

# Proximal Methods for Sparse Hierarchical Dictionary Learning: Supplementary Materials

## A Proximal Operator for the Tree-Structured Norm $\Omega$

Given a vector  $\mathbf{u} \in \mathbb{R}^p$  and a set  $\mathcal{G}$  of (possibly overlapping) groups of variables  $\{g\}_{g \in \mathcal{G}}$ , we are interested in solving the following problem

$$\min_{\mathbf{v} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{u} - \mathbf{v}\|_2^2 + \lambda \sum_{g \in \mathcal{G}} w_g \|\mathbf{v}_{|g}\|, \quad (1)$$

where  $\mathbf{v}_{|g}$  is the vector whose coordinates are equal to those of  $\mathbf{v}$  for indices in the set  $g$ , and 0 otherwise,  $(w_g)_{g \in \mathcal{G}}$  are a set of positive weights, and  $\lambda \geq 0$  is the regularization parameter. Let us denote by  $\Omega$  the norm

$$\Omega(\mathbf{v}) \triangleq \sum_{g \in \mathcal{G}} w_g \|\mathbf{v}_{|g}\|.$$

The function that maps a vector  $\mathbf{u}$  in  $\mathbb{R}^p$  to the solution of Eq. (1) is referred to as the *proximal operator* of  $\lambda\Omega$ .

We now present an algorithm showing that, even though we do not have a closed form for this problem, when the groups are tree-structured, where tree-structured is meant according to the following definition:

**Definition 1 (Tree-structured set of groups.)** *A set of groups  $\mathcal{G} = \{g\}_{g \in \mathcal{G}}$  is said to be tree-structured in  $\{1, \dots, p\}$ , if  $\bigcup_{g \in \mathcal{G}} g = \{1, \dots, p\}$  and for all  $g, h \in \mathcal{G}$ ,  $(g \cap h \neq \emptyset) \Rightarrow (g \subseteq h \text{ or } h \subseteq g)$ . For such a set of groups, there exists a (non-unique) total order relation  $\preceq$  such that:*

$$g \preceq h \Rightarrow \{g \subseteq h \text{ or } g \cap h = \emptyset\}.$$

We show in this paper that Eq. (1) can be solved *exactly* in linear time. The procedure aims at solving a dual formulation of Eq. (1) involving the dual norm of  $\|\cdot\|$  denoted by  $\|\cdot\|_*$  and defined as  $\|\boldsymbol{\kappa}\|_* = \max_{\|\mathbf{z}\| \leq 1} \mathbf{z}^\top \boldsymbol{\kappa}$  for any vector  $\boldsymbol{\kappa}$  in  $\mathbb{R}^p$ .<sup>1</sup> We first derive a dual problem based on conic duality [Boyd and Vandenberghe, 2004]. The rationale for using conic duality is to come up with a dual problem without overlapping variables.

<sup>1</sup>It is easy to show that the dual norm of the  $\ell_2$  norm is the  $\ell_2$  norm. The dual norm of the  $\ell_\infty$  is the  $\ell_1$ -norm. More generally, the dual norm of the  $\ell_q$ -norm,  $q > 1$  is the  $\ell_{q'}$ -norm, with  $1/q' + 1/q = 1$  (see Boyd and Vandenberghe, 2004).

**Lemma 1 (Dual of the proximal problem)**

Let  $\mathbf{u} \in \mathbb{R}^p$  and let us consider the problem

$$\begin{aligned} \max_{\boldsymbol{\xi} \in \mathbb{R}^{p \times |\mathcal{G}|}} & -\frac{1}{2} \left( \|\mathbf{u} - \sum_{g \in \mathcal{G}} \boldsymbol{\xi}^g\|_2^2 - \|\mathbf{u}\|_2^2 \right) \\ \text{s.t. } & \forall g \in \mathcal{G}, \|\boldsymbol{\xi}^g\|_* \leq \lambda w_g \text{ and } \boldsymbol{\xi}_j^g = 0 \text{ if } j \notin g, \end{aligned} \quad (2)$$

where  $\boldsymbol{\xi} = (\boldsymbol{\xi}^g)_{g \in \mathcal{G}}$  and  $\boldsymbol{\xi}_j^g$  denotes the  $j$ -th coordinate of the vector  $\boldsymbol{\xi}^g$  in  $\mathbb{R}^p$ . Then, problems (1) and (2) are dual to each other and strong duality holds. In addition, the pair of primal-dual variables  $\{\mathbf{v}, \boldsymbol{\xi}\}$  is optimal if and only if  $\boldsymbol{\xi}$  is a feasible point of the optimization problem (2) and

$$\begin{aligned} \mathbf{v} &= \mathbf{u} - \sum_{g \in \mathcal{G}} \boldsymbol{\xi}^g, \\ \text{and for all } g \in \mathcal{G}, & \begin{cases} \mathbf{v}_{|g}^\top \boldsymbol{\xi}^g = \|\mathbf{v}_{|g}\| \|\boldsymbol{\xi}^g\|_* \text{ and } \|\boldsymbol{\xi}^g\|_* = \lambda w_g, \\ \text{or } \mathbf{v}_{|g} = 0. \end{cases} \end{aligned} \quad (3)$$

**Proof.** The proof relies on tools from conic duality [Boyd and Vandenberghe, 2004]. We can equivalently rewrite problem (1) as

$$\min_{\mathbf{v} \in \mathbb{R}^p, \mathbf{z} \in \mathbb{R}^{|\mathcal{G}|}} \frac{1}{2} \|\mathbf{u} - \mathbf{v}\|_2^2 + \lambda \sum_{g \in \mathcal{G}} w_g z_g, \text{ such that } \|\mathbf{v}_{|g}\| \leq z_g, \forall g \in \mathcal{G},$$

by introducing the primal variables  $\mathbf{z} = (z_g)_{g \in \mathcal{G}} \in \mathbb{R}^{|\mathcal{G}|}$ , with the additional  $|\mathcal{G}|$  conic constraints  $\|\mathbf{v}_{|g}\| \leq z_g$ ,  $g \in \mathcal{G}$ .

