Approximation Theory and Applications

Christoph Ortner
ortner@math.ubc.ca
January 5, 2022

Contents

1 Introduction

{sec:intro}

> In mathematics, approximation theory is concerned with how functions can best be approximated with simpler functions, and with quantitatively characterizing the errors introduced thereby. Note that what is meant by best and simpler will depend on the application. (Wikipedia)

Approximation theory underpins much of numerical computation and arises also in several other branches of mathematics. It is one of the most mature disciplines of computational mathematics, to the extent that is can even be treated as a branch of pure mathematics for those who like to make this distinction. This course takes a more computational perspective. While it still focuses primarily on mathematics and theory, the choice of material is with an eye to applications in numerical simulation, modelling and machine learning rather than purely for its own sake. Theory and computational experiments will be developed in close collaboration. We will sometimes sacrifice optimality of the results for simplicity and stronger intuition.

The first question is to address what we mean by "simple functions". Briefly, we mean functions that are efficient and accurate (numerical stability!) to evaluate in (typically) floating point arithmetic on a modern processor. This simple observation already shows that approximation theory cannot be detached from numerical analysis and computer simulation. In Part I we will focus on univariate approximation initially with trigonometric polynomials (e^{inx}), algebraic polynomials (x^n) and then time permitting extend to splines and rational functions. In some PDEs but in particular in ML the approximation problems are often high-dimensional. In Part II of this course we will explore the challenges that one encounters in this setting and some classical and modern attempts to overcome them.

Motivation / Applications:

- Solving (partial) differential and integral equations
- machine learning, data-driven modelling, data assembly: The recent explosion in machine learning has given the field a new boost.

Themes:

1. Approximation spaces: what are "good" functions that we can combine to approximate general functions well; e.g.,
    - Global approximation: trigonometric and algebraic polynomials
    - Piecewise approximation: splines
    - rational functions
    - Ridge functions
    - Radial basis functions
    - sparse grids
    - artificial neural networks

2. Algorithms, constructive approximation:
   - best approximation, L^2-projection
   - interpolation
   - kernel methods
   - least squares
   - adaptive grids
   - Inverse problems, parameter estimation
3. Miscellaneous
   - Regularity
   - Numerical stability
   - Curse of dimensionality

1.1 Literature & Acknowledgements

{sec:acknowledgements}

Section 3 is largely based on random online available lecture notes but partly motivated by [Tre00, Tre13].

Section 4 largely follows [Tre13], adding only the Chebyshev transform and Jackson's theorem which are natural consequences of the material on trigonometric approximation. The book [Tre13] is available for free online at

`http://www.chebfun.org/ATAP/`

The section on splines is fairly standard material, but is based to some extend on the classical text [Pow81].

Exercises are partly based on gaps in the lecture material, partly adapted from these references.

2 Preliminaries

{sec:prelims}

2.1 Abstract Approximation Problems

We are concerned with approximating specific functions given to us, or classes of functions with specific properties, such as some given regularity, periodicity, symmetries, etc. To study generic approximation schemes it is therefore useful to begin by specifying a class $Y \subset X$ of functions of interest. Typically X will be an infinite-dimensional linear space, and Y an infinite-dimensional non-trivial subset of X. X will be endowed with a notion of distance d. We will later always assume this is given by a norm, but this is not important for now.

In linear approximation (which is what most of this module is about) we are given a set $B_N \subset X$, consisting of N linearly independent *basis functions*. Given some $f \in Y$ we then wish to find an approximation to f from $\mathrm{span} B_N =: Y_N$.

Fundamental questions/problem arising in this are, e.g.,

- Convergence: $\inf_{p \in Y_N} d(p, f) \to 0$ as $N \to \infty$

- Best approximation: Find $p_N \in Y_N$ such that $d(p_N, f)$ is minimal.
- Approximation to within some tolerance: given $\tau > 0$ find N (minimal?) and $p_N \in Y_N$ such that $d(p_N, f) < \tau$.
- Rates of approximation: $\inf_{p \in Y_N} d(p, f) \leq \epsilon_N$ and characterise the rate, possibly uniformly for all $f \in Y$
- Construction of approximations: Given f give an algorithm to construct an approximation p_N e.g., the best approximant.
- Evaluation: efficient and numerically stable construction and evaluation of p_N.

