Journal of Fourier Analysis and Applications (2026) 32:42
https://doi.org/10.1007/s00041-026-10256-0

RESEARCH

Learning and Leveraging Anisotropy Parameters in ANOVA Approximation

Felix Bartel[1] · **Pascal Schröter**[2]

Received: 31 October 2025 / Revised: 8 March 2026 / Accepted: 17 March 2026
© The Author(s) 2026

Abstract
We present a Fourier-based approach for high-dimensional function approximation. To this end, we analyze the truncated ANOVA (analysis of variance) decomposition and learn the anisotropic smoothness properties of the target function from scattered data. This smoothness information is then incorporated into our approximation algorithm to improve the accuracy. Specifically, we employ least squares approximation using trigonometric polynomials in combination with frequency boxes of optimized aspect ratios. These frequency boxes allow for the application of the Nonequispaced Fast Fourier Transform (NFFT), which significantly accelerates the computation of the method. Our approach enables the efficient optimization of dozens of parameters to achieve high approximation accuracy with minimal overhead. Numerical experiments demonstrate the practical effectiveness of the proposed method.

Keywords High-dimensional approximation · Parameter selection · FFT

Mathematics Subject Classification 41A63 · 65T40 · 65T50

1 Introduction

Scattered data approximation methods are typically designed with a specific class of functions in mind. Any available prior information about the function to be approximated may be leveraged to fine-tune the approximation scheme, improving the accuracy. In this work, we focus on high-dimensional functions with anisotropic smoothness properties, i.e., functions whose smoothness may vary significantly across different directions in the input space.

✉ Felix Bartel
felix.bartel@ku.de

Pascal Schröter
pascal.schroeter@math.tu-chemnitz.de

[1] Mathematisch-Geographische Fakultät, KU Eichstätt-Ingolstadt, 85270 Eichstätt, Germany

[2] Faculty of Mathematics, Chemnitz University of Technology, 09107 Chemnitz, Germany

Published online: 02 April 2026

Rather than presuming knowledge of such smoothness properties a priori, we introduce a novel data-driven approach to learn the anisotropic smoothness directly from scattered function samples. This learned information is then used to adapt the approximation procedure accordingly, yielding a substantial improvement in accuracy. Moreover, since a better approximation enables a more precise estimation of the smoothness parameters, we embed this process into an iterative refinement loop. This approach allows for the simultaneous estimation and tuning of dozens of smoothness-related parameters, which is performed efficiently and integrated directly into the approximation process. Crucially, this enhancement only brings a small computational overhead to the algorithm, ensuring that the method remains scalable and computationally efficient.

To validate our approach, we conduct numerical experiments in $d = 2, 5, 9$ dimensional space. The code for the presented method has been integrated into the `pyANOVAapprox` software available on GitHub as a `Python` package, cf. [30]. The results demonstrate not only the significant practical improvement in approximation quality but also align with our theoretical analysis.

Existing work includes:

- The detection of anisotropic smoothness parameters via the (infinitely many) coefficients of a hyperbolic wavelet basis, which was investigated in [27]. The key part is that the wavelet basis is universal in that there is no need for adaptation of the basis or a priori knowledge on the anisotropy. However, this method does not apply when only samples of a function are given.
- In [13] adaptive cubature based on "steady decay of Fourier coefficients" was investigated, which introduces a stopping criterion for sampling from rank-1 lattices and digital nets based on a predetermined target accuracy. The approach we present in this paper is not adaptive, in that the samples are given but rather fixed to begin with.

For our purposes we use the trigonometric ANOVA decomposition, which is well-established for high-dimensional approximation, see e.g. [6, 23, 24]. The parameter of this method is the frequency index set $\mathcal{I} \subset \mathbb{Z}^d$, which is a union of differently sized and shaped (axis parallel) boxes, cf. Figure1.

This allows for the use of fast Fourier techniques implemented in the `pyANOVAapprox` Python package [30] and `ANOVAapprox.jl` Julia package [6, 28], utilizing the Nonequispaced Fast Fourier Transform (`NFFT`) [14]. So far cubes of frequencies with equal side length have been used with brute force or manual, heuristic choices for their size. We automate this choice, including frequency boxes with non-equal side lengths, by distributing the frequency budget such that it optimizes the error decay. To model this, we use a map constructing an increasing sequence of frequency index sets

$$\Psi : \mathbb{N} \to \mathcal{P}(\mathbb{Z}^d),\ m \mapsto \mathcal{I} \quad \text{such that} \quad |\Psi(m)| = m\,.$$

When every ANOVA term has a certain anisotropic Sobolev smoothness (see Sect. 2.2) we observe a polynomial error decay $\|f - P_{\Psi(m)} f\|_{L_2} \lesssim m^{-s_\Psi}$, where the degree

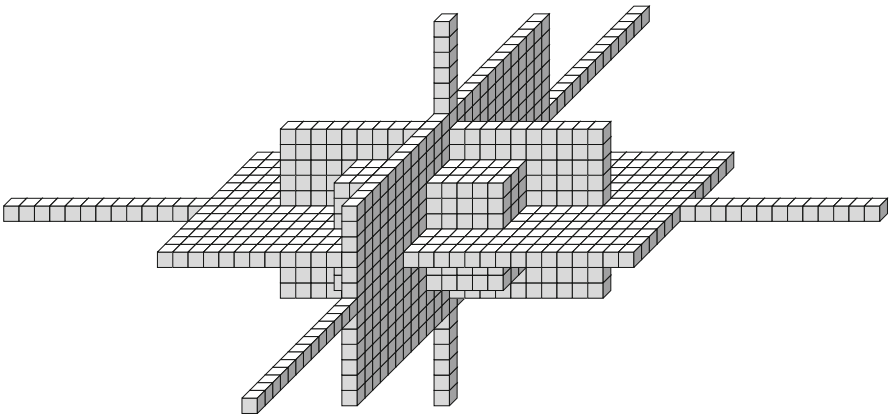

Fig. 1 Example frequencies in dimension $d = 3$ used in ANOVA approximation with 12 bandwidth parameters

s_Ψ depends on the given smoothness and the chosen Ψ. Our goal is to choose Ψ such that we have optimal approximation properties, i.e., the error decay s_Ψ is maximal.

Structure of the paper. We start by introducing the ANOVA approximation, anisotropic Sobolev spaces, and the cross-validation score in Sect. 2. In Sect. 3 we develop a method for learning the smoothness properties of a given function from samples, which we then use in Sect. 4 to improve the approximation accuracy. We end with three numerical experiments in Sect. 5 and some final remarks in Sect. 6.

Notation. In this paper we write $A_n \lesssim B_n$ or $B_n \gtrsim A_n$ if there exists $C > 0$ such that $A_n \leq C B_n$ for all $n \in \mathbb{N}$; when both relations hold we write $A_n \sim B_n$; $\langle \cdot, \cdot \rangle$ is the Euclidean inner product; $[d] := \{1, \dots, d\}$; $\mathbb{N}$ are the natural numbers, $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$, and $2\mathbb{N}_0$ are all even non-negative integers; $\mathbb{T} := \mathbb{R}/\mathbb{N}$ is the one-dimensional torus.

2 Preliminaries

2.1 ANOVA Approximation

The analysis of variance (ANOVA) has its origin in statistics with the goal of identifying dimension interactions of multivariate, high-dimensional functions. We only give a brief introduction with the domain restriction being the d-dimensional torus $\mathbb{T}^d$ for simplicity, while in-depth literature and extensions can be found in e.g. [8, 12, 16, 18, 20, 25, 29]. The core idea is that certain functions are representable as a sum of lower-dimensional functions, like

$$f(x_1, \dots, x_9) = \left| x_1 - \frac{1}{2} \right| + \cos(2\pi x_1)\cos(2\pi x_2) + \frac{\sin(2\pi x_3)}{2 + \sin(2\pi x_4)\sin(2\pi x_5)}.$$

The above function f is nine-dimensional but may be decomposed into a sum of one one-dimensional function, one two-dimensional function, and one three-dimensional one with five variables not even occurring. This assumption occurs, e.g., naturally in calculations of the electronic structure problem for molecules in [11], where component-wise interactions are intrinsic. Even when this assumption is not given, the truncation to lower-dimensional terms has been proven to beat past methods in practice on benchmark problems, cf. [29, Chapter 6].

A central tool for the analysis are integral projections. Let $\mathfrak{u} \subseteq [d]$ be a subset of coordinate indices and $\mathfrak{u}^\complement = [d] \setminus \mathfrak{u}$ its complement. Further, for vectors $\boldsymbol{x} \in \mathbb{T}^d$ indexed with a subset $\mathfrak{u} \subseteq [d]$, we define $\boldsymbol{x}_\mathfrak{u} := (x_j)_{j\in\mathfrak{u}}$. The *integral projection* of f with respect to $\mathfrak{u} \subseteq [d]$ is then given by

$$P_\mathfrak{u} f(\boldsymbol{x}) := \int_{\mathbb{T}^{d-|\mathfrak{u}|}} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}_{\mathfrak{u}^\complement} .$$

For $\mathfrak{u} \subseteq [d]$, the *ANOVA terms* and *ANOVA decomposition* are given by

$$f_\mathfrak{u} = P_\mathfrak{u} f - \sum_{\mathfrak{v}\subsetneq\mathfrak{u}} f_\mathfrak{v} \quad \text{and} \quad f = \sum_{\mathfrak{u}\subseteq[d]} f_\mathfrak{u} .$$

This orthogonal decomposition connects to a decomposition in Fourier space, where it divides the Fourier coefficients into disjoint sets of frequencies depending on their support $\operatorname{supp} \boldsymbol{k} := \{j \in [d] : k_j \neq 0\}$:

$$f_\mathfrak{u} = \sum_{\substack{\boldsymbol{k}\in\mathbb{Z}^d \\ \operatorname{supp}\boldsymbol{k}=\mathfrak{u}}} \hat{f}_{\boldsymbol{k}} \exp(2\pi \mathrm{i}\langle \boldsymbol{k}, \cdot\rangle) ,$$

with $\hat{f}_{\boldsymbol{k}} = \langle f, \exp(2\pi\mathrm{i}\langle \boldsymbol{k}, \cdot\rangle)\rangle_{L_2} = \int_{\mathbb{T}^d} f(\boldsymbol{x}) \exp(-2\pi\mathrm{i}\langle \boldsymbol{k}, \boldsymbol{x}\rangle) \, \mathrm{d}\boldsymbol{x}$. For other domains and orthonormal systems, this works analogously, see e.g. [29], or more involved with wavelets in [17]. The number of ANOVA terms is 2^d and therefore grows exponentially in the dimension, which reflects the well-known curse of dimensionality. The idea to circumvent this is to truncate the decomposition and only take a certain number of terms into account. It is common to truncate to lower-dimensional terms $f_\mathfrak{u}$ with $|\mathfrak{u}| \leq d_s$, with d_s being called superposition dimension. Then, the number of terms with respect to the spatial dimension d is $\sum_{j=1}^{d_s} \binom{d_s}{d} \in \mathcal{O}(d^{d_s})$, which grows polynomially in d instead of exponentially. Further, among the terms $f_\mathfrak{u}$, it is possible to find the ones contributing most to the overall function via sensitivity analysis, reducing the number of terms even more, which is basically comparing the normalized L_2 norms of the ANOVA terms, cf. [6, 22].

