

Full Recovery from Point Values: an Optimal Algorithm for Chebyshev Approximability Prior

Simon Foucart*— Texas A&M University

Abstract

Given pointwise samples of an unknown function belonging to a certain model set, one seeks in Optimal Recovery to recover this function in a way that minimizes the worst-case error of the recovery procedure. While it is often known that such an optimal recovery procedure can be chosen to be linear, e.g. when the model set is based on approximability by a subspace of continuous functions, a construction of the procedure is rarely available. This note uncovers a practical algorithm to construct a linear optimal recovery map when the approximation space is a Chebyshev space of univariate functions that has dimension at least three and contains the constants.

Key words and phrases: Optimal recovery, Chebyshev spaces, ℓ_1 -minimization, simplex algorithm.

AMS classification: 41A05, 41A10, 41A50, 90C05.

1 Problem Setting

Throughout this note, one works in the space $C(\mathcal{X})$ of continuous functions on a compact set \mathcal{X} equipped with the uniform norm defined for $f \in C(\mathcal{X})$ by $\|f\|_{C(\mathcal{X})} = \max\{|f(x)|, x \in \mathcal{X}\}$. Given points $x^{(1)}, \dots, x^{(m)} \in \mathcal{X}$, an unknown function $f \in C(\mathcal{X})$ is observed via the point values

$$y_i = f(x^{(i)}), \quad i \in \{1, \dots, m\}.$$

This so-called *a posteriori* information alone is not enough to approximate/learn/recover f in any meaningful way. One also needs some *a priori* information, usually expressed by the membership of f to some model set \mathcal{K} , i.e., by $f \in \mathcal{K}$. The performance of a recovery procedure—which is nothing else than a map Δ from \mathbb{R}^m into $C(\mathcal{X})$ —can then be assessed via its worst-case error over \mathcal{K} , defined as

$$(1) \quad \text{wce}_{\mathcal{K}}(\Delta) := \sup_{f \in \mathcal{K}} \|f - \Delta([f(x^{(1)}); \dots; f(x^{(m)})])\|_{C(\mathcal{X})}.$$

*S. F. partially supported by grants from the NSF (DMS-2053172) and from the ONR (N00014-20-1-2787).
foucart@tamu.edu

The question being addressed in this note reads as follows:

(Q) Can one construct an optimal recovery procedure Δ , i.e., one that minimizes $\text{wce}_{\mathcal{K}}(\Delta)$?

This objective is too ambitious for a general model set \mathcal{K} , so one concentrates in particular on model sets based on approximation capabilities. Precisely, given a subset \mathcal{V} of $C(\mathcal{X})$ and a parameter $\varepsilon \geq 0$, one considers the approximability model

$$\mathcal{K}_{\mathcal{V},\varepsilon} = \{f \in C(\mathcal{X}) : \text{dist}_{C(\mathcal{X})}(f, \mathcal{V}) \leq \varepsilon\}.$$

The premise that the observed function f belongs to $\mathcal{K}_{\mathcal{V},\varepsilon}$ can be viewed as making explicit an assumption often appearing implicitly in numerical methods. Indeed, with \mathcal{P}_n denotes the space of polynomials of degree $< n$, one often targets error bounds featuring $\text{dist}_{C(\mathcal{X})}(f, \mathcal{P}_n)$ —hence no error when $f \in \mathcal{P}_n$: this is the exactness principle—and this presupposes that $\text{dist}_{C(\mathcal{X})}(f, \mathcal{P}_n)$ should be small. A typical example is supplied by the design of quadrature formulas, discussed in [16] along the lines of the exactness principle. In the rest of this note, the implicit-made-explicit assumption therefore takes the form of the prior $\text{dist}_{C(\mathcal{X})}(f, \mathcal{V}) \leq \varepsilon$ for some n -dimensional subspace \mathcal{V} of $C(\mathcal{X})$ sharing key similitudes with \mathcal{P}_n , i.e., Chebyshev spaces containing constant functions. Some important properties of Chebyshev spaces are recalled in Section 3. For now, one only mentions that these spaces essentially do not exist in dimension $d > 1$. Arguably, this restricts the impact of the positive answer to Question (Q) given for the case $\mathcal{X} = [-1, 1]$. However, even in this elementary case, a knowledge gap is still filled by the complete answer exposed in this note. For sure, some pieces were known—they are recalled in Section 4—but they did not provide a genuinely constructive recovery procedure. Here, a practical recovery algorithm is indeed provided. The correctness of this algorithm is justified in Section 5. Section 6 concludes with some related remarks, including a recipe to compute the maximum ratio of uniform and discrete norms over Chebyshev spaces.

2 Description of an Optimal Algorithm

Before launching into theoretical considerations, one directly puts forward the proposed procedure as Algorithm 1 below, with points requiring justification indicated by a triangle \triangleright . But first, one quickly elucidates the notation a_S and M_S for a vector $a \in \mathbb{R}^m$ and for a matrix $M \in \mathbb{R}^{n \times m}$ when $S \subseteq \{1, \dots, m\}$ is an index set of size $|S| = n$: they represent the subvector in \mathbb{R}^n , resp. the submatrix in $\mathbb{R}^{n \times n}$, obtained by keeping only entries, resp. columns, indexed by S . Note that $n \leq m$ is assumed, for otherwise there would exist $h \in \mathcal{V} \setminus \{0\}$ such that $h(x^{(1)}) = \dots = h(x^{(1)}) = 0$, which, by considering $f + th$ as $t \rightarrow \infty$, would yield $\text{wce}_{\mathcal{K}_{\mathcal{V},\varepsilon}}(\Delta) = \infty$ for any Δ .

Algorithm 1 Optimal recovery procedure from point values for Chebyshev approximability prior

Require: Points $x^{(1)}, \dots, x^{(m)} \in [-1, 1]$ (completed with $x^{(0)} = -1$ and $x^{(m+1)} = 1$ if necessary) and functions v_1, \dots, v_n forming a basis for the n -dimensional Chebyshev space $\mathcal{V} \subseteq C[-1, 1]$ containing the constant functions

