Learning Co-Sparse Analysis Operators with Separable Structures

Matthias Seibert, Julian Wörmann, Rémi Gribonval, Martin Kleinsteuber

Learning Co-Sparse Analysis Operators with Separable Structures

Matthias Seibert, *Student Member, IEEE,* Julian Wörmann, *Student Member, IEEE,*
Rémi Gribonval, *Fellow, IEEE,* and Martin Kleinsteuber, *Member, IEEE*

***Abstract*—In the co-sparse analysis model a set of filters is applied to a signal out of the signal class of interest yielding sparse filter responses. As such, it may serve as a prior in inverse problems, or for structural analysis of signals that are known to belong to the signal class. The more the model is adapted to the class, the more reliable it is for these purposes. The task of learning such operators for a given class is therefore a crucial problem. In many applications, it is also required that the filter responses are obtained in a timely manner, which can be achieved by filters with a separable structure.**

Not only can operators of this sort be efficiently used for computing the filter responses, but they also have the advantage that less training samples are required to obtain a reliable estimate of the operator.

The first contribution of this work is to give theoretical evidence for this claim by providing an upper bound for the sample complexity of the learning process. The second is a stochastic gradient descent (SGD) method designed to learn an analysis operator with separable structures, which includes a novel and efficient step size selection rule. Numerical experiments are provided that link the sample complexity to the convergence speed of the SGD algorithm.

***Index Terms*—Co-sparsity, separable filters, sample complexity, stochastic gradient descent**

I. Introduction

THE ability to sparsely represent signals has become standard practice in signal processing over the last decade. The commonly used synthesis approach has been extensively investigated and has proven its validity in many applications. Its closely related counterpart, the co-sparse analysis approach, was at first not treated with as much interest. In recent years this has changed and more and more work regarding the application and the theoretical validity of the co-sparse analysis model has been published. Both models assume that the signals $\mathbf{s}$ of a certain class are (approximately) contained in a union of subspaces. In the synthesis model, this reads as

$$\mathbf{s} \approx \mathbf{D}\mathbf{x}, \quad \mathbf{x} \text{ is sparse.} \tag{1}$$

In other words, the signal is a linear combination of a few columns of the synthesis dictionary $\mathbf{D}$. The subspace is determined by the indices of the non-zero coefficients of $\mathbf{x}$.

Opposed to that is the co-sparse analysis model

$$\mathbf{\Omega}\mathbf{s} \approx \boldsymbol{\alpha}, \quad \boldsymbol{\alpha} \text{ is sparse.} \tag{2}$$

$\mathbf{\Omega}$ is called the *analysis operator* and its rows represent filters that provide sparse responses. Here, the indices of the filters with zero response determine the subspace to which the signal belongs. This subspace is in fact the intersection of all hyperplanes to which these filters are normal vectors. Therefore, the information of a signal is encoded in its zero responses. In the following, $\boldsymbol{\alpha}$ is referred to as the *analyzed signal*.

While analytic analysis operators like the fused Lasso [1] and the finite differences operator, a close relative to the total variation [2], are frequently used, it is a well known fact that a concatenation of filters which is *adapted to a specific class of signals* produces sparser signal responses. Learning algorithms aim at finding such an optimal analysis operator by minimizing the average sparsity over a representative set of training samples. An overview of recently developed analysis operator learning schemes is provided in Section III.

Once an appropriate operator has been chosen there is a plethora of applications that it can be used for. Among these applications are regularizing inverse problems in imaging, cf. [3]–[6], where the co-sparsity is used to perform standard tasks such as image denoising or inpainting, bimodal super-resolution and image registration as presented in [7], where the joint sparsity of analyzed signals from different modalities is minimized, image segmentation as investigated in [8], where structural similarity is measured via the co-sparsity of the analyzed signals, classification as proposed in [9], where an SVM is trained on the co-sparse coefficient vectors of a training set, blind compressive sensing, cf. [10], where a co-sparse analysis operator is learned adaptively during the reconstruction of a compressively sensed signal, and finally applications in medical imaging, e.g., for structured representation of EEG signals [11] and tomographic reconstruction [12]. All these applications rely on the sparsity of the analyzed signal, and thus their success depends on how well the learned operator is adapted to the signal class.

An issue commonly faced by learning algorithms is that their performance rapidly decreases as the signal dimension increases. To overcome this issue some of the authors proposed separable approaches for both dictionary learning, cf. [13], and co-sparse analysis operator learning, cf. [14]. These separable approaches offer the advantage of a noticeably reduced nu-

The first two authors contributed equally to this work. This paper was presented in part at SPARS 2015, Cambridge, UK.

M. Seibert, J. Wörmann, and M. Kleinsteuber are with the Department of Electrical and Computer Engineering, TU München, Munich, Germany.
E-mail: {m.seibert,julian.woermann,kleinsteuber}@tum.de
Web: http://www.gol.ei.tum.de/

R. Gribonval heads the PANAMA project-team (Inria & CNRS), Rennes, France. E-mail: remi.gribonval@inria.fr

merical complexity. For example, for a separable operator for image patches of size $p\times p$ the computational burden for both learning and applying the filters is reduced from $\mathcal{O}(p^2)$ to $\mathcal{O}(p)$. We refer the reader to our previous work in [14] for a detailed introduction of separable co-sparse analysis operator learning.

In the paper at hand we show that separable analysis operators provide the additional benefit of requiring less samples during the training phase in order to learn a reliable operator. This is expressed via the sample complexity for which we provide a result for analysis operator learning, i.e., an upper bound η on the deviation of the expected co-sparsity w.r.t. the sample distribution and the average co-sparsity of a training set. Our main result presented in Theorem 9 in Section IV states that $\eta \propto C/\sqrt{N}$, where N is the number of training samples. The constant C depends on the constraints imposed on $\mathbf{\Omega}$ and we show that it is considerably smaller in the separable case. As a consequence, we are able to provide a generalization bound of an empirically learned analysis operator.

This generalization bound has a close relation to stochastic gradient descent methods. In Section V we introduce a geometric Stochastic Gradient Descent learning scheme for separable co-sparse analysis operators with a new variable step size selection that is based on the Armijo condition. The novel learning scheme is evaluated in Section VI. Our experiments confirm the theoretical results on sample complexity in the sense that separable analysis operator learning shows an improved convergence rate in the test scenarios.

II. Notation

Scalars are denoted by lower-case and upper-case letters α, n, N, column vectors are written as small bold-face letters $\boldsymbol{\alpha}, \mathbf{s}$, matrices correspond to bold-face capital letters $\mathbf{A}, \mathbf{S}$, and tensors are written as calligraphic letters $\mathcal{A}, \mathcal{S}$. This notation is consistently used for the lower parts of the structures. For example, the i^{th} column of the matrix $\mathbf{X}$ is denoted by $\mathbf{x}_i$, the entry in the i^{th} row and the j^{th} column of $\mathbf{X}$ is symbolized by x_{ij}, and for tensors $x_{i_1 i_2 \dots i_T}$ denotes the entry in $\mathcal{X}$ with the indices i_j indicating the position in the respective mode. Sets are denoted by blackletter script $\mathfrak{C}, \mathfrak{F}, \mathfrak{X}$.

For the discussion of multidimensional signals, we make use of the operations introduced in [15]. In particular, to define the way in which we apply the separable analysis operator to a signal in tensor form we require the k-mode product.

Definition 1. Given the T-tensor $\mathcal{S} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_T}$ and the matrix $\mathbf{\Omega} \in \mathbb{R}^{J_k \times I_k}$, their k-mode product is denoted by

$$\mathcal{S} \times_k \mathbf{\Omega}.$$

The resulting tensor is of the size $I_1 \times I_2 \times \dots \times I_{k-1} \times J_k \times I_{k+1} \times \dots \times I_T$ and its entries are defined as

$$(\mathcal{S} \times_k \mathbf{\Omega})_{i_1 i_2 \dots i_{k-1} j_k i_{k+1} \dots i_T} = \sum_{i_k=1}^{I_k} s_{i_1 i_2 \dots i_T} \cdot \omega_{j_k i_k}$$

for $j_k = 1, \dots, J_k$.

The k-mode product can be rewritten as a matrix-vector product using the Kronecker product $\otimes$ and the vec-operator that rearranges a tensor into a column vector such that

$$\begin{aligned} &\mathcal{A} = \mathcal{S} \times_1 \mathbf{\Omega}^{(1)} \times_2 \mathbf{\Omega}^{(2)} \dots \times_T \mathbf{\Omega}^{(T)} \\ \Leftrightarrow\ &\operatorname{vec}(\mathcal{A}) = (\mathbf{\Omega}^{(1)} \otimes \mathbf{\Omega}^{(2)} \otimes \dots \otimes \mathbf{\Omega}^{(T)}) \cdot \operatorname{vec}(\mathcal{S}). \end{aligned} \tag{3}$$

We also make use of the mapping

$$\begin{aligned} \iota \colon \mathbb{R}^{J_1 \times I_1} \times \dots \times \mathbb{R}^{J_T \times I_T} &\to \mathbb{R}^{\prod_k J_k \times \prod_k I_k} \\ (\mathbf{\Omega}^{(1)}, \dots, \mathbf{\Omega}^{(T)}) &\mapsto \mathbf{\Omega}^{(1)} \otimes \dots \otimes \mathbf{\Omega}^{(T)}. \end{aligned} \tag{4}$$

The remainder of notational comments, in particular those required for the discussion of the sample complexity, will be provided in the corresponding sections.

III. Related Work

As we have pointed out, learning an operator adapted to a class of signals yields a sparser representation than those provided by analytic filter banks. It thus comes as no surprise that there exists a variety of analysis operator learning algorithms, which we briefly review in the following.

In [16] the authors present an adaptation of the well known K-SVD dictionary learning algorithm to the co-sparse analysis operator setting. The training phase consists of two stages. In the first stage the rows of the operator that determine the subspace that each signal resides in are determined. In the subsequent stage each row of the operator is updated to be the vector that is "most orthogonal" to the signals associated with it. These two stages are repeated until a convergence criterion is met.

