Iterative hard clustering of features

Vincent Roulet, Fajwel Fogel, Alexandre d'Aspremont, Francis Bach

ITERATIVE HARD CLUSTERING OF FEATURES

VINCENT ROULET, FAJWEL FOGEL, ALEXANDRE D'ASPREMONT, AND FRANCIS BACH

ABSTRACT. We seek to group features in supervised learning problems by constraining the prediction vector coefficients to take only a small number of values. This problem includes non-convex constraints and is solved using projected gradient descent. We prove exact recovery results using restricted eigenvalue conditions. We then extend these results to combine sparsity and grouping constraints, and develop an efficient projection algorithm on the set of grouped and sparse vectors. Numerical experiments illustrate the performance of our algorithms on both synthetic and real data sets.

INTRODUCTION

In a prediction problem, getting compressed or structured predictors can both improve prediction performance and help interpretation. Numerous methods have been developed to select a few key features (see e.g. [Tang et al., 2014]). In particular an extensive literature has been developed to tackle this problem by enforcing sparsity on the prediction vector (see e.g. [Bach et al., 2012]). Here we focus instead on the problem of grouping features. In text classification for example, this amounts to group words that have the same meaning for the task (see e.g. [Gupta et al., 2009] and references therein). In biology, this can be used to retrieve groups of genes that have the same impact on a disease (see e.g. [Balding, 2006, Segal et al., 2003]). More generally this approach can be seen as a supervised quantization of the feature space (see e.g. [Nova and Estévez, 2014] and references therein).

The idea of grouping features to reduce dimensionality of the problem is of course not new. Hastie et al. [2001] used for example supervised learning methods to select group of predictive variables formed by hierarchical clustering. Several models also developed mutual information-based algorithms to remove redundant features, e.g. [Peng et al., 2005, Song et al., 2013, Yu and Liu, 2003]. More recently, regularizers were developed to enforce grouped vectors [Bondell and Reich, 2008, Petry et al., 2011, She et al., 2010]. In particular, Bach [2011] analyzed geometrical properties induced by convex relaxations of submodular functions that lead to group structures. This geometrical perspective was also investigated by Bühlmann et al. [2013], who studied recovery performance of group norms induced by hierarchical clustering methods based on canonical correlations. Finally Shen and Huang [2010] developed an homotopy method to extract homogeneous subgroups of predictors.

In this paper, we study a simple approach to the problem: while sparsity enforces a small number of non-zero coefficient of the prediction vector, we enforce a small number of different coefficient *values*, i.e. we quantize this vector. This naturally induces groups of features that share the same weight in the prediction. We formulate this problem for regression and analyze the geometry induced by the constraints in Section 1. In Section 2 we present a simple projected gradient scheme similar to the Iterative Hard Thresholding (IHT) [Blumensath and Davies, 2009] algorithm used in compressed sensing. While constraints are non-convex, projection on the feasible set reduces to a k-means problem that can be solved exactly with dynamic programming [Bellman, 1973, Wang and Song, 2011]. We analyze the recovery performance of this projected gradient scheme. Although the quantized structure is similar to sparsity, we show that quantizing the prediction vector, while helping interpretation, does not allow to significantly reduce the number of observations required to retrieve the original vector, as in the sparse case.

We then extend the application of the projected gradient scheme to both select and group features in Section 4 by developing a new dynamic program that gives the exact projection on the set of sparse and quantized vectors. Finally, in Section 5, numerical experiments illustrate the performance of our methods on both synthetic and real datasets involving large corpora of text from movie reviews. We show that the use of k-means steps makes our approach fast and scalable while comparing favorably with standard benchmarks and providing meaningful insights on the data structure.

1. Problem Formulation

We present here our formulation for regression and extend its application for classification in Appendix A.

1.1. **Regression with grouped features.**

Given n observations $y_1, \ldots, y_n \in \mathbb{R}$ from data points $x_1, \ldots, x_n \in \mathbb{R}^d$, linear regression aims at finding a regression vector $w \in \mathbb{R}^d$ that fits the data such that

$$y_i \approx w^T x_i, \quad \text{for all} \quad i = 1, \ldots, n.$$

To assess the quality of a prediction vector w, one defines a loss function ℓ that measures their accuracy error $\ell(w^T x, y)$ on a sample (x, y). A common choice of loss, that we investigate here, is the squared loss $\ell_{\text{square}}(w^T x, y) = \frac{1}{2}(w^T x - y)^2$. A classical approach to compute a linear regression vector is then to minimize the empirical loss function

$$L(w) = \frac{1}{n} \sum_{i=1}^{n} \ell(w^T x_i, y_i).$$

In order to prevent the computed prediction parameters from over-fitting the given set of samples, one often adds a regularizer $R(w)$ of the regression vector to the minimization problem. This notably reduces the effect of noise or outliers in the data. Candidate regression parameters are then given by the minimization problem

$$\text{minimize } L(w) + \lambda R(w)$$

in variable $w \in \mathbb{R}^d$, where $\lambda \geq 0$ is a regularization parameter.

Structural information on the task can then be added. For example, one can enforce the regression vectors w to be sparse, i.e. to have few non-zeros coefficients. To this end the support $\text{Supp}(w) = \{i \in \{1, \ldots, d\}, w_i \neq 0\}$ of the variable is constrained to be small such that sparse regression problem reads

$$\begin{array}{ll} \text{minimize} & L(w) + \lambda R(w) \\ \text{subject to} & \mathbf{Card}(\text{Supp}(w)) \leq s, \end{array}$$

in variable $w \in \mathbb{R}^d$ where s is the desired sparsity and for a set S, $\mathbf{Card}(S)$ denotes its cardinality.

Here instead we impose to the regression vectors w to take at most Q *different* coefficient values $v_1, \ldots, v_Q$. Each coefficient v_q is assigned to a group of features g_q, and the regression vector w then defines a partition $G = \{g_1, \ldots, g_Q\}$ of the features. Formally, a vector w defines the partition

$$\text{Part}(w) = \{g \subset \{1, \ldots, d\} : (i, j) \in g \times g, \text{ iff } w_i = w_j\}.$$

formed by maximal groups of equal coefficients of w. For example the vector $w = (1, 3, 3, 2, 1)^T \in \mathbb{R}^5$ forms the partition $\text{Part}(w) = \{\{1, 5\}, \{4\}, \{2, 3\}\}$ of $\{1, \ldots 5\}$. Denoting $\mathbf{Card}(G)$ the number of (non-empty) groups of a partition G, then linear regression enforcing Q groups of features then reads

$$\begin{array}{ll} \text{minimize} & L(w) + \lambda R(w) \\ \text{subject to} & \mathbf{Card}(\text{Part}(w)) \leq Q, \end{array} \tag{1}$$

in variable $w \in \mathbb{R}^d$.

2. Iterative Hard Clustering

2.1. **Algorithm presentation.** We propose to tackle directly the non-convex problem (1) by using a projected gradient scheme, which amounts to iteratively cluster features at each gradient step. We thus transpose the Iterative Hard Thresholding [Blumensath and Davies, 2009] algorithm studied in sparse compressed sensing to the problem of grouping features.

The algorithm relies on the fact that projecting a point w on the feasible set amounts to a clustering problem

$$\text{minimize} \sum_{q=1}^{Q} \sum_{i \in g_q} (w_i - v_q)^2, \tag{2}$$

in the variables $v_1, \ldots, v_Q \in \mathbb{R}$ that are the Q coordinates of the projected vector and $G = \{g_1, \ldots, g_Q\}$ a partition of $\{1, \ldots, d\}$ that can be represented by an assignment matrix $Z \in \{0,1\}^{d \times Q}$, such that $Z_{iq} = 1$ if $i \in g_q$ and $Z_{iq} = 0$ otherwise. This is in fact a k-means problem in one dimension that can be solved in polynomial time by dynamic programming [Bellman, 1973, Wang and Song, 2011]. Given a vector w, whose coordinates we want to cluster in Q groups, we denote by $[Z, v] = \text{k-means}(w, Q)$ respectively the assignment matrix and the vector of coordinates produced by the dynamic program. A projected gradient scheme for problem (1) is described in Algorithm 1.