Ensure: $m \geq n \geq 3$

do create the matrix $M \in \mathbb{R}^{n \times m}$ having entries $M_{j,i} = v_j(x^{(i)})$, $j \in \{1, \dots, n\}$, $i \in \{1, \dots, m\}$
for $k \in \{1, \dots, m+1\}$ **do**
 select a point $z^{(k)} \in (x^{(k-1)}, x^{(k)})$
 create the vector $b^{(k)} \in \mathbb{R}^n$ having entries $b_j^{(k)} = v_j(z^{(k)})$, $j \in \{1, \dots, n\}$
 compute an index set $S_k \subseteq \{1, \dots, m\}$ of size $|S_k| = n$ so that the vector $a^{(k)} \in \mathbb{R}^m$ supported on S_k and satisfying $a_{S_k}^{(k)} = M_{S_k}^{-1} b^{(k)}$ is a solution to ▷

$$\underset{a \in \mathbb{R}^m}{\text{minimize}} \|a\|_1 \quad \text{subject to } Ma = b^{(k)}$$

end for

for $i \in \{1, \dots, m\}$ **do**

 define a continuous function a_i^\sharp on $[-1, 1]$ by ▷

$$a_i^\sharp(x) = \begin{cases} \sum_{j=1}^n [M_{S_k}^{-1}]_{i,j} v_j(x) & \text{if } i \in S_k \\ 0 & \text{if } i \notin S_k \end{cases}, \quad x \in (x^{(k-1)}, x^{(k)}), \quad k \in \{1, \dots, m+1\}$$

$$a_i^\sharp(x^{(\ell)}) = \begin{cases} 1 & \text{if } i = \ell \\ 0 & \text{if } i \neq \ell \end{cases}, \quad \ell \in \{1, \dots, m\}$$

end for

return the optimal recovery map $\Delta^\sharp : y \in \mathbb{R}^m \mapsto \sum_{i=1}^m y_i a_i^\sharp \in C[-1, 1]$ ▷

To confirm that Algorithm 1 acts as intended, three points need to be accounted for: the existence and computability of an index set S_k with the required property, the continuity of the functions $a_1^\sharp, \dots, a_m^\sharp$, and the optimality of the recovery map Δ^\sharp . The last two points rely on results about Chebyshev spaces and Optimal Recovery, which are covered in Sections 3 and 4. As for the first point, it can be explained right now. Concerning existence, recall that an optimization program

$$(2) \quad \underset{a \in \mathbb{R}^m}{\text{minimize}} \|a\|_1 \quad \text{subject to } Ma = b$$

always admits an n -sparse solution $\hat{a} \in \mathbb{R}^m$ (this can be proved along the lines of [13, Theorem 12.7] and [9, Theorem 6.1]), say supported on some $S \subseteq \{1, \dots, m\}$ with $|S| = n$. The constraint $M\hat{a} = b$ then reads $M_S \hat{a}_S = b$, i.e., $\hat{a}_S = M_S^{-1} b$. Concerning computability, it can be realized by recasting (2) as a standard-form linear program to be solved via the simplex algorithm. Precisely, introducing a slack variable $c = [a^+; a^-] \in \mathbb{R}^{2m}$ with nonnegative vectors $a^+, a^- \in \mathbb{R}^m$ satisfying $a = a^+ - a^-$

and $|a| = a^+ + a^-$, the ℓ_1 -minimization (2) is equivalent to

$$\underset{c \in \mathbb{R}^{2m}}{\text{minimize}} \sum_{j=1}^{2m} c_j \quad \text{subject to } [M \mid -M]c = b \text{ and } c \geq 0.$$

Solving the latter with the simplex algorithm yields a solution $\hat{c} = [\hat{a}^+; \hat{a}^-]$ which is an extreme point of the feasibility polytope and as such (see e.g. [9, Lemma 20.2]) is n -sparse. In turn, the solution $\hat{a} = \hat{a}^+ - \hat{a}^-$ to (2) is also n -sparse.

3 Reminders on Chebyshev Spaces

Chebyshev subspaces are at the center of Approximation Theory: they are defined as the subspaces from which best approximants always exist and are unique. Remarkably, in $C(\mathcal{X})$, they coincide with the subspaces for which Lagrange interpolation is always possible and unique. Precisely, an n -dimensional subspace \mathcal{V} of $C(\mathcal{X})$ is a Chebyshev space if and only if, for any distinct points $\xi^{(1)}, \dots, \xi^{(n)} \in \mathcal{X}$ and any values $\gamma_1, \dots, \gamma_n \in \mathbb{R}$, there exists a unique function $v \in \mathcal{V}$ such that $v(\xi^{(1)}) = \gamma_1, \dots, v(\xi^{(n)}) = \gamma_n$. By considering the linear map $v \in \mathcal{V} \mapsto [v(\xi^{(1)}); \dots; v(\xi^{(n)})] \in \mathbb{R}^n$, this is easily seen to be equivalent to the invertibility of the $n \times n$ matrix with entries $v_j(\xi^{(i)})$, where (v_1, \dots, v_n) denotes a basis for \mathcal{V} . Fixing such a basis, the determinant of this matrix must be nonzero, and hence must be either always positive or always negative for all pointsets $\Xi = (\xi^{(1)}, \dots, \xi^{(n)})$ satisfying $\xi^{(1)} < \dots < \xi^{(n)}$. Protected against division by zero, one can now consider the function $L_{\Xi, i} \in \mathcal{V}$ defined for $x \in \mathcal{X}$ by

$$L_{\Xi, i}(x) = \frac{\begin{vmatrix} \dots & v_1(\xi^{(i-1)}) & v_1(x) & v_1(\xi^{(i+1)}) & \dots \\ & \vdots & \vdots & \vdots & \\ \dots & v_n(\xi^{(i-1)}) & v_n(x) & v_n(\xi^{(i+1)}) & \dots \end{vmatrix}}{\begin{vmatrix} \dots & v_1(\xi^{(i-1)}) & v_1(\xi^{(i)}) & v_1(\xi^{(i+1)}) & \dots \\ & \vdots & \vdots & \vdots & \\ \dots & v_n(\xi^{(i-1)}) & v_n(\xi^{(i)}) & v_n(\xi^{(i+1)}) & \dots \end{vmatrix}}.$$

It is called the i th fundamental Lagrange interpolator on Ξ , by virtue of $L_{\Xi, i}(\Xi^{(j)}) = \delta_{i, j}$ for $i, j \in \{1, \dots, n\}$. Note that $L_{\Xi, i}$ has no zeros besides $\xi^{(1)}, \dots, \xi^{(i-1)}, \xi^{(i+1)}, \dots, \xi^{(n)}$, otherwise $\gamma_1 = \dots = \gamma_n = 0$ could be interpolated on a set of n distinct points by two different functions from \mathcal{V} , namely by 0 and by $L_{\Xi, i}$. Note also that, for a fixed $x \notin \{\xi^{(1)}, \dots, \xi^{(n)}\}$, the sequence $(L_{\Xi, i}(x))_{i=1}^n$ cannot keep a constant sign when $n \geq 3$: if $x \in (\xi^{(i)}, \xi^{(i+1)})$ for some $i \in \{1, \dots, n-1\}$, then $L_{\Xi, i}(x)$ and $L_{\Xi, i+1}(x)$ are both positive, but $L_{\Xi, i-1}(x)$ or $L_{\Xi, i+2}(x)$ —whichever exists—is negative, and if $x < \xi^{(1)}$, say, then $L_{\Xi, 1}(x)$ is positive, but $L_{\Xi, 2}(x)$ is negative.

