Applied and Computational Harmonic Analysis 71 (2024) 101653

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

journal homepage: www.elsevier.com/locate/acha

Letter to the Editor

Eigenmatrix for unstructured sparse recovery

Lexing Ying [1]

Department of Mathematics, Stanford University, Stanford, CA 94305, United States of America

ARTICLE INFO

Communicated by Radu Balan

MSC:
30B40
65R32

Keywords:
Sparse recovery
Prony's method
ESPRIT algorithm

ABSTRACT

This note considers the unstructured sparse recovery problems in a general form. Examples include rational approximation, spectral function estimation, Fourier inversion, Laplace inversion, and sparse deconvolution. The main challenges are the noise in the sample values and the unstructured nature of the sample locations. This note proposes the eigenmatrix, a data-driven construction with desired approximate eigenvalues and eigenvectors. The eigenmatrix offers a new way for these sparse recovery problems. Numerical results are provided to demonstrate the efficiency of the proposed method.

1. Introduction

This note considers the unstructured sparse recovery problems of a general form. Let X be the parameter space, typically a subset of $\mathbb{R}$ or $\mathbb{C}$, and S be the sampling space. $G(s,x)$ is the kernel function for $s \in S$ and $x \in X$, and is assumed to be analytic in x. Suppose that

$$f(x) = \sum_{k=1}^{n_x} w_k \delta(x - x_k)$$

is the unknown sparse signal, where n_x is the number of spikes, $\{x_k\}$ are the spike locations, and $\{w_k\}$ are the spike weights. The observable of the problem is

$$u(s) := \int_X G(s,x) f(x) dx = \sum_{k=1}^{n_x} G(s,x_k) w_k$$

for $s \in S$.

Let $\{s_j\}$ be a set of n_s unstructured sample locations in S and $u_j := u(s_j)$ be the exact values. Suppose that we are only given the noisy observations $\tilde{u}_j := u_j(1+\sigma Z_j)$, where Z_j are independently identically distributed (i.i.d.) random variables with zero mean and unit variance, and σ is the noise magnitude. The task is to recover the spike locations $\{x_k\}$ and weights $\{w_k\}$.

Quite a few sparse recovery problems can be cast into this general form. Below is a partial list.

E-mail address: lexing@stanford.edu.

[1] This work is partially supported by NSF grants DMS-2011699 and DMS-2208163. The author thanks Laurent Demanet and Lin Lin for helpful discussions and the anonymous referee for constructive comments.

https://doi.org/10.1016/j.acha.2024.101653
Received 1 December 2023; Received in revised form 26 February 2024; Accepted 4 March 2024

- Rational approximation. $G(s,x) = \frac{1}{s-x}$, X is typically a set in $\mathbb{C}$, and $\{s_j\}$ are locations separated from X. Two common cases of X are the unit disk and the half-plane.
- Spectral function estimation of many-body quantum systems. $G(s,x) = \frac{1}{s-x}$, X is a real interval $[-b,b]$ of the complex plane, and $\{s_j\}$ is the Matsubara grid on the imaginary axis.
- Fourier inversion. For example $G(s,x) = \exp(\pi isx)$, X is the interval $[-1,1]$, and $\{s_j\}$ is a set of real numbers.
- Laplace inversion. $G(s,x) = x\exp(-sx)$, X is an interval $[c_1,c_2]$ of the positive real axis, and $\{s_j\}$ is a set of positive real numbers.
- Sparse deconvolution. $G(s,x)$ is a translational invariant kernel, such as $G(s,x) = \frac{1}{1+\gamma(s-x)^2}$, X is a real interval, and $\{s_j\}$ is a set of real numbers.

The primary challenges of the current setup come from two sources. First, the kernel $G(s,x)$ can be quite general. Second, the sample locations $\{s_j\}$ are unstructured, which excludes many existing algorithms that exploit special structures. Third, the sample values $\{\tilde{u}_j\}$ are noisy, which raises stability issues when the recovery problem is quite ill-posed.

1.1. Contribution

This note introduces the *eigenmatrix* for these unstructured sparse recovery problems. By defining the vector-valued function $\mathbf{g}(x) = [G(s_j,x)]_{1\le j\le n_s}$ for $x \in X$, we introduce the eigenmatrix as an $n_s \times n_s$ matrix M that satisfies, for any $x \in X$

$$M\mathbf{g}(x) \approx x\mathbf{g}(x).$$

This is a data-driven object that depends on $G(\cdot,\cdot)$, X, and the sample locations $\{s_j\}$. The main features of the eigenmatrix are

- It assumes no special structure of the sample locations $\{s_j\}$.
- It offers a rather unified approach to these sparse recovery problems.
- As the numerical results suggest, even when the recovery problem is ill-conditioned, the reconstruction can be quite robust with respect to noise.

1.2. Related work

There has been a long list of works devoted to the sparse recovery problems mentioned above.

Rational approximation has a long history in numerical analysis. Some of the well-known methods are the RKFIT algorithm [5], barycentric interpolation [6], Pade approximation [16], vector fitting [18], and AAA [26].

Spectral function approximation is a key computational task for many-body quantum systems. Well-known methods include Pade approximation [4,34,37], maximum entropy methods [3,21,22,24,31], and stochastic analytic continuation [17,23,32,36]. Several most recent algorithms are [13,14,20,41,42].

Fourier inversion is a vast field with many different problem setups. When X and S are dual discrete grids with $\{s_j\}$ chosen randomly, this is the compressive sensing problem [9,12,15], and there is a vast literature on methods based on the ℓ_1 convex relaxation. When X is an interval and $\{s_j\}$ are equally spaced grid points, this becomes the line spectrum estimation or superresolution problem [11]. Both Prony-type methods [19,29,30,33] and optimization-based approaches [8,10,25] are well-studied for this field.

Laplace inversion is a longstanding computational problem. Most established algorithms [1,38–40] assume the capability of accessing the sample values at any arbitrary locations. For the case of equally-spaced sample locations, Prony-type methods have been proposed in [7,28]. The work in [27,35] further extends the Prony-type methods to the kernels associated with more general first-order and second-order differential operators.

For sparse deconvolution, when $\{s_j\}$ forms a uniform grid, it is closely related to the superresolution problem. However, when $\{s_j\}$ are unstructured, the literature is surprisingly limited.

The rest of the note is organized as follows. Section 2 reviews Prony's method and the ESPRIT algorithms for the special case of the exponential kernel with the uniform sampling grid. Section 3 describes the eigenmatrix approach for the general kernels and unstructured grids. Section 4 presents the numerical experiments of the applications mentioned above. Section 5 concludes with a discussion for future work.

2. Prony and ESPRIT

To motivate the eigenmatrix construction, we first briefly review Prony's method and the ESPRIT algorithm. Consider the recovery problem with $X = \mathbb{S} \equiv \{z : |z| = 1\} \subset \mathbb{C}$, $S = \mathbb{R}$, $G(s,x) = x^s$ (with the branch cut at $x = -1$), and $s_j = j$ for $j \in \mathbb{Z}$. Here, we make the simplifying assumption that $\{s_j\}$ is the whole integer lattice, though only a finite chunk is required in the actual implementations. Most presentations of Prony's method and the ESPRIT algorithm start with the Hankel matrix. However, our presentation emphasizes the role of the shifting operator in order to motivate the eigenmatrix approach in Section 3.