This primal problem is convex and satisfies Slater's conditions for generalized conic inequalities (i.e., existence of a feasible point in the interior of the domain), which implies that strong duality holds [Boyd and Vandenberghe, 2004]. We now consider the Lagrangian  $\mathcal{L}$  defined as

$$\mathcal{L}(\mathbf{v}, \mathbf{z}, \boldsymbol{\tau}, \boldsymbol{\xi}) = \frac{1}{2} \|\mathbf{u} - \mathbf{v}\|_2^2 + \lambda \sum_{g \in \mathcal{G}} w_g z_g - \sum_{g \in \mathcal{G}} \begin{pmatrix} z_g \\ \mathbf{v}_{|g} \end{pmatrix}^\top \begin{pmatrix} \tau_g \\ \boldsymbol{\xi}^g \end{pmatrix},$$

with the dual variables  $\boldsymbol{\tau} = (\tau_g)_{g \in \mathcal{G}}$  in  $\mathbb{R}^{|\mathcal{G}|}$ , and  $\boldsymbol{\xi} = (\boldsymbol{\xi}^g)_{g \in \mathcal{G}}$  in  $\mathbb{R}^{p \times |\mathcal{G}|}$ , such that for all  $g \in \mathcal{G}$ ,  $\boldsymbol{\xi}_j^g = 0$  if  $j \notin g$  and  $\|\boldsymbol{\xi}^g\|_* \leq \tau_g$ .

The dual function is obtained by taking the derivatives of  $\mathcal{L}$  with respect to the primal variables  $\mathbf{v}$  and  $\mathbf{z}$  and equating them to zero, which leads to

$$\begin{aligned} \mathbf{v} - \mathbf{u} - \sum_{g \in \mathcal{G}} \boldsymbol{\xi}^g &= 0, \\ \forall g \in \mathcal{G}, \quad \lambda w_g - \tau_g &= 0. \end{aligned}$$

After simplifying the Lagrangian and flipping the sign of  $\boldsymbol{\xi}$ , we obtain the dual problem in Eq. (2). As far as the optimality conditions are concerned, they are derived from the

Karush–Kuhn–Tucker conditions for generalized conic inequalities [Boyd and Vandenberghe, 2004]. We have that  $\{\mathbf{v}, \mathbf{z}, \boldsymbol{\tau}, \boldsymbol{\xi}\}$  are optimal if and only if

$$\begin{aligned} \mathbf{v} - \mathbf{u} + \sum_{g \in \mathcal{G}} \boldsymbol{\xi}^g &= 0, \\ \forall g \in \mathcal{G}, \quad \lambda w_g - \tau_g &= 0, \\ \forall g \in \mathcal{G}, \quad z_g \tau_g - \mathbf{v}_{|g}^\top \boldsymbol{\xi}^g &= 0, \quad (\text{Complementary slackness}) \\ \forall g \in \mathcal{G}, \quad \|\mathbf{v}_{|g}\| &\leq z_g, \\ \forall g \in \mathcal{G}, \quad \|\boldsymbol{\xi}^g\|_* &\leq \tau_g. \end{aligned}$$

Combining the complementary slackness with the definition of the dual norm, we have

$$\forall g \in \mathcal{G}, \quad z_g \tau_g = \mathbf{v}_{|g}^\top \boldsymbol{\xi}^g \leq \|\mathbf{v}_{|g}\| \|\boldsymbol{\xi}^g\|_*.$$

Furthermore, using the fact that  $\forall g \in \mathcal{G}$ ,  $\|\mathbf{v}_{|g}\| \leq z_g$  and  $\|\boldsymbol{\xi}^g\|_* \leq \tau_g = \lambda w_g$ , we obtain the following chain of inequalities

$$\forall g \in \mathcal{G}, \quad \lambda z_g w_g = \mathbf{v}_{|g}^\top \boldsymbol{\xi}^g \leq \|\mathbf{v}_{|g}\| \|\boldsymbol{\xi}^g\|_* \leq z_g \|\boldsymbol{\xi}^g\|_* \leq \lambda z_g w_g,$$

for which equality must hold. We notably have

$$\begin{cases} \mathbf{v}_{|g}^\top \boldsymbol{\xi}^g = \|\mathbf{v}_{|g}\| \|\boldsymbol{\xi}^g\|_*, \\ z_g \|\boldsymbol{\xi}^g\|_* = \lambda z_g w_g. \end{cases}$$

If  $\mathbf{v}_{|g} \neq 0$ , then  $z_g$  cannot be equal to zero, which implies in turn that  $\|\boldsymbol{\xi}^g\|_* = \lambda w_g$ .

