

Topics in Nonlinear Approximation

Peter Oswald

March 31, 2003

1 Introductory comments

This is an extended version of a lecture for undergraduate students on topics of nonlinear approximation theory given at International University Bremen on March 14, 2003. My intention was to introduce, on relatively simple examples and without assuming knowledge about any advanced mathematical concepts, to some features of nonlinear approximation theory which attracted new attention in the last ten years. The summary contains more than what was actually covered in the lecture. It is my hope that, with more details and extra material included, a better picture has emerged. I plan to update and diversify the material as the occasion arises, and would be grateful for comments and suggestions.

Below, I assume basic knowledge about linear spaces and mappings between them, distances, norms, and Euclidean structure. As illustrative examples I use \mathbb{R}^d , \mathbb{C}^d , and simple classes of (piecewise) continuous resp. square-integrable functions with values in \mathbb{R} resp. \mathbb{C} , things that are hopefully not too hard to swallow even for a first-year student. Although there are some precise statements and even proofs given in the text, I have kept the exposition on the level of an informal discussion. You are welcome to try the formulated problems. Some references to monograph/survey literature are included (there are no really good and contemporary undergraduate textbooks that cover the subject, see [3] for a classical introductory book).

2 What is (linear) approximation theory about?

Approximation theory provides a link between discrete (finite-dimensional) and continuous (infinite-dimensional) mathematical models and their theories. Its main applications are in numerical analysis and in hierarchical modelling (multiple models of different resolution or descriptive power for the same phenomenon). There are close ties to functional analysis, optimization, complexity theory etc.

For instance, in order to do computations for engineering applications, we often replace arbitrary functions in physical models by simple ones (polynomials, step functions, etc.) which can be described by finitely many parameters. Approximation

theory answers questions such as under which assumptions and how well this can be done, how many parameters need to be invested to do it within a certain accuracy, and so on. Usually, college students get confronted with this type of questions when the deviation of differentiable functions from their Taylor polynomials (the remainder term) is discussed in the Calculus course.

Here are two classical examples:

Example 1 ([3],[10],[6, Chapter 3]). Best approximation and Chebyshev polynomials. Chebyshev (1859) has studied the following question: What is the best approximation of a continuous function $f(x) : I \rightarrow \mathbb{R}$ by polynomials of degree $\leq n$, and how to find/characterize it? Here, by I we denote a bounded, closed interval $I \subset \mathbb{R}$, and "best" is defined as a minimization problem: Find $p_n(x) = \sum_{k=0}^n c_k x^k$ such that

$$\|f - p_n\| := \sup_{x \in I} |f(x) - p_n(x)| \longmapsto \min.$$

The minimum value is denoted by $E_n(f)$ and called *best approximation*, the minimizers $p_n^*(x)$ are called the *best approximating polynomials*, and the mapping $f \rightarrow p_n^*$ *operator of best approximation*. The functional $\|\cdot\|$ is a norm on the linear space $C(I)$ of all continuous functions, and $\|f - g\|$ is interpreted as distance measure between two functions $f, g \in C(I)$. Note that $\|f_n - f\| \rightarrow 0$ is equivalent to the uniform convergence of the sequence $f_n(x)$ to the limit function $f(x)$ on the interval I , and that for continuous $f(x)$ the sup in the definition of $\|f\|$ can be replaced by max (this explains names such as maximum norm or norm of uniform convergence).

Chebyshev proved that the best approximating polynomial always exists and is unique, however, it depends on $f(x)$ in a nonlinear, complicated way. Most importantly, he gave a criterion which allows to check whether a given $p_n(x)$ is the best approximating polynomial for a given $f(x)$ (about 75 years later his alternation criterion was turned into an algorithm by Remez (1935) which is still one of the best ways to find best approximating polynomials with high accuracy).

Chebyshev's name is probably better known from the Chebyshev polynomials

$$T_n(x) = \cos(n \arccos(x)), \quad x \in [-1, 1], \quad n = 0, 1, \dots$$

Problem 1. By using trigonometry, prove that the above formula defines a polynomial of degree n whose leading coefficient is 2^{n-1} .

The polynomials $T_n(x)$ are often used to expand elementary and special functions (this leads to much faster evaluation algorithms than the use of power series, and was in the early years of mainframe computing one of the preferred methods to implement standard functions). The connection with the problem of best approximation is the following: If we solve the best approximation problem with polynomials of degree $\leq n - 1$ for $f(x) = x^n$ on the interval $[-1, 1]$, i.e., find $p_{n-1}^*(x)$ such that

$$\|x^n - p_{n-1}^*(x)\| = E_{n-1}(x^n),$$

then $T_n(x) = 2^{n-1}(x^n - p_{n-1}^*(x))$.

Problem 2. Justify this, and compute $E_{n-1}(x^n)$! (Hint: By Problem 1, we can write $2^{-(n-1)}T_n(x) = x^n - q_{n-1}(x)$ for some polynomial $q_{n-1}(x)$ of degree $\leq n-1$. This shows $E_{n-1}(x^n) \leq 2^{-(n-1)}$. Suppose $E_{n-1}(x^n) < 2^{-(n-1)}$. Consider the difference

$$r_{n-1}(x) = p_{n-1}^*(x) - q_{n-1}(x) = (x^n - q_{n-1}(x)) - (x^n - p_{n-1}^*(x)),$$

(a polynomial of degree $\leq n-1$), and check that it has at least n sign changes and thus n zeros in $(-1, 1)$.)

Because of their nonlinear dependence on $f(x)$, finding polynomials of best approximation is cumbersome, and early on the interest has been on investigating more constructive approximation methods with algebraic polynomials, and estimating their quality (for f with infinitely many derivatives, the sequence of Taylor polynomials would be one such constructive method). Other examples include Lagrange interpolation and sequences of Bernstein polynomials (1912). The latter are defined for continuous $f(x)$ on $[-1, 1]$ as

$$B_n f(x) = \sum_{k=0}^n f\left(\frac{2k-n}{n}\right) \binom{n}{k} (1+x)^k (1-x)^{n-k},$$

and have neat properties such as preserving positivity and diminishing oscillatory behavior (properties that are of interest in geometric modelling). Figure 1 shows the error $e(x) = p(x) - x^n$ between the monomial x^n and polynomials obtained by some of the above mentioned standard approximation schemes (Taylor (at $x_0 = 0$), Lagrange interpolation, best approximating, and Bernstein polynomials of degree $n-2$, and another Bernstein polynomial of degree $m \gg n$). We show only the cases $n = 6$ and $n = 10$. It is evident that the Taylor polynomial is extremely well-approximating (and superior to other schemes) in the vicinity of $x_0 = 0$, and that, with the exception of the best-approximating polynomial the errors have a tendency to grow near the endpoints of the interval $[-1, 1]$. Note that in contrast to the other polynomial schemes, B_n is not a projector on the space of polynomials of degree n , and that, although $B_m f \rightarrow f$, $m \rightarrow \infty$, for all $f \in C(-1, 1)$, the convergence rate is slow (there is no free lunch here: additional properties such as positivity come at the price of limited convergence speed).

Besides the problem of best approximation and driven by the needs of applied mathematics, another basic direction of approximation theory is about the speed of approximation processes in dependence on properties of the objects to be approximated. Its origin goes back to the Weierstrass theorem (polynomials are dense in the space of continuous functions) which can be rephrased as

$$E_n(f) \longrightarrow 0 \quad \forall f \in C(I).$$

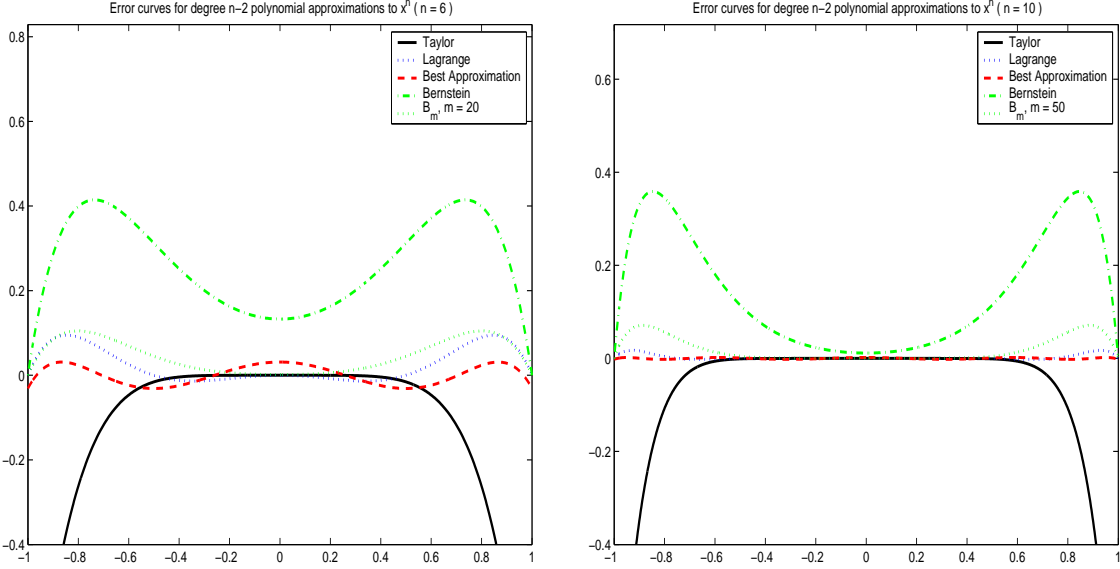


Figure 1: Comparison of polynomial approximation schemes

How fast does $\{E_n(f)\}$ tend to zero, and how does the rate of convergence (decay of $E_n(f)$ with $n \rightarrow \infty$) relate to the smoothness properties of f ? In the beginning of last century, Jackson (1913) and Bernstein (1916) have laid the foundation to what is now the theory of approximation spaces (Besov-Hölder-Lipschitz classes), see [6, Chapter]. We will not go into this direction but rather put the finger on another, more constructive aspect of quantitative approximation: Does a specific approximation method such as $\{B_n f(x)\}$ or alike achieve a good (or even optimal compared $\{E_n(f)\}$) asymptotic rate of convergence and, if not, is there a better one?