In the exercises of Section 2 we will collect a few basic examples and generic facts.

2.2 Basics

In this section we briefly review some fact from analysis and linear algebra, and most importantly, complex analysis. Facts about function spaces, measure theory and functional analysis will be used only minimally throughout this course and lack of background in those topics should not dissuade anybody from participating!

2.2.1 $\mathbb{R}^N$

The majority of the analysis in this module is for general N-dimensional systems of ODEs. We will use the structure of $\mathbb{R}^N$ as a vector space, supplied with the Euclidean norm and inner product

$$x \cdot y := x^T y = \sum_{i=1}^{N} x_i y_i, \quad \text{and} \quad |x| := \sqrt{x \cdot x}$$

Key inequalities that we will use on a regular basis are the *triangle inequality*

$$|x + y| \leq |x| + |y| \qquad \text{for } x, y \in \mathbb{R}^N, \tag{2.1}$$ {{eq:triangle_ineq}}

the *Cauchy–Schwarz inequality*

$$|x \cdot y| \leq |x||y| \qquad \text{for } x, y \in \mathbb{R}^N, \tag{2.2}$$ {{eq:cauchyschwarz_ineq}}

and *Cauchy's inequalities*,

$$ab \leq \tfrac{1}{2}a^2 + \tfrac{1}{2}b^2 \qquad \text{for } a, b \in \mathbb{R}, \tag{2.3}$$ {{eq:cauchy_ineq}}

$$ab \leq \tfrac{\varepsilon}{2}a^2 + \tfrac{1}{2\varepsilon}b^2 \qquad \text{for } a, b \in \mathbb{R}, \varepsilon > 0. \tag{2.4}$$ {{eq:cauchy_eps_ineq}}

2.2.2 Smooth functions

Recall from the introductory analysis modules the definitions of continuous functions and of uniform convergence. Here, we define the spaces, for an interval $D \subset \mathbb{R}$,

$$C(D) := \big\{ f : D \to \mathbb{R} \big| f \text{ is continuous on } D \big\}$$

If D is compact ($D = [a, b]$ for $a, b \in \mathbb{R}$), then $C(D)$ is *complete* when equipped with the sup-norm

$$\|f\|_\infty := \|f\|_{L^\infty} := \|f\|_{L^\infty(D)} := \sup_{x \in D} |f(x)|.$$

We will more typically write $\|f\|_\infty$ if it is clear over which set the supremum is taken. Note also that D need not be compact in the definition of $\|\cdot\|_{\infty,D}$.

Moreover, we define the spaces of j times continuously differentiable functions

$$C^j(D) := \{f : D \to \mathbb{R} \big| f \text{ is } j \text{ times continuously differentiable on } D\},$$

and the associated norms

$$\|f\|_{C^j} := \|f\|_{C^j(D)} := \max_{n=0,\dots j} \|f^{(n)}\|_{\infty,D},$$

where $f^{(n)}$ denotes the nth derivative.

We also define $C^\infty(D) := \bigcup_{j>0} C^j(D)$.

We say $f : D \to \mathbb{R}$ is Hölder continuous if there exists $\sigma \in (0, 1]$ such that

$$|f(x) - f(x')| \leq C|x - x'|^\sigma \qquad \forall x, x' \in D.$$

The associated space is denoted by $C^{0,\sigma}$. If $\sigma = 1$ then we call f *Lipschitz continuous*. Further, we define the space $C^{j,\sigma}(D) := \{u \in C^j(D)|u^{(j)} \in C^{0,\sigma}(D)\}$.

