23nd International Conference on Artificial Neural Networks,
Sofia, 2013

DOI: https://doi.org/10.1007/978-3-642-40728-4

Group Fused Lasso

Carlos M. Alaíz, Álvaro Barbero, and José R. Dorronsoro

Dpto. Ingeniería Informática & Inst. Ingeniería del Conocimiento,
Universidad Autónoma de Madrid, 28049 Madrid, Spain
{carlos.alaiz,alvaro.barbero,jose.dorronsoro}@uam.es

Abstract. We introduce the Group Total Variation (GTV) regularizer, a modification of Total Variation that uses the $\ell_{2,1}$ norm instead of the ℓ_1 one to deal with multidimensional features. When used as the only regularizer, GTV can be applied jointly with iterative convex optimization algorithms such as FISTA. This requires to compute its proximal operator which we derive using a dual formulation. GTV can also be combined with a Group Lasso (GL) regularizer, leading to what we call Group Fused Lasso (GFL) whose proximal operator can now be computed combining the GTV and GL proximals through Dykstra algorithm. We will illustrate how to apply GFL in strongly structured but ill-posed regression problems as well as the use of GTV to denoise colour images.

Keywords: Group Fused Lasso, Group Total Variation, Group Lasso, Fused Lasso, Total Variation.

1 Introduction

The irruption of big data, i.e., the need to study problems having very large sample sizes or very large dimensions or both, has resulted in a renewed interest in linear models, either because processing large samples with non-linear models is computationally demanding, or because a large dimension yields rich enough patterns so that methods enlarging pattern dimension such as the kernel trick add marginal value. Among linear models, Mean Square Error is the simplest fitting function, although it is well known that some regularizer has to be added, either to ensure good generalization, or just because the initial problem may be ill-posed. Classic choices include $\|w\|_2^2$ (ridge regression) and $\|w\|_1$ (Lasso [8]), and recently more ℓ_1-based regularizers such as Group Lasso [10] or Fused Lasso [9], have been introduced.

From a general point of view all these models can be stated as the problem of finding a $w^* \in \mathbb{R}^M$ which minimizes a certain functional $f(w) = f_{\mathrm{L}}(w) + f_{\mathrm{R}}(w)$ of the weights, with f_{R} the regularization term which somehow bounds the complexity of the model and f_{L} the loss functional. In more detail, assume a training set composed by P input patterns, $\{x^p\}_{p=1}^P$, with $x^p \in \mathbb{R}^M$, and their corresponding targets $\{y^p\}_{p=1}^P$, $y^p \in \mathbb{R}$. If $X \in \mathbb{R}^{P \times M}$ is the matrix having input patterns as rows and $y \in \mathbb{R}^P$ is the target vector, the overall problem for square loss can be written as

$$\min_{w\in\mathbb{R}^M} f(w) = \min_{w\in\mathbb{R}^M} f_{\mathrm{L}}(w) + \lambda f_{\mathrm{R}}(w) = \min_{w\in\mathbb{R}^M} \|Xw-y\|_2^2 + \lambda f_{\mathrm{R}}(w), \qquad (1)$$

where λ is a parameter to control the strength of the regularizer.

Taking $f_{\mathrm{R}}(w) = \|w\|_1 = \sum_{i=1}^{M} |w_i|$ results in the Lasso approach (LA), which enforces sparsity in the coefficients with an implicit feature selection, since only those inputs corresponding to nonzero coefficients have an impact in the model.