Example 2 ([1, 11, 3]). Summation of Fourier series. Each complex-valued 2π -periodic and integrable function $f(x)$ can formally be written as infinite Fourier series:

$$f(t) \sim \sum_{n=-\infty}^{\infty} c_n e^{int}, \quad c_n := \frac{1}{2\pi} \int_0^{2\pi} f(t) e^{-int} dt.$$

The system $\{e^{int}\}$ forms a complete orthogonal system in the Hilbert space $L_2(0, 2\pi)$ of square-integrable functions (two functions in this space are orthogonal if

$$(f, g) := \int_0^{2\pi} f(t) g(t)^* dt = 0,$$

and $\|f\|_2 := \sqrt{(f, f)}$ is the appropriate Hilbert space norm, where z^* denotes the complex conjugate to a complex number z , and later complex transposition for vectors resp. matrices). The Fourier method (finding $f(t)$ by first determining its Fourier coefficients c_n , and then summing the infinite series to recover $f(t)$) has triggered the

question of convergence and convergence speed of this series. For continuous periodic functions, the problem has been spiced up by a counterexample (Du Bois-Reymond, 1876) of a continuous 2π -periodic function $f(t)$ such that

$$\|f - S_n f\| \rightarrow \infty, \quad n \rightarrow \infty,$$

where

$$S_n f(t) := \sum_{|k| \leq n} c_k e^{ikt}$$

are the partial sums of its Fourier series. Bad news: not every natural approximation process does even converge (i.e., one has to be careful and knowledgeable about when to trust such a method). One way to approach the problem is to give conditions on $f(x)$ such that the partial sums can be used to recover $f(x)$. E.g., if we define the appropriate best approximations by

$$\tilde{E}_n(f) := \inf_{a_k, |k| \leq n} \|f - \sum_{|k| \leq n} a_k e^{ik \cdot}\|$$

then we can establish that

$$\tilde{E}_n(f) \leq \|f - S_n f\| \leq C \log(n+1) \tilde{E}_n(f), \quad n \geq 0.$$

Here (and throughout the expositon), unspecified positive constants C do not depend on other parameters appearing in inequalities, and may change from one appearance to the other. We also use the symbols o , O , \asymp with their usual meaning (the latter symbol $A \asymp B$ stands for a two-sided inequality of the form $C'A \leq B \leq CA$ or, equivalently, for $A = O(B)$ and $B = O(A)$).

Problem 3. Establish the above two-sided inequality by using the fact that S_n is a linear projector on the subspace of Fourier polynomials of degree $\leq n$, and that $\|S_n f\| \leq C \log(n+1) \|f\|$ for all $f \in C(0, 2\pi)$. Also, show that $\tilde{E}_n(f) \rightarrow 0$ for any continuous 2π -periodic $f(t)$ (this is the Weierstrass theorem for Fourier polynomials, a clever use of the coordinate transformation $x = \cos t$ reduces it to the one for polynomials on $[-1, 1]$).

Thus, if the function $f(t)$ can be approximated by Fourier polynomials with a speed of at least $\tilde{E}_n(f) = o(1/\log(n+1))$ then the partial sums converge to $f(t)$ uniformly, and the loss compared to the optimal speed is logarithmic (this explains why ordinary people trust the partial Fourier sums without further questioning). This argument can be refined by the theory of approximation spaces (which provide a classification of functions according to the decay of $\{\tilde{E}_n(f)\}$ or similar measures for approximation speed), and links to harmonic analysis and the study of scales of function spaces (Lipschitz, Hölder, Lebesgue, Hardy, Sobolev, Besov,...) can be established.

Alternatively, one may want to modify the approximation method, i.e., replace the sequence $\{S_n f\}$ by something better. A simple repair is provided by so-called *linear*

summation methods which are defined by a bi-infinite matrix $\Lambda = ((\lambda_{nk}, n \geq 0, k \in \mathbb{Z}))$, where each row (fixed n) represents a multiplier sequence $\lambda_n := \{\lambda_{nk}, k \in \mathbb{Z}\}$ with finitely many non-zero entries, and the λ_{nk} in each column (fixed k) tend to 1 as $n \rightarrow \infty$. Then

$$\Lambda_n f(t) := \sum_{k \in \mathbb{Z}} \lambda_{nk} c_k e^{ikt}, \quad n \geq 0,$$

defines a sequence of Fourier polynomials that may serve as replacement for $\{S_n f(t)\}$. How this may help is illustrated by looking at two classical summation methods, the Féjer method (1900)

$$F_n f(t) := \frac{S_0 f(t) + \dots + S_n f(t)}{n+1} = \sum_{|k| \leq n} \left(1 - \frac{|k|}{n+1}\right) c_k e^{ikt},$$

and the de la Vallée-Poussin method (1918)

$$V_n f(t) := \frac{1}{n} (S_n f(t) + \dots + S_{2n-1} f(t)) = 2F_{2n-1} f(t) - F_{n-1} f(t).$$

Figure 2 depicts the multiplier sequences λ_n for the three methods given by S_n , F_n , and V_n .

The basic result on the Féjer and the de la Vallée-Poussin methods are summarized as follows: Both converge for any continuous 2π -periodic $f(t)$,

$$\lim_{n \rightarrow \infty} \|f - F_n f\| = \lim_{n \rightarrow \infty} \|f - V_n f\| = 0,$$

and the latter does it, up to constants, with a speed comparable to the optimal one given by the best approximations:

$$\tilde{E}_{2n-1}(f) \leq \|f - V_n f\| \leq 4\tilde{E}_n(f), \quad n \geq 0.$$

Summary.

- Approximation theory studies properties of approximation processes. In the linear part of the theory, these are characterized by a ladder $\{X_n\}$ of finite-dimensional subspaces of growing dimension of a linear (metric or normed) space X , and mappings into them. In our examples, the ladders coincided with the subspaces of algebraic resp. Fourier polynomials of degree n in spaces of continuous functions, and the mappings were given by linear operators such as B_n , S_n , F_n , etc., or nonlinear mappings such as the operator of best approximation in the first example.
- The directions of the investigation depend on the class of objects $K \subset X$ to be approximated, the distance measure (norm, metric), the chosen subspaces X_n and mappings.

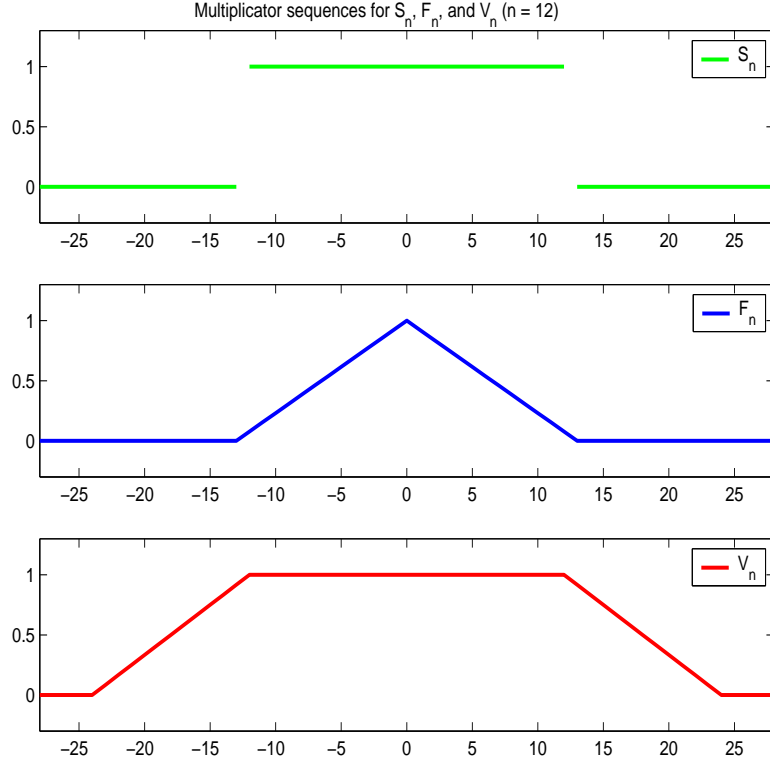


Figure 2: Multiplier sequences λ_n

- One of the central questions is the asymptotic convergence speed of an approximation method as $n \rightarrow \infty$ which is often compared to the best possible speed characterized by the corresponding best approximations. In this part, asymptotic results that are accurate within a constant or slightly growing factor are often acceptable.
- Another typical assumption in linear approximation theory is that the individual $f \in K$ to be approximated does not influence the choice of $\{X_n\}$ and mappings, those are uniformly applied to any $f \in K$. What has traditionally been done is to optimize the approximation method with respect to K (e.g., in our second example, with $K = X = C(0, 2\pi)$, the search was for a better summation method of reconstructing $f(t)$ from its Fourier coefficients). On an abstract level, this has led to concepts such as Kolmogorov widths which formalize the problem of finding the n -dimensional subspace $X_n \subset X$ which best fits $K \subset X$:

$$d_n(K, X) = \inf_{X_n: \dim X_n = n} \sup_{f \in K} \inf_{x_n \in X_n} \|f - x_n\|.$$

- Approximation theory overlaps with other areas such as real, complex, and functional analysis, the theory of numerical methods, optimization, and complexity

theory.

3 Introduction to nonlinear approximation

What is nonlinear approximation? Something which is not linear approximation! More seriously, there is no exact definition, and as of now nonlinear approximation theory consists of several classical and some more recent, more or less well-studied pieces. The early developments are reflected in the monograph [2].

There are two main differentiators. First, the ladder of linear subspaces $\{X_n\}$ is replaced by nonlinear sets $\{M_n \subset X\}$ parametrized by finitely many parameters (the container space X is, as a rule, still a linear space with a conventional metric or norm).

Example 3 ([3, 2, 8]). Rational approximation and approximation by exponential sums. These are classical examples, and given by the sets

$$M_{n,n'} = \left\{ r_{n,n'}(x) := \frac{\sum_{k=0}^{n'} b_k x^k}{\sum_{l=0}^n a_l x^l} \right\}$$

of rational functions (quotients of polynomials of degree $\leq n'$ in the numerator, and $\leq n$ in the denominator) resp.

$$M_n = \left\{ \sum_{k=1}^n c_k e^{\lambda_k x} \right\}$$

of exponential sums (useful in the modelling of mixtures of growth/decay processes), and

$$M_n = \left\{ \sum_{k=1}^n c_k e^{-(x-a_k)^2/(2\sigma_k^2)} \right\}$$

of sums of Gaussians (a typical ansatz for empirical density function approximation in statistics). See [3, Chapter 5], [2, Chapter V-VII], and [8, Chapter 7-9, 12] for results, history, and many examples that demonstrate how the additional degrees of freedom in these nonlinear M_n may help achieve much smaller approximation errors than the linear methods for algebraic resp. Fourier polynomials discussed in Section 2.

