Bayesian hierarchical dictionary learning

IOP Publishing Inverse Problems
Inverse Problems **39** (2023) 024006 (22pp) https://doi.org/10.1088/1361-6420/acad21

Bayesian hierarchical dictionary learning

N Waniorek[1,2], D Calvetti[1,*] and E Somersalo[1]

[1] Department of Mathematics, Applied Mathematics and Statistics, Case Western Reserve University, Cleveland, OH, United States of America
[2] Committee on Computational and Applied Mathemtics, University of Chicago, Chicago, IL United States of America

E-mail: dxc57@case.edu

Received 14 June 2022; revised 17 November 2022
Accepted for publication 20 December 2022
Published 3 February 2023

Abstract

Dictionary learning, aiming at representing a signal in terms of the atoms of a dictionary, has gained popularity in a wide range of applications, including, but not limited to, image denoising, face recognition, remote sensing, medical imaging and feature extraction. Dictionary learning can be seen as a possible data-driven alternative to solve inverse problems by identifying the data with possible outputs that are either generated numerically using a forward model or the results of earlier observations of controlled experiments. Sparse dictionary learning is particularly interesting when the underlying signal is known to be representable in terms of a few vectors in a given basis. In this paper, we propose to use hierarchical Bayesian models for sparse dictionary learning that can capture features of the underlying signals, e.g. sparse representation and nonnegativity. The same framework can be employed to reduce the dimensionality of an annotated dictionary through feature extraction, thus reducing the computational complexity of the learning task. Computed examples where our algorithms are applied to hyperspectral imaging and classification of electrocardiogram data are also presented.

Keywords: hyperspectral imaging, nonnegative matrix factorization, sparse coding

(Some figures may appear in colour only in the online journal)

1. Introduction

In the current data driven age, dictionary learning is a very natural answer to the demand of extracting information out of massive data sets available in a variety of applications. The process of representing a signal in terms of dictionary entries has become very popular in

* Author to whom any correspondence should be addressed.

several areas, including signal analysis [30], image denoising [17, 20], medical imaging [23, 32, 35], remote sensing [3, 27], artificial intelligence [31, 34] and astrophysics [26, 29], and it has affected how we now regard tasks like feature extraction and model selection. In some applications the signals are known to have a sparse representation or more generally, are compressible in terms of properly defined underlying dictionary, in the sense that only a few components contribute in a significant manner in the representation. When that is the case, ideally the algorithms should exploit the sparsity. The process of selecting the dictionary and finding a sparse representation of signals, often referred to as sparse coding, has been the topic of active research over the past decade and a half. One advantage of sparse representation is that the signal can be explained concisely in terms of few feature vectors, which leads to effective compression of the information, but also helps the interpretation of the signals. There are several examples of signals admitting a sparse representation over given bases [15, 34]. In many imaging and interpolation applications, it is typical to have a sparse representation over discrete cosine or wavelet transform bases. One of the most common applications of sparse dictionary learning is compressed sensing [18], allowing a compression and recovery of high dimensional information from few linear observations. The core recovery algorithms of sparse coding include the basis pursuit [16], comprising minimization of the ℓ_1-norm of the unknown of interest subject to linear constraints, referred to as LASSO in computational statistical literature [28].

Recently there has been an interest in algorithms for sparse dictionary learning which not only favor representations in terms of a few dictionary members, but also simultaneously reduce the size of the dictionary by extracting the most relevant features. In fact, when a training set of possible outputs is available, a judicious reduction of the underlying dictionary is often feasible due to the redundancies between the training set entries comprising the uncompressed dictionary. Adaptive dictionary learning may not only yield more accurate representations, but it may also reduce substantially the computational complexity of the problem [1, 20].

Dictionary learning is inherently a linear inverse problem where the unknowns to be estimated are the coefficients of the dictionary entries, or atoms. In this paper we adhere to the Bayesian paradigm whereas all unknown parameters are modeled as random variables, hence described in terms of probability distributions [10]. In the Bayesian framework any *a priori* belief about the solution is encoded in the prior probability distribution. The sparsity requirement can be seen as a prior belief, hence favoring a solution with very few nonzero entries can be reduced to the problem of finding a suitable prior. A natural choice for this purpose is the family of hierarchical conditionally Gaussian priors with independent components, each with a zero mean [9, 11] and unknown variances that in turn are modeled as mutually independent random variables. If the signal is believed *a priori* to be sparse, a natural choice for the prior of the variances, usually referred to as hyperprior, is a positive fat tailed distribution with a small expected value, thus favoring small outputs with occasional outliers [4, 7, 14].

In this paper we show that conditionally Gaussian priors with generalized gamma hyperpriors are very well suited for sparse dictionary learning applications. A particular novelty of this paper is to show that using the computational efficiency of Krylov subspace iterative solvers for linear systems equipped with prior conditioning [6] within an iterative alternating scheme (IAS) algorithm [7, 14] it is possible solve sparse dictionary learning problems of high dimensionality at very small computational cost. Moreover, we show the viability of the approach in applications where the dictionary is known to consist only of nonnegative signals and the signal to be encoded is believed to be nonnegative. In those cases the nonnegativity can guide the adaptation of the dictionary, generating effectively a sparse non-negative matrix factorization

(NMF) [21]. In the computed examples, the viability of the approach is demonstrated with an application to hyperspectral imaging and classification of electrocardiograms (ECGs).

The paper is organized as follows. In the next section we formulate the dictionary learning problem with positivity and sparsity both in the supervised and unsupervised contexts. In section 3 the problems are recast in the form of computing maximum *a posteriori* (MAP) estimators for appropriately selected Bayesian models, and variants of the IAS algorithm to solve the problems are presented. Finally, section 4 contains computed examples of both supervised and unsupervised dictionary learning.

2. Dictionary learning, sparsity and positivity

Dictionary learning algorithms aim to solve a rather wide class of problems for which the data matrix $\mathsf{X} \in \mathbb{R}^{n\times p}$ is approximated by the product of two matrices,

$$\mathsf{X} = \mathsf{WH}, \quad \mathsf{W} \in \mathbb{R}^{n\times k}, \quad \mathsf{H} \in \mathbb{R}^{k\times p}.$$

The factor matrices W and H may be subjected to additional constraints, for example non-negativity and/or sparsity. In the context of dictionary learning, the matrix W is referred to as the *dictionary*, and its columns, denoted by $w^{(j)}$, $1 \leqslant j \leqslant k$ as dictionary entries or atoms, while the entries of H are the coefficients. Depending on the problem, the atoms may either be given, or alternatively, learning the dictionary based on a training set is part of the problem. In the following, we recall the basic concepts and introduce the notation to be used in the rest of the article.

Non-negativity: In several applications, the non-negativity of the matrix entries of W and/or H may be required. We will use the notation $\mathsf{A} \geqslant 0$ to indicate that all entries of A are non-negative.

Sparsity: Define the ℓ_0-norm of a vector $x \in \mathbb{R}^m$ as the cardinality of its support,

$$\|x\|_0 = \mathrm{card}(\mathrm{supp}(x)) = \#\{j \in \{1,\dots,m\} \text{ such that } x_j \neq 0\}.$$

We say that a vector x is sparse if $\|x\|_0 \ll m$, a concept that is to some extent subjective due to the interpretation of the meaning of '$\ll$'. More generally, we say that the vector x is compressible with a threshold value $\delta > 0$ if

$$\|x\|_{0,\delta} = \#\{j \in \{1,\dots,m\} \text{ such that } |x_j| \geqslant \delta\} \ll m.$$

Next we summarize some of the dictionary learning approaches proposed in the literature, and their applications.

2.1. Unsupervised learning

Dictionary learning approaches have been successfully employed in applications related to inverse problems. In inverse problems, the goal is to estimate an unknown quantity x from indirect and noisy observations, the most standard model being

$$b = f(x) + \varepsilon,$$

where $f : \mathbb{R}^n \to \mathbb{R}^m$ is a presumably known function, and ε represents additive noise. Instead of pursuing the solution by optimization based methods or Bayesian statistical methods, in some important applications [23, 26], a different, data-driven approach is assumed: by using a computer model, a large family of possible outcomes,

$$\{w^{(1)},\dots,w^{(N)}\}, \quad w^{(j)} = f(x^{(j)}),$$

is first generated, and the observation b is then matched with the outcomes, through the solution of a minimization problem of the form

$$\text{minimize } \|b - \mathsf{W}h\| \text{ subject to constraints } h \geqslant 0,\ \|h\|_0 \ll N.$$

A sparse coefficient vector h is tantamount to saying that it is possible to explain the data with a few candidate feature vectors, or a mixture of elementary models, or endforms, as in hyperspectral imaging [3, 27]. Thus, the atoms in the given dictionary W represent a template set to be used to explain an observation, and the problem has a formal similarity to matched filtering or query matching algorithms. When the dictionary is very rich, the matching process may become computationally demanding. In those cases it may be necessary to compress the dictionary in a learned way. Let $\mathsf{X} \in \mathbb{R}^{n\times p}$ represent a computed or measured data set, and assume that we want to reduce the number of atoms by removing the redundant ones. If we assume, furthermore, that the columns $x^{(j)}$ of X represent non-negative quantities, the corresponding dictionary learning problem can be written as

$$(\mathsf{W}^*, \mathsf{H}^*) = \text{argmin}\big\{\|\mathsf{X} - \mathsf{WH}\|_F,\ (\mathsf{W}, \mathsf{H}) \in \mathbb{R}^{n\times k} \times \mathbb{R}^{k\times p}\big\} \text{ subject to } \mathsf{W} \geqslant 0, \mathsf{H} \geqslant 0, \tag{1}$$

where the rank k is chosen arbitrarily. The non-negativity of W and H is required in order to allow an interpretation of the atoms $w^{(j)}$ in physical terms. The minimization problem (1) can be interpreted as a NMF; see e.g. [13, 21] and references therein. To favor sparse solutions, we can recast (1) in the form

$$\begin{aligned}(\mathsf{W}^*, \mathsf{H}^*) = \text{argmin}\big\{\|\mathsf{X} - \mathsf{WH}\|_F,\ (\mathsf{W}, \mathsf{H}) \in \mathbb{R}^{n\times k} \times \mathbb{R}^{k\times p}\big\} \text{ subject to } \mathsf{W} \geqslant 0,\\ \mathsf{H} \geqslant 0, \mathsf{H} \text{ sparse.}\end{aligned} \tag{2}$$

Sparsity promoting NMF algorithms have been proposed in the literature, see, e.g. [22]. In this paper, we address the sparsity in terms of hierarchical Bayesian hypermodels, as discussed in detail in section 3.

