Short-and-Sparse Deconvolution — A Geometric Approach

Anonymous Author(s)
Affiliation
Address
email

Abstract

Short-and-sparse deconvolution (SaSD) is the problem of extracting localized, recurring motifs in signals with spatial or temporal structure. Variants of this problem arise in applications such as image deblurring, microscopy, neural spike sorting, and more. SaSD is challenging in both theory and practice, as natural optimization formulations are nonconvex. Moreover, practical deconvolution problems involve smooth motifs (kernels) whose spectra decay rapidly, resulting in poor conditioning and numerical challenges. This paper is motivated by recent theoretical advances [1, 2], which characterize the optimization landscape of a particular nonconvex formulation of SaSD. This is used to derive a *provable* algorithm which exactly solves certain non-practical instances of the SaSD problem. We leverage the key ideas from this theory (sphere constraints, data-driven initialization) to develop a *practical* algorithm, which performs well on data arising from a range of application areas. We highlight key additional challenges posed by the ill-conditioning of real SaSD problems, and suggest heuristics (acceleration, continuation, reweighting) to mitigate them. Experiments demonstrate both the performance and generality of the proposed method.

1 Introduction

Many signals arising in science and engineering can be modeled as superpositions of basic, recurring motifs, which encode critical information about a physical process of interest. Signals of this type can be modeled as the convolution of a zero-padded short kernel $\boldsymbol{a}_0 \in \mathbb{R}^{p_0}$ (the motif) with a longer sparse signal $\boldsymbol{x}_0 \in \mathbb{R}^m$ ($m \gg p_0$) which encodes the locations of the motifs in the sample[1]:

$$\boldsymbol{y} \;=\; \iota \boldsymbol{a}_0 \;\circledast\; \boldsymbol{x}_0. \tag{1}$$

We term this a short-and-sparse (SaS) model. In practice, often only $\boldsymbol{y}$ is directly observed. *Short-and-sparse deconvolution* (SaSD) is the problem of recovering both $\boldsymbol{a}_0$ and $\boldsymbol{x}_0$ from $\boldsymbol{y}$. Variants of this problem arise in areas such as microscopy [3], astronomy [4], and neuroscience [5]. SaSD is a challenging inverse problem in both theory and practice. Natural formulations are nonconvex, and until recently very little algorithmic theory was available. Moreover, practical instances are typically ill-conditioned, due to the spectral decay of the kernel $\boldsymbol{a}_0$.

This paper is motivated by recent theoretical advances in nonconvex optimization – and in particular, on the geometry of SaSD. [1, 2] study particular optimization formulations for SaSD and show that the landscape is largely driven by the *problem symmetries* of SaSD. They derive provable methods for idealized problem instances, which exactly recover $(\boldsymbol{a}_0, \boldsymbol{x}_0)$ up to trivial ambiguities. While inspiring, these methods are *not practical* and perform poorly on real problem instances. Where the

[1]For simplicity, (1) uses cyclic convolution; algorithms are results also apply to linear convolution with minor modifications. Here ι denotes the zero padding operator.

Submitted to 33rd Conference on Neural Information Processing Systems (NeurIPS 2019). Do not distribute.

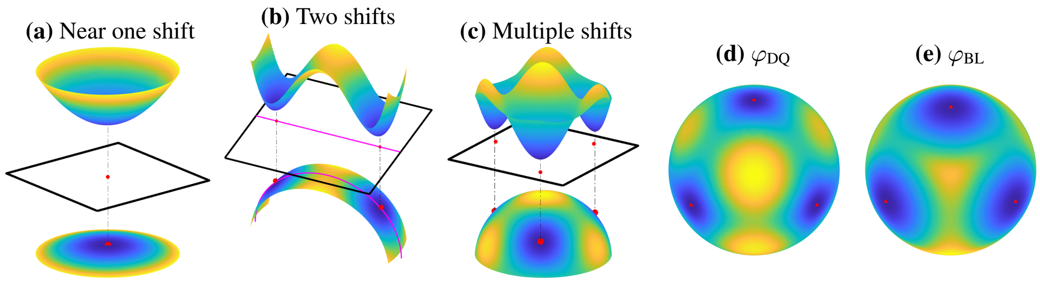

Figure 1: **Geometry of $\varphi_{\mathbf{DQ}}$ near superpositions of shifts** of $\boldsymbol{a}_0$ [2]. **(a)** Regions near single shifts are strongly convex. **(b)** Regions between two shifts contain a saddle-point, with negative curvature pointing towards each shift and positive curvature orthogonally. **(c)** The span of three shifts. When $\mu_s(\boldsymbol{a}_0) \approx 0$, marginalizations of the **(d)** Dropped Quadratic and **(e)** Bilinear Lasso ($\varphi_{\mathrm{BL}} \doteq \min_{\boldsymbol{x}} \Psi_{\mathrm{BL}}(\boldsymbol{a}, \boldsymbol{x})$) are similar empirically.

emphasis of [1, 2] is on theoretical guarantees, here we focus on practical computation. We show how to combine ideas from this theory with heuristics that better address the properties of practical deconvolution problems, to build a novel method that performs well on data arising in a range of application areas. A critical issue in moving from theory to practice is the poor conditioning of naturally-occurring deconvolution problems: we show how to address this with a combination of ideas from sparse optimization, such as momentum, continuation, and reweighting. The end result is a general purpose method, which we demonstrate on data from neural spike sorting, calcium imaging and fluorescence microscopy.

Notation The zero-padding operator is denoted by $\iota : \mathbb{R}^p \to \mathbb{R}^m$. Projection of a vector $\boldsymbol{v} \in \mathbb{R}^p$ onto the sphere is denoted by $\mathcal{P}_{\mathbb{S}^{p-1}}(\boldsymbol{v}) \doteq \boldsymbol{v}/\left\|\boldsymbol{v}\right\|_2$, and $\mathcal{P}_{\boldsymbol{z}}(\boldsymbol{v}) \doteq \boldsymbol{v} - \langle \boldsymbol{v}, \boldsymbol{z}\rangle \boldsymbol{z}$ denotes projection onto the tangent space of $\boldsymbol{z} \in \mathbb{S}^{p-1}$. The Riemannian gradient of a function $f : \mathbb{S}^{p-1} \to \mathbb{R}$ on the sphere is given by $\operatorname{grad} f \doteq \mathcal{P}_{\mathbb{S}^{p-1}} \circ \nabla f$.

2 The role of symmetry in SaSD

2.1 Symmetry and shift-coherence

An important observation of the SaSD problem is that it admits multiple equivalent solutions. This is purely due to the cyclic convolution between $\boldsymbol{a}_0$ and $\boldsymbol{x}_0$, which exhibits the trivial ambiguity[2]

$$\boldsymbol{y} \;=\; \iota \boldsymbol{a}_0 \circledast \boldsymbol{x}_0 \;=\; \left(\alpha s_\ell\left[\iota \boldsymbol{a}_0\right]\right) \;\circledast\; \left(\tfrac{1}{\alpha} s_{-\ell}\left[\boldsymbol{x}_0\right]\right),$$

for any nonzero scalar α and cyclic shift $s_\ell\left[\cdot\right]$. Since these *scale and shift symmetries* create several acceptable candidates for $\boldsymbol{a}_0$ and $\boldsymbol{x}_0$, they largely drive the behavior of certain nonconvex optimization problems formulated for SaSD. Another important aspect of SaSD is the *shift-coherence* of its kernel,

$$\mu(\boldsymbol{a}_0) \doteq \max_{\ell \neq 0} \left|\left\langle \iota \boldsymbol{a}_0, \mathrm{s}_\ell\left[\iota \boldsymbol{a}_0\right]\right\rangle\right| \in [0, 1]\,. \tag{2}$$

Geometrically, the shifts of $\boldsymbol{a}_0$ grow further apart on the sphere as $\mu(\boldsymbol{a}_0)$ diminishes. SaSD problems are also "easier" when $\mu(\boldsymbol{a}_0)$ is small, in the sense that they can be solved with denser $\boldsymbol{x}_0$ as overlapping shifts are easier to distinguish.