In some problems the features can present a spatial structure which we may want the models to capture. One way to do this is to enforce similarity among the coefficients corresponding to nearby features. If we do not consider any multidimensional feature structure, this can be achieved using a Total Variation (TV) regularizer $\mathrm{TV}_1(w) = \sum_{i=2}^{M} |w_i - w_{i-1}|$, which penalizes the differences between consecutive coefficients. Some sparsity may also be wanted and the overall regularizer to be used is then $f_{\mathrm{R}}(w) = \|w\|_1 + \hat{\lambda}\,\mathrm{TV}_1(w) = \|w\|_1 + \hat{\lambda}\|Dw\|_1$, where $D \in \mathbb{R}^{(M-1)\times M}$ is the differencing matrix with $D_{i,i} = -1$, $D_{i,i+1} = 1$ and $D_{ij} = 0$ elsewhere. The resulting model is called the Fused Lasso (FL).

Neither LA nor FL do consider any possible group structure on the problem features and, therefore, the resulting models will not reflect it even if it may be present. Assume, however, that the pattern features x have such a group structure. We may then see x as a collection of multidimensional features, that is, x has NV components that come in N groups with V features each and therefore $x = (x_{1,1}, x_{1,2}, \ldots, x_{1,V}, x_{2,1}, x_{2,2}, \ldots x_{2,V}, \ldots, x_{N,1}, x_{N,2}, \ldots, x_{N,V})^\top \in \mathbb{R}^{NV}$. The first subscript in $x_{n,v}$ indicates the group (or the multidimensional feature) and the second subscript the group feature so x is decomposed in N blocks $x_n = (x_{n,1}, \ldots, x_{n,V})^\top$ that contain V variables. The mixed $\ell_{2,1}$ norm is possibly the easiest and most natural regularizer in this framework. More precisely, for a vector w with the above group structure, its $\ell_{2,1}$ norm $\|w\|_{2,1}$ is defined as $\|w\|_{2,1} = \sum_{n=1}^{N} \|w_n\|_2$, which is just the ℓ_1 norm of the ℓ_2 group norms. This leads to the Group Lasso model (GL) whose regularizer is then $f_{\mathrm{R}}(w) = \|w\|_{2,1}$.

In this work we will extend GL to a fused setting, introducing first a new Group Total Variation regularizer (GTV) defined as:

$$\mathrm{GTV}(w) = \sum_{n=2}^{N} \sqrt{\sum_{v=1}^{V} (w_{n,v} - w_{n-1,v})^2},$$

and considering a full regularization functional that adds the GTV term to the standard $\ell_{2,1}$ regularizer of GL. We can write it in compact notation as

$$f_{\mathrm{R}}(w) = \|w\|_{2,1} + \hat{\lambda}\|\bar{D}w\|_{2,1}, \text{ with } \bar{D} = \begin{pmatrix} -I & I & & \\ & \ddots & \ddots & \\ & & -I & I \end{pmatrix}. \qquad (2)$$

$\bar{D} \in \mathbb{R}^{(N-1)V\times NV}$ is the group differencing matrix, and $I \in \mathbb{R}^{V\times V}$ stands for the identity matrix. We call this model Group Fused Lasso (GFL). Notice that if $V = 1$ we recover FL, and if $V = M$, i.e., there is a single group with M

variables, GFL boils down to a variant of FL using a TV_2 regularizer, also known as ℓ_2-Variable Fusion [2].

We will solve the GFL optimization problem through convex proximal optimization techniques. We will essentially apply a variant of the FISTA algorithm which, in turn, requires that we can compute the proximal operator of the GFL regularizer, something we will do in Sect. 2. We point out that GFL with only the group $\|\bar{D}w\|_{2,1}$ penalty has been introduced in [6]. However, its solution is different from ours, as it reduces this GFL to a GL model that is then solved by a group LARS algorithm. We believe our approach to be better suited to deal with the full general GFL case. We shall illustrate the behaviour of GFL over two examples in Sect. 3, and we will close the paper in Sect. 4 with a discussion and pointers to further work.

2 Solving Group Fused Lasso with Proximal Methods

All the ℓ_1 regularizers of Sect. 1 lead to non-differentiable optimization problems, which prevents solving them by standard gradient-based methods. However, they fit very nicely under the paradigm of Proximal Methods (PMs) that we briefly review next. Recall that the function to be minimized in (1) is $f_{\mathrm{L}}(w) + f_{\mathrm{R}}(w)$, where we include the penalty factor λ in $f_{\mathrm{R}}(w)$.