2.2. Supervised learning

Unlike in unsupervised dictionary learning, in the supervised dictionary learning (SDL) context, the training data $\mathsf{X}_{\text{train}}$ are assigned class labels, hence it is natural to partition the columns of the data matrix according to their classification, i.e.

$$\mathsf{X}_{\text{train}} = \begin{bmatrix} \mathsf{X}_1 & \cdots & \mathsf{X}_L \end{bmatrix} \in \mathbb{R}^{n\times p}, \tag{3}$$

with each submatrix $X_\ell \in \mathbb{R}^{n\times p_\ell}$ comprising the data in class ℓ, $1 \leqslant \ell \leqslant L$. SDL algorithms are typically used for classification of vectors with unknown labeling into the classes.

We discuss briefly two SDL algorithms most relevant for our work, the *sparse representation-based classification* (SRC) algorithm, developed by Wright *et al* [34], and *Metaface*, developed by Yang *et al* [36].

The SRC algorithm uses the full training set (3) as a dictionary, $\mathsf{W} = \mathsf{X}_{\text{train}}$, $\mathsf{W}_\ell = \mathsf{X}_\ell$. Given the test data matrix $\mathsf{X}_{\text{test}} \in \mathbb{R}^{n\times q}$ whose columns are the vectors $x^{(j)}$ with unknown class attributes, the SCR algorithm seeks to represent $x^{(j)}$ in terms of the dictionary,

$$x^{(j)} \approx \mathsf{W}h^{(j)} = \sum_{\ell=1}^{L} \mathsf{W}_\ell h^{(j,\ell)}, \quad h^{(j)} = \begin{bmatrix} h^{(j,1)} \\ \vdots \\ h^{(j,L)} \end{bmatrix}, \quad h^{(j,\ell)} \in \mathbb{R}^{p_\ell},$$

using sparse coding. In matrix form, this is equivalent to solving the problem

$$\mathsf{H}^* = \text{argmin}\big\{\|\mathsf{X}_{\text{test}} - \mathsf{WH}\|_F{}^2,\ \mathsf{H} \in \mathbb{R}^{p\times q}\big\} \text{ subject to } \mathsf{H} \text{ sparse}, \mathsf{H} \geqslant 0. \tag{4}$$

The vector $x^{(j)}$ is then assigned to the class that minimizes the Euclidian norm of the residual,

$$r_\ell(x^{(j)}) = \|x^{(j)} - \mathsf{W}_\ell h^{(j,\ell)}\|^2.$$

If the dictionary X is overcomplete, the solution of (4) may be numerically demanding.

In order to reduce the computational burden of the SRC algorithm, Metaface proposes to find smaller subdictionaries to represent the classes in the original dictionary. The subdictionary W_ℓ corresponding to the ℓth class is learned by solving a minimization problem of the form (1) over each class separately,

$$(\mathsf{W}_\ell^*, \mathsf{H}_\ell^*) = \operatorname{argmin}\left\{ \|\mathsf{X}_\ell - \mathsf{W}_\ell \mathsf{H}_\ell\|_F{}^2 \right\} \text{ subject to } \mathsf{W}_\ell, \mathsf{H}_\ell \geqslant 0,\ \mathsf{H}_\ell \text{ sparse},\ 1 \leqslant \ell \leqslant L.$$

A vector with unknown class attribute is then represented in terms of the learned dictionary $\mathsf{W}^* = [\mathsf{W}_1^*, \ldots, \mathsf{W}_L^*]$, and, as in SRC, assigned to the class that yields the smallest residual.

3. Sparsity, positivity, and hierarchical Bayesian models

The algorithm that we propose for computing non-negative and possibly sparse matrix factorizations is based on a hierarchical solver for linear inverse problems. Consider the linear inverse problem of estimating $x \in \mathbb{R}^n$ based on the observation

$$b = \mathsf{A}x + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \Sigma), \tag{5}$$

where $\mathsf{A} \in \mathbb{R}^{m\times n}$ with $m \leqslant n$ is the forward model operator, $x \in \mathbb{R}^n$ is the unknown of interest, $\Sigma \in \mathbb{R}^{m\times m}$ is the symmetric positive covariance matrix of the additive Gaussian noise ε, and $b \in \mathbb{R}^m$ is the indirect noisy observation. In the Bayesian framework, the likelihood density of b conditioned on x is of the form

$$\pi_{b|x}(b \mid x) \propto \exp\left(-\frac{1}{2}(b - \mathsf{A}x)^{\mathsf{T}}\Sigma^{-1}(b - \mathsf{A}x)\right) = \exp\left(-\frac{1}{2}\|\mathsf{S}b - \mathsf{S}\mathsf{A}x\|^2\right),$$

where $\Sigma^{-1} = \mathsf{S}^{\mathsf{T}}\mathsf{S}$ is a symmetric factorization of the noise precision matrix. To set up a prior model capable of accounting for possible sparsity and positivity, we first postulate a component-wise conditionally Gaussian prior model, $x_j|\theta_j \sim \mathcal{N}(0, \theta_j)$, $1 \leqslant j \leqslant n$, where θ_j is the prior variance of x_j, yielding a conditional prior model

$$\begin{aligned}\pi_{x|\theta}(x \mid \theta) &= \left(\frac{1}{2\pi}\right)^{n/2} \frac{1}{\theta_1^{1/2}\cdots\theta_n^{1/2}} \exp\left(-\frac{1}{2}\sum_{j=1}^n \frac{x_j^2}{\theta_j}\right) \\ &\propto \exp\left(-\frac{1}{2}\sum_{j=1}^n \frac{x_j^2}{\theta_j} - \frac{1}{2}\sum_{j=1}^n \log\theta_j\right).\end{aligned}$$

In order for the model to favor sparse solutions, we define a hyperprior model for the variances such that most of the time small positive values are favored, but occasional large outlier values are allowed. For reasons of computational convenience, distributions from the generalized gamma family have been proven to provide a versatile class of hyperpriors [7]. The model case is provided by the gamma distribution,

$$\pi_\theta(\theta) = \prod_{j=1}^n \pi_{\theta_j}(\theta_j), \quad \pi_{\theta_j}(\theta_j) = \frac{\theta_j^{\beta-1}}{\vartheta_j^\beta \Gamma(\beta)} \exp\left(-\frac{\theta_j}{\vartheta_j}\right), \tag{6}$$

where $\beta > 0$ is a shape parameter, and ϑ_j are scale parameters. Combining the likelihood model with the prior model, Bayes' formula yields the posterior distribution

$$\begin{aligned}\pi_{x,\theta|b} &\propto \pi_{b|x}(b \mid x)\pi_{x|\theta}(x \mid \theta)\pi_\theta(\theta)\\ &\propto \exp\left(-\frac{1}{2}\|\mathsf{S}b - \mathsf{SA}x\|^2 - \frac{1}{2}\sum_{j=1}^{n}\frac{x_j^2}{\theta_j} + \eta\sum_{j=1}^{n}\log\theta_j - \sum_{j=1}^{n}\frac{\theta_j}{\vartheta_j}\right),\\ \eta &= \beta - \frac{3}{2},\end{aligned} \tag{7}$$

where we require that $\beta > 3/2$ to guarantee that $\eta > 0$. The MAP estimate $(x_{\mathrm{MAP}}, \theta_{\mathrm{MAP}})$ is the maximizer of (7) or, equivalently, the minimizer of the Gibbs energy,

$$\mathscr{E}(x,\theta) = \overbrace{\|\mathsf{S}b - \mathsf{SA}x\|^2 + \sum_{j=1}^{n}\frac{x_j^2}{\theta_j}}^{\mathscr{E}(x|\theta)} \underbrace{- 2\sum_{j=1}^{n}\left(\eta\log\theta_j - \frac{\theta_j}{\vartheta_j}\right)}_{\mathscr{E}(\theta|x)}. \tag{8}$$

It was shown in [5, 14] that the Gibbs energy has a unique global minimum, and the minimizer can be found by the IAS algorithm, that proceeds by alternating the two minimization steps:

(a) update x, setting $x^t = \operatorname{argmin}\mathscr{E}(x \mid \theta^{t-1})$,
(b) update θ, setting $\theta^t = \operatorname{argmin}\mathscr{E}(\theta \mid x^t)$.

