Non-convex regularization in remote sensing

Devis Tuia *Senior Member, IEEE*, Remi Flamary, Michel Barlaud,

***Abstract*— In this paper, we study the effect of different regularizers and their implications in high dimensional image classification and sparse linear unmixing. Although kernelization or sparse methods are globally accepted solutions for processing data in high dimensions, we present here a study on the impact of the form of regularization used and its parametrization. We consider regularization via traditional squared (ℓ_2) and sparsity-promoting (ℓ_1) norms, as well as more unconventional nonconvex regularizers (ℓ_p and Log Sum Penalty). We compare their properties and advantages on several classification and linear unmixing tasks and provide advices on the choice of the best regularizer for the problem at hand. Finally, we also provide a fully functional toolbox for the community[1].**

***Index Terms*— Hyperspectral, sparsity, regularization, remote sensing, non-convex, classification, unmixing.**

I. Introduction

Remote sensing image processing [1] is a fast moving area of science. Data acquired from satellite or airborne sensors and converted into useful information (land cover maps, target maps, mineral compositions, biophysical parameters) have nowadays entered many applicative fields: efficient and effective methods for such conversion are therefore needed. This is particularly true for data sources such as hyperspectral and very high resolution images, whose data volume is big and structure is complex: for this reason many traditional methods perform poorly when confronted to this type of data. The problem is even more exhacerbated when dealing with multi-source and mult-imodal data, representing different views of the land being studied (different frequencies, different seasons, angles, ...). This created the need for more advanced techniques, often based on statistical learning [2].

Among such methodologies, regularized methods are certainly the most successful. Using a regularizer imposes some constraints on the class of functions to be preferred during the optimization of the model and can thus be beneficial if we know what these properties are. The more often, regularizers are used to favour simpler functions over very complex ones, in order to avoid overfitting of the training data: in classification, the support vector machine uses this form of regularization [3], [4], while in regression examples can be found in kernel ridge regression or Gaussian processes [5].

Manuscript received September x, 2014;

This research was partially funded by the Swiss National Science Foundation, under grant PP00P2-150593.

DT is with the University of Zurich, Switzerland. Email: devis.tuia@geo.uzh.ch, web: http://geo.uzh.ch/ , Phone: +4144 635 51 11, Fax: +4144 635 68 48.

MB is with is with the University of Nice Sophia Antipolis, France.

RF is with the University of Nice Sophia Antipolis, OCA and Lagrange Lab., France. Email: remi.flamary@unice.fr, web: remi.flamary.com

DT and RF contributed equally to the paper.

Digital Object Identifier xxxx

[1] https://github.com/rflamary/nonconvex-optimization

But smoothness-promoting regularizers are not the only ones that can be used: depending on the properties one wants to promote, other choices are becoming more and more popular. A first success story is the use of Laplacian regularization [6]: by enforcing smoothness in the local structure of the data, one can promote the fact that points that are similar in the input space must have a similar decision function (Laplacian SVM [7], [8] and dictionary-based methods [9], [10]) or be projected close after a feature extraction step (Laplacian eigenmaps [11] and manifold alignment [12]). Another popular property to be enforced, on which we will focus the rest of this paper, is sparsity [13]. Sparse models have only a part of the initial coefficients which is active (i.e. non-zero) and are thus compact. This is desirable in classification when the dimensionality of the data is very high (e.g. when adding many spatial filters [14], [15] or using convolutional neural networks [16], [17]) or in sparse coding when we need to find a relevant dictionary to express the data [18]. Even though non-sparse models can work well in terms of overall accuracy, they still store information about the training samples to be used at test time: if such information is very high dimensional and the number of training samples is important, the memory requirements, the model complexity and – as a consequence – the execution time are strongly affected. Therefore, when processing next generation, large data using models generating millions of features [19], [20], sparsity is very much needed to make models portable, while remaining accurate. For this reason, sparsity has been extensively used in i) spectral unmixing [21], where a large variety of algorithms is deployed to select endmembers as a small fraction of the existing data [18], [22], [23], ii) image classification, where sparsity is promoted to have portable models either at the level of the samples used in reconstruction-based methods [24], [25] or in feature selection schemes [15], [26], [27] iii) and in more focused applications such as 3-D reconstruction from SAR [28], phase estimation [29] or pansharpening [30].

A popular approach to recover sparse features is to solve a convex optimization problem involving the ℓ_1 norm (or Lasso) regularization [31]–[33]. Proximal splitting methods have been shown to be highly effective in solving sparsity-constrained problems [34]–[36]. The Lasso formulation based on the penalty on the ℓ_1 norm of the model has been shown to be an efficient shrinkage and sparse model selection method in regression [37]–[39]. However, the Lasso regularizer is known to promote biased estimators leading to suboptimal classification performances when strong sparsity is promoted [40], [41]. A way out of this dilemma between sparsity and performance is to re-train a classifier, this time non-sparse, after the feature selection has been performed with Lasso [15]. Such scheme

works, but at the price of training a second model, thus leading to extra computational effort and to the risk of suboptimal solutions, since we are training a model with the features that were considered optimal by another. In unmixing, synthetic examples also show that the Lasso regularization is not the one leading to the best abundance estimation [42].

In recent years, there has been a trend in the study of unbiased sparse regularizers. These regularizers, typically the ℓ_0, ℓ_q and Log Sum Penalty (LSP [40]), can solve the dilemma between sparsity and performance, but are non-convex and therefore cannot be solved by known off-the-shelf convex optimization tools. Therefore, such regularizers have until now received little attention in the remote sensing community. A handful of papers using ℓ_q norm are found in the field of spectral unmixing [42]–[44] where authors consider nonnegatrive matrix factorization solutions; in the modelling of electromagnetic induction responses, where the model parameters were estimated by regularized least squares estimation [45]; in feature extraction using deconvolutional networks [46] and in structured prediction, where authors use a non-convex sparse classifier to provide posterior probabilities to be used in a graph cut model [47]. In all these studies, the non-convex regularizer outperformed the Lasso, while still providing sparse solutions.

In this paper, we give a critical explanation and theoretical motivations for the success of regularized classification, with a focus on non-convex methods. By comparing it with other traditional regularizers (ridge ℓ_2 and Lasso ℓ_1), we advocate the use of non-convex regularization in remote sensing image processing tasks: non-convex optimization marries the advantages of accuracy and sparsity in a single model, without the need of unbiasing in two steps or reduce the level of sparsity to increase performance. We also provide a freely available toolbox for the interested readers that would like to enter this growing field of investigation.

The reminder of this paper is as follows: in Section II, we present a general framework for regularized remote sensing image processing and discuss different forms of convex and non-convex regularization. We will also present the optimization algorithm proposed. Then, in Section III we apply the proposed non-convex regularizers to the problem of multi- and hyper-spectral image classification and therefore present the specific data term for classification and study it in synthetic and real examples. In Section IV we apply our proposed framework to the problem of linear unmixing, present the specific data term for unmixing and study the behaviour of the different regularizers in simulated examples involving true spectra from the USGS library. Section V concludes the paper.

II. Optimization and non convex regularization

In this Section, we give an intuitive explanation of regularized models. We first introduce the general problem of regularization and then explore convex and non-convex regularization schemes, with a focus on sparsity-inducing regularizers. Finally, we present the optimization algorithms to solve non-convex regularization, with accent put on proximal splitting methods such as GIST [48].

TABLE I
Definition of the regularization terms considered

Regularization term	$g(\lvert w_k\rvert)$
Ridge, ℓ_2 norm	$\lvert w_k\rvert^2$
Lasso, ℓ_1 norm	$\lvert w_k\rvert$
Log sum penalty (LSP)	$\log(\lvert w_k\rvert/\theta+1)$
ℓ_p with $0<p<1$	$\lvert w_k\rvert^p$

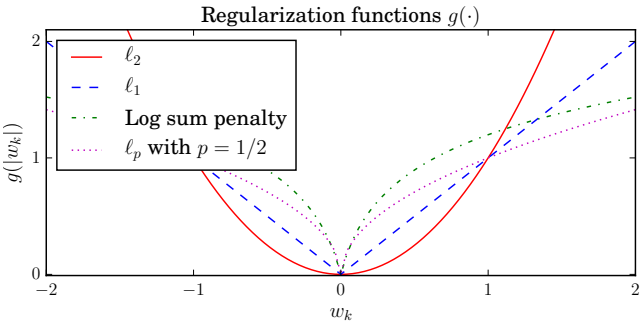

Fig. 1. Illustration of the regularization terms $g(\cdot)$. Note that both ℓ_2 and ℓ_1 regularizations are convex and that log sum penalty and ℓ_p with $p=1/2$ are concave on their positive orthant.