Denote by $\partial h(w)$ the subdifferential at w of a convex function h; since both terms are convex and $f_{\mathrm{L}}(w)$ is differentiable, w^* will be a minimum of $f_{\mathrm{L}}(w) + f_{\mathrm{R}}(w)$ iff $0 \in \partial(f_{\mathrm{L}}(w^*) + f_{\mathrm{R}}(w^*))$ [3] or, by the Moreau–Rockafellar theorem, $0 \in \nabla f_{\mathrm{L}}(w^*) + \partial f_{\mathrm{R}}(w^*)$. Equivalently, we have $-\gamma\nabla f_{\mathrm{L}}(w^*) \in \gamma\lambda\partial f_{\mathrm{R}}(w^*)$ for any $\gamma > 0$ and, also, $w^* - \gamma\nabla f_{\mathrm{L}}(w^*) \in w^* + \gamma\partial f_{\mathrm{R}}(w^*) = (I + \gamma\partial f_{\mathrm{R}})(w^*)$. Thus, the set function $(I + \gamma\partial f_{\mathrm{R}})^{-1}$ verifies

$$w^* \in (I + \gamma\partial f_{\mathrm{R}})^{-1} \left(w^* - \gamma\nabla f_{\mathrm{L}}(w^*)\right). \tag{3}$$

Now, if F is a convex, lower semicontinuous function, its proximal operator at w with step $\gamma > 0$ is defined as

$$z_w = \mathrm{prox}_{\gamma;F}(w) = \arg\min_{z\in\mathbb{R}^M} \left\{\frac{1}{2}\|z - w\|_2^2 + \gamma F(z)\right\}.$$

Notice that then we have $0 \in z_w - w + \gamma\partial F(z_w)$, that is, $z_w \in (I+\partial F)^{-1}(w)$. For a general convex F, it can be shown [3] that ∂F is a monotone operator and, while in principle $(I+\partial F)^{-1}$ would be just a set-function, it is actually uniquely valued. Therefore, it defines a function for which $\mathrm{prox}_{\gamma;F}(w) = z_w = (I + \partial F)^{-1}(w)$ holds. Thus, going back to (3), it follows that $w^* = \mathrm{prox}_{\gamma;f_{\mathrm{R}}}(w^* - \gamma\nabla f_{\mathrm{L}}(w^*))$, which immediately suggests an iterative algorithm of the form

$$w^{k+1} = \mathrm{prox}_{\gamma;f_{\mathrm{R}}}\left(w^k - \gamma\nabla f_{\mathrm{L}}(w^k)\right).$$

This is at the heart of the well known proximal gradient method [7] and of its ISTA and FISTA (Fast Iterative Shrinkage–Thresholding Algorithm) extensions [4]. In particular, we will focus on FISTA, based on the pair of equations:

$$w^k = \mathrm{prox}_{\frac{1}{K};f_{\mathrm{R}}}\left(z^k - \frac{1}{K}\nabla f_{\mathrm{L}}(z^k)\right), \quad z^{k+1} = w^k + \frac{t^k - 1}{t^{k+1}}(w^k - w^{k-1}),$$

where $t^{k+1} = \frac{1}{2}(1+\sqrt{1+4t_k^2})$ and K is a Lipschitz constant for ∇f_{L}. Notice that these algorithms require at each step the computation of the proximal operator at the current w^k. We discuss next these operators for GFL.