In [4] it is postulated that the analysis operator is a uniformly normalized tight frame, i.e., the columns of the operator are orthogonal to each other while all rows have the same ℓ_2-norm. Given noise contaminated training samples, an algorithm is proposed that outputs an analysis operator as well as noise free approximations of the training data. This is achieved by an alternating two stage optimization algorithm. In the first stage the operator is updated using a projected subgradient algorithm, while in the second stage the signal estimation is updated using alternating direction method of multipliers (ADMM).

A concept very similar to that of analysis operator learning is called sparsifying transform learning. In [17] a framework for learning overcomplete sparsifying transforms is presented. This algorithm consists of two steps, a sparse coding step where the sparse coefficient is updated by only retaining the largest coefficients, and a transform update step where a standard conjugate gradient method is used and the resulting operator is obtained by normalizing the rows.

The authors of [6] propose a method specialized on image processing. Instead of a patch-based approach, an image-based model is proposed with the goal of enforcing coherence across overlapping patches. In this framework, which is based on higher-order filter-based Markov Random Field models, all possible patches in the entire image are considered at once during the learning phase. A bi-level optimization scheme is proposed that has at its heart an unconstrained optimization problem w.r.t. the operator, which is solved using a quasi-Newton method.

Dong *et al.* [18] propose a method that alternates between a hard thresholding operation of the co-sparse representation and an operator update stage where all rows of the operator are simultaneously updated using a gradient method on the sphere. Their target function has the form $\|\mathbf{A}-\mathbf{\Omega S}\|_F^2$, where $\mathbf{A}$ is the co-sparse representation of the signal $\mathbf{S}$.

Finally, Hawe *et al.* [3] propose a geometric conjugate gradient algorithm on the product of spheres, where analysis operator properties like low coherence and full rank are incorporated as penalty functions in the learning process.

Except for our previous work [14], to our knowledge the only other analysis operator learning approach that offers a separable structure is proposed in [19] for the two-dimensional setting. Therein, an algorithm is developed that takes as an input noisy 2D patches $\hat{\mathbf{S}}_i$, $i = 1,\dots,N$ and then attempts to find $\mathbf{S}_i, \mathbf{\Omega}_1, \mathbf{\Omega}_2$ that minimize $\sum_{i=1}^N \|\mathbf{S}_i - \hat{\mathbf{S}}_i\|_F^2$ such that $\|\mathbf{\Omega}_1 \mathbf{S}_i \mathbf{\Omega}_2^\top\|_0 \leq l$, where l is a positive integer that serves as an upper bound on the number of non-zero entries, and the rows of $\mathbf{\Omega}_1$ and $\mathbf{\Omega}_2$ have unit norm. This problem is solved by alternating between a sparse coding stage and an operator update stage that is inspired by the work in [16] and relies on singular value decompositions.

While, to our knowledge, there are no sample complexity results for separable co-sparse analysis operator learning, results for many other matrix factorization schemes exist. Examples for this can be found in [20]–[22], among others. Specifically, [22] provides a broad overview of sample complexity results for various matrix factorizations. It is also of particular interest to our work since the sample complexity of separable dictionary learning is discussed. The bound derived therein has the form $c\sqrt{\beta \log(N)/N}$ where the driving constant β behaves proportional to $\sum_i p_i d_i$ for multidimensional data in $\mathbb{R}^{p_1 \times \dots \times p_T}$ and dictionaries $\mathbf{D}^{(i)} \in \mathbb{R}^{p_i \times d_i}$. This is an improvement over the non-separable result where the driving constant is proportional to the product over all i, i.e., $\beta \propto \prod_i p_i d_i$. The argumentation used to derive the results in [22] is different from the one we employ throughout this paper. While the results in [22] are derived by determining a Lipschitz constant and then using an argument based on covering numbers and concentration of measure, we follow a different approach. Following the work in [20], we employ McDiarmid's inequality in combination with a symmetrization argument and Rademacher averages. This approach offers better results when discussing tall matrices as in the case of co-sparse analysis operator learning.

IV. Sample Complexity

Co-sparse analysis operator learning aims at finding a set of filters concatenated in an operator $\mathbf{\Omega}$ which generates an optimal sparse representation $\mathbf{A} = \mathbf{\Omega S}$ of a set of training samples $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_N]$. This is achieved by solving the optimization problem

$$\arg\min_{\mathbf{\Omega} \in \mathfrak{C}} \tfrac{1}{N} \sum_{j=1}^N f(\mathbf{\Omega}, \mathbf{s}_j) \tag{5}$$

where $f(\mathbf{\Omega}, \mathbf{s}) = g(\mathbf{\Omega s}) + p(\mathbf{\Omega})$ with the sparsity promoting function g and the penalty function p. By restricting $\mathbf{\Omega}$ to the constraint set $\mathfrak{C}$ it is ensured that certain trivial solutions are avoided, see e.g. [4]. The additional penalty function is used to enforce more specific constraints. We will discuss appropriate choices of constraint sets at a later point in this section while the penalty function will be concretized in Section V.

Before we can provide our main theoretical result, we first introduce several concepts from the field of statistics in order to make this work self-contained.

A. Rademacher & Gaussian Complexity

In the following, we consider the set of samples $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_N]$, $\mathbf{s}_i \in \mathfrak{X}$, where each sample is drawn according to an underlying distribution $\mathbb{P}$ over $\mathfrak{X}$. Furthermore, given the above defined function $f\colon \mathfrak{C} \times \mathfrak{X} \to \mathbb{R}$, we consider the class $\mathfrak{F} = \{f(\mathbf{\Omega}, \cdot) : \mathbf{\Omega} \in \mathfrak{C}\}$ of functions that map the sample space $\mathfrak{X}$ to $\mathbb{R}$. We are interested in finding the function $f \in \mathfrak{F}$ for which the expected value

$$\mathbb{E}[f] \coloneqq \mathbb{E}_{\mathbf{s} \sim \mathbb{P}}[f(\mathbf{\Omega}, \mathbf{s})]$$

is minimal. However, due to the fact that the distribution $\mathbb{P}$ of the data samples is not known, in general, it is not possible to determine the optimal solution to this problem and we are limited to finding a minimizer of the empirical mean for a given set of N samples $\mathbf{S}$ drawn according to the underlying distribution. The empirical mean is defined as

$$\hat{\mathbb{E}}_{\mathbf{S}}[f] \coloneqq \tfrac{1}{N} \sum_{i=1}^N f(\mathbf{\Omega}, \mathbf{s}_i).$$

In order to evaluate how well the empirical problem approximates the expectation we pursue an approach that relies on the Rademacher complexity. We use the definition introduced in [23].

Definition 2. Let $\mathfrak{F} \subset \{f(\mathbf{\Omega}, \cdot) : \mathbf{\Omega} \in \mathfrak{C}\}$ be a family of real valued functions defined on the set $\mathfrak{X}$. Furthermore, let $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_N]$ be a set of samples with $\mathbf{s}_i \in \mathfrak{X}$. The *empirical Rademacher complexity* of $\mathfrak{F}$ with respect to the set of samples $\mathbf{S}$ is defined as

$$\hat{R}_{\mathbf{S}}(\mathfrak{F}) \coloneqq \mathbb{E}_\sigma \left[\sup_{f \in \mathfrak{F}} \tfrac{1}{N} \sum_{i=1}^N \sigma_i f(\mathbf{\Omega}, \mathbf{s}_i) \right]$$

where $\sigma_1, \dots, \sigma_N$ are independent Rademacher variables, i.e., random variables with $Pr(\sigma_i = +1) = Pr(\sigma_i = -1) = 1/2$ for $i = 1, \dots, N$.

This definition differs slightly from the standard one, where the absolute value of the argument within the supremum is taken, cf. [24]. Both definitions coincide when $\mathfrak{F}$ is closed under negation, i.e., when $f \in \mathfrak{F}$ implies $-f \in \mathfrak{F}$. As proposed in [23], the definition of the empirical Rademacher complexity as above has the property that it is dominated by the standard empirical Rademacher complexity. Furthermore, $\hat{R}_{\mathbf{S}}(\mathfrak{F})$ vanishes when the function class $\mathfrak{F}$ consists of a single constant function.

Definition 2 is based on a fixed set of training samples $\mathbf{S}$. However, we are generally interested in the correlation of $\mathfrak{F}$ with respect to a distribution $\mathbb{P}$ over $\mathfrak{X}$. This encourages the following definition.

Definition 3. Let $\mathfrak{F}$ be as before and $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_N]$ be a set of samples $\mathbf{s}_i$, $i = 1, \dots, N$ drawn i.i.d. according to a predefined probability distribution $\mathbb{P}$. Then the *Rademacher complexity* of $\mathfrak{F}$ is defined as

$$R_N(\mathfrak{F}) \coloneqq \mathbb{E}_{\mathbf{S}}[\hat{R}_{\mathbf{S}}(\mathfrak{F})].$$

With these definitions it is possible to provide generalization bounds for general function classes. Examples for this can be found in [24]. In addition to the Rademacher complexity, another measure of complexity is required to obtain bounds for our concrete case at hand. As before, the definition used here slightly differs from the standard definition, which can be found in [24].

Definition 4. Let $\mathfrak{F} \subset \{f(\mathbf{\Omega}, \cdot) : \mathbf{\Omega} \in \mathfrak{C}\}$ be a family of real valued functions defined on the set $\mathfrak{X}$. Furthermore, let $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_N]$ be a set of samples with $\mathbf{s}_i \in \mathfrak{X}$. Then the *empirical Gaussian complexity* of the function class $\mathfrak{F}$ is defined as

$$\hat{G}_{\mathbf{S}}(\mathfrak{F}) = \mathbb{E}_\gamma \left[\sup_{f \in \mathfrak{F}} \tfrac{1}{N} \sum_{i=1}^{N} \gamma_i f(\mathbf{\Omega}, \mathbf{s}_i) \right]$$

where $\gamma_1, \dots, \gamma_N$ are independent Gaussian $\mathcal{N}(0,1)$ random variables. The *Gaussian complexity* of $\mathfrak{F}$ is defined as

$$G_N(\mathfrak{F}) = \mathbb{E}_{\mathbf{S}} \left[\hat{G}_{\mathbf{S}}(\mathfrak{F}) \right].$$

Based on the similar construction of Rademacher and Gaussian complexity it is not surprising that it is possible to prove that they fulfill a similarity condition. For us, it is only of interest to upper bound the Rademacher complexity with the Gaussian complexity.