Observe that the updating step (a) is a standard least squares problem, while step (b) can be performed component-wise by solving the first order optimality condition

$$\frac{\partial}{\partial\theta_j}\mathscr{E}(\theta \mid x^t) = 0, \tag{9}$$

which has an explicit solution,

$$\theta_j^t = \frac{1}{2}\vartheta_j\left(\eta + \sqrt{\eta^2 + \frac{2(x_j^t)^2}{\vartheta_j}}\right), \quad 1 \leqslant j \leqslant n. \tag{10}$$

It has been proved in the cited articles that the parameter η controls the sparsity of the solution, and that at the limit as $\eta \to 0+$ the solution computed by the IAS algorithm converges to the ℓ_1-penalized least squares solution,

$$x_1 = \operatorname{argmin}\left\{\frac{1}{2}\|\mathsf{S}b - \mathsf{SA}x\|^2 + \sqrt{2}\sum_{j=1}^{n}\frac{|x_j|}{\sqrt{\vartheta_j}}\right\},$$

while for larger values of η, the minimizer resembles an ℓ_2-penalized least squares solution. In particular, for small η, the IAS can be thought as a computationally efficient alternative for ℓ_1-penalized regularization, or basis pursuit algorithms [16]. The selection of the scaling parameters ϑ_j are related to the noise level and sensitivity, see [14] for details. A computationally efficient Krylov subspace-based iterative solver for large least squares problems, e.g. a *prior-conditioned CGLS algorithm* (pCGLS) with an early stopping criterion can be used to approximate updates (a) of x [6]. It has been shown that in few iterations the IAS-pCGLS algorithm learns the sparsity structure of the unknown x, and takes advantage of it in the subsequent iterations, significantly reducing the computational burden. We omit the details, that can be found in the cited articles.

The IAS algorithm described above can extended to more general hyperprior models. More specifically, the gamma hyperprior (6) can be replaced by other heavy tailed distributions, for example by any member of the family of the generalized gamma distributions,

$$\pi_\theta(\theta) = \frac{|r|^n}{\Gamma(\beta)^n} \prod_{j=1}^{N} \frac{1}{\vartheta_j} \left(\frac{\theta_j}{\vartheta_j}\right)^{r\beta-1} \exp\left(-\left(\frac{\theta_j}{\vartheta_j}\right)^r\right), \quad r \neq 0. \tag{11}$$

By setting $r = 1$, (11) yields the gamma distribution, while $r = -1$ gives the inverse gamma distribution. The MAP estimate for generalized gamma hyperpriors were analyzed in detail in [7], where it was shown that while $r < 1$ favors more strongly sparse solutions than $r = 1$, the computation of the MAP estimate yields a non-convex optimization problem with no guarantee of unique minimizer. In particular, in the case where $r = -1$, the MAP estimate corresponds to the minimizer of the Gibbs energy, the counterpart of (8), of the form

$$\mathscr{F}(x,\theta) = \|\mathsf{S}b - \mathsf{SA}x\|^2 + \sum_{j=1}^{n} \frac{x_j^2}{\theta_j} - 2\sum_{j=1}^{n} \left(\kappa \log\theta_j - \frac{\vartheta_j}{\theta_j}\right), \quad \kappa = \beta + \frac{3}{2}.$$

In IAS, the updating step of x remains unaltered, while the first order optimality condition (9) yields an explicit solution in this case, too, given by

$$\theta_j = \frac{\vartheta_j}{2\kappa}\left(\frac{(x_j^t)^2}{\vartheta_j} + 2\right), \tag{12}$$

as is easy to verify.

Unlike the case $r = 1$ the hyperprior with $r = -1$ does not lead to a convex objective function, and therefore the IAS minimization algorithm may get trapped into local minima. To avoid local minima of suboptimal quality, it is important to start the iterations with $r = -1$ near a local minimum that has the same sparsity structure as the unique solution of the convex problem $r = 1$. A hybrid IAS algorithm, proposed in [8], starts the IAS iteration with $r = 1$, and once the solution is sufficiently close to the global minimizer, the hyperprior is switched to a greedier one, e.g. the inverse gamma model $r = -1$. To make the two prior models compatible, we require that the scaling parameters in the models are chosen so that the formulas (10) and (12) coincide when $x_j^t = 0$.

Finally, it was shown in [8] that by using a Yoshida–Moreau envelope, the algorithm can be easily modified to ensure that the solution is non-negative. Effectively, the positivity is enforced simply by projecting the iterates x^t onto the positive cone $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \mid x_j \geqslant 0\}$ at the end of each updating round. Further details can be found in the two cited articles. For completeness, we summarize the IAS iteration in an algorithmic form. Without loss of generality, we may assume that the linear problem (5) is whitened, that is $\Sigma = \mathsf{I}_m$, the $m \times m$ unit matrix.

IAS algorithm

Given: Matrix $\mathsf{A} \in \mathbb{R}^{m\times n}$, right hand side b, hyperparameters β and ϑ_j, $1 \leqslant j \leqslant n$, stopping parameters $\tau > 0$ and T.
Initialize: Set $\theta_j = \vartheta_j$, $\Delta_\theta = \infty$, $t = 1$.
Iterate until stopping criterion $t > T$ or $\Delta_\theta < \tau$ is met:

1. Update x: Solve the least squares problem
$$x^t = \operatorname{argmin}\left\{\|b - \mathsf{A}x\|^2 + \|\mathsf{D}_{\theta^{t-1}}^{-1/2}x\|^2\right\}, \quad \mathsf{D}_{\theta^{t-1}} = \operatorname{diag}(\theta^{t-1}).$$
2. Update θ: If $r = 1$, use formula (10), if $r = -1$, use (12).
3. **Optional:** Project onto the positive cone, setting $x^t = \max(0, x^t)$.
4. Compute
$$\Delta_\theta = \frac{\|\theta^t - \theta^{t-1}\|}{\|\theta^{t-1}\|}.$$
5. Check the convergence criterion, advance the counter $t \to t + 1$.

In the following section, we will write symbolically the MAP solution (x, θ) given (b, A) as

$$(x, \theta) = \mathrm{IAS}(b, \mathsf{A}),$$

and if, in addition, the non-negativity $x \geqslant 0$ is required and the optional step 3 in the IAS algorithm above is activated, we will write

$$(x, \theta) = \mathrm{IAS}_+(b, \mathsf{A}).$$

The choices of the parameter values are not explicitly indicated in this notation, but will be specified when we discuss computed examples.

3.1. IAS-based unsupervised dictionary learning: IAS-NMF

The IAS-based unsupervised dictionary learning algorithm with the full training set as dictionary, typically yielding an overcomplete frame to represent the data, has been discussed in detail in [25]. Here we restrict the discussion to the problem of learning an economy-size dictionary. In particular, when we restrict our attention to the case of non-negative data, the problem can be reduced to an NMF algorithm with the option of sparse coding.

In the description of the algorithm, the columns of a matrix are indexed by superscripts, e.g.

$$\mathsf{X} = [x^{(1)}, \ldots, x^{(p)}], \quad x^{(j)} \in \mathbb{R}^n,$$

while the rows are indexed by subscripts, that is,

$$\mathsf{X} = \begin{bmatrix} x_{(1)}^{\mathsf{T}} \\ \vdots \\ x_{(n)}^{\mathsf{T}} \end{bmatrix}, \quad x_{(j)} \in \mathbb{R}^p.$$

The IAS-based NMF algorithm can be summarized as follows:

IAS-NMF algorithm

Given: data matrix $\mathsf{X} \in \mathbb{R}^{n\times p}$, $\mathsf{X} \geqslant 0$, rank $k > 0$,
Initialize: $\mathsf{W}^0 \in \mathbb{R}^{n\times k}$, $t = 0$.
Iterate until stopping criterion is met:

1. Update H: Set $h^{(j)} = \mathrm{IAS}_+(x^{(j)}, \mathsf{W}^{(t)})$ for $1 \leqslant j \leqslant p$, and

$$\mathsf{H}^{(t+1)} = [h^{(1)}, \ldots, h^{(p)}];$$

2. Scale the rows of $\mathsf{H}^{(t+1)}$,

$$h_{(j)} \to \tfrac{h_{(j)}}{\|h_{(j)}\|_1}, \quad 1 \leqslant j \leqslant k.$$

3. Update W: Set $w_{(j)} = \mathrm{IAS}_+\left(x_{(j)}, \left(\mathsf{H}^{(t+1)}\right)^{\mathsf{T}}\right)$ for $1 \leqslant j \leqslant n$, and set

$$\mathsf{W}^{(t+1)} = \begin{bmatrix} w_{(1)}^{\mathsf{T}} \\ \vdots \\ w_{(n)}^{\mathsf{T}} \end{bmatrix};$$

4. Check the convergence criterion, advance the counter $t \to t+1$.

Observe that due to the identity

$$\mathsf{WH} = \mathsf{WLL}^{-1}\mathsf{H},$$

where $\mathsf{L} \in \mathbb{R}^{k\times k}$ is an arbitrary diagonal matrix with positive diagonal, the NMF algorithm leaves the freedom to scale either the columns of W or the rows of H. Here we choose to scale the rows, using the ℓ_1-norm that suits best to the application that will be discussed in the following section.