2.2 Landscape geometry under shift-incoherence

A natural approach to solving SaSD is to formulate it as a suitable optimization problem. For instance, consider the *Bilinear Lasso* (BL) problem, which minimizes the squared error between the observation $\boldsymbol{y}$ and its reconstruction $\boldsymbol{a} \circledast \boldsymbol{x}$, plus a ℓ_1-norm sparsity penalty on $\boldsymbol{x}$,

$$\min_{\boldsymbol{a} \in \mathbb{S}^{p-1}, \boldsymbol{x} \in \mathbb{R}^m} \;\left[\Psi_{\mathrm{BL}}(\boldsymbol{a}, \boldsymbol{x}) \;\doteq\; \tfrac{1}{2}\left\|\boldsymbol{y} - \iota \boldsymbol{a} \circledast \boldsymbol{x}\right\|_2^2 + \lambda \left\|\boldsymbol{x}\right\|_1\right]. \tag{BL}$$

We will later see that the recovered kernel length p should be set slightly larger than p_0.

[2] We therefore assume w.l.o.g. that $\left\|\boldsymbol{a}_0\right\|_2 = 1$ in this paper.

The Bilinear Lasso is a *nonconvex* optimization problem, as the shift symmetries of SaSD create discrete local minimizers in the objective landscape. The regularizing effect created by problem symmetries is a fairly general phenomenon [6] and, as [2] shows, its influence extends far beyond local minimizers. The authors there analyze the *Dropped Quadratic* (DQ) objective[3]

$$\Psi_{\mathrm{DQ}}(\boldsymbol{a},\boldsymbol{x}) \simeq \Psi_{\mathrm{BL}}(\boldsymbol{a},\boldsymbol{x}), \qquad \text{when } \mu(\boldsymbol{a}) \simeq 0.$$

This non-practical objective is a valid simplification of the Bilinear Lasso (BL) when the true kernel is itself incoherent, i.e. $\mu(\boldsymbol{a}_0) \simeq 0$ (see Figures 1d and 1e or Section A in the appendix). We are particularly interested in the objective of the *marginalized objective*[4]

$$\min_{\boldsymbol{a}\in\mathbb{S}^{p-1}} \left[\varphi_{\mathrm{DQ}}(\boldsymbol{a}) \doteq \min_{\boldsymbol{x}\in\mathbb{R}^m} \Psi_{\mathrm{DQ}}(\boldsymbol{a},\boldsymbol{x}) \right], \tag{3}$$

which is greatly simplified when $\boldsymbol{x}$ is generic due to concentration of measure, whilst also reducing the space of optimization to a significantly smaller dimension $p \ll m$.

Regularity in the span of a few shifts. Under suitable conditions on $\boldsymbol{a}_0$ and $\boldsymbol{x}_0$, φ_{DQ} enjoys a number of nice properties on the sphere. Suppose $\boldsymbol{a} \simeq \alpha_1 s_{\ell_1}[\boldsymbol{a}_0] + \alpha_2 s_{\ell_2}[\boldsymbol{a}_0] \in \mathbb{S}^{p-1}$ is near the span of two shifts[5] of $\boldsymbol{a}_0$. If $\alpha_1 \simeq 1$ (or $\alpha_2 \simeq 0$), [2] asserts that $\boldsymbol{a}$ is in a strongly convex region of φ_{DQ}, containing a single minimizer near $s_{\ell_1}[\boldsymbol{a}_0]$, and vice versa (Figure 1a). Near the *balanced point* $\alpha_1 \simeq \alpha_2$, the influence $s_{\ell_1}[\boldsymbol{a}_0]$ and $s_{\ell_2}[\boldsymbol{a}_0]$ on φ_{DQ} creates a saddle-point, characterized by large negative curvature along the two shifts and positive curvature in orthogonal directions (Figure 1b). Between these two cases, large negative gradients point towards individual shifts.

This characterization of φ_{DQ} — strong convexity near single shifts, and saddle-points near balanced points — extends to regions of the sphere spanned by *several* shifts (Figure 1c); we elaborate more on multiple shifts in Section A.1 of the supplementary material. This regional landscape guarantees that $\boldsymbol{a}_0$ can be efficiently recovered up to a signed shift using methods for first and second-order descent, as soon as $\boldsymbol{a}$ can be brought sufficiently close to the span of a few shifts.

Optimization on the sphere. These nice properties of φ_{DQ} depend strongly on its restriction to the sphere, which creates a small but uniform increase in the *Riemannian* curvature of φ_{DQ} [7]. As a result, the sphere prevents the creation of spurious local minimizers from being created as a result of any particular points of the constraint surface posessing nonsmoothness or large positive curvature[6].

Initializing near a few shifts. The landscape structure of φ_{DQ} makes single shifts of $\boldsymbol{a}_0$ easy to locate, *if* $\boldsymbol{a}$ is initialized near a span of a few shifts. Fortunately, this is a relatively simple matter in SaSD, as $\boldsymbol{y}$ *is itself a sparse superposition of shifts*. Setting $p = 3p_0 - 2$, we initialize $\boldsymbol{a}$ by randomly choosing a length-p_0 window $\widetilde{\boldsymbol{y}}_i \doteq [y_i\ y_{i+1} \ldots y_{i+p_0-1}]^T$ and setting

$$\boldsymbol{a}^{(0)} \doteq \mathcal{P}_{\mathbb{S}^{p-1}}\big([\,\mathbf{0}_{p-1}\,;\,\widetilde{\boldsymbol{y}}_i\,;\,\mathbf{0}_{p-1}\,]\big). \tag{4}$$

This brings $\boldsymbol{a}^{(0)}$ suitably close to the sum of a few shifts of $\boldsymbol{a}_0$; any border-truncation effects are absorbed by padding the ends of $\widetilde{\boldsymbol{y}}_i$[7].

Implications for practical computation. The Dropped Quadratic problem (3) shows us an example of a nice regional landscape for SaSD where efficient recovery of $\boldsymbol{a}_0$ is guaranteed when $\mu(\boldsymbol{a}_0) \simeq 0$. As SaS applications are often motivated by sharpening or resolution tasks [10, 11, 12], however, a practical algorithm must be able to handle cases where motifs are smooth and shift-coherent (i.e., $\mu(\boldsymbol{a}_0) \approx 1$). The Dropped Quadratic is therefore a poor approximation to the Bilinear Lasso for practical purposes, yet the two problems share qualitatively similar landscapes (Figures 1d and 1e). This suggests that he algorithmic implications discussed in Section 2 — namely optimization on the sphere and data initialization — are also applicable in practical settings.

---

[3] As the intention here is apply some key messages from the Dropped Quadratic towards the Bilinear Lasso, we intentionally omit the concrete form of $\Psi_{\mathrm{DQ}}(\boldsymbol{a})$. Readers may refer to Section A for more details.

[4] $\boldsymbol{x}_0$ can be recovered via convex optimization once $\boldsymbol{a}_0$ is found.

[5] Setting $p > p_0$ ensures that $\mathbb{S}^{p-1}$ contains at least two shifts.

[6] Conversely, the popular ℓ_1-norm constraint set tends to create trivial sparse minimizers w.r.t. $\boldsymbol{a}$ [8, 9, 1].

[7] This initialization strategy is improved and made rigorous in [2] — readers may refer to Section A.2 in the supplementary material for more details.

Algorithm 1 Inertial Alternating Descent Method (iADM)

Input: Initializations $\boldsymbol{a}^{(0)} \in \mathbb{S}^{p-1}$, $\boldsymbol{x} \in \mathbb{R}^m$; observation $\boldsymbol{y} \in \mathbb{R}^m$; penalty $\lambda \geqslant 0$; momentum $\alpha \in [0, 1)$.
Output: $(\boldsymbol{a}^{(k)}, \boldsymbol{x}^{(k)})$, a local minimizer of Ψ_{BL}.

Initialize $\boldsymbol{a}^{(1)} = \boldsymbol{a}^{(0)}$, $\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)}$.
for $k = 1, 2, \ldots$ until converged **do**

Update $\boldsymbol{x}$ with accelerated proximal gradient step:

$$\begin{aligned} \boldsymbol{w}^{(k)} &\leftarrow \boldsymbol{x}^{(k)} + \alpha \cdot \left(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right) \\ \boldsymbol{x}^{(k+1)} &\leftarrow \mathrm{soft}_{\lambda t_k}\left[\boldsymbol{w}^{(k)} - t_k \cdot \nabla_{\boldsymbol{x}} \psi_\lambda\left(\boldsymbol{a}^{(k)}, \boldsymbol{w}^{(k)}\right)\right], \end{aligned}$$

where $\mathrm{soft}_\lambda(\boldsymbol{v}) \doteq \mathrm{sign}(\boldsymbol{v}) \odot \max(|\boldsymbol{v} - \lambda|, \boldsymbol{0})$ denotes the soft-thresholding operator.