Lemma 5. *Let $\mathfrak{F}$ be a class of functions mapping from $\mathfrak{X}$ to $\mathbb{R}$. For any set of samples $\mathbf{S}$, the empirical Rademacher complexity can be upper bounded with the empirical Gaussian complexity via*

$$\hat{R}_{\mathbf{S}}(\mathfrak{F}) \leq \sqrt{\pi/2} \cdot \hat{G}_{\mathbf{S}}(\mathfrak{F}).$$

This can be seen by noting that $\mathbb{E}[|\gamma_i|] = \sqrt{2/\pi}$ and by using Jensen's inequality, cf. [25].

B. Generalization Bound for Co-Sparse Analysis Operator Learning

In this section we provide the concrete bounds for the sample complexity of co-sparse analysis operator learning. At the beginning of Section IV we briefly mentioned that the key role of constraint sets is to avoid trivial solutions [4]. A very simple constraint, that achieves this goal is to require that each row of the learned operator has unit ℓ_2-norm, cf. [3], [4]. The set of all matrices that fulfills this property has a manifold structure and is often referred to as the *oblique manifold*

$$\mathrm{Ob}(m,p) = \{\mathbf{\Omega} \in \mathbb{R}^{m \times p} : (\mathbf{\Omega}\mathbf{\Omega}^\top)_{ii} = 1,\ i = 1, \dots, m\}. \tag{6}$$

This is the constraint set we employ for *non-separable* operator learning, which we want to distinguish from learning operators with separable structure.

A separable structure on the operator is enforced by further restricting the constraint set to the subset $\{\mathbf{\Omega} \in \mathrm{Ob}(m,p) : \mathbf{\Omega} = \iota(\mathbf{\Omega}^{(1)}, \dots, \mathbf{\Omega}^{(T)}),\ \mathbf{\Omega}^{(i)} \in \mathrm{Ob}(m_i, p_i)\}$ with the appropriate dimensions $m = \prod_i m_i$ and $p = \prod_i p_i$. The mapping ι is defined in Equation (4). The fact that $\iota(\mathbf{\Omega}^{(1)}, \dots, \mathbf{\Omega}^{(T)})$ is an element of $\mathrm{Ob}(m,p)$ is readily checked. While ι is not bijective onto $\mathrm{Ob}(m,p)$, this does not pose a problem for our scenario. This way of expressing separable operators is related to signals $\mathcal{S}$ in tensor form via

$$\iota(\mathbf{\Omega}^{(1)}, \dots, \mathbf{\Omega}^{(T)})\mathbf{s} = \mathrm{vec}^{-1}(\mathbf{s}) \times_1 \mathbf{\Omega}^{(1)} \dots \times_T \mathbf{\Omega}^{(T)}$$

with $\mathrm{vec}^{-1}(\mathbf{s}) = \mathcal{S}$.

To provide concrete results we will require the ability to bound the absolute value of the realization of a function to its expectation. We use McDiarmid's inequality, cf. [26], to tackle this task.

Theorem 6 (McDiarmid's Inequality)**.** *Suppose $X_1, \dots, X_N$ are independent random variables taking values in a set $\mathfrak{X}$ and assume that $f : \mathfrak{X}^N \to \mathbb{R}$ satisfies*

$$\sup_{x_1, \dots, x_N, \hat{x}_i} |f(x_1, \dots, x_N) - f(x_1, \dots, x_{i-1}, \hat{x}_i, x_{i+1}, \dots, x_N)| \leq c_i \qquad \text{for } 1 \leq i \leq N.$$

It follows that for any $\varepsilon > 0$

$$Pr(\mathbb{E}[f(X_1, \dots, X_N)] - f(X_1, \dots, X_N) \geq \varepsilon) \leq \exp\left(-\frac{2\varepsilon^2}{\sum_{i=1}^N c_i^2} \right).$$

We are now ready to state a preliminary result.

Lemma 7. *Let $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_N]$ be a set of samples independently drawn according to a distribution within the unit ℓ_2-ball in $\mathbb{R}^p$. Let $f(\mathbf{\Omega}, \mathbf{s}) = g(\mathbf{\Omega}\mathbf{s}) + p(\mathbf{\Omega})$ as previously defined where the sparsity promoting function g is λ-Lipschitz. Finally, let the function class $\mathfrak{F}$ be defined as $\mathfrak{F} = \{f(\mathbf{\Omega}, \cdot) : \mathbf{\Omega} \in \mathrm{Ob}(m,p)\}$. Then the inequality*

$$\mathbb{E}[f] - \hat{\mathbb{E}}_{\mathbf{S}}[f] \leq \sqrt{2\pi}\ \hat{G}_{\mathbf{S}}(\mathfrak{F}) + 3\sqrt{\frac{2\lambda^2 m \ln(2/\delta)}{N}}, \tag{7}$$

holds with probability greater than $1 - \delta$.

Proof. When considering the difference $\mathbb{E}[f] - \hat{\mathbb{E}}_{\mathbf{S}}[f]$ the penalty function p can be omitted since it is independent of the samples and therefore cancels out. Now, in order to bound the difference $\mathbb{E}[f] - \hat{\mathbb{E}}_{\mathbf{S}}[f]$ for all $f \in \mathfrak{F}$, we consider the equivalent problem of bounding $\sup_{f \in \mathfrak{F}}(\mathbb{E}[f] - \hat{\mathbb{E}}_{\mathbf{S}}[f])$. To do this we introduce the random variable

$$\Phi(\mathbf{S}) = \sup_{f \in \mathfrak{F}}(\mathbb{E}[f] - \hat{\mathbb{E}}_{\mathbf{S}}[f]).$$

The next step is to use McDiarmid's inequality to bound $\Phi(\mathbf{S})$. Since $\mathbf{\Omega}$ is an element of the constraint set $\mathrm{Ob}(m,p)$, its largest singular value is bounded by $\sqrt{m}$. Furthermore, due to the assumptions that g is λ-Lipschitz and $\|\mathbf{s}_i\|_2 \leq 1$, the function value of $f(\mathbf{\Omega}, \mathbf{s})$ changes by at most $2\lambda\sqrt{m}$ when varying $\mathbf{s}$. Since f appears within the empirical average in Φ, we get the result that the function value of Φ varies by at most $2\lambda\sqrt{m}/N$ when changing a single sample in the set $\mathbf{S}$. Thus, McDiarmid's inequality stated in Theorem 6 with a

target probability of δ yields the bound

$$\Phi(\mathbf{S}) \leq \mathbb{E}_{\mathbf{S}}[\Phi(\mathbf{S})] + \sqrt{\frac{2\lambda^2 m \ln(1/\delta)}{N}} \tag{8}$$

with probability greater than $1-\delta$. By using a standard symmetrization argument, cf. [27], and another instance of McDiarmid's inequality we can then first upper bound $\mathbb{E}_{\mathbf{S}}[\Phi(\mathbf{S})]$ by $2R_N(\mathfrak{F})$ and then by $2\hat{R}_{\mathbf{S}}(\mathfrak{F})$, yielding

$$\sup_{f\in\mathfrak{F}}(\mathbb{E}[f] - \hat{\mathbb{E}}_{\mathbf{S}}[f]) \leq 2\hat{R}_{\mathbf{S}}(\mathfrak{F}) + 3\sqrt{\frac{2\lambda^2 m \ln(2/\delta)}{N}}$$

with probability greater than $1-\delta$. A more detailed derivation of these bounds can be found in the appendix. Lemma 5 then provides the proposed bound. □

The last ingredient for the proof of our main theorem is Slepian's Lemma, cf. [28], which is used to provide an estimate for the expectation of the supremum of a Gaussian process.

Lemma 8 (Slepian's Lemma)**.** *Let X and Y be two centered Gaussian random vectors in $\mathbb{R}^N$ such that*

$$\mathbb{E}[|Y_i - Y_j|^2] \leq \mathbb{E}[|X_i - X_j|^2] \quad \text{for } i \neq j.$$

Then

$$\mathbb{E}\left[\sup_{1\leq 1\leq N} Y_i\right] \leq \mathbb{E}\left[\sup_{1\leq i\leq N} X_i\right].$$

With all the preliminary work taken care of we are now able to state and prove our main results.

Theorem 9. *Let $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_N]$ be a set of samples independently drawn according to a distribution within the unit ℓ_2-ball in $\mathbb{R}^p$. Let $f(\mathbf{\Omega}, \mathbf{s}) = g(\mathbf{\Omega}\mathbf{s}) + p(\mathbf{\Omega})$ as previously defined where the sparsity promoting function g is λ-Lipschitz. Finally, let the function class $\mathfrak{F}$ be defined as $\mathfrak{F} = \{f(\mathbf{\Omega}, \cdot) : \mathbf{\Omega} \in \mathfrak{C}\}$, where $\mathfrak{C}$ is either $\mathrm{Ob}(m,p)$ for the non-separable case or the subset $\{\mathbf{\Omega} \in \mathrm{Ob}(m,p) : \mathbf{\Omega} = \iota(\mathbf{\Omega}^{(1)}, \dots, \mathbf{\Omega}^{(T)}), \mathbf{\Omega}^{(i)} \in \mathrm{Ob}(m_i, p_i)\}$ for the separable case. Then we have*

$$\mathbb{E}[f] - \hat{\mathbb{E}}_{\mathbf{S}}[f] \leq \sqrt{2\pi}\,\frac{\lambda C_{\mathfrak{C}}}{\sqrt{N}} + 3\sqrt{\frac{2\lambda^2 m \ln(2/\delta)}{N}} \tag{9}$$

with probability at least $1-\delta$, where $C_{\mathfrak{C}}$ is a constant that depends on the constraint set. In the non-separable case the constant is defined as $C_{\mathfrak{C}} = m\sqrt{p}$, whereas in the separable case it is given as $C_{\mathfrak{C}} = \sum_i m_i\sqrt{p_i}$.