For the stopping criterion, we use the condition

$$\frac{\|\mathsf{W}^{(t)} - \mathsf{W}^{(t-1)}\|_F}{\|\mathsf{W}^{(t-1)}\|_F} + \frac{\|\mathsf{H}^{(t)} - \mathsf{H}^{(t-1)}\|_F}{\|\mathsf{H}^{(t-1)}\|_F} < \delta, \tag{13}$$

where δ is a threshold parameter.

We will write the algorithm in a functional form using the shorthand notation

$$(\mathsf{W}, \mathsf{H}) = \mathrm{IAS\_NMF}(\mathsf{X}, k),$$

with the understanding that the model parameters of the algorithm need to be specified as part of the input.

3.2. IAS-based SDL

We shall apply the IAS algorithm to SDL problems for classification, considering the SRC algorithm with the IAS optimization, and two slightly different NMF-based algorithms. The first algorithm is summarized in the following steps.

IAS-SRC algorithm

Given: annotated training data with L classes, $\mathsf{X}_{\text{train}} = [(\mathsf{X}_{\text{train}})_1, \ldots, (\mathsf{X}_{\text{train}})_L] \in \mathbb{R}^{n\times p}$, an unlabeled test data set $\mathsf{X} \in \mathbb{R}^{n\times q}$.
Define: the dictionary $\mathsf{W} = \mathsf{X}_{\text{train}}$, $\mathsf{W}_\ell = (\mathsf{X}_{\text{train}})_\ell \in \mathbb{R}^{n\times p_\ell}$, $1 \leqslant \ell \leqslant L$.
Classify: For each column $x^{(j)}$ of X,

1. Compute $h = \text{IAS}_+(x^{(j)}, \mathsf{W}) \in \mathbb{R}^p$, yielding the approximation

$$x^{(j)} \approx \sum_{\ell=1}^{L} \mathsf{W}_\ell h^{(\ell)}, \quad \text{where} \quad h = \begin{bmatrix} h^{(1)} \\ \vdots \\ h^{(L)} \end{bmatrix}, \quad h^{(\ell)} \in \mathbb{R}^{p_\ell}.$$

2. Classify $x^{(j)}$ to the class that minimizes the residual

$$r_\ell(x^{(j)}) = \|x^{(j)} - \mathsf{W}_\ell h^{(\ell)}\|.$$

For comparison, we then consider a version in which the full training set as dictionary is replaced by a reduced size learned dictionary similar to Metaface, and a slight modification of it, referred to here as a Metadictionary algorithm. The IAS-NMF algorithm, appropriately modified, can be used for SDL by applying it to the class-specific submatrices of the training data. We summarize the steps in the following algorithm.

IAS-Metaface algorithm

Given: annotated training data with L classes, $(\mathsf{X}_{\text{train}})_\ell$, $1 \leqslant \ell \leqslant L$, rank $k > 0$, an unlabeled test data set X.
For each class ℓ, $1 \leqslant \ell \leqslant L$, compute the NMF factorizations,

$$(\mathsf{W}_\ell, \mathsf{H}_\ell) = \text{IAS\_NMF}((\mathsf{X}_{\text{train}})_\ell, k), \quad 1 \leqslant \ell \leqslant L.$$

Write $\mathsf{W} = [\mathsf{W}_1, \ldots, \mathsf{W}_L]$.
Classify: For each column $x^{(j)}$ of X:

1. Compute $h = \text{IAS}_+(x^{(j)}, \mathsf{W}) \in \mathbb{R}^{kL}$, yielding the approximation

$$x^{(j)} \approx \sum_{\ell=1}^{L} \mathsf{W}_\ell h^{(\ell)}, \quad \text{where} \quad h = \begin{bmatrix} h^{(1)} \\ \vdots \\ h^{(L)} \end{bmatrix}, \quad h^{(\ell)} \in \mathbb{R}^{k}.$$

2. Classify $x^{(j)}$ to the class that minimizes the residual

$$r_\ell(x^{(j)}) = \|x^{(j)} - \mathsf{W}_\ell h^{(\ell)}\|.$$

The IAS-Metadictionary algorithm is obtained by a slight modification of the above one, whereby the vectors of unknown label are represented separately in terms of each subdictionary W_ℓ rather than merging the subdictionaries into a single dictionary W. This is explained in algorithmic form below.

IAS-Metadictionary algorithm

Given: annotated training data with L classes, $(\mathsf{X}_{\text{train}})_\ell$, $1 \leqslant \ell \leqslant L$, rank $k > 0$, an unlabeled test data set X.
For each class ℓ, $1 \leqslant \ell \leqslant L$, compute the NMF factorizations,

$$(\mathsf{W}_\ell, \mathsf{H}_\ell) = \text{IAS\_NMF}((\mathsf{X}_{\text{train}})_\ell, k), \quad 1 \leqslant \ell \leqslant L.$$

Classify: For each column $x^{(j)}$ of X:

1. Compute $h^{(\ell)} = \text{IAS}_+(x^{(j)}, \mathsf{W}_\ell) \in \mathbb{R}^k$, $1 \leqslant \ell \leqslant L$ yielding the approximations

$$x^{(j)} \approx \mathsf{W}_\ell h^{(\ell)}, \quad 1 \leqslant \ell \leqslant L.$$

2. Classify $x^{(j)}$ to the class that minimizes the residual

$$r_\ell(x^{(j)}) = \|x^{(j)} - \mathsf{W}_\ell h^{(\ell)}\|.$$

Thus, the difference between the two reduced dictionary algorithms is that in the latter, the hypothesis is that the best reduced approximation is obtained by using a learned subdictionary based on the correct class of the training data, avoiding possible ambiguities of combining features from different subdictionaries.

4. Computed examples

In this section, we apply the IAS-based dictionary learning algorithms to two different problems, hyperspectral imaging and classification of ECGs.

4.1. NMF-IAS versus K-SVD

To highlight the flexibility and special features of the algorithms, we start by a brief comparison with one of the state-of-the-art algorithms in dictionary learning, the K-SVD [1].

Standard NMF algorithms, e.g. the multiplicative updating (see, e.g. [21]) do not lead automatically to sparse representations, although sparsity-promoting versions have been proposed [22]. To address the sparsity, the K-SVD algorithm seeks a sparse coefficient matrix, while iteratively learning a compact dictionary, or 'code book', as it is referred to in [1]. We briefly review the idea of the algorithm. The algorithm is an alternating iterative algorithm, seeking a low rank approximation $\mathsf{X} \approx \mathsf{WH}$, where $\mathsf{W} \in \mathbb{R}^{n\times k}$, $\mathsf{H} \in \mathbb{R}^{k\times p}$, and k is decided by the user. Here, the coefficient matrix H is required to be sparse, and is resolved through a least squares problem by minimizing

$$F(h^{(j)}) = \|x^{(j)} - \mathsf{W}h^{(j)}\|^2, \quad 1 \leqslant j \leqslant p,$$

using a basis pursuit algorithm, W representing the current update of the dictionary. This rather standard step is followed by the dictionary update, which is characteristic to the algorithm: Denoting by $w^{(j)} \in \mathbb{R}^n$ the columns of W and by $h_{(j)}^{\mathsf{T}} \in \mathbb{R}^p$ the rows of H, the product WH is written as a sum of rank-one matrices, and the dictionary updating is performed entry-by-entry. Hence, by writing

$$\|\mathsf{X} - \mathsf{WH}\|_{\text{F}} = \|\big(\mathsf{X} - \sum_{j\neq k} w^{(j)} h_{(j)}^{\mathsf{T}}\big) - w^{(k)} h_{(k)}^{\mathsf{T}}\|_{\text{F}} = \|\mathsf{E}^{(k)} - w^{(k)} h_{(k)}^{\mathsf{T}}\|_{\text{F}},$$

we observe that updating the kth dictionary entry amounts to approximating the current error matrix $\mathsf{E}^{(k)}$ by a rank-one matrix. To not spoil the sparsity properties of the updated $h_{(k)}^{\mathsf{T}}$, the columns corresponding to zero entries in the current vector $h_{(k)}^{\mathsf{T}}$ are deleted from $\mathsf{E}^{(k)}$ and $h_{(k)}^{\mathsf{T}}$, leading to a reduced minimization problem,

$$\text{minimize } \|\widetilde{\mathsf{E}}^{(k)} - w^{(k)}\widetilde{h}_{(k)}^{\mathsf{T}}\|_{\mathrm{F}},$$

where $\widetilde{\mathsf{E}}^{(k)}$ and $\widetilde{h}^{(k)}$ are the reduced arrays. The optimal rank-one approximation of the reduced error matrix is now computed through SVD, leading to an update of $w^{(k)}$ and $h_{(k)}^{\mathsf{T}}$.

To demonstrate the flexibility of the IAS-based algorithm, we implement the K-SVD algorithm using the IAS algorithm instead of the basis pursuit ℓ_1-penalized minimization algorithm for updating H, followed by the SVD-based updating of the dictionary. We apply the algorithm to the standard handwritten digits NIST database, selecting for the training data the data vectors corresponding to handwritten digits '1', '8', and '0'. The training data X, containing the 16×16 images in its columns, is of size 256×432. The rank parameter is set $k = 20$. For comparison, we also run the IAS-NMF algorithm, using the same parameter values to update the coefficient matrix.

The two algorithms are run by using the Gamma hyperprior in the IAS step, with parameters

$$\eta_H = 10^{-3}, \quad \vartheta_H = 10^{-3}$$

in the common sparse coding step of updating H. In the IAS-NMF algorithm, the parameters for updating W are set at values

$$\eta_W = 10^{-4}, \quad \vartheta_W = 10^{-3}.$$

The stopping condition for both algorithm is given by the threshold parameter in (13) set to value $\delta = 0.1$.