Introduce the infinitely-long vector-valued function $\mathbf{g}(x) = [G(s_j,x)]_{j\in\mathbb{Z}} = [x^j]_{j\in\mathbb{Z}}$. Let M be the shifting operator that moves each entry up by one slot, i.e., $(M\mathbf{v})_j = \mathbf{v}_{j+1}$ for any vector $\mathbf{v}$. Then

$$M\mathbf{g}(x) = x\mathbf{g}(x).$$

Define the vector $\mathbf{u} = [u_j]_{j\in\mathbb{Z}}$ of the exact observations $u_j = u(s_j) = \sum_k G(s_j, x_k)w_k = \sum_k w_k x_k^j$. Since $\mathbf{u} = \sum_k \mathbf{g}(x_k)w_k$, we have for any $t \geq 0$

$$M^t\mathbf{u} = \sum_k \mathbf{g}(x_k)w_k x_k^t.$$

For the Prony's method, consider the matrix

$$\begin{bmatrix}\mathbf{u} & M\mathbf{u} & \dots & M^{n_x}\mathbf{u}\end{bmatrix} = \begin{bmatrix}\mathbf{g}(x_1) & \dots & \mathbf{g}(x_{n_x})\end{bmatrix}\begin{bmatrix} w_1 & & \\ & \ddots & \\ & & w_{n_x}\end{bmatrix}\begin{bmatrix}1 & x_1 & \dots & x_1^{n_x}\\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n_x} & \dots & x_{n_x}^{n_x}\end{bmatrix}$$

Let $\mathbf{p}$ be a non-zero vector in its null space, i.e.,

$$\begin{bmatrix}\mathbf{u} & M\mathbf{u} & \dots & M^{n_x}\mathbf{u}\end{bmatrix}\mathbf{p} = 0 \quad \text{with} \quad \mathbf{p} = \begin{bmatrix} p_0 \\ \vdots \\ p_{n_x}\end{bmatrix}. \tag{1}$$

Therefore,

$$\begin{bmatrix}1 & x_1 & \dots & x_1^{n_x}\\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n_x} & \dots & x_{n_x}^{n_x}\end{bmatrix}\begin{bmatrix} p_0 \\ \vdots \\ p_{n_x}\end{bmatrix} = 0.$$

This implies that $\{x_k\}$ are the roots of $p(x) = p_0 + p_1 x + \dots p_{n_x} x^{n_x}$. Therefore, one can identify $\{x_k\}$ via rootfinding once $\mathbf{p}$ is computed.

For the ESPRIT algorithm, consider the matrix

$$\begin{bmatrix}\mathbf{u} & M\mathbf{u} & \dots & M^{\ell}\mathbf{u}\end{bmatrix} = \begin{bmatrix}\mathbf{g}(x_1) & \dots & \mathbf{g}(x_{n_x})\end{bmatrix}\begin{bmatrix} w_1 & & \\ & \ddots & \\ & & w_{n_x}\end{bmatrix}\begin{bmatrix}1 & x_1 & \dots & x_1^{\ell}\\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n_x} & \dots & x_{n_x}^{\ell}\end{bmatrix}$$

with $\ell > n_x$. Let USV^* be the rank-n_x singular value decomposition (SVD) of this matrix. The matrix V^* takes the form

$$V^* = P\begin{bmatrix}1 & x_1 & \dots & x_1^{\ell}\\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n_x} & \dots & x_{n_x}^{\ell}\end{bmatrix}$$

where P is an unknown non-degenerate $n_x \times n_x$ matrix. Let Z_0 and Z_1 be the submatrices of V^* by excluding the first column and the last column, respectively, i.e.,

$$Z_0 = P\begin{bmatrix}1 & \dots & x_1^{\ell-1}\\ \vdots & \ddots & \vdots \\ 1 & \dots & x_{n_x}^{\ell-1}\end{bmatrix}, \quad Z_1 = P\begin{bmatrix}x_1 & \dots & x_1^{\ell}\\ \vdots & \ddots & \vdots \\ x_{n_x} & \dots & x_{n_x}^{\ell}\end{bmatrix}.$$

By introducing

$$Z_1(Z_0)^+ = P\begin{bmatrix} x_1 & & \\ & \ddots & \\ & & x_{n_x}\end{bmatrix}P^{-1},$$

one can identify $\{x_k\}$ by computing the eigenvalues of $Z_1(Z_0)^+$.

For both methods, given $\{x_k\}$, the sample weights $\{w_k\}$ can be computed via, for example, the least square solve.

Remark 1. For most problems, the sample values $\{\tilde{u}_j\}$ have noise. As a result, the sample locations and weights obtained above are only approximations. Many implementations of the Prony and ESPRIT methods have a postprocessing step, where these approximations are used as the initial guesses of the following optimization problem

$$\min_{\tilde{x}_k, \tilde{w}_k} \sum_j \left|\sum_k \tilde{x}_k^j \tilde{w}_k - \tilde{u}_j\right|^2. \tag{2}$$

Remark 2. For an actual problem, the number of spikes n_x is not known a priori. An important question is how to pick the right degree of the polynomial $p(x)$ (for Prony's method) or the rank of the truncated SVD (for the ESPRIT algorithm). The general criteria are that the objective value of (2) (after postprocessing) should be within the noise level, and the degree d should be as small as possible. Commonly used criteria include AIC [2] and BIC [33].

3. Eigenmatrix

3.1. Main idea

The discussion above uses two special features of the problem: (a) the kernel is of the exponential form, and (b) $\{s_j\}$ forms an equally spaced grid. These two features together allow one to write down the shifting operator M explicitly. However, these two features no longer hold for sparse recovery problems with a general kernel $G(\cdot,\cdot)$ or unstructured sample locations $\{s_j\}$.

To address this challenge, we take a data-driven approach. Let $\mathbf{g}(x)$ now be the n_s-dimensional vector $[G(s_j,x)]_{1\le j\le n_s}$. The main idea is to introduce an *eigenmatrix* M of size $n_s \times n_s$ such that for all $x \in X$

$$M\mathbf{g}(x) \approx x\mathbf{g}(x).$$

The reason why M is called the eigenmatrix is because *it is designed to have the desired approximate eigenvalues and eigenvectors*.

Below, we detail how to apply the eigenmatrix idea to complex and real cases. Since the choice of degree (in Prony's algorithm) and the rank of truncated SVD (in the ESPRIT algorithm) remain unchanged, we present the algorithm for fixed n_x in order to simplify the discussion.