Algorithm 1 Iterative Hard Clustering

```
Inputs: Data $(X, y)$, $Q$, $\lambda \geq 0$, $\gamma_t$
Initialize $w_0 \in \mathbb{R}^{d \times K}$ (e.g. $w_0 = 0$)
for t = 1,...,T do
    $w_{t+1/2} = w_t - \gamma_t(\nabla L(w_t) + \lambda \nabla R(w_t))$
    $[Z_{t+1}, v_{t+1}] = \text{k-means}(w_{t+1/2}, Q)$
    $w_{t+1} = Z_{t+1} v_{t+1}$
end for
Output: $\hat{w} = w_T$
```

In practice we stop the algorithm when changes in objective values of (1) are below some prescribed threshold ϵ. We use a backtracking line search on the stepsize γ_t that guarantees decreasing of the objective. At each iteration if

$$\bar{w}_{t+1} \quad = \text{k-means}\left(w_t - \gamma_t(\nabla L(w_t) + \lambda \nabla R(w_t)), Q\right)$$

decreases the objective value we keep it and we increase the stepsize by a constant factor $\gamma_{t+1} = \alpha \gamma_t$ with $\alpha > 1$. If $\bar{w}_{t+1}$ increase the objective value we decrease the stepsize by a constant factor $\gamma_t \leftarrow \beta \gamma_t$, with $\beta < 1$, compute new $\bar{w}_{t+1}$ and iterate this operation until $\bar{w}_{t+1}$ decrease the objective value or the stepsize reaches the stopping value ϵ used as as a stopping criterion on the objective values. We observed better results with this line search than with constant stepsize, in particular when the number of samples is small.

Using this strategy we ususally observed convergence of the projected gradient algorithm in less than 100 iterations which makes it scalable. The complexity of its core operations amounts to a k-means problem, which can be solved in $O(d^2 Q)$ operations.

3. Analysis of Iterative Hard Clustering

We now analyze convergence of the Iterative Hard Clustering scheme to retrieve the true regressor w_* in the regression problem. We study the convergence of the projected gradient algorithm applied to a regression problem enforcing Q groups of features. We use a squared loss and no regularization. Therefore our problem reads

$$\begin{array}{ll} \text{minimize} & \frac{1}{2n}\|Xw - y\|_2^2 \\ \text{subject to} & \mathbf{Card}(\text{Part}(w)) \leq Q \end{array} \tag{3}$$

in $w \in \mathbb{R}^d$, where $X = (x_1, \ldots, x_n)^T \in \mathbb{R}^{n \times d}$ is the matrix of data points and $y = (y_1, \ldots, y_n)^T \in \mathbb{R}^n$ is the vector of observations. For the analysis, we use a constant step size $\gamma_t = 1$ and initialize the algorithm with $w_0 = 0$. We assume that the observations y are generated by a linear model whose coefficients w_* satisfy the constraints above, up to additive noise, that is

$$y = Xw_* + \eta$$

where $\eta \sim \mathcal{N}(0, \sigma^2)$ and $\mathbf{Card}(\text{Part}(w_*)) \leq Q$. Hence we analyze the performance of the algorithm to recover w_* and the partition $\text{Part}(w_*)$ of the features and to this end we detail the geometry induced by the constraints.

3.1. **Geometry induced by partitions.** Denote $\mathcal{P}$ the set of partitions of $\{1, \ldots, d\}$ whose definition is recalled below.

Definition 3.1. ***Partitions*** *A collection G of subsets of $\{1, \ldots, d\}$ is a partition of $\{1, \ldots, d\}$ if for any $g, g' \in G \times G$, $g \neq g'$ implies $g \cap g' = \emptyset$ and if $\bigcup_{g \in G} g = \{1, \ldots, d\}$.*

Pair of partitions can then be compared as follows.

Definition 3.2. ***Sup- and sub-partitions*** *Let $G, G' \in \mathcal{P}$ be two partitions. G is a* sup-partition *of G' (or G' is a* sub-partition *of G), denoted*

$$G \succeq G',$$

if for any $g' \in G'$ there exists $g \in G$ such that $g' \subset g$, or equivalently if any $g \in G$ is a union of groups g' of G'.

Relation $\succeq$ is transitive, reflexive and anti-symmetric, such that it is a partial order on the set of partitions. Notice that the number of (non-empty) groups of partitions, $\mathbf{Card}(\cdot)$, decreases with the partial order $\succeq$.

Following proposition highlights the geometry induced by a single partition of the features.

Proposition 3.3. *Any partition $G \in \mathcal{P}$ defines a linear subspace*

$$E_G = \{w \in \mathbb{R}^d : \text{Part}(w) \succeq G\} \tag{4}$$

of vectors w that can be partitioned by G in groups of equal coefficients. For any partitions $G, G' \in \mathcal{P}$, if $G \succeq G'$ then $E_G \subset E_{G'}$.

Proof. Given a partition $G \in \mathcal{P}$ and a vector $w \in \mathbb{R}^d$, G is a sub-partition of $\text{Part}(w)$, i.e. $\text{Part}(w) \succeq G$, if and only if the groups of G are subsets of equal coefficients of w. Now, if, for some $w_1, w_2 \in \mathbb{R}^d$, groups of G are subsets of equal coefficients of both w_1 and w_2, they will also be subset of equal coefficients of any linear combination of w_1, w_2. Therefore E_G is a linear subspace. Second statement follows from the transitivity of $\succeq$. ■

Since $w \in E_{\text{Part}(w)}$, the feasible set for the regression problem (1) enforcing Q groups of features is then a union of subspaces:

$$\begin{aligned} \{w : \mathbf{Card}(\text{Part}(w)) \leq Q\} &= \textstyle\bigcup_{G \in \mathcal{P}\,:\,\mathbf{Card}(G) \leq Q} E_G \\ &= \textstyle\bigcup_{G \in \mathcal{P}\,:\,\mathbf{Card}(G) = Q} E_G \end{aligned}$$

Second equality comes from the fact that if a partition $G \in \mathcal{P}$ has strictly less than Q groups, i.e., $\mathbf{Card}(G) < Q$, some of its groups can always be split to form a new partition G' such that $G \succeq G'$ and $\mathbf{Card}(G') = Q$. Therefore it is sufficient to consider subspaces generated by partitions into exactly Q groups.

3.2. **Convergence analysis.** Without constraints, a gradient descent applied to (3) would act as a fixed point algorithm whose contraction factor depends on the singular values of the Hessian X^TX of the problem. Here we will show that the projected gradient scheme exhibits the same behavior, except that the contraction factor will depend on restricted singular values on small subspaces defined by partitions. These subspaces belong to the following collections

$$\begin{array}{ll} \mathcal{E}_1 & = \{E_G : G \in \mathcal{P}_Q\} \\ \mathcal{E}_2 & = \{E_{G_1} + E_{G_2} : G_1, G_2 \in \mathcal{P}_Q\} \\ \mathcal{E}_3 & = \{E_{G_1} + E_{G_2} + E_{G_3} : G_1, G_2, G_3 \in \mathcal{P}_Q\}, \end{array} \tag{5}$$

where $\mathcal{P}_Q = \{G \in \mathcal{P} : \mathbf{Card}(G) = Q\}$ denotes the set of partitions into exactly Q clusters. For a given subspace E of $\mathbb{R}^d$, we denote U_E any orthonormal basis of E and for a given matrix $X \in \mathbb{R}^{n\times d}$ we denote $\sigma_{\min}(XU_E/\sqrt{n})$ and $\sigma_{\max}(XU_E/\sqrt{n})$ respectively the smallest and largest singular values of $XU_E/\sqrt{n}$, i.e. the smallest and largest restricted singular values of $X/\sqrt{n}$ on E.

The next proposition adapts the proof of IHT in our context using that feasible set is a union of subspaces.

Proposition 3.4. *Iterative Hard Clustering Algorithm 1 with constant step size $\gamma_t = 1$ and initialization $w_0 = 0$, applied to (3) outputs iterates w_t that converge to the original w_* as*

$$\|w_* - w_t\|_2 \leq \rho^t \|w_*\|_2 + \frac{1-\rho^t}{1-\rho}\nu\|\eta\|_2,$$

where

$$\begin{array}{rcl} \rho & = & 6\max_{E\in\mathcal{E}_3} \max(\delta_E, \delta_E^3), \\ \nu & = & 2/\sqrt{n}\max_{E\in\mathcal{E}_2} \sigma_{\max}(XU_E/\sqrt{n}) \end{array}$$

where, for any subspace E of $\mathbb{R}^d$, δ_E is the smallest non-negative constant that satisfies

$$1 - \delta_E \leq \sigma_{\min}\left(XU_E/\sqrt{n}\right) \leq \sigma_{\max}\left(XU_E/\sqrt{n}\right) \leq 1 + \delta_E.$$