Often, the parameters which characterize M_n can be split into “nonlinear” and “linear” ones, and fixing the nonlinear parameters produces linear subspaces (i.e., the subjects of linear approximation theory). For instance, in the last example, the a_k (expectations) and σ_k (variances) which characterize the individual normal distributions, are the nonlinear parameters. If we would fix them beforehand, we would have to determine only the coefficients c_k (i.e., the appropriate linear combination of fixed basis functions) which are naturally interpreted as the linear parameters in M_n . Thus,

often nonlinear approximation processes can be viewed as nonlinearly parametrized families of linear approximation methods.

Secondly, the approximation method itself is a nonlinear mapping from K into M_n , i.e., the approximation $f_n \in M_n$ to f depends nonlinearly on $f \in K$. If we interpret $M_n = \cup_{(t_1, \dots, t_{k_n})} X_n(t_1, \dots, t_{k_n})$ in the above suggested way as a union of a family of linear subspaces depending on the nonlinear parameters t_1, \dots, t_{k_n} , then it appears as if by choosing appropriate $t_k = t_k(f)$, $k = 1, \dots, k_n$, we find linear approximation methods individually adapted to each f from the target set K . This interpretation provides a convenient approach to analyzing adaptive computational methods (such as grid adaptation for finite element/difference discretizations of differential equations), and has led to the theory of n -term approximation from dictionaries discussed in Section 4 below.

The following example deals with approximation by step functions which is the simplest case of another well established branch of nonlinear approximation theory, *free knot spline approximation* [2, Chapter VIII], [6, Section 12.8].

Example 4. Linear and nonlinear approximation with step functions.

Let

$$X_n \equiv X_n(\mathcal{P}_n) := \{h(x) = c_k, x \in I_k = [t_{k-1}, t_k), k = 1, m \dots, n\},$$

be the set of all step functions over a partition

$$\mathcal{P}_n = \{t_0 = 0 < t_1 < \dots < t_{n-1} < t_n = 1\}$$

of $[0, 1]$ into n subintervals (to be 100% precise, let us set $I_n = [t_{n-1}, t_n]$). It is convenient to simultaneously interpret a partition as a collection of mutually disjoint intervals $\mathcal{P}_n = \{I_k, k = 1, \dots, n\}$ with $\cup_k I_k = [0, 1]$, and to express step functions as linear combinations of characteristic functions:

$$h(x) = \sum_{k=1}^n c_k \chi_{I_k}(x), \quad \chi_I(x) = \begin{cases} 1, & x \in I, \\ 0, & x \in [0, 1] \setminus I. \end{cases}$$

This shows that for each fixed \mathcal{P}_n , the space X_n represents a n -dimensional linear subspace of step functions, while

$$M_n = \cup_{\mathcal{P}_n} X_n(\mathcal{P}_n) = \{h = \sum_{k=1}^n c_k \chi_{I_k}, I_k \cap I_l = \emptyset, k \neq l, \cup I_k = [0, 1]\}$$

is the nonlinear set of all step functions with $\leq n$ support intervals. The locations of the $n - 1$ interior knots t_k of the partition resp. the location of the n intervals I_k can be considered as the nonlinear parameters of M_n while the c_k are another n linear parameters.

Problem 4. While for linear subspaces we always have $\alpha X_n + \beta X_n = X_N$ (they are closed under linear operations), this is not true for M_n . Find $M_n + M_m = ?$.

How well can one approximate by using elements from this ladder of nonlinear sets $\{M_n\}$ versus some ladder of linear subspaces $\{X_n\}$ (obtained by fixing a sequence $\{\mathcal{P}_n\}$)? We will again take the norm $\|\cdot\|$ from Example 1 as the distance measure. In practical terms, this approximation problem is then about the construction of optimal look-up tables for function evaluation. For simplicity, let us take the subset K of all monotone continuous functions f on $[0, 1]$ with $f(0) = 0$, $f(1) = 1$, or even more concrete,

$$K' = \{f_\alpha(x) = x^\alpha, \quad 0 < \alpha < 1\} \subset K.$$

The functions in K' are smooth in $(0, 1]$ but have a singularity at the left endpoint (unbounded first derivative). Since unboundedness of some derivative near a particular point is a typical singularity in many applied problems, sets such as K' represent an important test case for the “quality assessment” of approximation processes. The first point we want to make is that fixing a partition sequence $\{\mathcal{P}_n\}$ beforehand, and using a linear approximation process based on the corresponding ladder $\{X_n\}$ will not work uniformly well for all of K' resp. K . E.g., let us take the sequence of uniform partitions

$$\mathcal{P}_n = \{0 < \frac{1}{n} < \dots < \frac{n-1}{n} < 1\}, \quad n \geq 1,$$

in the definition of X_n . Since on any (open or closed) interval $I = \langle a, b \rangle$, the best approximating constant to a continuous function is given by

$$c^*(f; I) = (\sup_{x \in I} f(x) + \inf_{x \in I} f(x))/2,$$

and thus for monotone continuous functions by

$$c^*(f; I) = (f(a) + f(b))/2,$$

for a given $f \in K$ estimates for the corresponding best approximations

$$E_n(f) := \inf_{h \in X_n} \|f - h\|$$

can easily be found. For an arbitrary \mathcal{P}_n , the resulting formula is

$$E_n(f) = \max_{k=1, \dots, n} |f(t_k) - f(t_{k-1})|/2, \quad f \in K, \quad n \geq 1,$$

and for the particular choice of uniform partitions and $f_\alpha \in K'$ we get

$$E_n(f_\alpha) = \frac{n^{-\alpha}}{2}, \quad n \geq 1.$$

Thus, with respect to the whole class K' resp. K , the convergence of any approximation process based on the ladder $\{X_n\}$ may be arbitrarily slow.

Problem 5. Verify the details! Also show that at least $E_n(f) \rightarrow 0$ for all $f \in K$.

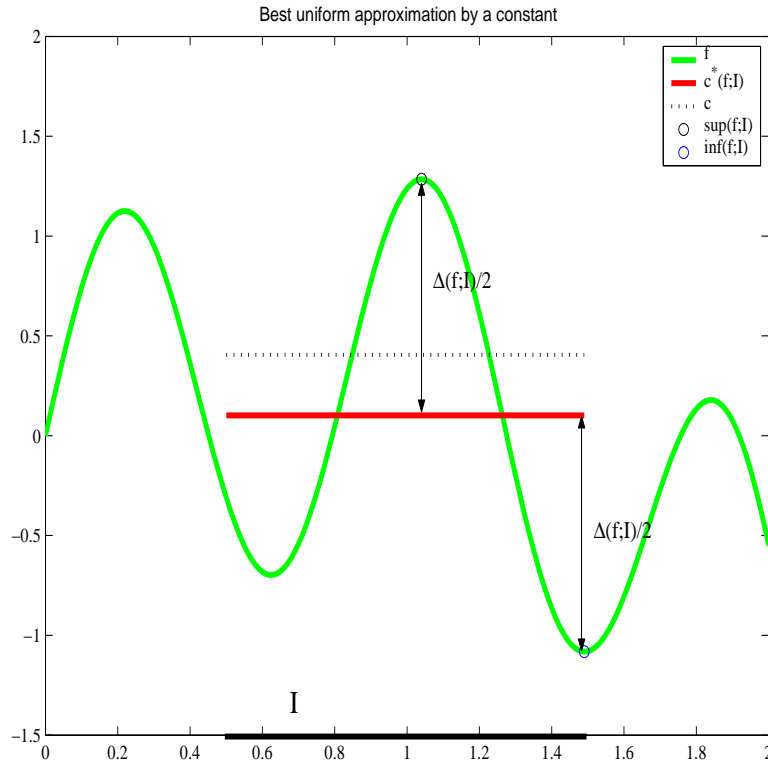


Figure 3: Best uniform approximation by constants

(Hint: Figure 3 illustrates the best approximation problem with constants on a single interval.)

Problem 6. You may argue that I chose a bad sequence of partitions \mathcal{P}_n (although there is no feeling of guilt on my part since uniform partitions are very natural in this context because they would lead to the most easy-to-use look-up tables). However, for any choice of $\{\mathcal{P}_n\}$ and any n , one can construct weird enough $f \in K$ such that $E_n(f) = 1$. But even for the smaller class K' and an arbitrarily fixed $\{\mathcal{P}_n\}$, one has $\sup_\alpha E_n(f_\alpha) = 1$. (Hint: Look at the the left-most interval of each \mathcal{P}_n !).

After all these sad news, here is a surprisingly positive result: If we allow to freely move the partition and adapt it to the individual function then for each $f \in K$ we have

$$\sigma_n(f) := \inf_{h \in M_n} \|f - h\| = \frac{1}{2n}, \quad n \geq 1.$$

See [2, 6] for related, more general results. Figure 4 illustrates how the optimal partition can be found for an arbitrary $f \in K$: Take $t_k = f^{-1}(k/n)$, $k = 1, \dots, n-1$ (if f is not strictly increasing then the inverse function f^{-1} is set-valued, and $=$ needs to be replaced by \in).