A. Optimization problem

Regularized models address the following optimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} \quad L(\mathbf{w})+\lambda R(\mathbf{w}) \tag{1}$$

where $L(\cdot)$ is a smooth function (Lipschitz gradient), $\lambda > 0$ is a regularization parameter and $R(\cdot)$ is a regularization function. This kind of problem is extremely common in data mining, denoising and parameter estimation.

$L(\cdot)$ is often an empirical loss that measures the discrepancy between a model $\mathbf{w}$ and a dataset containing real life observations.

The regularization term $R(\cdot)$ is added to the optimization problem in order to promote a simple model, which has been shown to lead to a better estimation [49]. All the regularization terms discussed in this work are of the form :

$$R(\mathbf{w})=\sum_k g(|w_k|) \tag{2}$$

where g is a monotonically increasing function. This means that the complexity of the model $\mathbf{w}$ can be expressed as a sum of the complexity of each feature k in the model.

The specific form of the regularizer will change the assumptions made on the model. In the following, we discuss several classes of regularizers of increasing complexity: differentiable, non-differentiable (*i.e.* sparsity inducing) and finally both non-differentiable and non-convex. A summary of all the regularization terms investigated in this work is given in Table I, along with an illustration of the regularization as a function of the value of the coefficient w_k (Fig. 1).

B. Non-sparse regularization

One of the most common regularizers is the square ℓ_2 norm of model $\mathbf{w}$, *i.e.*, $R(\mathbf{w})=\|\mathbf{w}\|^2$ ($g(\cdot)=(\cdot)^2$). This regularization will penalize large values in the vector $\mathbf{w}$ but

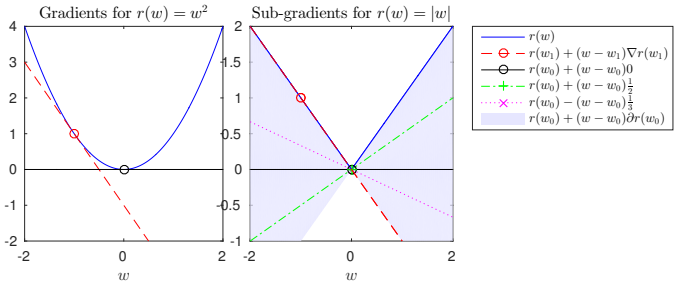

Fig. 2. Illustration of gradients and subgradients on a differentiable ℓ_2 (left) and non-differentiable ℓ_1 (right) function.

is isotropic, *i.e.* it will not promote a given direction for the vector $\mathbf{w}$. This regularization term is also known as ℓ_2, quadratic or ridge regularization and is commonly used in linear regression and classification. For instance, logistic regression is often regularized with a quadratic term. Also note that Support Vector Machine are regularized using the ℓ_2 norm in the Reproducing Kernel Hilbert Space of the form $R(\mathbf{w}) = \mathbf{w}^\top \mathbf{K}\mathbf{w}$ [50].

C. Sparsity promoting regularization

In some cases, not all the features or observations are of interest for the model. In order to get a better estimation, one wants the vector $\mathbf{w}$ to be sparse, *i.e.* to have several components exactly 0. For linear prediction, sparsity in the model $\mathbf{w}$ implies that not all features are used for the prediction[2]. This means that the features showing a non-zero value in w_k are then "selected". Similarly, when estimating a mixture one can suppose that only few materials are present, which again implies sparsity of the abundance coefficients.

In order to promote sparsity in $\mathbf{w}$ one needs to use a regularization term that increases when the number of active component grows. The obvious choice is to use the ℓ_0 pseudo-norm that returns directly the number of non-zero coefficients in $\mathbf{w}$. Nevertheless, the ℓ_0 term is non-convex and non-differentiable, and cannot be optimized exactly unless all the possible subsets are tested. Despite recent works aiming at solving directly this problem via discrete optimization [51], this approach is still computationally impossible even for medium-sized problems. Greedy optimization methods have been proposed to solve this kind of optimization problem and have lead to efficient algorithms such as Orthogonal Matching Pursuit (OMP) [52] or Orthogonal Least Square (OLS) [53]. However, one of the most common approaches to promote sparsity without recurring to the ℓ_0 regularizer is to use the ℓ_1 norm instead. This approach, also known as the Lasso in linear regression, has been widely used in compressed sensing in order to estimate with precision a few component in a large sparse vector.

Now we discuss the intuition why using a regularization term such as ℓ_1 promotes sparsity. The reason behind the sparsity of the ℓ_1 norm lies in the non-differentiability at 0 shown in Fig. 1 (dashed blue line). For the sake of readability, we will suppose here that $R(\cdot)$ is convex, but the intuition is the same and the results can be generalized to the non-convex functions presented in the next section. For a more illustrative example we use a 1D comparison between the ℓ_2 and ℓ_1 regularizers (Fig. 2).

- When both the data and regularization term are differentiable, a stationary point $\mathbf{w}^\star$ has the following property:

$$\nabla L(\mathbf{w}^\star) + \lambda \nabla R(\mathbf{w}^\star) = \mathbf{0}. \tag{3}$$

  In other words, the gradients of both functions have to cancel themselves exactly. This is true for the ℓ_2 regularizer everywhere, but also for the ℓ_1, with the exception of $w_k = 0$. If we consider the ℓ_2 regularizer as an example (left plot in Fig. 2), we see that each point has a specific gradient, corresponding to the tangent to each point (e.g. the red dashed line). The stationary point is reached in this case for $w_k = 0$, as given by the black line in the left plot of Fig. 2.
- When the second term in Eq (3) is not differentiable (as in the ℓ_1 case at 0 presented in the right plot of Fig. 2), the gradient is not unique anymore and one has to use the sub-gradients and sub-differentials. For a convex function $R(\cdot)$ a sub-gradient at $\mathbf{w}^t$ is a vector $\mathbf{x}$ such that $R(\mathbf{w}) \geq \mathbf{x}^\top(\mathbf{w} - \mathbf{w}^t) + R(\mathbf{w}^t)$, *i.e.* it is the slope of a linear function that remains below the function. In 1D, a sub-gradient defines a line touching the function at the non-differentiable point (in the case of Fig. 2, at 0), but stays below the function everywhere else, e.g. the black and green dotted-dash lines in Fig. 2 (right). The sub-differential $\partial R(\mathbf{w}^t)$ is the set of all the sub-gradients that respect the minoration relation above. The sub-differential is illustrated in Fig. 2, by the the area in light blue, which contains all possible solutions.
  Now the optimality constraints cannot rely on equality since the sub-gradient is not unique, which leads to the following optimality condition

$$\mathbf{0} \in \nabla L(\mathbf{w}^\star) + \lambda \partial R(\mathbf{w}^\star) \tag{4}$$

  This is very interesting in our case because this condition is much easier to satisfy than Eq. (3). Indeed, we just need to have a single sub-gradient in the whole sub-differential $\partial R(\cdot)$ that can cancel the gradient $\nabla L(\cdot)$. In other words, only one of the possible straight lines in the blue area is needed to cancel the gradient, thus making the chances for a null coefficient much higher. For instance, when using the ℓ_1 regularization, the sub-differential of variable w_i in 0 is the set $[-\lambda, \lambda]$. When λ becomes large enough it is larger than all the components of the gradient $\nabla L(\cdot)$ and the only solution verifying the conditions is the null vector $\mathbf{0}$.

The ℓ_1 regularization has been largely studied. Because it is convex meaning it avoids the problem of local minima, and many efficient optimization procedures exists to solve it (e.g. LARS [54], Forward Backward Splitting [55]). But the sparsity of the solution using ℓ_1 regularization often comes with a cost in term of generalization. While theoretical studies show that under some constraint the Lasso can recover the true relevant

[2]Note that zero coefficients might happen also in the ℓ_2 solution, but the regularizer itself does not promote their appearence.