3.2. Complex analytic case

To simplify the discussion, assume first that X is the unit disc $\mathbb{D}$, and we will comment on the general case at the end. Define for each x the vector $\mathbf{g}(x) := [G(s_j,x)]_{1\le j\le n_s}$. The first step is to construct M such that $M\mathbf{g}(x) \approx x\mathbf{g}(x)$ for $x \in \mathbb{D}$. Numerically, it is more robust to use the normalized vector $\hat{\mathbf{g}}(x) = \mathbf{g}(x)/\|\mathbf{g}(x)\|$ since the norm of $\mathbf{g}(x)$ can vary significantly depending on x. The condition then becomes

$$M\hat{\mathbf{g}}(x) \approx x\hat{\mathbf{g}}(x), \quad x \in \mathbb{D}.$$

We enforce this condition on a uniform grid $\{a_t\}_{1\le t\le n_a}$ of size n_a on the boundary of the unit disk

$$M\hat{\mathbf{g}}(a_t) \approx a_t\hat{\mathbf{g}}(a_t).$$

Define an $n_s \times n_a$ matrix $\hat{G} = [\hat{\mathbf{g}}(a_t)]_{1\le t\le n_a}$ with $\hat{\mathbf{g}}(a_t)$ as columns and also an $n_a \times n_a$ diagonal matrix $\Lambda = \mathrm{diag}(a_t)$. The above condition can be written in a matrix form as

$$M\hat{G} \approx \hat{G}\Lambda. \tag{3}$$

Remark 3. The main guideline for the choice of n_a is that the columns of $\hat{G}$ are numerically linearly independent. To see why this is essential, let us consider the extreme case of a kernel $G(s,x)$ constant in x. Here, the columns are linearly dependent, and there is no way to recover individual $\{x_k\}$. In practice, different sampling locations $\{s_j\}$ lead to different choices of n_a. In practice, n_a is chosen such that the condition number of $\hat{G}$ is bounded below 10^7.

When the columns of $\hat{G}$ are numerically linearly independent, (3) suggests the following choice of the eigenmatrix

$$M := \hat{G}\Lambda\hat{G}^{\dagger}, \tag{4}$$

where the pseudoinverse $\hat{G}^{\dagger}$ is computed by thresholding the singular values of $\hat{G}$. Since M is to be applied repetitively as in (1), the threshold is set so that the norm of M is bounded by a small constant such as 3.

Remark 4. A key question is why enforcing the condition on the uniform grid $\{a_t\}$ is enough. The following calculation shows why. $M\mathbf{g}(a_t) \approx a_t\mathbf{g}(a_t)$ implies $M\mathbf{g}(x) \approx x\mathbf{g}(x)$ for all $x \in \mathbb{D}$:

$$\begin{aligned} M\mathbf{g}(x) &= \frac{1}{2\pi i}\int_{\partial\mathbb{D}} \frac{M\mathbf{g}(z)}{z-x}dz \approx \frac{1}{2\pi i}\sum_t \frac{M\mathbf{g}(a_t)}{a_t-x}(ia_t\frac{2\pi}{n_a}) \\ &\approx \frac{1}{2\pi i}\sum_t \frac{a_t\mathbf{g}(a_t)}{a_t-x}(ia_t\frac{2\pi}{n_a}) \approx \frac{1}{2\pi i}\int_{\partial\mathbb{D}} \frac{z\mathbf{g}(z)}{z-x}dz = x\mathbf{g}(x), \end{aligned}$$

where the first and third approximations use the exponential convergence of the trapezoidal rule for analytic functions $\mathbf{g}(x)$ and $x\mathbf{g}(x)$, and the second approximation directly comes from $M\mathbf{g}(a_t) \approx a_t\mathbf{g}(a_t)$. The equalities are applications of the Cauchy integral theorem.

Remark 5. The second question is, what if X is not $\mathbb{D}$? For a general connected domain X with smooth boundary, let $\phi(t) : \mathbb{D} \to X$ be the one-to-one map between $\mathbb{D}$ and X from the Riemann mapping theorem. We then consider the new kernel $G(s,t) = G(s,\phi(t))$ between S and $\mathbb{D}$ and use the above algorithm to recover the locations $\{\tilde{t}_k\}$ in $\mathbb{D}$. Once $\{\tilde{t}_k\}$ are available, we set $\tilde{x}_k = \phi(\tilde{t}_k)$.

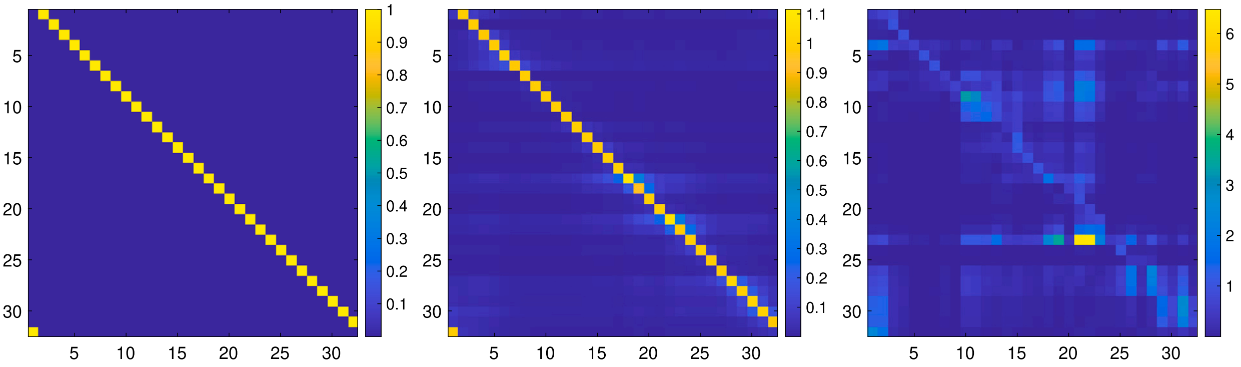

Fig. 1. The eigenmatrix for the example in Section 2. $X = \mathbb{S} \equiv \{z : |z| = 1\} \subset \mathbb{C}$, $S = \mathbb{R}$, $G(s,x) = x^s$ (with the branch cut at $x = -1$). $n_s = 32$. Left: M when $s_j = j$ for $0 \le j < n_s$. Middle: M when s_j is a random perturbation of the integer lattice. Right: M when $\{s_j\}$ are chosen uniformly in $[0, n_s]$. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

To provide an idea about the eigenmatrix, we apply the above construction to the problem given in Section 2 with $n_s = 32$. Fig. 1 plots the eigenmatrix for three cases: (a) $\{s_j = j\}$ is a integer lattice (the classical case); (b) $\{s_j\}$ is a random perturbation of the integer lattice; (c) $\{s_j\}$ are chosen uniformly in $[0, n_s]$. In the integer lattice case, the eigenmatrix approach reproduces the shifting matrix. In the perturbative case, the eigenmatrix is quite close to the shifting matrix, as expected. In the fully random case, the eigenmatrix is quite different but still keeps the overall trend.