In order to compute an approximation from samples, we truncate the Fourier series of each ANOVA term. To this end we define for a bandwidth vector $\boldsymbol{m} = (m_1, \ldots, m_d) \in 2\mathbb{N}_0$

$$\widetilde{\mathcal{I}}_{\boldsymbol{m}} := \bigtimes_{j=1}^{d} \begin{cases} \{0\} & \text{if } m_j = 0 , \\ [-m_j/2, m_j/2) \cap \mathbb{Z} \setminus \{0\} & \text{otherwise,} \end{cases} \tag{2.1}$$

which is a box of frequencies for the ANOVA term $f_{\mathfrak{u}}$ with $\mathfrak{u} = \operatorname{supp} \boldsymbol{m}$. The final frequency index set for the overall ANOVA approximation then becomes

$$\mathcal{I} = \bigcup_{\mathfrak{u} \in U} \widetilde{\mathcal{I}}_{\boldsymbol{m}_{\mathfrak{u}}}, \tag{2.2}$$

with an example depicted in Figure 1.

Given points $\boldsymbol{X} = \{\boldsymbol{x}^1, \dots, \boldsymbol{x}^n\} \subseteq \mathbb{T}^d$ and samples $\boldsymbol{y} = [y_1, \dots, y_n]^\top \in \mathbb{C}^n$, we define the *least squares ANOVA approximation*

$$S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y} := \arg\min \left\{ \sum_{i=1}^{n} |g(\boldsymbol{x}^i) - y_i|^2 : g \in \operatorname{span}\{\exp(2\pi \mathrm{i} \langle \boldsymbol{k}, \cdot \rangle)\}_{\boldsymbol{k} \in \mathcal{I}} \right\}. \tag{2.3}$$

Given the full rank of the system matrix $\boldsymbol{L} := [\exp(2\pi \mathrm{i} \langle \boldsymbol{k}, \boldsymbol{x}^i \rangle)]_{i \in [n], \boldsymbol{k} \in \mathcal{I}} \in \mathbb{C}^{n \times |\mathcal{I}|}$, the Fourier coefficients of the approximation are computable by solving a system of equations

$$S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y} = \sum_{\boldsymbol{k} \in \mathcal{I}} \hat{g}_{\boldsymbol{k}} \exp(2\pi \mathrm{i} \langle \boldsymbol{k}, \cdot \rangle) \quad \text{with} \quad \hat{\boldsymbol{g}} = [\hat{g}_{\boldsymbol{k}}]_{\boldsymbol{k} \in \mathcal{I}} = (\boldsymbol{L}^* \boldsymbol{L})^{-1} \boldsymbol{L}^* \boldsymbol{y}.$$

We solve that system with the iterative `LSQR` method [21], using only matrix-vector products. With uniformly random points and at least logarithmic oversampling $n \geq 10|\mathcal{I}|(\log|\mathcal{I}| + t)$, $t > 0$, we know to have with probability exceeding $1 - 2\exp(-t)$ the condition number $\sigma_{\max}^2(\boldsymbol{L})/\sigma_{\min}^2(\boldsymbol{L}) \leq 3$, cf. [3, Lemmata 6.2 and 6.4]. With that well-conditioned system matrix $\boldsymbol{L}$, the solution of a system of equations up to machine precision $\texttt{eps} = 10^{-16}$ requires at most 56 iterations, cf. [10, Theorem 3.1.1]. In our numerical experiments the maximal number of iterations does not exceed 25. Thus, the overall computational cost of the approximation is governed by a constant multiple of the computational cost of one matrix-vector product.

Because of the box structure in the frequencies, the Nonequispaced Fast Fourier Transform (`NFFT`, cf. [14]) is applicable, making the approximation algorithm fast and parallelizable to a nearly arbitrary extent, cf. [6, 29]. This is implemented in the Python package `pyGroupedTransforms` [31] and the Julia package `GroupedTransforms.jl` [2]. For n points and accuracy ε for the matrix-vector product, this yields a computational cost of

$$\mathcal{O}\Big(\sum_{\mathfrak{u} \in U} \Big(\prod_{j \in \mathfrak{u}} m_{\mathfrak{u},j} \Big) \log \Big(\prod_{j \in \mathfrak{u}} m_{\mathfrak{u},j} \Big) + (\log \varepsilon)^{d_s} n \Big) = \mathcal{O}\Big(|\mathcal{I}| \log |\mathcal{I}| + (\log \varepsilon)^{d_s} n \Big)$$

in an FFT-like fashion. The naive matrix-vector product would yield a computational cost and memory requirements of $\mathcal{O}(|\mathcal{I}|n)$. The functionality is wrapped in the `pyANOVAapprox` Python package [30] and the `ANOVAapprox.jl` Julia package [28], where the fast matrix-vector product is used with the least squares approximation.

The error of the ANOVA approximation decomposes into the individual ANOVA terms as well.

Lemma 2.1 *The error of the ANOVA approximation* $S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{y}$ (2.3) *with* $\mathcal{I} = \bigcup_{\mathrm{u}\in U} \widetilde{\mathcal{I}}_{\boldsymbol{m}_\mathrm{u}}$ *splits into the error of the approximation of the individual ANOVA terms* f_u*, i.e.,*

$$\|S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{y} - f\|_{L_2}^2 = \sum_{\mathrm{u}\in U} \|P_{\widetilde{\mathcal{I}}_{\boldsymbol{m}_\mathrm{u}}} S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{y} - f_\mathrm{u}\|_{L_2}^2 + \sum_{\mathrm{u}\in\mathcal{P}([d])\setminus U} \|f_\mathrm{u}\|_{L_2}^2 .$$

Proof The proof follows from the orthogonality of the ANOVA decomposition. □

Thus, to describe the overall error behavior, it suffices to investigate the individual ANOVA terms.

2.2 Anisotropic Sobolev Spaces

We model the individual ANOVA terms to be in *anisotropic Sobolev spaces*

$$H^{s_1,\ldots,s_d} := \left\{ f \in L_2 : \frac{\partial^{s_1}}{\partial x_1^{s_1}} f, \ldots, \frac{\partial^{s_d}}{\partial x_d^{s_d}} f \in L_2 \right\},$$

with smoothness parameters $s_1, \ldots, s_d \in \mathbb{N}_0$. These spaces capture different smoothness properties for different dimensions, see e.g. [19]. Truncating the frequencies to boxes comes naturally with these spaces, justifying the use of the `NFFT`.

Lemma 2.2 *For* $s_1, \ldots, s_d \in \mathbb{N}_0$ *we have* $f \in H^{s_1,\ldots,s_d}$ *if and only if*

$$\|f\|_{H^{s_1,\ldots,s_d}}^2 := \sum_{\boldsymbol{k}\in\mathbb{Z}^d} \max\{1, |k_1|^{2s_1}, \ldots, |k_d|^{2s_d}\} |\hat{f}_{\boldsymbol{k}}|^2 < \infty . \tag{2.4}$$

Proof For the derivative of trigonometric polynomials, we have $\partial^{s_j}/(\partial x_j^{s_j}) \exp(2\pi\mathrm{i}\langle\boldsymbol{k},\cdot\rangle) = (2\pi\mathrm{i}k_j)^{s_j} \exp(2\pi\mathrm{i}\langle\boldsymbol{k},\cdot\rangle)$. Thus, we have the following Fourier sum for the derivatives of a given function $f = \sum_{\boldsymbol{k}\in\mathbb{Z}^d} \hat{f}_{\boldsymbol{k}} \exp(2\pi\mathrm{i}\langle\boldsymbol{k},\cdot\rangle)$

$$\frac{\partial^{s_j}}{\partial x_j^{s_j}} f = \sum_{\boldsymbol{k}\in\mathbb{Z}^d} (2\pi\mathrm{i}k_j)^{s_j} \hat{f}_{\boldsymbol{k}} \exp(2\pi\mathrm{i}\langle\boldsymbol{k},\cdot\rangle) .$$

By Parseval's identity, it is immediate that $f \in H^{s_1,\ldots,s_d}$, given the stated Fourier coefficient decay. For the reverse direction, we have

$$\begin{aligned}
\|f\|_{H^{s_1,\ldots,s_d}}^2 &= \sum_{\boldsymbol{k}\in\mathbb{Z}^d} \max\{1, |k_1|^{2s_1}, \ldots, |k_d|^{2s_d}\} |\hat{f}_{\boldsymbol{k}}|^2 \\
&\le \sum_{\boldsymbol{k}\in\mathbb{Z}^d} (1 + |k_1|^{2s_1} + |k_d|^{2s_d}) |\hat{f}_{\boldsymbol{k}}|^2 \\
&= \sum_{\boldsymbol{k}\in\mathbb{Z}^d} |\hat{f}_{\boldsymbol{k}}|^2 + \sum_{j=1}^{d} \sum_{\boldsymbol{k}\in\mathbb{Z}^d} |k_j|^{2s_j} |\hat{f}_{\boldsymbol{k}}|^2 ,
\end{aligned}$$

where all sums are finite due to $f \in H^{s_1,\dots,s_d}$ and Parseval's identity. □

Note, with similar arguments we have that $f \in H^{s_1,\dots,s_d}$ automatically implies the presumably stronger condition $\partial^{\|\boldsymbol{\alpha}\|_1}/(\partial x_1^{\alpha_1} \dots \partial x_d^{\alpha_d}) f \in L_2$ for all $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_d] \in \mathbb{N}_0^d$ such that $\alpha_1/s_1 + \dots + \alpha_d/s_d \le 1$.

From the characterization in terms of the decay of the Fourier coefficients, we immediately obtain the generalization of the anisotropic Sobolev spaces to non-integer smoothness by using the norm (2.4). Note that because of the equivalence of ℓ_p (quasi)-norms, one could do the same for ℓ_p-balls, cf. [3, Section 3.6.1].

Knowing the decay in the Fourier coefficients, we are able to investigate how the truncated Fourier sum behaves.