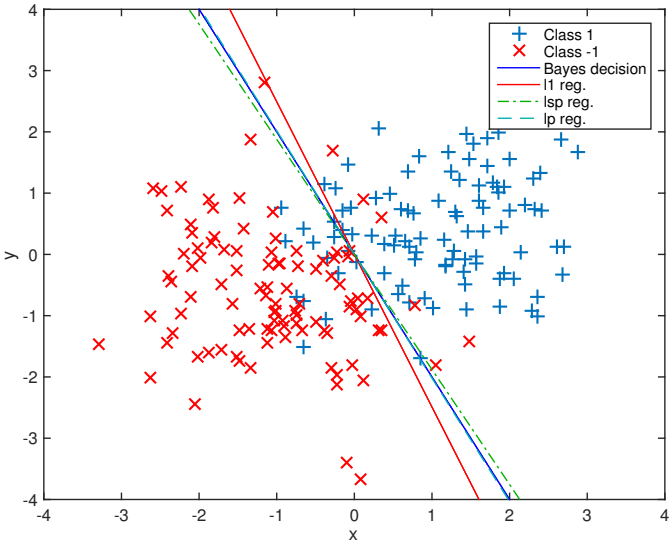

Fig. 3. Example for a 2-class toy example with 2 discriminant features and 18 noisy features. The regularization parameter of each method has been chosen as the minimal value that leads to the correct sparsity with only 2 features selected.

variables and their sign, the solution obtained will be biased toward $\mathbf{0}$ [56]. Figure 3 illustrates the bias in a two-class toy dataset: the ℓ_1 decision function (red line) is biased with respect to the Bayes decision function (blue line). In this case, the bias corresponds to a rotation of the separating hyperplane. In practice, one can deal with this bias by estimating again the model on selected subset of variables using an isotropic norm (e.g. ℓ_2) [15], but this requires to solve again an optimization problem. The approach we propose in this paper is to use a non-convex regularization term that will still promote sparsity, while minimizing the aforementioned bias. To this end, we present non-convex regularization in the next section.

D. Non-convex regularization

In order to promote more sparsity while reducing the bias, several works have looked at non-convex, yet continuous regularization. Such regularizers have been proposed for instance in statistical estimation [57], compressed sensing [40] or in machine learning [41]. Popular examples are the Smoothly Clipped Absolute Deviation (SCAD) [57], the Minimax Concave Penalty (MCP) [58] and the Log Sum Penalty (LSP) [40] considered below (see [48] for more examples). In the following we will investigate two of them in more detail: ℓ_p pseudo-norm with $p = \frac{1}{2}$ and LSP, both also displayed in Figure 1.

All the non-convex regularization above share some particular characteristics that make them of interest in our case. First (and as the ℓ_0 pseudo-norm and ℓ_1 norm) they all have a non-differentiability in $\mathbf{0}$, which – as we have seen in the previous section – promotes sparsity. Second they are all concave in their positive orthant, which limits the bias because their gradient will decrease for large values of w_i limiting the shrinkage (as compared to the ℓ_1 norm, whose gradient for $w_i \neq 0$ is constant). Intuitively, this means that with a non-convex regularization it will become more difficult for large coefficients to be shrinked toward 0, because their gradient is small. On the contrary, the ℓ_1 norm will treat all coefficients equally and apply the same attraction to the stationary point to all of them. The decision functions for the LSP and ℓ_p norms are shown in Fig. 3 and are much closer to the actual (true) Bayes decision function.

E. Optimization algorithms

Thanks to the differentiability of the $L(\cdot)$ term, the optimization problem can be solved using proximal splitting methods [55]. The convergence of those algorithms to a global minimum are well studied in the convex case. For non-convex regularization, recent works have proved that proximal methods can be used with non-convex regularizers when a simple closed form solution of the proximity operator for the regularization can be computed [48]. Recent works have studied the convergence of proximal methods with non-convex regularization and proved convergence to a local stationnary point for a large family of loss functions [59].

In this work, we used the General Iterative Shrinkage and Thresholding (GIST) algorithm proposed in [48]. This approach is a first order method that consists in iteratively linearizing $L(\cdot)$ in order to solve very simple proximal operators at each iteration. At each iteration $t+1$ one computes the model update w^{t+1} by solving

$$\min_{\mathbf{w}} \quad \nabla L(\mathbf{w}^t)^\top(\mathbf{w}-\mathbf{w}^t) + \lambda R(\mathbf{w}) + \frac{\mu}{2}\|\mathbf{w}-\mathbf{w}^t\|_2^2. \quad (5)$$

When μ is a Lipschitz constant of $L(\cdot)$, the cost function above is a majorization of $L(\cdot)+\lambda R(\cdot)$ which ensures a decrease of the objective function at each iteration. Problem (5) can be reformulated as a proximity operator

$$\text{prox}_{\lambda R}(\mathbf{v}) = \arg\min_{\mathbf{w}} \quad \lambda R(\mathbf{w}) + \frac{\mu}{2}\|\mathbf{w}-\mathbf{v}^t\|_2^2, \quad (6)$$

where $\mathbf{v}^t = \mathbf{w}^t - \frac{1}{\mu}\nabla L(\mathbf{w}^t)$ can be seen as a gradient step *w.r.t.* $L(\cdot)$ followed by a proximal operator at each iteration. Note that the efficiency of a proximal algorithm depends on the existence of a simple closed form solution for solving the proximity operator in Eq. (6). Luckily, there exists numerous operators in the convex case (detailed list in [55]) and some non-convex proximal operator can be computed on the regularization used in our work (see [48, Appendix 1] for LSP and [60, Equ. 11] for ℓ_p with $p = 1/2$). Note that efficient methods, which estimate the Hessian matrix [61], [62] exist, as well as a wide range of methods based on DC programming, which have shown to work very well in practice [62], [63] and can handle the general case $p \in (0, 1]$ for the ℓ_p pseudo-norm (see [64] for an implementation).

Finally, when one wants to perform variable selection using the ℓ_0 pseudo-norm as regularization, the exact solution of the combinatorial problem is not always necessary. As mentioned above, greedy optimization methods have been proposed to solve this kind of optimization problem and have lead to efficient algorithms such as Orthogonal Matching Pursuit (OMP) [52] or Orthogonal Least Square (OLS) [53]. In this paper, we won't consider these methods in detail, but they have been shown to perform well on least square minimization problems.

III. Classification with feature selection

In this section, we tackle the problem of sparse classification. Through a toy example and a series of real data experiments, we will study the interest of non convex regularization.

A. Model

The model we will consider in the experiments is a simple linear classifier of the form $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b$ where $\mathbf{w} \in \mathbb{R}^d$ is the normal vector to the separating hyperplane and b is a bias term. In the binary case ($y_i \in [-1; 1]$), the estimation is performed by solving the following regularized optimization problem:

$$\min_{\mathbf{w},b} \quad \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y_i, f(\mathbf{x}_i)) + R(\mathbf{w}), \tag{7}$$

where $R(\mathbf{w})$ is one of the regularizers in Table I and $\mathcal{L}(y_i, f(\mathbf{x}_i))$ is a classification loss that measures the discrepancy between the prediction $f(\mathbf{x}_i)$ and the true label y_i. Hereafter, we will use the squared hinge loss:

$$\mathcal{L}(y_i, f(\mathbf{x}_i)) = \max(0, 1 - y_i f(\mathbf{x}_i))^2.$$

When dealing with multi-class problems, we use a One-Against-All procedure, i.e. we learn one linear function $f_k(\cdot)$ per class k and then predict the final class for a given observed pixel $\mathbf{x}$ as the solution of $\arg\min_k f_k(\mathbf{x})$. In practice, this leads to an optimization problem similar to Eq. (7), where we need to estimate a matrix $\mathbf{W}$, containing the coefficients per each class. The number of coefficients to be estimated is therefore the size d of the input space multiplied by the number of classes C.

B. Toy example

First we consider in detail the toy example in Fig. 3: the data considered are 20-dimensional, where the first two dimensions are discriminative (they correspond to those plotted in Fig. 3), while the other are not (they are generated as Gaussian noise). The correct solution is therefore to assign non zero coefficients to the two discriminative features and $w_k = 0$ for all the others.

Figure 3 show the classifiers estimated for the smallest value of the regularization term λ, which leads to the correct sparsity level (2 features selected). This ensures that we have selected the proper components, while minimizing the bias for all methods. This also illustrates that the ℓ_1 classifier has a stronger bias (i.e. provides a decision function further away from the optimal Bayes classifier) than the classifiers regularized by non-convex functions.

Let's now focus on the effect of the regularization term and of its strength, defined by the regularization parameter λ in Eq. (7). Figure 4 illustrates a regularization path, i.e., all the solutions obtained by increasing the regularization parameter λ[3]. Each line corresponds to one input variable

[3]A "regularization path" for the Lasso is generally computed using homotopy algorithms [65]. However, experiments show that the computational complexity of the complete Lasso path remains high for high-dimensional data. Therefore, in our experiments we used an approximate path (i.e., a discrete sampling of λ values along the path).