3.3. Real analytic case

To simplify the discussion, assume that X is the interval $[-1,1]$, and we will comment on the general case later. Let us define for each x the vector $\mathbf{g}(x) = [G(s_j, x)]_{1\le j\le n_s}$. The first step is to construct M such that $M\mathbf{g}(x) \approx x\mathbf{g}(x)$ for $x \in [-1,1]$. Numerically, it is again more robust to use the normalized vector $\hat{\mathbf{g}}(x) = \mathbf{g}(x)/\|\mathbf{g}(x)\|$ and consider the modified condition

$$M\hat{\mathbf{g}}(x) \approx x\hat{\mathbf{g}}(x), \quad x \in [-1,1].$$

We enforce this condition on a Chebyshev grid $\{a_t\}_{1\le t\le n_a}$ of size n_a on the interval $[-1,1]$:

$$M\hat{\mathbf{g}}(a_t) \approx a_t\hat{\mathbf{g}}(a_t).$$

Introduce the $n_s \times n_s$ matrix $\hat{G} = [\hat{\mathbf{g}}(a_t)]_{1\le t\le n_a}$ with columns $\hat{\mathbf{g}}(a_t)$ as well as the $n_a \times n_a$ diagonal matrix $\Lambda = \mathrm{diag}(a_t)$. The condition now reads

$$M\hat{G} \approx \hat{G}\Lambda.$$

When the columns of $\hat{G}$ are numerically linearly independent, this again suggests the following choice of the eigenmatrix for the real analytic case

$$M := \hat{G}\Lambda\hat{G}^\dagger,$$

where the pseudoinverse $\hat{G}^\dagger$ is computed by thresholding the singular values of $\hat{G}$.

Remark 6. We claim that, for real analytic kernels $G(s,x)$, enforcing the condition at the Chebyshev grid $\{a_t\}$ is sufficient. To see this,

$$M\mathbf{g}(x) \approx M\left(\sum_k c_t(x)\mathbf{g}(a_t)\right) = \sum_t c_t(x)M\mathbf{g}(a_t) \approx \sum_t c_t(x)(a_t\mathbf{g}(a_t)) \approx x\mathbf{g}(x),$$

where $c_t(x)$ is the Chebyshev quadrature for x associated with grid $\{a_t\}$. Here, the first and third approximations use the convergence property of the Chebyshev quadrature for analytic functions $\mathbf{g}(x)$ and $x\mathbf{g}(x)$, and the second approximation directly comes from $M\mathbf{g}(a_t) \approx a_t\mathbf{g}(a_t)$.

Remark 7. The next question is, what if X is not the interval $[-1,1]$? For a general interval or analytic segment X, let $\phi : [-1,1] \to X$ be a smooth one-to-one map between $[-1,1]$ and X. By considering the kernel $G(s,t) = G(s,\phi(t))$ instead and applying the above algorithm, we can recover the locations $\{\tilde{t}_k\}$ in $[-1,1]$. Finally, set $\tilde{x}_k = \phi(\tilde{t}_k)$.

3.4. Putting together

With the eigenmatrix M available, the rest is similar to Prony's method and the ESPRIT algorithm. Define the vector $\tilde{\mathbf{u}} := [\tilde{u}_j]_{1\le j\le n_s}$ from the noisy sample values.

For Prony's method, consider

$$\begin{bmatrix}\tilde{\mathbf{u}} & M\tilde{\mathbf{u}} & \dots & M^{n_x}\tilde{\mathbf{u}}\end{bmatrix} \approx \begin{bmatrix}\mathbf{g}(x_1) & \dots & \mathbf{g}(x_{n_x})\end{bmatrix}\begin{bmatrix}w_1 & & \\ & \ddots & \\ & & w_{n_x}\end{bmatrix}\begin{bmatrix}1 & x_1 & \dots & x_1^{n_x}\\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n_x} & \dots & x_{n_x}^{n_x}\end{bmatrix}.$$

Let $\tilde{\mathbf{p}}$ be a non-zero vector in its null-space

$$\begin{bmatrix}\tilde{\mathbf{u}} & M\tilde{\mathbf{u}} & \dots & M^{n_x}\tilde{\mathbf{u}}\end{bmatrix}\tilde{\mathbf{p}} = 0 \quad \text{with} \quad \tilde{\mathbf{p}} = \begin{bmatrix}\tilde{p}_0\\ \vdots \\ \tilde{p}_{n_x}\end{bmatrix}$$

Therefore,

$$\begin{bmatrix}1 & x_1 & \dots & x_1^{n_x}\\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n_x} & \dots & x_{n_x}^{n_x}\end{bmatrix}\begin{bmatrix}\tilde{p}_0\\ \vdots \\ \tilde{p}_{n_x}\end{bmatrix} \approx 0.$$

The roots of the polynomial $\tilde{p}(x) := \tilde{p}_0 + \tilde{p}_1 x + \dots + \tilde{p}_{n_x} x^{n_x}$ provide the estimators $\{\tilde{x}_k\}$ for $\{x_k\}$.

For the ESPRIT algorithm, consider the matrix

$$\begin{bmatrix}\tilde{\mathbf{u}} & M\tilde{\mathbf{u}} & \dots & M^{\ell}\tilde{\mathbf{u}}\end{bmatrix} \approx \begin{bmatrix}\mathbf{g}(x_1) & \dots & \mathbf{g}(x_{n_x})\end{bmatrix}\begin{bmatrix}w_1 & & \\ & \ddots & \\ & & w_{n_x}\end{bmatrix}\begin{bmatrix}1 & x_1 & \dots & x_1^{\ell}\\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n_x} & \dots & x_{n_x}^{\ell}\end{bmatrix}$$

with $\ell > n_x$. Let $\tilde{U}\tilde{S}\tilde{V}^*$ be the rank-n_x truncated SVD of this matrix. The matrix $\tilde{V}^*$ satisfies

$$\tilde{V}^* \approx P\begin{bmatrix}1 & x_1 & \dots & x_1^{\ell}\\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n_x} & \dots & x_{n_x}^{\ell}\end{bmatrix}$$

where P is an unknown non-degenerate $n_x \times n_x$ matrix. Let $\tilde{Z}_0$ and $\tilde{Z}_1$ be the submatrices of $\tilde{Z}$ by excluding the first column and the last column, respectively, i.e.,

$$\tilde{Z}_0 \approx P\begin{bmatrix}1 & \dots & x_1^{\ell-1}\\ \vdots & \ddots & \vdots \\ 1 & \dots & x_{n_x}^{\ell-1}\end{bmatrix}, \quad \tilde{Z}_1 \approx P\begin{bmatrix}x_1 & \dots & x_1^{\ell}\\ \vdots & \ddots & \vdots \\ x_{n_x} & \dots & x_{n_x}^{\ell}\end{bmatrix}.$$

By introducing

$$\tilde{Z}_1(\tilde{Z}_0)^+ \approx P\begin{bmatrix}x_1 & & \\ & \ddots & \\ & & x_{n_x}\end{bmatrix}P^{-1},$$

one can get estimates $\{\tilde{x}_k\}$ for $\{x_k\}$ by computing the eigenvalues of $\tilde{Z}_1(\tilde{Z}_0)^+$.

With $\{\tilde{x}_k\}$ available, the least square solve

$$\min_{\tilde{w}_k} \sum_j \left|\sum_k G(s_j, \tilde{x}_k)\tilde{w}_k - \tilde{u}_j\right|^2$$

gives the estimators $\{\tilde{w}_k\}$ for $\{w_k\}$ for both methods.

4. Numerical results

This section applies the eigenmatrix approach to the unstructured sparse recovery problems mentioned in Section 1. In all examples, the spike weights $\{w_k\}$ are set to be 1 and the noises $\{Z_j\}$ are Gaussian.