Proof. To describe the Iterative Hard Clustering algorithm, we define for $t \geq 0$,

$$\begin{array}{rl} w_{t+1/2} = & w_t - \gamma_t \nabla L(w_t) = w_t - \frac{1}{n}X^TX(w_t - w_*) + \frac{1}{n}X^T\eta \\ w_{t+1} = & \underset{w\in\mathbb{R}^d\,:\,\mathbf{Card}(\mathrm{Part}(w))\leq Q}{\operatorname{argmin}} \|w - w_{t+1/2}\|_2^2, \end{array}$$

where w_{t+1} is given exactly by the solution of a k-means problem in one dimension. The analysis of convergence relies on the characterization of the subspaces that contain w_*, w_t and w_{t+1}. We define therefore

$$\begin{array}{ll} E_{t,*} & = E_{\mathrm{Part}(w_t)} + E_{\mathrm{Part}(w_*)} \\ E_{t+1,*} & = E_{\mathrm{Part}(w_{t+1})} + E_{\mathrm{Part}(w_*)} \\ E_{t,t+1,*} & = E_{\mathrm{Part}(w_t)} + E_{\mathrm{Part}(w_{t+1})} + E_{\mathrm{Part}(w_*)}, \end{array}$$

and the orthogonal projections on these set respectively $P_{t,*}, P_{t+1,*}, P_{t,t+1,*}$. Bound on the error can then be computed as follows:

$$\begin{array}{rcl} \|w_* - w_{t+1}\|_2 & = & \|P_{t+1,*}(w_* - w_{t+1})\|_2 \\ & \leq & \|P_{t+1,*}(w_* - w_{t+1/2})\|_2 + \|P_{t+1,*}(w_{t+1/2} - w_{t+1})\|_2. \end{array} \tag{6}$$

In the second term, as $\mathbf{Card}(\mathrm{Part}(w_*)) \leq Q$ and $w_{t+1} = \underset{w\,:\,\mathbf{Card}(\mathrm{Part}(w))\leq Q}{\operatorname{argmin}} \|w - w_{t+1/2}\|_2^2$, we have

$$\|w_{t+1} - w_{t+1/2}\|_2^2 \leq \|w_* - w_{t+1/2}\|_2^2$$

which is equivalent to

$$\|P_{t+1,*}(w_{t+1} - w_{t+1/2})\|_2^2 + \|(I - P_{t+1,*})w_{t+1/2}\|_2^2 \leq \|P_{t+1,*}(w_* - w_{t+1/2})\|_2^2 + \|(I - P_{t+1,*})w_{t+1/2}\|_2^2$$

and this last statement implies

$$\|P_{t+1,*}(w_{t+1} - w_{t+1/2})\|_2 \leq \|P_{t+1,*}(w_* - w_{t+1/2})\|_2.$$

This means that we get from (6)

$$\begin{aligned}
\|w_* - w_{t+1}\|_2 &\leq 2\|P_{t+1,*}(w_* - w_{t+1/2})\|_2 \\
&= 2\|P_{t+1,*}(w_* - w_t - \frac{1}{n}X^T X(w_* - w_t) - \frac{1}{n}X^T\eta)\|_2 \\
&\leq 2\|P_{t+1,*}(I - \frac{1}{n}X^T X)(w_* - w_t)\|_2 + \frac{2}{n}\|P_{t+1,*}(X^T\eta)\|_2 \\
&= 2\|P_{t+1,*}(I - \frac{1}{n}X^T X)P_{t,*}(w_* - w_t)\|_2 + \frac{2}{n}\|P_{t+1,*}(X^T\eta)\|_2 \\
&\leq 2\|P_{t+1,*}(I - \frac{1}{n}X^T X)P_{t,*}\|_2\|w_* - w_t\|_2 + \frac{2}{n}\|P_{t+1,*}X^T\|_2\|\eta\|_2.
\end{aligned}$$

Now, assuming

$$\begin{aligned}
2\|P_{t+1,*}(I - \frac{1}{n}X^T X)P_{t,*}\|_2 &\leq \rho \qquad (7) \\
\frac{2}{n}\|P_{t+1,*}X^T\|_2 &\leq \nu \qquad (8)
\end{aligned}$$

and developing the latter inequality over t, using that $w_0 = 0$, we get

$$\|w_* - w_t\|_2 \leq \rho^t\|w_*\|_2 + \frac{1-\rho^t}{1-\rho}\nu\|\eta\|_2.$$

Bounds ρ and ν can then be given by restricted singular values of X. For ν in (8), we have

$$\|P_{t+1,*}X^T\|_2 = \|XP_{t+1,*}\|_2 \overset{\vartheta}{\leq} \max_{E\in\mathcal{E}_2}\|XP_E\|_2 = \max_{E\in\mathcal{E}_2}\sigma_{\max}(XU_E).$$

For ϑ, as noticed in previous section, if for example $\mathbf{Card}(\text{Part}(w_*)) < Q$, there always exists $G \in \mathcal{P}$ such that $\text{Part}(w_*) \succeq G$, $\mathbf{Card}(G) = Q$ and so $E_{\text{Part}(w_*)} \subset E_G$. Therefore there exists $F_{t+1,*}$ that contain $E_{t+1,*}$ and belong to $\mathcal{E}_2$, such that we can restrict our attention to restricted singular values on subspaces in $\mathcal{E}_2$ (defined from partitions in exactly Q groups).

For ρ in (7), we have

$$\begin{aligned}
\|P_{t+1,*}(I - X^T X)P_{t,*}\|_2 &\overset{\vartheta_1}{\leq} \|P_{t,t+1,*}(I - \frac{1}{n}X^T X)P_{t,t+1,*}\|_2 \\
&\overset{\vartheta_2}{\leq} \max_{E\in\mathcal{E}_3}\|P_E(I - \frac{1}{n}X^T X)P_E\|_2 \\
&= \max_{E\in\mathcal{E}_3}\|U_E(I - \frac{1}{n}U_E^T X^T XU_E)U_E^T\|_2 \\
&= \max_{E\in\mathcal{E}_3}\|I - \frac{1}{n}U_E^T X^T XU_E\|_2,
\end{aligned}$$

where for a subspace E, U_E denotes any orthonormal basis of it. In ϑ_1 we used that $E_{t,t+1,*}$ contain $E_{t,*}$ and $E_{t+1,*}$. In ϑ_2 we use the same argument as for ν to restrict our attention to subspaces defined by partitions into exactly Q groups. Finally, for a subspace E if $\delta_E \geq 0$ satisfies

$$1 - \delta_E \leq \sigma_{\min}\left(XU_E/\sqrt{n}\right) \leq \sigma_{\max}\left(XU_E/\sqrt{n}\right) \leq 1 + \delta_E,$$

then [Vershynin, 2010, Lemma 5.38] shows

$$\|I - \frac{1}{n}U_E^T X^T XU_E\|_2 \leq 3\max\{\delta_E, \delta_E^2\},$$

which concludes the proof by taking the maximum of δ_E over $\mathcal{E}_3$. ■

If the contraction factor is sufficient, convergence of the projected gradient scheme to the original vector is ensured up to a constant error of the order of the noise as the classical IHT algorithm does for sparse signals [Blumensath and Davies, 2009].

3.3. **Recovery performance on random instances.** We observe now that for isotropic independent sub-Gaussian data x_i the restricted singular values introduced in Proposition 3.4 depend on the number of subspaces that define partitions and their dimension. This proposition reformulates results of Vershynin [2010, Theorems 5.39, 5.65] in our context.

Proposition 3.5. *Let $\mathcal{E}$ be a collection of subspaces of $\mathbb{R}^d$ of dimension at most D and denote N their number. If the samples are n isotropic independent sub-gaussian random variables forming a design matrix $X = (x_1, \ldots, x_n)^T \in \mathbb{R}^{n\times d}$, then for all $E \in \mathcal{E}$*

$$1 - \delta - \epsilon \leq \sigma_{\min}\left(\frac{XU_E}{\sqrt{n}}\right) \leq \sigma_{\max}\left(\frac{XU_E}{\sqrt{n}}\right) \leq 1 + \delta + \epsilon,$$

holds with probability larger than $1 - \exp(-c\epsilon^2 n)$, where $\delta = C_0\sqrt{\frac{D}{n}} + \sqrt{\frac{\log(N)}{cn}}$ and C_0, c depend only on the sub-gaussian norm of the x_i.

Proof. Let $E \in \mathcal{E}$, denote U_E one of its orthonormal basis and $D_E = \dim(E) \leq D$ its dimension. The rows of XU_E are orthogonal projections of the rows of X onto E, so they are still independent sub-gaussian isotropic random vectors. We can therefore apply [Vershynin, 2010, Theorem 5.39] on $XU_E \in \mathbb{R}^{n\times D_E}$. Hence for any $s \geq 0$, with probability at least $1 - 2\exp(-cs^2)$, the smallest and largest singular values of $XU_E/\sqrt{n}$ are bounded as

$$1 - C_0\sqrt{\frac{Q}{n}} - \frac{s}{\sqrt{n}} \leq \sigma_{\min}\left(\frac{XU_E}{\sqrt{n}}\right) \leq \sigma_{\max}\left(\frac{XU_E}{\sqrt{n}}\right) \leq 1 + C_0\sqrt{\frac{Q}{n}} + \frac{s}{\sqrt{n}}, \qquad (9)$$

where c and C_0 depend only on the sub-gaussian norm of the x_i. Now, by taking the union bound, (9) holds for any $G \in \mathcal{P}_Q$ with probability $1 - 2N\exp(-cs^2)$.