Proof. Given the results of Lemma 7 it remains to upper bound the empirical Gaussian complexity. We discuss the two considered constraint sets separately in the following.

Non-Separable Operator: In order to find a bound for $\hat{G}_{\mathbf{S}}(\mathfrak{F})$ we define the two Gaussian processes $G_{\mathbf{\Omega}} = \frac{1}{N}\sum_{i=1}^N \gamma_i f(\mathbf{\Omega}, \mathbf{s}_i)$ and $H_{\mathbf{\Omega}} = \frac{\lambda}{\sqrt{N}}\langle \mathbf{\Xi}, \mathbf{\Omega}\rangle_F = \frac{\lambda}{\sqrt{N}}\sum_{i,j}\xi_{ij}\omega_{ij}$ with γ_i and ξ_{ij} i.i.d. Gaussian random variables. These two processes fulfill the condition

$$\mathbb{E}_\gamma[|G_{\mathbf{\Omega}} - G_{\mathbf{\Omega}'}|^2] \leq \tfrac{\lambda^2}{N}\|\mathbf{\Omega} - \mathbf{\Omega}'\|_F^2 = \mathbb{E}_\xi[|H_{\mathbf{\Omega}} - H_{\mathbf{\Omega}'}|^2],$$

where the inequality holds since $f(\mathbf{\Omega}, \mathbf{s}_i)$ is λ-Lipschitz w.r.t. the Frobenius norm in its first component when omitting the penalty term, i.e.,

$$|f(\mathbf{\Omega}, \mathbf{s}_i) - f(\mathbf{\Omega}', \mathbf{s}_i)| = |g(\mathbf{\Omega}\mathbf{s}_i) - g(\mathbf{\Omega}'\mathbf{s}_i)| \leq \lambda\|\mathbf{\Omega} - \mathbf{\Omega}'\|_F$$

for all $\mathbf{\Omega}, \mathbf{\Omega}' \in \mathfrak{C}$. Thus, we can apply Slepian's Lemma, cf. Lemma 8, which provides the inequality

$$\mathbb{E}_\gamma[\sup_{\mathbf{\Omega}\in\mathfrak{C}} G_{\mathbf{\Omega}}] \leq \mathbb{E}_\xi[\sup_{\mathbf{\Omega}\in\mathfrak{C}} H_{\mathbf{\Omega}}]. \tag{10}$$

Note, that the left-hand side of this inequality is the empirical Gaussian complexity of our learning problem. Considering the constraint set $\mathfrak{C}$ the expression on the right-hand side can be bounded via

$$\begin{aligned}\mathbb{E}_\xi[\sup H_{\mathbf{\Omega}}] &= \tfrac{\lambda}{\sqrt{N}}\mathbb{E}_\xi[\sup_{\mathbf{\Omega}\in\mathfrak{C}}\langle \mathbf{\Xi}, \mathbf{\Omega}\rangle_F] \\ &= \tfrac{\lambda}{\sqrt{N}}\mathbb{E}_\xi[\sum_{j=1}^m \|\boldsymbol{\xi}_j\|_2] \leq \tfrac{\lambda}{\sqrt{N}} m\sqrt{p}.\end{aligned}$$

Here, $\boldsymbol{\xi}_j \in \mathbb{R}^p$, $j = 1, \dots, m$ denotes the transposed of the j-th row of $\mathbf{\Xi}$.

Separable Operator: To consider the separable analysis operator in the sense of our previous work [14], we define the set of functions

$$\begin{aligned}\hat{f}\colon \mathfrak{C}\times\mathbb{R}^p &\to \mathbb{R}, \\ \mathbf{\Omega} &\mapsto g(\iota(\mathbf{\Omega})\mathbf{s}).\end{aligned}$$

The function $\hat{f}$ operates on the direct product of manifolds $\mathfrak{C} = \mathrm{Ob}_1 \times \mathrm{Ob}_2 \times \ldots \times \mathrm{Ob}_T$ and utilizes the function ι as defined in Equation (4). The signals $\mathbf{s}_i \in \mathbb{R}^p$ can be interpreted as vectorized versions of tensorial signals $\mathcal{S}_i \in \mathbb{R}^{p_1\times\ldots\times p_T}$ where $p = \prod p_i$. Above, we showed that f is λ-Lipschitz w.r.t. the Frobenius norm on its first variable $\mathbf{\Omega}$. As $\mathfrak{C}$ is a subset of a large oblique manifold, the same holds true for $\hat{f}$.

Similar to before, we define two Gaussian processes $G_{\mathbf{\Omega}} = \frac{1}{N}\sum_{i=1}^N \gamma_i \hat{f}(\mathbf{\Omega}, \mathbf{s}_i)$ with $\mathbf{\Omega} \in \mathfrak{C}$, and $H_{\mathbf{\Omega}} = \frac{\lambda}{\sqrt{N}}\sum_{i=1}^T \langle \mathbf{\Xi}^{(i)}, \mathbf{\Omega}^{(i)}\rangle_F$. The expected value $\mathbb{E}[|H_{\mathbf{\Omega}} - H_{\mathbf{\Omega}'}|^2]$ can be equivalently written as $\frac{\lambda^2}{N}\mathbb{E}[(\sum_{i=1}^T \mathrm{tr}((\mathbf{\Xi}^{(i)})^\top\mathbf{\Omega}^{(i)}))^2]$ and the inequality

$$\mathbb{E}_\gamma[|G_{\mathbf{\Omega}} - G_{\mathbf{\Omega}'}|^2] \leq \tfrac{\lambda^2}{N}\|\mathbf{\Omega} - \mathbf{\Omega}'\|_F^2 = \mathbb{E}_\xi[|H_{\mathbf{\Omega}} - H_{\mathbf{\Omega}'}|^2]$$

holds, just as in the non-separable case. Hence, we are able to apply Slepian's lemma which yields the inequality $\mathbb{E}[\sup_{\mathbf{\Omega}\in\mathfrak{C}} G_{\mathbf{\Omega}}] \leq \mathbb{E}[\sup_{\mathbf{\Omega}\in\mathfrak{C}} H_{\mathbf{\Omega}}]$. It only remains to provide an upper bound for the right-hand side.

Using the fact that $\mathfrak{C}$ is now the direct product of oblique manifolds we get

$$\begin{aligned}\mathbb{E}_\xi\left[\sup_{\mathbf{\Omega}\in\mathfrak{C}} H_{\mathbf{\Omega}}\right] &= \mathbb{E}_\xi\left[\sup_{\mathbf{\Omega}\in\mathfrak{C}} \tfrac{\lambda}{\sqrt{N}}\sum_{i=1}^T \mathrm{tr}\left((\mathbf{\Xi}^{(i)})^\top\mathbf{\Omega}^{(i)}\right)\right] \\ &= \tfrac{\lambda}{\sqrt{N}}\sum_{i=1}^T \mathbb{E}_\xi\left[\sup_{\mathbf{\Omega}^{(i)}\in\mathrm{Ob}_i} \mathrm{tr}\left((\mathbf{\Xi}^{(i)})^\top\mathbf{\Omega}^{(i)}\right)\right] \\ &= \tfrac{\lambda}{\sqrt{N}}\sum_{i=1}^T \mathbb{E}_\xi\left[\sum_{j=1}^{m_i}\|\boldsymbol{\xi}_j^{(i)}\|_2\right] \leq \tfrac{\lambda}{\sqrt{N}}\sum_{i=1}^T m_i\sqrt{p_i},\end{aligned}$$

where $\boldsymbol{\xi}_j^{(i)}$ denotes the transposed of the j-th row of $\mathbf{\Xi}^{(i)}$. The last inequality holds due to Jensen's inequality and the fact that all ξ_{ij} are $\mathcal{N}(0,1)$ random variables. □

Remark 1. Theorem 9 can be extended to the absolute value $|\mathbb{E}[f]-\hat{\mathbb{E}}_{\mathbf{S}}[f]|$ of the deviation by redefining the function class as $\mathcal{F}\cup(-\mathcal{F})$.

V. Stochastic Gradient Descent for Analysis Operator Learning

Stochastic Gradient Descent (SGD) is particularly suited for large scale optimization and thus a natural choice for many machine learning problems.

Before we describe a geometric SGD method that respects the underlying constraints on the analysis operator, we follow the discussion of SGD methods provided by Bottou in [29] in order to establish a connection of the excess error and the sample complexity result derived in the previous section. Let $f^\star = \arg\min_{f\in\mathfrak{F}} \mathbb{E}[f]$ be the best possible prediction function, let $f^\star_{\mathbf{S}} = \arg\min_{f\in\mathfrak{F}} \hat{\mathbb{E}}_{\mathbf{S}}[f]$ be the best possible prediction function for a set of training samples $\mathbf{S}$, and let $\tilde{f}_{\mathbf{S}}$ be the solution found by an optimization method with respect to the provided set of samples $\mathbf{S}$. Bottou proposes that the so-called excess error $\mathcal{E} = \mathbb{E}[\tilde{f}_{\mathbf{S}}] - \mathbb{E}[f^\star]$ can be decomposed as the sum $\mathcal{E} = \mathcal{E}_{\text{est}} + \mathcal{E}_{\text{opt}}$. Here, the estimation error $\mathcal{E}_{\text{est}} = \mathbb{E}[f^\star_{\mathbf{S}}] - \mathbb{E}[f^\star]$ measures the distance between the optimal solution for the expectation and the optimal solution for the empirical average while the optimization error $\mathcal{E}_{\text{opt}} = \mathbb{E}[\tilde{f}_{\mathbf{S}}] - \mathbb{E}[f^\star_{\mathbf{S}}]$ quantifies the distance between the optimal solution for the empirical average and the solution obtained via an optimization algorithm.

While $\mathcal{E}_{\text{opt}}$ is dependent on the optimization strategy, the estimation error $\mathcal{E}_{\text{est}}$ is closely related to the previously discussed sample complexity. Lower bounds on the sample complexity also apply to the estimation error as specified in the following Corollary.