Once the eigenmatrix M is constructed, the reported numerical results are obtained using ESPRIT. The results of Prony's method are similar but slightly less robust. In each plot, blue, green, and red stand for the exact solution, the result before postprocessing, and the one after postprocessing.

Example 1 *(Rational approximation)*. The problem setup is

- $G(s, x) = \frac{1}{s-x}$.

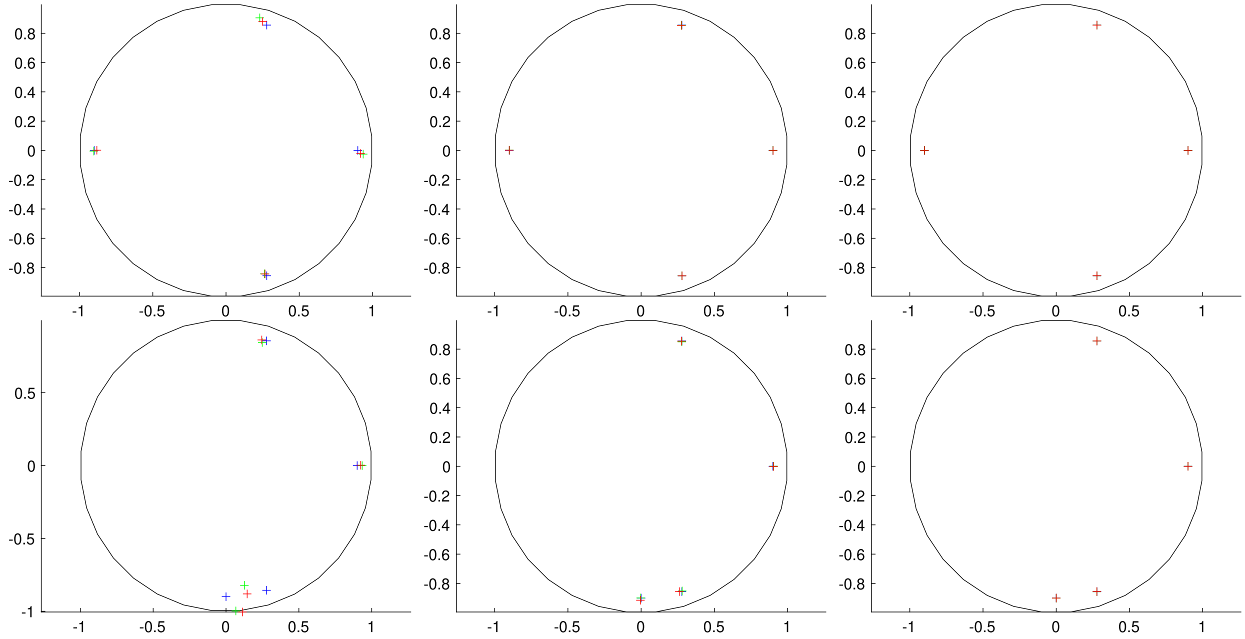

Fig. 2. Rational approximation. $G(s,x)=\frac{1}{s-x}$. $X=\mathbb{D}$. $\{s_j\}$ are random points outside the unit disk, each with a modulus between 1.2 and 2.2. $n_s=40$. Columns: σ equals to 10^{-2}, 10^{-3}, and 10^{-4}. Rows: the easy test (well-separated) and the hard test (with two nearby spikes).

- $X=\mathbb{D}$.
- $\{s_j\}$ are random points outside the disk, each with a modulus between 1.2 and 2.2. $n_s=40$.

Fig. 2 summarizes the experimental results. $n_a=32$. The three columns correspond to noise levels σ equal to 10^{-2}, 10^{-3}, and 10^{-4}. Two tests are performed. In the first one (top row), $\{x_k\}$ are well-separated from each other. The plots show accurate recovery of the spike locations from all σ values. In the second one (bottom row), two of the spike locations are close to each other: the reconstruction at $\sigma=10^{-2}$ shows a noticeable error, while the results for $\sigma=10^{-3}$ and $\sigma=10^{-4}$ are accurate. The plots also suggest that the eigenmatrix approach results in fairly accurate (green) initial guesses for the postprocessing step.

Example 2 *(Spectral function approximation).* The problem setup is

- $G(s,x)=\frac{1}{s-x}$.
- $X=[-1,1]$.
- $\{s_j\}$ is the Matsubara grid from $-\frac{(2N-1)\pi}{\beta}i$ to $\frac{(2N-1)\pi}{\beta}i$ with $\beta=100$ and $N=128$. Hence, $n_s=256$.

Fig. 3 summarizes the experimental results. $n_a=32$. The three columns correspond to σ equal to 10^{-2}, 10^{-3}, and 10^{-4}, respectively. Two tests are performed. In the first one (top row), $\{x_k\}$ are well-separated. The reconstructions are accurate for all σ values. In the second one (bottom row), two of the spike locations are within 0.1 distance from each other. In this harder case, the reconstructions also remain accurate for all σ values. Notice that the eigenmatrix provides a sufficiently accurate initial guess for postprocessing.

Example 3 *(Fourier inversion).* The problem setup is

- $G(s,x)=\exp(\pi isx)$.
- $X=[-1,1]$.
- $\{s_j\}$ are randomly chosen points in $[-5,5]$. $n_s=128$.

Fig. 4 summarizes the experimental results. $n_a=32$. The three columns correspond to σ equal to 10^{-2}, 10^{-3}, and 10^{-4}. Two tests are performed. In the first one (top row), $\{x_k\}$ are well-separated. The reconstructions are accurate for all σ values. In the second one (bottom row), two of the spike locations are within 0.1 distance from each other. The reconstructions are also accurate for all σ values. The eigenmatrix is again able to provide sufficient accurate initial guesses for the postprocessing step.