Not surprisingly, the K-SVD reaches faster the stopping condition than NSF-IAS. In the test run corresponding to results shown here, the K-SVD required 8 iterations, while NSF-IAS converged in 28 iterations. Typically, the convergence of the latter requires 2-3 times more iterations. Likewise, the relative error of the approximation at the end of iterations,

$$e = \frac{\|\mathsf{X} - \mathsf{WH}\|_{\mathrm{F}}}{\|\mathsf{X}\|_{\mathrm{F}}},$$

is lower for K-SVD, $e = 0.28$ versus $e = 0.33$ for NMF-IAS, reflecting the optimal low-rank approximation property of SVD. A comparison of the reduced dictionaries found by the methods reveal the merits of the latter algorithm. In figure 1, the dictionary entries are plotted as 16×16 images. First, we observe that K-SVD does not produce non-negative dictionary entries as opposed to NMF-IAS. Second, the dictionary entries of K-SVD mix the features of underlying digits, while NMF-IAS gives feature vectors that are strongly associated to particular digits. Finally, unlike K-SVD, the IAS-NMF algorithm strongly promotes sparsity of the feature vectors. Summarizing, the interpretability of the latter dictionary is clearly better than the former.

4.2. Hyperspectral imaging

Hyperspectral imaging is a remote sensing application in which an airborne infrared/near infrared camera records the upwelling radiance of a ground target at several wavelengths over a spectral range, thus generating a stack of single frequency images. Another way of describing the data is to think that each pixel in the scenery corresponds to a spectrum of intensities

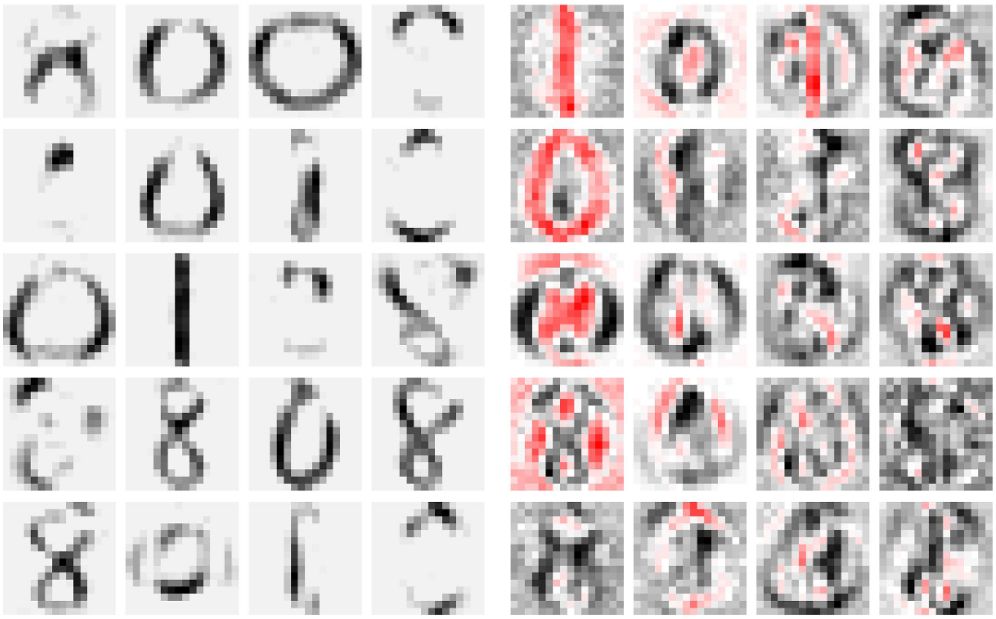

Figure 1. The feature vectors computed with IAS-NMF (left) and with the hybrid IAS-KSVD algorithm (right). In the former, the light gray background corresponds to zero, while in the latter, negative pixel values are indicated by red, positive by gray, white representing zero.

at different frequencies, the spectrum being a fingerprint, or a spectral signature of the pixel, carrying information about the properties of the corresponding patch of the ground target. The NMF algorithm has been used for blind hyperspectral unmixing, aiming at identifying materials in the captured scene through sparse representation of the spectra in terms of the atoms called endmembers, see, e.g. [3] for an overview. We demonstrate the efficiency of the IAS algorithm by applying it to an NMF problem of hyperspectral data.

The test data considered here is the 220 band airborne visible/infrared imaging spectrometer (AVIRIS) hyperspectral sensor data, consisting of a 145×145 pixel scenery of agricultural land, each pixel corresponding to 220 spectral channels, the wavelengths ranging from 400 to 2500 nanometers [2]. We arrange the spectra in a data matrix $\mathsf{X} \in \mathbb{R}^{n\times p}$, where $n = 220$, $p = (145)^2 = 21025$, and seek a low rank approximation

$$\mathsf{X} = \mathsf{WH}, \quad \mathsf{W} \in \mathbb{R}^{n\times k}, \quad \mathsf{H} \in \mathbb{R}^{k\times p},$$

where $\mathsf{W}, \mathsf{H} \geqslant 0$ and H is sparse. We run the IAS-NMF algorithm using the gamma hyperprior model (6). The algorithm is tested with two values of the rank parameter, $k = 3$ and $k = 20$. In both cases, the parameters controlling the sparsity are set to

$$\eta_H = 10^{-3}, \quad \eta_W = 10^{-1},$$

therefore promoting strongly the sparsity of H, while not requiring any sparsity from W. The scaling parameters are assumed constant with respect to the index j, and the constant values are set to

$$\vartheta_H = \vartheta_W = 10^{-3}.$$

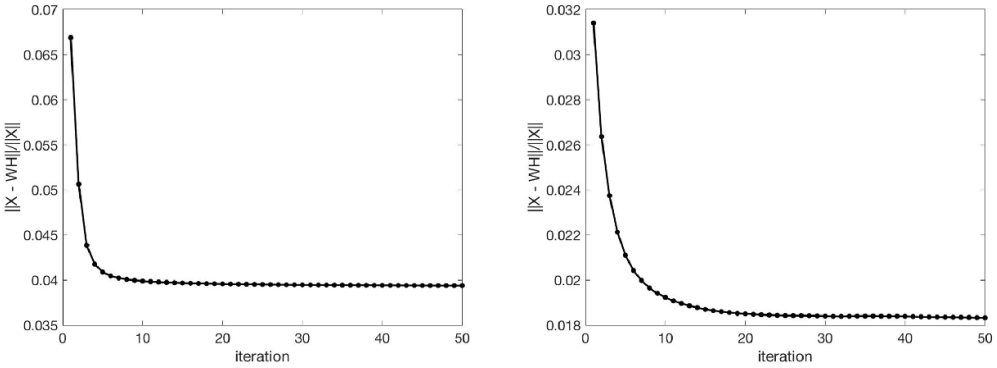

Figure 2. Convergence of the relative discrepancy (14) for $k=3$ (left) and for $k=20$ (right).

To initialize the NMF iterations we set $\mathsf{W}^0 \in \mathbb{R}^{n\times k}$ equal to k randomly selected spectra from the data matrix X. The standard deviation of the approximation error in both cases is set to $\sigma = 0.1$. We run a maximum of 50 iterations of the NMF algorithm for each choice of k, and choose a relatively loose stopping condition by setting the threshold parameter in (13) to $\delta = 0.1$. Figure 2 shows two realizations, corresponding to $k=3$ and $k=20$ respectively, of the Frobenius norm of the relative discrepancy,

$$d_t = \|\mathsf{X} - \mathsf{W}^t\mathsf{H}^t\|_F, \tag{14}$$

as a function of the iteration t, $1 \leqslant t \leqslant 50$.

To visualize the results, consider first the case $k=3$. For each pixel j, consider the relative importances of the feature vectors in the representation of the pixels and normalize the columns of H so that

$$\|h^{(j)}\|_1 = h_1^{(j)} + h_2^{(j)} + h_3^{(j)} = 1.$$

We then interpret the weight vectors $h^{(j)}$ as red-blue-green (RGB) color code triplets, thus using them to assign a color to each pixel. An interpretative color map is obtained by defining a triangle T with vertices $v^{(1)}$, $v^{(2)}$ and $v^{(3)}$. Each point v in the triangle can be represented in terms of the barycentric coordinates (ξ_1, ξ_2, ξ_3) as

$$v = \xi_1 v^{(1)} + \xi_2 v^{(2)} + \xi_3 v^{(3)} \in T, \quad \xi_1 + \xi_2 + \xi_3 = 1,$$

and interpreting the barycentric coordinates as RGB values, a unique color can be assigned to each point in the triangle. Interpreting the coefficients $h_i^{(j)}$ as barycentric coordinates, each pixel is therefore mapped into the color triangle in a unique way. Figure 3 shows the RGB triangle and the color coded map determined by the algorithm. Similar colors of different pixels indicate similar composition of the upwelling radiance, and therefore point towards similar properties of the ground targets.

For $k=20$, we scale the matrix H by the maximum of its entries and visualize the rows of H as images by interpreting the coefficient corresponding to a given pixel as the intensity level of the pixel. Figure 4 shows the rows of H as images, plotted by using the Matlab color scale 'parula', where dark blue corresponds to zero, and yellow to one. The predominance of blue in the images is an indication of the sparsity of the representation.