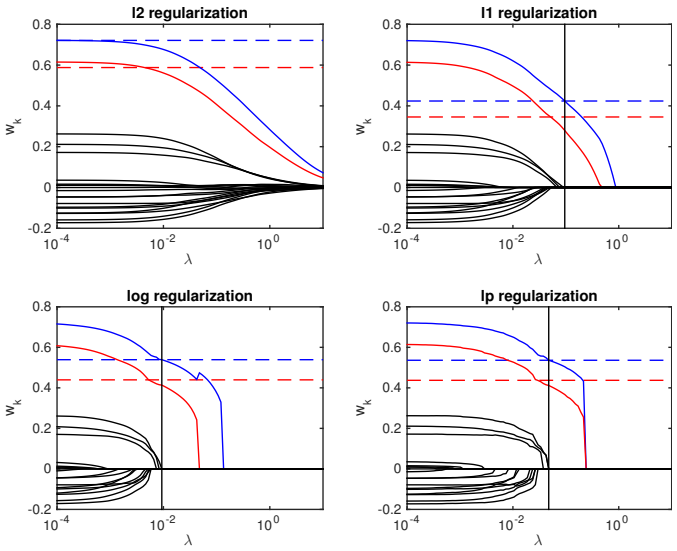

Fig. 4. Regularization paths for the toy example in Fig. 3. Each line corresponds to the coefficients w_k attributed to each feature along the different values of λ. The best fit is met for each regularizer at the black vertical line, where all coefficients but two are 0. The unbiased Bayes classifier coefficients (the correct coefficients) are represented by the horizontal dashed lines.

and those with the largest coefficients (and in color) are the discriminative ones. Considering the ℓ_2 regularization (top left panel in Fig. 4), no sparsity is achieved and, even if the two correct features have the largest coefficients, the solution is not compact. The ℓ_1 solution (top right panel) shows a correct sparse solution for $\lambda = 10^{-1}$ (vertical black line, where all the coefficients but two are 0), but the smallest coefficient is biased (it is smaller than expected by the Bayes classifier, represented by the horizontal dashed lines). The two non-convex regularizers (bottom line of Fig. 4) show the correct features selected, but a smaller bias: the coefficient retrieved are closer to the optimal ones of the Bayes classifier. Moreover, the non zero coefficients stay close to the correct values for a wider set of regularization parameters and then drop directly to zero: this means that the non-convex model either has not enough features to train or has little feature with the right coefficients, contrarily to the ℓ_1 that can retrieve sparse solution with wrong coefficients, as it can be seen in the part to the right of the vertical black line of the ℓ_1 regularization path.

C. Remote sensing images

Data. The real datasets considered are three very high resolution remote sensing images.

1) Thetford Mines. The first dataset is acquired over the Thetford mines site in Québec, Canada and contains two data sources: a VHR color image (three channels, red-green-blue) at 20 cm resolution and a long wave infrared (LWIR, 84 channels) hyperspectral image at approximatively 1 m resolution[4]. The LWIR images are downsampled by a factor 5, to match the resolution of

[4]The data were proposed as the Data Fusion Contest 2014 [66] and are available on the IADF TC website for download http://www.grss-ieee.org/community/technical-committees/data-fusion/

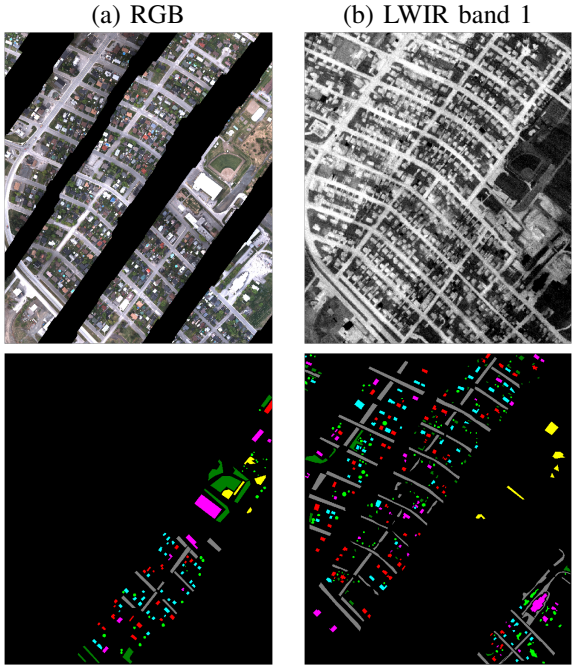

Fig. 5. The THETFORD MINES 2014 dataset used in the classification experiments, along with its labels.

the RGB data, leading to a $(4386 \times 3769 \times 87)$ datacube. The RGB composite, band 1 of the LWIR data and the train / test ground truths are provided in Fig. 5.

2) HOUSTON. The second image is a CASI image acquired over Houston with 144 spectral bands at $2.5m$ resolution. A field survey is also available (14'703 labeled pixels, divided in 14 land use classes). A LiDAR DSM was also available and was used as an additional feature[5]. The CASI image was corrected with histogram matching for a large shadowed part on the right side (as in [27]) and the DSM was detrended by a 3m trend on the left-right direction. Image, DSM and ground truth are illustrated in Fig. 6.
3) ZURICH SUMMER. The third dataset is a series of 20 QuickBird images acquired over the city of Zurich, Switzerland, in August 2002[6]. The data have been pansharpened at 0.6 m spatial resolution and a dense ground truth is provided for each image. Eight classes are depicted: buildings, roads, railway, water, swimming pools, trees, meadows and bare soil. More information on the data can be found in [68]. To reduce computational complexity, we extracted a set of superpixels using the Felzenszwalb algorithm [69], which reduced the number of samples from $\sim 10^6$ pixels per image to a few thousands. An example of the superpixels extracted on image tile #3 is given in Fig. 7.

Setup. For all datasets, contextual features were added to

[5]The data were proposed as the Data Fusion Contest 2013 [67] and are available on the IADF TC website for download http://www.grss-ieee.org/community/technical-committees/data-fusion/

[6]The dataset is freely available at https://sites.google.com/site/michelevolpiresearch/data/zurich-dataset

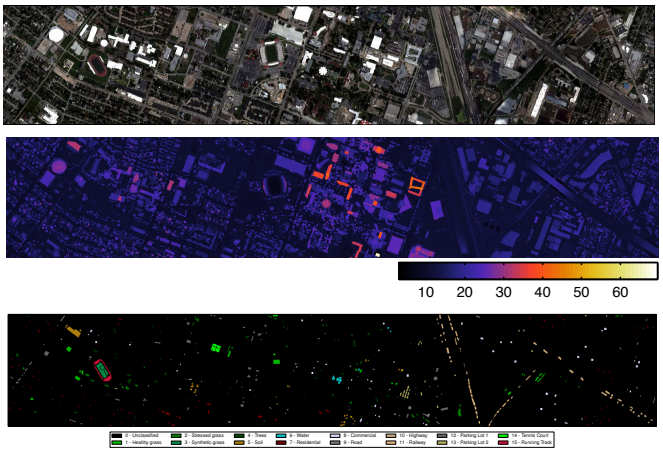

Fig. 6. The HOUSTON dataset used in the classification experiments: (top) true color representation of the hyperspectral image (144 bands); (middle): detrended LiDAR DSM; (bottom) labeled samples (all the available ones, in 15 classes).

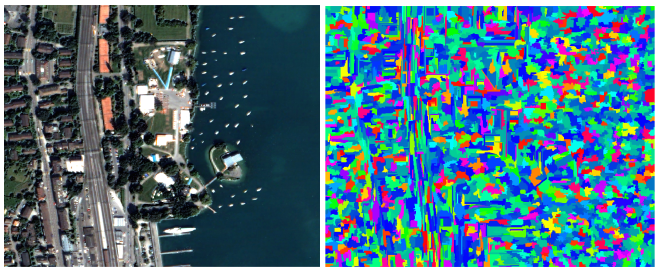

Fig. 7. Example on tile #3 of the superpixels extracted by the Felzenszwalb algorithm [69].

the spectral bands, in order to improve the geometric quality of classification [14]: morphological and texture filters were added, following the list in [15]. Each image was processed to extract the most effective filters for its processing:

- For the THETFORD MINES dataset, the filters were extracted from the RGB image and from a normalized ratio between the red band and the average of the LWIR bands (following the strategy of the winners of the 2014 Data Fusion Contest [66]), which approaches a vegetation index. Given the extremely high resolution of the dataset, the filters were computed with the size range $\{7, ... 23\}$, leading to 100 spatial features.
- For the HOUSTON case, the filters were calculated on both the 3 first principal components projections extracted from the hyperspectral image and the DSM. Given the smaller resolution of this dataset, the convolution sizes of the local filters are in the range $\{3, ..., 15\}$ pixels. This leads to 240 spatial features.
- For the ZURICH SUMMER dataset spatial filters were computed directly on the four spectral bands, plus the NDVI and the NDWI indices. Then, average, minimum, maximum and standard deviation values per superpixel were extracted as feature values. Since the spatial resolution is comparable to the one of the HOUSTON dataset, the same sizes of convolution filters are used, leading to a total of 360 spatial features.