Observe that to solve problem (2) for the complete GFL regularizer, we need the proximal operator of the sum of the GTV and GL terms. Both regularizers are not separable, so their joint proximal operator cannot be built by the usual expedient of applying consecutively the proximal operators of GTV and GL. However, we can still solve the proximal problem by the Proximal Dykstra (PD) [7] algorithm, which allows to compute the proximal operator of the sum of several terms combining their individual proximal operators in an iterative fashion. Therefore we can focus on computing each proximal operator separately. In our case, the proximal operator of the GL regularizer is just the group soft-thresholding [1] defined as $\operatorname{prox}_{\gamma;\|\cdot\|_{2,1}}(w_{n,v}) = w_{n,v}(1 - \gamma/\|w_n\|_2)^+$, and we will derive now the proximal operator for GTV, following an analogous argument to the one in [2] for TV. We have to solve

$$\operatorname{prox}_{\gamma;\mathrm{GTV}}(w) = \operatorname{argmin}_{z\in\mathbb{R}^M} \frac{1}{2}\|z-w\|_2^2 + \gamma\|\bar{D}z\|_{2,1}, \tag{4}$$

which is a particular case of the more general problem $\inf_{z\in\mathbb{R}^M} f(z) + \gamma r(Bz)$, where $B \equiv \bar{D}$, $r(\cdot) \equiv \|\cdot\|_{2,1}$ and $f(y) \equiv \frac{1}{2}\|y-w\|_2^2$. In turn, this is equivalent to $\inf_{z,v} f(z) + \gamma r(v)$ s.t. $v = Bz$, with $z \in \mathbb{R}^M$ and $v \in \mathbb{R}^{(N-1)V}$. Writing its Lagrangian as $\mathcal{L}(z, v; u) = f(z) + \gamma r(v) + u \cdot (Bz - v)$ with $u \in \mathbb{R}^{(N-1)V}$, we can transform the equivalent saddle point problem $\inf_{z,v} \sup_u \mathcal{L}(z, v, u)$ into the dual problem

$$\inf_u f^*(-B^\top u) + \gamma r^*\left(\frac{1}{\gamma}u\right),$$

by means of the Fenchel Conjugate $F^*(\hat{x}) = -\inf_x \{f(x) - x\cdot\hat{x}\}$ [3]. Going back to (4), it is easy to see that for $f(z) = \frac{1}{2}\|z-w\|_2^2$, we have $f^*(s) = \frac{1}{2}s\cdot s + s\cdot w$. The conjugate of the $\ell_{2,1}$ norm can be derived using the definition of Fenchel Conjugate and the conjugate of the ℓ_2 norm (the indicator function of the unitary ball), obtaining that $r^*(s)$ is the indicator function of the unitary balls for each group, $\iota_{\bigwedge_{n=1}^{N-1}\|s_n\|_2\le 1}$. Therefore, the dual problem becomes:

$$\min_u \left\{\frac{1}{2}\|\bar{D}^\top u\|_2^2 - u^\top\bar{D}w + \iota_{\bigwedge_{n=1}^{N-1}\|u_n\|_2\le\gamma}\right\} \equiv \begin{array}{l}\displaystyle\min_u \left\{\frac{1}{2}\|\bar{D}^\top u - w\|_2^2\right\}\\ \text{s.t. } \|u_n\|_2 \le \gamma,\ 1 \le n \le N-1,\end{array} \tag{5}$$

where we have completed squares and changed the indicator function to a set of constraints. Since problem (5) is quadratic with simple convex constraints, it can be easily solved using projected gradient. After that, z_w (i.e., the result of the proximal operator) can be recovered from the dual solution u^* through the equality $z_w = w - \bar{D}^\top u^*$, which follows from $0 = \nabla_z\mathcal{L} = z_w - w + B^\top u^*$.