Taking $s = \sqrt{\frac{\log(N)}{c}} + \epsilon\sqrt{n}$, we get for all $G \in \mathcal{P}_Q$,

$$1 - \delta - \epsilon \leq \sigma_{\min}\left(\frac{XU_E}{\sqrt{n}}\right) \leq \sigma_{\max}\left(\frac{XU_E}{\sqrt{n}}\right) \leq 1 + \delta + \epsilon,$$

with probability at least $1 - 2\exp(-c\epsilon^2 n)$, where $\delta = C_0\sqrt{\frac{Q}{n}} + \sqrt{\frac{\log(N)}{cn}}$. ■

To ensure approximate recovery of the projected gradient scheme in Proposition 3.4, one needs to control restricted singular values of X on subspaces in $\mathcal{E}_3$ in order to ensure that the contraction factor ρ is strictly less than one. Precisely, we need to ensure for that for any $E \in \mathcal{E}_3$, there exists $0 \leq \delta < 1/6$ such that

$$1 - \delta \leq \sigma_{\min}\left(\frac{XU_E}{\sqrt{n}}\right) \leq \sigma_{\max}\left(\frac{XU_E}{\sqrt{n}}\right) \leq 1 + \delta.$$

Denoting D_3 and N_3 respectively the largest dimension of the subspaces in $\mathcal{E}_3$ and the number of these subspaces, the last proposition shows that when observations x_i are isotropic independent sub-gaussian, their number n must therefore satisfy

$$C_0\sqrt{\frac{D_3}{n}} < \frac{1}{6} \quad \text{and} \quad \sqrt{\frac{\log(N_3)}{cn}} < \frac{1}{6}$$

which is roughly

$$n = \Omega(D_3) \quad \text{and} \quad n = \Omega(\log(N_3)) \qquad (10)$$

to ensure contraction. The first condition in (10) means that subspaces must be low-dimensional, in our case $D_3 = 3Q$, and we naturally want the number of groups to be small. The second condition in (10) means that the structure (partitioning here) is restrictive enough, i.e., that the number N_3 of possible configurations is small enough.

To compute N_3, denote N_Q the number of partitions in exactly Q groups such that $N_3 = \binom{N_Q}{3}$. The number of partitions into Q groups is then given by the the Stirling number of second kind $N_Q = \left\{ {d \atop Q} \right\}$, that can be bounded as

$$Q^{d-Q} \leq \left\{ {d \atop Q} \right\} \leq \frac{1}{2}(ed/Q)^Q Q^{d-Q}. \tag{11}$$

Using standard bounds on the binomial coefficients this means

$$N_3 \geq \left(\frac{N_Q}{3}\right)^3 \geq \frac{Q^{3d-3Q}}{27}.$$

Therefore although the intrinsic dimension of our variables is of order $3Q$, the number of subspaces N_3 is such that we need roughly $n \geq 3d\log(Q)$ observations, i.e., approximately as many samples as features, so the grouping structure is not specific enough to reduce the number of samples required by a projected gradient scheme to converge. On the other hand, given this many samples, the algorithm provably converges to a clustered output, which helps interpretation.

As a comparison, classical sparse recovery problems have the same structure [Rao et al., 2012], as s-sparse vectors for instance can be described as $\{w = Zv,\ Z \in \{0,1\}^{d\times s},\ Z^T 1 = 1,\ v \in \mathbb{R}^s\}$ and so are part of a "union of subspaces". However in the case of sparse vectors the number of subspaces grows as d^s which means recovery requires much less samples than features.

4. Sparse and grouped linear models

Projected gradient schemes are simple but scalable algorithms to tackle constrained structures of linear models. It has been developed for sparsity through the Iterative Hard Thresholding algorithm [Blumensath and Davies, 2009], we presented its version to group features in Section 2, we now extend it to both select s features and group them in Q groups. A regression problem that enforces predictors to have at most s non-zeros coefficients grouped in at most Q groups reads

$$\begin{array}{ll} \text{minimize} & L(w) + \lambda R(w) \\ \text{subject to} & \mathbf{Card}(\text{Supp}(w)) \leq s \\ & \mathbf{Card}(\text{Part}(w)) \leq Q + 1 \end{array} \tag{12}$$

in variable $w \in \mathbb{R}^d$, where L and R are respectively the loss and the regularizer of the prediction problem as introduced in Section 1 and $\lambda \geq 0$ is a regularization parameter. Naturally we take $Q \leq s$ as one cannot cluster s features in more than s groups.

Adapting the projected gradient for this problem essentially means producing an efficient algorithm for the projection step. To this end, we develop a new dynamic program to get the projection on s-sparse vectors whose non-zero coefficients form Q groups. Analysis of the recovery performance of this scheme will then directly follow from counting arguments similar to those used for clustered vectors.

4.1. **Projection on s-sparse Q-grouped vectors.**

4.1.1. *Formulation of the problem.* A feasible point $w \in \mathbb{R}^d$ for problem (12) is described by the partition of its coordinates $G = \{g_0, \ldots, g_{Q_G}\}$ in groups of equal coefficients, where g_0 is the group of zero coefficients, and $v_1, \ldots, v_{Q_G}$ the possible values of the non-zero coefficients. $Q_G = \mathbf{Card}(G) - 1 \geq 0$ denotes here the number of (non-empty) groups of a partition $G \in \mathcal{P}$.

Let us fix a point $u \in \mathbb{R}^d$, its distance to a feasible point w reads

$$\|u - w\|_2^2 = \sum_{i \in g_0} u_i^2 + \sum_{q=1}^{Q_G} \sum_{i \in g_q} (u_i - v_q)^2, \tag{13}$$

for given $G = \{g_0, \dots g_{Q_G}\} \in \mathcal{P}$ and $v \in \mathbb{R}^{Q_G}$. For a fixed partition $G \in \mathcal{P}$, hence a fixed subspace, minimization in v gives the barycenters of the groups $g_1, \dots, g_{Q_G}$ of non-zero coefficients denoted

$$\mu_q = \frac{1}{s_q} \sum_{i \in g_q} u_i \quad \text{for } q = 1, \dots Q_G,$$

where $s_q = \mathbf{Card}(g_q)$ is the size of the q^{th} group. Inserting them in (13), the distance to a subspace of sparse grouped coefficients defined by a partition $G \in \mathcal{P}$ can be developed as

$$\sum_{i \in g_0} u_i^2 + \sum_{q=1}^{Q_G} \sum_{i \in g_q} (u_i^2 + \mu_q^2 - 2 v_q u_i) = \sum_{i=1}^{d} u_i^2 - \sum_{q=1}^{Q_G} s_q \mu_q^2.$$

Projection on the feasible set of (12), that minimizes the above distance for all possible partitions in Q groups of s non-zeros coefficients, amounts then to solve

$$\begin{array}{ll} \text{maximize} & \sum_{q=1}^{Q_G} s_q \mu_q^2 \\ \text{subject to} & \mathbf{Card}\left(\bigcup_{q=1}^{Q_G} g_q \right) \leq s, \quad 0 \leq Q_G \leq Q, \end{array} \tag{14}$$

in the partition $G = \{g_0, \dots, g_{Q_G}\} \in \mathcal{P}$, where $\mu_q = \frac{1}{s_q} \sum_{i \in g_q} u_i$ and $Q_G = \mathbf{Card}(G) - 1$.

This problem amounts to select a number $s' \leq s$ of features and cluster them in a number $Q' \leq Q$ groups whose barycenters have maximal magnitude for the objective in (14). The objective can then be split into positive and negative barycenters to treat each resulting problem independently and then find the best balance between both parts.

4.1.2. *Dynamic programing.* To solve problem (14), observe first that the objective is clearly increasing with the number of groups, as it allows more degrees of freedom to approximate u. Furthermore if the number s' of selected features is fixed, the number of groups cannot exceed it, i.e. $Q' \leq s'$, and it can therefore be set at $\min(s', Q)$.

A solution of (14) that selects $s' \leq s$ features is then composed of a partition of j points into q groups that define positive barycenters, and a partition of the $s' - j$ remaining points into $\min(s', Q) - q$ groups that define negative centers. We therefore tackle (14) by searching for the best parameters s', j, q that balance optimally the objective into positive and negative barycenters.

To this end, we define $f_+(j, q)$ the optimal value of (14) when picking j points clustered in q groups of positive barycenters, *i.e.* the solution of the problem

$$\begin{array}{ll} \text{maximize} & \sum_{p=1}^{q} s_p \mu_p^2 \\ \text{subject to} & \mu_p = \frac{1}{s_p} \sum_{i \in g_p} u_i > 0 \\ & \mathbf{Card}\left(\bigcup_{p=1}^{q} g_p \right) = j, \end{array} \tag{$P_+(j,q)$}$$

in disjoint groups $g_1, \dots, g_q \subset \{1, \dots, d\}$. This problem is not always feasible, as it may not be possible to find q clusters of positive barycenters with j points. In that case we denote its solution $f_+(j, q) = -\infty$. We define similarly $f_-(j, q)$ the optimal value of (14) when picking j points clustered in q groups forming only negative barycenters. The best balance between the two, which solves (14), is then given by solving:

$$\begin{array}{ll} \text{maximize} & f_+(j, q) + f_-(s' - j, Q' - q) \\ \text{subject to} & 0 \leq j \leq s', \ 0 \leq q \leq Q', \\ & 0 \leq s' \leq s, \ Q' = \min(s', Q), \end{array} \tag{15}$$

in variables s', j and q.

