon \mathcal{X} \to]-\infty, +\infty]$ closed, proper and convex. We also denote by $\operatorname{argmin} h$ the set of minimizers of h or the minimizer of h when it is unique.

2. Proposed bilevel problem

2.1. Original problem

We tackle the problem of learning the groups by means of a bilevel multi-task learning problem, where the tasks are supposed to share a common group structure. In addition, we encapsulate the group structure by means of an hyperparameter $\theta \in \{0,1\}^{P\times L}$, defining at most L groups, such that $(\forall l \in \{1, \ldots, L\}, \forall p \in \{1, \ldots, P\})$, $\theta_{l,p} = 1$ if the p-th feature belongs to the l-th group, and 0 otherwise. In order to select θ, we consider the following continuous relaxation

Problem 2.1 (Exact bilevel problem)**.** *Let* $C\colon \mathrm{x} = (x_1, \ldots, x_T) \mapsto (1/T)\sum_{t=1}^{T} C_t(x_t)$*,* $C_t\colon \mathbb{R}^P \to \mathbb{R}$ *is smooth. Given some observations* $y_t \in \mathbb{R}^N$ *and regression matrices* $D_t \in \mathbb{R}^{N\times P}$ *for* $t \in \{1, \ldots, T\}$*, as well as some regularization parameters* $\lambda > 0$ *and* $\epsilon > 0$*, solve*

$$\underset{\theta\in\Theta}{\text{minimize}}\ \mathcal{U}(\theta) \quad \textit{with} \quad \begin{cases} \hat{\mathrm{x}}(\theta) = \operatorname{argmin}_{\mathrm{x}\in\mathbb{R}^{P\times T}}\ \mathcal{L}(\mathrm{x}, \theta), \\ \mathcal{U}(\theta) = C(\hat{\mathrm{x}}(\theta)), \end{cases} \tag{2}$$

[1] Department of Computational Statistics and Machine Learning, Istituto Italiano di Tecnologia, Genova, Italy. Correspondence to: Jordan Frecon <jordan.frecon@iit.it>.

Workshop on Modern Trends in Nonconvex Optimization for Machine Learning. The 35th International Conference on Machine Learning, Stockholm.

where

$$\begin{cases} \mathcal{L}(\mathrm{x},\theta) = \dfrac{1}{T}\displaystyle\sum_{t=1}^{T} \ell_t(x_t,\theta), \\ \ell_t(x,\theta) = \dfrac{1}{2}\|y_t - D_t x\|_2^2 + \dfrac{\epsilon}{2}\|x\|_2^2 + \lambda \displaystyle\sum_{l=1}^{L} \|\theta_l \odot x\|_2, \\ \Theta = \left\{\theta \in [0,1]^{P\times L} \;\middle|\; \displaystyle\sum_{l=1}^{L} \theta_l = \mathbb{1}_P\right\}. \end{cases} \tag{3}$$

The penalty term $(\epsilon/(2T))\sum_{t=1}^{T}\|x_t\|_2^2$, for some $\epsilon > 0$, has been added in order to ensure strong convexity of $\mathcal{L}$. A typical choice for C_t is the validation error $C_t(\hat{x}_t(\theta)) = (1/2)\|y_t^{(\mathrm{val})} - D_t^{(\mathrm{val})}\hat{x}_t(\theta)\|^2$ for which the selection of θ is motivated by the need of generalizing well to unseen data.

The following result concerns with the existence of solution of Problem 2.1 whose proof is given in the appendix.

Proposition 2.1 (Existence of solutions)**.** *Assume that Θ is a compact nonempty set of $\mathbb{R}_+^{P\times L}$, then $\theta \mapsto \hat{\mathrm{x}}(\theta)$ is continuous. Suppose in addition that C is a continuous function. Then Problem 2.1 admits solutions.*

2.2. Approximate problem

Usually, we don't have a closed form expression for $\hat{\mathrm{x}}(\theta)$ but we rather have an iterative mapping converging to $\hat{\mathrm{x}}(\theta)$ that we arbitrary stop after Q iterations. Henceforth, we actually solve an approximate problem of the following form.