Reciprocally, starting from the optimality conditions of Lemma 1, we can derive the Karush–Kuhn–Tucker conditions displayed above. More precisely, we define for all  $g \in \mathcal{G}$ ,

$$\tau_g \triangleq \lambda w_g \quad \text{and} \quad z_g \triangleq \|\mathbf{v}_{|g}\|.$$

The only condition that needs to be discussed is the complementary slackness. If  $\mathbf{v}_{|g} = 0$ , then it is easily satisfied. Otherwise, combining the definitions of  $\tau_g$ ,  $z_g$  and the fact that

$$\mathbf{v}_{|g}^\top \boldsymbol{\xi}^g = \|\mathbf{v}_{|g}\| \|\boldsymbol{\xi}^g\|_* \quad \text{and} \quad \|\boldsymbol{\xi}^g\|_* = \lambda w_g,$$

we end up with the desired complementary slackness.  $\blacksquare$

After removing the constant terms, the dual problem can be rewritten as

$$\min_{\boldsymbol{\xi} \in \mathbb{R}^p \times |\mathcal{G}|} \frac{1}{2} \|\mathbf{u} - \sum_{g \in \mathcal{G}} \boldsymbol{\xi}^g\|_2^2 \quad \text{s.t.} \quad \forall g \in \mathcal{G}, \quad \|\boldsymbol{\xi}^g\|_* \leq \lambda w_g \quad \text{and} \quad \boldsymbol{\xi}_j^g = 0 \quad \text{if } j \notin g. \quad (4)$$

The structure of the dual problem, i.e., the separability of the constraints for each  $\boldsymbol{\xi}^g$ ,  $g \in \mathcal{G}$ , makes it possible to solve (2) by block coordinate ascent [Bertsekas, 1999]. The proximal operator defined in Eq. (1) has to be solved *exactly* in order to enjoy the properties of proximal methods. This can not be guaranteed in general by Algorithm 1 in a finite number of iterations. However, we next prove that it is actually possible to compute *exactly* the proximal operator in only *one pass* of Algorithm 1 when  $\|\cdot\|$  is the

---

**Algorithm 1** Block coordinate ascent in the dual
 

---

Inputs:  $\mathbf{u} \in \mathbb{R}^p$  and set of groups  $\mathcal{G}$ .  
 Outputs:  $(\mathbf{v}, \boldsymbol{\xi})$  (primal-dual solutions).  
 Initialization:  $\mathbf{v} = \mathbf{u}, \boldsymbol{\xi} = 0$ .  
**while** ( *maximum number of iterations not reached* ) **do**  
   **for**  $g \in \mathcal{G}$  **do**  
      $\mathbf{v} \leftarrow \mathbf{u} - \sum_{h \neq g} \boldsymbol{\xi}^h$ .  
      $\boldsymbol{\xi}^g \leftarrow \Pi_{\lambda w_g}^*(\mathbf{v}|_g)$ .  
   **end for**  
**end while**  
 $\mathbf{v} \leftarrow \mathbf{u} - \sum_{g \in \mathcal{G}} \boldsymbol{\xi}^g$ .

---

$\ell_2$  or  $\ell_\infty$  norm, provided that the nested groups in  $\mathcal{G}$  are appropriately ordered. This result constitutes the main technical contribution of the paper.

Before stating this result, we need to introduce two lemmas. The first lemma characterizes the projections onto norm balls.

**Lemma 2 (Projection on the dual ball)**

Let  $\mathbf{v} \in \mathbb{R}^p$  and  $t > 0$ . We have

$$\boldsymbol{\kappa} = \Pi_t^*(\mathbf{v})$$

$$\text{if and only if } \begin{cases} \text{if } \|\mathbf{v}\|_* \leq t, & \boldsymbol{\kappa} = \mathbf{v}, \\ \text{otherwise,} & \|\boldsymbol{\kappa}\|_* = t \text{ and } \boldsymbol{\kappa}^\top(\mathbf{v} - \boldsymbol{\kappa}) = \|\boldsymbol{\kappa}\|_* \|\mathbf{v} - \boldsymbol{\kappa}\|. \end{cases}$$

**Proof.** When the vector  $\mathbf{v}$  is already in the ball of  $\|\cdot\|_*$  with radius  $t$ , i.e.,  $\|\mathbf{v}\|_* \leq t$ , the situation is simple, since the projection  $\Pi_t^*(\mathbf{v})$  obviously gives  $\mathbf{v}$  itself. On the other hand, a necessary and sufficient optimality condition for having

$$\boldsymbol{\kappa} = \Pi_t^*(\mathbf{v}) = \arg \min_{\|\mathbf{y}\|_* \leq t} \|\mathbf{v} - \mathbf{y}\|_2$$

is that the residual  $\mathbf{v} - \boldsymbol{\kappa}$  lies in the normal cone of the constraint set [Borwein and Lewis, 2006], that is, for all  $\mathbf{y}$  such that  $\|\mathbf{y}\|_* \leq t$ ,  $(\mathbf{v} - \boldsymbol{\kappa})^\top(\mathbf{y} - \boldsymbol{\kappa}) \leq 0$ . The displayed result then follows from the definition of the dual norm, namely  $\|\boldsymbol{\kappa}\|_* = \max_{\|\mathbf{z}\| \leq 1} \mathbf{z}^\top \boldsymbol{\kappa}$ . ■

The second lemma shows that, given two nested groups  $g, h$  such that  $g \subseteq h \subseteq \{1, \dots, p\}$ , if  $\boldsymbol{\xi}^g$  is updated before  $\boldsymbol{\xi}^h$  in Algorithm 1, then the optimality condition of  $\boldsymbol{\xi}^g$  is not perturbed by the update of  $\boldsymbol{\xi}^h$ . In other words, this lemma indicates that the correct order to consider in Algorithm 1 is  $\preceq$  (see Definition 1).