For the $f_\alpha \in K'$ the optimal partition is given by $t_k = (k/n)^{1/\alpha}$ (a so-called al-

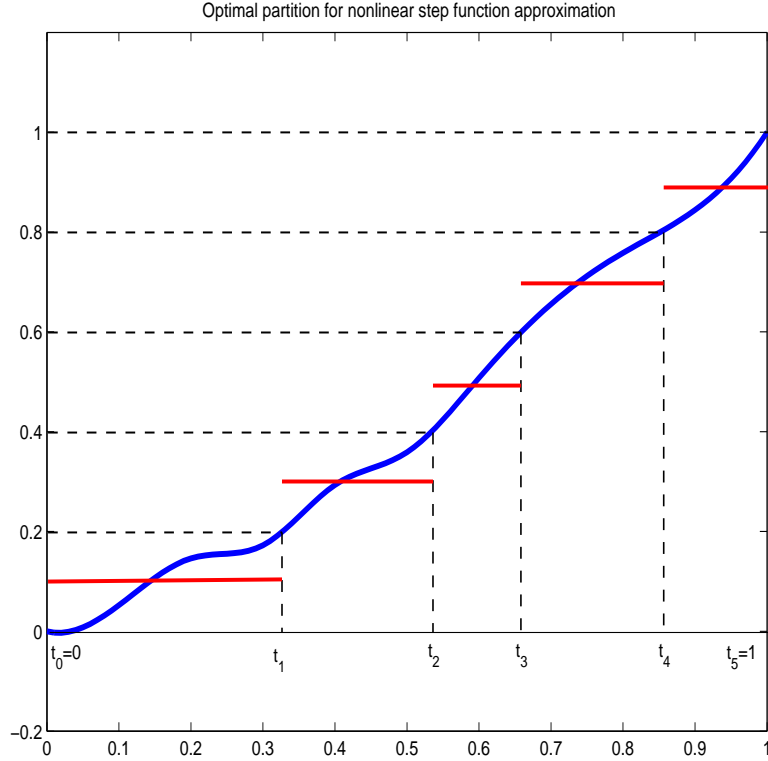


Figure 4: Optimal partition for monotone f

gebraically graded mesh) and α -dependent, as expected one needs more and smaller intervals near $x = 0+$ if α comes closer to 0 and the singularity becomes more severe. This example (and its obvious practical implications for the treatment of singularities in signal analysis and the solution of differential equations) has triggered a lot of research starting from the 70-ties. A powerful extension is to add more “nonlinear” degrees of freedom by allowing the local approximation on each interval to be a polynomial of fixed higher degree (instead of a constant) or even of arbitrary degree, whereas the degree may change from interval to interval, and only the sum of degrees (more precisely the number of linear degrees of freedom) is kept bounded by n . The latter became the basis of the hp -adaptive method introduced by Babuska et al., see [7] (h stands for step-size adaption, p for polynomial degree adaption), which is one of the most powerful but also complicated discretization approaches in use for partial differential and integral equation formulations in mathematical physics.

Problem 7. How fast can $f_{1/2}(x) = \sqrt{x}$, $x \in [0, 1]$, be approximated by
a) piecewise linear continuous functions (e.g., an interpolating polygon) with $\leq n$ pieces, or
b) with a piecewise polynomial function on m intervals where the degrees r_1, \dots, r_m of these polynomials satisfy $r_1 + \dots + r_m + m \leq n$?

Answers are expected to be qualitative, and given by estimates involving unspecified constants C , not by equalities (the latter is a hard problem, at least, for b)). (Hint: For a), the asymptotic speed of approximation is $\leq Cn^{-2}$ which is one order better than for piecewise constant functions. It is again achieved on a certain graded partition. For b), the speed is exponential: $\leq Cq^{-\sqrt{n}}$ for some $q < 1$. To prove this, use a partition which is geometrically refined towards $x = 0+$, e.g., $t_k = 2^{-(m-k)}$, and a linear distribution of degrees $r_k = k - 1$, $k = 1, \dots, m$, i.e., lower degree near the singularity, and higher degree away from it, and choose the appropriate $m \asymp \sqrt{n}$. Use local Taylor polynomial expansions and their remainder term expression to estimate the distance on each interval. If you don't like to do careful analytical estimations, don't try this problem!)

There is one little problem with the above approach: it is very hard to carry it over to the more interesting and practically needed case of functions of several variables. Already in two dimensions, the geometry description of an arbitrary partition of a domain can be arbitrarily complex, and even if we use triangulations into n triangles (or quadrilaterals or polygons etc.) finding their optimal location becomes a very hard optimization problem. This explains why after huge initial interest in variable partition/degree approximation with univariate splines it soon became quiet in this area of nonlinear approximation theory.

Things changed in the late 80-ies, when, on the one hand, wavelet methods and other discrete multiscale decomposition techniques conquered the imagination of theoretical and applied mathematicians (as well as non-mathematicians), and, on the other, it was realized that we do not need arbitrary partitions but only a countable subset of them to get the improvements over linear approximation processes based on fixed partition sequences.

Example 5. Nonlinear approximation with dyadic step functions. For approximating every $f_\alpha \in K'$ with step functions at the asymptotically optimal speed of $O(n^{-1})$, one can work with a “countable” subset of M_n :

$$M_{n,\mathcal{D}} := \left\{ h = \sum_{k=1}^n c_k \chi_{I_k}, \quad I_k \in \mathcal{D}, \quad I_k \cap I_l = \emptyset, \quad k \neq l, \quad \cup I_k = [0, 1] \right\},$$

where \mathcal{D} is the set of all dyadic intervals of the form $I_{j,i} := \langle (i-1)2^{-j}, i2^{-j} \rangle$, $i = 1, \dots, 2^j$, $j \geq 0$. At this point, it does not matter whether we have open, closed, or semi-open intervals, and we use the notation $\langle a, b \rangle$ for either of them. Let

$$\sigma_{n,\mathcal{D}}(f) := \inf_{h \in M_{n,\mathcal{D}}} \|f - h\| \geq \sigma_n(f)$$

denote the best approximation w.r.t. the nonlinear set $M_{n,\mathcal{D}} \subset M_n$. Note that \mathcal{D} is countable, and can be organized neatly into a dyadic tree (Figure 5) which mimics the way these intervals would be created in a real-world adaptive algorithm by consecutive refinement steps. Each node represents a dyadic interval, the edges connect a dyadic

interval of level j with its two “children” (the two dyadic half-intervals obtained by one local refinement step) of level $j + 1$ obtained by subdividing it.

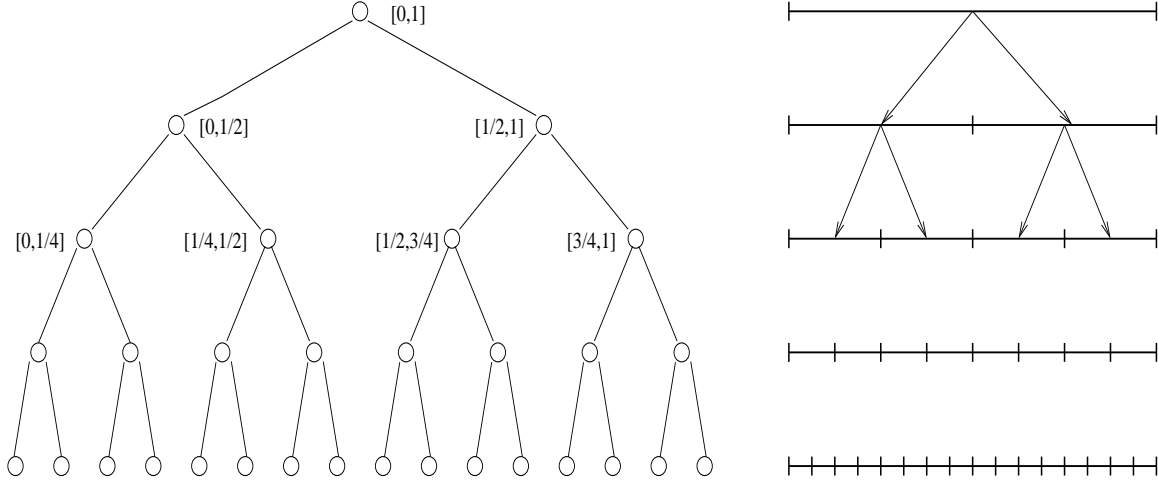


Figure 5: Refinement and tree structure of dyadic intervals

Here is the adaptive algorithm which can be used to produce a good approximation from $M_{n,\mathcal{D}}$, and prove the promised $O(n^{-1})$ estimate. To formulate it, define

$$\Delta(f; I) := \sup_{x \in I} f(x) - \inf_{x \in I} f(x)$$

for any interval I and any continuous f . For continuous and monotone f and $I = \langle a, b \rangle$, we obviously have $\Delta(f; I) = |f(b) - f(a)|$. Below, the letter \mathcal{I} always stands for a collection of dyadic intervals.

Algorithm for nonlinear approximation with dyadic step functions.

1. Initialization: Set $\mathcal{I}_1 = \{[0, 1]\}$, and $h_1 = c^*(f, [0, 1])\chi_{[0,1]}$.
2. Recursion: For $s = 1, \dots, n - 1$, pick a dyadic interval $I \in \mathcal{I}_s$ such that

$$\Delta(f; I) = m(f; \mathcal{I}_s) := \max_{I' \in \mathcal{I}_s} \Delta(f; I').$$

Then define \mathcal{I}_{s+1} by removing I from \mathcal{I}_s , and adding its two dyadic children I^\pm instead. Moreover, set

$$h_{s+1} := \sum_{I \in \mathcal{I}_{s+1}} c^*(f; I)\chi_I.$$

3. Output: Return h_n as the output step function.

Obviously, at each recursion stage \mathcal{I}_s represents a partition of $[0, 1]$ into s dyadic subintervals, and

$$\|f - h_s\| = m(f; \mathcal{I}_s)/2.$$

Thus, we always have

$$|\mathcal{I}_s|^{-1} = 2\sigma_{|\mathcal{I}_s|}(f) \leq 2\sigma_{s,\mathcal{D}}(f) \leq 2\|f - h_s\| = m(f; \mathcal{I}_s), \quad s \geq 1.$$

These observations also suggest that we could have stopped the recursion (Step 2 of the algorithm) not by inspecting the cardinality of \mathcal{I}_s but by checking the condition $m(f; \mathcal{I}_s) \leq \epsilon$, where ϵ is a given accuracy threshold. Either way, we still need to know how s and $m(f; \mathcal{I}_s)$ relate to each other.

To do this for $f = f_\alpha \in Z'$, let us observe that our above Algorithm is a *greedy* one: in each step, the interval I with the largest error indicator $\Delta(f; I)$ is eliminated and replaced by its children (note that the children's error indicators cannot be larger!). Thus, after we have finished, we have eliminated all dyadic intervals I with

$$\Delta(f; I) > m(f; \mathcal{I}_n),$$

and possibly some but not all with $\Delta(f; I) = m(f; \mathcal{I}_n)$. Thus, let us set $\epsilon := m(f; \mathcal{I}_n)$, and count how many dyadic intervals can satisfy $\Delta(f; I) > \epsilon$ resp. $\Delta(f; I) \geq \epsilon$. These two numbers, denoted by n_ϵ and n'_ϵ obviously sandwich our n : $n_\epsilon \leq |\mathcal{I}_n| = n < n'_\epsilon$.