Problem 2.2 (Approximate bilevel problem)**.** *Let C and Θ be as in Problem 2.1. Given two mappings $\mathcal{A}$ and $\mathcal{B}$, as well as a maximum number of inner iterations $Q \in \mathbb{N}$, solve*

$$\underset{\theta\in\Theta}{\text{minimize}}\ \mathcal{U}_Q(\theta),\ \textit{where} \begin{cases} \mathrm{u}^{(0)}(\theta) \textit{ is chosen arbitrarily} \\ \textit{for } q = 0,1,\dots,Q-1 \\ \left\lfloor\ \mathrm{u}^{(q+1)}(\theta) = \mathcal{A}(\mathrm{u}^{(q)}(\theta),\theta) \right. \\ \mathrm{x}^{(Q)}(\theta) = \mathcal{B}(\mathrm{u}^{(Q)}(\theta),\theta), \\ \mathcal{U}_Q(\theta) = C(\mathrm{x}^{(Q)}(\theta)). \end{cases} \tag{4}$$

The following theorem gives the conditions under which the approximate problem converges to the exact one as the number of inner iterations Q grows.

Theorem 2.1 (Convergence of the approximate problem)**.** *In addition to the assumptions of Problem 2.2, suppose that the iterates $\{\mathrm{x}^{(Q)}(\theta)\}_{Q\in\mathbb{N}}$ converge to $\hat{\mathrm{x}}(\theta)$ uniformly on Θ as $Q \to +\infty$. Then the approximate Problem 2.2 converges to the exact Problem 2.1 in the following sense*

$$\begin{cases} \inf_{\theta\in\Theta}\mathcal{U}_Q(\theta) \underset{Q\to+\infty}{\longrightarrow} \inf_{\theta\in\Theta}\mathcal{U}(\theta), \\ \operatorname{argmin}_{\theta\in\Theta}\mathcal{U}_Q(\theta) \underset{Q\to+\infty}{\longrightarrow} \operatorname{argmin}_{\theta\in\Theta}\mathcal{U}(\theta), \end{cases} \tag{5}$$

where the latter convergence is meant as set convergence (see appendix for details).

Theorem 2.1 justifies the minimization of $\mathcal{U}_Q$ (for sufficiently large Q) instead of $\mathcal{U}$. Concerning the lower level problem in (2)-(3), since it is nonsmooth, a nonsmooth solver is usually employed, meaning that $\mathcal{A}$ and $\mathcal{B}$ in (4) are nonsmooth. This causes $\mathcal{U}_Q$ to be nonsmooth, besides being nonconvex. In that case, minimizing $\mathcal{U}_Q$ is a challenge. Indeed, even just determining a (hyper)subgradient of $\mathcal{U}_Q$ in a stable fashion by recursively computing a subgradient of $\mathrm{u}^{(q)}(\theta)$ might be hopeless. Therefore, we embrace the idea proposed in (Ochs et al., 2016) to devise a smooth algorithm by relying on Bregman proximity operators and we make two advances. First, we propose a new algorithm based on a dual forward-backward scheme with Bregman distances where $\mathcal{A}$ and $\mathcal{B}$ are smooth. Second, we justify the convergence of the approximate bilevel problem with such inner Bregman scheme to the exact one by elaborating on recent results (Bauschke et al., 2016). This approach finally gives a smooth function $\mathcal{U}_Q$ whose gradient can be recursively computed by applying the standard chain rule (Griewank & Walther, 2008).

3. Algorithmic solution

3.1. Principle

In order the solve Problem 2.2, we propose the following projected gradient descent algorithm

$$(\forall k \in \{0,\dots,K-1\}),\ \theta^{(k+1)} = \mathcal{P}_\Theta\big(\theta^{(k)} - \nu\nabla\mathcal{U}_Q(\theta^{(k)})\big) \tag{6}$$

where $\mathcal{P}_\Theta$ denotes the projection onto the unit simplex Θ (see (Condat, 2016) for an efficient projection method on Θ), and $\nu > 0$ is a given step-size. Overall, this procedure requires to compute the Q-th iterate $\mathrm{x}^{(Q)}(\theta^{(k)})$ as well as the hypergradient $\nabla\mathcal{U}_Q(\theta^{(k)})$.

3.2. Solving the lower level problem

In this section, we draw our attention to the lower level problem in (2)-(3). Since it is separable with respect to the tasks, without loss of generality we can deal with a single task omitting the index t.