Corollary 10. *Under the same conditions as in Theorem 9 the estimation error is upper bounded by*

$$\mathcal{E}_{est} \leq 2\sqrt{2\pi}\,\frac{\lambda C_{\mathfrak{C}}}{\sqrt{N}} + 6\sqrt{\frac{2\lambda^2 m \ln(2/\delta)}{N}} \tag{11}$$

with probability at least $1-\delta$.

Proof. The estimation error can be bounded via

$$\begin{aligned}\mathcal{E}_{\text{est}} &= \mathbb{E}[f^\star_{\mathbf{S}}] - \mathbb{E}[f^\star] \leq \mathbb{E}[f^\star_{\mathbf{S}}] - \mathbb{E}[f^\star] - \hat{\mathbb{E}}_{\mathbf{S}}[f^\star_{\mathbf{S}}] + \hat{\mathbb{E}}_{\mathbf{S}}[f^\star] \\ &\leq |\mathbb{E}[f^\star_{\mathbf{S}}] - \hat{\mathbb{E}}_{\mathbf{S}}[f^\star_{\mathbf{S}}]| + |\hat{\mathbb{E}}_{\mathbf{S}}[f^\star] - \mathbb{E}[f^\star]|,\end{aligned}$$

where the first inequality holds since $f^\star_{\mathbf{S}}$ is the minimizer of $\hat{\mathbb{E}}_{\mathbf{S}}$, and therefore $\hat{\mathbb{E}}_{\mathbf{S}}[f^\star_{\mathbf{S}}] \leq \hat{\mathbb{E}}_{\mathbf{S}}[f^\star]$ and the final result follows from Theorem 9 and its subsequent remark. □

A. Geometric Stochastic Gradient Descent

Ongoing from the seminal work of [30], SGD type optimization methods have attracted attention to solve large-scale machine learning problems [31], [32]. In contrast to full gradient methods that in each iteration require the computation of the gradient with respect to all the N training samples $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_N]$ in SGD the gradient computation only involves a small batch randomly drawn from the training set in order to find the $\mathbf{\Omega} \in \mathfrak{C}$ which minimizes the expectation $\mathbb{E}_{\mathbf{s}\sim\mathbb{P}}[f(\mathbf{\Omega}, \mathbf{s})]$. Accordingly, the cost of each iteration is independent of N (assuming the cost of accessing each sample is independent of N).

In the following we use the notation $\mathbf{s}_{\{k(i)\}}$ to denote a signal batch of cardinality $|k(i)|$, where $k(i)$ represents an index set randomly drawn from $\{1, 2, \ldots, N\}$ at iteration i.

In order to account for the constraint set $\mathfrak{C}$ we follow [33] and propose a geometric SGD optimization scheme. This requires some modifications to classic SGD. The subsequent discussion provides a concise introduction to line search optimization methods on manifolds. For more insights into optimization on manifolds in general we refer the reader to [34] and to [33] for optimization on manifolds using SGD in particular.

In Euclidean space the direction of steepest descent at a point $\mathbf{\Omega}$ is given by the negative (Euclidean) gradient. For optimization on an embedded manifold $\mathfrak{C}$ this role is taken over by the negative Riemannian gradient, which is a projection of the gradient onto the respective tangent space $T_{\mathbf{\Omega}}$. To keep notation simple, we denote the Riemannian gradient w.r.t. $\mathbf{\Omega}$ at a point $(\mathbf{\Omega}, \mathbf{s})$ by $\mathbf{G}(\mathbf{\Omega}, \mathbf{s}) = \Pi_{T_{\mathbf{\Omega}}\mathfrak{C}}(\nabla_{\mathbf{\Omega}} f(\mathbf{\Omega}, \mathbf{s}))$. Optimization methods on manifolds find a new iteration point by searching along geodesics instead of following a straight path. We denote a geodesic emanating from point $\mathbf{\Omega}$ in direction $\mathbf{H}$ by $\Gamma(\mathbf{\Omega}, \mathbf{H}, \cdot)$. Following this geodesic for distance t then results in the new point $\Gamma(\mathbf{\Omega}, \mathbf{H}, t) \in \mathfrak{C}$. Finally, an appropriate step size t has to be computed. A detailed discussion on this topic is provided in Section V-B.

Using these definitions, an update step of geometric SGD reads as

$$\mathbf{\Omega}_{i+1} = \Gamma(\mathbf{\Omega}_i, -\mathbf{G}(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i)\}}), t_i). \tag{12}$$

Since the SGD framework only provides a noisy estimate of the objective function in each iteration, a stopping criterion based on the average over previous iterations is chosen to terminate the optimization scheme. First, let $f(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i)\}})$ denote the mean over all signals in the batch $\mathbf{s}_{\{k(i)\}}$ associated to $\mathbf{\Omega}_i$ at iteration i. That is, $f(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i)\}}) = \frac{1}{|k(i)|}\sum_{j=1}^{|k(i)|} f(\mathbf{\Omega}_i, \mathbf{s}_j)$, for $\mathbf{s}_j \in \mathbf{s}_{\{k(i)\}}$. With the mean cost for a single batch at hand we are able to calculate the total average including all previous iterations. This reads as $\phi_i = \frac{1}{i}\sum_{j=0}^{i-1} f(\mathbf{\Omega}_{i-j}, \mathbf{s}_{\{k(i-j)\}})$. Furthermore, let $\bar{\phi}_i$ denote the mean over the last l values of ϕ_i. Finally, we are able to state our stopping criterion. The optimization terminates if the relative variation of ϕ_i, which is denoted as

$$v = \left(|\phi_i - \bar{\phi}_i|\right)/\bar{\phi}_i, \tag{13}$$

falls below a certain threshold δ. In our implementation, we set $l = 200$ with a threshold $\delta = 5 \cdot 10^{-5}$.

B. Step size selection

Regarding the convergence rate, a crucial factor of SGD optimization is the selection of the step size (often also referred to as learning rate). For convex problems, the step size is typically based on the Lipschitz continuity property. If the Lipschitz constant is not known in advance, an appropriate learning rate is often chosen by using approximation techniques. In [35], the authors propose a basic line search that

Algorithm 1 SGD Backtracking Line Search

Require: $a_i^0 > 0$, $b \in (0,1)$, $c \in (0,1)$, $\mathbf{\Omega}_i$, $\bar{f}(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i-1)\}})$, $\mathbf{G}(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i)\}})$, $k_{max} = 40$
Set: $a \leftarrow a_i^0$, $k \leftarrow 1$
while $\bar{f}(\Gamma(\mathbf{\Omega}_i, -\mathbf{G}(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i)\}}), a), \mathbf{s}_{\{k(i)\}}) > \bar{f}(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i-1)\}}) - a \cdot c \cdot \|\mathbf{G}(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i)\}})\|_F^2 \wedge k < k_{max}$ **do**
  $a \leftarrow b \cdot a$
  $k \leftarrow k + 1$
end while
Output: $t_i \leftarrow a$

sequentially halves the step size if the current estimate does not minimize the cost. Other approaches involve some predefined heuristics to iteratively shrink the step size [36] which has the disadvantage of requiring the estimation of an additional hyper-parameter. We suggest a more variable approach by proposing a variation of a backtracking line search algorithm adapted to SGD optimization.

As already stated in Section IV-A, our goal is to find a set of separable filters such that the empirical sparsity over all N samples from our training set is minimal. Now, recall that instead of computing the gradient with respect to the full training set, the SGD framework approximates the true gradient by means of a small signal batch or even a single signal sample. That is, the reduced computational complexity comes at the cost of updates that do not minimize the overall objective. However, it is assumed that on average the SGD updates approach the minimum of the optimization problem stated in (5), i.e., the empirical mean over all training samples. We utilize this proposition to automatically find an appropriate step size such that for the next iterate an averaging Armijo condition is fulfilled.

To be precise, starting from an initial step size a_i^0 the step length a_i is successively shrunk until the next iterate fulfills the Armijo condition. That is, we have

$$\begin{aligned}\bar{f}(\mathbf{\Omega}_{i+1}, \mathbf{s}_{\{k(i)\}}) \leq \bar{f}(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i-1)\}}) \\ - a_i \cdot c \cdot \|\mathbf{G}(\mathbf{\Omega}_i, \mathbf{s}_{\{k(i)\}})\|_F^2,\end{aligned} \tag{14}$$

with some constant $c \in (0,1)$. Here, $\bar{f}$ denotes the average cost over a predefined number of previous iterations. The average is calculated over the function values $f(\mathbf{\Omega}_{i+1}, \mathbf{s}_{\{k(i)\}})$, i.e., over the cost of the optimized operator with respect to the respective signal batch. This is achieved via a sliding window implementation that reads

$$\bar{f}(\mathbf{\Omega}_{i+1}, \mathbf{s}_{\{k(i)\}}) = \tfrac{1}{w} \sum_{j=0}^{w-1} f(\mathbf{\Omega}_{i+1-j}, \mathbf{s}_{\{k(i-j)\}}), \tag{15}$$

with w denoting the window size.

If (14) is not fulfilled, i.e., if the average including the new sample is not at least as low as the previous average, the step size a_i goes to zero. To avoid needless line search iterations we stop the execution after a predefined number of trials k_{max} and proceed with the next sample without updating the filters and with resetting a_{i+1} to its initial value a_i^0. The complete step size selection approach is summarized in Algorithm 1. In our experiments we set the parameters to $b = 0.9$ and $c = 10^{-4}$.

C. Cost Function and Constraints

An appropriate sparsity measure for our purposes is provided by

$$g(\boldsymbol{\alpha}) \coloneqq \sum\nolimits_{j=1}^{m} \log\left(1 + \nu \alpha_j^2\right). \tag{16}$$

This function serves as a smooth approximation to the ℓ_0-quasi-norm, cf. [7], but other smooth sparsity promoting functions are also conceivable.