To finish this section, we observe that the form of the $\ell_{2,1}$ norm implicitly assumes a 1-dimensional spatial structure for the data. However, many problems of interest, such as image processing, present a natural multidimensional structure that cannot be captured by the $\ell_{2,1}$ penalty. Working only with the GTV penalty, and as in [2], a solution for this is to combine several 1-dimensional GTV penalties to obtain a multidimensional GTV. For example, for problems with a 2-dimensional structure, we penalize changes in both row and column-wise adjacent features. More precisely, denoting the i-th row by $w^{[i,\cdot]}$ and the j-th column by $w^{[\cdot,j]}$, we can define the 2-dimensional GTV regularizer as $\mathrm{GTV}^{2d}(w) = \sum_i \mathrm{GTV}\left(w^{[i,\cdot]}\right) + \sum_j \mathrm{GTV}\left(w^{[\cdot,j]}\right)$. This can be easily extended to more than two dimensions but, again, notice that this multidimensional GTV regularizer is the sum of 1-dimensional GTVs. Those corresponding to the same dimension (for example, the terms $\mathrm{GTV}\left(w^{[i,\cdot]}\right)$ corresponding to the different columns) apply over different variables, and are therefore separable, so the proximal operator of the summation of a particular dimension can be computed just by composing the individual proximal operators. Nevertheless, each complete summation applies over all the variables, and they cannot be separated. In order to combine the proximal operators of the different dimensions we can use once again the PD algorithm. Similarly, for the case of a complete multidimensional GFL linear model, we should use PD to compute the proximal operator of the multidimensional GTV regularizer, and then combine the GTV and GL proximal operators applying again PD.

3 Experiments

We will present next an application of the GFL model over a synthetic regression example and the use of the GTV regularizer for colour image denoising.

We consider first a synthetic structured linear problem where pattern features are divided into 100 3-dimensional groups, i.e., we have $N = 100$ and $V = 3$. The optimal weights are structured in 4 consecutive segments of 25 groups with constant values for the three group coordinates. This defines an optimal weight $w^* = (w_1^*, w_2^*, w_3^*, w_4^*)^\top$ with each w_i^* constant; w^* is thus built in such a way that it makes the features to be simultaneously either active or inactive and in such a way that adjacent features have a block behaviour. The optimal w^* is then perturbed to obtain a weight vector of the form $\tilde{w}_{n,v} = w^*_{n,v} + \eta_{n,v}$ with $\eta \sim \mathcal{N}(0, 0.1)$ Gaussian noise. Random independent patterns x^p are then generated by a $\mathcal{N}(0,1)$ distribution, and the values $y^p = \tilde{w} \cdot x^p + \hat{\eta}_p$ with $\hat{\eta} \sim \mathcal{N}(0, 0.1)$ then define a regression problem. Notice that the underlying spatial structure of the weights imposes also an spatial structure on the y^p values. Moreover, if the number of generated x patterns is well below the problem dimension of 300, we will end up with an ill-posed problem. We will consider 600, 300, 100 and 50 training patterns and solve the regression problem using LA, GL, FL and GFL. In the latter case, we apply the complete 1-dimensional GFL linear model (with both the 1-dimensional GTV and the GL terms). The corresponding regularization parameters are chosen so that the estimated weights are closest to the generating weights in the ℓ_1 distance. Table 3 presents the corresponding results in terms of the distances $\|w - w^*\|_1$ and $\|w - w^*\|_2$. As can be seen,

Table 1. Distance to optimal weights for the considered structured linear regression models as a function of the number of training samples (lower is better).

Mod	**Training Size**			
	600	**300**	**100**	**50**
LA	23.59	29.91	1016.60	1284.88
GL	23.70	30.75	1024.45	1304.23
FL	10.61	11.28	*13.88*	29.60
GFL	*9.35*	*10.93*	15.57	*26.43*

$\|w - w^*\|_1$

Mod	**Training Size**			
	600	**300**	**100**	**50**
LA	1.74	2.21	96.28	126.47
GL	1.76	2.26	92.25	128.76
FL	0.86	0.97	1.26	2.40
GFL	*0.72*	*0.92*	*1.24*	*2.05*

$\|w - w^*\|_2$

GFL achieves the lowest $\|w - w^*\|_1$ distance in all the cases but one, and the lowest $\|w - w^*\|_2$ for all of them. Only FL is comparable, whereas LA and GL values are clearly worse for the 600 and 300 pattern problems and markedly fail when used with few training samples. As reference value, observe that the distances of the perturbed weights to the original ones are $\|\tilde{w} - w^*\|_1 = 24.81$ and $\|\tilde{w} - w^*\|_2 = 1.78$, close to the FL and GFL values but far away from the LA, GL ones. Moreover, Fig. 3 shows how GFL recovers quite well the inherent structure of the problem.