Example 4 *(Laplace inversion).* The problem setup is

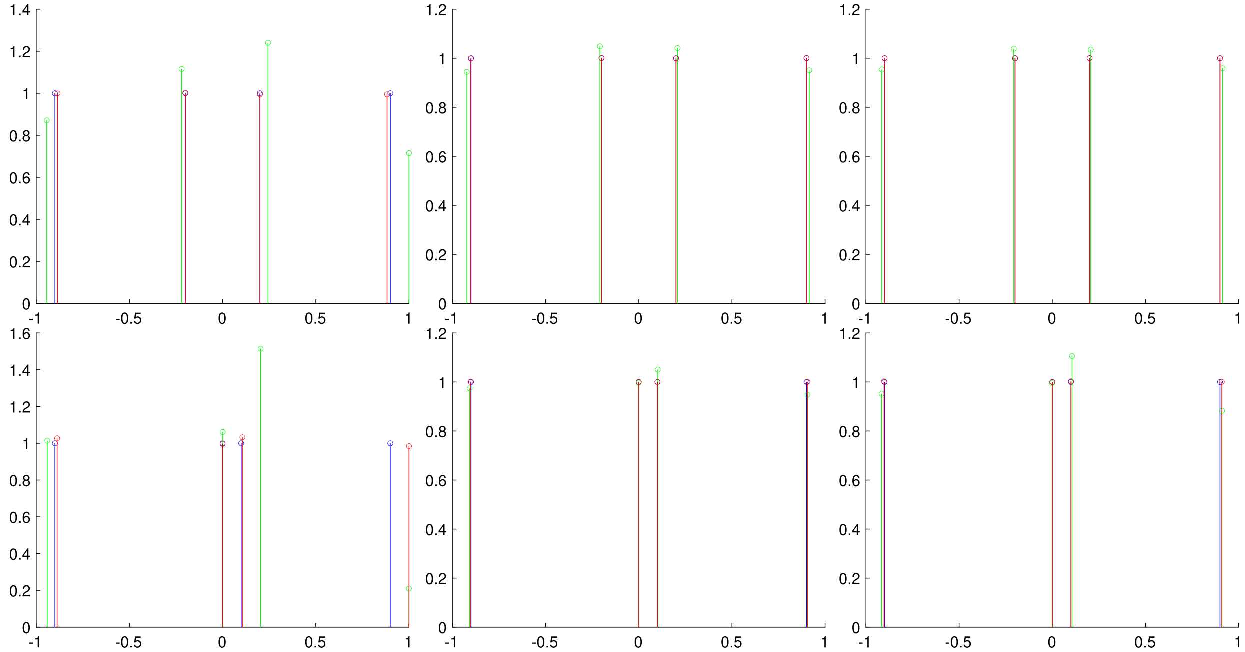

Fig. 3. Spectral function approximation. $G(s,x) = \frac{1}{s-x}$. $X = [-1,1]$. $\{s_j\}$ is the Matsubara grid from $-\frac{(2N-1)\pi}{\beta}i$ to $\frac{(2N-1)\pi}{\beta}i$ with $\beta = 100$ and $N = 128$. Columns: σ equals to 10^{-2}, 10^{-3}, and 10^{-4}. Rows: the easy test (well-separated) and the hard test (with two nearby spikes).

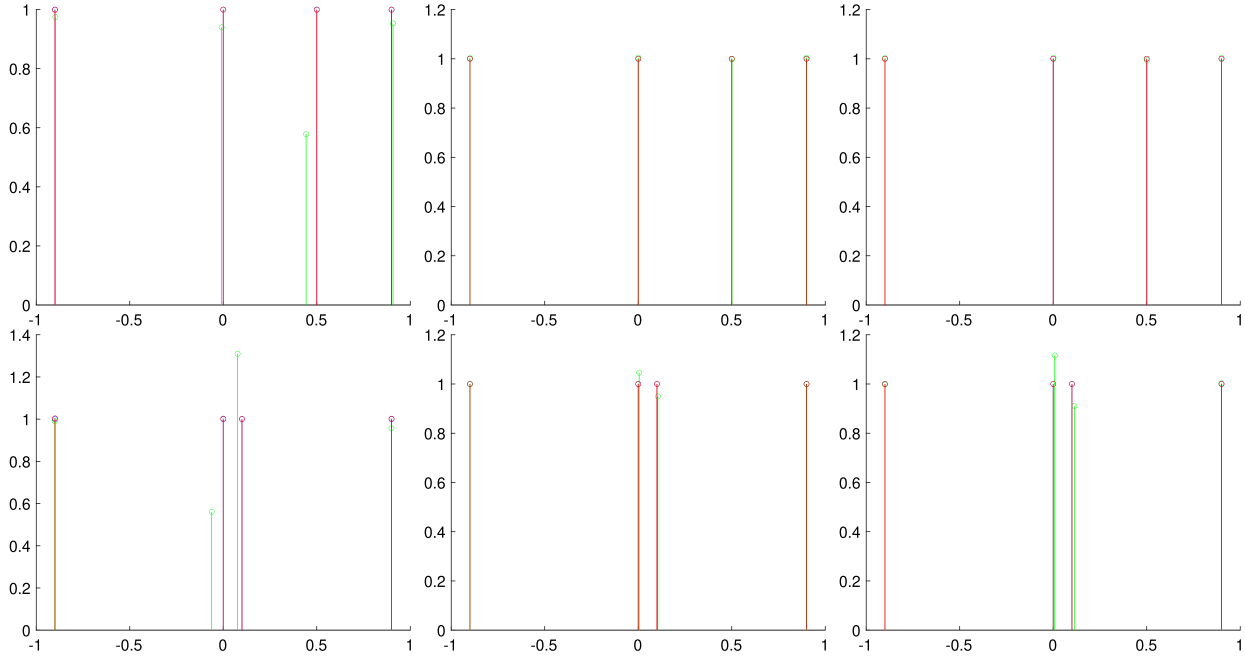

Fig. 4. Fourier inversion. $G(s,x) = \exp(\pi isx)$. $X = [-1,1]$. $\{s_j\}$ are randomly chosen points in $[-5,5]$. $n_s = 128$. Columns: σ equals to 10^{-2}, 10^{-3}, and 10^{-4}, respectively. Rows: the easy test (well-separated) and the hard test (with two nearby spikes).

- $G(s,x) = x\exp(-sx)$.
- $X = [0.1, 2.1]$.
- $\{s_j\}$ are random samples in $[0,10]$. $n_s = 100$.

Fig. 5 summarizes the experimental results. $n_a = 32$. The inverse Laplace transform is well-known for its sensitivity to noise. As a result, significantly smaller noise magnitudes are used in this example: the three columns correspond to σ equal to 10^{-5}, 10^{-6}, and