To further investigate the sparsity properties of the coefficient matrix, figure 5 shows (part) of the histogram of the values of the matrix H, scaled by it maximal entry $H_{\max} = \max h_{ij}$. We

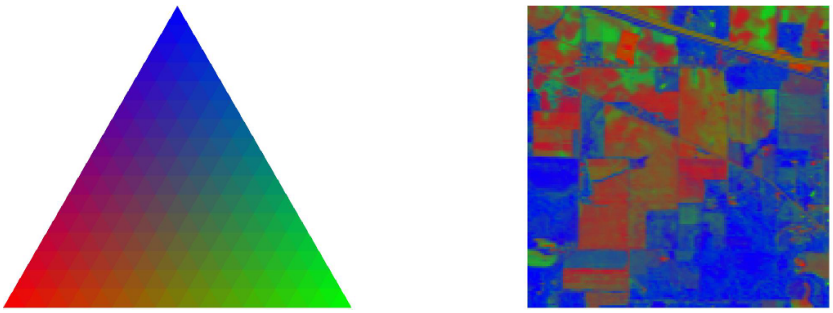

Figure 3. Graphical representation of the hyperspectral data using NMF with $k = 3$. Each pixel corresponds to an RGB vector that defines the color of the pixel (right). Since the sum of the RGB components is one, they may be interpreted as barycentric coordinates of a triangle. Coloring each point inside the triangle with a color corresponding to its barycentric coordinate (left), we get an interpretative map of the colors. Pixels with the same color correspond to the same point inside the triangle, indicating that the spectra of the pixels are close to each others.

observe that a significant portion, roughly 35%, of the entries are in the first bin, hence close to zero. The figure shows also the cumulative distribution of the values,

$$\mathrm{cdf}(\mathsf{H})(s) = \frac{1}{N}\#\{h_{ij} < sH_{\max}\}, \quad 0 < s \leqslant 1,$$

where $N = 20 \times (145)^2$ is the number of entries in H. The plot shows that approximately 90% of the pixel values are less than 20% of the maximal entry value, highlighting the algorithm's capability to explain the data with few feature vectors.

4.3. Classification of ECG data

In the second computed example, we consider the annotated data set ECG200 [38] consisting of ECGs of both normal heartbeats and heartbeats with a supraventricular premature beat. The ECG curves are all scaled to have an equal length of $n = 96$ arbitrary time units, thus defining the dimensionality of the data. The data contains a set of 100 ECG curves in the training set, and another set of 100 curves in the test set, see figure 6. The number of normal heartbeats in the training data and in the test data are 69 and 64, respectively, the corresponding number of abnormal heartbeats therefore being 31 and 36, respectively. We observe that the normal heartbeats have very similar means over the two sets and relatively little variability around the mean, while in the abnormal heartbeat data there is more variability. The task here is to use SDL for classifying the test data.

To test the IAS-based dictionary classification, we consider first the SRC-IAS classification ('training set as dictionary'). In this case, we need to find a sparse non-negative matrix H satisfying the minimization condition (4). To compute the coefficient matrix, we use the hybrid version of the IAS algorithm with the following parameter choices: For each column vector $x^{(j)}$, we first run a maximum of 50 outer iteration rounds, that is, 50 updates of the pair (x, θ), with the gamma hyperprior (6), stopping the iterations if the relative change in θ drops under the threshold of 1%,

$$\frac{\|\theta^t - \theta^{t-1}\|}{\|\theta^{t-1}\|} < \tau = 0.01. \tag{15}$$

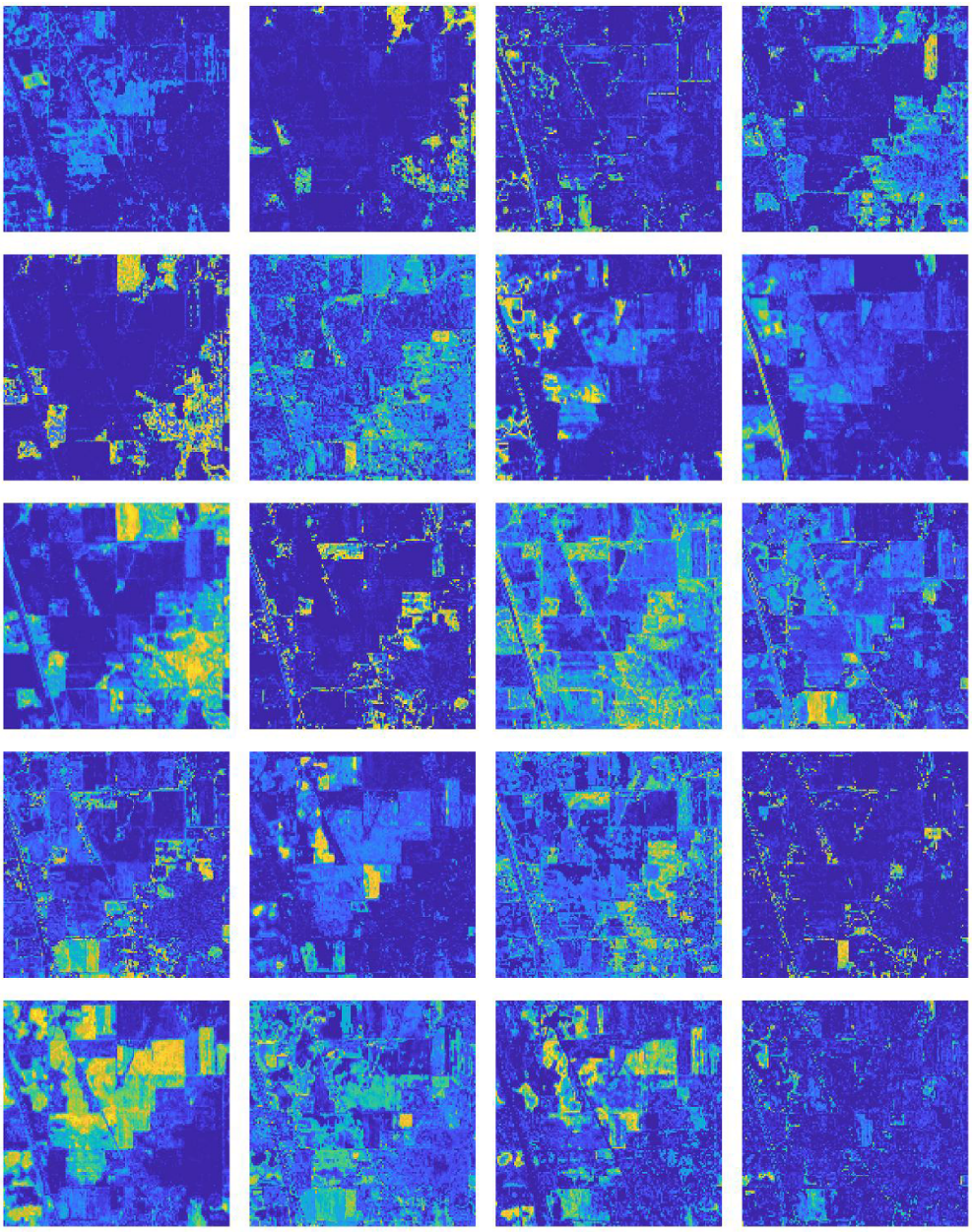

Figure 4. The rows of the matrix $\mathsf{H} \in \mathbb{R}^{20\times(145)^2}$ scaled by the maximum entry of the matrix, interpreted as 145×145 images, the entries defining the intensity level ($0 =$ dark blue, $1 =$ yellow). The sparsity, or compressibility, of H is reflected by the predominance of blue.

Using the computed θ as an input for each column vector, we then launch the more greedy IAS algorithm with inverse gamma hyperprior, (11) with $r = -1$, running a maximum of 50 outer iterations, again stopping when the relative change of θ is less than 1%. The hyperparameter

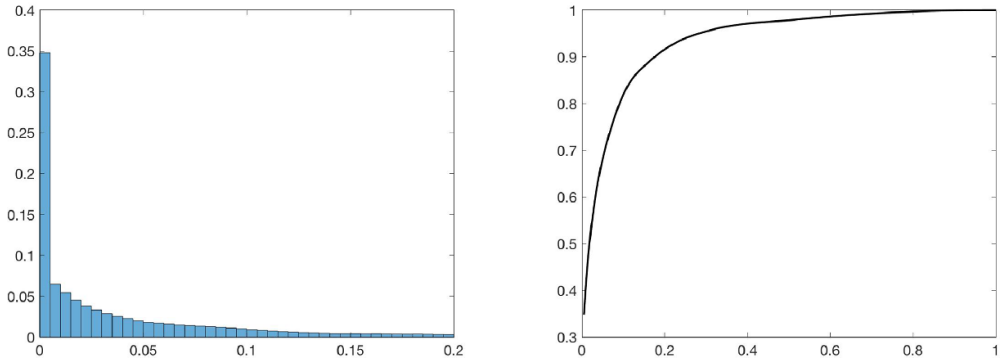

Figure 5. Left: histogram of the relative entry values of the matrix H, plotted over the interval $[0, 0.2]$. Right: cumulative distribution $\mathrm{cdf}(\mathsf{H})(s)$ of the entry values, showing that most of the entries are concentrated to the low end of the value range.