In the multivariate situation, it is easy to realize that the matrix with entries $v_j(\xi^{(i)})$ cannot be invertible for all choices of distinct points $\xi^{(1)}, \dots, \xi^{(n)}$. This is formalized by Mairhuber–Curtis theorem: if $d > 1$ and $\mathcal{X} \subseteq \mathbb{R}^d$ contains an interior point, then there is no Chebyshev subspace \mathcal{V} of $C(\mathcal{X})$ with dimension $n \geq 2$. Thus, one usually considers Chebyshev spaces on $\mathcal{X} = [-1, 1]$ and $\mathcal{X} = \mathbb{T}$, with the prototypical examples being spaces of algebraic polynomials and of trigonometric

polynomials. In these univariate situations, involving a simple notion of differentiation, Lagrange interpolation can painlessly be generalized to Hermite interpolation, leading to the introduction of extended Chebyshev spaces as spaces for which Hermite interpolation is always possible and unique. On compact intervals $\mathcal{X} = [a, b]$, extended Chebyshev spaces even turn out to be extended complete Chebyshev spaces¹. These spaces do not need a formal definition here, as it suffices to say that they are characterized by the existence of a basis (u_0, \dots, u_{n-1}) of the form

$$\begin{aligned} u_0(x) &= w_0(x), \\ u_1(x) &= w_0(x) \int_t^x w_1(x_1) dx_1, \\ &\vdots \\ u_{n-1}(x) &= w_0(x) \int_t^x w_1(x_1) \int_t^{x_1} w_2(x_2) \cdots \int_t^{x_{n-2}} w_{n-1}(x_{n-1}) dx_{n-1} \cdots dx_2 dx_1, \end{aligned}$$

relative to positive weights $w_0 \in C^{n-1}[a, b]$, $w_1 \in C^{n-2}[a, b]$, \dots , $w_{n-1} \in C[a, b]$ and a point $t \in [a, b]$. Clearly, taking $w_0 = \cdots = w_{n-1} = 1$ generates the shifted monomial basis with $u_j(x) = (x-t)^j/j!$.

4 Reminders on Optimal Recovery

The question addressed in this note is an instance of the generic Optimal Recovery problem, on which a brief rundown is laid out here. The following results, either classical or due to [6] for the approximability model, can all be found in [9, Chapters 9 and 10]. In an abstract setting, an object f from a normed space F —not necessarily a function space—is assumed to belong to a model set $\mathcal{K} \subseteq F$ and is observed via $y = \Lambda f$ for some linear map $\Lambda : F \rightarrow \mathbb{R}^m$, i.e., via $y_i = \lambda_i(f)$, $i = 1, \dots, m$, for some linear functionals $\lambda_1, \dots, \lambda_m \in F^*$. The goal is to recover not necessarily f itself, but $Q(f)$ for some linear map $Q : F \rightarrow Z$, and to do so in an optimal way, so as to minimize the worst-case error over \mathcal{K} , defined as

$$\text{wce}_{\mathcal{K}, Q}(\Delta) := \sup_{f \in \mathcal{K}} \|Q(f) - \Delta(\Lambda f)\|_Z.$$

If the model set \mathcal{K} is symmetric and convex and if the quantity of interest $Q : F \rightarrow \mathbb{R}$ is a linear functional, it is well known that the set of recovery maps $\Delta : \mathbb{R}^m \rightarrow \mathbb{R}$ minimizing $\text{wce}_{\mathcal{K}, Q}(\Delta)$ contains a linear map Δ^\sharp . This linearity result is a typical statement in Optimal Recovery (see [14] for an in-depth discussion), but it is not always constructive. In case of the approximability set $\mathcal{K}_{\mathcal{V}, \varepsilon} = \{f \in F : \text{dist}_F(f, \mathcal{V}) \leq \varepsilon\}$ relative to a linear subspace \mathcal{V} of F and a parameter $\varepsilon \geq 0$, such a linear optimal recovery map is given as $\Delta^\sharp = \langle a^\sharp, \cdot \rangle$, where $a^\sharp \in \mathbb{R}^m$ is a solution to

$$(3) \quad \underset{a \in \mathbb{R}^m}{\text{minimize}} \left\| Q - \sum_{i=1}^m a_i \lambda_i \right\|_{F^*} \quad \text{subject to} \quad \sum_{i=1}^m a_i \lambda_i(v) = Q(v) \quad \text{for all } v \in \mathcal{V}.$$

¹This result can be found in [3, Theorem 5, p 97] with a different terminology. It has also been reproved in a simpler way in [10, Appendix A.3].

This result was proved in [6] as a consequence of the Hahn–Banach extension theorem. It can also be deduced from the above linearity result—which is a consequence of the Hahn–Banach separation theorem—using the following argument already outlined in [8]: since it is enough to minimize the worst-case error over $\mathcal{K}_{\mathcal{V},\varepsilon}$ among linear recovery maps $\Delta_a = \langle a, \cdot \rangle$, and since the worst-case error for linear maps Δ_a is

$$\begin{aligned}
 \text{wce}_{\mathcal{K}_{\mathcal{V},\varepsilon},Q}(\Delta_a) &= \sup_{f \in C(\mathcal{X})} \left\{ \left| Q(f) - \sum_{i=1}^m a_i \lambda_i(f) \right| : \|f - v\|_{C(\mathcal{X})} \leq \varepsilon \text{ for some } v \in \mathcal{V} \right\} \\
 &= \sup_{\substack{f \in C(\mathcal{X}) \\ v \in \mathcal{V}}} \left\{ \left| Q(f) - \sum_{i=1}^m a_i \lambda_i(f) \right| : \|f - v\|_{C(\mathcal{X})} \leq \varepsilon \right\} \\
 &= \sup_{\substack{g \in C(\mathcal{X}) \\ v \in \mathcal{V}}} \left\{ \left| \left(Q(g) - \sum_{i=1}^m a_i \lambda_i(g) \right) + \left(Q(v) - \sum_{i=1}^m a_i \lambda_i(v) \right) \right| : \|g\|_{C(\mathcal{X})} \leq \varepsilon \right\} \\
 &= \sup_{g \in C(\mathcal{X})} \left\{ \left| Q(g) - \sum_{i=1}^m a_i \lambda_i(g) \right| : \|g\|_{C(\mathcal{X})} \leq \varepsilon \right\} + \sup_{v \in \mathcal{V}} \left\{ \left| Q(v) - \sum_{i=1}^m a_i \lambda_i(v) \right| \right\} \\
 (4) \quad &= \left\| Q - \sum_{i=1}^m a_i \lambda_i \right\|_{F^*} \times \varepsilon + \begin{cases} +\infty & \text{if } Q(v) \neq \sum_{i=1}^m a_i \lambda_i(v) \text{ for some } v \in \mathcal{V} \\ 0 & \text{if } Q(v) = \sum_{i=1}^m a_i \lambda_i(v) \text{ for all } v \in \mathcal{V} \end{cases},
 \end{aligned}$$

the minimization of the latter among all $a \in \mathbb{R}^m$ indeed reduces to the program (3).