Update $\boldsymbol{a}$ with accelerated Riemannian gradient step:

$$\begin{aligned} \boldsymbol{z}^{(k)} &\leftarrow \mathcal{P}_{\mathbb{S}^{p-1}}\left(\boldsymbol{a}^{(k)} + \tfrac{\alpha}{\langle \boldsymbol{a}^{(k)}, \boldsymbol{a}^{(k-1)} \rangle} \cdot \mathcal{P}_{\boldsymbol{a}^{(k-1)}}\left(\boldsymbol{a}^{(k)}\right)\right) \\ \boldsymbol{a}^{(k+1)} &\leftarrow \mathcal{P}_{\mathbb{S}^{p-1}}\left(\boldsymbol{z}^{(k)} - \tau_k \cdot \mathrm{grad}_{\boldsymbol{a}}\, \psi_\lambda\left(\boldsymbol{z}^{(k)}, \boldsymbol{x}^{(k+1)}\right)\right). \end{aligned}$$

end for

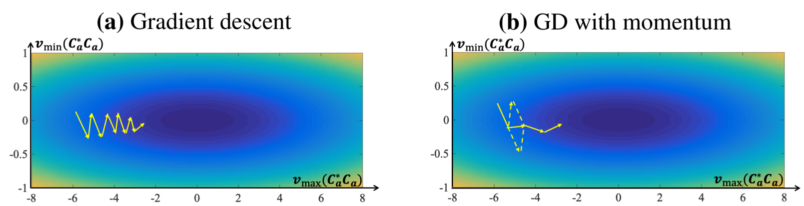

Figure 2: **Momentum acceleration.** a) Iterates of gradient descent oscillate on ill-conditioned functions. b) Momentum dampens oscillation and speeds up convergence.

3 Designing a practical SaSD algorithm

In this section, we borrow the algorithmic implications from the Dropped Quadratic (3) and build an algorithm based on the Bilinear Lasso (BL), which more accurately accounts for interactions between (highly coherent) shifts of the ground truth. We show how to address the negative effects of large coherence using a number of heuristics, leading to an efficient algorithm for SaSD.

Several algorithms for SaSD-type problems have been developed for specific applications, such as image deblurring [8, 4, 12] and neuroscience [13, 14, 5], and image super-resolution [15, 16, 17], or are augmented with additional structure [18, 19, 20]. Here we will isntead attempt to leverage recent developments in algorithmic theory in SaSD (Section 2) to build an algorithm that performs well in general practical settings.

3.1 Solving Bilinear Lasso with accelerated alternating descent

When $\boldsymbol{a}_0$ is shift-coherent, the Hessian of Ψ_{BL} becomes ill-conditioned as $\boldsymbol{a}$ converges to single shifts. Such situations are known to cause slow convergence for first-order methods [21]. One remedy is to add *momentum* [22, 23] to standard first-order iterations. For instance, consider augmenting gradient descent, on some smooth $f(\boldsymbol{z})$ with stepsize τ, using the term $\boldsymbol{w}$,

$$\boldsymbol{w}^{(k)} \leftarrow \boldsymbol{z}^{(k)} + \alpha \cdot (\boldsymbol{z}^{(k)} - \boldsymbol{z}^{(k-1)}) \tag{5}$$

$$\boldsymbol{z}^{(k+1)} \leftarrow \boldsymbol{w}^{(k)} - \tau \cdot \nabla f(\boldsymbol{w}^{(k)}). \tag{6}$$

Here, α controls the momentum added[8]. As illustrated in Figure 2, this additional term improves convergence by reducing oscillations of the iterates for ill-conditioned problems. Momentum has been shown to improve convergence for nonconvex and nonsmooth problems [24, 25]. In Algorithm 1, we provide an inertial alternating descent method (iADM) for finding local minimizers of Ψ_{BL}. It modifies iPALM [24] to perform updates on $\boldsymbol{a}$ via retraction on the sphere [7][9].

---

[8]Setting $\alpha = 0$ removes momentum and reverts to standard gradient descent.

[9]The stepsizes t_k and τ_k are obtained by backtracking [26, 24] to ensure sufficient decrease for $\Psi_{\mathrm{BL}}\left(\boldsymbol{a}^{(k)}, \boldsymbol{w}^{(k)}\right) - \Psi_{\mathrm{BL}}\left(\boldsymbol{a}^{(k)}, \boldsymbol{x}^{(k+1)}\right)$ and $\Psi_{\mathrm{BL}}\left(\boldsymbol{z}^{(k)}, \boldsymbol{x}^{(k+1)}\right) - \Psi_{\mathrm{BL}}\left(\boldsymbol{a}^{(k+1)}, \boldsymbol{w}^{(k+1)}\right)$.

Algorithm 2 SaS-BD with homotopy continuation

Input: Observation $\boldsymbol{y} \in \mathbb{R}^m$, motif size p_0; momentum $\alpha \in [0,1)$; initial $\lambda^{(1)}$ final $\lambda^\star$, penalty decrease $\eta \in (0,1)$; precision factor $\delta \in (0,1)$.

Output: Solution path $\big\{(\hat{\boldsymbol{a}}^{(n)}, \hat{\boldsymbol{x}}^{(n)}; \lambda^{(n)})\big\}$ for SaSD.

Set number of iterations $N \leftarrow \lfloor \log(\lambda^\star/\lambda^{(1)}) \,/\, \log \eta \rfloor$.

Initialize $\hat{\boldsymbol{a}}^{(0)} \in \mathbb{R}^{3p_0-2}$ using Equation (4), $\hat{\boldsymbol{x}}^{(0)} = \boldsymbol{0} \in \mathbb{R}^m$.

for $n = 1, \ldots, N$ **do**

Minimize $\Psi_{\lambda^{(n)}}$ to precision $\delta\lambda^{(n)}$ with Algorithm 1:

$$\big(\hat{\boldsymbol{a}}^{(n)}, \hat{\boldsymbol{x}}^{(n)}\big) \leftarrow \mathrm{iADM}\big(\, \hat{\boldsymbol{a}}^{(n-1)}, \hat{\boldsymbol{x}}^{(n-1)}; \boldsymbol{y}, \lambda^{(n)}, \alpha \,\big).$$

Update $\lambda^{(n+1)} \leftarrow \eta\lambda^{(n)}$.

end for

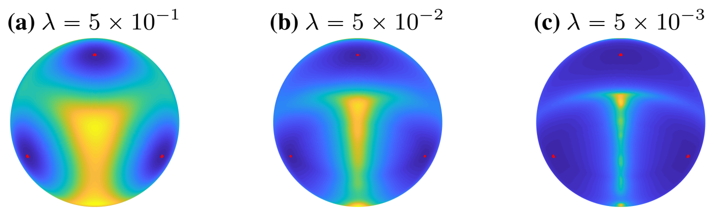

Figure 3: **Bilinear-lasso objective φ_λ on the sphere** $\mathbb{S}^{p-1}$, for $p = 3$ and varying λ. The function landscape of φ_λ flattens as sparse penalty λ decreases from left to right.

3.2 A SaSD algorithm using homotopy continuation

It is also possible to improve optimization by modifying the objective Ψ_{BL} directly through the sparsity penalty λ. Variations of this idea appear in both [1] and [2], and can also help to mitigate the effects of large shift-coherence in practical problems.

When solving (BL) in the noise-free case, it is clear that larger choices of λ encourage sparser solutions for $\boldsymbol{x}$. Conversely, smaller choices of λ place local minimizers of the marginal objective $\varphi_{\mathrm{BL}}(\boldsymbol{a}) \doteq \min_{\boldsymbol{x}} \Psi_{\mathrm{BL}}(\boldsymbol{a}, \boldsymbol{x})$ closer to signed-shifts of $\boldsymbol{a}_0$ by emphasizing reconstruction quality. When $\mu(\boldsymbol{a}_0)$ is large, however, φ_{BL} becomes ill-conditioned as $\lambda \to 0$ due to the poor spectral conditioning of $\boldsymbol{a}_0$, leading to severe flatness near local minimizers (Figure 3) and the creation spurious local minimizers when noise is present. At the expense of precision, larger values of λ limit $\boldsymbol{x}$ to a small set of support patterns and simplify the landscape ofφ_{BL}. It is therefore important both for fast convergence and accurate recovery for λ to be chosen appropriately.