The estimation for these two integers is the same, so we do this for n'_ϵ . Let us fix an arbitrary $j \geq 0$, and denote by $n_j(\epsilon)$ the number of all $I_{j,i}$ with this j and $\Delta(f_\alpha, I_{j,i}) \geq \epsilon$. By the concavity properties of $f_\alpha(x) = x^\alpha$, $0 < \alpha < 1$, we have

$$1 - (1 - 2^{-j})^\alpha = \Delta(f_\alpha, I_{j,2^j}) \leq \Delta(f_\alpha, I_{j,i}) \leq \Delta(f_\alpha, I_{j,1}) = 2^{-j\alpha}.$$

Thus, we have 3 types of j . The first group (small $j \leq j_0$) is characterized by

$$1 - (1 - 2^{-j})^\alpha \geq \epsilon \quad \implies \quad n_j(\epsilon) = 2^j,$$

and since j_0 (the largest such j) satisfies $2^{j_0} \asymp \alpha/\epsilon$, we have

$$\sum_{j \leq j_0} n_j(\epsilon) = 2^{j_0} - 1 \leq C \frac{\alpha}{\epsilon}$$

with some absolute constant C for all the j from the first group. The second group (large $j \geq j_1$) is defined by

$$2^{-j\alpha} < \epsilon \quad \implies \quad n_j(\epsilon) = 0,$$

and does not contribute to the overall count.

For the remaining intermediate range of j , we have $1 \leq n_j(\epsilon) < 2^j$. Looking closer, we see that for those j

$$\epsilon \leq \Delta(f_\alpha, I_{j,i}) = 2^{-j\alpha}(i^\alpha - (i-1)^\alpha) \leq C\alpha 2^{-j\alpha} i^{\alpha-1}$$

implies $n_j(\epsilon) \leq C((\alpha/\epsilon)2^{-j\alpha})^{1/(1-\alpha)}$. This upper bound is a geometric progression, and its sum is majorized by

$$\sum_{j_0 < j < j_1} n_j(\epsilon) \leq C \left(\frac{\alpha}{\epsilon}\right)^{1/(1-\alpha)} \sum_{j > j_0} 2^{-j\alpha/(1-\alpha)} \leq C \frac{1-\alpha}{\epsilon}.$$

To get the last estimation step done (with a constant independent of α), one has to be careful but it is true. Thus, altogether we arrive at

$$n < n'_\epsilon = \sum_{j=0}^{\infty} n_j(\epsilon) \leq \frac{C}{\epsilon} = \frac{C}{m(f_\alpha; \mathcal{I}_n)},$$

or, the other way around, at

$$m(f_\alpha; \mathcal{I}_n) \leq Cn^{-1}, \quad 0 < \alpha < 1, \quad n \geq 1.$$

Together with the lower estimate we already had, this is exactly what we wanted to prove, namely, that

$$\sigma_n(f) \asymp \sigma_{n, \mathcal{D}}(f) \asymp 1/n \quad \forall f = f_\alpha \in K',$$

where the constant factors appearing in the \asymp relation do not depend on α . The same result holds for wide classes of f (not for all $f \in K$ though, try to find counterexamples!).

The above approach has important other merits. It is constructive, at least on a theoretical level, and simple. The possibly complicated optimization w.r.t. the nonlinear parameters in M_n has been replaced by a simple greedy algorithm in a convenient data structure. Most importantly, the game can be played for any space dimension. E.g., in two dimensions one could repeatedly quadrisect starting from a suitable coarse initial partition, thus producing a quadtree structure with finitely many root nodes. Although the error indicators and counting arguments employed above may become a bit more involved, they still work. The first complete theoretical verification of this concept (the proof of characterization theorems for classes of multivariate functions defined by their speed of nonlinear approximation) has been given by DeVore, Jawerth, Popov (see [5] for some introduction to the subject) by using wavelet decompositions and coefficient thresholding rather than the above error indicators, but is based on the same simple ideas.

Summary:

- Nonlinear approximation is about choosing approximants from nonlinear sets resp. about adapting the approximation method to the individual $f \in K$.
- In some cases, the rate of approximation of a nonlinear method turns out to be superior over the best achievable rate of a comparable linear method (as was the case for step function approximation with fixed and variable partitions). To find these situations, and describe the exact conditions under which significant gains are possible, is one of the central problems in nonlinear approximation theory.

- Since nonlinear sets and mappings are more difficult to handle, the construction of simplified nonlinear approximation schemes and algorithms (such as the greedy algorithm of Example 5), their theoretical properties, and the practical details of reducing their complexity become topics of independent interest. One such framework, n -term approximation from dictionaries, will be discussed in the next section.

4 n -term approximation from dictionaries

Let us introduce the following language. As before, let X be a normed space with norm $\|\cdot\|$, and let $D = \{\phi_\lambda : \lambda \in \Lambda\} \subset X$ be a fixed set of elements of unit norm, $\|\phi_\lambda\| = 1$ for all $\lambda \in \Lambda$. When D is countable, we could take $\Lambda = \mathbb{N}$ after renumeration but prefer to leave the index notation in this more general form. We call D a *dictionary*, and ϕ_λ *dictionary elements* (or basis elements or atoms, if you like this more). The nonlinear approximation sets are the sets of n -term “polynomials” w.r.t. the dictionary D given by

$$M_{n,D} := \left\{ g_{\Lambda_n} = \sum_{\lambda \in \Lambda_n} c_\lambda \phi_\lambda : |\Lambda_n| \leq n \right\},$$

i.e., n -term polynomials are linear combinations of dictionary elements with $\leq n$ non-zero elements. Finally, define the sequence of best nonlinear n -term approximations in X associated with D by

$$\sigma_{n,D}(f) := \inf_{g \in M_{n,D}} \|f - g\|, \quad n \geq 1.$$

E.g., the set $M_{n,\mathcal{D}}$ in Example 5 is a proper subset of $M_{n,D}$ if we take the dictionary $D := \{\chi_I : I \in \mathcal{D}\}$.

The main questions one wants to ask are:

- Are there enough good examples for these notions which would motivate the study of nonlinear approximation processes associated with $M_{n,D}$?
- Can one prove meaningful results for the nonlinear n -term best approximations $\sigma_{n,D}(f)$, e.g., find (and eventually characterize) the classes $K \subset X$ for which these quantities decay in a prescribed manner? How does this fare in comparison with appropriate linear methods (e.g., w.r.t. the ladders of subspaces $\{X_n = \text{span}(\phi_\lambda : \lambda \in \Lambda_n)\}$ obtained by fixing appropriate sequences of index sets $\Lambda_n \subset \Lambda$, $|\Lambda_n| = n$, $n \geq 1$)?
- Are there simple algorithms that would result in efficient nonlinear approximation methods associated with $\{M_{n,D}(f)\}$, and how do they compare with the lower bounds obtained from $\sigma_{n,D}(f)$ resp. with the approximation rate of comparable linear methods?

- Finally, can one demonstrate that this is more than a theoretical concept, by relating it to research in optimal design, data and signal compression, adaptive solution methods, etc., and by achieving significant “savings” in those application areas?

Although one can find partial results on all these questions in the literature, there is no way that I can exhaustively answer them here. What I will do is give some more examples of various dictionaries D , touch some aspects of the raised questions, and hopefully increase your interest in this subject.

Example 6. Singular value decomposition. Let X be the set of all complex $N \times N$ matrices $A = ((a_{kl}))$ equipped with the Frobenius norm:

$$\|A\| := \left(\sum_{k,l=1}^N |a_{kl}|^2 \right)^{1/2}.$$

As dictionary we take the set of all rank-1 matrices of unit Frobenius norm:

$$D = \{xy^* = ((x_k y_l^*)) : x, y \in \mathbb{C}^N, \|x\| = \|y\| = 1\}.$$

Here, elements of \mathbb{C}^N are interpreted as column vectors. Since any $N \times N$ matrix can be represented by a linear combination of at most N rank-1 matrices, we have $\sigma_{n,D}(A) = 0$ for all $n \geq N$. For the remaining $1 \leq n < N$, we have the following formula:

$$\sigma_{n,D}(A)^2 = \sum_{k=n+1}^N s_k(A)^2,$$

where $s_k(A)$ denotes the singular values of A in decreasing order (note that $\{s_k(A)^2\}$ coincides with the set of eigenvalues of the symmetric non-negative definite matrix AA^*). This follows from the singular value decomposition (SVD)

$$A = XSY^* = \sum_{k=1}^N s_k(A)x^{(k)}(y^{(k)})^*$$

of an arbitrary square matrix A into the product of a unitary matrix X , a diagonal matrix $S = \text{diag}(s_k(A), k = 1, \dots, N)$ containing the singular values, and the complex transpose of another unitary Y of the same dimension. In this expression, the $x^{(k)}$, $k = 1, \dots, N$, denote the column vectors of X , similarly for Y .

Problem 8. Prove the above formula! (Hint: It is enough to prove the case $n = 1$, just use the unitarity of X, Y , and that the columns of a unitary matrix form an orthonormal basis in \mathbb{C}^N , the rest goes by induction, and if you look closely you will see the greedy algorithm for computing the SVD. You may also consult a textbook on numerical linear algebra.)

This example shows that seemingly unrelated problems of linear algebra can be formulated via n -term approximation theory. It also has an applied aspect: The SVD is often used to compress matrices/streams of vector data, so the above reformulation suggests a way of quantifying the compression/quality tradeoffs. The continuous analogon (approximation of functions $f(x, y)$ of two variables on a rectangle by linear combinations of products $g(x)h(y)$ of functions of one variable) has been considered by E. Schmidt (1916), and is central to the celebrated theory of Hilbert-Schmidt (integral) operators.