It is also worth pointing out that the minimal worst-case error—aka intrinsic error—cannot exceed the so-called null error. Over the model set $\mathcal{K}_{\mathcal{V},\varepsilon}$, this means that the minimal worst-case error for any linear quantity of interest $Q : F \rightarrow Z$ is lower-bounded by the product of the approximability parameter ε and an indicator $\mu_{\mathcal{V},Q}(\Lambda)$ of the compatibility between the model (through \mathcal{V}) and the observation process (through Λ). Precisely, one has

$$(5) \quad \min_{\Delta: \mathbb{R}^m \rightarrow \mathbb{R}} \text{wce}_{\mathcal{K}_{\mathcal{V},\varepsilon},Q}(\Delta) \geq \mu_{\mathcal{V},Q}(\Lambda) \times \varepsilon, \quad \text{where } \mu_{\mathcal{V},Q}(\Lambda) := \sup_{h \in \ker(\Lambda) \setminus \{0\}} \frac{\|Q(h)\|_Z}{\text{dist}_{C(\mathcal{X})}(h, \mathcal{V})},$$

with equality occurring when $Q : F \rightarrow \mathbb{R}$ is a linear functional. The indicator $\mu_{\mathcal{V},Q}(\Lambda)$ was introduced in the article [6], which provides references to its earlier appearances in the case $Q = \text{Id}$, where it can be interpreted as the reciprocal of an angle between \mathcal{V} and $\ker(\Lambda)$.

In the framework of this note, where the space F is $C(\mathcal{X})$ and the observation functionals λ_i are evaluations $\delta_{x^{(i)}}$ at points $x^{(i)}$, if $Q = \delta_x$ is the evaluation at a point $x \notin \{x^{(1)}, \dots, x^{(m)}\}$, then one has $\|Q - \sum_{i=1}^m a_i \lambda_i\|_{F^*} = 1 + \|a\|_1$, hence an optimal recovery map over $\mathcal{K}_{\mathcal{V},\varepsilon}$ takes the form $\Delta^\sharp = \langle a^\sharp(x), \cdot \rangle$, where $a^\sharp(x) \in \mathbb{R}^m$ is a solution to

$$(6) \quad \underset{a \in \mathbb{R}^m}{\text{minimize}} \|a\|_1 \quad \text{subject to} \quad \sum_{i=1}^m a_i v(x^{(i)}) = v(x) \text{ for all } v \in \mathcal{V}.$$

Here and in the rest of this section, the subspace $\mathcal{V} \subseteq C(\mathcal{X})$ is arbitrary and the following facts do not yet rely on it being a Chebyshev space. For instance, the above optimization program always

reduces to (2) with $b = b(x) := [v_1(x); \dots; v_n(x)]$, simply by remarking that the constraint in (6) is met for all $v \in \mathcal{V}$ if and only if it is met for all of the elements of a basis (v_1, \dots, v_n) for \mathcal{V} . Furthermore, since $Q = \delta_x$ is a linear functional, equality holds in (5), which, in conjunction with (4), leads to an important identity already found in [6, Subsection 4.2], namely

$$\mu_{\mathcal{V}, \delta_x}(\Lambda_{\mathbf{X}}) = 1 + \|a^\sharp(x)\|_1,$$

where $\Lambda_{\mathbf{X}} : F \rightarrow \mathbb{R}^m$ denotes the linear map defined by $\Lambda_{\mathbf{X}}(f) = [f(x^{(1)}); \dots; f(x^{(m)})]$.

For the full recovery problem, i.e., for $Q = \text{Id}$, it was noticed in [6] that solving (6) for all $x \in \mathcal{X}$ demonstrates the existence of an optimal recovery map $\Delta^\sharp : \mathbb{R}^m \rightarrow C(\mathcal{X})$ which is linear. Precisely, if $a^\sharp(x) \in \mathbb{R}^m$ represents again a (not necessarily unique) solution to (6), then the linear map Δ^\sharp defined by $\Delta^\sharp(y)(x) = \sum_{i=1}^m y_i a_i^\sharp(x)$ minimizes the worst-case error over $\mathcal{K}_{\mathcal{V}, \varepsilon}$. Indeed, for any $f \in \mathcal{K}_{\mathcal{V}, \varepsilon}$, with $v \in \mathcal{V}$ chosen so that $\text{dist}_{C(\mathcal{X})}(f, \mathcal{V}) = \|f - v\|_{C(\mathcal{X})}$, one has

$$\begin{aligned} \|f - \Delta^\sharp(\Lambda_{\mathbf{X}} f)\|_{C(\mathcal{X})} &= \max_{x \in \mathcal{X}} \left| f(x) - \sum_{i=1}^m f(x^{(i)}) a_i^\sharp(x) \right| = \max_{x \in \mathcal{X}} \left| (f - v)(x) - \sum_{i=1}^m (f - v)(x^{(i)}) a_i^\sharp(x) \right| \\ &\leq \max_{x \in \mathcal{X}} \left\| \delta_x - \sum_{i=1}^m a_i^\sharp(x) \delta_{x^{(i)}} \right\|_{C(\mathcal{X})^*} \times \|f - v\|_{C(\mathcal{X})} \leq \max_{x \in \mathcal{X}} (1 + \|a^\sharp(x)\|_1) \times \varepsilon \\ &= \max_{x \in \mathcal{X}} \mu_{\mathcal{V}, \delta_x}(\Lambda_{\mathbf{X}}) \times \varepsilon. \end{aligned}$$

Taking the supremum over $f \in \mathcal{K}_{\mathcal{V}, \varepsilon}$ while remarking $\max_{x \in \mathcal{X}} \mu_{\mathcal{V}, \delta_x}(\Lambda_{\mathbf{X}}) = \mu_{\mathcal{V}, \text{Id}}(\Lambda_{\mathbf{X}})$, it follows with the help of (5) that

$$\text{wce}_{\mathcal{K}_{\mathcal{V}, \varepsilon}, \text{Id}}(\Delta^\sharp) \leq \mu_{\mathcal{V}, \text{Id}}(\Lambda_{\mathbf{X}}) \times \varepsilon \leq \min_{\Delta: \mathbb{R}^m \rightarrow \mathbb{R}} \text{wce}_{\mathcal{K}_{\mathcal{V}, \varepsilon}, \text{Id}}(\Delta).$$