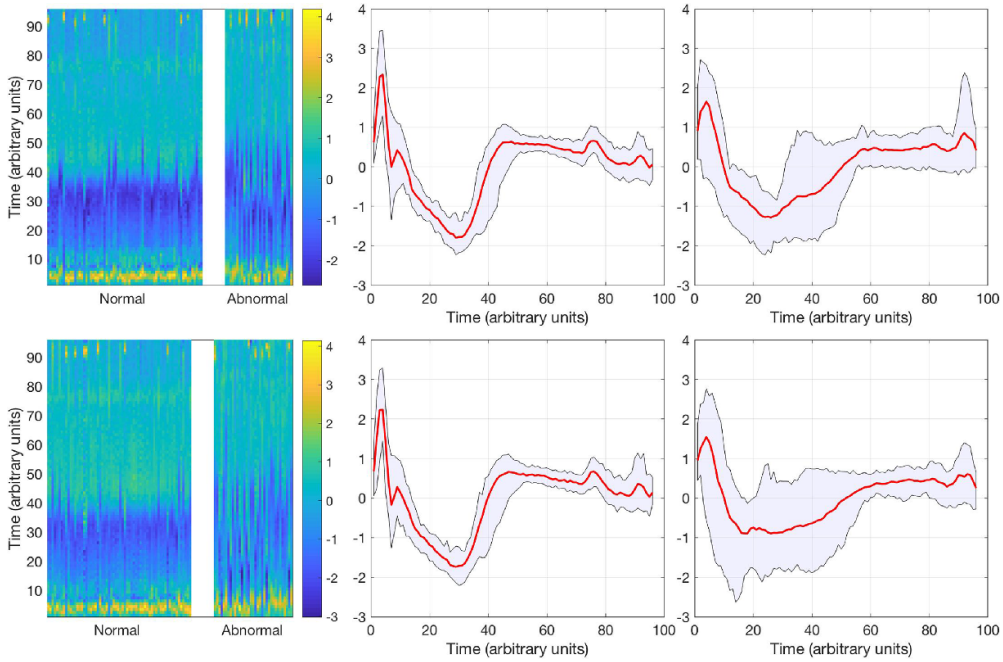

Figure 6. The training data (top row) consist of 100 electocardiograms scaled to the same time interval, with 69 corresponding to normal heartbeats and 31 to abnormal heartbeats with a supraventricular premature beat. The center panel shows the average over the normal heartbeats, with an envelope of 75% quantile. The right panel shows the mean and quantile envelopes corresponding to the abnormal beat. The bottom row corresponds to the test data, with 64 normal heartbeats and 36 abnormal ones.

values for the hybrid IAS were set to $(\eta, \vartheta) = (10^{-6}, 10^{-4})$ for the gamma hyperprior and $(\beta, \vartheta) = (1, 2.5 \times 10^{-10})$ for the inverse gamma hyperprior. We refer to [8] for the details on how the compatibility conditions guides the choice of the hyperparameters.

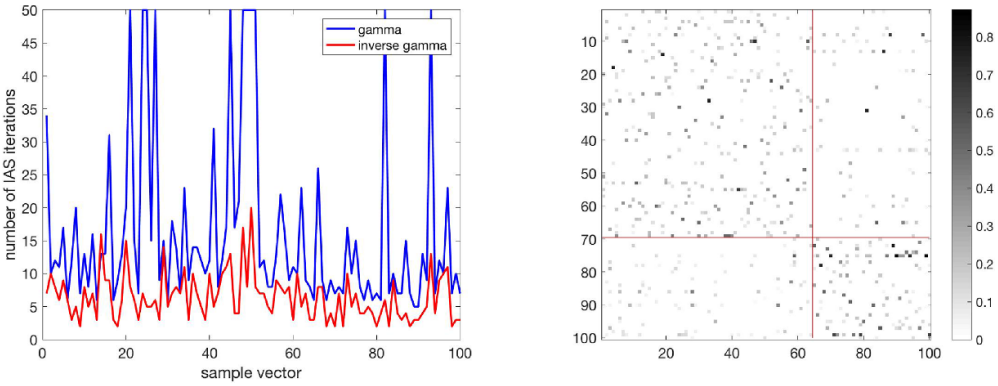

Figure 7. Left: number of IAS iterations needed to reach the stopping condition (15). Right: the coefficient matrix H. The dictionary and the test data are ordered so that the normal curves are at the beginning of the list, followed by the abnormal curves. The horizontal red line indicates the boundary between normal and abnormal ECG curves in the training data, and the vertical one the corresponding division line in the test data.

Figure 7 (left) shows the numbers of IAS iterations needed to meet the stopping criterion: The average number of iterations is 16.2 for the gamma hyperprior, and 6.4 for the inverse gamma hyperprior. Figure 7 (right) shows a black-and-white image of the matrix H, with white corresponding to zero, and black to the maximum ($\approx$0.87) of the absolute values of the entries of the matrix $\mathsf{H} \in \mathbb{R}^{p\times p}$ with $p = 100$. The red lines marks the division between normal and abnormal cases: Ideally, the off-diagonal blocks of the H-matrix should vanish, corresponding to the case of no confusion between the normal and abnormal curves. The predominance of white in the figure is an indication of the sparsity of the coefficient vectors. Indeed,

$$\frac{\#\{h_{ij} > 0.01 \times \max\{h_{ij}\}\}}{p^2} = 0.054,$$

that is, the matrix H is rather strongly compressible. The performance of the SRC classifier is summarized in the confusion matrix below, showing that the performance of the classifier is 89% correct classifications.

	Normal	Abnormal
Classified as normal	60	4
Classified as abnormal	7	29

In comparison, consider next the two reduced subdictionary algorithms applied to the ECG classification problem. In this particular example, there is no reason to expect that either of the subdictionaries $\mathsf{W}_1 \in \mathbb{R}^{n\times k}$ (normal heartbeat) and $\mathsf{W}_2 \in \mathbb{R}^{n\times k}$ (abnormal heartbeat), are sparse or non-negative, while the corresponding coefficient matrices $\mathsf{H}_1 \in \mathbb{R}^{k\times p}$ and $\mathsf{H}_2 \in \mathbb{R}^{n\times p}$ are both sparse and non-negative. The flexibility of the IAS algorithm allows us to choose the hyperparameters separately for the W-update and H-update so that in the former, sparsity is not requested. The values used in the IAS iterations are given in the table below. Observe that since we do not seek a sparse solution for W, we do not use the hybrid algorithm for updates of W, and therefore, the parameters of the second phase of the iteration are omitted.

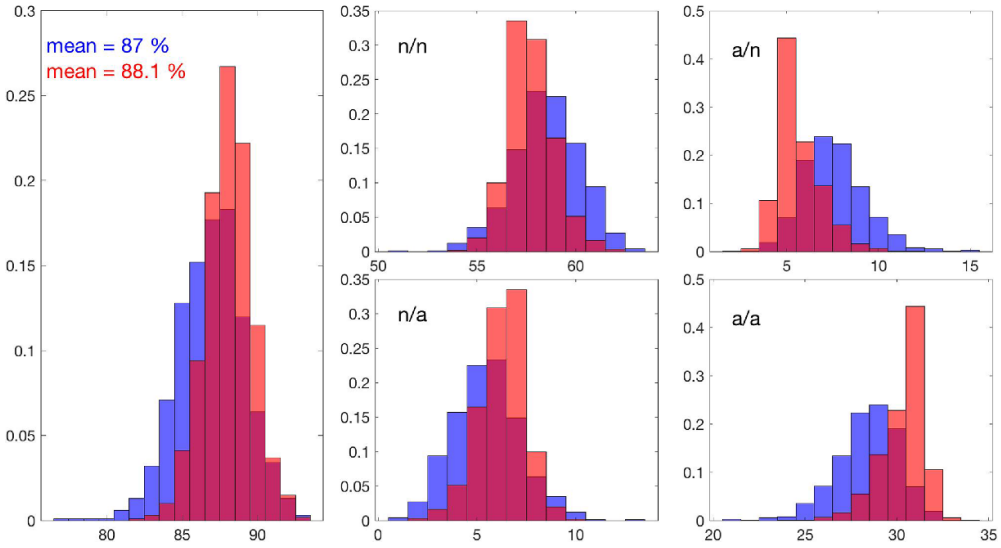

Figure 8. The left histogram shows the distribution of correct classifications over 1000 independent runs of the IAS-Metaface algorithm (blue) and the IAS-Metadictionary algorithm (red). On the right, the four histograms correspond to the distributions of the entries of the confusion matrix. The color coding is the same as in histogram on the left. Here, n/n stands for 'normal classified as normal', n/a as 'normal classified as abnormal' and so on.

	Phase 1: IAS with gamma				Phase 2: IAS with inverse gamma			
	η	ϑ	τ	T	β	ϑ	τ	T
H update	10^{-6}	10^{-4}	0.01	50	2.5×10^{-10}	1	0.01	50
W update	0.1	0.01	0.01	50	—	—	—	—

In our computed experiment, we choose the rank of the approximate matrix factorizations to be $k = 6$, that is, both normal and abnormal heartbeats are approximated by six feature vectors each. Due to the random initialization of the factorization algorithm, the classification result is also random, and in order to test the performance, we perform 1000 independent runs and report the statistics of the outcome. We set the stopping criterion of the NMF iteration to be rather loose, using $\delta = 0.05$ in the criterion (13). With this choice, the number of iterations in the NMF algorithm varies between 10 and 50. For comparison, we run the classification by using both the IAS-Metaface and IAS-Metadictionary algorithms.