**Lemma 3 (Projections with nested groups)**

Let  $\|\cdot\|$  denote either the  $\ell_2$  or  $\ell_\infty$  norm, and  $g$  and  $h$  be two nested groups—that is,  $g \subseteq h \subseteq \{1, \dots, p\}$ . Let  $\mathbf{v}$  be a vector in  $\mathbb{R}^p$ , and let us consider the successive projections

$$\boldsymbol{\kappa}^g \triangleq \Pi_{t_g}^*(\mathbf{v}|_g) \text{ and } \boldsymbol{\kappa}^h \triangleq \Pi_{t_h}^*(\mathbf{v}|_h - \boldsymbol{\kappa}^g)$$

with  $t_g, t_h > 0$ . Then, we have as well

$$\boldsymbol{\kappa}^g = \Pi_{t_g}^*(\mathbf{v}_{|g} - \boldsymbol{\kappa}_{|g}^h).$$

**Proof.** The proof mostly relies on the optimality condition derived in Lemma 2. We thus have to prove that either

$$\boldsymbol{\kappa}^g = \mathbf{v}_{|g} - \boldsymbol{\kappa}_{|g}^h, \text{ if } \|\mathbf{v}_{|g} - \boldsymbol{\kappa}_{|g}^h\|_* \leq t_g,$$

or

$$\|\boldsymbol{\kappa}^g\|_* = t_g \text{ and } \boldsymbol{\kappa}^{g\top}(\mathbf{v}_{|g} - \boldsymbol{\kappa}_{|g}^h - \boldsymbol{\kappa}^g) = \|\boldsymbol{\kappa}^g\|_* \|\mathbf{v}_{|g} - \boldsymbol{\kappa}_{|g}^h - \boldsymbol{\kappa}^g\|.$$

Note that the feasibility of  $\boldsymbol{\kappa}^g$ , i.e.,  $\|\boldsymbol{\kappa}^g\|_* \leq t_g$ , is one of the hypothesis in the Lemma.

Let us first assume that  $\|\boldsymbol{\kappa}^g\|_* < t_g$ . We necessarily have that  $\mathbf{v}_{|g}$  also lies in the interior of the ball of  $\|\cdot\|_*$  with radius  $t_g$ , and it holds that  $\boldsymbol{\kappa}^g = \mathbf{v}_{|g}$ . Since  $g \subseteq h$ , we have that the vector  $\mathbf{v}_{|h} - \boldsymbol{\kappa}^g = \mathbf{v}_{|h} - \mathbf{v}_{|g}$  has only zero entries on  $g$ . As a result,  $\boldsymbol{\kappa}_{|g}^h = 0$  and we obtain

$$\boldsymbol{\kappa}^g = \mathbf{v}_{|g} = \mathbf{v}_{|g} - \boldsymbol{\kappa}_{|g}^h,$$

which is the desired conclusion. From now on, we assume that  $\|\boldsymbol{\kappa}^g\|_* = t_g$ . It then remains to show that

$$\boldsymbol{\kappa}^{g\top}(\mathbf{v}_{|g} - \boldsymbol{\kappa}_{|g}^h - \boldsymbol{\kappa}^g) = \|\boldsymbol{\kappa}^g\|_* \|\mathbf{v}_{|g} - \boldsymbol{\kappa}_{|g}^h - \boldsymbol{\kappa}^g\|.$$

We now distinguish the proof, depending on the used norm.

**$\ell_2$  norm:** in the particular case of the  $\ell_2$  norm, the optimality condition for the projection amounts to check when equality holds in the Cauchy-Schwartz inequality, i.e., when the vectors have same signs and are linearly dependent. Thus, there exists  $\rho_g, \rho_h > 0$  such that  $\rho_g \boldsymbol{\kappa}^g = \mathbf{v}_{|g} - \boldsymbol{\kappa}^g$  and  $\rho_h \boldsymbol{\kappa}^h = \mathbf{v}_{|h} - \boldsymbol{\kappa}^g - \boldsymbol{\kappa}^h$ .

Note that the case  $\rho_h = 0$  leads to  $\mathbf{v}_{|h} - \boldsymbol{\kappa}^g - \boldsymbol{\kappa}^h = 0$ , and therefore  $\mathbf{v}_{|g} - \boldsymbol{\kappa}^g - \boldsymbol{\kappa}_{|g}^h = 0$  since  $g \subseteq h$ , which directly gives the result. The case  $\rho_g = 0$  implies  $\mathbf{v}_{|g} - \boldsymbol{\kappa}^g = 0$  and therefore  $\boldsymbol{\kappa}_{|g}^h = 0$ , giving the result as well. We can therefore assume now  $\rho_h > 0$  and  $\rho_g > 0$ . Some algebra leads to

$$\boldsymbol{\kappa}^g = \frac{\rho_h + 1}{\rho_g \rho_h} (\mathbf{v}_{|g} - \boldsymbol{\kappa}^g - \boldsymbol{\kappa}_{|g}^h),$$

and consequently

$$\boldsymbol{\kappa}^{g\top}(\mathbf{v}_{|g} - \boldsymbol{\kappa}^g - \boldsymbol{\kappa}_{|g}^h) = \|\boldsymbol{\kappa}^g\|_2 \|\mathbf{v}_{|g} - \boldsymbol{\kappa}^g - \boldsymbol{\kappa}_{|g}^h\|_2.$$