Problem 3.1. *Given some observation $y \in \mathbb{R}^N$, a regression matrix $D \in \mathbb{R}^{N\times P}$, regularization parameters $\lambda > 0$ and $\epsilon > 0$, as well as some group structure $\theta \in \Theta$, find*

$$\hat{x}(\theta) = \underset{x\in\mathbb{R}^P}{\operatorname{argmin}} \Big\{\ell(x,\theta) := f(x) + g(A_\theta x)\Big\}, \tag{7}$$

where $f = (\frac{1}{2}\|y - D\cdot\|_2^2 + \frac{\epsilon}{2}\|\cdot\|_2^2) \in \Gamma_0(\mathbb{R}^P)$ is smooth and ϵ-strongly convex, $g\colon \omega \mapsto \lambda\sum_{l=1}^{L}\|\omega_l\|_2$ is nonsmooth and belongs to $\Gamma_0(\mathbb{R}^{P\times L})$, and A_θ is the linear operator defined as $A_\theta : x \in \mathbb{R}^P \mapsto (\theta_1 \odot x, \dots, \theta_L \odot x) \in \mathbb{R}^{P\times L}$.

Since the proximity operator of $g\circ A_\theta$ cannot be computed in closed form, we cannot use the standard forward-backward algorithm (Combettes & Wajs, 2005) to solve Problem 3.1. Therefore, we tackle its dual problem.

Problem 3.2. *Find a solution $\hat{u}(\theta)$ of*

$$\underset{u\in\mathbb{R}^{P\times L}}{\text{minimize}} \left\{\tilde{\ell}(u,\theta) := f^*(-A_\theta^* u) + g^*(u)\right\}, \qquad (8)$$

where f^* and g^* denote the Fenchel conjugates of f and g respectively, and where A_θ^* is the adjoint operator of A_θ, that is, $A_\theta^* \colon u \in \mathbb{R}^{P\times L} \mapsto \sum_{l=1}^{L} \theta_l \odot u_l \in \mathbb{R}^P$.

Note that the dual Problem 3.2 admits a solution, since strong duality holds and the primal Problem 3.1 has a solution. Moreover it is a smooth constrained convex optimization problem. Indeed, since f is closed and ϵ-strongly convex, it follows that f^* is everywhere differentiable with ϵ^{-1}-Lipschitz continuous gradient and hence $\nabla[f^*\circ(-A_\theta^*)] = -A_\theta \nabla f^* \circ (-A_\theta^*)$ is $\|A_\theta\|^2\epsilon^{-1}$-Lipschitz continuous. Besides, we have $\nabla f^* = (\nabla f)^{-1} = (D^\top D + \epsilon \mathbb{1}_P)^{-1}(\cdot + D^\top y)$. On the other hand, g^* is the indicator function of the product of L balls $\mathcal{B}_2(\lambda)\times\ldots\mathcal{B}_2(\lambda) := \mathcal{B}_2(\lambda)^L$, i.e., $g^*(u) = \sum_{l=1}^{L} \imath_{\mathcal{B}_2(\lambda)}(u_l)$, where $\mathcal{B}_2(\lambda)$ is the closed ball of $\mathbb{R}^P$ centered at zero and of radius λ.

We propose to solve Problem 3.1 by applying a forward-backward algorithm with Bregman distances to the dual Problem 3.2 (Bauschke et al., 2016; Van Nguyen, 2017) and using the primal-dual link $x = \nabla f^*(-A_\theta^* u)$. This algorithm calls for a Bregman proximity operator of g^* which can be made smooth with an appropriate choice of the Bregman distance. In the following, we provide the related details.

Definition 3.1 (Bregman proximity operator (Van Nguyen, 2017))**.** Let $\mathcal{X}$ be an Euclidean space, $h \in \Gamma_0(\mathcal{X})$ and let $\Phi \in \Gamma_0(\mathcal{X})$ be a Legendre function. Then, the Bregman proximity operator (in Van Nguyen sense) of h with respect to Φ is $\operatorname{prox}_h^\Phi(v) = \operatorname{argmin}_{u\in\mathcal{X}} h(u) + \Phi(u) - \langle u, v\rangle$.