When problem parameters – such as the severity of noise, or p_0 and θ – are not known a priori, a *homotopy continuation method* [27, 28, 29] can be used to obtain a *range* of solutions for SaSD. Using the initialization (4), a rough estimate $(\hat{\boldsymbol{a}}^{(1)}, \hat{\boldsymbol{x}}^{(1)})$ is first obtained by solving (BL) with iADM using a large choice for $\lambda^{(1)}$; this estimate is refined by gradually decreasing $\lambda^{(n)}$ to produce the *solution path* $\big\{(\hat{\boldsymbol{a}}^{(n)}, \hat{\boldsymbol{x}}^{(n)}; \lambda^{(n)})\big\}$. Homotopy also ensures that $\boldsymbol{x}$ remains sparse along the solution path, effectively providing the objective Ψ_{BL} with (restricted) strong convexity w.r.t. both $\boldsymbol{a}$ and $\boldsymbol{x}$ throughout optimization [30]. As a result, homotopy achieves linear convergence for SaSD where sublinear convergence is expected otherwise (Figures 4c and 4d). In Algorithm 2, we provide a complete algorithm for SaSD combining Bilinear Lasso and homotopy continuation.

4 Experiments

4.1 Synthetic experiments

We begin with simulations of SaSD in both coherent and incoherent settings. For coherent settings we use a descretized Gaussian kernel $\boldsymbol{a}_0 = \boldsymbol{g}_{n_0,2}$, where $\boldsymbol{g}_{p,\sigma} \doteq \mathcal{P}_{\mathbb{S}^{p-1}}\Big(\Big[\exp\Big(-\frac{(2i-p-1)^2}{\sigma^2(p-1)^2}\Big)\Big]_{i=1}^{p}\Big)$. Incoherent kernels are simulated by sampling $\boldsymbol{a}_0 \sim \mathrm{Unif}(\mathbb{S}^{n_0-1})$ uniformly on the sphere.

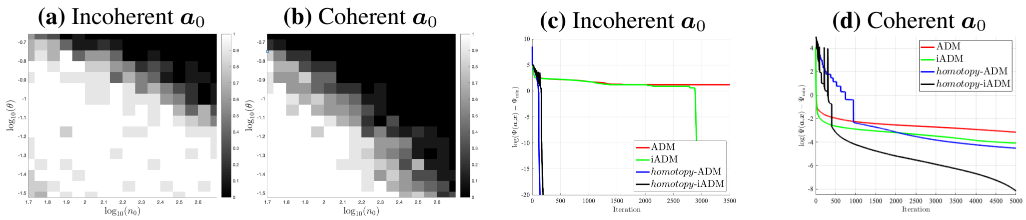

Figure 4: **Synthetic experiments for Bilinear Lasso.** *Success probability* **(a, b)***:* $\boldsymbol{x}_0 \sim_{\text{i.i.d.}} \mathcal{BR}(\theta)$, the success probability of SaS-BD by solving (BL), shown by increasing brightness, is large when the sparsity rate θ is sufficiently small compared to the length of $\boldsymbol{a}_0$, and vice versa. Success with a fixed sparsity rate is more likely when $\boldsymbol{a}_0$ is incoherent. *Algorithmic convergence* **(c, d)***:* convergence of function value for iADM with $\alpha_k = (k-1)/(k+1)$ vs. $\alpha_k = 0$ (ADM); with and without homotopy. Homotopy significantly improves convergence rate, and momentum improves convergence when $\boldsymbol{a}_0$ is coherent.

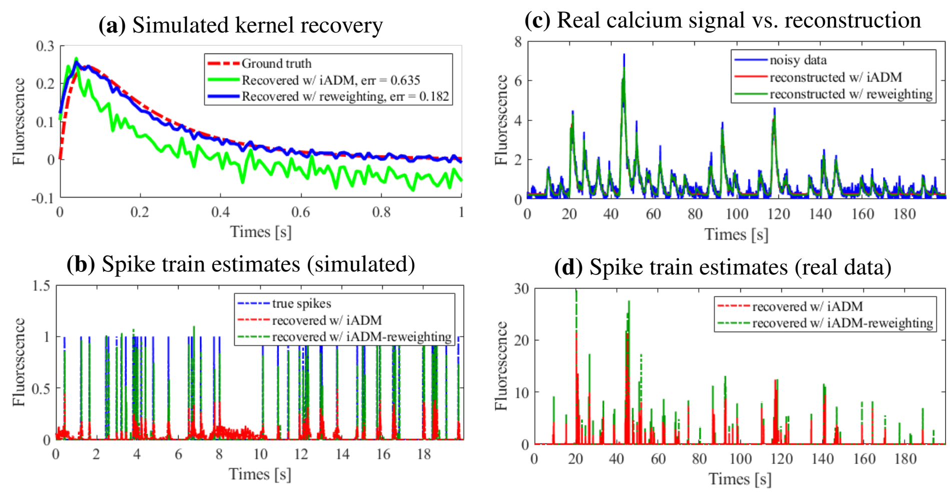

Figure 5: **Deconvolution for calcium imaging** using Algorithm 2 with iADM and with reweighting (Section B, supplementary material). *Simulated data:* **(a)** recovered AR2 kernel; **(b)** estimate of spike train. *Real data:* **(c)** reconstructed calcium signal **(d)** estimate of spike train. Reweighting improves estimation quality in each case.

4.1.1 Recovery performance

We test recovery probability for varying kernel lengths n_0 and sparsity rates θ. To ensure the problem size is sufficiently large, we set $m = 100n_0$. For each n_0 and θ, we randomly generate[10] $\boldsymbol{x} \sim_{\text{i.i.d.}} \mathcal{BR}(\theta)$ for both coherent and incoherent $\boldsymbol{a}_0$. We solve ten trials of (BL) on clean observation data $\boldsymbol{a}_0 \circledast \boldsymbol{x}_0$ using iADM with $\lambda = \frac{10^{-2}}{\sqrt{n_0\theta}}$. The probability of recovering a signed shift of $\boldsymbol{a}_0$ is shown in Figure 4. Recovery is likely when sparsity is low compared to the kernel length. The coherent problem setting has a smaller success region compared to the incoherent setting.

4.1.2 Momentum and homotopy

We demonstrate the effects of momentum acceleration and homotopy on convergence of the objective Ψ_λ. We deconvolve clean observations with $n_0 = 10^2$, $m = 10^4$, and $\theta = n_0^{-3/4}$ for both coherent and incoherent $\boldsymbol{a}_0$. Algorithm 1 with data initialization is used to solve (BL) with $\lambda = \frac{0.3}{\sqrt{n_0\theta}}$, with and without momentum ($\alpha = 0$) and homotopy. For iADM with momentum, we use an iteration dependent $\alpha_k = \frac{k-1}{k+2}$ [24]. With homotopy, we apply Algorithm 2 with the hyperparameters $\lambda^{(1)} = \max_\ell |\langle s_\ell[\boldsymbol{a}^{(0)}], \boldsymbol{y}\rangle|$ [29], $\lambda^\star = \frac{0.3}{\sqrt{n_0\lambda}}$, $\eta = 0.8$, and $\delta = 0.1$. The final solve of (BL), regardless of method, uses a precision of $\varepsilon^\star = 10^{-6}$. The results show the effectiveness of momentum and homotopy on coherent problem settings, see Figures 4c and 4d.

[10] $\mathcal{BR}(\theta)$ denotes the Bernoulli-Rademacher distribution, which has values ± 1 w.p. $\theta/2$ and zero w.p. $1-\theta$.

4.2 Imaging applications

We demonstrate the performance and generality of the proposed method with experiments on calcium fluorescence imaging, a popular modality for studying spiking activity in large neuronal populations [31], and stochastic optical reconstruction microscopy (STORM) [32, 33, 34], a superresolution microscopy modality used to image structures within living cells.[11]

4.2.1 Sparse deconvolution of calcium signals

Neural spike trains are temporal signals created by action potentials, which induce a transient change in the amount of calcium present in the surrounding environment. The observed calcium concentration can be modeled as a convolution $\boldsymbol{a}_0 \circledast \boldsymbol{x}_0$ of the transient response $\boldsymbol{a}_0$ and the spike train $\boldsymbol{x}_0$. Typically neither $\boldsymbol{a}_0$ nor $\boldsymbol{x}_0$ is perfectly known ahead of time.

We first test our method on synthetic data generated according to an AR2 model for $\boldsymbol{a}_0$, which was used for deconvolution in [14]. This kernel is highly shift-coherent and poses a challenging problem setting for SaSD. Here, $\boldsymbol{x}_0 \sim_{\text{i.i.d.}} \text{Bernoulli}(n_0^{-4/5}) \in \mathbb{R}^{10^4}$ and additive noise is generated as $\boldsymbol{n} \sim_{\text{i.i.d.}} \mathcal{N}(0, 5 \cdot 10^{-2})$. Figures 5a and 5b demonstrate accurate recovery of both $\boldsymbol{a}_0$ and $\boldsymbol{x}_0$ in this synthetic setting. We next test our method on real data[12]; Figures 5c and 5d demonstrate recovery of spike locations in the real setting. Although iADM provides decent performance in each case, noise suppression and estimation quality can be improved by stronger sparsification methods, such as the reweighting technique [37] — see Section B of the supplementary material.