Lemma 2.3 *Let $\boldsymbol{m} = (m_1, \dots, m_d) \in (2\mathbb{N})^d$ be a bandwidth vector and $s_1, \dots, s_d > 0$ smoothness parameters. When projecting functions from anisotropic Sobolev spaces $H^{s_1,\dots,s_d}$ to nonempty frequency boxes*

$$\mathcal{I}_{\boldsymbol{m}} := \bigtimes_{j=1}^{d} [-m_j/2, m_j/2) \cap \mathbb{Z}$$

we obtain

$$\sup_{\|f\|_{H^{s_1,\dots,s_d}} \le 1} \|f - P_{\mathcal{I}_{\boldsymbol{m}}} f\|_{L_2}^2 = \max\left\{ \left(\frac{m_1}{2}\right)^{-2s_1}, \dots, \left(\frac{m_d}{2}\right)^{-2s_d} \right\}.$$

Proof For the upper bound, we use

$$\begin{aligned} \|f - P_{\mathcal{I}_{\boldsymbol{m}}} f\|_{L_2}^2 &= \sum_{\boldsymbol{k} \notin \mathcal{I}_{\boldsymbol{m}}} |\hat{f}_{\boldsymbol{k}}|^2 = \sum_{\boldsymbol{k} \notin \mathcal{I}_{\boldsymbol{m}}} (\max\{1, k_1^{s_1}, \dots, k_d^{s_d}\})^{-2} |\max\{1, k_1^{s_1}, \dots, k_d^{s_d}\} \hat{f}_{\boldsymbol{k}}|^2 \\ &\le \|f\|_{H^{s_1,\dots,s_d}}^2 \sup_{\boldsymbol{k} \notin \mathcal{I}_{\boldsymbol{m}}} (\max\{k_1^{s_1}, \dots, k_d^{s_d}\})^{-2} \\ &= \|f\|_{H^{s_1,\dots,s_d}}^2 \left(\inf_{\boldsymbol{k} \notin \mathcal{I}_{\boldsymbol{m}}} \max\{k_1^{s_1}, \dots, k_d^{s_d}\} \right)^{-2}, \end{aligned}$$

which evaluates to the assertion due to the definition of $\mathcal{I}_{\boldsymbol{m}}$.

For the lower bound, we construct a fooling function consisting of a trigonometric monomial

$$g = \max\{\ell_1^{-s_1}, \dots, \ell_d^{-s_d}\} \exp(2\pi \langle \boldsymbol{\ell}, \cdot \rangle) \quad \text{for} \quad \boldsymbol{\ell} \in \arg\min_{\boldsymbol{k} \notin \mathcal{I}_{\boldsymbol{m}}} \{\min\{k_1^{s_1}, \dots, k_d^{s_d}\}\}.$$

This function has an $H^{s_1,\dots,s_d}$ norm of one, and it holds

$$\sup_{\|f\|_{H^{s_1,\dots,s_d}} \le 1} \|f - P_{\mathcal{I}_{\boldsymbol{m}}} f\|_{L_2}^2 \ge \|g - P_{\mathcal{I}_{\boldsymbol{m}}} g\|_{L_2}^2 = \sup_{\boldsymbol{k} \notin \mathcal{I}_{\boldsymbol{m}}} \max\{k_1^{-2s_1}, \dots, k_d^{-2s_d}\}.$$

□

Lemma 2.3 shows the advantage of using frequency boxes instead of cubes when approximating in anisotropic Sobolev spaces. When we have a frequency budget of $|\mathcal{I}_{\boldsymbol{m}}| = m \in \mathbb{N}$ approximating with frequency cubes with side length $m_j = \sqrt[d]{m}$ for $j = 1, \dots, d$ yields

$$\sup_{\|f\|_{H^{s_1,\dots,s_d}} \le 1} \|f - P_{\mathcal{I}_{\boldsymbol{m}}} f\|_{L_2}^2 \sim m^{-2\min\{s_1,\dots,s_d\}/d},$$

whereas the optimal box ratio $m_j = (m^{1/(1/s_1+\cdots+1/s_d)})^{1/s_j}$ for $j = 1, \dots, d$ gives

$$\sup_{\|f\|_{H^{s_1,\dots,s_d}} \le 1} \|f - P_{\mathcal{I}_{\boldsymbol{m}}} f\|_{L_2}^2 \sim m^{-2/(1/s_1+\cdots+1/s_d)}.$$

For $d = 2$, $s_1 = 1$, and $s_2 = 3$, this would make a difference of m^{-1} in contrast to $m^{-3/2}$ for the optimal box ratio, which is the core motivation for this paper.

2.3 Fast Cross-Validation

The central question of this paper is to choose parameters such that the approximation has a small prediction error. It is therefore crucial to have a fast and reliable estimator of the L_2 error. A basic idea is to split the data into a training set and a validation set for estimating the error. Doing this multiple times, we obtain a reasonable estimator for the L_2 error functional known as cross-validation score, which is widely used in learning, see e.g., [7, 9, 26, 32]. A special case is where the partitionings seclude single points, then the training sets become $\{(\boldsymbol{x}^1, y_1), \dots, (\boldsymbol{x}^{i-1}, y_{i-1}), (\boldsymbol{x}^{i+1}, y_{i+1}), \dots, (\boldsymbol{x}^n, y_n)\} \subseteq \mathbb{T}^d \times \mathbb{C}$ and the validation sets $\{(\boldsymbol{x}^i, y_i)\} \subseteq \mathbb{T}^d \times \mathbb{C}$. This leads to the so-called *leave-one-out cross-validation score*.

Definition 2.4 Let $S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y} \colon \mathbb{T}^d \to \mathbb{C}$ be an approximation based on the data samples $\{(\boldsymbol{x}^1, y_1), \dots, (\boldsymbol{x}^n, y_n)\} \subseteq \mathbb{T}^d \times \mathbb{C}$. Further, let $S_{\mathcal{I}}^{\boldsymbol{X}_{-i}} \boldsymbol{y}_{-i} \colon \mathbb{T}^d \to \mathbb{C}$ be the same method applied to the samples with the i-th sample omitted. The *cross-validation score* is defined via

$$\mathrm{CV}(S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}) = \frac{1}{n} \sum_{i=1}^{n} \left| \left(S_{\mathcal{I}}^{\boldsymbol{X}_{-i}} \boldsymbol{y}_{-i} \right)(\boldsymbol{x}^i) - y_i \right|^2.$$

This parameter choice strategy is used widely in practice, and theoretical validation for the least squares approximation was shown in [3, Corollary 9.11].

A drawback of the cross-validation score is the numerical complexity of having to compute the n approximations $S_{\mathcal{I}}^{\boldsymbol{X}_{-i}} \boldsymbol{y}_{-i}$ for $i = 1, \dots, n$. To circumvent this, the *approximated cross-validation score* of the least squares approximation $S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}$ was

introduced in [5] via

$$\mathrm{FCV}(S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{y}) = \frac{1}{n}\sum_{i=1}^{n}\frac{|(S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{y})(\boldsymbol{x}^i) - y_i|^2}{(1-|\mathcal{I}|/n)^2}\,. \tag{2.5}$$

It was shown that this is the same as the actual cross-validation score $\mathrm{CV}(S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{y})$ for exact quadrature points, cf. [5]. If we do not have exact quadrature like with the scattered data setting assumed in this paper, it is still an excellent approximation and can be used instead, cf. [3, Theorem 9.20].

3 Learning Anisotropy from the ANOVA Approximation

In this section we propose a method to estimate the smoothness parameters of a function f based on samples. We model every ANOVA term coming from an anisotropic Sobolev space, i.e., $f_{\mathfrak{u}} \in H^{s_{\mathfrak{u}}}$ with smoothness parameters $\boldsymbol{s}_{\mathfrak{u}} = [s_{\mathfrak{u},j}]_{j\in\mathfrak{u}}$. The natural choice of frequencies is then a union of frequency boxes $\mathcal{I} = \bigcup_{\mathfrak{u}\in U}\widetilde{\mathcal{I}}_{\boldsymbol{m}_{\mathfrak{u}}}$ with $\widetilde{\mathcal{I}}_{\boldsymbol{m}_{\mathfrak{u}}}$ from (2.1). This makes the discussed fast Fourier methods in the software package `pyANOVAapprox` and `ANOVAapprox.jl` introduced in Sect. 2.1 applicable. Further, the truncation error splits into its ANOVA components

$$\|f - P_{\mathcal{I}}f\|_{L_2}^2 = \sum_{\mathfrak{u}\in U}\|f_{\mathfrak{u}} - P_{\widetilde{\mathcal{I}}_{\boldsymbol{m}_{\mathfrak{u}}}}f\|_{L_2}^2 + \sum_{\mathfrak{u}\in\mathcal{P}([d])\setminus U}\|f_{\mathfrak{u}}\|_{L_2}^2\,.$$

With a reasonable choice of U, the second sum becomes small. We estimate the first sum by the worst-case error and obtain with Lemma 2.3

$$\|f - P_{\mathcal{I}}f\|_{L_2}^2 \le \sum_{\mathfrak{u}\in U}\max\left\{\left(\frac{m_{\mathfrak{u},j}}{2}\right)^{-2s_{\mathfrak{u},j}}\right\}_{j\in\mathfrak{u}}\|f_{\mathfrak{u}}\|_{H^{s_{\mathfrak{u}}}}^2 + \sum_{\mathfrak{u}\in\mathcal{P}([d])\setminus U}\|f_{\mathfrak{u}}\|_{L_2}^2\,. \tag{3.1}$$

Our goal is to extract the smoothness parameters $s_{\mathfrak{u},j}$ in order to adapt the bandwidth parameters $m_{\mathfrak{u},j}$ defining $\mathcal{I}$ and controlling the approximation error. Without loss of generality, we aim to estimate the smoothness parameter $s_{\mathfrak{u},\mathfrak{u}_1}$ of the ANOVA term $\mathfrak{u}$. In order to do so, we use projections, where we vary the bandwidth in that specific dimension of that ANOVA term. For small bandwidths $(m_{\mathfrak{u},\mathfrak{u}_1}/2)^{s_{\mathfrak{u},\mathfrak{u}_1}} \le \min\{(m_{\mathfrak{u},j}/2)^{s_{\mathfrak{u},j}}\}_{j\in\{\mathfrak{u}_2,\dots,\mathfrak{u}_{|\mathfrak{u}|}\}}$ the error is then dominated by that dimension $\mathfrak{u}_1$ and we have

$$\|f_{\mathfrak{u}} - P_{\widetilde{\mathcal{I}}_{\boldsymbol{m}_{\mathfrak{u}}}}f\|_{L_2}^2 \le \max\left\{\left(\frac{m_{\mathfrak{u},j}}{2}\right)^{-2s_{\mathfrak{u},j}}\right\}_{j\in\mathfrak{u}}\|f_{\mathfrak{u}}\|_{H^{s_{\mathfrak{u}}}}^2 = \left(\frac{m_{\mathfrak{u},\mathfrak{u}_1}}{2}\right)^{-2s_{\mathfrak{u},\mathfrak{u}_1}}\|f_{\mathfrak{u}}\|_{H^{s_{\mathfrak{u}}}}^2\,. \tag{3.2}$$

For $(m_{\mathfrak{u},\mathfrak{u}_1}/2)^{s_{\mathfrak{u},\mathfrak{u}_1}} \ge \min\{(m_{\mathfrak{u},j}/2)^{s_{\mathfrak{u},j}}\}_{j\in\mathfrak{u}\setminus\{\mathfrak{u}_1\}}$ the error then flattens. We use the first range in order to extract the decay $s_{\mathfrak{u},\mathfrak{u}_1}$.