In the section on sample complexity we introduced the oblique manifold as a suitable constraint set. Additionally, there are two properties we wish to enforce on the learned operator as motivated in [3]. (i) Full rank of the operator and (ii) No identical filters. This is achieved by incorporating two penalty functions into the cost function, namely

$$\begin{aligned} h(\mathbf{\Omega}) &= -\tfrac{1}{p \log(p)} \log\det\left(\tfrac{1}{m}(\mathbf{\Omega})^\top \mathbf{\Omega}\right), \\ r(\mathbf{\Omega}) &= -\sum_{k<l} \log\left(1 - \left((\boldsymbol{\omega}_k)^\top (\boldsymbol{\omega}_l)\right)^2\right). \end{aligned}$$

The function h promotes (i) whereas r enforces (ii). Hence, the final optimization problem for a set of training samples $\mathbf{s}_i$ (vectorized versions of signals $\mathcal{S}_i$ in tensor form) is given as

$$\begin{aligned} &\arg\min_{\mathbf{\Omega}} \ \tfrac{1}{N} \sum_{j=1}^{N} f(\mathbf{\Omega}, \mathbf{s}_j) \\ &\text{subject to} \quad \mathbf{\Omega} = (\mathbf{\Omega}^{(1)}, \ldots, \mathbf{\Omega}^{(T)}), \\ &\qquad\qquad \mathbf{\Omega}^{(i)} \in \mathrm{Ob}(m_i, p_i), \quad i = 1, \ldots, T, \end{aligned} \tag{17}$$

with the function

$$f(\mathbf{\Omega}, \mathbf{s}) = g\left(\iota(\mathbf{\Omega})\mathbf{s}\right) + \kappa h(\iota(\mathbf{\Omega})) + \mu r(\iota(\mathbf{\Omega})). \tag{18}$$

The parameters κ and μ are weights that control the impact of the full rank and incoherence condition. With this formulation of the optimization problem both separable as well as non-separable learning can be handled with the same cost function allowing for a direct comparison of these scenarios.

VI. Experiments

The purpose of the experiments presented in this section is, on the one side, to give some numerical evidence of the sample complexity results from Section IV, and on the other side, to demonstrate the efficiency and performance of our proposed learning approach from Section V.

A. Learning from natural image patches

The task of our first experiment is to demonstrate that separable filters can be learned from less training samples compared to learning a set of unstructured filters. We generated a training set that consists of $N = 500\,000$ two-dimensional normalized samples of size $\mathbf{S}_i \in \mathbb{R}^{7\times 7}$ extracted at random from natural images. The learning algorithm then provides two operators $\mathbf{\Omega}^{(1)}, \mathbf{\Omega}^{(2)} \in \mathbb{R}^{8\times 7}$ resulting in 64 separable filters. We compare our proposed separable approach with a version of the same algorithm, that does not enforce a separable structure on the filters and thus outputs a non-separable analysis operator $\mathbf{\Omega} \in \mathbb{R}^{64\times 49}$.

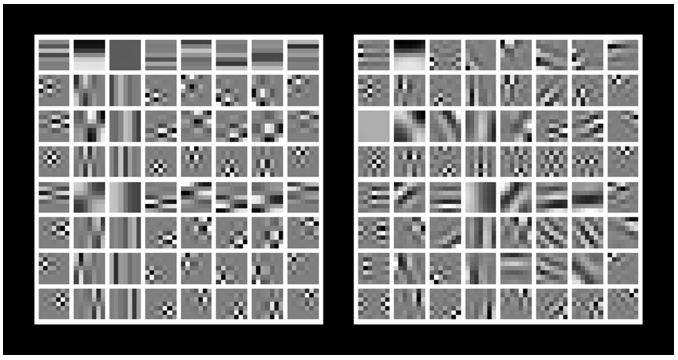

Fig. 1. Left: Learned filters with separable structure. Right: Result after learning the filters without separability constraint.

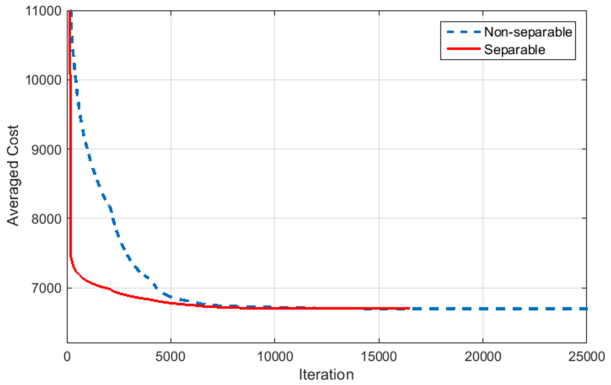

Fig. 2. Convergence comparison between the SGD based learning framework with and without separability constraint imposed on the filters. The dotted line denotes the averaged cost for the non-separable case. The solid graph indicates the averaged cost when a separable structure is enforced on the filters.

The weighting parameters for the constraints in (18) are set to $\kappa = 6500$, $\mu = 0.0001$. The factor that controls the slope in the sparsity measure defined in (16) is $\nu = 500$. At each iteration of the SGD optimization, a batch of 500 samples is processed. The averaging window size in (15) for the line search is fixed to $w = 2000$. In all our experiments we start learning from random filter initializations.

To visualize the efficiency of the separable learning approach the averaged function value at iteration i as defined in (15) is plotted in Figure 2. While the dotted curve corresponds to the learning framework that does not enforce a separable structure on the filters, the solid graph visualizes the cost with separability constraint. The improvement in efficiency is twofold. First, imposing separability leads to a faster convergence to the empirical mean of the cost in the beginning of the optimization. Second, the optimization terminates after fewer iterations, i.e., less training samples are processed until no further update of the filters is observed. In order to offer an idea of the learned structures, the separable and non-separable filters obtained via our learning algorithm are shown in Figure 1 as 7×7 2D-filter kernels.

B. Analysis operator recovery from synthetic data

As mentioned in the introduction, there are many application scenarios where a learned analysis operator can be employed, ranging from inverse problems in imaging, to registration, segmentation and classification tasks. Therefore, in order to provide a task independent evaluation of the proposed learning algorithm, we have conducted experiments that are based on synthetic data and investigated how well a learned operator $\mathbf{\Omega}_{\text{learned}}$ approximates a ground truth operator $\mathbf{\Omega}_{\text{GT}}$. When measuring the accuracy of the recovery we have to take into account that there is an inherent sign and permutation ambiguity in the learned filters. Hence, we consider the absolute values of the correlation of the filters over all possible permutations.

To be precise, let us denote $\tilde{\boldsymbol{\omega}}_i$ as the i^{th}-row of $\mathbf{\Omega}_{\text{learned}}$ and $\boldsymbol{\omega}_j$ as the j^{th}-row of $\mathbf{\Omega}_{\text{GT}}$, both represented as column vectors. We define the deviation of these filters from each other as $c_{ij} = 1 - |\tilde{\boldsymbol{\omega}}_i^\top \boldsymbol{\omega}_j|$. Doing this for all possible combinations of i and j we obtain the confusion matrix $\mathbf{C}$, where the (i,j)-entry $\mathbf{C}$ is 0 if $\tilde{\boldsymbol{\omega}}_i$ is equal to $\boldsymbol{\omega}_j$. Building the confusion matrix accounts for the permutation ambiguity between $\mathbf{\Omega}_{\text{GT}}$ and $\mathbf{\Omega}_{\text{learned}}$. Next, we utilize the Hungarian-method [37] to determine the path through the confusion matrix $\mathbf{C}$ with the lowest accumulated cost under the constraint that each row and each column is visited only once. In the end, the coefficients along the path are accumulated and this sum serves as our error measure denoted as $H(\mathbf{C})$. In other words, we aim to find the lowest sum of entries in $\mathbf{C}$ such that in each line a single entry is picked and no column is used twice. With this strategy we prevent that multiple retrieved filters $\tilde{\boldsymbol{\omega}}_i$ are matched to the same filter $\boldsymbol{\omega}_j$, i.e., the error measure $H(\mathbf{C})$ is zero if and only if all filters in $\mathbf{\Omega}_{\text{GT}}$ are recovered.

Following the procedure in [38], we generated a synthetic set of samples of size $\mathbf{S}_i \in \mathbb{R}^{7\times 7}$ w.r.t. to $\mathbf{\Omega}_{\text{GT}}$. As the ground truth operator we chose the separable operator obtained in the previous Subsection VI-A. The generated signals exhibit a predefined co-sparsity after applying the ground truth filters to them. The set of samples has the size $N = 500\,000$. The co-sparsity, i.e., the number of zero filter responses is fixed to 15. Additive white Gaussian noise with standard deviation 0.05 is added to each normalized signal sample. We now aim at retrieving the underlying original operator that was used to generate the signals. Again, we compare our separable approach against the same framework without the separability constraint.

In order to compare the performance of the proposed SGD algorithm in the separable and non-separable case, we conduct an experiment where the size of the training sample batch that is used for the gradient and cost calculation is varied. The employed batch sizes are $\{1, 10, 25, 50, 75, 100, 250, 500, 1000\}$, while the performance is evaluated over ten trials, i.e., ten different synthetic sets that have been generated in advance.

Figure 3 summarizes the results for this experiment. For each batch size the error over all ten trials is illustrated. The left box corresponds to the separable approach and accordingly the right box denotes the error for the non-separable filters. While the horizontal dash inside the boxes indicates the median over all ten trials the boxes represent the mid-50%. The dotted dashes above and below the boxes indicate the maximum and minimum error obtained.

It is evident that the separable operator learning algorithm

TABLE I
COMPARISON BETWEEN THE CONVERGENCE SPEED AND AVERAGE ERROR FOR THE FILTER RECOVERY EXPERIMENT. FOR EACH BATCH SIZE THE AVERAGE NUMBER OF ITERATIONS UNTIL CONVERGENCE IS GIVEN ALONG WITH THE AVERAGE ERROR WHICH DENOTES THE MEAN OF THE VALUES FOR $H(\mathbf{C})$ OVER ALL THE TEN TRIALS. UPPER PART: LEARNING SEPARABLE FILTERS. LOWER PART: LEARNING NON-SEPARABLE FILTERS.