It remains to compute f_+ and f_- efficiently. We present our approach for f_+ that transposes to f_-. Let $S_+ \subset \{1, \dots, d\}$ be the optimal subset of indexes taken for ($P_+(j, q)$) and $i \in S_+$. If there exists $j \in \{1, \dots, d\} \setminus S_+$ such that $u_j \geq u_i$, then swapping j and i would increase the magnitude of the barycenter of the group that i belongs to and so the objective. Therefore ($P_+(j, q)$) amounts to a partitioning problem on the j largest values of u. From now on, assume coefficients of u to be in decreasing order $u_1 \geq \ldots \geq u_d$. For ($P_+(j,q)$) a feasible problem, denote $g_1,\ldots,g_q$ the optimal partition of $\{1,\ldots,j\}$ whose corresponding barycenters are in decreasing order. Let i be the index of the largest coefficient of u in g_q, then necessarily $g_1,\ldots,g_{q-1}$ is optimal to solve ($P_+(i-1,q-1)$). f_+ can then be computed recursively as

$$f_+(j,q) = \max_{\substack{q \leq i \leq j \\ \mu(u_i,\ldots,u_j)>0}} f_+(i-1,q-1) + (j-i+1)\mu(u_i,\ldots,u_j)^2, \tag{16}$$

where $\mu(u_i,\ldots,u_j) = \frac{1}{j-i+1}\sum_{l=i}^{j} u_l$ can be computed in constant time using that

$$\mu(x_i,\ldots,x_j) = \frac{u_i + (j-i)\mu(u_{i+1},\ldots,u_j)}{j-i+1}.$$

By convention $f_+(j,q) = -\infty$ if is not possible to find q clusters of positive barycenters with j points such that ($P_+(j,q)$) is not feasible. Values of f_+ are stored to compute (15). Two auxiliary variables I_+ and v_+ store respectively the indexes of the largest value of x in group g_q and the barycenter of the group g_q. The same dynamic program can be used to compute f_-, I_- and v_-, defined similarly as I_+ and v_+, by reversing the order of the values of x. A grid search on $f(j,q,s') = f_+(j,q) + f_-(s'-j, Q'-q)$, with $Q' = \min(s',Q)$, gives the optimal balance between positive and negative barycenters. A backtrack on I_- and I_+ finally gives the best partition and the projection with the associated barycenters given in v_- and v_+.

f_+ is initialized as a grid of $s+1$ and $Q+1$ columns such that $f_+(0,q) = 0$ for any q, $f_+(j,0) = 0$ and $f_+(j,1) = j\mu(u_1,\ldots,u_j)^2$ for any $j \geq 1$. I_+ and v_+ are initialized by $I_+(j,1) = 1$ and $\mu_+(j,1) = \mu(u_1,\ldots,u_j)$.

Each dynamic program needs only to build the best partitions for the s smallest or largest partitions so they cost $O(s^2Q)$ elementary operations. The grid search and the backtrack cost respectively $O(s^2Q)$ and $O(Q)$ elementary operations. Overall, the complexity of the projection does not exceed $O(s^2Q)$.

4.2. **Recovery performance.** Analysis of recovery performance of the projected gradient for sparse clustered vectors follows the one provided in Section 2. Our problem is to recover an original vector w_* such that $\mathbf{Card}(\text{Supp}(w_*)) \leq s$ and $\mathbf{Card}(\text{Part}(w_*)) \leq Q+1$ that generates n noisy observations y_i from data points x_i as

$$y = Xw_* + \eta$$

where $\eta \sim \mathcal{N}(0,\sigma^2)$, where $X = (x_1,\ldots,x_n)^T \in \mathbb{R}^{n\times d}$ is the matrix of data points and $y = (y_1,\ldots,y_n) \in \mathbb{R}^n$ is the vector of observations. To this end we attempt to solve a regression problem enforcing Q groups of s features with a squared loss and no regularization, which reads

$$\begin{array}{ll} \text{minimize} & \frac{1}{2n}\|Xw-y\|_2^2 \\ \text{subject to} & \mathbf{Card}(\text{Supp}(w)) \leq s, \\ & \mathbf{Card}(\text{Part}(w)) \leq Q+1 \end{array} \tag{17}$$

As in Section 2, we use a projected gradient scheme with constant step size $\gamma_t = 1$ and initialized at $w_0 = 0$, the only difference is the projection step that is given here by the dynamic program presented in last section.

First we detail the geometry of the feasible set of (12). A given subset $S \subset \{1,\ldots,d\}$ defines a linear subspace

$$E_S = \{w \in \mathbb{R}^d : \text{Supp}(w) \subset S\}.$$

By combining a subset $S \in \{1,\ldots,d\}$ with a partition $G \in \mathcal{P}$ we get a linear subspace

$$E_{S,G} = \{w : \text{Supp}(w) \subset S,\ \text{Part}(w) \succeq G\} = E_S \cap E_G.$$

Vectors in $E_{S,G}$ have at most $\mathbf{Card}(G)-1$ different non-zero coefficients such that $\dim(E_{S,G}) = \mathbf{Card}(G)-1$. The feasible set of (12) is then a union of subspaces, namely,

$$\bigcup_{\substack{S\in\{1,\ldots,d\}\,:\,\mathbf{Card}(S)\leq s\\ G\in\mathcal{P}\,:\,\mathbf{Card}(G)\leq Q+1}} E_{S,G}.$$

Analysis of convergence made in Proposition 3.4 for the clustered case relies only on the fact that the feasible set is a union of subspaces and that the projection on it can be computed exactly so it applies also in this case. However the contraction factor will now depend on restricted singular values of the data on a smaller collection of subspaces. Precisely, define

$$\begin{array}{ll}\tilde{\mathcal{E}} & = \{E_{S,G} : \mathbf{Card}(S) = s,\ \mathbf{Card}(G) = Q+1\}\\ \tilde{\mathcal{E}}_3 & = \{E_1 + E_2 + E_3 : E_1, E_2, E_3 \in \tilde{\mathcal{E}}\}.\end{array}$$

The contraction factor depends then on the restricted singular values of the matrix X on subspaces belonging to $\tilde{\mathcal{E}}_3$. Since $\dim(E_{S,G}) = \mathbf{Card}(G)-1$, subspaces in $\tilde{\mathcal{E}}_3$ have a dimension at most $3Q$. Denoting N and N_3 the cardinality of respectively $\tilde{\mathcal{E}}$ and $\tilde{\mathcal{E}}_3$, we have $N_3 = \binom{N}{3}$. Subspaces of $\tilde{\mathcal{E}}$ are defined by selecting s features among d and partitioning these s features into Q groups so that their number is $N = \binom{d}{s}\left\{{s \atop Q}\right\}$. Using classical bounds on the binomial coefficient and (11), we can roughly bound N for $s \geq 3$, $Q \geq 3$ by

$$N \leq \left(\frac{ed}{s}\right)^s \frac{1}{2}\left(\frac{e}{Q}\right)^Q s^Q Q^{s-Q} \leq d^s s^Q Q^{s-Q}$$

and so

$$N_3 \leq \left(\frac{eN}{3}\right)^3 \leq N^3 \leq (d^s s^Q Q^{s-Q})^3$$

Propositions 3.4 and 3.5 adapted in this case thus predict that the number of observations must satisfy

$$n = \Omega(s\log d + Q\log(s) + (s-Q)\log(Q))$$

for a projected gradient scheme to recover approximately w_*. It produces $Q+1$ cluster of features, one being a cluster of zero features, reducing dimensionality, while needing roughly as many samples as non-zero features.

5. Numerical experiments

We now test our methods, first on artificial datasets to check their robustness to noisy data and then on real data extracted from movie reviews.

5.1. **Synthetic dataset.** We first test the robustness of our algorithms for an increasing number of training samples or level of noise in the labels. We generate a linear model in dimension $d = 100$ with a vector $w_* \in \mathbb{R}^d$ that has only $Q = 5$ different values uniformly distributed around 0. We sample n Gaussian random points x_i with noisy observations $y_i = w^T x_i + \eta$, where $\eta \sim \mathcal{N}(0, \sigma^2)$. We vary the number of samples n or the level of noise σ and measure $\|w_* - \hat{w}\|_2$, the l_2 norm of the difference between the true vector of weights w^* and the estimated ones $\hat{w}$.