The right-hand side in the definition of Hölder continuity is a special case of a *modulus of continuity*. We say that $f \in C([a, b])$ has a *modulus of continuity* $\omega : [0, \infty) \to \mathbb{R}$ if ω is monotonically increasing, $\omega(r) \to 0$ as $r \to 0$ and

$$|f(x) - f(x')| \leq \omega(|x - x'|) \qquad \forall x, x' \in [a, b].$$

Indeed, any f that is continuous on a closed interval has a modulus of continuity, which can simply be defined by

$$\omega(r) := \sup_{x \neq y \in [a,b], |x-y| \leq r} |f(x) - f(y)|.$$

2.2.3 Integrable functions

Sometimes it will be convenient to consider measurable functions, and for the sake of precision we briefly review the relevant definitions. For $D = (a, b)$ an interval and $f : D \to \mathbb{R}$ measurable (i.e., $f^{-1}(B)$ is a Lebesgue set whenever B is a Lebesgue set), we define

$$\|f\|_{L^p} := \|f\|_{L^p(D)} := \left(\int_D |f|^p \, dx \right)^{1/p}, \qquad 1 \leq p < \infty,$$

and

$$\|f\|_{L^\infty} := \|f\|_{L^\infty(D)} := \text{ess.} \sup_{x \in D} |f(x)|.$$

We define the spaces

$$L^p(D) := \{f : D \to \mathbb{R} \big| f \text{ is measureable and } \|f\|_{L^p(D)} < \infty\}.$$

Finally, we recall the concept of absolutely continuous functions and weak derivatives. If $g \in L^1(a,b)$ and

$$f(x) = c + \int_a^x g(t)\,dt,$$

then we say that f is absolutely continuous and write $f' = g$. Note that this derivative is now no longer defined in a pointwise sense, but it is normally defined via integration by parts,

$$\int_a^b f(x)\varphi'(x)\,dx = -\int_a^b f'(x)\varphi(x)\,dx \qquad \forall \varphi \in C^1(a,b), \varphi(a) = \varphi(b) = 0.$$

If f is absolutely continous, then we also write $f \in W^{1,1}(a,b)$. If f is absolutely continuous and $f' \in L^p$, then we say $f \in W^{1,p}(a,b)$. These are the so-called Sobolev spaces. An immediate generalisation is that, if $f \in C^{m-1}(a,b)$, with $f^{(m-1)}$ absolutely continous and $f^{(m)} \in L^p$ then we write $f \in W^{m,p}$.

We will only make *minimal* use of Lebesgue and Sobolev spaces, but it is occasionally convenient to use them. A student who is deeply uncomfortable with these spaces should feel confident to substitute technical rigour for intuition gained through examples and discuss this with the instructor.

2.2.4 Normed Spaces and Hilbert spaces

A tuple $(X, \|\cdot\|)$ is called a normed space or normed vector space if it is a linear space over the field $\mathbb{F}$ and $\|\cdot\| : X \to \mathbb{R}$ defines a norm, i.e., for all $f, g \in X, \lambda \in \mathbb{F}$

- $\|f + \lambda g\| \leq \|f\| + |\lambda|\|g\|$
- $\|f\| \geq 0$ and $\|f\| = 0$ iff $f = 0$.

X is called a Banach space if it is complete (i.e. all Cauchy sequences in X have a limit in X).

If D is compact then the spaces $(C^j, \|\cdot\|_{C^j})$ and $(L^p, \|\cdot\|_{L^p})$ are Banach spaces. $C^{j,\sigma}$ may also be made into Banach spaces, though we won't need this.

A tuple $(X, \langle\cdot,\cdot\rangle)$ is called a Hilbert space over $\mathbb{F} \in \{\mathbb{R}, \mathbb{C}\}$ if the following conditions are satisfied:

- X is a linear vector space
- $\langle\cdot,\cdot\rangle : X \times X \to \mathbb{F}$ is an inner product, i.e., for all $f, g, h \in X, \lambda \in \mathbb{F}$ we have
  - $\langle f, g\rangle = \overline{\langle g, f\rangle}$
  - $\langle f + \lambda g, h\rangle = \langle f, h\rangle + \lambda\langle g, h\rangle$
  - $\langle f, f\rangle \geq 0$
  - $\langle f, f\rangle = 0$ iff. $f = 0$.
- X is complete under the norm $\|f\| := \langle f, f\rangle^{1/2}$.

The most common example we will encounter are L^2-type spaces. In particular, if D is an interval (or in fact any measurable set), then $L^2(D)$ equipped with the inner product

$$\langle f, g\rangle_{L^2} := \int_D f\overline{g}\,dx$$

is a Hilbert space.