The joint spatial-spectral input space is obtained by stacking

the original images to the spatial filters above. It is therefore of dimension 188 in the THETFORD MINES data, 384 in the HOUSTON data and 366 in the ZURICH SUMMER case.

Regarding the classifier, we considered the linear classifier of Eq. (7) with a squared hinge loss:

- In the THETFORD MINES case, we use 5000 labeled pixels per class. Given the spatial resolution of the image and the 568‘242 labeled points available in the training ground truth, this only represents approximatively 5% of the labeled pixels in the training image. For test, we use the entire test ground truth, which is spatially disconnected to the training one (except for the class ‘soil’, see Fig. 5) and carries 1.5 million labeled pixels.
- In the HOUSTON case, we also proceed with pixel classification. All the models are trained with 60 labeled pixels per class, randomly selected, and all the remaining labeled pixels are considered as the test set. We report performances on the entire test set provided in the Data Fusion contest 2013, which is spatially disconnected from the training set (Fig. 6).
- For the ZURICH SUMMER data, we deal with superpixels and 20 separate images. We used images #1-15 to train the classifier and then tested on the five remaining images (Fig. 8). Given the complexity of the task (not all the images have all the classes and the urban fabrics depicted vary from scene to scene), we used 90% of the available superpixels in the 15 training images, which resulted in 30‘649 superpixels. All the labeled superpixels in the test images (8‘960 superpixels) are used as test set.

Regarding the regularizers, we compare the four regularizers of Tab. I (ℓ_1, ℓ_2, Log sum penalty and ℓ_p with $p = 1/2$) and study the joint behavior of accuracy and sparsity along a regularization path, i.e. for different values of λ: below $\lambda = \{1e^{-5}, \dots, 1e^{-1}\}$, with 18 steps. For each step, the experiment was repeated ten times with different train/test sets (each run with the same training samples for all regularizers) and the average Kappa and number of active coefficients is reported in Fig 9. Also note that we report the total number of coefficients in the multiclass case, $w_{j,k}$, which is equal to the number of features multiplied by the number of classes, plus one additional feature per class (bias term). In total, the model estimates 1‘504 coefficients in the case of the THETFORD MINES data, while for the HOUSTON and ZURICH SUMMER cases it deals with 5‘775 and 3‘294 coefficients, respectively.

Results. The results are reported in Fig. 9, comparing the regularization paths for the four regularizers and the three datasets presented above. The graphs can be read as a ROC curve: the most desirable situation would be a classifier with both accuracy and little active features, i.e., a score close to the top-left corner. The ℓ_2 model shows no variation on the sparsity axis (all the coefficients are active) and very little variability on the accuracy one: it is therefore represented by a single green dot. It is remarkably accurate, but is the less compact model, since it has all the coefficients active. Employing the ℓ_1 regularizer (red line), as it is mainly done in the literature of sparse classification, achieves a sharp decrease in the number of active coefficients, but at the price of a steep decrease in performances of the classifier. When using 100 active coefficients, the ℓ_1 model suffers of a 20% drop in performance, a trend is observed in all the experiments reported.

Using the non-convex regularizers provides the best of both worlds: the ℓ_p regularizer (black line with ‘□’ markers) in particular, but also the Log sum penalty regularizer (blue line with ‘×’ markers) achieve improvements of about 15-20% with respect to the ℓ_1 model. More stable results along the regularization path are observed: the non-convex regularizers are less biased than the ℓ_1 norm in classification and achieve competitive performances with respect to the (non-sparse) ℓ_2 model with a fraction of the features (around 1-2%). Note that the models of all experiments were initialized with the $\mathbf{0}$ vector.This is sensible for the non-convex problem, since all the regularization discussed in the paper (even ℓ_2) tend to shrink the model toward this point. By initializing at $\mathbf{0}$ for non-convex regularization, we simply promote a local solution not too far from this neutral point. In other words one can see the initialization as an additional regularization. Moreover the experiments show that the non-convexity leads to state-of-the-art performance.

IV. SPARSE LINEAR UNMIXING

In this section we express the sparse linear unmixing problem in the same optimization framework as Eq. (7). We discuss the advantage of using non-convex optimization. The performance of the ℓ_2, ℓ_1 and the non convex ℓ_p and LSP regularization terms are then compared on a simulated example using real reflectance spectra (as in [18]).

A. Model

Sparse linear unmixing can be expressed as the following optimization problem

$$\min_{\boldsymbol{\alpha} \geq 0} \quad \frac{1}{2}\|\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda R(\boldsymbol{\alpha}), \tag{8}$$

where $\mathbf{y}$ is a noisy spectrum observed and $\mathbf{D}$ is a matrix containing a dictionary of spectra (typically a spectral library). This formulation adds a positivity constraint to the vector $\boldsymbol{\alpha}$ *w.r.t.* problem (7). In practice, (8) can be reformulated as the following unconstrained optimization problem

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{2}\|\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda R(\boldsymbol{\alpha}) + \imath_{\boldsymbol{\alpha}\geq 0}, \tag{9}$$

where $\imath_{\boldsymbol{\alpha}\geq 0}$ is the indicator function that has value $+\infty$ when one of the component of $\boldsymbol{\alpha}$ is > 0 and value 0 when it is in the positive orthant. By supposing that $\imath_{\boldsymbol{\alpha}\geq 0}$ is equivalent to $\lambda\imath_{\boldsymbol{\alpha}\geq 0}, \forall\lambda > 0$, we can gather the last two terms into $\tilde{R}(\boldsymbol{\alpha}) = R(\boldsymbol{\alpha}) + \imath_{\boldsymbol{\alpha}\geq 0}$, thus leading to a problem similar to Eq. (7). All the optimization procedures discussed above can therefore be used for this reformulation, as long as the proximal operator *w.r.t.* $\tilde{R}(\cdot)$ can be computed efficiently. The proximal operator for all the regularization terms in Table I with additional positivity constraints can be obtained by an orthogonal projection on the positive orthant followed by the proximal of R :

$$\text{prox}_{\lambda R + \imath_{\boldsymbol{\alpha}\geq 0}}(\mathbf{v}) = \text{prox}_{\lambda R}(\max(\mathbf{v}, 0)), \tag{10}$$

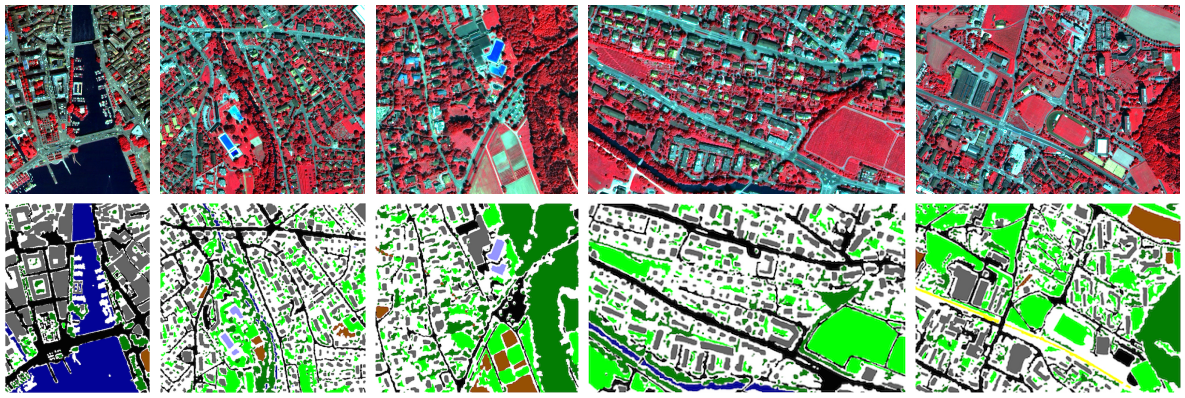

Fig. 8. The five test images of the Zurich Summer dataset (from left to right, tiles #16 to #20), along with their ground truth.