Example 7. Spherical codes. This is another finite-dimensional example, with background in mathematical coding theory (or geometric optimization, if you wish). A code of length N can be considered as a collection of N codewords

$$\mathcal{C} = [x^{(0)}, \dots, x^{(N-1)}].$$

The idea is to encode a given message (bitstream) by means of these codewords, and send them out over a transmission channel (wire, optical fiber, air, ...) which distorts them. At the receiver end, there is a detection and decoding effort which matches the detected but distorted codeword to the closest exact codeword, and then returns information in readable format (hopefully, something identical or very close to the original message). Usually, something can be assumed or learned about the channel which leads to a certain probabilistic channel model, and allows to tie the error probability (i.e., the chances that codeword $x^{(k)}$ was sent but so much distorted that it was decoded as $x^{(l)}$, $l \neq k$) to some distance between the codewords. Now, this already looks like approximation theory!

Without going into further explanations, let us proceed to a special case. Think of a codebook that consists of complex unit vectors of length $n \ll N$, i.e., $x^{(k)} \in \mathbb{C}^n$ and $\|x^{(k)}\| = 1$ for all $k = 0, \dots, N-1$. A practical distance measure turns out to be the angle $0 \leq \alpha_{kl} = \angle(x^{(k)}, x^{(l)}) \leq \pi/2$ between two codewords described by

$$\cos(\alpha_{kl}) := |(x^{(k)}, x^{(l)})| = \left| \sum_{s=1}^n x_s^{(k)} (x_s^{(l)})^* \right|.$$

So, intuitively, constructing a good code means to find the unit vectors $x^{(k)}$ such that

$$\min_{k \neq l} \alpha_{kl} \longmapsto \max \iff \max_{k \neq l} |(x^{(k)}, x^{(l)})| \longmapsto \min$$

for given $n \ll N$. Check that this has to do with the geometrical problem of finding the largest $d > 0$ such that packing N spherical caps of diameter d without overlap onto the unit sphere of \mathbb{C}^n is possible (optimal packing).

Here is an explicit construction of a code-candidate: Take n real numbers $r_k > 0$ such that $\sum_k r_k = 1$, select n integers $0 \leq m_1 < m_2 < \dots < m_n < N$, and define the

N codewords by

$$x^{(k)} := \begin{bmatrix} \sqrt{r_1} e^{i2\pi m_1 k/N} \\ \sqrt{r_2} e^{i2\pi m_2 k/N} \\ \vdots \\ \sqrt{r_n} e^{i2\pi m_n k/N} \end{bmatrix}, \quad k = 0, \dots, N-1.$$

This is a very specific example of an algebraic code (associated with a cyclic group) which has certain advantages when implementing the decoder. How should I choose the r_k and n_k such that this proposed code is good in the sense explained above?

This can be treated as a nonlinear, mixed-integer optimization problem but I want to show its connection to n -term approximation. Introduce the auxiliary complex Fourier polynomial

$$P(t) = \sum_{s=1}^n r_s e^{im_s t} \in X := \left\{ \sum_{m=0}^{N-1} c_m e^{imt} : c_m \in \mathbb{R} \right\}.$$

This is convenient since the distance properties of the code translate into properties of $P(t)$. Indeed,

$$(x^{(k)}, x^{(l)}) = \sum_{s=1}^n \sqrt{r_s} e^{i2\pi m_s k/N} \sqrt{r_s} e^{-i2\pi m_s l/N} = P(2\pi(k-l)/N) \quad \forall k, l.$$

For $k = l$, we get the normalization condition since $P(0) = \sum_s r_s = 1$. Thus, the best code results if the polynomial $P(t)$ solves the minmax problem

$$\max_{k=1, \dots, N-1} |P(2\pi k/N)| \longmapsto \min$$

on the class of all n -term polynomials (with $P(0) = 1$) from the above N -dimensional container space X of Fourier polynomials.

Finally, observe that the polynomial

$$F(t) := \frac{1}{N} \sum_{m=0}^{N-1} e^{imt} = \frac{1 - e^{iNt}}{N(1 - e^{it})} \in X$$

(the complex Dirichlet kernel) satisfies $F(0) = 1$ and $F(2\pi k/N) = 0$ for all $k = 1, \dots, N-1$, and would generate the perfect code if it were an n -term polynomial, and not an N -term polynomial. With this in mind, here is the matching dictionary/ n -term approximation formulation of our code problem. Consider the above set X of Fourier polynomials, and define the norm by

$$\|P\| := \max_{k=0, \dots, N-1} |P(2\pi k/N)|$$

Problem 9. Verify the norm axioms!

As dictionary we take the basis $D = \{e^{imt} : m = 0, \dots, N-1\}$ of complex exponentials in X . Then finding the best nonlinear n -term approximation $\sigma_{n,D}(F)$ to the Dirichlet kernel $F \in X$ comes very close to solving our best code problem! “Comes very close” and not “is equivalent” because we also need to satisfy the constraints on the coefficients r_k . Nevertheless, the proposed language can be used to attack the qualitative part of code construction (how much error probability will result for which pairs of n, N ?). As far as I know, the problem of finding sharp estimates for $\sigma_{n,D}(F)$ in terms of n, N is not completely solved for the intermediate range of $\log N \ll n \ll N^{1/2}$, and represents a very hard problem with connections to other unsolved problems related to the geometry of finite-dimensional spaces.

Example 8. Orthonormal basis dictionaries. Let $X = H$ be a (complex or real) Hilbert space, with scalar product (f, g) and norm $\|f\| = \sqrt{(f, f)}$. For simplicity, we assume that H is separable (or finite-dimensional). We already had several examples of those, e.g., the space of square-integrable periodic functions $L_2(0, 2\pi)$ of Example 2, or the finite-dimensional coordinate spaces $\mathbb{R}^N, \mathbb{C}^N$, where $(x, y) = y^*x$. Let $D = \{e_k\}$ be a complete orthonormal system (CONS) in H , where the index set is \mathbb{N} (or a finite section of \mathbb{N}). This means that $(e_k, e_l) = 0$ for all $k \neq l$ and $(e_k, e_k) = \|e_k\|^2 = 1$ for all k . By the Riesz-Fischer theorem, every $f \in H$ can be written as

$$f = \sum_k c_k(f) e_k, \quad c_k(f) := (f, e_k),$$

and

$$\|f\|^2 = \sum_k |c_k(f)|^2.$$

This allows to investigate approximation processes related to the pair (H, D) in terms of the coefficient sequences $\{c_k(f)\}$. In particular, the nonlinear n -term approximation problem can be studied in a straightforward way.

Here are some simple facts. First, the best n -term approximation from a CONS D in a Hilbert space can be realized by the following “theoretical” greedy algorithm:

Algorithm for n -term approximation from an orthonormal basis.

1. Initialization: Set $\Lambda_0 = \emptyset$, and $r_0 = 0$.
2. Recursion: While $s < n$, compute all $c_k(r_s)$, and choose the index \hat{k} with the largest in absolute value computed coefficient, i.e.,

$$|c_{\hat{k}}(r_s)| = \max_k |c_k(r_s)|.$$

Set $\Lambda_{s+1} = \Lambda_s \cup \{\hat{k}\}$ and $r_{s+1} = r_s - c_{\hat{k}}(r_s) e_{\hat{k}}$.

3. Output: For $g_n := \sum_{k \in \Lambda_n} c_k(f) e_k = f - r_n$ we have

$$\|f - g_n\| = \|r_n\| = \sigma_{n,D}(f) = \left(\sum_{k \notin \Lambda_n} |c_k(f)|^2 \right)^{1/2}.$$

The algorithm is “theoretical” in the separable case since without having some kind of a priori information (an “oracle”), we cannot compute infinitely many $c_k(r_s)$ resp. safely find the index with the maximal value of $|c_k(r_s)|$!

One may also write down a closed expression. Let $\{c_k^*(f)\}$ be the decreasing rearrangement of $\{c_k(f)\}$ which is defined as the sequence of real positive numbers obtained by sorting the sequence $\{|c_k(f)| > 0\}$ in decreasing order (zero coefficients can be neglected, and if $\{|c_k(f)| > 0\}$ is finite then zeros are appended to $\{c_k^*(f)\}$ as necessary). Note that the sort (as well as the maximum search in the recursion step 2 of the above algorithm) are well-defined since $c_n(f) \rightarrow 0$, and can be described by an index mapping $k \rightarrow n_k$, where $n_k \neq n_l$ for $k \neq l$, such that $\{n_k\}$ contains all indices with non-zero coefficients and

$$c_k^*(f) = |c_{n_k}(f)|, \quad |c_{n_1}(f)| \geq |c_{n_2}(f)| \geq \dots$$

This index mapping is not unique if several $c_k(f) \neq 0$ have the same absolute value but any choice is fine. E.g.,

$$\{c_k(f)\} = \{1/2, 1, -1, 0, 1/4, 1/3, -1/3, 0, 1/6, 1/5, -1/5, 0, \dots\}$$

could lead to

$$\begin{aligned} \{c_k^*(f)\} &= \{1, 1, 1/2, 1/3, 1/3, 1/4, 1/5, 1/5, 1/6, \dots\}, \\ \{n_k\} &= \{2, 3, 1, 7, 6, 5, 10, 11, 9, \dots\}. \end{aligned}$$

With this definition of $\{c_k^*(f)\}$ at hand, we have

$$\sigma_{n,D}(f) = \|f - \sum_{k=1}^n c_{n_k}(f) e_k\| = \left(\sum_{k>n} c_k^*(f)^2 \right)^{1/2}.$$

Problem 10. Verify these formulas (and that our greedy algorithm gives the same result) by using the expression for $\|f\|^2$ in terms of the coefficients $c_k(f)$. (Hint: Prove that if g_{Λ_n} is a n -term polynomial w.r.t. a CONS D , then $\|f - g_{\Lambda_n}\|^2 \geq \sum_{k \notin \Lambda_n} |c_k(f)|^2$, and figure out when equality holds.)

Secondly, some meaningful theorems about the convergence speed can be proved. E.g., the following equivalence holds: If D is a CONS in H then

$$\sigma_{n,D}(f) = O(n^{-\alpha}), \quad n \rightarrow \infty \iff c_k^*(f) = O(k^{-(\alpha+1/2)}), \quad k \rightarrow \infty.$$

In one direction, this is trivial since substituting the assumption $c_k^*(f) \leq Ck^{-(\alpha+1/2)}$ into the already known formula for the best nonlinear n -term approximations yields

$$\sigma_{n,D}(f)^2 \leq C^2 \sum_{k>n} k^{-2\alpha-1} \leq C' n^{-2\alpha}.$$

In the other direction, we will use the same error formula together with the monotonicity of the decreasing rearrangement $\{c_k^*(f)\}$:

$$Cn^{-2\alpha} \geq \sigma_{n,D}(f)^2 \geq \sum_{k=n+1}^{2n} c_k^*(f)^2 \geq nc_{2n}^*(f)^2,$$

which gives $c_{2n+1}^*(f)^2 \leq c_{2n}^*(f)^2 \leq Cn^{-2\alpha-1}$.