For now this uses the L_2-projection, which is not available to us. With more information – like Wavelet coefficients – this was already investigated in [27]. We have

approximated Fourier coefficients from the least squares ANOVA approximation. The error of the L_2-projection to a frequency set $\mathcal{I}_{(\boldsymbol{m}_\mathfrak{u})'}$ is equal to the 2-norm of all Fourier coefficients of the tail outside $\mathcal{I}_{(\boldsymbol{m}_\mathfrak{u})'}$, i.e.

$$\|f - P_{\mathcal{I}_{(\boldsymbol{m}_\mathfrak{u})'}} f\|_{L_2}^2 = \sum_{\boldsymbol{k} \notin \mathcal{I}_{(\boldsymbol{m}_\mathfrak{u})'}} |\hat{f}_{\boldsymbol{k}}|^2 .$$

The least squares ANOVA approximation (2.3) works with a finite frequency index set $\mathcal{I}_{\boldsymbol{m}_\mathfrak{u}}$ to begin with and gives only an estimate of the exact Fourier coefficients. By taking the 2-norm of the approximated Fourier coefficients in $\mathcal{I}_{\boldsymbol{m}_\mathfrak{u}} \setminus \mathcal{I}_{(\boldsymbol{m}_\mathfrak{u})'}$, this gives a reasonable estimate, as the following lemma shows.

Lemma 3.1 *Let $f : \mathbb{T}^d \to \mathbb{C}$ be a function and $g \in W$ an approximation thereof from a function space W. Further, let $W = V_1 \oplus V_2$ and $g = g_1 + g_2$ with $g_1 \in V_1$ and $g_2 \in V_2$. Then*

$$\|g_2\|_{L_2} - \|f - g\|_{L_2} \le \|f - P_{V_1} f\|_{L_2} \le \|g_2\|_{L_2} + \|f - g\|_{L_2} .$$

Proof We obtain the left-hand inequality using

$$\|g_2\|_{L_2} \le \|g_2 - P_{V_2} f\|_{L_2} + \|P_{V_2} f\|_{L_2} \le \|g - f\|_{L_2} + \|f - P_{V_1} f\|_{L_2} .$$

The right-hand inequality follows from

$$\|f - P_{V_1} f\|_{L_2} \le \|f - g_1\|_{L_2} = \|f - g + g_2\|_{L_2} \le \|f - g\|_{L_2} + \|g_2\|_{L_2} .$$

□

We apply this by choosing $g = S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}$, the least squares ANOVA approximation. For extracting the smoothness in dimension j of the ANOVA term in dimensions $\mathfrak{v}$, we use the decomposition $g_1 = P_{\mathcal{I}'(\mathfrak{v},j,m)} S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}$ and $g_2 = P_{\mathcal{I} \setminus \mathcal{I}'(\mathfrak{v},j,m)} S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}$ with

$$\mathcal{I}'(\mathfrak{v}, j, m') := \widetilde{\mathcal{I}}_{(m_{\mathfrak{v},1},\ldots,m_{\mathfrak{v},j-1},m',m_{\mathfrak{v},j+1},\ldots,m_{\mathfrak{v},d})} \cup \bigcup_{\mathfrak{u} \in U \setminus \{v\}} \widetilde{\mathcal{I}}_{\boldsymbol{m}_\mathfrak{u}} . \tag{3.3}$$

This yields a vector

$$\left[\| P_{\mathcal{I} \setminus \mathcal{I}'(v,j,m')} S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y} \|_{L_2}^2 \right]_{m' \in \{0,\ldots,m_{\mathfrak{u},j}\}} ,$$

which contains the sought smoothness decay, which eventually flattens for large m', as explained in (3.2). In order to extract the smoothness information, we have to identify the $\bar{m}$ where the flattening begins in order to estimate the smoothness from the components $0, \ldots, \bar{m}$.

With an initial guess, the frequency boxes are likely not optimal, and many of the exact Fourier coefficients have a smaller magnitude than the truncation error. In the approximation, this error spreads evenly among the coefficients of $S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}$. In particular,

the part of the function in $\operatorname{span}\{\exp(2\pi \mathrm{i}\langle \boldsymbol{k}, \cdot\rangle)\}_{\boldsymbol{k}\in\mathcal{I}}$ will be reconstructed, and the remainder resembles the approximation of noise either from the measurement process or the truncation itself, for which the even spread is quantified in the following lemma:

Lemma 3.2 *Let $\mathcal{I} \subseteq \mathbb{Z}^d$ be a frequency index set, $t > 0$, $\boldsymbol{X}$ be i.i.d. uniformly random points with $|\boldsymbol{X}| \geq 10|\mathcal{I}|(\log|\mathcal{I}| + t)$ for the points $\boldsymbol{X}$, and $\boldsymbol{\varepsilon} \in \mathbb{C}^n$ be i.i.d. mean-zero, random noise with variance σ^2. The expected magnitude of the approximated Fourier coefficients $S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{\varepsilon} = \sum_{\boldsymbol{k}\in\mathcal{I}} \hat{g}_{\boldsymbol{k}} \exp(2\pi \mathrm{i}\langle \boldsymbol{k}, \cdot\rangle)$ then equals*

$$\frac{2\sigma^2}{3n} \leq \mathbb{E}_{\boldsymbol{\varepsilon}}(|\hat{g}_{\boldsymbol{k}}|^2) \leq \frac{2\sigma^2}{n}\,.$$

with probability $1 - 2\exp(-t)$ in the random choice of points.

Proof Applying the least squares approximation (2.3) to i.i.d. noise $\boldsymbol{\varepsilon}$ with variance σ^2 gives the approximated Fourier coefficients $\hat{\boldsymbol{g}} = (\boldsymbol{L}^*\boldsymbol{L})^{-1}\boldsymbol{L}^*\boldsymbol{\varepsilon}$. Thus,

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\varepsilon}}(|\hat{g}_k|^2) &= \mathbb{E}_{\boldsymbol{\varepsilon}}\Big(\Big|\sum_{i=1}^{n}\big[(\boldsymbol{L}^*\boldsymbol{L})^{-1}\boldsymbol{L}^*\big]_{i,k}\varepsilon_i\Big|^2\Big)\\
&= \sum_{i=1}^{n}\sum_{j=1}^{n}\big[(\boldsymbol{L}^*\boldsymbol{L})^{-1}\boldsymbol{L}^*\big]_{i,k}\big[\boldsymbol{L}(\boldsymbol{L}^*\boldsymbol{L})^{-1}\big]_{k,j}\mathbb{E}_{\boldsymbol{\varepsilon}}(\varepsilon_i\overline{\varepsilon_j})\\
&= \sigma^2\sum_{i=1}^{n}\big[(\boldsymbol{L}^*\boldsymbol{L})^{-1}\boldsymbol{L}^*\big]_{i,k}\big[\boldsymbol{L}(\boldsymbol{L}^*\boldsymbol{L})^{-1}\big]_{k,i}\\
&= \sigma^2\big[(\boldsymbol{L}^*\boldsymbol{L})^{-1}\boldsymbol{L}^*\boldsymbol{L}(\boldsymbol{L}^*\boldsymbol{L})^{-1}\big]_{k,k}\\
&= \sigma^2\big[(\boldsymbol{L}^*\boldsymbol{L})^{-1}\big]_{k,k}\,.
\end{aligned}$$

To estimate the diagonal entries $[(\boldsymbol{L}^*\boldsymbol{L})^{-1}]_{k,k}$, we use [3, Lemmata 6.2 and 6.4], which gives

$$\frac{n}{2} \;\leq\; \lambda_{\min}(\boldsymbol{L}^*\boldsymbol{L}) \;\leq\; \lambda_{\max}(\boldsymbol{L}^*\boldsymbol{L}) \;\leq\; \frac{3}{2n}\,,$$

with the stated probability $1 - 2\exp(-t)$. This is equivalent to the Rayleigh–Ritz quotient satisfying

$$\frac{2n}{3} \;\leq\; \frac{\boldsymbol{x}^*(\boldsymbol{L}^*\boldsymbol{L})^{-1}\boldsymbol{x}}{\boldsymbol{x}^*\boldsymbol{x}} \;\leq\; \frac{2}{n} \quad \text{for all} \quad \boldsymbol{x} \in \mathbb{C}^{|\mathcal{I}|}\,.$$

In particular, we have for $\boldsymbol{x} = \boldsymbol{e}_k$

$$\frac{2n}{3} \;\leq\; [\boldsymbol{L}^*\boldsymbol{L}]^{-1}_{k,k} \;\leq\; \frac{2}{n}\,.$$

□

Thus, the flat plane corresponds to the most common value c in the magnitude of all approximated Fourier coefficients. For each $\mathfrak{u}$ and $j \in [|\mathfrak{u}|]$, we set $\bar{m}$ to be the largest m such that

$$\left[\| P_{\mathcal{I}\setminus\mathcal{I}'(\mathfrak{v},j,m)} S_{\mathcal{I}}^{X} \boldsymbol{y}\|_{L_2}^2\right]_m > c^2 |\mathcal{I} \setminus \mathcal{I}'(\mathfrak{v}, j, m)| \,.$$

It remains to estimate the smoothness from $[\| P_{\mathcal{I}\setminus\mathcal{I}'(\mathfrak{v},j,m)} S_{\mathcal{I}}^{X} \boldsymbol{y}\|_{L_2}^2]_{m\in\{0,\dots,\bar{m}\}}$. For that we use weighted linear least squares in the log-log scale.

Theorem 3.3 *Let $C_1, C_2, s > 0$ and $C_1 i^{-2s} \le y_i \le C_2 i^{-2s}$ for $i = 1, \dots, n$, modeling the error being in a tube with slope $-2s$. Applying weighted linear least squares in the log-log scale with weights $\omega_i = 1/(H_n i)$, where H_n is the n-th harmonic number and points $x_i = \log i$ yields the approximated decay behavior $D i^{-2t}$ with*

$$D = \exp\left(\frac{\sum_{i=1}^n \omega_i \log^2(i) \sum_{i=1}^n \omega_i \log(y_i) - \sum_{i=1}^n \omega_i \log(i)\log(y_i) \sum_{i=1}^n \omega_i \log(i)}{\sum_{i=1}^n \omega_i \log^2(i) - \left[\sum_{i=1}^n \omega_i \log(i)\right]^2}\right)$$

and

$$t = -\frac{1}{2}\frac{\sum_{i=1}^n \omega_i \log(i)\log(y_i) - \sum_{i=1}^n \omega_i \log(i) \sum_{i=1}^n \omega_i \log(y_i)}{\sum_{i=1}^n \omega_i \log^2(i) - \left[\sum_{i=1}^n \omega_i \log(i)\right]^2} \,.$$

If $n \ge 3$, the error for the smoothness parameters is bounded by

$$|t - s| \le \frac{4\log(C_2/C_1)}{\log(n)}$$

and

$$\log(C_1) - 4\log\left(\frac{C_2}{C_1}\right) \le \log(D) \le \log(C_2) + 4\log\left(\frac{C_2}{C_1}\right).$$

Proof The solution of the weighted least squares is derived by computing the roots of the linear least squares functional

$$\sum_{i=1}^n \omega_i \left|\log(y_i) - \log(D i^{-2t})\right|^2 = \sum_{i=1}^n \omega_i \left|\log(y_i) - \log(D) - 2t\log(i)\right|^2$$

using basic linear algebra.