In Table 1 and 2, we compare the proposed algorithms to Least Squares regularized by the squared norm (LS), Least Squares regularized by the squared norm followed by k-means on the weights (using associated centroids as predictors) (LSK) and OSCAR (OS) [Bondell and Reich, 2008]. For OSCAR we used a submodular approach [Bach et al., 2012] to compute the corresponding proximal algorithm, which makes it scalable. "Oracle" refers to the Least Square solution given the true assignments of features and can be seen as the best achievable error rate. We study the performance of our model with a squared loss and regularized by the squared Euclidean norm of the variable. We solve with Iterative Hard Clustering (IHC) (initialized with the solution of Least Square followed by k-means). When varying the number of samples, noise on labels is set to $\sigma = 0.5$ and when varying level of noise σ number of samples is set to $n = 150$.

Regularization parameters of the models were all cross-validated using a logarithmic grid. Results were averaged over 50 experiments and figures after the ± sign correspond to one standard deviation.

	n = 50	n = 75	n = 100	n = 125	n = 150
Oracle	0.16±0.06	0.14±0.04	0.10±0.04	0.10±0.04	0.09±0.03
LS	61.94±17.63	51.94±16.01	21.41±9.40	1.02±0.18	0.70±0.09
LSK	62.93±18.05	57.78±17.03	10.18±14.96	0.31±0.19	0.19±0.12
IHC	63.31±18.24	52.72±16.51	5.52±14.33	**0.14**±0.09	**0.09**±0.04
OS	**61.54**±17.59	52.87±15.90	11.32±7.03	1.25±0.28	0.71±0.10

TABLE 1. Measure of $\|w_* - \hat{w}\|_2$, the l_2 norm of the difference between the true vector of weights w^* and the estimated ones $\hat{w}$ along number of samples n.

	σ = 0.05	σ = 0.1	σ = 0.5	σ = 1
Oracle	0.86±0.27	1.72±0.54	8.62±2.70	17.19±5.43
LS	7.04±0.92	14.05±1.82	70.39±9.20	140.41±18.20
LSK	1.44±0.46	2.88±0.91	19.10±12.13	48.09±27.46
IHC	**0.87**±0.27	**1.74**±0.52	**9.11**±4.00	26.23±18.00
OS	14.43±2.45	18.89±3.46	71.00±10.12	140.33±18.83

TABLE 2. Measure of $\|w_* - \hat{w}\|_2$, the l_2 norm of the difference between the true vector of weights w^* and the estimated ones $\hat{w}$ along level of noise σ.

We observe that IHC gives significantly better results than other methods and even reach the performance of the Oracle for $n > d$ and for small σ, while for $n \leq d$ results are in the same range.

5.2. **Predicting ratings from reviews using groups of words.** We perform "sentiment" analysis on newspaper movie reviews. We use the publicly available dataset introduced by Pang and Lee [2005] which contains movie reviews paired with star ratings. We treat it as a regression problem, taking responses for y in $(0, 1)$ and word frequencies as covariates. The corpus contains $n = 5006$ documents and we reduced the initial vocabulary to $d = 5623$ words by eliminating stop words, rare words and words with small TF-IDF on the whole corpus. We evaluate our algorithms for regression with clustered features against standard regression approaches: Least-Squares (LS), Least-Squares followed by k-means on predictors (LSK), Lasso and Iterative Hard Thresholding (IHT). We also tested our projected gradient with sparsity constraint, initialized by the solution of LSK (IHCS). Number of clusters, sparsity constraints and regularization parameters were 5-fold cross-validated using respectively grids going from 5 to 15, $d/2$ to $d/5$ and logarithmic grids. Cross validation and training were made on 80% on the dataset and tested on the remaining 20%. It gave $Q = 15$ number of clusters and $d/2$ sparsity constraint for our algorithms. Results are reported in Table 3, the ± sign shows one standard deviation when varying the training and test sets on 20 experiments.

All methods perform similarly except plain IHT and Lasso whose hypotheses does not seem appropriate for the problem. Our approaches have the benefit of reducing dimensionality from 5623 to 15 words and provide meaningful cluster of words. The clusters with highest absolute weights are also the ones with smallest number of words, which confirms the intuition that only a few words are highly discriminative. We illustrate this in Table 4, picking randomly words of the four clusters within which predictor weights have largest magnitude.

6. CONCLUSION AND FUTURE WORK

We presented new algorithmic schemes to group features with potentially additional sparsity constraints. To this end, we introduced a combinatorial structure on the prediction vector akin to the one used for sparsity

LS	LSK	IHC	OS	IHCS	IHT	Lasso
1.51±0.06	1.53±0.06	1.52±0.06	1.47±0.07	1.53±0.06	2.19±0.12	3.77±0.17

TABLE 3. 100 × mean square errors for predicting movie ratings associated with reviews.

2 most negative clusters	bad, awful, worst, boring, ridiculous, watchable, suppose, disgusting,
2 most positive clusters	perfect, hilarious, fascinating, great wonderfully, perfectly, good-spirited, world, unexpected, gem, recommendation, excellent, rare, marvelous, mature send, delightful, funniest

TABLE 4. Clustering of words on movie reviews. We show clusters of words within which associated predictor weights have largest magnitude. First row presents ones associated to a negative coefficient and therefore bad feelings about movies, second row ones to a positive coefficient and good feelings about movies.

and identify the corresponding projections to the set of constraints. On one side, our numerical results validate the performance of these schemes, their cheap projection cost and empirical convergence make them suitable for large data sets where they provide an efficient reduction of dimension. On the other side, our theoretical analysis of recovery performance shows the difficulty of the problem of grouping features compared to standard sparsity. While constraining the number of groups of identical features appears natural and can be tackled with dynamic programming, it leads to a hard recovery problem that needs a large number of samples to be solved. This paves the way of defining other combinatorial penalties on partitions of level sets for which one can obtain better recovery results, provided that projection can be done easily. Notice finally that such combinatorial structures can lead to regularizers as illustrated in Appendix B. The problem is then to provide an efficient corresponding proximal operator of the regularizer.

Acknowledgements. The authors are part of PSL Research University.

The authors would like to acknowledge support from a starting grant from the European Research Council (ERC project SIPA), an AMX fellowship, the MSR-Inria Joint Centre, as well as support from the chaire *Économie des nouvelles données*, the *data science* joint research initiative with the *fonds AXA pour la recherche* and a gift from Société Générale Cross Asset Quantitative Research.

REFERENCES

Arthur, D. and Vassilvitskii, S. [2007], k-means++: The advantages of careful seeding, *in* 'Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms', Society for Industrial and Applied Mathematics, pp. 1027–1035.

Bach, F., Jenatton, R., Mairal, J. and Obozinski, G. [2012], 'Optimization with sparsity-inducing penalties', *Found. Trends Mach. Learn.* **4**(1), 1–106.

Bach, F. R. [2011], Shaping level sets with submodular functions, *in* 'Advances in Neural Information Processing Systems', pp. 10–18.

Balding, D. J. [2006], 'A tutorial on statistical methods for population association studies', *Nature reviews. Genetics* **7**(10), 781.

Bellman, R. [1973], 'A note on cluster analysis and dynamic programming', *Mathematical Biosciences* **18**(3), 311–312.

Blumensath, T. and Davies, M. E. [2009], 'Iterative hard thresholding for compressed sensing', *Applied and Computational Harmonic Analysis* **27**(3), 265–274.

Bondell, H. D. and Reich, B. J. [2008], 'Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar', *Biometrics* **64**(1), 115–123.

Bühlmann, P., Rütimann, P., van de Geer, S. and Zhang, C.-H. [2013], 'Correlated variables in regression: clustering and sparse estimation', *Journal of Statistical Planning and Inference* **143**(11), 1835–1858.

Gupta, V., Lehal, G. S. et al. [2009], 'A survey of text mining techniques and applications', *Journal of emerging technologies in web intelligence* **1**(1), 60–76.

Hastie, T., Tibshirani, R., Botstein, D. and Brown, P. [2001], 'Supervised harvesting of expression trees', *Genome Biology* **2**(1), research0003–1.

Hastie, T., Tibshirani, R. and Friedman, J. [2008], *The elements of statistical learning: data mining, inference and prediction*, 2 edn, Springer.

Jacob, L., Obozinski, G. and Vert, J.-P. [2009], Group lasso with overlap and graph lasso, *in* 'Proceedings of the 26th annual international conference on machine learning', ACM, pp. 433–440.

Nova, D. and Estévez, P. A. [2014], 'A review of learning vector quantization classifiers', *Neural Computing and Applications* **25**(3-4), 511–524.

Obozinski, G. and Bach, F. [2012], 'Convex relaxation for combinatorial penalties', *arXiv preprint arXiv:1205.1240* .

Pang, B. and Lee, L. [2005], Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales, *in* 'Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics', Association for Computational Linguistics, pp. 115–124.

Peng, H., Long, F. and Ding, C. [2005], 'Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy', *IEEE Transactions on pattern analysis and machine intelligence* **27**(8), 1226–1238.