4.2.2 Sparse blind deconvolution for super-resolution fluorescence microscopy

The spatial resolution of fluorescence microscopy is often limited by the diffraction of light: its wavelength (i.e. several hundred nanometers) is often larger than typical molecular length-scales in cells, preventing a detailed characterization of most subcellular structures.

The STORM technique is developed to overcome this resolution limit. Instead of activating all the fluorophores at the same time, STORM multiplexes the image by randomly activating photoswitchable fluorescent probes over multiple frames, each containing a subset of the molecules present (Figure 6). If the location of these molecules can be precisely determined for each frame, synthesizing all deconvolved frames will produce a super-resolution microscopy image with nanoscale resolution.

For each image frame, the localization task can be formulated via the SaS model

$$\underbrace{\boldsymbol{Y}_t}_{\text{STORM frame}} = \underbrace{\iota \boldsymbol{A}_0}_{\text{point spread function}} \boxed{*} \underbrace{\boldsymbol{X}_{0,t}}_{\text{sparse point sources}} + \underbrace{\boldsymbol{N}_t}_{\text{noise}}. \tag{7}$$

Here we will solve the localization task on the single-molecule localization microscopy (SMLM) benchmarking dataset[13] via SaSD, recovering both the PSF $\boldsymbol{A}_0$ and the point source map $\boldsymbol{X}_{0,t}$ simultaneously. We apply iADM with reweighting (see Section B of the supplementary material) on frames from the video sequence "Tubulin" containing 500 frames of size 128×128, where each pixel is of 100nm^2 resolution[14]; the fluorescence wavelength is 690nm and the imaging frequency is $f = 25\text{Hz}$. The recovered activation maps individual time frames and the aggregated super-resolution image is shown in Figure 6. These results demonstrate that our approach can accurately predict the PSF and the activation map for each video frame, producing higher resolution microscopy images[15].

4.2.3 Localizing neurons in calcium images

Our methods are easily extended to handle superpositions of multiple SaS signals. In calcium imaging, this can potentially be used to track the neurons in video sequences, a challenging task due to (non-)rigid motion, overlapping sources, and irregular background noise [38, 39]. We consider frames

[11] Other such methods developed for similar modalities include photoactivated localization microscopy (PALM) [35], and fluorescence photoactivation localization microscopy (fPALM) [36].

[12] Obtained at `http://spikefinder.codeneuro.org`.

[13] Data can be accessed at `http://bigwww.epfl.ch/smlm/datasets/index.html`.

[14] Here we solve SaSD on the same 128×128 grid. In practice, the localization problem is solved on a finer grid, so that the resulting resolution can reach $20 - 30$ nm.

[15] The recovered PSF is provided in Section C in the supplementary material.

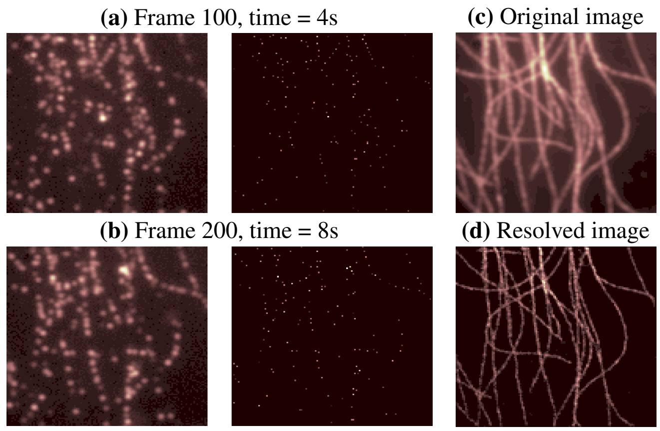

Figure 6: **SaSD for STORM imaging. (a, b)** Individual frames (left) and predicted point process map using SaSD (right). **(c, d)** shows the original microscopy and the super-resolved image obtained by our method.

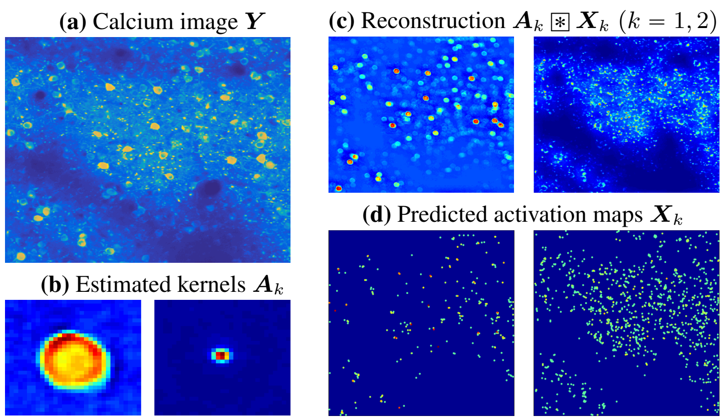

Figure 7: **Classification of calcium images. (a)** Original calcium image; **(b)** respective kernel estimates; **(c)** reconstructed images with the **(left)** neuron and **(right)** dendrite kernels; **(d)** respective occurence map estimates.

video obtained via the two-photon calcium microscopy dataset from the Allen Institute for Brain Science[16], shown in Figure 7. Each frame contains the cross section of several neurons and dendrites, which have distinct sizes. We model this as the SaS signal $\boldsymbol{Y}_t = \iota \boldsymbol{A}_1 \circledast \boldsymbol{X}_{1,t} + \iota \boldsymbol{A}_2 \circledast \boldsymbol{X}_{2,t}$, where each summand consists of neurons or dendrites exclusively. By extending Algorithm 2 to recover each of the kernels $\boldsymbol{A}_k$ and maps $\boldsymbol{X}_k$, we can solve this *convolutional dictionary learning* (SaS-CDL) problem which allows us to seperate the dendritic and neuronal components from this image for localization of firing activity, etc. As a result, the application of SaS-CDL as a denoising or analysis tool for calcium imaging videos provides a very promising direction for future research.

5 Discussion

Many nonconvex inverse problems — including SaSD — are strongly regulated by their problem symmetries. Understanding this regularity (and when and how it breaks down) can provide a strong basis for developing effective algorithms. In this paper, we have attempted to illustrate this point by combining geometric intuition with practical heuristics motivated by common challenges in real deconvolution to produce an efficient, general purpose method that performs well on data arising from a range of application areas. Our approach, therefore, can serve as a general baseline for studying and developing extensions to SaSD, such as SaS-CDL [40, 41, 42] and Bayesian models [43, 18].

[16]Obtained at http://observatory.brain-map.org/visualcoding/.

References

[1] Y. Zhang, Y. Lau, H.-w. Kuo, S. Cheung, A. Pasupathy, and J. Wright, "On the global geometry of sphere-constrained sparse blind deconvolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4894–4902.

[2] H.-W. Kuo, Y. Zhang, Y. Lau, and J. Wright, "Geometry and symmetry in short-and-sparse deconvolution," in *International Conference on Machine Learning (ICML)*, June 2019.

[3] S. Cheung, Y. Lau, Z. Chen, J. Sun, Y. Zhang, J. Wright, and A. Pasupathy, "Beyond the fourier transform: A nonconvex optimization approach to microscopy analysis," *Submitted*, 2017.

[4] D. Briers, D. D. Duncan, E. R. Hirst, S. J. Kirkpatrick, M. Larsson, W. Steenbergen, T. Stromberg, and O. B. Thompson, "Laser speckle contrast imaging: theoretical and practical limitations," *Journal of biomedical optics*, vol. 18, no. 6, p. 066018, 2013.

[5] A. H. Song, F. Flores, and D. Ba, "Spike sorting by convolutional dictionary learning," *arXiv preprint arXiv:1806.01979*, 2018.

[6] J. Sun, Q. Qu, and J. Wright, "When are nonconvex problems not scary?" *arXiv preprint arXiv:1510.06096*, 2015.

[7] P.-A. Absil, R. Mahoney, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2009.

[8] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1964–1971.

[9] A. Benichoux, E. Vincent, and R. Gribonval, "A fundamental pitfall in blind deconvolution with sparse and shift-invariant priors," in *ICASSP-38th International Conference on Acoustics, Speech, and Signal Processing-2013*, 2013.