In order to prove the error estimates on t and D, we first note $\sum_{i=1}^n \omega_i \log^2(i) \ge \left[\sum_{i=1}^n \omega_i \log(i)\right]^2$. Thus,

$$s - t = s + \frac{1}{2}\frac{\sum_{i=1}^n \omega_i \log(i)\log(y_i) - \sum_{i=1}^n \omega_i \log(i) \sum_{i=1}^n \omega_i \log(y_i)}{\sum_{i=1}^n \omega_i \log^2(i) - \left[\sum_{i=1}^n \omega_i \log(i)\right]^2}$$

$$\le s + \frac{1}{2}\frac{\sum_{i=1}^n \omega_i \log(i)\left[\log(C_2) - 2s\log(i)\right] - \sum_{i=1}^n \omega_i \log(i) \sum_{i=1}^n \omega_i \left[\log(C_1) - 2s\log(i)\right]}{\sum_{i=1}^n \omega_i \log^2(i) - \left[\sum_{i=1}^n \omega_i \log(i)\right]^2}$$

$$= s + \frac{1}{2}\Big(\log\Big(\frac{C_2}{C_1}\Big)\frac{\sum_{i=1}^n \omega_i \log(i)}{\sum_{i=1}^n \omega_i \log^2(i) - \big[\sum_{i=1}^n \omega_i \log(i)\big]^2} - 2s\Big)$$

$$= \frac{\log(C_2/C_1)}{2}\frac{\sum_{i=1}^n \omega_i \log(i)}{\sum_{i=1}^n \omega_i \log^2(i) - \big[\sum_{i=1}^n \omega_i \log(i)\big]^2}\,. \tag{3.4}$$

We obtain the same estimate for $t - s$ analogously. In order to estimate the latter fraction, we first need to estimate the sums by integrals, taking their monotonicity into account

$$\sum_{i=1}^n \frac{\log(i)}{i} \le \frac{\log 2}{2} + \frac{\log 3}{3} + \int_3^n \frac{\log x}{x}\,\mathrm{d}x = \frac{\log 2}{2} + \frac{\log 3}{3} + \frac{\log^2(n)}{2} - \frac{\log^2 3}{2} \tag{3.5}$$

and

$$\frac{\log^3 n}{3} - \frac{\log^3 8}{3}\sum_{i=1}^n \frac{\log^2(i)}{i} \ge \frac{\log^3(n+1)}{3} - \frac{\log^3 8}{3} + \sum_{i=1}^7 \frac{\log^2(i)}{i}\,. \tag{3.6}$$

Using (3.5), (3.6), and $\log(n) \le H_n$ in (3.4), we obtain for $n \ge 3$

$$\frac{\sum_{i=1}^n \omega_i \log(i)}{\sum_{i=1}^n \omega_i \log^2(i) - \big[\sum_{i=1}^n \omega_i \log(i)\big]^2} = \frac{\sum_{i=1}^n \frac{\log(i)}{i}}{\sum_{i=1}^n \frac{\log^2(i)}{i} - \frac{1}{H_n}\Big[\sum_{i=1}^n \frac{\log(i)}{i}\Big]^2}$$

$$\le \frac{\frac{\log^2(n)}{2} - \frac{\log^2 3}{2} + \frac{\log 2}{2} + \frac{\log 3}{3}}{\frac{\log^3(n+1)}{3} - \frac{\log^3 8}{3} + \sum_{i=2}^7 \frac{\log^2(i)}{i} - \frac{1}{\log(n)}\Big[\frac{\log^2(n)}{2} - \frac{\log^2 3}{2} + \frac{\log 2}{2} + \frac{\log 3}{3}\Big]^2} \le \frac{7}{\log(n)}\,,$$

where the last inequality follows from simple analysis of the expression at hand.

For the upper bound on $\log(D)$ we use

$$\log D \le \frac{\sum_{i=1}^n \omega_i \log^2(i)\big[\log(C_2)\sum_{i=1}^n \omega_i - 2s\sum_{i=1}^n \omega_i \log(i)\big]}{\sum_{i=1}^n \omega_i \log^2(i) - \big[\sum_{i=1}^n \omega_i \log(i)\big]^2}$$

$$- \frac{\big[\log(C_1)\sum_{i=1}^n \omega_i \log(i) - 2s\sum_{i=1}^n \omega_i \log^2(i)\big]\sum_{i=1}^n \omega_i \log(i)}{\sum_{i=1}^n \omega_i \log^2(i) - \big[\sum_{i=1}^n \omega_i \log(i)\big]^2}$$

$$= \frac{\log(C_2)\sum_{i=1}^n \omega_i \log^2(i) - \log(C_1)\big[\sum_{i=1}^n \omega_i \log(i)\big]^2}{\sum_{i=1}^n \omega_i \log^2(i) - \big[\sum_{i=1}^n \omega_i \log(i)\big]^2}$$

$$= \log(C_2) + \log\Big(\frac{C_2}{C_1}\Big)\frac{\big[\sum_{i=1}^n \omega_i \log(i)\big]^2}{\sum_{i=1}^n \omega_i \log^2(i) - \big[\sum_{i=1}^n \omega_i \log(i)\big]^2}\,.$$

Analogously, we obtain for the lower bound

$$\log(D) \geq \log(C_1) + \log\left(\frac{C_2}{C_1}\right) \frac{\left[\sum_{i=1}^n \omega_i \log(i)\right]^2}{\sum_{i=1}^n \omega_i \log^2(i) - \left[\sum_{i=1}^n \omega_i \log(i)\right]^2}.$$

It remains to estimate the fraction for $n \geq 3$ by using (3.5), (3.6), and $\log(n) \leq H_n$:

$$\frac{\left[\sum_{i=1}^n \omega_i \log(i)\right]^2}{\sum_{i=1}^n \omega_i \log^2(i) - \left[\sum_{i=1}^n \omega_i \log(i)\right]^2} = \frac{\left[\sum_{i=1}^n \frac{\log(i)}{i}\right]^2}{H_n \sum_{i=1}^n \frac{\log^2(i)}{i} - \left[\sum_{i=1}^n \frac{\log(i)}{i}\right]^2}$$

$$\leq \frac{\left[\frac{\log^2(n)}{2} - \frac{\log^2 3}{2} + \frac{\log 2}{2} + \frac{\log 3}{3}\right]^2}{\log(n)\left[\frac{\log^3(n+1)}{3} - \frac{\log^3 8}{3} + \sum_{i=2}^7 \frac{\log^2(i)}{i}\right] - \left[\frac{\log^2(n)}{2} - \frac{\log^2 3}{2} + \frac{\log 2}{2} + \frac{\log 3}{3}\right]^2} \leq 4,$$

where the last inequality follows from simple analysis of the expression at hand. □

We summarize our procedure in Algorithm 1.

Algorithm 1 `learning smoothness parameters`

Input:	$S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}$	ANOVA approximation
Output:	$J_{\mathfrak{u}}$ for $\mathfrak{u} \in U$	sets of dimensions for which the smoothness estimation succeeded
	$D_{\mathfrak{u},j}$ and $s_{\mathfrak{u},j}$ for $j \in \mathfrak{u}$, $\mathfrak{u} \in U$	estimated smoothness parameters

define c to be the most common magnitude of the Fourier coefficients of $S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}$
for $\mathfrak{u} \in U$ **do**
  set $J_{\mathfrak{u}} \leftarrow \emptyset$
  for $j \in \mathfrak{u}$ **do**
    find the largest $\bar{m}_{\mathfrak{u},j}$ such that $\left[\|P_{\mathcal{I}\setminus\mathcal{I}'(\mathfrak{u},j,m)} S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}\|_{L_2}^2\right]_m > c^2 |\mathcal{I}\setminus\mathcal{I}'(\mathfrak{u},j,m)|$ for $m = 0, \ldots, \bar{m}$ with $\mathcal{I}'(\mathfrak{u},j,m)$ as in (3.3)
    if $\bar{m}_{\mathfrak{u},j} \geq 3$ **then**
      define $\boldsymbol{v}_{\mathfrak{u},j} \leftarrow \left[\|P_{\mathcal{I}\setminus\mathcal{I}'(v,j,m')} S_{\mathcal{I}}^{\boldsymbol{X}} \boldsymbol{y}\|_{L_2}^2\right]_{m' \in \{0,\ldots,\bar{m}_{\mathfrak{u},j}\}}$
      compute $D_{\mathfrak{u},j}$ and $s_{\mathfrak{u},j}$ via weighted linear least squares in the log-log scale applied to $\boldsymbol{v}_{\mathfrak{u},j}$ according to Theorem 3.3
      set $J_{\mathfrak{u}} \leftarrow J_{\mathfrak{u}} \cup \{j\}$
    end if
  end for
end for
return $J_{\mathfrak{u}}$, $D_{\mathfrak{u},j}$, and $s_{\mathfrak{u},j}$

4 Using Anisotropy in ANOVA Approximation

In this section we use the estimated smoothness parameters $D_{\mathfrak{u},j}$ and $s_{\mathfrak{u},j}$ from Sect. 3 in order to compute a new set of frequencies $\psi(m) = \mathcal{I}$, improving the approximation quality. As it may happen that we are not able to detect the smoothness parameters for

certain dimensions for a lack of available data, we define the set $J_{\mathfrak{u}}$, which collects all dimensions $j \in \mathfrak{u}$ for which the smoothness parameter estimation was successful, cf. Algorithm 1.

According to [4, Theorem 1.1], given logarithmic oversampling $n \geq 10|\mathcal{I}|(\log|\mathcal{I}|+t)$, the error of the least squares ANOVA approximation $S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{y}$ is bounded by

$$\|f - S_{\mathcal{I}}^{\boldsymbol{X}}\boldsymbol{y}\|_{L_2}^2 \lesssim \|f - P_{\mathcal{I}}\boldsymbol{y}\|_{L_2}^2 + \sigma^2\frac{|\mathcal{I}|}{n}. \tag{4.1}$$

with probability exceeding $1 - 3\exp(-t)$. These two summands resemble

- the truncation error behaving the same as the truncated Fourier sum $\|P_{\mathcal{I}}f - f\|_{L_2}$, which we already know from (3.1), and
- the error due to noise, which only depends increasingly on the number of frequencies and not their shape.