Petry, S., Flexeder, C. and Tutz, G. [2011], 'Pairwise fused lasso'.

Rao, N., Recht, B. and Nowak, R. [2012], 'Signal Recovery in Unions of Subspaces with Applications to Compressive Imaging', *ArXiv e-prints* .

Segal, M. R., Dahlquist, K. D. and Conklin, B. R. [2003], 'Regression approaches for microarray data analysis', *Journal of Computational Biology* **10**(6), 961–980.

She, Y. et al. [2010], 'Sparse regression with exact clustering', *Electronic Journal of Statistics* **4**, 1055–1096.

Shen, X. and Huang, H.-C. [2010], 'Grouping pursuit through a regularization solution surface', *Journal of the American Statistical Association* **105**(490), 727–739.

Song, Q., Ni, J. and Wang, G. [2013], 'A fast clustering-based feature subset selection algorithm for high-dimensional data', *IEEE transactions on knowledge and data engineering* **25**(1), 1–14.

Tang, J., Alelyani, S. and Liu, H. [2014], 'Feature selection for classification: A review', *Data Classification: Algorithms and Applications* p. 37.

Vershynin, R. [2010], 'Introduction to the non-asymptotic analysis of random matrices', *arXiv preprint arXiv:1011.3027* .

Wang, H. and Song, M. [2011], 'Ckmeans. 1d. dp: optimal k-means clustering in one dimension by dynamic programming', *The R Journal* **3**(2), 29–33.

Yu, L. and Liu, H. [2003], Feature selection for high-dimensional data: A fast correlation-based filter solution, *in* 'Proceedings of the 20th international conference on machine learning (ICML-03)', pp. 856–863.

Appendix A. Extended formulation for classification

A.1. **Formulation for regression with assignment matrices.** A partition G of $\{1,\ldots,d\}$ into Q groups can be encoded by an assignment matrix $Z\in\{0,1\}^{d\times Q}$, whose rows index the features and columns index the groups, such that

$$Z_{iq}=\begin{cases}1 & \text{if} \quad i\in g_q\\ 0 & \text{otherwise.}\end{cases}$$

Observe that a partition G into Q groups $g_1,\ldots g_Q$ is independent of the ordering of the groups, namely, $g_{\pi(1)},\ldots g_{\pi(Q)}$, where π is a permutation of $\{1,\ldots,Q\}$, describes as well G. Consequently a partition can be encoded by several assignment matrices, these are identical up to a permutation of their columns defining the groups.

A binary matrix $Z\in\{0,1\}^{p\times Q}$ describes a partition G of $\{1,\ldots,p\}$ into Q groups, if and only if it satisfies $Z\mathbf{1}=\mathbf{1}$ as it encodes the fact that each element belongs to exactly one group. Since groups of a partition are disjoints, columns of assignment matrices are orthogonal. Their squared Euclidean norm and ℓ_1 norm are equal to the size of the groups they represent, i.e. $\|Z_q\|_2^2=\|Z_q\|_1=Z^T\mathbf{1}=\mathbf{Card}(g_q)$, where Z_q is the q^{th} column of an assignment matrix Z of a partition $G=(g_1,\ldots,g_Q)$. Combining two previous comments, we conclude that size of the groups are the squared singular values of the assignment matrix., i.e. $Z^TZ=\mathbf{diag}(s)$ where $s=(\mathbf{Card}(g_1),\ldots,\mathbf{Card}(g_Q))$ encodes the size of the groups $g_1,\ldots,g_Q$ that Z represents.

A regression vector w that has at most Q values can then be described by an assignment matrix Z and the prediction weights $v_1,\ldots,v_Q$ such that $w_i=\sum_{q=1}^Q Z_{iq}v_q=(Zv)_i$. Therefore linear regression enforcing Q group of features reads

$$\begin{array}{ll}\text{minimize} & L(w)+\lambda R(w)\\ \text{subject to} & w=Zv, \quad Z\in\{0,1\}^{d\times Q}, \quad Z\mathbf{1}=\mathbf{1}\end{array}\tag{18}$$

in variables $w\in\mathbb{R}^d$, $v\in\mathbb{R}^Q$ and Z, where $\lambda\geq 0$ is a regularization parameter.

Notice also that affine regression problems that seek for a regression vector $w\in\mathbb{R}^d$ and an intercept $b\in\mathbb{R}$ such that $y\approx w^Tx+b$ can be treated similarly. It suffices to add a constant feature equals to one to data points x and to consider the resulting problem in dimension $d+1$. In this case regularization function R and partitioning constraints apply only on the first d dimensions of the resulting problem.

A.2. **Classification.** Numerous models have been proposed for classification, we refer the interesting reader to Hastie et al. [2008] for a detailed presentation. Here we briefly present one of them, namely one-vs-all linear classification, in order to focus on the optimization problem that will be constrained to group features. In classification, data points $x_1,\ldots,x_n\in\mathbb{R}^d$ belong to one of K classes, which can be encoded by binary vectors $y_i\in\{-1,1\}^K$ such that $y_{ik}=1$ if i^{th} point belongs to class k and -1 otherwise. One-vs-all linear classification aims then at computing hyperplanes defining regions of space where points are more likely to belong to a a given class. Such hyperplanes are defined by their normals $w_1,\ldots,w_K$, forming a matrix of linear classifiers $W\in\mathbb{R}^{d\times K}$ whose classification error on a sample (x,y) is measured by a loss $\ell(W^Tx,y)$ such as the squared loss $\ell_{\text{square}}(W^Tx,y)=\frac{1}{2}\|W^Tx-y\|_2^2$. One searches then to minimize the empirical loss function

$$L(W)=\frac{1}{n}\sum_{i=1}^n\ell(W^Tx_i,y_i).$$

As for regression, a regularizer $R(W)$ can be added on the linear classifiers. Candidate classification parameters are then given by solving

$$\text{minimize } L(W)+\lambda R(W)$$

in variable $W\in\mathbb{R}^{d\times K}$, where $\lambda\geq 0$ is a regularization parameter.

To group features, we will enforce the classifiers to share the same partition of their coefficients. Namely, if this partition is encoded by an assignment matrix Z and $v_k=(v_{1k},\ldots,v_{Qk})$ represent the Q different coefficients of the k^{th} linear classifier w_k, then $w_k = Zv_k$. Linear classification enforcing Q groups of constraints then reads

$$\begin{array}{ll} \text{minimize} & L(W) + \lambda R(W) \\ \text{subject to} & W = ZV, \quad Z \in \{0,1\}^{d\times Q}, \quad Z\mathbf{1} = \mathbf{1} \end{array} \tag{19}$$

in variables $W \in \mathbb{R}^{d\times K}$, $V \in \mathbb{R}^{Q\times K}$ and Z, where $\lambda \geq 0$ is a regularization parameter. Observe that constraints in (19) are essentially the same as the ones in (18), except that these are formulated on matrices. However this simple difference has important algorithmic implications. Projection on the feasible set is again a k-means problem but in dimension K such that it can not be solved exactly. However careful initializations as made by k-means++ [Arthur and Vassilvitskii, 2007] offers logarithmic approximations of the solution.

Notice that for binary classification, a vector of labels of dimension one is sufficient to encode the class information, such that binary classification reduces to a problem of the form (1). As for regression, this setting can be applied to compute affine hyperplanes by extending the problem in $d+1$ dimension and by applying regularization and constraints only on the first d dimensions.

Appendix B. Norm for grouping features

In this section, we seek to develop a norm that induce groups of features by regularization rather than enforcing it by constraints. We begin by detailing the geometrical interpretation of standard algebraic tools used to describe partitions.

B.1. **Geometrical interpretation of algebraic tools.** In Proposition 3.3 we defined subspaces from partitions of $\{1,\ldots,d\}$. Here we relate them to standard algebraic tools used to represent partitions. First for a partition $G = \{g_1,\ldots,g_Q\}$ into Q groups, $w \in E_G$ has at most Q different values and can be encoded using assignment matrices as presented in Section A.1. In other words, for an assignment matrix Z of G, one has

$$E_G = \{w = Zv, v \in \mathbb{R}^d\}$$

Columns of Z are orthogonal since since groups are disjoints and not null if G has no empty groups. In this case, Z is therefore an orthogonal basis of E_G. As mentioned in Section A.1, several assignment matrices can encode a partition, i.e. several binary matrices form a basis of a subspace E_G. However E_G and the orthogonal projector on it are for their part uniquely defined by G.