**$\ell_\infty$  norm:** here, the optimality condition comes down to analyzing when equality holds in the  $\ell_\infty$ - $\ell_1$  Hölder inequality. Specifically,  $\boldsymbol{\kappa}^g = \Pi_{t_g}^*(\mathbf{v}_{|g})$  holds if and only if for all  $\boldsymbol{\kappa}_j^g \neq 0, j \in g$ , we have

$$\mathbf{v}_j - \boldsymbol{\kappa}_j^g = \|\mathbf{v}_{|g} - \boldsymbol{\kappa}^g\|_\infty \text{sign}(\boldsymbol{\kappa}_j^g).$$

Looking at the same condition for  $\kappa^h$ , we have that  $\kappa^h = \Pi_{t_h}^*(\mathbf{v}_{|h} - \kappa_g)$  holds if and only if for all  $\kappa_j^h \neq 0, j \in h$ , we have

$$\mathbf{v}_j - \kappa_j^g - \kappa_j^h = \|\mathbf{v}_{|h} - \kappa^g - \kappa^h\|_\infty \text{sign}(\kappa_j^h).$$

From those relationships we notably deduce that for all  $j \in g$  such that  $\kappa_j^g \neq 0$ ,  $\text{sign}(\kappa_j^g) = \text{sign}(\mathbf{v}_j) = \text{sign}(\kappa_j^h) = \text{sign}(\mathbf{v}_j - \kappa_j^g) = \text{sign}(\mathbf{v}_j - \kappa_j^g - \kappa_j^h)$ . Let  $j \in g$  such that  $\kappa_j^g \neq 0$ . At this point, using the equalities we have just presented,

$$|\mathbf{v}_j - \kappa_j^g - \kappa_j^h| = \begin{cases} \|\mathbf{v}_{|g} - \kappa^g\|_\infty & \text{if } \kappa_j^h = 0 \\ \|\mathbf{v}_{|h} - \kappa^g - \kappa^h\|_\infty & \text{if } \kappa_j^h \neq 0 \end{cases}$$

Since  $\|\mathbf{v}_{|g} - \kappa^g\|_\infty \geq \|\mathbf{v}_{|g} - \kappa^g - \kappa_{|g}^h\|_\infty$  (which can be shown using the sign equalities above), and  $\|\mathbf{v}_{|h} - \kappa^g - \kappa^h\|_\infty \geq \|\mathbf{v}_{|g} - \kappa^g - \kappa_{|g}^h\|_\infty$  (since  $g \subseteq h$ ), we have

$$\|\mathbf{v}_{|g} - \kappa^g - \kappa_{|g}^h\|_\infty \geq |\mathbf{v}_j - \kappa_j^g - \kappa_j^h| \geq \|\mathbf{v}_{|g} - \kappa^g - \kappa_{|g}^h\|_\infty$$

and therefore for all  $\kappa_j^g \neq 0, j \in g$ ,

$$\mathbf{v}_j - \kappa_j^g - \kappa_j^h = \|\mathbf{v}_{|g} - \kappa^g - \kappa_{|g}^h\|_\infty \text{sign}(\kappa_j^g),$$

which gives the result. ■

A remarkable property of this algorithm makes it even more practical in many situations. Proposition 1 tells us indeed that convergence can be reached in one pass when  $\|\cdot\|$  is the  $\ell_2$  or  $\ell_\infty$  norm. Interestingly, we have observed that this was not true in general when  $\|\cdot\|$  is an  $\ell_q$  norm, for  $q \neq 2$  and  $q \neq \infty$ .

**Proposition 1 (Convergence in one pass)** *Suppose that the groups in  $\mathcal{G}$  are ordered according to  $\preceq$  and that the norm  $\|\cdot\|$  is either the  $\ell_2$  or  $\ell_\infty$  norm. Then, after initializing  $\xi$  to 0, **one pass** of Algorithm 1 with the order  $\preceq$  gives the solution of Eq. (2).*

**Proof.** The proof largely relies on Lemma 3. We proceed by induction, by showing that we keep the optimality conditions of Eq. (2) satisfied after each update in Algorithm 1. By definition of Algorithm 1, note that the feasibility of  $\xi$  is always guaranteed. We consider the following induction hypothesis

$$\mathcal{H}(h) \triangleq \{\forall g \preceq h, \text{ it holds that } \xi^g = \Pi_{\lambda w_g}^*([\mathbf{u} - \sum_{g' \neq g} \xi^{g'}]_{|g})\}.$$

Since the dual variables  $\xi$  are initially equal to zero, the summation over  $g' \neq g$  in the definition of  $\mathcal{H}$  can be instead taken over  $g' \preceq h, g' \neq g$ , leading to

$$\mathcal{H}(h) = \{\forall g \preceq h, \text{ it holds that } \xi^g = \Pi_{\lambda w_g}^*([\mathbf{u} - \sum_{g' \preceq h, g' \neq g} \xi^{g'}]_{|g})\}.$$

We initialize the induction with the *first* group in  $\mathcal{G}$ , that, by definition of  $\preceq$ , does not contain any other group. The first step of Algorithm 1 easily shows that the induction hypothesis  $\mathcal{H}$  is satisfied for this first group.