Finding a good frequency shape ψ with a fixed frequency budget $m \in \mathbb{N}$ for the least squares ANOVA approximation $S_{\psi(m)}^{\boldsymbol{X}}\boldsymbol{y}$ is therefore the same as finding good frequencies for the truncated Fourier sum $P_{\psi(m)}f$ of which we know the error behavior (3.1). Thus, we are able to compute the optimal bandwidths by solving the optimization problem

$$\begin{aligned} \min_{m_{\mathfrak{u},j}} \quad & \sum_{\mathfrak{u}\in U} \max_{j\in J_{\mathfrak{u}}} C_{\mathfrak{u},j}(m_{\mathfrak{u},j}-1)^{-2s_{\mathfrak{u},j}} \\ \text{s.t.} \quad & \sum_{\mathfrak{u}\in U}\prod_{j=1}^{|\mathfrak{u}|}(m_{\mathfrak{u},j}-1) = m-1\,. \end{aligned} \tag{4.2}$$

Lemma 4.1 *Let $d \in \mathbb{N}$ be the dimension, $m \in \mathbb{N}$ the frequency budget, $U \subseteq \mathcal{P}([d])$ the active ANOVA terms, and $J_{\mathfrak{u}} \subseteq [|\mathfrak{u}|]$ for $\mathfrak{u} \in U$ the ANOVA terms for which we have smoothness parameters. Further, let $C_{\mathfrak{u},j} > 0$, $s_{\mathfrak{u},j} > 0$ for $j \in J_{\mathfrak{u}}$, and $m_{\mathfrak{u},j} > 0$ for $j \in \mathfrak{u}\setminus J_{\mathfrak{u}}$. Then the solution of (4.2) is given by computing $\lambda > 0$ such that the following monotone equation is fulfilled*

$$\sum_{\mathfrak{u}\in U} B_{\mathfrak{u}}^{\frac{1}{1+A_{\mathfrak{u}}}}(\lambda A_{\mathfrak{u}})^{-\frac{A_{\mathfrak{u}}}{1+A_{\mathfrak{u}}}} = m-1\,, \tag{4.3}$$

with

$$A_{\mathfrak{u}} := \frac{1}{2}\sum_{j\in J_{\mathfrak{u}}}\frac{1}{s_{\mathfrak{u},j}} \quad \text{and} \quad B_{\mathfrak{u}} := \prod_{j\in J_{\mathfrak{u}}} C_{\mathfrak{u},j}^{\frac{1}{2s_{\mathfrak{u},j}}} \prod_{j\in[|\mathfrak{u}|]\setminus J_{\mathfrak{u}}} m_{\mathfrak{u},j} - 1\,.$$

Finally, we obtain the bandwidths optimizing (4.2) via

$$m_{\mathfrak{u},j} = \left(\frac{C_{\mathfrak{u},j}}{\left[\lambda B_{\mathfrak{u}}A_{\mathfrak{u}}\right]^{\frac{1}{1+A_{\mathfrak{u}}}}}\right)^{\frac{1}{2s_{\mathfrak{u},j}}} + 1\,.$$

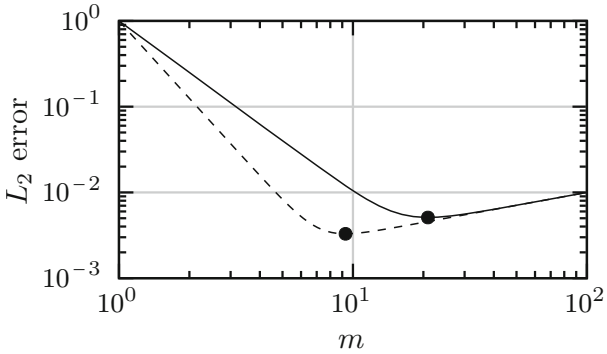

Fig. 2 Error behavior $\|S^{\boldsymbol{X}}_{\Psi(m)}\boldsymbol{y} - f\|^2_{L_2} \lesssim m^{-2s_\Psi} + \sigma^2 m/n$ for $s_\Psi = 2$ (solid) and $s_\Psi = 3$ (dashed) in the presence of noise

Proof Note that the individual terms in the inner max of the optimization problem can be assumed equal, as otherwise there are bandwidths $m_{\mathrm{u},j}$ yielding a smaller target value whilst satisfying the constraint. With the substitution

$$z_{\mathrm{u}} = C_{\mathrm{u},j}(m_{\mathrm{u},j} - 1)^{-2s_{\mathrm{u},j}} \quad \Leftrightarrow \quad m_{\mathrm{u},j} = \left(\frac{z_{\mathrm{u}}}{C_{\mathrm{u},j}}\right)^{2s_{\mathrm{u},j}} + 1\,,$$

the reduced optimization problem has the form

$$\begin{aligned} \min_{m_{\mathrm{u},j}} \quad & \sum_{\mathrm{u}\in U} z_{\mathrm{u}} \\ \text{s.t.} \quad & \sum_{\mathrm{u}\in U} \prod_{j=1}^{|\mathrm{u}|} \left(\frac{z_{\mathrm{u}}}{C_{\mathrm{u},j}}\right)^{-\frac{1}{2s_{\mathrm{u},j}}} = \sum_{\mathrm{u}\in U} B_{\mathrm{u}} z_{\mathrm{u}}^{-A_{\mathrm{u}}} = m - 1\,. \end{aligned}$$

The solution is obtained by computing the roots of the Lagrangian

$$\begin{aligned} \frac{\partial \mathcal{L}(z_{\mathrm{u}}, \lambda)}{\partial z_{\mathrm{u}}} &= 1 - \lambda A_{\mathrm{u}} B_{\mathrm{u}} z_{\mathrm{u}}^{-A_{\mathrm{u}}-1} \stackrel{!}{=} 0 \\ \Leftrightarrow \qquad z_{\mathrm{u}} &= (\lambda A_{\mathrm{u}} B_{\mathrm{u}})^{\frac{1}{A_{\mathrm{u}}+1}}\,. \end{aligned}$$

Plugging this into the constraint yields the defining equation for λ. With λ computed, this gives z_{u} and, then, $m_{\mathrm{u},j}$. □

In order to implement Lemma 4.1, we need to solve the nonlinear equation (4.3), which we do with bisection using the monotonicity. Thus, having smoothness information and a frequency budget m, we are able to compute improved bandwidths. In Sect. 3 we covered how to estimate the smoothness, so it remains to choose the frequency budget. For that we use the known error behavior from (4.1). Consequently, in the absence of noise, the frequency budget $|\mathcal{I}| = m$ should be chosen as large as possible while still satisfying the logarithmic oversampling condition. When noise is

present, one has to find m such that over- and underfitting are balanced, i.e., the L_2 error is smallest. Instead of minimizing the L_2 error, which is not available to use, we minimize the cross-validation score from Sect. 2.3 in order to find the optimal frequency budget m. The expected behavior and possible gain are depicted in Figure 2.

5 Numerical Results

In this section we test our approach with three different numerical examples. For all of them, we conduct two experiments.

- We sample the function exactly at $n = 100\,000$ uniformly random points $\boldsymbol{X}$ and use a frequency budget m such that we have logarithmic oversampling $m \log m = n$ We initialize the smoothness parameters with $D_{\mathfrak{u},j} = 1$ and $t_{\mathfrak{u},j} = 1$ for all $j \in \mathfrak{u}$ and $\mathfrak{u} \in U$. This gives a frequency index set $\psi_1(m)$ for which we compute the first approximation $S^{\boldsymbol{X}}_{\psi_1(m)}\boldsymbol{y}$. From that approximation we estimate new smoothness parameters according to Algorithm 1, which we use for a new frequency index set $\psi_2(m)$ and a new approximation $S^{\boldsymbol{X}}_{\psi_2(m)}\boldsymbol{y}$. We repeat this for 9 iterations and approximate the L_2 error for every iteration using another set of 1 000 000 uniformly random points.
- In a second experiment, we use noisy function values $\boldsymbol{y} = [f(\boldsymbol{x}^i) + \varepsilon_i]_{i=1}^n$ with Gaussian noise and a signal-to-noise ratio of

$$\mathrm{SNR}_{\mathrm{dB}} = 10 \log_{10}\left(\frac{\sum_{i=1}^n |f(\boldsymbol{x}^i)|^2}{\sum_{i=1}^n |\varepsilon_i|^2}\right) = 50\,.$$

  For the initial smoothness parameters $D_{\mathfrak{u},j} = 1$ and $t_{\mathfrak{u},j} = 1$ for all $j \in \mathfrak{u}$ and $\mathfrak{u} \in U$, we compute the fast cross-validation score, cf. sect. 2.3, and approximate the L_2 error for several values of $m \in \{300, \ldots, 10\,000\}$. We choose m such that it minimizes the fast cross-validation score $\mathrm{FCV}(S^{\boldsymbol{X}}_{\mathcal{I}}\boldsymbol{y})$ defined in (2.5) and estimate the smoothness parameters according to Algorithm 1. We repeat this 3 times.

Note that this setup gives plenty of output for academic evaluation. For a practical implementation, the number of iterations from the first experiment could be reduced, and the cross-validation score of the second experiment would be used in conjunction with an optimization procedure to reduce the computation time further. The corresponding functionality is integrated into the `pyANOVAapprox` software package [30] and the examples can be found at https://github.com/NFFT/AttributeRankingExamples/tree/main/BandwidthDetection.

5.1 Example with Complete ANOVA Decomposition

The first example has spatial dimension $d = 2$ with the function being

$$f(\boldsymbol{x}) = \sqrt{\frac{378000}{2281}}\Big(p_2(x_1) + p_4(x_2) + p_4(x_1)p_2(x_2)\Big)\,, \tag{5.1}$$

Table 1 Estimated rates $s_{\mathfrak{u},j}$ for the $d = 2$ example for every dimension of each ANOVA term $\mathfrak{u}$ with the analytical rates in brackets

	$\mathfrak{u} = \{1\}$	$\mathfrak{u} = \{2\}$	$\mathfrak{u} = \{1, 2\}$
$j = 1$	1.612 (1.5)		3.958 (3.5)
$j = 2$		3.859 (3.5)	1.717 (1.5)

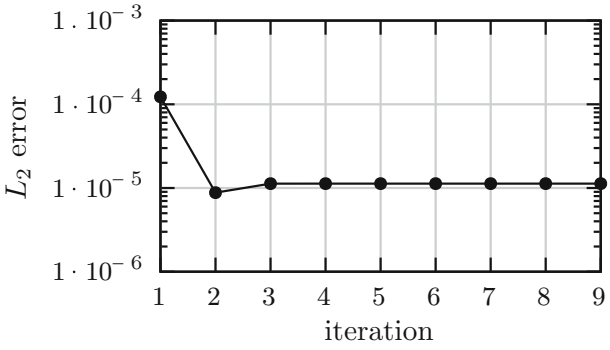

Fig. 3 L_2 error for the $d = 2$ example

where the prefactor is such that $\|f\|_{L_2} = 1$ and p_2 and p_4 are the Bernoulli polynomials

$$p_2 = x^2 - x + 1/6 \quad \text{and} \quad p_4 = x^4 - 2x^3 + x^2 - 1/30\,.$$

Their Fourier series is given by

$$p_n(x) = -\frac{n!}{(2\pi \mathrm{i})^n} \sum_{k \neq 0} \frac{\exp(2\pi \mathrm{i} k x)}{k^n}\,.$$

Thus, p_n has smoothness $s = n - 1/2$. With the zeroth Fourier coefficient zero, the ANOVA decomposition is immediately given by $f_{\{1\}}(x_1) = p_2(x_1)$, $f_{\{2\}}(x_2) = p_4(x_2)$, and $f_{\{1,2\}}(x_1, x_2) = p_4(x_1)p_2(x_2)$. We use all ANOVA terms $\{1\}$, $\{2\}$, and $\{1, 2\}$.