This establishes the optimality of Δ^\sharp , provided Δ^\sharp maps in $\mathcal{C}(\mathcal{X})$, i.e., provided $x \in \mathcal{X} \mapsto a^\sharp(x) \in \mathbb{R}^m$ can be made continuous by properly selecting the minimizers $a^\sharp(x)$ for all $x \in \mathcal{X}$. This was the intricate part of the argument in [6], carried out under the proviso that the space \mathcal{V} contains the constant functions. But evidently, solving (6) for all $x \in \mathcal{X}$ does not constitute a practical algorithm, as opposed to Algorithm 1. Still, the above considerations are the basis of the validation of Algorithm 1. Note in passing that, for an arbitrary subspace \mathcal{V} , it could be practical to solve (6) for all x in a fine discretization of \mathcal{X} . This would not answer to Question (Q) satisfactorily, but could nonetheless provide a good ersatz.

5 Validation of the Proposed Algorithm

It is now time to validate Algorithm 1 by showing that it does indeed return an optimal recovery map—a linear one, to boot—for the full recovery problem in $C[-1, 1]$ over $\mathcal{K}_{\mathcal{V}, \varepsilon}$ when \mathcal{V} is a Chebyshev space of dimension $n \geq 3$ and containing the constant functions. Recall that proving

this statement amounts to justifying the last two points indicated by some \triangleright in Algorithm 1, i.e., the continuity of $a_1^\sharp, \dots, a_m^\sharp$ and the optimality of the recovery map Δ^\sharp . This task will rely on a characterization, for any $x \in [-1, 1]$, of an n -sparse minimizer of the program (6), written here as

$$(7) \quad \underset{a \in \mathbb{R}^m}{\text{minimize}} \|a\|_1 \quad \text{subject to } Ma = b(x),$$

recalling that the matrix $M \in \mathbb{R}^{n \times m}$ and the vector $b(x) \in \mathbb{R}^n$ have entries

$$M_{j,i} = v_j(x^{(i)}) \quad \text{and} \quad b_j(x) = v_j(x).$$

The characterization uses (in one direction only) the following simple observation.

Lemma 1. Let \mathcal{V} be an n -dimensional Chebyshev subspace of $\mathcal{C}[-1, 1]$. For $x \notin \{x^{(1)}, \dots, x^{(m)}\}$ and $S \subseteq \{1, \dots, m\}$ of size $|S| = n$, all the entries of $M_S^{-1}b(x) \in \mathbb{R}^n$ are nonzero.

Proof. Let $c \in \mathbb{R}^n$ stand for $M_S^{-1}b(x)$. The identity $M_S c = b(x)$ reads $\sum_{i \in S} c_i v(x^{(i)}) = v(x)$ for all $v \in \mathcal{V}$. For $j \in S$, specifying the latter when v is the fundamental Lagrange interpolator L_j with zeros at $x^{(i)}$, $i \in S \setminus \{j\}$, and equal to one at $x^{(j)}$ yields $c_j = L_j(x)$, which is nonzero. \square

The above-mentioned characterization of n -sparse solutions to (6)-(7), stated next, is inspired by the simplex algorithm's certificate of optimality.

Proposition 2. Let \mathcal{V} be an n -dimensional Chebyshev subspace of $\mathcal{C}[-1, 1]$. For $x \notin \{x^{(1)}, \dots, x^{(m)}\}$ and $S \subseteq \{1, \dots, m\}$ of size $|S| = n$, the vector $a^{(S)}(x) \in \mathbb{R}^m$ defined by $[a^{(S)}(x)]_S = M_S^{-1}b(x)$ and $[a^{(S)}(x)]_{S^c} = 0$ is a solution to (7) if and only if

$$(8) \quad \|M_{S^c}^\top M_S^{-\top} \text{sgn}(M_S^{-1}b(x))\|_\infty \leq 1.$$

Proof. For simplicity of notation, the dependence on x is removed throughout the proof, so one writes $a^{(S)}$ instead of $a^{(S)}(x)$ and b instead of $b(x)$.

Suppose on the one hand that (8) holds. Then, for any $a \in \mathbb{R}^m$ such that $Ma = b$, in view of $a_S = M_S^{-1}(b - M_{S^c}a_{S^c})$, one has

$$\begin{aligned} \|a\|_1 - \|a^{(S)}\|_1 &= \|a_{S^c}\|_1 + \|a_S\|_1 - \|M_S^{-1}b\|_1 = \|a_{S^c}\|_1 + \|M_S^{-1}(b - M_{S^c}a_{S^c})\|_1 - \|M_S^{-1}b\|_1 \\ &\geq \|a_{S^c}\|_1 + \langle \text{sgn}(M_S^{-1}b), M_S^{-1}(b - M_{S^c}a_{S^c}) \rangle - \langle \text{sgn}(M_S^{-1}b), M_S^{-1}b \rangle \\ &= \|a_{S^c}\|_1 - \langle \text{sgn}(M_S^{-1}b), M_S^{-1}M_{S^c}a_{S^c} \rangle = \|a_{S^c}\|_1 - \langle M_{S^c}^\top M_S^{-\top} (\text{sgn}(M_S^{-1}b)), a_{S^c} \rangle \\ &\geq \|a_{S^c}\|_1 - \|M_{S^c}^\top M_S^{-\top} (\text{sgn}(M_S^{-1}b))\|_\infty \|a_{S^c}\|_1 \geq 0. \end{aligned}$$

This means that $\|a\|_1 \geq \|a^{(S)}\|_1$ for any feasible vector $a \in \mathbb{R}^m$ in (7), i.e., that $a^{(S)}$ is indeed a solution to (7).