We now assume that  $\mathcal{H}(h)$  is true and consider the next group  $h'$ ,  $h \preceq h'$ , in order to prove that  $\mathcal{H}(h')$  is also satisfied. We have for each group  $g \subseteq h$ ,

$$\xi^g = \Pi_{\lambda w_g}^*([\mathbf{u} - \sum_{g' \preceq h, g' \neq g} \xi^{g'}]_{|g}).$$

Following the update of the group  $h'$ , we have

$$\begin{aligned} \xi^{h'} &= \Pi_{\lambda w_{h'}}^*([\mathbf{u} - \sum_{g' \preceq h} \xi^{g'}]_{|h'}) \\ &= \Pi_{\lambda w_{h'}}^*([\mathbf{u} - \sum_{g' \preceq h', g' \neq h'} \xi^{g'}]_{|h'}). \end{aligned}$$

At this point, we can apply Lemma 3 for each group  $g \subseteq h$ , which proves

$$\begin{aligned} \xi^g &= \Pi_{\lambda w_g}^*([\mathbf{u} - \sum_{g' \preceq h, g' \neq g} \xi^{g'} - \xi^{h'}]_{|g}) \\ &= \Pi_{\lambda w_g}^*([\mathbf{u} - \sum_{g' \preceq h', g' \neq g} \xi^{g'}]_{|g}). \end{aligned}$$

As a result, the induction hypothesis  $\mathcal{H}(h')$  is true.

Therefore, after one complete pass over  $g \in \mathcal{G}$ , the dual variable  $\xi$  satisfies the optimality conditions for Eq. (2), which implies that the pair  $\{\mathbf{v}, \xi\}$  is optimal. Since strong duality holds,  $\mathbf{v}$  is the solution of Eq. (1).  $\blacksquare$

Now that we have proven the convergence of our algorithm in one pass when  $\|\cdot\|$  is the  $\ell_2$  or the  $\ell_\infty$  norm, we study its complexity. Since one pass of Algorithm 1 involves  $p$  projections on the dual balls (respectively the  $\ell_2$  and the  $\ell_1$  balls) of vectors in  $\mathbb{R}^p$ , a naive implementation leads to a polynomial complexity in  $O(p^2)$ , since each of these projections can be obtained in  $O(p)$  operations [Duchi et al., 2008]. However, we show that in these cases, the primal solution  $\mathbf{v}$ , which is the quantity of interest, can be obtained with a smaller complexity. We present a fast recursive implementation in Algorithm 2. Two new notations are used: For a group  $g$  in  $\mathcal{G}$ , we denote by  $r(g)$  the index of the variable that is at the root of the subtree corresponding to  $g$ , and  $C(g)$  is the list of groups that are included in the group  $g$ . The next lemma ensures the correctness of the previous algorithm and gives its complexity.

**Lemma 4 (Correctness of Algorithm 2)** *Algorithm 2 gives the solution of the primal problem Eq. (1) with  $\|\cdot\| = \ell_2$  in  $O(p)$  operations.*

**Proof.** One notices first that the procedure `retrieveNorm` is called one time for each group, computing a set of scalars  $(\rho_g)_{g \in \mathcal{G}}$  in an order which is compatible with the convergence in one pass of Algorithm 1—that is, the children of a node are processed prior to the node itself. Following such an order, the update of the group  $g$  in the original Algorithm 1 computes the variable  $\xi^g$  which updates implicitly the primal variable as follows

$$\mathbf{v}_{|g} \leftarrow \left(1 - \frac{\lambda w_g}{\|\mathbf{w}_{|g}\|_2}\right) \mathbf{v}_{|g}.$$

Observing now the behavior of the recursive implementation in Algorithm 2, it is possible to show by induction that for all group  $g$  in  $\mathcal{G}$ , when the procedure `retrieveNorm(g)`

---

**Algorithm 2** Fast implementation of Algorithm 1 when  $\|\cdot\| = \ell_2$

---

**Require:**  $\mathbf{u} \in \mathbb{R}^p$ , set of groups  $\mathcal{G}$  and  $g_0$  (index of the group that contains everybody).

- 1: Variables:  $\boldsymbol{\tau} = (\rho_g)_{g \in \mathcal{G}}$  in  $\mathbb{R}^{|\mathcal{G}|}$ ,  $\mathbf{v}$  in  $\mathbb{R}^p$  (output).
- 2: retrieveNorm( $g_0$ )
- 3: recursiveScaling( $g_0, 1$ )
- 4: **Return**  $\mathbf{v}$  (primal solution)

**Procedure** retrieveNorm( $g$ )

- 1:  $s \leftarrow \mathbf{u}_{r(g)} \mathbf{u}_{r(g)}$  (auxiliary variable)
- 2: **for**  $h \in C(g)$  **do**
- 3:    $s \leftarrow s + \text{retrieveNorm}(h)$
- 4: **end for**
- 5:  $\rho_g = \max(0, 1 - \lambda w_g / \sqrt{s})$
- 6: **Return**  $s \rho_g \rho_g$

**Procedure** recursiveScaling( $g, s$ )

- 1:  $\rho_g \leftarrow s \rho_g$
  - 2:  $\mathbf{v}_{r(g)} \leftarrow \rho_g \mathbf{u}_{r(g)}$
  - 3: **for**  $h \in C(g)$  **do**
  - 4:   recursiveScaling( $h, \rho_g$ )
  - 5: **end for**
- 

is called, the auxiliary variable  $s$  takes the same value as  $\|\mathbf{v}_{|g}\|_2^2$  evaluated at the iteration  $g$  of Algorithm 1. Therefore, after calling the procedure retrieveNorm( $g_0$ ), where  $g_0$  is the root of the tree, the values  $\rho_g$  correspond to the successive scaling factors of the variable  $\mathbf{v}_{|g}$  that are obtained during the execution of Algorithm 1. After having computed all the scaling factors  $\rho_g$ ,  $g \in \mathcal{G}$ , the procedure recursiveScaling ensures that each variable  $j$  in  $\{1, \dots, p\}$  is scaled by the product of all the  $\rho_h$ , where  $h$  is an ancestor of the variable  $j$ .