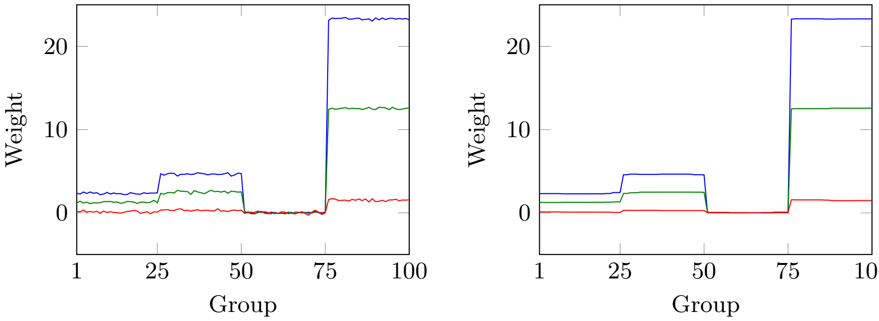

Fig. 1. Noisy weights (left) and weights recovered by GFL (right), using 600 patterns. The three colours represent different variables of the same group.

We consider next how to apply GTV to denoise colour images. Notice that images have a natural spatial structure, as pixels change smoothly and can be considered nearly constant in nearby regions (except in objects borders). Therefore, TV regularization has been extensively used for this task [5] on gray level images, in the form of the denoising model $\min_I \frac{1}{2}\|I - \tilde{I}\|_2^2 + \mathrm{TV}^{2d}(I)$ for a noisy image $\tilde{I}$ and some bidimensional form of TV, whose block structure permits to preserve the borders. When dealing with colour images a possible option is to apply TV denoising independently to each of the three RGB layers. However, we can also consider each pixel as a multi-valued (R,G,B) feature, making GTV fit naturally into this problem using the whole of the problem structure. Specifically, we will use the 2-dimensional GTV proximal operator, which can be easily computed as explained in Sect. 2. We will work with two different colour images. The first one (*peppers*) is perturbed by additive noise as $\tilde{I} = I + n$, with I the original image and $n \sim \mathcal{N}(0, 0.05)$. For the second image (*Lena*) we consider speckle noise, i.e., multiplicative uniform noise, with $\tilde{I} = I + uI$, where u is uniform with 0 mean and variance 0.25. Our goal here is to compare the potential advantages of GTV over 2-dimensional TV and for each model we select the optimal GTV and TV penalties as the ones that give the best Improved Signal-to-Noise Ratio (ISNR) over a single perturbed sample for each image. We then test TV and GTV denoising over 10 other different perturbations for additive and multiplicative noise. In all cases GTV performed better than TV, yielding an average ISNR of 10.73 ± 0.36 for additive noise and of 12.24 ± 0.24 for multiplicative noise; on the other hand, the ISNR averages for TV are 8.68 ± 0.27 and 10.97 ± 0.41, respectively. Figure 3 contains an example of denoising for the two different image and noise models described above.