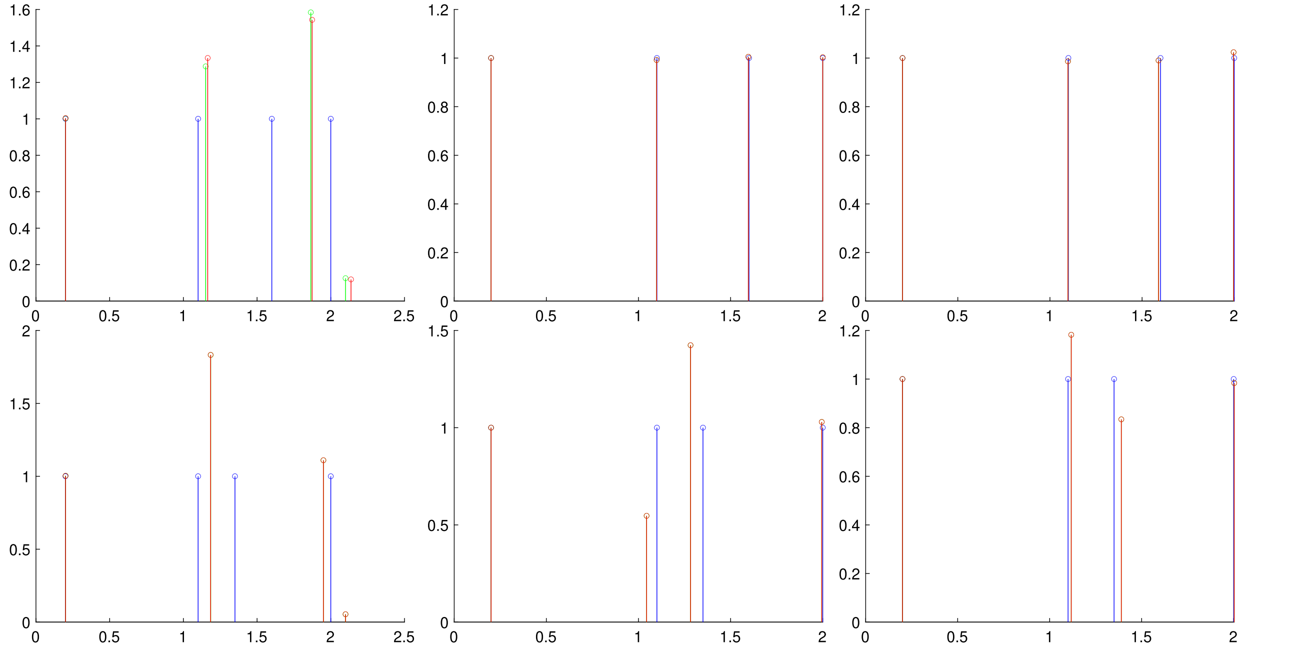

Fig. 5. Laplace inversion. $G(s,x) = \exp(-sx)$. $X = [0.1, 2.1]$. $\{s_j\}$ are random samples in $[0,10]$. $n_s = 100$. Columns: σ equals to 10^{-5}, 10^{-6}, and 10^{-7}, respectively. Rows: the easy test (well-separated) and the hard test (with two nearby spikes).

10^{-7}. Two tests are performed. In the first one (top row), $\{x_k\}$ are well-separated. The reconstructions are acceptable for $\sigma = 10^{-6}$ and accurate for $\sigma = 10^{-7}$. In the second harder test (bottom row), two of the spike locations are within 0.25 distance from each other. The reconstructions provide reasonable reconstructions at $\sigma = 10^{-7}$, but significant errors for larger σ values.

Example 5 *(Sparse deconvolution).* The problem setup is

- $G(s,x) = \frac{1}{1+\gamma(s-x)^2}$ with $\gamma = 4$.
- $X = [-1, 1]$.
- $\{s_j\}$ are random samples from $[-5, 5]$. $n_s = 100$.

Fig. 6 summarizes the experimental results. $n_a = 32$. The three columns correspond to σ equal to 10^{-2}, 10^{-3}, and 10^{-4}. Two tests are performed. In the first one (top row), $\{x_k\}$ are well-separated. The reconstructions are reasonable for $\sigma = 10^{-2}$ and accurate for the smaller σ values. In the second one (bottom row), two of the spike locations are within 0.1 distance from each other. The reconstructions are accurate for σ equal to 10^{-3} and 10^{-4}.

Remark 8. The numerical experience suggests two lessons important for accurate reconstruction. First, it is important to fully exploit the prior information about the support of the spikes, i.e., making the candidate parameter set X as compact as possible. Second, using the Chebyshev grid (for the real case) and the uniform grid (for the complex case) ensures that $M\mathbf{g}(x) \approx x\mathbf{g}(x)$ up to a high accuracy numerically.

5. Discussions

This note introduces the eigenmatrix construction for unstructured sparse recovery problems. It assumes no structure on the sample locations and offers a rather unified framework for such sparse recovery problems. This note is only an exploratory study of the data-driven approach for unstructured sparse recovery, and there are several clear directions for future work.

- Providing a more principled way of choosing the size of the grid $\{a_t\}$ and the thresholding value for computing M.
- Providing the error estimates of the eigenmatrix approach for the problems mentioned in Section 1.
- Once the eigenmatrix is constructed, the recovery algorithm presented above follows Prony's method and the ESPRIT algorithm. An immediate extension is to combine the eigenmatrix with other algorithms, such as MUSIC and the matrix pencil method.

Data availability

No data was used for the research described in the article.

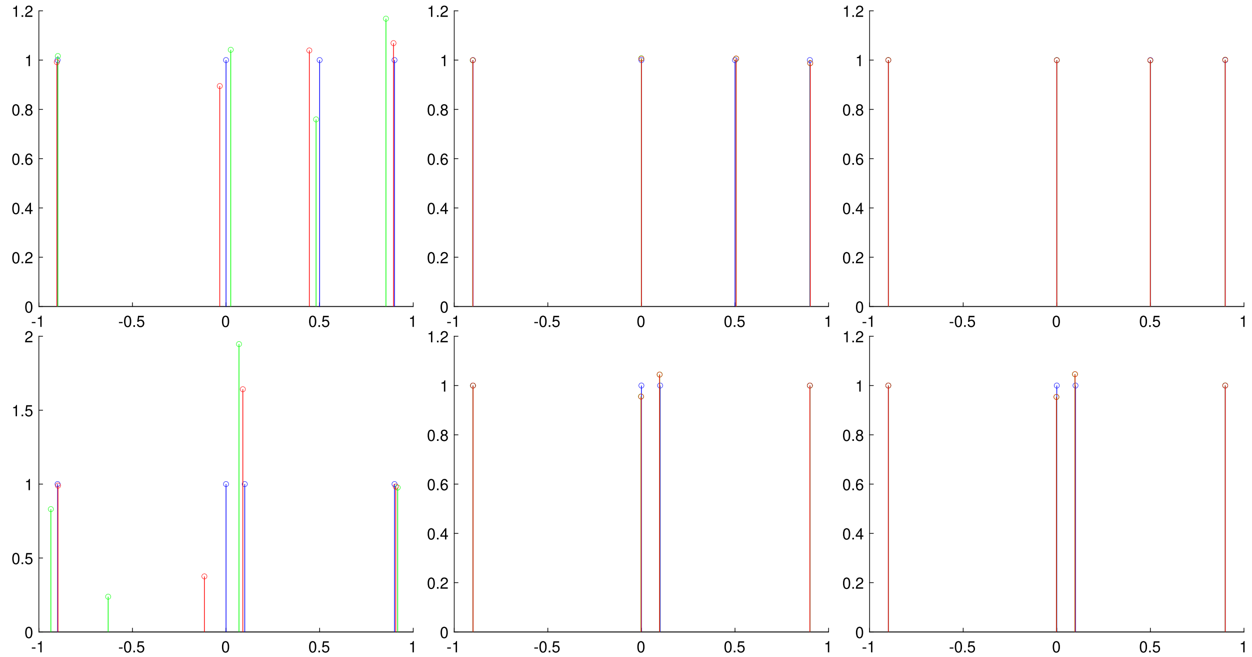

Fig. 6. Sparse deconvolution. $G(s,x)=\frac{1}{1+\gamma(s-x)^2}$ with $\gamma=4$. $X=[-1,1]$. $\{s_j\}$ are random samples from $[-5,5]$. $n_s=100$. Columns: σ equals to 10^{-2}, 10^{-3}, and 10^{-4}, respectively. Rows: the easy test (well-separated) and the hard test (with two nearby spikes).

References

[1] Joseph Abate, Peter P. Valkó, Multi-precision Laplace transform inversion, Int. J. Numer. Methods Eng. 60 (5) (2004) 979–993.

[2] Hirotogu Akaike, Information theory and an extension of the maximum likelihood principle, in: Selected Papers of Hirotugu Akaike, 1998, pp. 199–213.