	1	10	25	50	75	100	250	500	1000
Iterations	450	422	2573	3721	4595	5780	9673	13637	15198
Avg. error	37.22	33.28	1.39	1.08	0.75	0.56	0.31	0.24	0.14
Iterations	1812	1800	2786	3866	10147	13595	19679	21087	20611
Avg. error	41.21	40.73	38.68	27.74	9.67	1.89	1.38	1.23	1.14

TABLE II
COMPARISON BETWEEN THE SGD OPTIMIZATION AND CONJUGATE GRADIENT OPTIMIZATION. AVERAGE NUMBER OF ITERATIONS AND PROCESSING TIMES OVER TEN TRIALS.

	Iterations	time in sec	$H(\mathbf{C})$
SGD (separable)	12358	1176	0.48
SGD (non-separable)	21660	1554	1.24
CG (GOAL [3])	601	2759	1.01

achieves better recovery of the ground truth operator for smaller batch sizes, which indicates that it requires less samples in order to produce good recovery results. Table I shows the average number of iterations until convergence and the averaged error over all trials. As can be seen from the table, the separable approach requires less samples to achieve good accuracy, has a faster convergence and a smaller recovery error compared to the non-separable method. As an example, the progress of the recovery error for a batch size of 500 is plotted in Figure 4.

Finally, in order to show the efficiency of the SGD-type optimization we compare our method to an operator learning framework that utilizes a full gradient computation at each iteration. We chose the algorithm proposed in [3] which learns a set of non-separable filters via a geometric conjugate gradient on manifolds approach. Again, we generated ten sets of $N = 500\,000$ samples with a predefined cosparsity. Based on this synthetic set of signals, we measured the mean computation time and number of iterations until the stopping criterion from Section V-A is fulfilled. Table II summarizes the results for the proposed separable SGD, the non-separable SGD and the non-separable CG implementations. While the CG based optimization converges after only a few iterations, the overall execution time is worse compared to SGD due to the high computational cost for each full gradient calculation.

Figure 4 and Table II in particular support the theoretical results obtained for the sample complexity. They illustrate that the separable SGD algorithm requires less iterations, and therefore fewer samples, to reach the dropout criterion compared to the SGD algorithm that does not enforce separability of the learned operator.

C. Comparison with related approaches on image data

In order to show that the operator learned with separable structures is applicable to real world signal processing tasks, we have conducted a simple image denoising experiment. Rather than outperforming existing denoising algorithms, the message conveyed by this experiment is that using separable

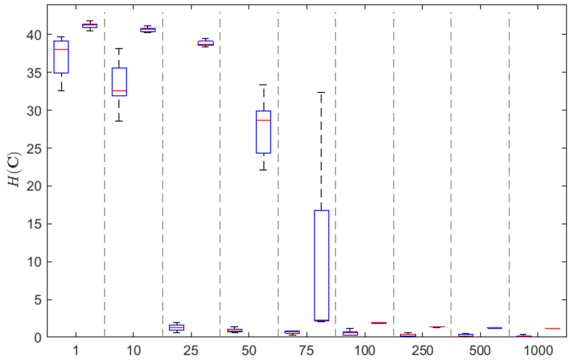

Fig. 3. The horizontal axis indicates the batch size. For each size, on the left the recovery error for the separable case is plotted, whereas the error for the non-separable case is plotted on the right. All generated signals exhibit a co-sparsity of 15 and for each batch size the error is evaluated over 10 trials. The horizontal dash inside the boxes indicates the median error over all trials. The boxes represent the central 50% of the errors obtained, while the dashed lines above and below the boxes indicate maximal and minimal errors.

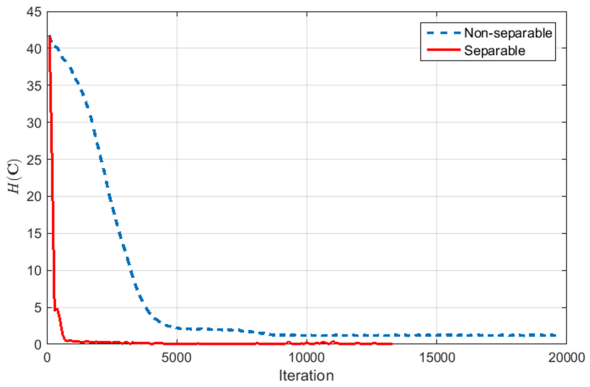

Fig. 4. Operator recovery error for a batch size of 500. The dotted graph indicates the progress over the iterations for the non-separable case, while the solid curve shows the progress for retrieving separable filters.

filters only slightly reduces the reconstruction performance. We compare our separable operator (sepSGD) against other learning schemes that provide a set of filters without imposing a separability constraint. Specifically, in addition to our non-separable SGD implementation (SGD), we have chosen the geometric analysis operator learning scheme (GOAL) from [3], the Analysis-KSVD algorithm (AKSVD) proposed in [16], and the method presented in [4] (CAOL) for comparison. All operators are learned from $N = 500\,000$ patches of size 7×7 extracted from eight different standard natural training images that are not included in the test set. All operators are of size 64×49. For sepSGD, SGD and GOAL we have set the parameters to the same values as already stated in VI-A. The parameters of the AKSVD and CAOL learning algorithms have been tuned such that the overall denoising performance is best for the chosen test images. Four standard test images (Barbara, Couple, Lena, and Man), each of size 512×512 pixels, have been artificially corrupted with additive white Gaussian noise with standard deviation $\sigma_n \in \{10, 20, 30\}$.

The learned operators are used as regularizers in the denois-

TABLE III
DENOISING EXPERIMENT FOR FOUR DIFFERENT TEST IMAGES CORRUPTED BY THREE NOISE LEVELS. ACHIEVED PSNR IN DECIBELS (DB).

σ_n / PSNR		Barbara	Couple	Lena	Man
10 / 28.13	sepSGD	32.13	32.76	34.10	32.71
	SGD	31.82	32.43	33.98	32.64
	GOAL	32.28	32.62	34.29	32.80
	AKSVD	31.75	31.69	33.51	31.97
	CAOL	30.44	30.35	31.41	30.72
20 / 22.11	sepSGD	27.89	28.97	30.46	29.00
	SGD	27.61	28.69	30.36	28.97
	GOAL	28.01	28.86	30.65	29.12
	AKSVD	27.49	27.68	29.85	28.22
	CAOL	26.05	26.23	27.27	26.63
30 / 18.59	sepSGD	25.64	26.83	28.24	27.02
	SGD	25.43	26.63	28.22	27.02
	GOAL	25.75	26.76	28.40	27.12
	AKSVD	25.37	25.66	27.75	26.33
	CAOL	23.72	24.00	24.89	24.35

ing task which is formulated as an inverse problem. We have utilized the NESTA algorithm [39] which solves the analysis-based unconstrained inverse problem

$$\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathbb{R}^n} \tau\|\mathbf{\Omega}^*(\mathbf{x})\|_1 + \tfrac{1}{2}\|\mathbf{y}-\mathbf{x}\|_2^2,$$

where $\mathbf{x}$ represents a vectorized image, $\mathbf{y}$ are the noisy measurements, and τ is a weighting factor. $\mathbf{\Omega}^*(\mathbf{x})$ denotes the operation of applying the operator $\mathbf{\Omega}^*$ to all overlapping patches of the image $\mathbf{x}$ by convolving each of the learned filters with the patches. For all operators, the weighting factor is set to $\tau \in \{0.18, 0.40, 0.60\}$ for the noise levels $\sigma_n \in \{10, 20, 30\}$, respectively. Table III summarizes the results of this experiment.

The presented results indicate that using separable filters does not reduce the image restoration performance to a great extent and that separable filters are competitive with non-separable ones. Furthermore, the SGD update has the advantage that the execution time of a single iteration in the learning phase does not grow with the size of the training set which is an important issue for extensive training set dimensions or online learning scenarios. Please note that we have not optimized the parameters of our learning scheme for the particular task of image denoising.

VII. CONCLUSION

We proposed a sample complexity result for analysis operator learning for signal distributions within the unit ℓ_2-ball, where we have assumed that the sparsity promoting function fulfills a Lipschitz condition. Rademacher complexity and McDiarmid's inequality were utilized to prove that the deviation of the empirical co-sparsity of a training set and the expected co-sparsity is bounded by $\mathcal{O}(C/\sqrt{N})$ with high probability, where N denotes the number of samples and C is a constant that among other factors depends on the (separable) structure imposed on the analysis operator during the learning process. Furthermore, we suggested a geometric stochastic gradient descent algorithm that allows to incorporate the separability constraint during the learning phase. An important aspect of this algorithm is the line search strategy which we designed in such a way that it fulfills an averaging Armijo condition. Our theoretical results and our experiments confirmed that learning algorithms benefit from the added structure present in separable operators in the sense that fewer training samples are required in order for the training phase to provide an operator that offers good performance. Compared to other co-sparse analysis operator learning methods that rely on updating the cost function with respect to a full set of training samples in each iteration, our proposed method benefits from a dramatically reduced training time, a common property among SGD methods. This characteristic further endorses the choice of SGD methods for co-sparse analysis operator learning.