The dual forward-backward algorithm with Bregman distances (FBB) for Problem 3.1 is as follows. Given some step-size $\beta > 0$ and $u^{(0)}(\theta)$, then

$$\left\{\begin{array}{l} \text{for } q = 0,1,\ldots,Q-1 \\ \left\lfloor \begin{array}{l} u^{(q+1)}(\theta) = \operatorname{prox}_{\beta g^*}^{\Phi}\big(\nabla\Phi(u^{(q)}(\theta)) \\ \qquad\qquad\qquad +\beta A_\theta \nabla f^*(-A_\theta^* u^{(q)}(\theta))\big) \end{array}\right. \\ x^{(Q)}(\theta) = \nabla f^*(-A_\theta u^{(Q)}(\theta)). \end{array}\right. \qquad (9)$$

The updating rules in (9) define the mappings $\mathcal{A}$ and $\mathcal{B}$ in Problem 2.2. In this case, the mapping $\mathcal{B}$ is smooth, whereas the smoothness of $\mathcal{A}$ depends on the choice of Φ. We consider $\Phi(u) = \sum_{l=1}^{L}\phi(u_l)$ to make $\mathcal{A}$ separable. Moreover, in order to obtain a smooth update, we resort to the following Legendre function which handles the ball constraint.

Definition 3.2. The separable Hellinger-like function is defined as $\Phi\colon u \mapsto \sum_{l=1}^{L}\phi(u_l)$ where for every $u_l \in \mathbb{R}^P$,

$$\phi(u_l) = \begin{cases} -\sqrt{\lambda^2 - \|u_l\|_2^2}, & \text{if} \quad u_l \in \mathcal{B}_2(\lambda), \\ +\infty, & \text{otherwise.} \end{cases} \qquad (10)$$

For such choice, the corresponding forward-backward scheme with Bregman distance is given in Algorithm 1 (see appendix). The following theorem addresses the convergence of Algorithm 1. The proof is given in the appendix.

Theorem 3.1 (Convergence of dual FBB scheme)**.** *The sequence $\{x^{(Q)}(\theta)\}_{Q\in\mathbb{N}}$ generated by Algorithm 1 converges to the solution $\hat{x}(\theta)$ of Problem 3.1 uniformly on Θ for any step-size $0 < \beta < \lambda^{-1}\epsilon\|A_\theta\|^{-2}$. In addition when $\beta = \lambda^{-1}\epsilon\|A_\theta\|^{-2}/2$,*

$$\frac{1}{2}\|x^{(Q)}(\theta) - \hat{x}(\theta)\|_2^2 \leq \frac{2\lambda\epsilon^{-2}}{Q}\|A_\theta\|^2 D_\Phi(\hat{u}(\theta), u^{(0)}). \qquad (11)$$

This result applies to every task of the lower level objective in Problem 2.1 and hence we have that the sequence $\{\mathrm{x}^{(Q)}(\theta)\}_{Q\in\mathbb{N}}$, collecting all the tasks, converge uniformly to $\hat{\mathrm{x}}(\theta)$ on Θ. Therefore, the requirements of Theorem 2.1 are met and the solutions of Problems 2.2 converge to the solutions of Problem 2.1 as $Q \to +\infty$. This provides, to the best of our knowledge, the first theoretical justification of the framework proposed in (Ochs et al., 2016).

3.3. Computation of the hypergradient

In this section, we discuss the computation of the (hyper)gradient of $\mathcal{U}_Q$. It follows from (4) that, for every $\theta\in\Theta$,

$$\nabla\mathcal{U}_Q(\theta) = \frac{1}{T}\sum_{t=1}^{T} \underbrace{[(x_t^{(Q)})'(\theta)]^\top}_{\mathbb{R}^{(P\times L)\times P}} \underbrace{\nabla C_t(x_t^{(Q)}(\theta))}_{\mathbb{R}^P} \in \mathbb{R}^{P\times L}, \qquad (12)$$

where $\nabla_x C_t(x_t^{(Q)}(\theta)) = D_t^{(\mathrm{val})\top}(D_t^{(\mathrm{val})}x_t^{(Q)}(\theta) - y_t^{(\mathrm{val})})$. Instead of recursively computing $(u^{(q)})'(\theta)$ by forward differentiation, we implement the reverse mode (Griewank & Walther, 2008; Franceschi et al., 2017) which aims at evaluating the product $[(x_t^{(Q)})'(\theta)]^\top \nabla C_t(x_t^{(Q)}(\theta))$ itself. This method permits to store matrices of smaller size. The details are given in Algorithm 2 in the appendix. In addition, as suggested in (Griewank & Walther, 2008, Chapter 15), we implement a variant of Algorithm 2 in which all the derivatives of the mapping $\mathcal{M}$ are evaluated at the last iterate $x^{(Q)}$, $z^{(Q)}$, and $u^{(Q)}$ to reduce the execution time. In our experiments, we observe that the hypergradient is left unchanged by this operation as long as Q is large enough.