[3] K.S.D. Beach, Identifying the maximum entropy method as a special limit of stochastic analytic continuation, arXiv preprint arXiv:cond-mat/0403055, 2004.

[4] K.S.D. Beach, R.J. Gooding, F. Marsiglio, Reliable Padé analytical continuation method based on a high-accuracy symbolic computation algorithm, Phys. Rev. B 61 (8) (2000) 5147.

[5] Mario Berljafa, Stefan Guttel, The rkfit algorithm for nonlinear rational approximation, SIAM J. Sci. Comput. 39 (5) (2017) A2049–A2071.

[6] Jean-Paul Berrut, Lloyd N. Trefethen, Barycentric Lagrange interpolation, SIAM Rev. 46 (3) (2004) 501–517.

[7] Gregory Beylkin, Lucas Monzón, Nonlinear inversion of a band-limited Fourier transform, Appl. Comput. Harmon. Anal. 27 (3) (2009) 351–366.

[8] Emmanuel J. Candès, Carlos Fernandez-Granda, Towards a mathematical theory of super-resolution, Commun. Pure Appl. Math. 67 (6) (2014) 906–956.

[9] Emmanuel J. Candes, Justin K. Romberg, Terence Tao, Stable signal recovery from incomplete and inaccurate measurements, Commun. Pure Appl. Math. 59 (8) (2006) 1207–1223.

[10] Laurent Demanet, Nam Nguyen, The recoverability limit for superresolution via sparsity, arXiv preprint arXiv:1502.01385, 2015.

[11] David L. Donoho, Superresolution via sparsity constraints, SIAM J. Math. Anal. 23 (5) (1992) 1309–1331.

[12] David L. Donoho, For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution, Commun. Pure Appl. Math. 59 (6) (2006) 797–829.

[13] Jiani Fei, Chia-Nan Yeh, Emanuel Gull, Nevanlinna analytical continuation, Phys. Rev. Lett. 126 (5) (2021) 056402.

[14] Jiani Fei, Chia-Nan Yeh, Dominika Zgid, Emanuel Gull, Analytical continuation of matrix-valued functions: Carathéodory formalism, Phys. Rev. B 104 (16) (2021) 165111.

[15] Simon Foucart, Holger Rauhut, An Invitation to Compressive Sensing, Springer, 2013.

[16] Pedro Gonnet, Stefan Guttel, Lloyd N. Trefethen, Robust Padé approximation via SVD, SIAM Rev. 55 (1) (2013) 101–117.

[17] Olga Goulko, Andrey S. Mishchenko, Lode Pollet, Nikolay Prokof'ev, Boris Svistunov, Numerical analytic continuation: answers to well-posed questions, Phys. Rev. B 95 (1) (2017) 014102.

[18] Bjorn Gustavsen, Adam Semlyen, Rational approximation of frequency domain responses by vector fitting, IEEE Trans. Power Deliv. 14 (3) (1999) 1052–1061.

[19] Yingbo Hua, Tapan K. Sarkar, Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise, IEEE Trans. Acoust. Speech Signal Process. 38 (5) (1990) 814–824.

[20] Zhen Huang, Emanuel Gull, Lin Lin, Robust analytic continuation of Green's functions via projection, pole estimation, and semidefinite relaxation, Phys. Rev. B 107 (7) (2023) 075151.

[21] Mark Jarrell, James E. Gubernatis, Bayesian inference and the analytic continuation of imaginary-time quantum Monte Carlo data, Phys. Rep. 269 (3) (1996) 133–195.

[22] Gernot J. Kraberger, Robert Triebl, Manuel Zingl, Markus Aichhorn, Maximum entropy formalism for the analytic continuation of matrix-valued Green's functions, Phys. Rev. B 96 (15) (2017) 155128.

[23] Igor Krivenko, Malte Harland, Triqs/som: implementation of the stochastic optimization method for analytic continuation, Comput. Phys. Commun. 239 (2019) 166–183.

[24] Ryan Levy, J.P.F. LeBlanc, Emanuel Gull, Implementation of the maximum entropy method for analytic continuation, Comput. Phys. Commun. 215 (2017) 149–155.

[25] Ankur Moitra, Super-resolution, extremal functions and the condition number of Vandermonde matrices, in: Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing, 2015, pp. 821–830.

[26] Yuji Nakatsukasa, Olivier Sète, Lloyd N. Trefethen, The aaa algorithm for rational approximation, SIAM J. Sci. Comput. 40 (3) (2018) A1494–A1522.

[27] Thomas Peter, Gerlind Plonka, A generalized Prony method for reconstruction of sparse sums of eigenfunctions of linear operators, Inverse Probl. 29 (2) (2013) 025001.

[28] Daniel Potts, Manfred Tasche, Parameter estimation for nonincreasing exponential sums by Prony-like methods, Linear Algebra Appl. 439 (4) (2013) 1024–1039.

[29] R. Prony, Essai experimental et analytique, J. Éc. Polytech. (1795) 24–76.

[30] Richard Roy, Thomas Kailath, Esprit-estimation of signal parameters via rotational invariance techniques, IEEE Trans. Acoust. Speech Signal Process. 37 (7) (1989) 984–995.

[31] Michael Rumetshofer, Daniel Bauernfeind, Wolfgang von der Linden, Bayesian parametric analytic continuation of Green's functions, Phys. Rev. B 100 (7) (2019) 075137.

[32] Anders W. Sandvik, Stochastic method for analytic continuation of quantum Monte Carlo data, Phys. Rev. B 57 (17) (1998) 10287.

[33] Ralph Schmidt, Multiple emitter location and signal parameter estimation, IEEE Trans. Antennas Propag. 34 (3) (1986) 276–280.

[34] Johan Schött, Inka L.M. Locht, Elin Lundin, Oscar Grånäs, Olle Eriksson, Igor Di Marco, Analytic continuation by averaging Padé approximants, Phys. Rev. B 93 (7) (2016) 075104.

[35] Kilian Stampfer, Gerlind Plonka, The generalized operator based Prony method, Constr. Approx. 52 (2) (2020) 247–282.

[36] K. Vafayi, O. Gunnarsson, Analytical continuation of spectral data from imaginary time axis to real frequency axis using statistical sampling, Phys. Rev. B 76 (3) (2007) 035115.

[37] H.J. Vidberg, J.W. Serene, Solving the Eliashberg equations by means of N-point Padé approximants, J. Low Temp. Phys. 29 (3) (1977) 179–192.

[38] William T. Weeks, Numerical inversion of Laplace transforms using Laguerre functions, J. ACM 13 (3) (1966) 419–429.

[39] J. Weideman, Lloyd Trefethen, Parabolic and hyperbolic contours for computing the Bromwich integral, Math. Comput. 76 (259) (2007) 1341–1356.

[40] Jacob Andre C. Weideman, Algorithms for parameter selection in the weeks method for inverting the Laplace transform, SIAM J. Sci. Comput. 21 (1) (1999) 111–128.

[41] Lexing Ying, Analytic continuation from limited noisy Matsubara data, J. Comput. Phys. 469 (2022) 111549.

[42] Lexing Ying, Pole recovery from noisy data on imaginary axis, J. Sci. Comput. 92 (3) (2022) 107.