[10] B. Huang, M. Bates, and X. Zhuang, "Super-resolution fluorescence microscopy," *Annual review of biochemistry*, vol. 78, pp. 993–1016, 2009.

[11] E. Candes and C. Fernandez-Granda, "Towards a mathematical theory of super-resolution," *Communications on Pure and Applied Mathematics*, vol. 67, no. 6, pp. 906–956, 2014.

[12] P. Campisi and K. Egiazarian, *Blind image deconvolution: theory and applications*. CRC press, 2016.

[13] H. G. Rey, C. Pedreira, and R. Q. Quiroga, "Past, present and future of spike sorting techniques," *Brain research bulletin*, vol. 119, pp. 106–117, 2015.

[14] J. Friedrich, P. Zhou, and L. Paninski, "Fast online deconvolution of calcium imaging data," *PLoS computational biology*, vol. 13, no. 3, p. e1005423, 2017.

[15] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1167–1183, 2002.

[16] G. Shtengel, J. A. Galbraith, C. G. Galbraith, J. Lippincott-Schwartz, J. M. Gillette, S. Manley, R. Sougrat, C. M. Waterman, P. Kanchanawong, M. W. Davidson, *et al.*, "Interferometric fluorescent super-resolution microscopy resolves 3d cellular ultrastructure," *Proceedings of the National Academy of Sciences*, vol. 106, no. 9, pp. 3125–3130, 2009.

[17] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE transactions on image processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[18] D. Wipf and H. Zhang, "Revisiting bayesian blind deconvolution," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3595–3634, 2014.

[19] S. Ling and T. Strohmer, "Blind deconvolution meets blind demixing: Algorithms and performance bounds," *IEEE Transactions on Information Theory*, vol. 63, no. 7, pp. 4497–4520, 2017.

[20] P. Walk, P. Jung, G. E. Pfander, and B. Hassibi, "Blind deconvolution with additional autocorrelations via convex programs," *arXiv preprint arXiv:1701.04890*, 2017.

[21] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.

[22] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1–17, 1964.

[23] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM journal on imaging sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[24] T. Pock and S. Sabach, "Inertial proximal alternating linearized minimization (ipalm) for nonconvex and nonsmooth problems," *SIAM Journal on Imaging Sciences*, vol. 9, no. 4, pp. 1756–1787, 2016.

[25] C. Jin, P. Netrapalli, and M. I. Jordan, “Accelerated gradient descent escapes saddle points faster than gradient descent,” *arXiv preprint arXiv:1711.10456*, 2017.

[26] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[27] E. T. Hale, W. Yin, and Y. Zhang, “Fixed-point continuation for \ell_1-minimization: Methodology and convergence,” *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1107–1130, 2008.

[28] S. J. Wright, R. D. Nowak, and M. A. Figueiredo, “Sparse reconstruction by separable approximation,” *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, 2009.

[29] L. Xiao and T. Zhang, “A proximal-gradient homotopy method for the sparse least-squares problem,” *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1062–1091, 2013.

[30] A. Agarwal, S. Negahban, and M. J. Wainwright, “Fast global convergence rates of gradient methods for high-dimensional statistical recovery,” in *Advances in Neural Information Processing Systems*, 2010, pp. 37–45.

[31] C. Grienberger and A. Konnerth, “Imaging calcium in neurons,” *Neuron*, vol. 73, no. 5, pp. 862–885, 2012.

[32] M. J. Rust, M. Bates, and X. Zhuang, “Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (storm),” *Nature methods*, vol. 3, no. 10, p. 793, 2006.

[33] B. Huang, W. Wang, M. Bates, and X. Zhuang, “Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy,” *Science*, vol. 319, no. 5864, pp. 810–813, 2008.

[34] B. Huang, H. Babcock, and X. Zhuang, “Breaking the diffraction barrier: super-resolution imaging of cells,” *Cell*, vol. 143, no. 7, pp. 1047–1058, 2010.

[35] E. Betzig, G. H. Patterson, R. Sougrat, O. W. Lindwasser, S. Olenych, J. S. Bonifacino, M. W. Davidson, J. Lippincott-Schwartz, and H. F. Hess, “Imaging intracellular fluorescent proteins at nanometer resolution,” *Science*, vol. 313, no. 5793, pp. 1642–1645, 2006.

[36] S. T. Hess, T. P. Girirajan, and M. D. Mason, “Ultra-high resolution imaging by fluorescence photoactivation localization microscopy,” *Biophysical journal*, vol. 91, no. 11, pp. 4258–4272, 2006.

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

[38] E. A. Pnevmatikakis, D. Soudry, Y. Gao, T. A. Machado, J. Merel, D. Pfau, T. Reardon, Y. Mu, C. Lacefield, W. Yang, *et al.*, “Simultaneous denoising, deconvolution, and demixing of calcium imaging data,” *Neuron*, vol. 89, no. 2, pp. 285–299, 2016.

[39] A. Giovannucci, J. Friedrich, P. Gunn, J. Kalfon, B. L. Brown, S. A. Koay, J. Taxidis, F. Najafi, J. L. Gauthier, P. Zhou, *et al.*, “Caiman an open source tool for scalable calcium imaging data analysis,” *Elife*, vol. 8, p. e38173, 2019.

[40] H. Bristow and S. Lucey, “Optimization methods for convolutional sparse coding,” *arXiv preprint arXiv:1406.2407*, 2014.

[41] C. Garcia-Cardona and B. Wohlberg, “Convolutional dictionary learning: A comparative review and new algorithms,” *IEEE Transactions on Computational Imaging*, 2018.

[42] J. Sulam, V. Papyan, Y. Romano, and M. Elad, “Multilayer convolutional sparse modeling: Pursuit and dictionary learning,” *IEEE Transactions on Signal Processing*, vol. 66, no. 15, pp. 4090–4104, 2018.

[43] S. D. Babacan, R. Molina, and A. K. Katsaggelos, “Variational bayesian blind deconvolution using a total variation prior,” *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 12–26, 2008.

[44] C. Jin, R. Ge, P. Netrapalli, S. M. Kakade, and M. I. Jordan, “How to escape saddle points efficiently,” *arXiv preprint arXiv:1703.00887*, 2017.

Appendices

A Dropped Quadratic

Recall from Section 2.2 of the main text that SaSD can be formulated as the Bilinear Lasso problem

$$\min_{\boldsymbol{a}\in\mathbb{S}^{p-1},\boldsymbol{x}\in\mathbb{R}^m} \left[\Psi_{\mathrm{BL}}(\boldsymbol{a},\boldsymbol{x}) \;\doteq\; \tfrac{1}{2}\left\|\boldsymbol{y}-\iota\boldsymbol{a}\circledast\boldsymbol{x}\right\|_2^2+\lambda\left\|\boldsymbol{x}\right\|_1\right]. \tag{BL}$$

Unfortunately, this objective is challenging for analysis. A major culprit is that its marginalization

$$\varphi_{\mathrm{BL}}(\boldsymbol{a}) \;\doteq\; \min_{\boldsymbol{x}}\left\{\tfrac{1}{2}\left\|\boldsymbol{y}-\iota\boldsymbol{a}\circledast\boldsymbol{x}\right\|_2^2+\lambda\left\|\boldsymbol{x}\right\|_1\right\}, \tag{8}$$

generally does not admit closed form solutions due the convolution with $\boldsymbol{a}$ in the squared error term. This motivates [2] to study the nonconvex formulation

$$\min_{\boldsymbol{a}\in\mathbb{S}^{p-1},\boldsymbol{x}\in\mathbb{R}^m} \left[\Psi_{\mathrm{DQ}}(\boldsymbol{a},\boldsymbol{x}) \;\doteq\; \tfrac{1}{2}\left\|\boldsymbol{x}\right\|_2^2-\left\langle\iota\boldsymbol{a}\circledast\boldsymbol{x},\boldsymbol{y}\right\rangle+\left\|\boldsymbol{y}\right\|_2^2+\lambda\left\|\boldsymbol{x}\right\|_1\right]. \tag{9}$$

We refer to (9) as the *Dropped Quadratic* formulation, and it is quite easy to see that $\Psi_{\mathrm{DQ}}(\boldsymbol{a},\boldsymbol{x}) \approx \Psi_{\mathrm{DQ}}(\boldsymbol{a},\boldsymbol{x})$ when $\left\|\boldsymbol{a}\circledast\boldsymbol{x}\right\|^2 \approx \left\|\boldsymbol{x}\right\|^2$, i.e. if $\boldsymbol{a}$ is shift-incoherent, or $\mu_s(\boldsymbol{a}) \approx 0$. The marginalized objective function $\varphi_{\mathrm{BL}}(\boldsymbol{a}) \doteq \min_{\boldsymbol{x}} \Psi_{\mathrm{DQ}}(\boldsymbol{a},\boldsymbol{x})$ now has the closed form expression

$$\varphi_{\mathrm{DQ}}(\boldsymbol{a}) \;\doteq\; -\tfrac{1}{2}\left\|\mathrm{soft}_\lambda\left[\check{\boldsymbol{a}}\circledast\boldsymbol{y}\right]\right\|_2^2. \tag{10}$$

Here soft denotes the soft-thresholding operator, and $\check{\boldsymbol{a}}$ denotes the *adjoint kernel* of $\boldsymbol{a}$, i.e. the kernel s.t. $\left\langle\iota\boldsymbol{a}\circledast\boldsymbol{u},\boldsymbol{v}\right\rangle=\left\langle\boldsymbol{u},\check{\boldsymbol{a}}\circledast\boldsymbol{v}\right\rangle\ \forall\boldsymbol{u},\boldsymbol{v}\in\mathbb{R}^m$.