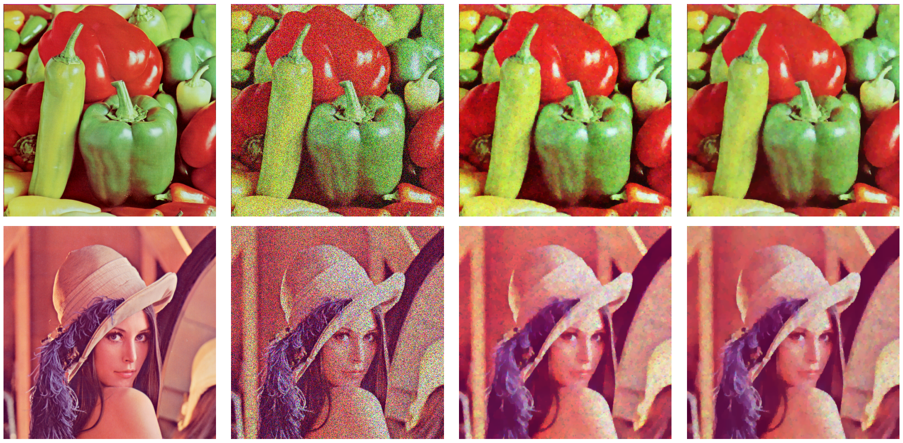
Original. Noisy. TV-cleaned. GTV-cleaned.

Fig. 2. Denoising with additive (upper row) and multiplicative (lower row) noise.

4 Conclusions

In this work we have proposed the Group Total Variation (GTV) regularizer, combining the multidimensional group-sparse features of the Group Lasso regularizer with the block spatial structure of the Total Variation penalty used by Fused Lasso. The GTV regularizer thus appears as a useful tool to reconstruct multidimensional patterns with a spatial structure that reflects smooth changes along the group features. Colour image denoising fits nicely in this framework and we have shown that GTV performs better than applying 1-dimensional Total Variation independently on each colour. Moreover, this GTV regularizer can be merged with a Group Lasso (GL) term, leading to what we call Group Fused Lasso (GFL). We have illustrated over a synthetic example how GFL effectively captures block structure when present, and makes use of it to address linear ill-posed problems with a number of features much larger than the sample size.

This kind of spatial structure can be found in other real world problems, particularly those for which the underlying data features are associated to geographical locations. Any sensible linear regression models for such problems should assign similar weight values to spatially close features, which is exactly the behaviour that GFL enforces. As further work we intend to study the advantages of GFL in such a kind of problems, which will require the use of the complete 2-dimensional GFL model as explained at the end of Sect. 2, and also to analyse the numerical complexity of the proposed models and possible ways to improve it.

Acknowledgement. With partial support from Spain's grant TIN2010-21575-C02-01 and the UAM–ADIC Chair for Machine Learning. The first author is supported by the FPU–MEC grant AP2008-00167.

References

1. Bach, F., Jenatton, R., Mairal, J., Obozinski, G.: Convex Optimization with Sparsity-Inducing Norms (2011), http://www.di.ens.fr/~fbach/opt_book.pdf
2. Barbero, A., Sra, S.: Fast newton–type methods for total variation regularization. In: Proceedings of the 28th International Conference on Machine Learning (ICML–11). pp. 313–320. New York, NY, USA (2011)
3. Bauschke, H., Combettes, P.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer (2011)
4. Beck, A., Teboulle, M.: A fast iterative shrinkage–thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences 2(1), 183–202 (2009)
5. Bioucas-Dias, J.M., Figueiredo, M.A.T.: A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration. Image Processing, IEEE Transactions on 16(12), 2992–3004 (2007)
6. Bleakley, K., Vert, J.P.: The group fused Lasso for multiple change-point detection. ArXiv e-prints (2011)
7. Combettes, P.L., Pesquet, J.C.: Proximal splitting methods in signal processing. Recherche 49, 1–25 (2009)
8. Tibshirani, R.: Regression shrinkage and selection via the Lasso. J. Roy. Statist. Soc. Ser. B 58(1), 267–288 (1996)
9. Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., Knight, K.: Sparsity and smoothness via the fused lasso. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67(1), 291–108 (2005)
10. Yuan, M., 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)