Suppose on the other hand that (8) does not hold. One considers an index $\ell \in S^c$ such that $|(M_{S^c}^\top M_S^{-\top} \text{sgn}(M_S^{-1}b))_\ell| > 1$. Then, for $t \in \mathbb{R}$, one defines a vector $a \in \mathbb{R}^m$ satisfying $Ma = b$ via

$$a_{S^c} = te_\ell \quad \text{and} \quad a_S = M_S^{-1}(b - tM_{S^c}e_\ell).$$

Since the entries of $M_S^{-1}b$ are all nonzero by Lemma 1, one has $\text{sgn}(a_S) = \text{sgn}(M_S^{-1}b)$ when $|t|$ is small enough, in which case

$$\begin{aligned} \|a\|_1 &= \|a_S\|_1 + \|a_{S^c}\|_1 = \langle \text{sgn}(a_S), a_S \rangle + |t| = \langle \text{sgn}(M_S^{-1}b), M_S^{-1}(b - tM_{S^c}e_\ell) \rangle + |t| \\ &= \|M_S^{-1}b\|_1 - t \langle M_{S^c}^\top M_S^{-\top} \text{sgn}(M_S^{-1}b), e_\ell \rangle + |t| = \|a^{(S)}\|_1 - t(M_{S^c}^\top M_S^{-\top} \text{sgn}(M_S^{-1}b))_\ell + |t|. \end{aligned}$$

Thus, when $t \neq 0$ is small enough in absolute value and chosen of the appropriate sign, one obtains $\|a\|_1 < \|a^{(S)}\|_1$, so that $a^{(S)}$ is not a solution to (7). \square

All the ingredients are now in place to complete the justification of the two remaining points, treated in reverse order of appearance.

Optimality. For $k \in \{1, \dots, m+1\}$, the fact that the index set $S_k \subseteq \{1, \dots, m\}$ of size $|S_k| = n$ is the support of a minimizer of $\|a\|_1$ subject to $Ma = b(z^{(k)})$ certifies, by Proposition 2, that $\|M_{S_k}^\top M_{S_k}^{-\top} \text{sgn}(M_{S_k}^{-1}b(z^{(k)}))\|_\infty \leq 1$. But according to Lemma 1, as x moves through the subinterval $(x^{(k-1)}, x^{(k)})$, none of the entries of $M_{S_k}^{-1}b(x)$ can vanish, meaning that $\text{sgn}(M_{S_k}^{-1}b(x))$ stays the same as $\text{sgn}(M_{S_k}^{-1}b(z^{(k)}))$, hence implying that $\|M_{S_k}^\top M_{S_k}^{-\top} \text{sgn}(M_{S_k}^{-1}b(x))\|_\infty \leq 1$ for any $x \in (x^{(k-1)}, x^{(k)})$. By Proposition 2 again, this ensures that the vector supported on S_k and equal to $M_{S_k}^{-1}b(x)$ there—this is precisely $a^\sharp(x)$ as defined in Algorithm 1—is a solution to (6)-(7) for any $x \in (x^{(k-1)}, x^{(k)})$ and any $k \in \{1, \dots, m+1\}$.

Moreover, one also notes that $a^\sharp(x^{(k)})$ —defined as $a^\sharp(x^{(k)}) = e_k$ —is a solution to (6)-(7) for $x = x^{(k)}$. Indeed, it meets the constraint in (6) and its ℓ_1 -norm is $\|e_k\|_1 = 1$, while any $a \in \mathbb{R}^m$ meeting the constraint in (6), in particular with $v = 1$, obeys $\sum_{i=1}^m a_i = 1$, so its ℓ_1 -norm satisfies $\|a\|_1 \geq 1$.

All in all, as outlined in Section 4, the fact that the vectors $a^\sharp(x)$ are solutions to (6)-(7) for all $x \in [-1, 1]$ guarantees that $\Delta^\sharp(y) = \sum_{i=1}^m y_i a_i^\sharp$ defines a recovery map that is optimal for the full approximation problem over $\mathcal{K}_{\mathcal{V}, \varepsilon}$, provided continuity of $a_1^\sharp, \dots, a_m^\sharp$ can be established, which is done next. \square

Continuity. Since the vector $a^\sharp(x) \in \mathbb{R}^m$ was defined for $x \in (x^{(k-1)}, x^{(k)})$ by $[a^\sharp(x)]_{S_k^c} = 0$ and $[a^\sharp(x)]_{S_k} = M_{S_k}^{-1}b(x)$, the map $x \mapsto a^\sharp(x)$ is readily continuous on each subinterval $(x^{(k-1)}, x^{(k)})$. To ensure overall continuity, one should check continuity at $x^{(1)}, \dots, x^{(m)}$. Given $k \in \{1, \dots, m\}$, one shall verify e.g. that $a^\sharp(x)$ tends to $a^\sharp(x^{(k)})$ as x tends to $x^{(k)}$ while belonging to $(x^{(k-1)}, x^{(k)})$.

From above, it is known that $a^\sharp(x^{(k)}) = e_k$ is a solution to (6)-(7) with $x = x^{(k)}$. One claims that the vector $\bar{a} := \lim_{x \nearrow x^{(k)}} a^\sharp(x)$ is also a solution to (6)-(7) with $x = x^{(k)}$. Indeed, the equality

$Ma^\sharp(x) = b(x)$ passes to the limit as $x \nearrow x^{(k)}$ to give $M\bar{a} = b(x^{(k)})$, so that $\bar{a} \in \mathbb{R}^m$ is feasible for (7). Moreover, for any $a \in \mathbb{R}^m$ satisfying $Ma = b(x^{(k)})$, let $\tilde{a} \in \mathbb{R}^m$ be defined by $\tilde{a}_{S_k^c} = a_{S_k^c}$ and $\tilde{a}_{S_k} = a_{S_k} + M_{S_k}^{-1}(b(x) - b(x^{(k)}))$. From $M_{S_k^c}\tilde{a}_{S_k^c} = M_{S_k^c}a_{S_k^c}$ and $M_{S_k}\tilde{a}_{S_k} = M_{S_k}a_{S_k} + b(x) - b(x^{(k)})$, one obtains $M\tilde{a} = M_{S_k^c}\tilde{a}_{S_k^c} + M_{S_k}\tilde{a}_{S_k} = Ma + b(x) - b(x^{(k)}) = b(x)$. By the minimality property of $a^\sharp(x)$ established before, one deduces that $\|\tilde{a}\|_1 \geq \|a^\sharp(x)\|_1$, and letting x tend to $x^{(k)}$ yields $\|a\|_1 \geq \|\bar{a}\|_1$, showing that \bar{a} is indeed a solution to (6)-(7) with $x = x^{(k)}$.

Since the two minimizers e_k and \bar{a} must have the same ℓ_1 -norm, one has $\|\bar{a}\|_1 = 1$. But it also holds that $\sum_{i=1}^m \bar{a}_i = 1$ because \bar{a} satisfies the constraint in (6), in particular with $v = 1$. This implies that $\bar{a}_i \geq 0$ for all $i \in \{1, \dots, m\}$. Besides, given $j \in S_k$, the constraint in (6) now written when v is the fundamental Lagrange interpolator L_j with zeros at $x^{(i)}$, $i \in S_k \setminus \{j\}$, and equal to one at $x^{(j)}$ yields $\bar{a}_j = L_j(x^{(k)})$. Thus, if $k \notin S_k$, the sequence $(L_j(x^{(k)}))_{j \in S_k}$ would keep a constant (positive) sign. As pointed in Section 3, this is impossible under the assumption $n \geq 3$. It has therefore been established that $k \in S_k$ —in other words, the index of the right endpoint (and of the left one by a similar argument) of the k th subinterval belongs to the support associated with this subinterval.