The above proved result is a simple necessary and sufficient condition for characterizing the $f \in H$ which have a prescribed speed of nonlinear approximation. It has been instrumental for the theory of nonlinear wavelet compression schemes but is not very constructive because telling something about $\{c_k^*(f)\}$ is even harder than telling something about $\{c_k(f)\}$. Here is a bit less optimal result, a sufficient condition in terms of $\{c_k(f)\}$: If D is a CONS in H and

$$f \in A_\tau(D) := \{f \in H : \sum_k |c_k(f)|^\tau < \infty\}$$

for some $0 < \tau < 2$ then

$$\sigma_{n,D}(f) = O(n^{-(1/\tau-1/2)}), \quad n \rightarrow \infty.$$

Indeed, from the error formula and definition of $\{c_k^*(f)\}$ we get

$$\sigma_{n,D}(f)^2 = \sum_{k>n} c_k^*(f)^2 \leq c_n^*(f)^{2-\tau} \sum_{k>n} c_k^*(f)^\tau \leq c_n^*(f)^{2-\tau} \sum_k |c_k(f)|^\tau \leq Cc_n^*(f)^{2-\tau}$$

where C depends on $f \in A_\tau(D)$, $0 < \tau < 2$. But by the same token,

$$nc_n^*(f)^\tau \leq \sum_{k \leq n} c_k^*(f)^\tau \leq \sum_k |c_k(f)|^\tau \leq C,$$

which implies $c_n^*(f)^\tau = O(n^{-1})$ or

$$c_n^*(f)^{2-\tau} = O(n^{-(2/\tau-1)}) = O(n^{-2(1/\tau-1/2)}), \quad n \rightarrow \infty.$$

This proves the statement which is more appealing since $f \in A_\tau(D)$ can often be tied to other, more easily verifiable conditions on $f \in H$ or more classical concepts. E.g., applied in the context of Example 2, the case $\tau = 1$ corresponds to the absolute convergence of Fourier series.

Some of the simple results in Example 8 have nontrivial extensions to *arbitrary dictionaries* $D = \{\phi_\lambda\}$ in a Hilbert space, see [5, 9]. From a theoretical and applied point of view, *redundant dictionaries* where the ϕ_λ do not represent a set of linearly independent directions in H , or extensions to *dictionaries in arbitrary normed spaces* X where the distance measure is not given by a scalar product and orthogonality has

no distinctive meaning, are of interest and present new challenges. I will conclude with an example which is a variation of Example 5.

Example 9. The Haar wavelet system and uniform approximation. The Haar system $\{H_m, m \geq 1\}$ is a CONS in $L_2(0, 1)$ consisting of dyadic step functions which was introduced by A. Haar (1909). After the trigonometric system (or system of Fourier exponentials, see Example 2), it is possibly the second famous, classical system of functions. We will discuss it here as a tool to approximate continuous functions, although at first glance this is a bit crazy because the linear combinations of Haar functions are discontinuous step functions (by the way, Haar had no problems with this “inconsistency”, as his fundamental result said that any $f \in C(0, 1)$ can be uniquely represented by a uniformly converging series w.r.t. to $\{H_m\}$, with coefficients that are easy to compute, see below). Therefore we present the Haar functions with the normalization appropriate for $C(0, 1)$.

Recall from Example 5, that \mathcal{D} is the set of all dyadic intervals. To avoid any problems, from now on all dyadic intervals will be taken of the form $[a, b)$ (i.e., closed to the left, and open to the right), except when $b = 1$, where they will be closed. If $I \in \mathcal{D}$, then we denote by I^\pm its left/right children, i.e., in the notation introduced in Example 5 we have $I_{j,i}^- = I_{j+1,2i-1}$, $I_{j,i}^+ = I_{j+1,2i}$ for all $i = 1, \dots, 2^j$, $j \geq 0$. Set $H_1 := \chi_{[0,1]}$, and $H_m := H_{I_{j,i}}$ for $m = 2^j + i > 1$, where $i = 1, \dots, 2^j$, $j \geq 0$ (throughout this example, and without further mentioning, $m > 1$ will be connected with j, i in this fashion), and

$$H_I(x) := \chi_{I^-}(x) - \chi_{I^+}(x) = \begin{cases} 1, & x \in I^-, \\ -1, & x \in I^+, \\ 0, & x \notin I, \end{cases} \quad I \in \mathcal{D}.$$

For any $f \in C(0, 1)$, we define

$$c_I(f) := |I|^{-1} \int_0^1 f(t) H_I(t) dt = |I|^{-1} \left(\int_{I^-} f(t) dt - \int_{I^+} f(t) dt \right),$$

and set

$$c_1(f) := \int_0^1 f(t) dt, \quad c_m(f) = c_{I_{j,i}}(f), \quad m > 1.$$

Problem 11. Using the equicontinuity of $f \in C(0, 1)$, verify that $c_m(f) \rightarrow 0$ as $m \rightarrow \infty$.

Motivated by the previous Example 9, let us take $X = C(0, 1)$, $D = \{H_m : m \geq 1\}$, and ask whether the same greedy construction gives good results, i.e., whether

$$\|f - \sum_{m \in \Lambda_n} c_m(f) H_m\| = \|f - \sum_{k=1}^n c_{n_k}(f) H_{n_k}\| \approx \sigma_{n,D}(f), \quad n \rightarrow \infty?$$

Here Λ_n is produced by the same algorithm (but with the $c_k(g_s)$ computed on the basis of Haar functions), and $\{n_k\}$ by the same rearrangement procedure for $\{c_k(f)\}$

as above. Unfortunately, this is not true. To construct a counterexample, look at the $2n$ -term Haar polynomial

$$G = (H_1 + \sum_{j=0}^{n-2} H_{2^{j+1}}) + \sum_{i=0}^{n-1} (1 + \epsilon) H_{2^{n+1-i}} \equiv G_1 + G_2, \quad \epsilon > 0,$$

which we have represented as the sum of two n -term Haar polynomials G_1 and G_2 . If $n \geq 2$ then G_1 is supported in $[0, 1/2)$ while G_2 is supported in $[1/2, 1]$. Figure 6 shows the graph of this step function for $n = 3$.

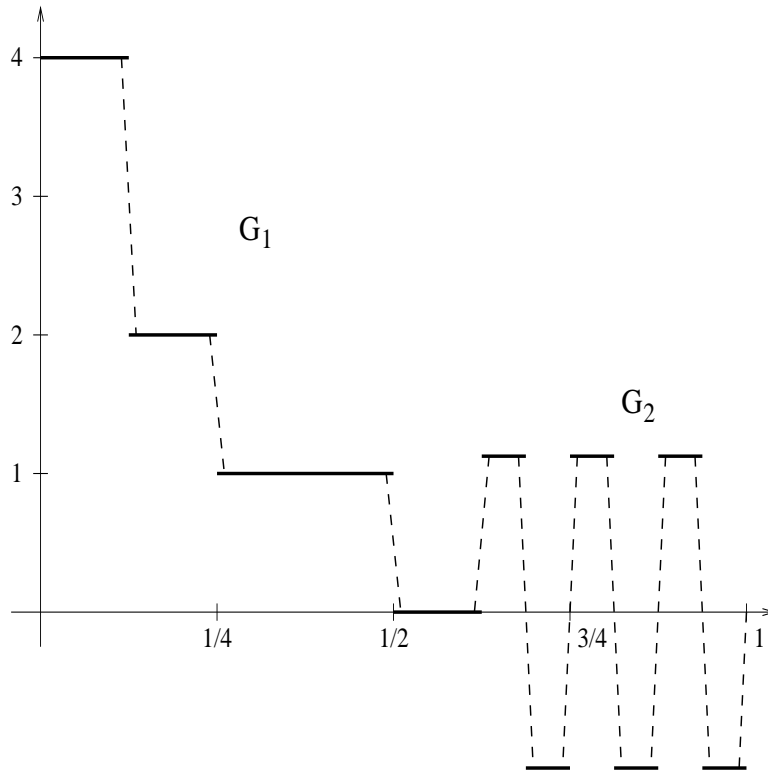


Figure 6: The counterexample $G(x)$ for $n = 3$

We will look at G as our candidate for n -term approximation, and show that if $\epsilon \rightarrow 0$ the result of the greedy procedure may be at least by a factor n worse than the best n -term approximation (the fact that G is not continuous does not matter too much; if one smoothes the step function G to a continuous, piecewise linear function in the way shown in Figure 8 by the dashed lines and makes the slopes of these lines very stiff, the conclusion will be the same). First of all, by definition of G we have n Haar coefficients $c_m(f)$ with value $1 + \epsilon$, another n with value 1, the rest vanishes. Therefore, after n steps the greedy algorithm has picked all terms with coefficients $1 + \epsilon$, and returned G_2 as the n -term polynomial of choice. Thus, the error of the

greedy approximation is

$$\|G_1\| \geq G_1(0) = H_1(0) + H_2(0) + \dots + H_{2^{n-2}+1}(0) = n.$$

But if we choose the n -term polynomial G_1 instead, the error would be $\|G_2\| = 1 + \epsilon$, which means that $\sigma_{n,D}(G) \leq 1 + \epsilon$. Thus, the error of the greedy algorithm (after n steps) may indeed be worse by at least a factor of n (to this end, let $\epsilon \rightarrow 0$).

The example shows that what was good for CONS in a Hilbert space (note that the properly scaled system of Haar functions is indeed a CONS in $L_2(0, 1)$ to which the discussion in Example 8 would apply) may not be good in non-Hilbert spaces resp. for distances that are far from Euclidean ones. This phenomenon has been investigated a lot, see [9].

In the particular example of the Haar dictionary D for approximation in $C(0, 1)$ there is the following remedy: instead of looking at the size of the coefficients $|c_k(f)|$ we could take other error indicators associated with the Haar functions, in the hope that things improve. E.g., in the spirit of Example 5, we could run the following algorithm (some of its properties are discussed in the problems formulated below):

Algorithm for uniform approximation with Haar functions.