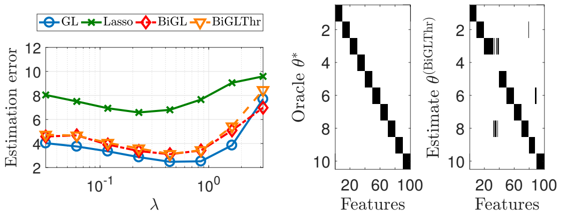

Figure 1. Left: the comparison of estimation errors show that the proposed (BiGL) and (BiGLThr) estimates yield performance close to the oracle (GL). Right: $\theta^{(\mathrm{BiGLThr})}$ satisfactorily reflects the oracle θ^*.

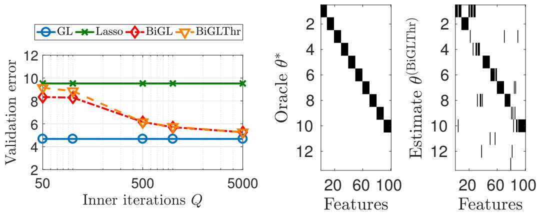

Figure 2. Left: impact of Q on the validation error. Right: an adequate estimation of the groups can be obtained even when the number of groups is set to 20 instead of 10.

3.4. Implementation details

The general scheme of the proposed bilevel method is given in Algorithm 3 in the appendix. Let us mention a few comments about its implementation.

First, all operations are computed in parallel with respect to the tasks. Second, since the hypergradient in (12) has the form of a sum of T terms, we implement a stochastic variant, by estimating the hypergradient $\nabla \mathcal{U}_Q$ on a single task chosen at random. Finally, we initialize $\theta^{(0)} = \mathcal{P}_\Theta(L^{-1}\mathbb{1}_{P\times L}+n)$ where $n \sim \mathcal{N}(0_{P\times L}, 0.1L^{-1}\mathbb{1}_{P\times L})$ in order to be as less informative as possible regarding features relationship while still breaking the symmetry by adding a small perturbation.

4. Numerical experiments

In this section, we devise synthetic experiments to illustrate and assess the performance of the proposed method.

Experimental setting. We consider the ill-posed setting ($N = 50$, $P = 100$) where θ^* is made of $L^* = 10$ groups equally distributed over the features. We fix $T = 500$ and every x_t^* is set to have non-zero coefficients equal to 1 in at most 2 groups chosen at random. All datasets are synthesized as follow. For every $t \in \{1, \dots, T\}$, $D_t \sim \mathcal{N}(0_{N\times P}, \mathbb{1}_{N\times P})$ is then normalized column-wise, and $y_t = D_t x_t^* + n$ where $n \sim \mathcal{N}(0_N, 0.3\mathbb{1}_N)$. We set $(Q = 500, \epsilon = 10^{-3}, \nu = 0.1, K = 5\cdot 10^3)$ and denote the proposed solution as $\theta^{(\mathrm{BiGL})}$. We also consider its threshold counterpart $\theta^{(\mathrm{BiGLThr})}$ where each feature is assigned to its most dominant group. These two solutions are compared to $\theta^{(\mathrm{Lasso})} = \mathrm{diag}(\mathbb{1}_P)$ and oracle Group Lasso $\theta^{(\mathrm{GL})} = \theta^*$.

Illustration. First, we illustrate the well-behaviour of the algorithmic solution, for various values of λ, when L^* is known. For every λ's, $\mathcal{U}_Q(\theta^{(k)})$ is shown in Figure 3 (appendix) to decrease as k grows. The corresponding solutions yield performance close to (GL) as shown by the validation, test and estimation error (see Figure 1 and appendix). More importantly, for λ which minimizes the validation error, denoted $\lambda_{\min}$, the estimate $\theta^{(\mathrm{BiGLThr})}$, reported in Figure 1 (right), satisfactorily reflects θ^*, thus showing that minimizing the validation error permits to infer the groups.