Finally, the desired conclusion $\bar{a} = e_k$ follows from the fact that \bar{a} and e_k are now known to both be supported on S_k and from the equality $M_{S_k}\bar{a} = M_{S_k}e_k$ inferred from $M\bar{a} = b(x^{(k)}) = Me_k$. \square

6 Concluding remarks

Now that the validity of Algorithm 1 is fully justified, a few comments will be beneficial to put the result in perspective.

Global vs local optimality. The recovery map Δ^\sharp produced by Algorithm 1 is *globally* optimal, in the sense that it minimizes the *global* worst-case error (1). There is also the notion of *local* worst-case error, defined at a particular $y \in \mathbb{R}^m$ by

$$\text{lwce}(\Delta, y) = \sup_{\substack{f \in \mathcal{K} \\ \Lambda_{\mathbf{X}} f = y}} \|f - \Delta(y)\|_{C(\mathcal{X})}.$$

A *locally* optimal recovery map is one that assigns, to each $y \in \mathbb{R}^m$, a minimizer over $g \in C(\mathcal{X})$ of $\sup\{\|f - g\|_{C(\mathcal{X})}, f \in \mathcal{K}_y\}$ where $\mathcal{K}_y := \{f \in \mathcal{K} : \Lambda_{\mathbf{X}} f = y\}$, i.e., a Chebyshev center of the set \mathcal{K}_y . This note makes no claim about local optimality. Note that a locally optimal recovery map may involve a costly computation at each $y \in \mathbb{R}^m$, while the cost of constructing the globally optimal recovery map Δ^\sharp can be offloaded to an offline stage producing $a_1^\sharp, \dots, a_m^\sharp$, after which the computation of $\Delta^\sharp(y) = \sum_{i=1}^m y_i a_i^\sharp$ is almost immediate. Note also that the recovery map Δ^\sharp is actually independent of $\varepsilon > 0$ and that it can be—at least abstractly—constructed knowing only the $v_j(x^{(i)})$ but without explicit expressions for the $v_j(x)$ themselves, until one requires an evaluation of $\Delta^\sharp(y)$ at some point $x \in [-1, 1]$.

Data and model consistency. The optimal recovery map Δ^\sharp put forward in this note is data-consistent—using another jargon, it is interpolatory. Indeed, for any $y \in \mathbb{R}^m$, one can see that $\Delta^\sharp(y)(x^{(\ell)}) = y_\ell$ for all $\ell \in \{1, \dots, m\}$ from

$$\left(\sum_{i=1}^m y_i a_i^\sharp \right) (x^{(\ell)}) = \sum_{i=1}^m y_i a_i^\sharp(x^{(\ell)}) = \sum_{i=1}^m y_i \delta_{i,\ell} = y_\ell.$$

However, it is not model-consistent—in other words, $\Delta^\sharp(y)$ does not always belong to $\mathcal{K}_{\mathcal{V},\varepsilon}$. Indeed, since $\Delta^\sharp(y)$ does not depend on ε , letting $\varepsilon \searrow 0$ in the inequality $\text{dist}_{C(\mathcal{X})}(\Delta^\sharp(y), \mathcal{V}) \leq \varepsilon$ would imply that $\Delta^\sharp(y) \in \mathcal{V}$. This is not the case, but notice that $\Delta^\sharp(y)$ is nonetheless made of pieces from \mathcal{V} .

Streaming data. The cost of Algorithm 1 is concentrated mostly on the solutions to about m linear programs, which can be prohibitive for large m . In the common situation of observation points $x^{(i)}$ arriving sequentially together with the values y_i , it is natural to wonder whether the work done for the construction of an optimal recovery map based on $x^{(1)}, \dots, x^{(m)}$ can be leveraged to facilitate the construction based on $x^{(1)}, \dots, x^{(m)}$ and an added $x^{(m+1)}$ —not belonging to $(x^{(m)}, 1)$ in the following discussion. When creating the new S_k —to fix the ideas, a support S associated with a subinterval I —a simple idea is to provide the simplex algorithm with a ‘warm start’, i.e., a good initial guess \tilde{S} . For instance, if I does not contain the added $x^{(m+1)}$ as an endpoint, then the guessed \tilde{S} can be chosen as the old S associated with I . But if I does contain $x^{(m+1)}$ as an endpoint, e.g. I is the left part of an old subinterval J split by $x^{(m+1)}$, since S should contain (the indices of) the left endpoint of J and of $x^{(m+1)}$, then the guessed \tilde{S} can be a slight modification of the old S associated with J obtained by removing (the index of) the right endpoint of J and replacing it by (the index of) $x^{(m+1)}$. Empirically, the speed-up is modest and becomes more significant when the size n of the supports gets closer to m .

Ratio of norms. The arguments underpinning Algorithm 1 allow one to compute the exact value of the compatibility indicator $\mu_{\mathcal{V},\text{Id}}(\Lambda_{\mathbf{X}})$ and, as an interesting side product, they transform into a practical numerical recipe to compute the exact value of the maximal ratio of uniform and discrete norms in \mathcal{V} , i.e., of

$$(9) \quad \rho_{\mathcal{V},\mathbf{X}} := \max_{v \in \mathcal{V} \setminus \{0\}} \frac{\|v\|_\infty}{\max_{i \in \{1, \dots, m\}} |v(x^{(i)})|}.$$

When $\mathcal{V} = \mathcal{P}_n$ is the space of algebraic polynomials of degree $< n$, this maximal ratio has been well studied—in particular, for equispaced points, see [4, 15]—and a computational method akin to Remez algorithm has been proposed in [1, Section 6]. The recipe uncovered here for a Chebyshev space \mathcal{V} of dimension $n \geq 3$ and containing the constant functions amounts to solving the $m+1$ linear programs from Algorithm 1 to create the index sets S_1, \dots, S_{m+1} . After that, it is summarized as

$$(10) \quad \rho_{\mathcal{V},\mathbf{X}} = \max_{k \in \{1, \dots, m+1\}} \max_{x \in [x^{(k-1)}, x^{(k)}]} \sum_{j \in S_k} \text{sgn}((M_{S_k}^{-1} b(z^{(k)}))_j) (M_{S_k}^{-1} b(x))_j.$$