1. Initialization: Set $\mathcal{I}_1 = \{[0, 1]\}$, and $g_1 = c_1(f)H_1$, $r_1 = f - g_1$.
2. Recursion: For $s = 1, \dots, n - 1$, pick the dyadic interval $I \in \mathcal{I}_s$ where the supremum in the definition of $\|r_s\|$ is attained,

$$\sup_{x \in I} |r_s(x)| = \|r_s\|.$$

Then define \mathcal{I}_{s+1} by removing I from \mathcal{I}_s , and adding its two dyadic children I^\pm . Moreover, set $g_{s+1} = g_s + c_I(f)H_I$, $r_{s+1} = f - g_{s+1}$.

3. Output: Return g_n as the output step function,

Problem 12. Show that this algorithm returns a result similar to the algorithm in Example 5, namely,

$$\|f - g_n\| \asymp \|f - h_n\|, \quad n \geq 1, \quad f \in C(0, 1).$$

(Hint: Observe that the function g_s is constant on each interval $I \in \mathcal{I}_s$, and that this constant value equals the average of f on I . Thus, if we define

$$\Delta'(f; I) := \sup_{x \in I} |f(x) - |I|^{-1} \int_I f(t) dt|$$

then Step 2 means picking the $I \in \mathcal{I}_s$ with the largest value of $\Delta'(f; I)$. Since $\Delta(f; I) \leq \Delta'(f; I) \leq 2\Delta(f; I)$, where $\Delta(f; I)$ was defined in Example 5, this allows for a comparison of the two algorithms.)

Problem 13. Do we also have $\|f - g_n\| \asymp \sigma_{n,D}(f)$ for all $f \in C([0, 1])$? Or $\sigma_{n,D}(f) \asymp$

$\sigma_{n,\mathcal{D}}(f)$? (Hint: Look at $f = H_1 + H_{2^{n+1}}$ for large n .)

Let us finally note that if we organize the Haar functions into a dyadic tree (as we did with the set \mathcal{D} of dyadic intervals in Example 5) then the selection of Haar functions according to the above algorithm amounts again to selecting a finite subtree. Tree-structured selections are natural and sometimes advantageous from a practical point of view because they are closer to the recursive organization of actual computations with Haar and more general wavelet decompositions. Thus, the lack of optimality of the algorithm for some weird f may not be such a bad thing if the ramifications of the computational world are taken into account. On the other hand, the error indicators used in the algorithms of Example 5 resp. 9 are hardly effective (since they would amount to solving optimization problems for each newly created interval), and need to be replaced by computationally less expensive indicators.

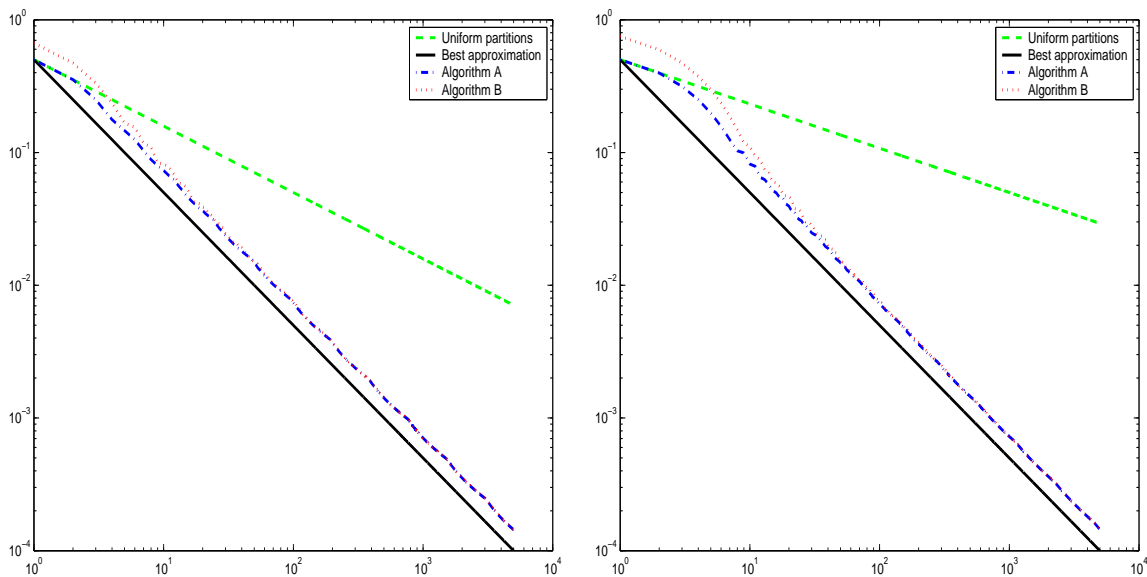


Figure 7: Theoretical implementation of greedy algorithms

In the concluding Figures we provide some numerical evidence on the above algorithm (Algorithm B), the algorithm from Example 5 (Algorithm A), in comparison to what we would get for the sequence of uniform partitions and the optimal partitions into n subintervals (see Example 4). Figure 7 gives the error curves in logarithmic scale for $1 \leq n \leq 5000$ when the methods are applied to $f_{1/2}(x) = \sqrt{x}$ (on the left) resp. to $f_{1/3}(x) = \sqrt[3]{x}$ (on the right). The errors of the greedy algorithms stay within a constant factor of the best possible result (as has been claimed) while the rate of approximation on uniform grids deteriorates with the singularity exponent α .

The graphs of Figure 8 show what happens if in the decision-making of the greedy algorithms A and B the maximum, minimum, and average values of f with respect to a dyadic interval I are numerically obtained by randomly selecting a fixed number

of points $x_1, \dots, x_L \in I$, computing $y_l = f(x_l)$, and then assuming the approximate values

$$\sup_{x \in I} f(x) \approx \max_l y_l, \quad \inf_{x \in I} f(x) \approx \min_l y_l, \quad |I|^{-1} \int_I f(x) dx \approx \frac{1}{L} \sum_l y_l.$$

With these modifications, the two algorithms can be performed for arbitrary continuous f , and only need constant (but proportional to L) amount of work per recursion step. In Figure 8, the resulting error curves are shown for $f_{1/3}$ and the choices $L = 5$, $L = 10$, and $L = 20$ (from left to right). Although the quality is worse than for the exact execution of the algorithms (compare the graph shown on the right in Figure 7), asymptotically we still get the optimal $O(n^{-1})$ rate and stay well below the result for the uniform partition sequence.

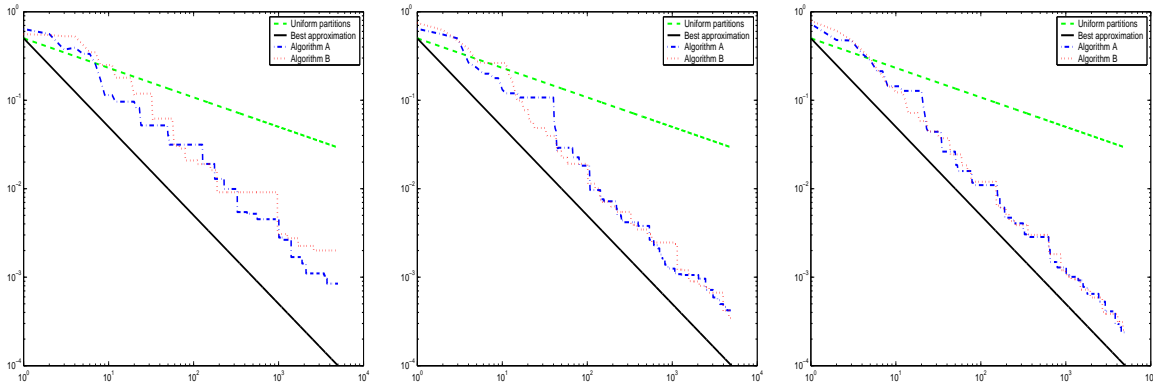


Figure 8: Numerical implementation of greedy algorithms

Summary.

- Nonlinear n -term approximation from dictionaries is a branch of nonlinear approximation theory where an approximation is individually adapted to $f \in K$ by picking the best n dictionary elements, and then building a linear projection into the subspace spanned by them. The simple framework covers a range of potential applications, from optimal design to adaptive algorithms based on error indicators.
- Because dictionaries are often discrete or countable, discrete algorithms and their properties become an integral part of the investigation into qualitative aspects such as the achievable rate of nonlinear approximation.
- In the future, a likely trend will be to work with huge and dynamically changing target sets K for which a model is not yet available or still fuzzy (such is the application area of learning theory). Then the choices for distance measures

in X (and X itself) and dictionaries adapted to these K and X , with good compression capabilities, are not given in advance but become design variables. Hopefully, some background in nonlinear approximation theory will prove helpful in such explorations.

References

- [1] N. K. Bary, *A Treatise on Trigonometric Series*, Macmillan, New York, 1964.
- [2] D. Braess, *Nonlinear Approximation Theory*, Springer, Berlin, 1986.
- [3] E. W. Cheney, *Introduction to Approximation Theory*, McGraw-Hill, New York, 1966.
- [4] W. Dahmen, Wavelet and multiscale methods for operator equations, *Acta Numerica* (1997), 55–228.
- [5] R. A. DeVore, Nonlinear Approximation, *Acta Numerica* (1998), 51–150.
- [6] R. A. DeVore, G. G. Lorentz, *Constructive Approximation*, Springer, Berlin, 1993.
- [7] M. Gui, I. Babuska, The h , p , and h - p version of the finite element method in 1 dimension. Part I-III, *Numer. Math.* 49 (1986), I: 577-612, II: 613–657, III: 659-683.
- [8] G. G. Lorentz, M. v. Golitschek, Yu. Makovoz, *Constructive Approximation, Advanced Problems*, Springer, Berlin, 1996.
- [9] V. N. Temlyakov, Nonlinear methods of approximation, IMI-Preprint 09-01, Univ. South Carolina, 2001.
- [10] M. F. Timan, *Theory of approximation of functions of a real variable*, MacMillan, New York, 1963.
- [11] A. Zygmund, *Trigonometric Series I-II*, Cambr. Univ. Press, Cambridge UK, 1959.