APPENDIX

Addition to proof of Lemma 7: In order to upper bound the expectation of $\Phi(\mathbf{S})$ in (8), we follow a common strategy which we outline in the following for the convenience of the reader. First, we introduce a set of ghost samples $\tilde{\mathbf{S}} = [\tilde{\mathbf{s}}_1, \ldots, \tilde{\mathbf{s}}_N]$ where all samples are drawn independently according to the same distribution as the samples in $\mathbf{S}$. For this setting the equations $\mathbb{E}_{\tilde{\mathbf{S}}}[\hat{\mathbb{E}}_{\tilde{\mathbf{S}}}[f]] = \mathbb{E}[f]$ and $\mathbb{E}_{\tilde{\mathbf{S}}}[\hat{\mathbb{E}}_{\mathbf{S}}[f]] = \hat{\mathbb{E}}_{\mathbf{S}}[f]$ hold. Using this, we deduce

$$\mathbb{E}_{\mathbf{S}}[\Phi(\mathbf{S})] \leq \mathbb{E}_{\mathbf{S},\tilde{\mathbf{S}}}\left[\sup_{f\in\mathfrak{F}} \tfrac{1}{N}\sum\nolimits_i (f(\mathbf{\Omega},\tilde{\mathbf{s}}_i) - f(\mathbf{\Omega},\mathbf{s}_i))\right] \quad (19)$$

$$= \mathbb{E}_{\sigma,\mathbf{S},\tilde{\mathbf{S}}}\left[\sup_{f\in\mathfrak{F}} \tfrac{1}{N}\sum\nolimits_i \sigma_i (f(\mathbf{\Omega},\tilde{\mathbf{s}}_i) - f(\mathbf{\Omega},\mathbf{s}_i))\right] \quad (20)$$

$$\leq 2R_N(\mathfrak{F}).$$

Here, the inequality (19) holds because of the convexity of the supremum and by application of Jensen's inequality, (20) is true since $\mathbb{E}[\sigma_i] = 0$ and the last inequality follows from the definition of the supremum and using the fact that negating a Rademacher variable does not change its distribution.

The next step is to bound the Rademacher complexity by the empirical Rademacher complexity. To achieve this, note that $\hat{R}_{\mathbf{S}}(\mathfrak{F})$, like Φ, fulfills the condition for McDiarmid's theorem with factor $2\lambda\sqrt{m}/N$. This leads to the final result.

ACKNOWLEDGMENT

This work was supported by the German Research Foundation (DFG) under grant KL 2189/8-1. The contribution of Rémi Gribonval was supported in part by the European Research Council, PLEASE project (ERC-StG-2011-277906).

REFERENCES

[1] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and smoothness via the fused lasso," *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 67, no. 1, pp. 91–108, 2005.
[2] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, 1992.
[3] S. Hawe, M. Kleinsteuber, and K. Diepold, "Analysis operator learning and its application to image reconstruction," *IEEE Trans. Image Process.*, vol. 22, no. 6, pp. 2138–2150, 2013.
[4] M. Yaghoobi, S. Nam, R. Gribonval, and M. E. Davies, "Constrained overcomplete analysis operator learning for cosparse signal modelling," *IEEE Trans. Signal Process.*, vol. 61, no. 9, pp. 2341–2355, 2013.

[5] Y. Chen, T. Pock, and H. Bischof, "Learning ℓ_1-based analysis and synthesis sparsity priors using bi-level optimization," in *Workshop on Analysis Operator Learning vs. Dictionary Learning, NIPS*, 2012.

[6] Y. Chen, R. Ranftl, and T. Pock, "Insights into analysis operator learning: A view from higher-order filter-based MRF model," *IEEE Trans. Image Process.*, vol. 23, no. 3, pp. 1060–1072, 2014.

[7] M. Kiechle, T. Habigt, S. Hawe, and M. Kleinsteuber, "A bimodal co-sparse analysis model for image processing," *Int. J. Comput. Vision*, pp. 1–15, 2014.

[8] C. Nieuwenhuis, S. Hawe, M. Kleinsteuber, and D. Cremers, "Co-sparse textural similarity for interactive segmentation," in *IEEE European Conf. Computer Vision*, 2014, pp. 285–301.

[9] S. Shekhar, V. M. Patel, and R. Chellappa, "Analysis sparse coding models for image-based classification," in *IEEE Int. Conf. Image Processing*, 2014, pp. 5207–5211.

[10] J. Wörmann, S. Hawe, and M. Kleinsteuber, "Analysis based blind compressive sensing," *IEEE Signal Process. Lett.*, vol. 20, no. 5, pp. 491–494, 2013.

[11] L. Albera, S. Kitic, N. Bertin, G. Puy, and R. Gribonval, "Brain source localization using a physics-driven structured cosparse representation of eeg signals," in *IEEE Int. Workshop on Machine Learning for Signal Processing*, 2014, pp. 1–6.

[12] L. Pfister and Y. Bresler, "Tomographic reconstruction with adaptive sparsifying transforms," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2014, pp. 6914–6918.

[13] S. Hawe, M. Seibert, and M. Kleinsteuber, "Separable dictionary learning," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2013, pp. 438–445.

[14] M. Seibert, J. Wörmann, R. Gribonval, and M. Kleinsteuber, "Separable cosparse analysis operator learning," in *Proc. European Signal Processing Conf.*, 2014.

[15] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. A.*, vol. 21, pp. 1253–1278, 2000.

[16] R. Rubinstein, T. Peleg, and M. Elad, "Analysis K-SVD: A dictionary-learning algorithm for the analysis sparse model," *IEEE Trans. Signal Process.*, vol. 61, no. 3, pp. 661–677, 2013.

[17] S. Ravishankar and Y. Bresler, "Learning overcomplete sparsifying transforms for signal processing," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2013, pp. 3088–3092.

[18] J. Dong, W. Wang, and W. Dai, "Analysis SimCO: A new algorithm for analysis dictionary learning," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2014, pp. 7193–7197.

[19] N. Qi, Y. Shi, X. Sun, J. Wang, and W. Ding, "Two dimensional analysis sparse model," in *IEEE Int. Conf. Image Processing*, 2013, pp. 310–314.

[20] A. Maurer and M. Pontil, "K-dimensional coding schemes in Hilbert spaces," *IEEE Trans. Inf. Theory*, vol. 56, no. 11, pp. 5839–5846, 2010.

[21] D. Vainsencher, S. Mannor, and A. M. Bruckstein, "The sample complexity of dictionary learning," *J. Mach. Learn. Res.*, vol. 12, pp. 3259–3281, 2011.

[22] R. Gribonval, R. Jenatton, F. Bach, M. Kleinsteuber, and M. Seibert, "Sample complexity of dictionary learning and other matrix factorizations," *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3469–3486, 2015.

[23] R. Meir and T. Zhang, "Generalization error bounds for Bayesian mixture algorithms," *J. Mach. Learn. Res.*, vol. 4, pp. 839–860, 2003.

[24] P. L. Bartlett and S. Mendelson, "Rademacher and Gaussian complexities: Risk bounds and structural results," *J. Mach. Learn. Res.*, vol. 3, pp. 463–482, 2003.

[25] R. Gribonval, R. Jenatton, and F. Bach, "Sparse and spurious: dictionary learning with noise and outliers," *IEEE Trans. Inf. Theory*, 2015, to appear.

[26] C. McDiarmid, "On the method of bounded differences," in *Surveys in Combinatorics*, ser. London Mathematical Society Lecture Note Series, no. 141. Cambridge University Press, 1989, pp. 148–188.

[27] S. Mendelson, "A few notes on statistical learning theory," in *Advanced lectures on machine learning*. Springer, 2003, pp. 1–40.

[28] M. Ledoux and M. Talagrand, *Probability in Banach Spaces: isoperimetry and processes*. Springer Science & Business Media, 2013, vol. 23.

[29] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. Int. Conf. Computational Statistics*, 2010, pp. 177–187.

[30] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, no. 3, pp. 400–407, 1951.

[31] L. Bottou and Y. LeCun, "Large scale online learning," in *Adv. Neural Information Processing Systems*, 2004, pp. 217–224.

[32] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, 2010.

[33] S. Bonnabel, "Stochastic gradient descent on riemannian manifolds," *IEEE Trans. Autom. Control*, vol. 58, no. 9, pp. 2217–2229, 2013.

[34] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[35] N. Le Roux, M. Schmidt, and F. Bach, "A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets," in *Adv. Neural Information Processing Systems*, 2012, pp. 2663–2671.

[36] L. Bottou, "Stochastic gradient tricks," in *Neural Networks, Tricks of the Trade, Reloaded*, ser. Lecture Notes in Computer Science (LNCS 7700). Springer, 2012, pp. 430–445.

[37] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1–2, pp. 83–97, 1955.

[38] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "The cosparse analysis model and algorithms," *Appl. Comput. Harmon. Anal.*, vol. 34, no. 1, pp. 30–56, 2013.

[39] S. Becker, J. Bobin, and E. J. Candés, "NESTA: A fast and accurate first-order method for sparse recovery," *SIAM J. Imaging Sci.*, vol. 4, no. 1, pp. 1–39, 2011.

Matthias Seibert received the Dipl.-math. degree from the Julius-Maximilians-Universität Würzburg, Würzburg, Germany, in 2012. He is currently pursuing the doctoral degree at the Department of Electrical and Computer Engineering, Technische Universität München, Munich, Germany. His current research interests include sparse signal models, geometric optimization, and learning complexity.

Julian Wörmann received the M.Sc. degree in electrical engineering from the Technische Universität München, Munich, Germany, in 2012, where he is currently pursuing the doctoral degree at the Department of Electrical and Computer Engineering. His current research interests include sparse signal models, geometric optimization, inverse problems in image processing, and computer vision.

Rémi Gribonval (FM'14) is a Senior Researcher with Inria (Rennes, France), and the scientific leader of the PANAMA research group on sparse audio processing. A former student at École Normale Supérieure (Paris, France), he received the Ph. D. degree in applied mathematics from Université de Paris-IX Dauphine (Paris, France) in 1999, and his Habilitation à Diriger des Recherches in applied mathematics from Université de Rennes I (Rennes, France) in 2007. His research focuses on mathematical signal processing, machine learning, approximation theory and statistics, with an emphasis on sparse approximation, audio source separation, dictionary learning and compressive sensing.

Martin Kleinsteuber received his PhD in Mathematics from the University of Würzburg, Germany, in 2006. After post-doc positions at National ICT Australia Ltd, the Australian National University, Canberra, Australia, and the University of Würzburg, he has been appointed assistant professor for geometric optimization and machine learning at the Department of Electrical and Computer Engineering, TU München, Germany, in 2009. He won the SIAM student paper prize in 2004 and the Robert-Sauer-Award of the Bavarian Academy of Science in 2008 for his works on Jacobi-type methods on Lie algebras. His research interests are in the areas of statistical signal processing, machine learning and computer vision, with an emphasis on optimization and learning methods for sparse and robust signal representations.