The summary statistics of these runs are presented in figure 8, showing the histograms of correct classifications as well as histograms of the entries of the confusion matrices. We observe that from the point of view of overall classification performance, the IAS-Metadictionary algorithm performs slightly better, and the confusion matrix histograms reveal that this slight gain in performance is due to higher success rate in correctly classifying the abnormal heartbeats.

5. Conclusions

Dictionary learning algorithms are popular in a variety of applications requiring classification or interpretation of complex data as in the applications discussed in this article. Dictionary learning algorithms are becoming increasingly popular as an alternative to traditional parameter estimation problems, or inverse problems, often paired with a complex generative model performed by an algorithm in which the measured data are matched with computed outputs in a library [23, 26, 29]. An essential part of the algorithms is a sparse coding step and, as the dictionaries get significantly large, the computational efficiency becomes a key issue. The present article demonstrates how sparsity-promoting hierarchical Bayesian methods can be used to improve computational efficiency in sparse dictionary learning applications.

The computed examples in the article demonstrate the potential of the Bayesian sparse coding algorithms. Future research include finding rigorous and robust ways of setting some of the model parameters. In particular, while in traditional inverse problems, the covariance of the likelihood function can be related to the measurement noise level and possible modeling errors, in the present problem the observation model corresponds to the ansatz of representing the training data in terms of low rank matrix factorization, with no *a priori* knowledge of how well one can expect the approximation to hold, making it difficult to use concepts like signal-to-noise ratio to set hyperparameters as in the case of traditional inverse problems [14]. In the Bayesian context, the unknown likelihood covariance can be seen as an unknown, therefore subject to probabilistic modeling. It is an open problem to develop data-driven methods of estimating the covariance, although the use of hypermodels to estimate an unknown noise level have been shown to be promising for inverse problems [12]. In conclusion, while in the traditional context of Bayesian inverse problems, the role of hyperparameters of the Gamma or generalized Gamma hyperprior models is well analyzed and understood to be related to the *a priori* sparsity of the solution as well as to sensitivity of the data to the parameters, in the context of dictionary learning, establishing easily intepretable criteria for choosing the parameters remains a project for the future.

Data availability statement

No new data were created or analysed in this study.

Acknowledgment

The work of D.C. was partially funded by NSF-DMS 1951446. The work of E.S.was partially funded by NSF-DMS 2204618.

ORCID iDs

N Waniorek https://orcid.org/0000-0003-1731-6151
D Calvetti https://orcid.org/0000-0001-5696-718X
E Somersalo https://orcid.org/0000-0001-5099-3512

References

[1] Aharon M, Elad M and Bruckstein A 2006 K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation *IEEE Trans. Signal Process.* **54** 4311–22

[2] Baumgardner M F, Biehl L L and Landgrebe D A 2015 *220 Band AVIRIS Hyperspectral Image Data Set (Indian Pine Test Site 3: Purdue University Research Repository) (12 June 1992)*
[3] Bioucas-Dias J M, Plaza A, Dobigeon N, Parente M, Du Q, Gader P and Chanussot J 2012 Hyperspectral unmixing overview: geometrical, statistical and sparse regression-based approaches *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **5** 354–79
[4] Calvetti D, Hakula H, Pursiainen S and Somersalo E 2009 Conditionally Gaussian hypermodels for cerebral source localization *SIAM J. Imaging Sci.* **2** 879–909
[5] Calvetti D, Pascarella A, Pitolli F, Somersalo E and Vantaggi B 2015 A hierarchical Krylov–Bayes iterative inverse solver for MEG with physiological preconditioning *Inverse Problems* **31** 125005
[6] Calvetti D, Pitolli F, Somersalo E and Vantaggi B 2018 Bayes meets Krylov: statistically inspired preconditioners for CGLS *SIAM Rev.* **60** 429–61
[7] Calvetti D, Pragliola M, Somersalo E and Strang A 2020 Sparse reconstructions from few noisy data: analysis of hierarchical Bayesian models with generalized gamma hyperpriors *Inverse Problems* **36** 025010
[8] Calvetti D, Pragliola M and Somersalo E 2020 Sparsity promoting hybrid solvers for hierarchical Bayesian inverse problems *SIAM J. Sci. Comput.* **42** A3761–84
[9] Calvetti D and Somersalo E 2007 A Gaussian hypermodel to recover blocky objects *Inverse Problems* **23** 733
[10] Calvetti D and Somersalo E 2007 *An Introduction to Bayesian Scientific Computing: Ten Lectures on Subjective Computing* vol 2 (New York: Springer Science & Business Media)
[11] Calvetti D and Somersalo E 2008 Hypermodels in the Bayesian imaging framework *Inverse Problems* **24** 034013
[12] Calvetti D and Somersalo E 2010 Subjective knowldege or objective belief? *Large-Scale Inverse Problems and Quantification of Uncertainty* ed L Biegler, G Biros, O Ghattas, M Heinkenschloss, D Keyes, B Mallick, Y Marzouk, L Tenorio, B van Bloemen Waanders, K Willcox (New York: Wiley) p 3370
[13] Calvetti D and Somersalo E 2020 *Mathematics of Data Science: A Computational Approach to Clustering and Classification* (Philadelphia, PA: SIAM)
[14] Calvetti D, Somersalo E and Strang A 2019 Hierachical Bayesian models and sparsity: ℓ_2-magic *Inverse Problems* **35** 035003
[15] Candes E and Tao T 2005 Decoding by linear programming *IEEE Trans. Inf. Theory* **12** 4203–15
[16] Chen S, Donoho D and Saunders M 1998 Atomic decomposition by basis pursuit *SIAM J. Sci. Comput.* **20** 33–61
[17] Dong W, Li X, Zhang L and Shi G 2011 Sparsity-based image denoising via dictionary learning and structural clustering *Conf. Computer Vision and Pattern* (IEEE) pp 457–64
[18] Donoho D L 2006 Compressed sensing *IEEE Trans. Inf. Theory* **52** 1289–306
[19] Duarte-Carvajalino J M and Sapiro G 2009 Learning to sense sparse signals: simultaneous sensing matrix and sparsifying dictionary optimization *IEEE Trans. Image Process.* **18** 1395–408
[20] Elad M and Aharon M 2006 Image denoising via sparse and redundant representations over learned dictionaries IEEE *Trans. Image Process.* **15** 3736–45
[21] Gillis N 2020 *Nonnegative Matrix Factorization* (Philadelphia, PA: SIAM)
[22] Hoyer P O 2004 Non-negative matrix factorization with sparseness constraints *J. Mach. Learn. Res.* **5** 1457–69
[23] Ma D, Gulani V, Seiberlich N, Liu K, Sunshine J L, Duerk J L and Griswold M A 2013 Magnetic resonance fingerprinting *Nature* **495** 187–92
[24] Mallat S 2009 *A Wavelet Tour of Signal Processing: The Sparse Way* 3rd edn (New York: Academic)
[25] Pragliola M, Calvetti D and Somersalo E 2022 Overcomplete representation in a hierarchical Bayesian framework *Inverse Problems Imaging* **16** 19–38
[26] Saiz-Pérez A, Torres-Forné A and Font J A 2021 Classification of the core-collapse supernova explosion mechanism with learned dictionaries (arXiv:2110.12941)
[27] Sun X, Qu Q, Nasrabadi N M and Tran T D 2014 Structured priors for sparse-representation-based hyperspectral image classification *IEEE Geosci. Remote Sens. Lett.* **11** 235–1239
[28] Tibshirani R 1996 Regression shrinkage and selection via the lasso *J. R. Stat. Soc.* B **58** 267–88
[29] Torres-Forné A, Marquina A, Font J A and Ibanez J M 2016 Denoising of gravitational wave signals via dictionary learning algorithms *Phys. Rev.* D **94** 124040
[30] Tošić I and Frossard P 2011 Dictionary learning *IEEE Signal Process. Magn.* **28** 27–38
[31] Van Nguyen H, Patel V M, Nasrabadi N M and Chellappa R 2013 Design of non-linear kernel dictionaries for object recognition *IEEE Trans. Image Process.* **22** 5123–35

[32] Vu T H, Mousavi H S, Monga V, Rao U and Rao G 2015 DFDL: Discriminative feature-oriented dictionary learning for histopathological image classification *Proc. IEEE Int. Symp. on Biomedical Imaging* vol 35 pp 990–4
[33] Wang L, Zhao L, Bi G and Wan C 2014 Hierarchical sparse signal recovery by variational Bayesian inference *IEEE Signal Process. Lett.* **21** 110–13
[34] Wright J, Yang A Y, Ganesh A, Sastry S S and Ma Y 2009 Robust face recognition vis sparse representation *IEEE Trans. Pattern Anal. Mach. Intell.* **31** 210–27
[35] Xu Q, Yu H, Mou X, Zhang L, Hsieh J and Wang G 2012 Low-dose x-ray CT reconstruction via dictionary learning *IEEE Trans. Med. Imaging* **31** 1682–97
[36] Yang M, Zhang L, Yang J and Zhang D 2010 Metaface learning for sparse representation based face recognition *17th IEEE Int. Conf. Image Processing (ICIP)* pp 1601–4
[37] Yang L, Fang J, Cheng H and Li H 2017 Sparse Bayesian dictionary learning with a Gaussian hierarchical model *Signal Process.* **130** 93–104
[38] Moody G and Mark R MIT-BIH Arrhythmia Database 2005 (available at: https://physionet.org/content/mitdb/1.0.0/)