A.1 Landscape Geometry

The rest of Section 2.2 discusses the regional characterization of φ_{DQ} in the span of a small number of shifts from $\boldsymbol{a}_0$. This language is made precise in the form of the subsphere

$$\mathcal{S}_{\mathcal{I}} \;\doteq\; \left\{\ \textstyle\sum_{\ell\in\mathcal{I}}\ \alpha_\ell s_\ell\left[\iota\boldsymbol{a}_0\right]\ :\ \alpha_\ell\in\mathbb{R}\ \right\}\ \bigcap\ \mathbb{S}^{p-1}, \tag{11}$$

spanned by a small set of cyclic shifts of $\iota\boldsymbol{a}_0$. Although we will not discuss the explicit distance function here, the characterization by [2] holds whenever $\boldsymbol{a}$ is close enough to such a subsphere with $|\mathcal{I}| \leqslant 4\theta p_0$, where θ is the probability that any individual entry of $\boldsymbol{x}_0$ is nonzero. Suppose we have $\boldsymbol{a} \approx \sum_{\ell\in\mathcal{I}}\ \alpha_\ell s_\ell\left[\iota\boldsymbol{a}_0\right]$ for some appropriate index set $\mathcal{I}$. Note that if $\mu_s\boldsymbol{a}_0 \approx 0$, then $\mu_s\boldsymbol{a} \approx 0,\ \forall\boldsymbol{a}\in\mathcal{S}_{\mathcal{I}}$. Now let $\alpha_{(1)}$ and $\alpha_{(2)}$ be the first and second largest coordinates of the shifts participating in $\boldsymbol{a}$, and let $s_{(1)}[\boldsymbol{a}_0]$ and $s_{(2)}[\boldsymbol{a}_0]$ be the corresponding shifts. Then

- If $\left|\frac{\alpha_{(2)}}{\alpha_{(1)}}\right| \approx 0$, then $\boldsymbol{a}$ is in a strongly convex region of φ_{DQ}, containing a *single* local minimizer corresponding to $s_{(1)}[\boldsymbol{a}_0]$.
- If $\left|\frac{\alpha_{(2)}}{\alpha_{(1)}}\right| \approx 1$, then $\boldsymbol{a}$ is near a saddle-point, with *negative curvature* pointing towards $s_{(1)}[\boldsymbol{a}_0]$ and $s_{(2)}[\boldsymbol{a}_0]$. If $\left|\frac{\alpha_{(3)}}{\alpha_{(2)}}\right| \approx 0$, i.e. $s_{(1)}[\boldsymbol{a}_0]$ and $s_{(2)}[\boldsymbol{a}_0]$ are the only two participating shifts, then φ_{DQ} is also characterized by *positive curvature* in all orthogonal directions.
- Otherwise, $\left\langle-\mathrm{grad}\varphi_{\mathrm{DQ}}(\boldsymbol{a}),\boldsymbol{z}-a\right\rangle$ takes on a *large positive value*, for either $u=s_{(1)}[\boldsymbol{a}_0]$ or $u=s_{(2)}[\boldsymbol{a}_0]$, i.e. the negative Riemannian gradient is large and points towards one of the participating shifts.

This is an example of a *ridable saddle* property [44] that allows many first and second-order methods to locate local minimizers. Since all local minimizers of φ_{DQ} near $\mathcal{S}_{\mathcal{I}}$ must correspond to signed-shifts of $\boldsymbol{a}_0$, this guarantees that the Dropped Quadratic formulation can be efficiently solved to recover $\boldsymbol{a}_0$ (and subsequently $\boldsymbol{x}_0$) for incoherent $\boldsymbol{a}_0$, as long as $\boldsymbol{a}$ is initialized near some appropriate subsphere and the sparsity coherence tradeoff $p_0\theta \lessapprox \left(\mu_s(\boldsymbol{a}_0)\right)^{-1/2}$ is satisfied. We note that this is a poor tradeoff rate, which reflects that the Dropped Quadratic formulation is non-practical and cannot handle SaSD problems involving kernels with high shift-coherence.

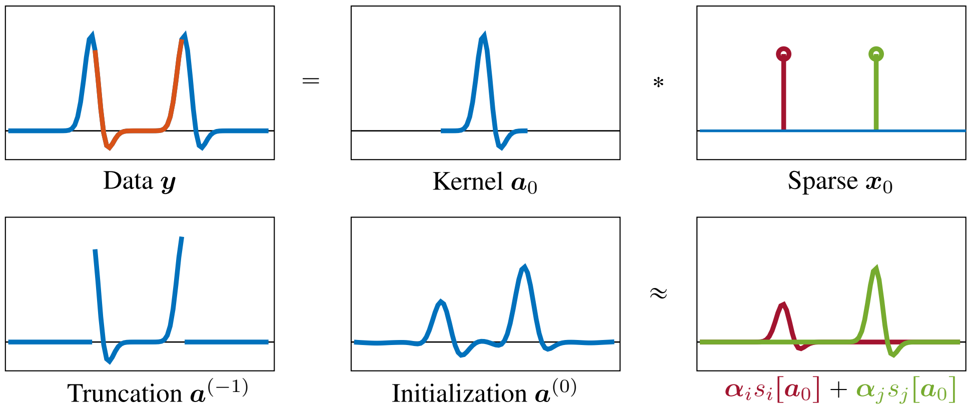

Figure 8: **Illustration of data-driven initialization for $\boldsymbol{a}$:** using a piece of the observed data $\boldsymbol{y}$ to generate a good initial point $\boldsymbol{a}^{(0)}$. Top: data $\boldsymbol{y} = \boldsymbol{a}_0 \circledast \boldsymbol{x}_0$ is a superposition of shifts of the true kernel $\boldsymbol{a}_0$. Bottom: a length-p_0 window contains pieces of just a few shifts. Bottom middle: one step of the generalized power method approximately fills in the missing pieces, yielding an initialization that is close to a linear combination of shifts of $\boldsymbol{a}_0$ (right).

A.2 Data-driven initialization

For the SaS-BD problem, we usually initialize $\boldsymbol{x}$ by $\boldsymbol{x}^{(0)} = \mathbf{0}$, so that our initialization is sparse. For the optimization variable $\boldsymbol{a} \in \mathbb{R}^n$, recall from Section 2.2 in the main text that it is desirable to obtain an initialization $\boldsymbol{a}^0$ which is close to the intersection of $\mathbb{S}^{p-1}$ and a subsphere $\mathcal{S}_{\mathcal{I}}$ spanned by a few shifts of $\boldsymbol{a}_0$. When $\boldsymbol{x}_0$ is sparse, our measurement $\boldsymbol{y}$ is a linear combination of a few shifts of $\boldsymbol{a}_0$. Therefore, an arbitrary consecutive p_0-length window $\widetilde{\boldsymbol{y}}_i \doteq [y_i \; y_{i+1} \ldots y_{i+p_0-1}]^T$ of the data $\boldsymbol{y}$ should be not far away from such a subspace $\mathcal{S}_{\mathcal{I}}$. As illustrated in Figure 8, one step of the generalized power method [2]

$$\widetilde{\boldsymbol{a}}^{(0)} \doteq \mathcal{P}_{\mathbb{S}^{p-1}}\left([\; \mathbf{0}_{p-1} \;;\; \widetilde{\boldsymbol{y}}_i \;;\; \mathbf{0}_{p-1} \;]\right) \tag{12}$$

$$\boldsymbol{a}^{(0)} = \mathcal{P}_{\mathbb{S}^{n-1}}\left(-\nabla\varphi_{\mathrm{DQ}}\left(\widetilde{\boldsymbol{a}}^{(0)}\right)\right) \tag{13}$$

produces a refined initialization that is very close to a subspace $\mathcal{S}_{\mathcal{I}}$ spanned by a few shifts of $\boldsymbol{a}_0$ with $|\mathcal{I}| \approx \theta p_0$. However, (13) is a relatively complicated for a simple idea. In practice, we find that the simple initialization $\boldsymbol{a}^{(0)} = \widetilde{\boldsymbol{a}}^{(0)}$ from (12) works suitably well for solving SaSD with (BL).