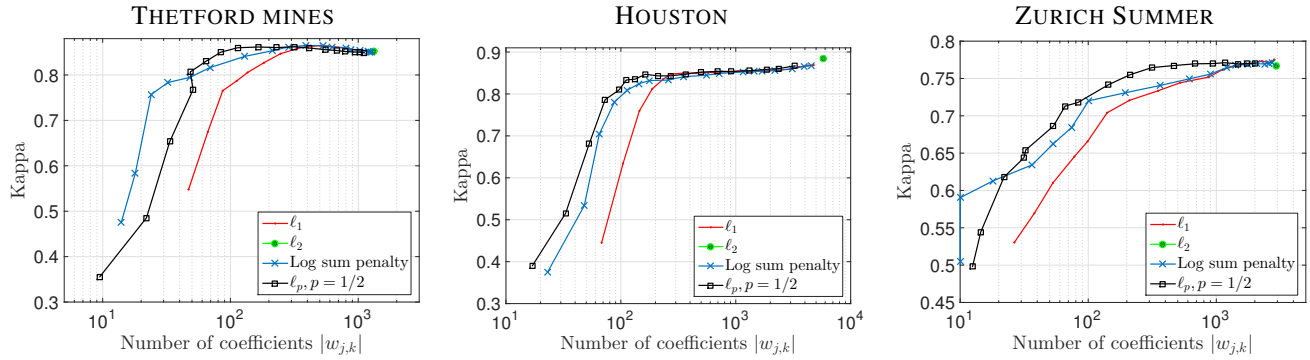

Fig. 9. Performance (Kappa) vs. compactness (number of coefficients $w_{j,k} > 0$) for the different regularizers in the THETFORD MINES, HOUSTON and ZURICH SUMMER datasets.

where $\max(\mathbf{v}, 0)$ is taken component-wise. This shows that we can use the exact same algorithm as in the classification experiments of Section III, since we have an efficient proximal operator.

We know that when the solution of Eq. (8) the resulting $\boldsymbol{\alpha}$ must only have a few nonzero components: one might want to promote more sparsity with a non-differentiable regularization term. Therefore, in the following we investigate the use of non-convex regularization for linear unmixing. We focus on problem (8), but a large part of the unmixing literature works with an additional constraint of sum to 1 for the $\boldsymbol{\alpha}$ coefficients. This additional prior can sometimes reflect a physical measure and adds some information to the optimization problem. In our framework, this constraint can make the direct computation of the proximal operator non-trivial. In this case it is more interesting to use multiple splitting instead of one and to use other algorithms such as generalized FBS [70] or ADMM, that has already been used for remote sensing applications [71].

B. Numerical experiments

In the unmixing application we consider an example simulated using the USGS spectral library[7]: from the library, we extract 23 spectra corresponding to different materials (by keeping spectra with less than 15° angular distance to each other). Using these 23 base spectra, we simulate mixed pixels by creating random linear combinations of $n_{act} \leq 23$ endmembers. The random weight of the active components are obtained using an uniform random generation in $[0, 1]$ (leading to weights that do not sum to 1). We then add to the resulting signatures some Gaussian noise $n \sim \mathcal{N}(0, \sigma^2)$. For each numerical experiments we solve the unmixing problem by least squares with the four regularizers of Table I: ℓ_2, ℓ_1, ℓ_p and LSP. An additional approach that consists in performing a hard thresholding on the positive least square solution (so, the ℓ_2) has also been investigated (named 'LS+threshold' hereafter). As for the previous example on classification, we calculate the unmixing performance on a regularization path, i.e. a series of values of the regularization parameter λ in Eq. (8), with $\lambda = [10^{-5}, ..., 10^3]$. We assess the success of the unmixing by the model error $\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_{true}\|^2$. We repeat the simulation 50 times, to account for different combination of the original elements of the dictionary: all results reported are averages over those 50 simulations.

First, we compare the different regularization schemes for different noise levels (Figure 10). We set $n_{act} = 3$ and report the model error along the regularization path (varying λ) on the top row of Figure 10. On the bottom row, we report the model error as a function of the number of selected components, again along the same regularization path. We observe that the nonconvex strategies achieve the lowest errors (triangle shaped markers) on low and medium noise levels, but also that ℓ_p seems to be more robust to noise. The ℓ_1 norm also achieves good results, in particular in high noise

[7]The dataset can be downloaded from `http://www.lx.it.pt/~bioucas/`

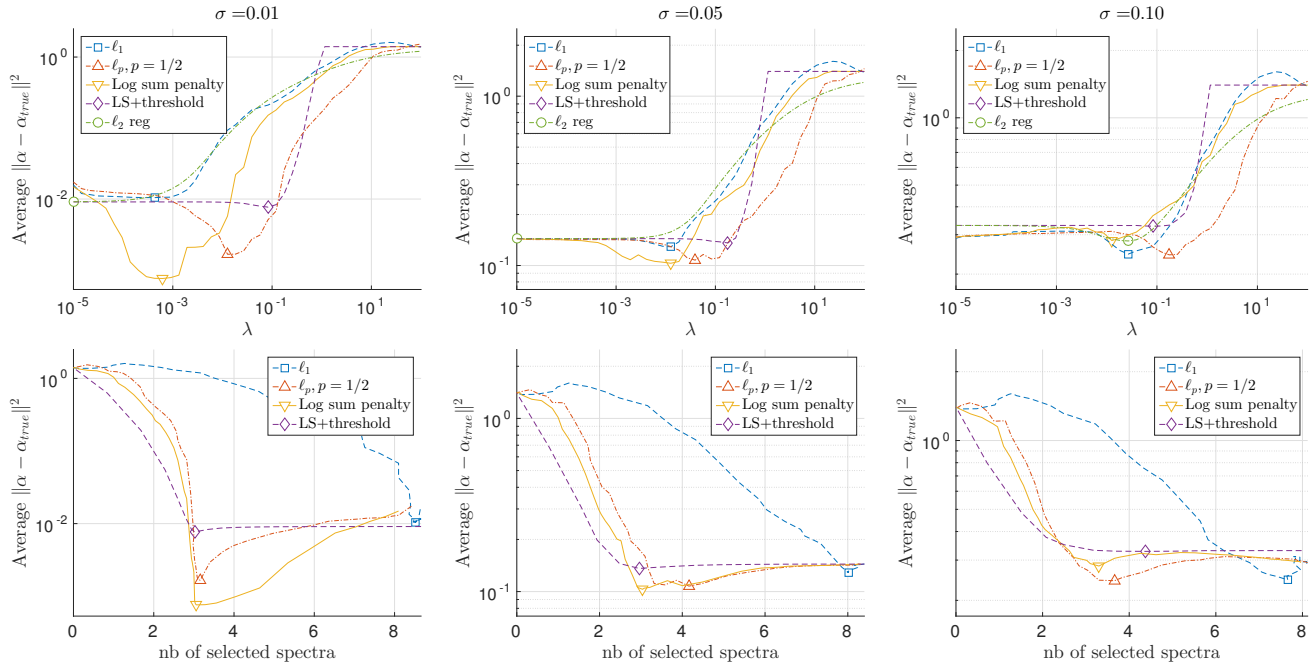

Fig. 10. Linear unmixing results on the simulated hyperspectral dataset. Each column represetns a different noise level: (a) $\sigma = 0.01$ (b) $\sigma = 0.05$ and (c) $\sigma = 0.10$. Model error $\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_{true}\|^2$ is plotted either as a function of the regularization parameter λ (top row) of of the number of active coefficients of the final solution (bottom row). The marker show the best performances of each regularization strategy.

situations. Regarding the error achieved per level of sparsity (represented in the bottom row of Fig. 10) , we observe that the nonconvex regularizers achieve far better reconstruction errors, in particular around the right number of active coefficient (here $n_{act} = 3$). On average, the best results are obtained by the the LSP and ℓ_p regularization. Note that the ℓ_1 regularizer needs a larger number of active component in order to achieve good model reconstruction (of the order of 9 when the actual number of coefficient is 3). The LS+threshold approach seem to work well for component selection, but leads to an important decrease in accuracy of the model.