For a given partition G and any assignment matrix Z of G, the orthogonal projection on E_G reads $M = Z(Z^TZ)^\dagger Z^T \in \mathbb{R}^{p\times p}$, where $A^\dagger$ denotes the pseudo-inverse of a matrix A, here $(Z^TZ)^\dagger = \mathbf{diag}(s^\dagger)$, where $s^\dagger_q = 1/\,\mathbf{Card}(g_q)$ if g_q is non-empty and $s^\dagger_q = 0$ otherwise. It is called the normalized equivalence matrix of G and satisfies

$$M_{ij} = \begin{cases} 1/\,\mathbf{Card}(g_q) & \text{if} \quad (i,j) \in (g_q \times g_q) \\ 0 & \text{otherwise.} \end{cases}$$

To represent more generally partitions of $\{1,\ldots,d\}$ in any number of groups one can use binary matrices $Z \in \{0,1\}^{d\times d}$ that satisfy $Z\mathbf{1} = \mathbf{1}$. Number of non-zero columns of such matrices are then the number of groups of the partition they represent. Once again partitions can be represented by several assignment matrices but are in bijection with the set of normalized equivalence matrices

$$\mathfrak{M} = \{M = Z(Z^TZ)^\dagger Z\ ,\ Z \in \{0,1\}^{d\times d}\ ,\ Z\mathbf{1} = \mathbf{1}\}. \tag{20}$$

Number of groups of a partition G is then equal to the rank of its normalized equivalence matrix (the dimension of E_G), i.e. $\mathbf{Card}(G) = \mathbf{Rank}(M) = \mathbf{Tr}(M)$, since M is a projector.

B.2. **Convex relaxation of combinatorial penalty.** Our framework constraints number of level sets of the variables, i.e. the function

$$\Omega(w) = \mathbf{Card}(\mathrm{Part}(w)).$$

Following Obozinski and Bach [2012] ,we investigate how this combinatorial function can be incorporated in standard Euclidean regularization by finding the tightest convex homogeneous envelope of

$$\Omega_2(w) = \frac{1}{2}\|w\|_2^2 + \frac{1}{2}\mathbf{Card}(\mathrm{Part}(w)).$$

Following proposition details its formulation

Proposition B.1. *The tightest convex homogeneous envelope of*

$$\Omega_2(w) = \frac{1}{2}\|w\|_2^2 + \frac{1}{2}\mathbf{Card}(\mathrm{Part}(w))$$

is

$$\|w\|_{\Omega_2} = \inf_{\substack{(x_M)_{M\in\mathcal{M}} \\ x=\sum_{M\in\mathcal{M}} Mx_M}} \sum_{M\in\mathcal{M}} \mathbf{Tr}(M)^{1/2}\|Mx_M\|_2,$$

where $\mathfrak{M}$ *defined in* (20) *is the set of normalized equivalence matrices of partitions of* $\{1,\dots,d\}$.

$\|w\|_{\Omega_2}$ *is a norm, whose dual norm is*

$$\|w\|_{\Omega_2}^* = \max_{M\in\mathcal{M}} \frac{\|Mx\|_2}{\mathbf{Tr}(M)^{1/2}}.$$

Proof. First we give an algebraic formulation of the combinatorial function Ω. Given a vector $w \in \mathbb{R}^d$, $\mathrm{Part}(w)$ is the largest partition (in terms of $\succeq$ presented in Definition 3.2) in groups of equal coefficients of w. It defines therefore the smallest subspace (see Proposition 3.3) on which w lies. $\mathbf{Card}(\mathrm{Part}(w))$ is then the dimension of the smallest subspace defined from partitions, on which w lies. Using normalized equivalence matrices that are orthogonal projections on these subspaces the combinatorial penalty Ω reads

$$\Omega(w) = \mathbf{Card}(\mathrm{Part}(w)) = \min_{\substack{M\in\mathfrak{M} \\ Mw=w}} \mathbf{Tr}(M).$$

Now, following Obozinski and Bach [2012], we begin by computing the homogenized version of Ω_2 defined as $h(w) = \inf_{\lambda>0} \frac{\Omega_2(\lambda w)}{\lambda}$, then we compute the Fenchel bi-conjugate of h. We have

$$\begin{aligned} h(w) &= \inf_{\lambda>0} \frac{1}{2}\|w\|_2^2\lambda + \frac{1}{2}\Omega(w)\lambda^{-1}. \\ &= \|w\|_2\Omega(w)^{1/2} \end{aligned}$$

Fenchel dual of h reads then

$$\begin{aligned} h^*(x) &= \sup_{w\in\mathbb{R}^d} x^Tw - \|w\|_2\Omega(w)^{1/2} \\ &= \sup_{w\in\mathbb{R}^d} \max_{\substack{M\in\mathfrak{M} \\ Mw=w}} x^Tw - \|w\|_2\,\mathbf{Tr}(M)^{\frac{1}{2}} \\ &= \max_{M\in\mathfrak{M}} \sup_{\substack{w\in\mathbb{R}^d \\ Mw=w}} x^Tw - \|w\|_2\,\mathbf{Tr}(M)^{\frac{1}{2}} \\ &= \max_{M\in\mathfrak{M}} \begin{cases} 0 & \text{if} \quad \|Mx\|_2 \leq \mathbf{Tr}(M)^{1/2} \\ +\infty & \text{otherwise} \end{cases} \\ &= \begin{cases} 0 & \text{if} \quad \max_{M\in\mathcal{M}} \|Mx\|_2\,\mathbf{Tr}(M)^{-1/2} \leq 1 \\ +\infty & \text{otherwise.} \end{cases} \end{aligned}$$

Define

$$\|w\|_{\Omega_2}^* = \max_{M \in \mathfrak{M}} \|Mx\|_2 \, \mathbf{Tr}(M)^{-1/2}.$$

$\|w\|_{\Omega_2}^*$ is convex as a finite maximum of convex functions, it is clearly homogeneous and as $\mathbf{I} \in \mathcal{M}$ we have $\|w\|_{\Omega_2}^* \implies w = 0$. Hence $\|w\|_{\Omega_2}^*$ is a norm. h^* is then the indicator function of the unit norm ball of $\|w\|_{\Omega_2}^*$.

Fenchel bi-dual of h is then

$$\begin{aligned}
h^{**}(w) &= \sup_{x \in \mathbb{R}^d} w^T x - h^*(x) \\
&= \sup_{x \in \mathbb{R}^d} w^T x - \sum_{M \in \mathfrak{M}} \sup_{\lambda_M \geq 0} \lambda_M (\|Mx\|_2 - \mathbf{Tr}(M)^{1/2}) \\
&= \inf_{(\lambda_M)_{M \in \mathfrak{M}},\, \lambda_M \geq 0} \sum_{M \in \mathfrak{M}} \mathbf{Tr}(M)^{1/2} \lambda_M + \sup_{x \in \mathbb{R}^d} w^T x - \sum_{M \in \mathfrak{M}} \lambda_M \|Mx\|_2 \\
&= \inf_{(\lambda_M)_{M \in \mathfrak{M}},\, \lambda_M \geq 0} \sum_{M \in \mathfrak{M}} \mathbf{Tr}(M)^{1/2} \lambda_M + \sup_{x \in \mathbb{R}^d} w^T x - \sum_{M \in \mathfrak{M}} \lambda_M \sup_{\|a_M\|_2 \leq 1} x^T M a_M \\
&= \inf_{\substack{(\lambda_M)_{M \in \mathfrak{M}},\, \lambda_M \geq 0 \\ (a_M)_{M \in \mathfrak{M}},\, \|a_M\|_2 \leq 1 \\ x = \sum_{M \in \mathfrak{M}} \lambda_M M a_M}} \sum_{M \in \mathfrak{M}} \mathbf{Tr}(M)^{1/2} \lambda_M \\
&= \inf_{\substack{(x_M)_{M \in \mathfrak{M}} \\ x = \sum_{M \in \mathfrak{M}} M x_M}} \sum_{M \in \mathfrak{M}} \mathbf{Tr}(M)^{1/2} \|M x_M\|_2 \\
&= \|w\|_{\Omega_2}.
\end{aligned}$$

Since h^* is the indicator function of the unit ball of $\|w\|_{\Omega_2}^*$, $\|w\|_{\Omega_2}$ is the dual norm of $\|w\|_{\Omega_2}^*$. ■

Computed norm $\|w\|_{\Omega_2}$ appears similar to the grouped norms defined for example by Jacob et al. [2009]. However we are not aware of algorithms that can compute the norm or its proximal operator such that its utility in practice is unclear.

INRIA - SIERRA PROJECT TEAM & D.I., UMR 8548,
ÉCOLE NORMALE SUPÉRIEURE, PARIS, FRANCE.
E-mail address: vincent.roulet@inria.fr

C.M.A.P., ÉCOLE POLYTECHNIQUE, UMR CNRS 7641
E-mail address: fajwel.fogel@cmap.polytechnique.fr

CNRS & D.I., UMR 8548,
ÉCOLE NORMALE SUPÉRIEURE, PARIS, FRANCE.
E-mail address: aspremon@ens.fr

INRIA - SIERRA PROJECT TEAM & D.I., UMR 8548,
ÉCOLE NORMALE SUPÉRIEURE, PARIS, FRANCE.
E-mail address: francis.bach@inria.fr