B Reweighted sparse penalization

When $\boldsymbol{a}_0$ is shift-coherent, minimization of the objective Ψ_{BL} with respect to $\boldsymbol{x}$ becomes sensitive to perturbations, creating “smudging” effects on the recovered map $\boldsymbol{x}$. These resolution issues can be remedied with stronger *concave* regularizers. A simple way of facilitating this with the Bilinear Lasso is to use a reweighting techinque [37]. The basic idea is to adaptively adjust the penalty by considering a weighted variant of the original Bilinear Lasso problem from (BL),

$$\min_{\boldsymbol{a}\in\mathbb{S}^{p-1},\boldsymbol{x}\in\mathbb{R}^m} \quad \Psi_{\mathrm{BL}}^{\boldsymbol{w}}(\boldsymbol{a},\boldsymbol{x}) \;\doteq\; \tfrac{1}{2}\left\|\boldsymbol{y} - \boldsymbol{a}\circledast\boldsymbol{x}\right\|_2^2 + \lambda\left\|\boldsymbol{w}\odot\boldsymbol{x}\right\|_1 \tag{14}$$

where $\boldsymbol{w} \in \mathbb{R}_+^m$ and $\odot$ denotes the Hadamard product. Here we will set the weights $\boldsymbol{w}$ to be roughly inverse to the magnitude of the true signal $\boldsymbol{x}_0$, i.e.,

$$w_i = \frac{1}{|x_{0,i}| + \varepsilon}. \tag{15}$$

In addition to choosing $\lambda > 0$, here $\varepsilon > 0$ trades off between sparsification strength (small ε) and algorithmic stability (large ε). Let $|x|_{(i)}$ denote the i-th largest entry of $|\boldsymbol{x}|$. For experiments in the main text, we set

$$\varepsilon = \max\left\{|x|_{(\lceil n/\log(m/n)\rceil)}, 10^{-3}\right\}.$$

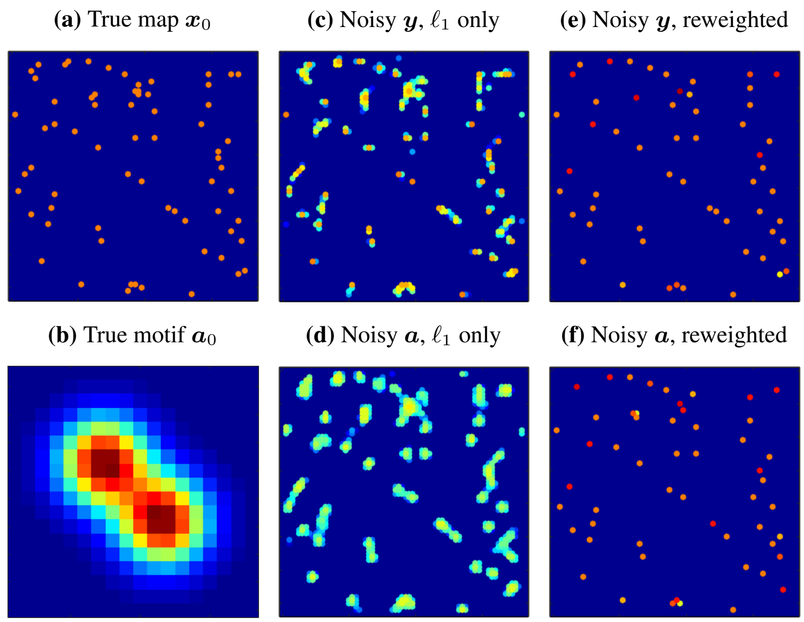

Figure 9: **Recovery of $\boldsymbol{x}_0$ with ℓ_1-reweighting. (a, b)** Truth signals. **(c)** Solving $\min_{\boldsymbol{x}} \Psi_{\mathrm{BL}}(\boldsymbol{a}, \boldsymbol{x})$ with noisy data and coherent $\boldsymbol{a}_0$ leads to low-quality estimates of $\boldsymbol{x}$; **(d)** performance suffers further when $\boldsymbol{a}$ is a noisy estimate of $\boldsymbol{a}_0$. **(e, f)** Reweighted ℓ_1 minimization alleviates this issue significantly.

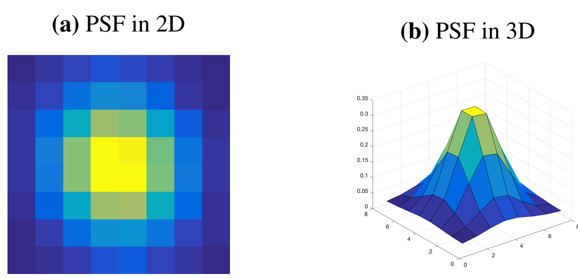

Figure 10: **Estimated PSF for STORM imaging.** The left hand side shows the estimated 8×8 PSF in 2D, the right hand side visualizes the PSF in 3D.

Starting with the initial weights $\boldsymbol{w}^{(0)} = \mathbf{1}_m$, Algorithm 3 successively solves (14), updating the weights using (15) at each outer loop iteration j. As $j \to \infty$, this method eventually becomes equivalent to replacing the ℓ_1-norm in (BL) with the nonconvex penalty $\sum_i \log(|x_i| + \varepsilon)$ [37].

Algorithm 3 Reweighted Bilinear Lasso

Input: Initializations $\hat{\boldsymbol{a}}^{(0)}, \hat{\boldsymbol{x}}^{(0)}$, penalty $\lambda > 0$

Output: Local minimizers $\hat{\boldsymbol{a}}^{(j)}, \hat{\boldsymbol{x}}^{(j)}$ of $\Psi_{\mathrm{BL}}^{\boldsymbol{w}^{(j)}}$.

Initialize $\boldsymbol{w}^{(1)} = \mathbf{1}_m$, $j \leftarrow 1$.

while not converged **do**

Using the initialization $\left(\hat{\boldsymbol{a}}^{(j-1)}, \hat{\boldsymbol{x}}^{(j-1)}\right)$ and weight $\boldsymbol{w}^{(j)}$, solve (14) — e.g. with iADM — to obtain solution $\left(\hat{\boldsymbol{a}}^{(j)}, \hat{\boldsymbol{x}}^{(j)}\right)$;

Set ε with (16) and update the weights as

$$\boldsymbol{w}^{(j+1)} = \frac{1}{\left|\hat{\boldsymbol{x}}^{(j)}\right| + \varepsilon}. \tag{16}$$

Update $\ell \leftarrow \ell + 1$.

end while

We could easily adopt our iADM algorithm to solve this subproblem, by taking the proximal gradient on x with different penalty λ_i for each entry x_i. Figure 9, as well as calcium imaging experiments in Section 4.2, Figure 5 of the main text, demonstrate improved estimation quality as a result of this method.

C Super-resolution with STORM imaging

For point source localization in STORM frames, recall that we use the SaS model from Section 4.2.2,

$$\underbrace{\boldsymbol{Y}_t}_{\text{STORM frame}} = \underbrace{\iota \boldsymbol{A}_0}_{\text{point spread function}} \boxast \underbrace{\boldsymbol{X}_{0,t}}_{\text{sparse point sources}} + \underbrace{\boldsymbol{N}_t}_{\text{noise}}. \tag{17}$$

We then apply our SaSD method to recover both $\boldsymbol{A}_0$ and $\boldsymbol{X}_{0,t}$ from $\boldsymbol{Y}_t$. We show our recovery of $\boldsymbol{X}_{0,t}$ as well as the super-resolved image using all available frames in Figure 6 of the main text. Since the main objective of STORM imaging is to recover the point sources, we have deferred the recovered PSF $\boldsymbol{A}_0$ to Figure 10 here.