In short, one needs to compute the maximal values of functions from \mathcal{V} on $m+1$ subintervals. The justification of (10) relies on the identity $\mu_{\mathcal{V}, \text{Id}}(\Lambda_{\mathbf{X}}) = 1 + \rho_{\mathcal{V}, \mathbf{X}}$ established in [6, Lemma 6.2] and on the following observation:

$$\begin{aligned}
 \mu_{\mathcal{V}, \text{Id}}(\Lambda_{\mathbf{X}}) &= \sup_{h \in \ker \Lambda_{\mathbf{X}} \setminus \{0\}} \frac{\|h\|_{C[-1,1]}}{\text{dist}_{C[-1,1]}(h, \mathcal{V})} = \max_{k \in \{1, \dots, m+1\}} \max_{x \in [x^{(k-1)}, x^{(k)}]} \sup_{h \in \ker \Lambda_{\mathbf{X}} \setminus \{0\}} \frac{|h(x)|}{\text{dist}_{C[-1,1]}(h, \mathcal{V})} \\
 &= \max_{k \in \{1, \dots, m+1\}} \max_{x \in [x^{(k-1)}, x^{(k)}]} (\mu_{\mathcal{V}, \delta_x}(\Lambda_{\mathbf{X}})) = \max_{k \in \{1, \dots, m+1\}} \max_{x \in [x^{(k-1)}, x^{(k)}]} (1 + \|a^\sharp(x)\|_1) \\
 &= 1 + \max_{k \in \{1, \dots, m+1\}} \max_{x \in [x^{(k-1)}, x^{(k)}]} \sum_{j \in S_k} |(M_{S_k}^{-1}b(x))_j| \\
 &= 1 + \max_{k \in \{1, \dots, m+1\}} \max_{x \in [x^{(k-1)}, x^{(k)}]} \sum_{j \in S_k} \text{sgn}((M_{S_k}^{-1}b(z^{(k)}))_j) (M_{S_k}^{-1}b(x))_j,
 \end{aligned}$$

where the last step made use of the fact that the signs of $M_{S_k}^{-1}b(x)$ do not change throughout the interval $[x^{(k-1)}, x^{(k)}]$. Of course, the practicality of the recipe stemming from (10) depends on the ability to compute the maxima of functions from \mathcal{V} . This task can be efficiently performed in Chebfun [7], an open-source MATLAB package for numerical computations with functions. This feature, together with the easy handling of piecewise functions, explains why Chebfun was preferred for the implementation of Algorithm 1 in the reproducible file accompanying this note (available on the author's webpage). Furthermore, if \mathcal{V} is a space of trigonometric or algebraic polynomials, then the maximum on a subinterval of a function from \mathcal{V} can be computed via semidefinite programming, as explained in [11], see Theorem 3.1 in particular.

Best choice of evaluation points. Throughout this note, the points $x^{(1)}, \dots, x^{(m)}$ were prescribed. If one could select them freely, one would naturally want to do so in a way that makes the minimal worst-case error as small as possible. According to the previous considerations, this amounts to minimizing over all pointsets \mathbf{X}_m of size m the ratio $\rho_{\mathcal{V}, \mathbf{X}_m}$ introduced in (9). This is bound to be a difficult problem, as it is unresolved even for $m = n$ and $\mathcal{V} = \mathcal{P}_n$. Indeed, it is not hard to see that $\rho_{\mathcal{P}_n, \mathbf{X}_n}$ coincides with the L_∞ -operator norm of the interpolation operator at $\mathbf{X}_n = \{x^{(1)}, \dots, x^{(n)}\}$, aka the Lebesgue constant. Pointsets with nearly optimal Lebesgue constant are known explicitly, but pointsets with genuinely optimal Lebesgue constant are not, even though they have been characterized a long time ago, see [2, 12].

Other observation functionals. The ℓ_1 -minimizations at the heart of Algorithm 1 appear thanks to the presence of point evaluations. The situation becomes more complicated if arbitrary observation functionals were involved. Nonetheless, if one had the freedom to use any observation functionals, it would be natural to wonder about the power of point evaluations: is the minimal worst-case error much smaller with unrestricted functionals than with point evaluations only? No attempts were made to answer this question in the context of this note, but some recent advances are worth pointing out in a related context where the recovery performance is assessed via the L_2 -norm rather than the uniform norm in (1), see e.g. [5] and the references therein.

References

- [1] B. Adcock, R. B. Platte, and A. Shadrin. *Optimal sampling rates for approximating analytic functions from pointwise samples*. IMA Journal of Numerical Analysis 39.3 (2019): 1360–1390.
- [2] C. de Boor and A. Pinkus. *Proof of the conjectures of Bernstein and Erdős concerning the optimal nodes for polynomial interpolation*. Journal of Approximation Theory 24 (1978): 289–303.
- [3] W. A. Coppel. *Disconjugacy*. Springer–Verlag, 1971.
- [4] D. Coppersmith and T. J. Rivlin. *The growth of polynomials bounded at equally spaced points*. SIAM Journal on Mathematical Analysis 23.4 (1992): 970–983.
- [5] M. Dolbeault, D. Krieg, and M. Ullrich. *A sharp upper bound for sampling numbers in L_2* . Applied and Computational Harmonic Analysis 63 (2023): 113–134.
- [6] R. DeVore, S. Foucart, G. Petrova, and P. Wojtaszczyk. *Computing a quantity of interest from observational data*. Constructive Approximation 49.3 (2019): 461–508.
- [7] T. A. Driscoll, N. Hale, and L. N. Trefethen, editors. *Chebfun Guide*. Pafnuty Publications, Oxford, 2014.
- [8] S. Foucart. *Instances of computational optimal recovery: refined approximability models*. Journal of Complexity 62 (2021): 101503.
- [9] S. Foucart. *Mathematical Pictures at a Data Science Exhibition*. Cambridge University Press, 2022.
- [10] S. Foucart. *Small-normed projections onto polynomial and spline spaces*. PhD thesis, University of Cambridge, 2006.
- [11] S. Foucart and V. Powers. *Basc: constrained approximation by semidefinite programming*. IMA Journal of Numerical Analysis 37.2 (2017): 1066–1085.
- [12] T. A. Kilgore. *A characterization of the Lagrange interpolating projection with minimal Tchebycheff norm*. Journal of Approximation Theory 24 (1978): 273–288.
- [13] S. Mallat. *A Wavelet Tour of Signal Processing: the Sparse Way* (third edition). Academic Press, 2008.
- [14] E. W. Packel. *Do linear problems have linear optimal algorithms?* SIAM Review, 30.3 (1988): 388–403.
- [15] E. A. Rakhmanov. *Bounds for polynomials with a unit discrete norm*. Annals of Mathematics (2007): 55–88.
- [16] L. N. Trefethen. *Exactness of quadrature formulas*. SIAM Review 64.1 (2022): 132–150.