In the noiseless case, the estimated rates are close to the actual rates with overestimation throughout, cf. Table 1. Notice that this example highlights that the ANOVA terms do not necessarily inherit smoothness among themselves but can behave entirely independently. When it comes to the L_2 error in Figure 3, we see an improvement of a factor of 10 with the first iteration, which does not change much in further iterations.

In order to depict the frequency distribution, we have drawn boxes in Figure 4 such that the area of the box represents the total amount of frequencies and each column corresponds to one ANOVA term with their width being the proportional number of frequencies. Each column is then divided into rows for each occurring dimension in the ANOVA term, with the height being the proportional bandwidth.

We observe that a lot more of the frequency budget was spent on the ANOVA term $\{1\}$ and only a few on $\{2\}$. This is to be expected, as it requires more frequencies to

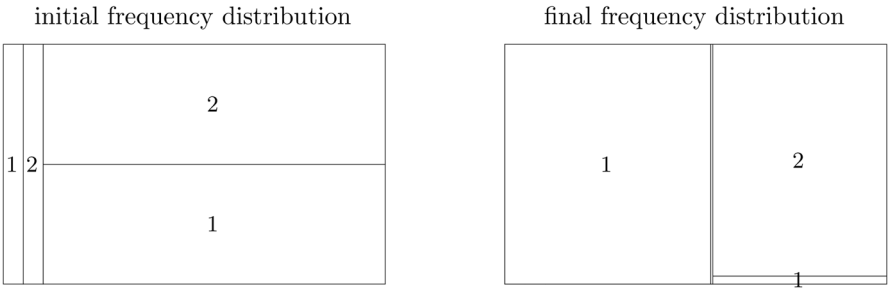

Fig. 4 Depiction of the frequency distribution in iteration 1 and 9 in the ANOVA terms for the $d = 2$ example

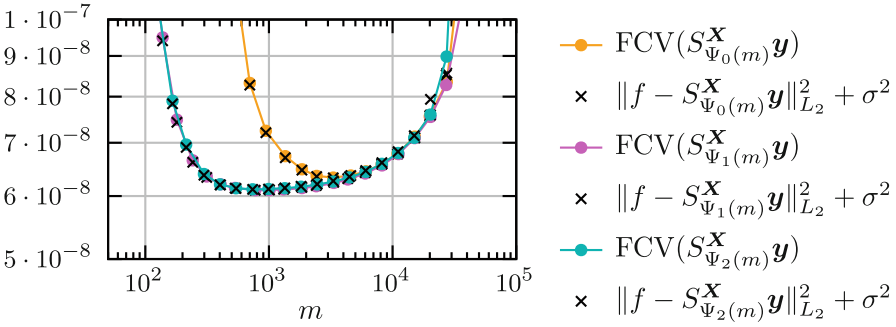

Fig. 5 Cross validation and L_2 error for the $d = 2$ example with Gaussian noise

approximate less smooth functions. Furthermore, the same effect is observed within the ANOVA term $\{1, 2\}$.

The outcome for the experiment with noise is depicted in Figure 5.

Foremost, we observe that the fast cross-validation score $\mathrm{FCV}(S^{\boldsymbol{X}}_{\mathcal{I}}\boldsymbol{y})$ is an excellent approximation for the L_2 error. Further, we observe the expected under- and overfitting behavior. With updating the frequency shape ψ, the overall error decay improves, which allows for a smaller error with fewer frequencies. In the third iteration, we only observe a very slight improvement. This aligns with our theoretical prediction from Figure 2.

5.2 Example with Fixed Superposition Dimension

In the second example, we use the function

$$f(\boldsymbol{x}) = \frac{1}{a(\boldsymbol{x})} \quad \text{with} \quad a(\boldsymbol{x}) = 1 + \frac{1}{2}\sum_{j=1}^{d} j^{-q} \sin(2\pi x_j) \quad \text{and} \quad q = 6$$

with spatial dimension $d = 5$. This function was considered in [1, 15] and solves the algebraic equation $a(\boldsymbol{x})f(\boldsymbol{x}) = 1$, mimicking the features of a partial differential

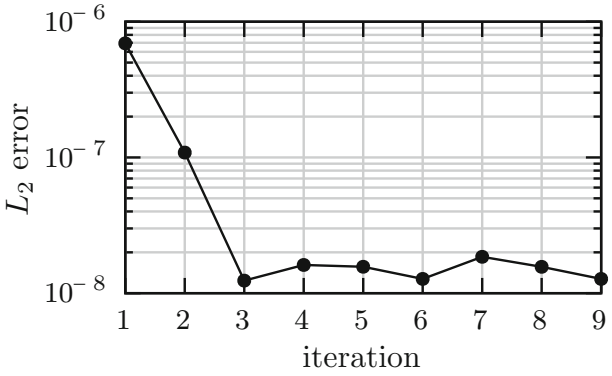

Fig. 6 L_2 error for the $d = 5$ example

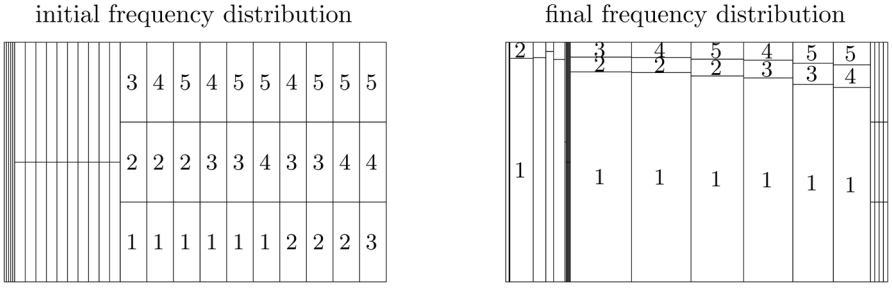

Fig. 7 Depiction of the frequency distribution in iteration 1 and 9 in the ANOVA terms for the $d = 5$ example

equation with a random coefficient whilst avoiding the complexity of a spatial variable or the need of a finite element solver.

For this function we restrict the ANOVA approximation to up to 3-dimensional terms $U = \{\mathfrak{u} \in \mathcal{P}([d]) : |\mathfrak{u}| \leq 3\}$. For the noise-free experiment, the L_2 error is depicted in Figure 6 and the frequency distribution in Figure 7.

In the first and second iterations, the L_2 error lessens by a factor of 10 each before it stabilizes. This shows the effect of a better approximation yielding a better estimation of the smoothness parameters, which in turn yields a better approximation before a fixed point is reached. In the frequency distribution we see that more frequencies are spent for smaller dimensions. This is to be expected, as the function has decaying weights with increasing dimension.

When noise is added, we obtain the outcome depicted in Figure 8.

Here we are restricted by a high minimum number of frequencies, as the number of ANOVA terms is high and we use at least $5^{|\mathfrak{u}|}$ frequencies in each to make decay rates detectable. With this high number of frequencies, we are forced to only work in the overfitting regime, where the number of points and frequencies dominates the error behavior but not the shape, which is why we do not see an improvement. This could be improved by manually omitting small ANOVA terms in terms of the L_2 norm or global sensitivity indices, cf. Sect. 2.1.

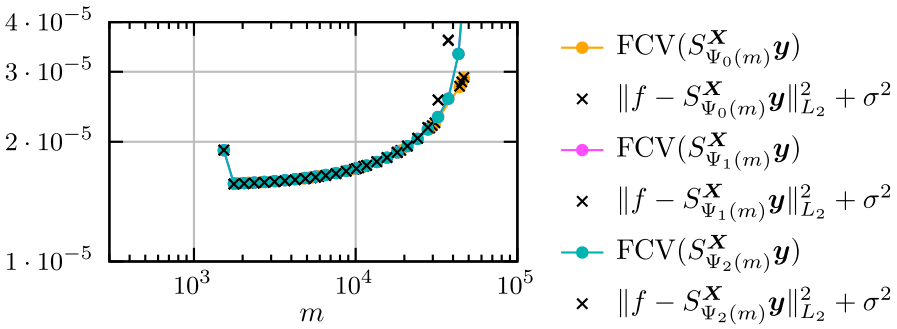

Fig. 8 Cross validation and L_2 error for the $d = 5$ example with Gaussian noise

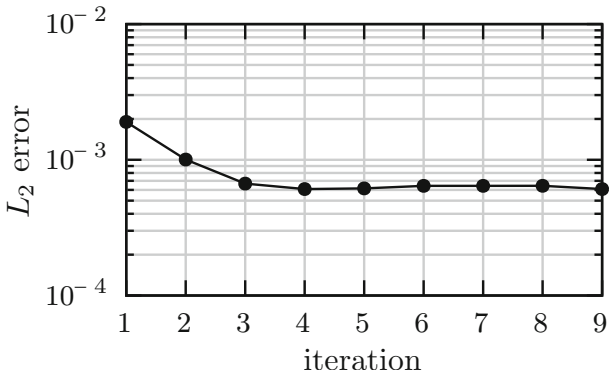

Fig. 9 L_2 error for the $d = 10$ example

5.3 Example with Known ANOVA Terms

The third example is a 9-dimensional combination of B-splines

$$f(\boldsymbol{x}) = \frac{1}{4.617\ldots}\Big(B_2(x_1)B_4(x_2)B_6(x_3) + B_2(x_4)B_4(x_5) + B_6(x_5)B_2(x_6) + B_4(x_6)B_6(x_7) + B_2(x_7)B_4(x_8) + B_6(x_8)B_2(x_9) + B_4(x_9)B_6(x_{10})\Big).$$

Functions of this type were already used in [6, 23, 24]. The B-spline of order n is a piecewise polynomial of order n, which has smoothness $s = n - 1/2$. In this example we assume to know the existing ANOVA terms, which could be determined via global sensitivity indices, cf.Sect. 2.1. In contrast to (5.1), for a product of B-splines, the lower-dimensional ANOVA terms have to be included as well.