The complexity of the algorithm is then easy to analyze: Each procedure retrieveNorm and recursiveScaling is called  $p$  times, each of call for a group  $g$  has a fixed number of operations plus as many operations as the number of children of  $p$ . Since each children can be called at most one time, the total number of operation of the algorithm is  $O(p)$ . ■

We then give a simple recursive implementation of Algorithm 1 for the case  $\|\cdot\| = \ell_\infty$ , in Algorithm 3 for obtaining the primal solution  $\mathbf{v}$ . The next lemma ensures the correctness of the previous algorithm and gives its complexity.

**Lemma 5 (Correctness of Algorithm 3)** *Algorithm 3 gives the solution of the primal problem Eq. (1) with  $\|\cdot\| = \ell_\infty$  in  $O(pd)$  operations, where  $d$  is the depth of the tree.*

**Proof.** Again, we notice that the order of the projections in Algorithm 3 is compatible with a convergence in one pass of Algorithm 1 with  $\|\cdot\| = \ell_\infty$ . Then, it is easy to show that by computing a variable  $\boldsymbol{\xi}^g = \Pi_{\lambda w_g}^*(\mathbf{v}_{|g})$ , Algorithm 1 implicitly updates the primal variable with the formula as in Algorithm 3. Both algorithm have therefore the same output primal variable  $\mathbf{v}$ .

---

**Algorithm 3** Fast implementation of Algorithm 1 when  $\|\cdot\| = \ell_\infty$

---

**Require:**  $\mathbf{u} \in \mathbb{R}^p$ , set of groups  $\mathcal{G}$  and  $g_0$  (index of the group that contains everybody).

- 1: Variables:  $\boldsymbol{\tau} = (\rho_g)_{g \in \mathcal{G}}$  in  $\mathbb{R}^{|\mathcal{G}|}$ ,  $\mathbf{v}$  in  $\mathbb{R}^p$  (output).
- 2: Initialization:  $\mathbf{v} \leftarrow \mathbf{u}$ .
- 3: recursiveProjection( $g_0$ )
- 4: **Return**  $\mathbf{v}$  (primal solution)

**Procedure** recursiveProjection( $g$ )

- 1: **for**  $h \in C(g)$  **do**
  - 2:   recursiveProjection( $h$ )
  - 3: **end for**
  - 4:  $\mathbf{v}_{|g} \leftarrow \mathbf{v}_{|g} - \Pi_{\lambda w_g}^*(\mathbf{v}_{|g})$ .
- 

To analyze the complexity of the procedure, one notice first that a projection on the  $\ell_1$ -ball can be done in linear time [Duchi et al., 2008]. The algorithm performs  $g$  projection, each of them involving  $|g|$  variables. By noticing that if  $g$  and  $h$  are two groups with the same depth in the tree, then  $g \cap h = \emptyset$ , it is easy to show that the number of variables involved in all the projections is less than or equal to  $dp$ , where  $d$  is the depth of the tree. Since the projections are linear in the number of variables, the total complexity is therefore  $O(dp)$ . ■

Then, the following proposition shows that these algorithms have a relatively small complexity.

**Proposition 2 (Complexity of the procedure)** *For tree-structured groups,*

- i) *When  $\|\cdot\|$  is the  $\ell_2$  norm, the primal solution  $\mathbf{v}$  of Algorithm 1 can be obtained in  $O(p)$  operations.*
- ii) *When  $\|\cdot\|$  is the  $\ell_\infty$  norm,  $\mathbf{v}$  can be obtained in  $O(pd)$  operations, where  $d$  is the depth of the tree.*

**Proof.** The proof is immediately obtained from the two Lemmas 4 and 5. ■

Note that the claim of having a linear complexity in the case  $\|\cdot\| = \ell_\infty$  is slightly abusive, since  $d$  could depend of  $p$  as well. For instance, in an unbalanced case, the worse case could be  $d = O(p)$ , in a balanced tree, one could have  $d = O(\log(p))$ . In practice, the structures we have considered experimentally are relatively flat, with a depth not exceeding  $d = 5$ .

## B Additional Results

We present in this section, additional results on natural image patches. We present on Figures 1 and 2 two tree-structured learned dictionaries.