In order to evaluate the ability of a method to estimate a good model and select the good active components at the same time, we run simulations with a fixed noise level $\sigma = 0.05$ but for a varying number of true active components n_{act}, from 1 to 23. In this configuration, we first find for all regularizations the smallest λ that leads to the correct number of selected component $n_{sel} = n_{act}$. The average model error as a function of n_{act} is reported in Figure 11(a). We can see that the non-convex regularization leads to better performances when the correct number of spectra is selected (compared to ℓ_1 and LS+threshold). In Figure 11(b) we report the number of selected components as a function of the true number of active components when the model error is minimal. We observe that nonconvex regularization manages to both select the correct components and estimate a good model when a small number of components are active ($n_{act} \leq 10$), but also that it fails (as ℓ_1 does) for large numbers of active components. This result illustrates the fact that non-convex regularization is more aggressive in term of sparsity and obviously performs best when sparsity is truly needed.

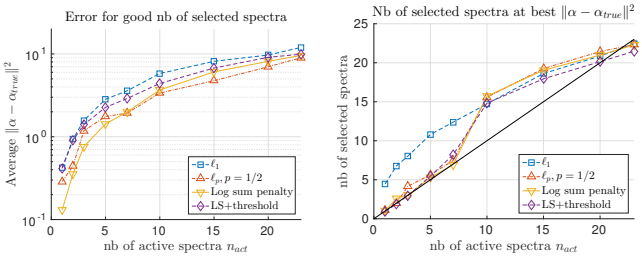

Fig. 11. Linear unmixing results on the simulated hyperspectral dataset for increasing number of active spectra in the mixture: (a) model error, for the best solution with the number of selected spectra closest to n_{act} and (b) number of selected spectra for the model with the lowest error.

V. Conclusions

In this paper, we presented a general framework for non-convex regularization in remote sensing image processing. We discussed different ways to promote sparsity and avoid the bias when sparsity is required via the use of non-convex regularizers. We applied the proposed regularization schemes to problems of hyperspectral image classification and linear unmixing: in all scenarios, we showed that non-convex regularization leads to the best performances when accounting for both sparsity and quality of the final product. Non convex regularizers promote compact solutions, but without the bias (and the decrease in performance) related to nondifferentiable convex norms such as the popular ℓ_1 norm.

Non convex regularization is a flexible and general framework that can be applied to every regularized processing scheme: keeping this in mind, we also provide a toolbox to the community to apply non-convex regularization to a wider

number of problems. The toolbox can now be accessed online (see also the Appendix of this article for a description of the toolbox).

VI. Acknowledgements

The authors would like to thank Telops Inc. (Québec, Canada) for acquiring and providing the THETFORD MINES data, as well as the IEEE GRSS Image Analysis and Data Fusion Technical Committee (IADFTC) and Dr. Michal Shimoni (Signal and Image Centre, Royal Military Academy, Belgium) for organizing the 2014 Data Fusion Contest, the Centre de Recherche Public Gabriel Lippmann (CRPGL, Luxembourg) and Dr. Martin Schlerf (CRPGL) for their contribution of the Hyper-Cam LWIR sensor, and Dr. Michaela De Martino (University of Genoa, Italy) for her contribution to data preparation.

The authors would like to thank the Hyperspectral Image Analysis group and the NSF Funded Center for Airborne Laser Mapping (NCALM) at the University of Houston for providing the HOUSTON data sets and the IEEE GRSS IADFTC for organizing the 2013 Data Fusion Contest.

The authors would like to acknowledge Dr. M. Volpi and Dr. Longbotham for making ther ZURICH SUMMER data available.

The authors would like to acknowledge Dr. Iordache and Dr. Bioucas-Dias for sharing the USGS library used in the unmixing experiment.

Appendix

A. Optimization toolbox

In order to promote the use of non-convex regularization in the remote sensing community, we provide the reader with a simple to use Matlab/Octave generic optimization toolbox. The code will provide a generic solver (complete rewriting of GIST) for problem (7) that is able to handle a number of regularization terms (at least all the terms in Table I) and any differentiable data fitting term L. We provide several function for performing multiclass classification tasks such as SVM, logistic regression and calibrated hinge loss. For linear unmixing we provide the least square loss, but extension to other possibly more robust data fitting terms can be performed easily. For instance, performing unmixing with the more robust Huber loss [72] would require the change of only two lines in function ``gist_least.m'', *i.e.* the computation of the Huber loss and its gradient. The toolbox can now be accessed at `https://github.com/rflamary/nonconvex-optimization`. It is freely available on Github.com as a community project and we welcome contributions.

References

[1] G. Camps-Valls, D. Tuia, L. Gómez-Chova, S. Jimenez, and J. Malo, *Remote Sensing Image Processing*, Synthesis Lectures on Image, Video, and Multimedia Processing. Morgan and Claypool, 2011.

[2] G. Camps-Valls, D. Tuia, L. Bruzzone, and J. Benediktsson, "Advances in hyperspectral image classification: Earth monitoring with statistical learning methods," *IEEE Signal Proc. Mag.*, vol. 31, no. 1, pp. 45–54, 2014.

[3] G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, 2005.

[4] G. Mountrakis, J. Ima, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS J. Photogramm. Rem. Sens.*, vol. 66, no. 3, pp. 247–259, 2011.

[5] J. Verrelst, J. Muñoz-Marí, L. Alonso, J. Delegido, J. P. Rivera, G. Camps-Valls, and J. Moreno, "Machine learning regression algorithms for biophysical parameter retrieval: Opportunities for Sentinel-2 and -3," *Remote Sens. Enviro.*, vol. 118, pp. 127–139, 2012.

[6] M. Belkin, I. Matveeva, and P. Niyogi, "On manifold regularization," in *Proceedings of the Tenth International Workshop on Artificial intellProceedings of the Tenth International Workshop on Artificial Intelligence and Statistics AISTATigence and Statistics (AISTAT)*, Bonn, Germany, 2005, pp. 17–24.

[7] L. Gómez-Chova, G. Camps-Valls, J. Muñoz-Marí, and J. Calpe, "Semisupervised image classification with Laplacian support vector machines," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 4, pp. 336–340, 2008.

[8] M. A. Bencherif, J. Bazi, A. Guessoum, N. Alajlan, F. Melgani, and H. Alhichri, "Fusion of extreme learning machine and graph-based optimization methods for active classification of remote sensing images," *IEEE Geosci. Remote Sens. Letters*, vol. 12, no. 3, pp. 527–531, 2015.

[9] W. Zhangyang, N. Nasrabadi, and T. S. Huang, "Semisupervised hyperspectral classification using task-driven dictionary learning with Laplacian regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1161–1173, 2015.

[10] X. Sun, N. Nasrabadi, and T. Tran, "Task-driven dictionary learning for hyperspectral image classification with structured sparsity constraints," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4457–4471, 2015.

[11] S. T. Tu, J. Y. Chen, W. Yang, and H. Sun, "Laplacian eigenmaps-based polarimetric dimensionality reduction for SAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 1, pp. 170–179, 2012.

[12] D. Tuia, M. Volpi, M. Trolliet, and G. Camps-Valls, "Semisupervised manifold alignment of multimodal remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 12, pp. 7708–7720, 2014.

[13] D. L. Donoho, "Compressed sensing," *IEEE Trans. Info. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[14] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proc. IEEE*, vol. 101, no. 3, pp. 652 – 675, 2013.

[15] D. Tuia, M. Volpi, M. Dalla Mura, A. Rakotomamonjy, and R. Flamary, "Automatic feature learning for spatio-spectral image classification with sparse SVM," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 10, pp. 6062–6074, 2014.

[16] C. Romero, A. und Gatta and G. Camps-Valls, "Unsupervised deep feature extraction for remote sensing image classification," *IEEE Transactions on Geoscience and Remote Sensing*, 2016, in press.

[17] M. Campos-Taberner, A. Romero-Soriano, C. Gatta, G. Camps-Valls, A. Lagrange, B. L. Saux, A. Beaupère, A. Boulch, A. Chan-Hon-Tong, S. Herbin, H. Randrianarivo, M. Ferecatu, M. Shimoni, G. Moser, and D. Tuia, "Processing of extremely high resolution LiDAR and optical data: Outcome of the 2015 IEEE GRSS Data Fusion Contest. Part A: 2D contest," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, in press.

[18] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2014–2039, 2011.

[19] P. Tokarczyk, J. Wegner, S. Walk, and K. Schindler, "Features, color spaces, and boosting: New insights on semantic classification of remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 1, pp. 280–295, 2014.

[20] M. Volpi and D. Tuia, "Dense semantic labeling with convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, in press.

[21] J. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ.*, vol. 5, no. 2, pp. 354–379, April 2012.

[22] Q. Qu, N. Nasrabadi, and T. Tran, "Abundance estimation for bilinear mixture models via joint sparse and low-rank representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 7, pp. 4404–4423, July 2014.