For the noise-free experiment, the L_2 error is depicted in Figure 9 and the frequency distribution in Figure 10.

We observe an improvement of the L_2 error with the first 2 iterations before it stabilizes. The overall improvement is not as good as in the previous experiments,

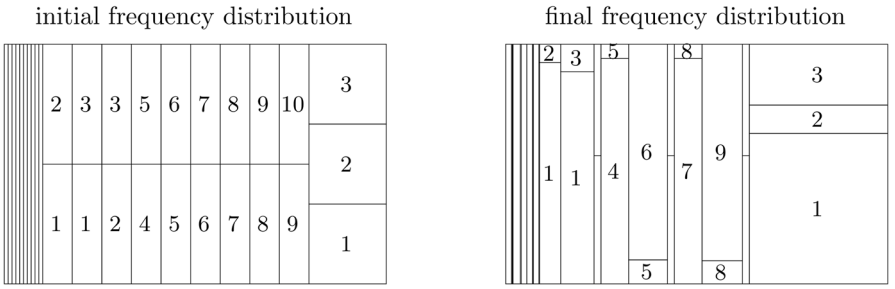

Fig. 10 Depiction of the frequency distribution in iteration 1 and 9 in the ANOVA terms for the $d = 10$ example

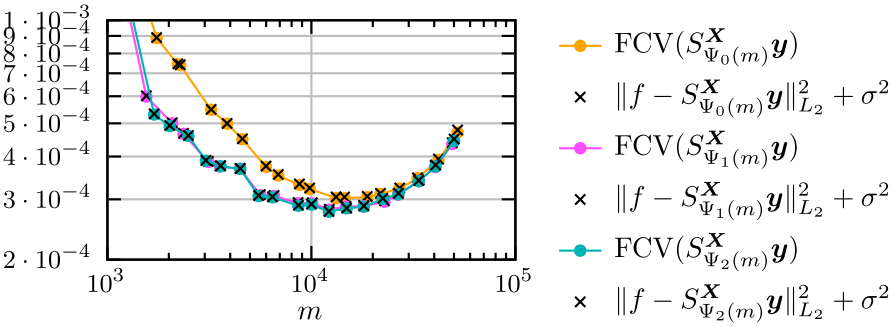

Fig. 11 Cross validation and L_2 error for the $d = 10$ example with Gaussian noise

with an overall factor of 3 in the L_2 norm accuracy. The shape of the final frequencies again resembles the respective smoothness of the approximated function.

The outcome for the experiment with noise is depicted in Figure 11.

The theoretical expectation of an improved convergence rate and smaller L_2 error is observed. The second iteration only gives marginal gains.

6 Concluding Remarks

In this paper we considered the hyperparameter selection problem in that we select the shape of frequencies for the least squares ANOVA approximation based on function samples. We set dozens of parameters based on the estimated smoothness properties of the function at hand, which we approximate from the Fourier coefficients of our approximation.

Whereas previous works for approximation [27] used linear information in the form of wavelet coefficients, we only relied on given samples, which is novel to the best of our knowledge. We utilized the well-established, fast, and memory-efficient least squares ANOVA approximation, which is a linear method. The hyperparameter tuning introduces nonlinearity in the method, which gains approximation quality without deteriorating efficiency.

Although we do not yet provide a self-contained theoretical guarantee for the entire procedure, each component of the method is supported by the theory presented in this work. In particular, the smoothness estimation relies on a steady decay of the Fourier coefficients, which is a strong assumption and needs further investigation. The numerical experiments show the advantage and reliability of the method.

Acknowledgements The authors would like to thank Daniel Potts for the fruitful discussions and valuable suggestions during the preparation of this work. Further, Felix Bartel acknowledges the time spent in Sydney with funding from the "High dimensional approximation, learning, and uncertainty" ARC discovery project.

Author Contributions F.B. and P.S. contributed to the manuscript equally.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Bartel, F., Gilbert, A.D., Kuo, F.Y., Sloan, I.H.: Minimal Subsampled Rank-1 Lattices for Multivariate Approximation with Optimal Convergence Rate. (2025) arXiv preprints https://doi.org/10.48550/arXiv.2506.07729
2. Bartel, F., Schmischke, M.: GroupedTransforms.jl: Julia Package, v1.2.0, GitHub (2024). https://github.com/NFFT/GroupedTransforms.jl
3. Bartel, F.: Least Squares in Sampling Complexity and Statistical Learning. PhD Thesis, Chemnitz University of Technology (2024). https://doi.org/10.58382/978-3-96100-204-7
4. Bartel, F.: Stability and error guarantees for least squares approximation with noisy samples in domain adaptation. SMAI J. Comput. Math. **9**, 95–120 (2023). https://doi.org/10.5802/smai-jcm.96
5. Bartel, F., Hielscher, R., Potts, D.: Fast cross-validation in harmonic approximation. Appl. Comput. Harmon. Anal. **49**(2), 415–437 (2020). https://doi.org/10.1016/j.acha.2020.05.002
6. Bartel, F., Potts, D., Schmischke, M.: Grouped transformations and regularization in high-dimensional explainable ANOVA approximation. SIAM J. Sci. Comput. **44**(3), 1606–1631 (2022). https://doi.org/10.1137/20M1374547
7. Blockeel, H., Struyf, J.: Efficient algorithms for decision tree cross-validation. J. Mach. Learn. Res. **3**(4–5), 621–650 (2003). https://doi.org/10.1162/jmlr.2003.3.4-5.621
8. Caflisch, R., Morokoff, W., Owen, A.B.: Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension. J. Comput. Finance **1**(1), 27–46 (1997). https://doi.org/10.21314/jcf.1997.005
9. De Vito, E., Pereverzyev, S., Rosasco, L.: Adaptive kernel methods using the balancing principle. Found. Comput. Math. **10**(4), 455–479 (2010). https://doi.org/10.1007/s10208-010-9064-2

10. Greenbaum, A.: Iterative Methods for Solving Linear Systems. Frontiers in Applied Mathematics, vol. 17, p. 220. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (1997). https://doi.org/10.1137/1.9781611970937
11. Griebel, M., Hamaekers, J., Heber, F.: BOSSANOVA - a bond order dissection approach for efficient electronic structure calculations. Oberwolfach Rep. **32**, 1804–1808 (2011). https://doi.org/10.4171/OWR/2011/32
12. Gu, C.: Smoothing Spline ANOVA Models. Springer, New York (2013). https://doi.org/10.1007/978-1-4614-5369-7
13. Hickernell, F.J., Jiménez Rugama, L.A., Li, D.: Adaptive quasi-Monte Carlo methods for cubature. In: Contemporary Computational Mathematics – a Celebration of the 80th Birthday of Ian Sloan. In vol. 2, pp. 597–619. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-72456-0_27
14. Keiner, J., Kunis, S., Potts, D.: Using NFFT 3–a software library for various nonequispaced fast Fourier transforms. ACM Trans. Math. Softw. **36**(4), 19–30 (2009). https://doi.org/10.1145/1555386.1555388
15. Keller, A., Kuo, F.Y., Nuyens, D., Sloan, I.H.: Regularity and Tailored Regularization of Deep Neural Networks, with Application to Parametric Pdes in Uncertainty Quantification. (2025) arXiv preprints. https://doi.org/10.48550/arXiv.2502.12496
16. Kuo, F.Y., Sloan, I.H., Wasilkowski, G.W., Woźniakowski, H.: On decompositions of multivariate functions. Math. Comp. **79**(270), 953–966 (2009). https://doi.org/10.1090/s0025-5718-09-02319-9
17. Lippert, L., Potts, D., Ullrich, T.: Fast hyperbolic wavelet regression meets ANOVA. Numer. Math. **154**(1–2), 155–207 (2023). https://doi.org/10.1007/s00211-023-01358-8
18. Liu, R., Owen, A.B.: Estimating mean dimensionality of analysis of variance decompositions. J. Am. Stat. Assoc. **101**(474), 712–721 (2006). https://doi.org/10.1198/016214505000001410
19. Nikol'skii, S.M.: Approximation of Functions of Several Variables and Imbedding Theorems. Springer, Berlin Heidelberg (1975). https://doi.org/10.1007/978-3-642-65711-5
20. Novak, E., Woźniakowski, H.: Tractability of Multivariate Problems. Vol. 1: Linear Information. EMS Tracts in Mathematics, vol. 6, p. 384. European Mathematical Society (EMS), Zürich (2008). https://doi.org/10.4171/026
21. Paige, C.C., Saunders, M.A.: LSQR: an algorithm for sparse linear equations and sparse least squares. ACM Trans. Math. Softw. **8**(1), 43–71 (1982). https://doi.org/10.1145/355984.355989
22. Potts, D., Schmischke, M.: Interpretable transformed ANOVA approximation on the example of the prevention of forest fires. Front. Appl. Math. Stat. **8** (2022) https://doi.org/10.3389/fams.2022.795250
23. Potts, D., Schmischke, M.: Approximation of high-dimensional periodic functions with Fourier-based methods. SIAM J. Numer. Anal. **59**(5), 2393–2429 (2021). https://doi.org/10.1137/20m1354921
24. Potts, D., Schmischke, M.: Learning multivariate functions with low-dimensional structures using polynomial bases. J. Comput. Appl. Math. **403**, 113821 (2022). https://doi.org/10.1016/j.cam.2021.113821
25. Rabitz, H., Alis, O.F.: General foundations of high-dimensional model representations. J. Math. Chem. **25**, 197–233 (1999). https://doi.org/10.1023/A:1019188517934
26. Rosset, S.: Bi-level path following for cross validated solution of kernel quantile regression. J. Mach. Learn. Res. **10**, 2473–2505 (2009). https://doi.org/10.1145/1390156.1390262
27. Schäfer, M., Ullrich, T., Vedel, B.: Hyperbolic wavelet analysis of classical isotropic and anisotropic Besov-Sobolev spaces. J. Fourier Anal. Appl. **27**(3), 55 (2021). https://doi.org/10.1007/s00041-021-09844-z
28. Schmischke, M.: ANOVAapprox.jl: Julia Package, v1.2.0, GitHub (2024). https://github.com/NFFT/ANOVAapprox.jl
29. Schmischke, M.: Interpretable Approximation of High-dimensional Data Based on the ANOVA Decomposition. PhD Thesis, Chemnitz University of Technology (2022). https://nbn-resolving.org/urn:nbn:de:bsz:ch1-qucosa2-791415
30. Schröter, P.: pyANOVAapprox: Python Package, v2.0.1, GitHub (2026). https://github.com/NFFT/pyANOVAapprox
31. Schröter, P.: pyGroupedTransforms: Python Package, v1.1.0, GitHub (2026). https://github.com/NFFT/pyGroupedTransforms
32. Tasche, M., Weyrich, N.: Smoothing inversion of Fourier series using generalized cross-validation. Results Math. **29**(1–2), 183–195 (1996). https://doi.org/10.1007/BF03322217

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.