Impact of Q. We propose to investigate the impact of Q on the validation error. To do so, we repeat the same experiment for $\lambda = \lambda_{\min}$ and different values of Q. Once the estimates $\theta^{(\mathrm{BiGL})}$ and $\theta^{(\mathrm{BiGLThr})}$ are obtained, the validation errors (where $\hat{\mathrm{x}}(\cdot)$ are computed *a posteriori* for 10^4 iterations) are plotted as functions of Q in Figure 2 (left). Results show that increasing Q sufficiently large permits to reach performance close to (GL). In addition, we stress out that, for $Q \geq 500$, the performance of (BiGL) and (BiGLThr) become indistinguishable thus exhibiting that the algorithm does tend to assign a single group to each feature.

Impact of T. We report in Figure 4 (appendix) the estimation errors as functions of T. We observe that the performance of (BiGLThr) gets closer to those of (GL) as T grows. Hence, this confirms that inferring θ^* is intrinsically a multi-task problem that benefits from having many tasks.

Impact of L. Whereas we have assumed $L^* = 10$ known, we now suggest to relax this assumption and let the algorithm find at most $L = 20$ groups. The estimate $\theta^{(\mathrm{BiGLThr})}$ is displayed in Fig. 2 (right) where 7 of the 10 surplus groups identified by the algorithm have been screened. Interestingly, only 3 excess groups still remain but they yield a minor influence as they only concern few features. Overall, θ^* is still satisfactorily estimated.

5. Conclusion

This contribution studied the problem of inferring the Group Lasso structure by solving a continuous bilevel problem. This method falls within the framework proposed in (Ochs et al., 2016). However, here we made progress on two fronts: i) we replaced the lower level Group Lasso problem by a new smooth dual forward-backward algorithm with Bregman distances; ii) we proved that the related approximate bilevel problem converges to the exact bilevel Group Lasso problem. This work paves the way to the inference of groups in even more general settings including classification and overlapping groups (Jacob et al., 2009).

References

Bauschke, H. H., Bolte, J., and Teboulle, M. A descent lemma beyond lipschitz gradient continuity: first-order methods revisited and applications. *Mathematics of Operations Research*, 42(2):330–348, 2016.

Beck, A. and Teboulle, M. A fast dual proximal gradient algorithm for convex minimization and applications. *Operations Research Letters*, 42(1):1 – 6, 2014.

Combettes, P. L. and Wajs, V. R. Signal recovery by proximal forward-backward splitting. *SIAM Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.

Condat, L. Fast projection onto the simplex and the ℓ_1-ball. *Mathematical Programming*, 158(1-2):575–585, 2016.

Franceschi, L., Donini, N., Frasconi, P., and Pontil, M. Forward and reverse gradient-based hyperparameter optimization. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1165–1173, Sydney, Australia, 06–11 Aug 2017.

Griewank, A. and Walther, A. *Evaluating Derivatives*. Society for Industrial and Applied Mathematics, second edition, 2008.

Hernández-Lobato, D. and Hernández-Lobato, J. M. Learning feature selection dependencies in multi-task learning. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 26*, pp. 746–754. Curran Associates, Inc., 2013.

Jacob, L., Obozinski, G., and Vert, J.-P. Group lasso with overlap and graph lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 433–440, New York, NY, USA, 2009. ISBN 978-1-60558-516-1.

Jenatton, R., Audibert, J.-Y., and Bach, F. Structured variable selection with sparsity-inducing norms. *J. Mach. Learn. Res.*, 12:2777–2824, November 2011.

Lounici, K., Pontil, M., Van De Geer, S., and Tsybakov, A. B. Oracle inequalities and optimal inference under group sparsity. *Ann. Statist.*, 39(4):2164–2204, 08 2011.

Micchelli, C. A., Morales, J., and Pontil, M. Regularizers for structured sparsity. *Adv. Comput. Math.*, 38(3):455–489, 2013.

Ochs, P., Ranftl, R., Brox, T., and Pock, T. Techniques for gradient-based bilevel optimization with non-smooth lower level problems. *Journal of Mathematical Imaging and Vision*, 56(2):175–194, Oct 2016.

Shervashidze, N. and Bach, F. Learning the structure for structured sparsity. *IEEE Transactions on Signal Processing*, 63(18):4894–4902, Sept 2015.

Van Nguyen, Q. Forward-backward splitting with bregman distances. *Vietnam Journal of Mathematics*, 45(3):519–539, 2017.

Yuan, M. and Lin, Y. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68 (1):49–67, 2006.