[23] M.-D. Iordache, J. Bioucas-Dias, and A. Plaza, "Collaborative sparse regression for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 341–354, Jan 2014.

[24] Y. Chen, N. Nasrabadi, and T. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, Oct 2011.

[25] K. Tan, S. Zhou, and Q. Du, "Semisupervised discriminant analysis for hyperspectral imagery with block-sparse graph," *IEEE Geosci. Remote Sens. Letters*, vol. 12, no. 8, pp. 1765–1769, Aug 2015.

[26] B. Song, J. Li, M. Dalla Mura, P. Li, A. Plaza, J. Bioucas-Dias, J. Atli Benediktsson, and J. Chanussot, "Remotely sensed image classification using sparse representations of morphological attribute profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 5122–5136, Aug 2014.

[27] D. Tuia, N. Courty, and R. Flamary, "Multiclass feature learning for hyperspectral image classification: sparse and hierarchical solutions," *ISPRS J. Int. Soc. Photo. Remote Sens.*, vol. 105, pp. 272–285, 2015.

[28] X. X. Zhu and R. Bamler, "Super-resolution power and robustness of compressive sensing for spectral estimation with application to spaceborne tomographic SAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 1, pp. 247–258, Jan 2012.

[29] H. Hongxing, J. Bioucas-Dias, and V. Katkovnik, "Interferometric phase image estimation via sparse coding in the complex domain," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2587–2602, May 2015.

[30] S. Li and B. Yang, "A new pan-sharpening method using a compressed sensing technique," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 2, pp. 738–746, Feb 2011.

[31] D. L. Donoho and P. B. Stark, "Uncertainty principles and signal recovery," *SIAM Journal on Applied Mathematics*, vol. 49, no. 3, pp. 906–931, 1989.

[32] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via 1 minimization," *Proceedings of the National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003.

[33] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9, pp. 589–592, 2008.

[34] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.

[35] S. Mosci, L. Rosasco, M. Santoro, A. Verri, and S. Villa, "Solving structured sparsity regularization with proximal methods," in *Machine Learning and Knowledge Discovery in Databases*, pp. 418–433. Springer, 2010.

[36] P. L. Combettes and J.-C. Pesquet, "Proximal thresholding algorithm for minimization over orthonormal bases," *SIAM Journal on Optimization*, vol. 18, no. 4, pp. 1351–1376, 2007.

[37] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[38] T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu, "The entire regularization path for the support vector machine," *Journal of Machine Learning Research*, vol. 5, pp. 1391–1415, 2004.

[39] D. L. Donoho and B. F. Logan, "Signal recovery and the large sieve," *SIAM Journal on Applied Mathematics*, vol. 52, no. 2, pp. 577–591, 1992.

[40] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier analysis and applications*, vol. 14, no. 5-6, pp. 877–905, 2008.

[41] T. Zhang, "Analysis of multi-stage convex relaxation for sparse regularization," *J. Mach. Learn. Res.*, vol. 11, pp. 1081–1107, 2010.

[42] J. Sigurdsson, M. Ulfarsson, and J. Sveinsson, "Hyperspectral unmixing with ℓ_q regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 6793–6806, Nov 2014.

[43] Y. Qian, S. Jia, J. Zhou, and A. Robles-Kelly, "Hyperspectral unmixing via $\ell_{1/2}$ sparsity-constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4282–4297, Nov 2011.

[44] W. Wang and Y. Qian, "Adaptive $\ell_{1/2}$ sparsity-constrained nmf with half-thresholding algorithm for hyperspectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ.*, vol. 8, no. 6, pp. 2618–2631, June 2015.

[45] M.-H. Wei, J. McClellan, and W. Scott, "Estimation of the discrete spectrum of relaxations for electromagnetic induction responses using ℓ_p-regularized least squares for $0 < p < 1$," *IEEE Geosci. Remote Sens. Letters*, vol. 8, no. 2, pp. 233–237, March 2011.

[46] J. Zhang, P. Zhong, Y. Chen, and S. Li, "$\ell_{1/2}$ -regularized deconvolution network for the representation and restoration of optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2617–2627, May 2014.

[47] S. Jia, X. Zhang, and Q. Li, "Spectral-spatial hyperspectral image classification using $\ell_{1/2}$ regularized low-rank representation and sparse representation-based graph cuts," *IEEE J. Sel. Topics Appl. Earth Observ.*, vol. 8, no. 6, pp. 2473–2484, June 2015.

[48] P. Gong, C. Zhang, Z. Lu, J. Z. Huang, and J. Ye, "A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems," in *Proc. ICML*, Atlanta, GE, 2013.

[49] O. Bousquet and A. Elisseeff, "Stability and generalization," *The Journal of Machine Learning Research*, vol. 2, pp. 499–526, 2002.

[50] B. Schölkopf, C. J. Burges, and A. J. Smola, *Advances in kernel methods: support vector learning*, MIT press, 1998.

[51] S. Bourguignon, J. Ninin, H. Carfantan, and M. Mongeau, "Exact sparse approximation problems via mixed-integer programming: Formulations and computational performance," *IEEE Trans. Signal Proc.*, vol. 64, no. 6, pp. 1405–1419, March 2016.

[52] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*. IEEE, 1993, pp. 40–44.

[53] S. Chen, C. F. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *Neural Networks, IEEE Transactions on*, vol. 2, no. 2, pp. 302–309, 1991.

[54] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, et al., "Least angle regression," *The Annals of statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[55] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*, pp. 185–212. Springer, 2011.

[56] H. Zou, "The adaptive Lasso and its oracle properties," *J. American Stat. Assoc.*, vol. 101, no. 476, pp. 1418–1429, 2006.

[57] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *J. American Stat. Assoc.*, vol. 96, no. 456, pp. 1348–1360, 2001.

[58] C. Zhang, "Nearly unbiased variable selection under minimax concave penalty," *Annals of Statistics*, vol. 38, no. 2, pp. 894–942, 2010.

[59] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, "Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the kurdyka-lojasiewicz inequality," *Mathematics of Operations Research*, vol. 35, no. 2, pp. 438–457, 2010.

[60] Z. Xu, X. Chang, F. Xu, and H. Zhang, "$L_{1/2}$ regularization: a thresholding representation theory and a fast solver," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 23, no. 7, pp. 1013–1027, 2012.

[61] E. Chouzenoux, J.-C. Pesquet, and A. Repetti, "A block coordinate variable metric forward-backward algorithm," 2013.

[62] A. Rakotomamonjy, R. Flamary, and G. Gasso, "Dc proximal newton for non-convex optimization problems," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 27, no. 3, pp. 636–647, 2016.

[63] L. Laporte, R. Flamary, S. Canu, S. Djean, and J. Mothe, "Nonconvex regularizations for feature selection in ranking with sparse svm," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 25, no. 6, pp. 1118–1130, 2014.

[64] G. Gasso, A. Rakotomamonjy, and S. Canu, "Recovering sparse signals with a certain family of nonconvex penalties and dc programming," *EEE Trans. Signal Proc.*, vol. 57, no. 12, pp. 4686–4698, 2009.

[65] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization path for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, pp. 1–122, 2010.

[66] W. Liao, X. Huang, F. Van Collie, A. Gautama, W. Philips, H. Liu, T. Zhu, M. Shimoni, G. Moser, and D. Tuia, "Processing of thermal hyperspectral and digital color cameras: outcome of the 2014 data fusion contest," *IEEE J. Sel. Topics Appl. Earth Observ.*, vol. 8, no. 6, pp. 2984–2996, 2015.

[67] F. Pacifici, Q. Du, and S. Prasad, "Report on the 2013 IEEE GRSS data fusion contest: Fusion of hyperspectral and LiDAR data," *IEEE Remote Sens. Mag.*, vol. 1, no. 3, pp. 36–38, 2013.

[68] M. Volpi and V. Ferrari, "Semantic segmentation of urban scenes by learning local class interactions," in *IEEE/CVF CVPRW Earthvision*, 2015.

[69] P. Felzenszwalb and D. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, vol. 59, no. 2, pp. 167–181, 2004.

[70] H. Raguet, J. Fadili, and G. Peyré, "A generalized forward-backward splitting," *SIAM Journal on Imaging Sciences*, vol. 6, no. 3, pp. 1199–1226, 2013.

[71] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Total variation spatial regularization for sparse hyperspectral unmixing," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 50, no. 11, pp. 4484–4502, 2012.

[72] P. J. Huber et al., "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964.