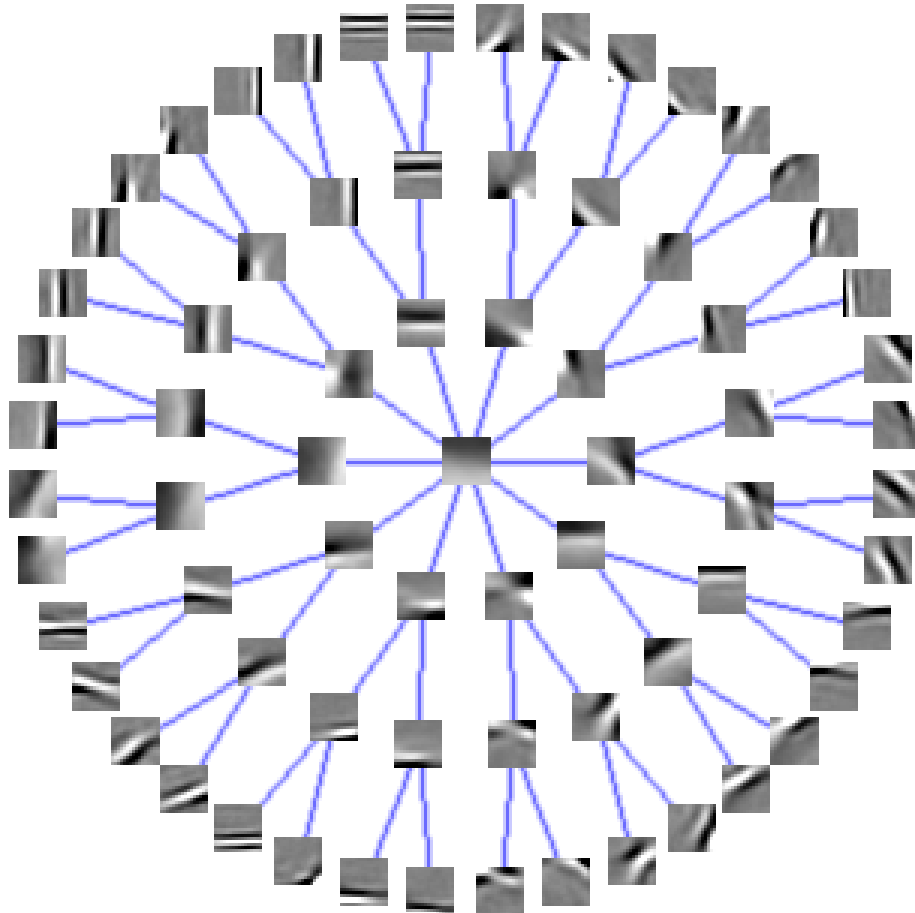


Figure 1: Learned dictionary with a tree structure of depth 4. The root of the tree is in the middle of the figure. The branching factors are  $p_1 = 10$ ,  $p_2 = 2$ ,  $p_3 = 2$ . The dictionary is learned on 50,000 patches of size  $16 \times 16$  pixels.

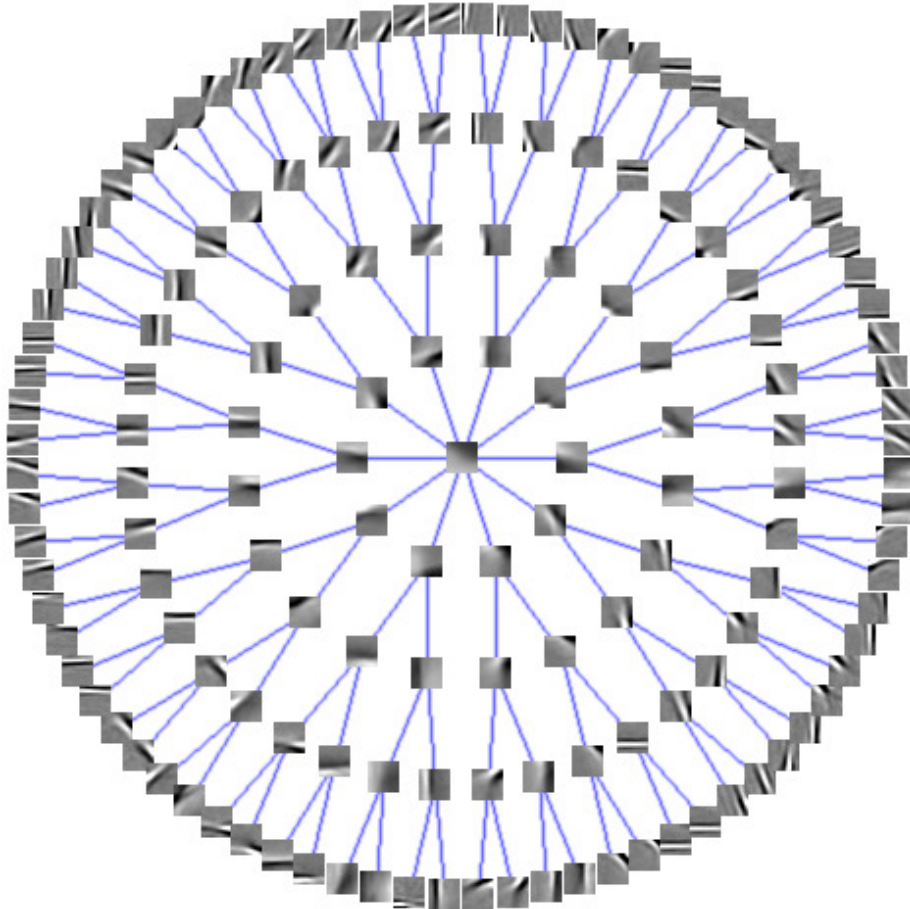


Figure 2: Learned dictionary with a tree structure of depth 5. The root of the tree is in the middle of the figure. The branching factors are  $p_1 = 10$ ,  $p_2 = 2$ ,  $p_3 = 2$ ,  $p_4 = 2$ . The dictionary is learned on 50,000 patches of size  $16 \times 16$  pixels.

## References

- D. P. Bertsekas. *Nonlinear programming*. Athena Scientific Belmont, 1999.
- J. M. Borwein and A. S. Lewis. *Convex analysis and nonlinear optimization: Theory and examples*. Springer, 2006.
- S. P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.
- J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the  $\ell_1$ -ball for learning in high dimensions. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2008.