2.3 Analytic functions

A proper study of analytic functions requires far more time than we have available. But some basics will suffice for the most important ideas. To save time (and unfortunately skip some beautiful structures of complex numbers) we will work exclusively with the definitions via power series.

Recall therefore that each power series

$$\sum_{n=0}^{\infty} c_n (z - z_0)^n$$

has a radius of convergence

$$r = \frac{1}{\limsup_{n\to\infty} \sqrt[n]{|c_n|}}$$

That is, the series converges absolutely and uniformly for $|z - z_0| < r$. It is an easy consequence to show that the function defined by the power series is differentiable in that ball. (in fact, C^∞, but of course in complex analysis this always follows.)

Definition 2.1. *Let $D \subset \mathbb{C}$ be open and $f : D \to \mathbb{C}$. We say that f is analytic at a point $z_0 \in D$ if there exists a power series $\sum_{n=0}^{\infty} c_n (z - z_0)^n$ with positive radius of convergence $r > 0$ such that*

$$f(z) = \sum_{n=0}^{\infty} c_n (z - z_0)^n \qquad \forall z \in D, |z - z_0| < r.$$

We say f is analytic in D if it is analytic in each point $z_0 \in D$ and write $f \in A(D)$.

We will need two simple concepts around analytic functions: (1) continuations; and (2) path integrals. We will formulate simplified versions that are sufficient for our purposes and only give rough ideas of the proofs in the lectures (these are not contained in these lecture notes).

Proposition 2.2 (Analytic Continuation).

(i) Let $D \subset \mathbb{C}$ be open, $f : D \to \mathbb{C}$ and let $D' \subset D$ be the set of points in which f is analytic. Then D' is open.

(ii) Let $D' \subset D \subset \mathbb{C}$, with D open and connected and D' contains a line segment $\{(1 - t)z_0 + tz_1 | t \in [0, 1]\}$ with $z_0 \neq z_1$. Let $f : D' \to \mathbb{C}$ be analytic and let $f_1, f_2 : D \to \mathbb{C}$ be two analytic continuations of f to D i.e., f_j are analytic on D and $f_j = f$ on D'. Then, $f_1 = f_2$.

(Note: this result can be significantly generalised, but we will only need it for the case when D' is a line segment, and this case is fairly straightforward and intuitive to prove.)

(iii) Let $f \in A([a, b])$ then there exists $D \supset [a, b]$ open in $\mathbb{C}$ such that f can be uniquely extended to a function $f \in A(D)$.

Concerning path integrals, let $\mathcal{C}$ be a continuous and piecewise smooth oriented curve in $\mathbb{C}$, i.e., we identify $\mathcal{C}$ with a parametrisation $(\zeta(t))_{t\in[0,1]}$, then we define

$$\int_{\mathcal{C}} f(z)\,dz := \int_{t=0}^{1} f(\zeta(t))\zeta'(t)\,dt.$$

Note that this definition makes sense even if ζ is not C^1, but only piecewise C^1, and least with finitely many pieces.

If $\mathcal{C}$ is a Jordan curve (simple and closed), then we assume that the orientation is counter-clockwise and we will write

$$\oint_{\mathcal{C}} f(z)\,dz := \int_{\mathcal{C}} f(z)\,dz = \int_{t=0}^{1} f(\zeta(t))\zeta'(t)\,dt$$

and call this a *contour integral.*

Proposition 2.3 (Cauchy's Integral Theorem). *Let $D \subset \mathbb{C}$ be open and simply connected, f analytic in D and $\mathcal{C} \subset D$ a Jordan curve, then*

$$\oint_{\mathcal{C}} f(z)\,dz = 0.$$

2.4 Exercises

Exercise 2.1 (Best Approximations). {exr:prelims:bestapprox}

(i) Let X be a vector space endowed with a norm $\|\cdot\|$, $X_N \subset X$ with $\dim X_N = N < \infty$ and let $Y_N \subset X_N$ be closed. (E.g. $Y_N = X_N$ is admissible.) Prove that for all $f \in X$ there exists a best approximation $p_N \in Y_N$, i.e.,

$$\|p_N - f\| = \inf_{y_N \in Y_N} \|y_N - f\|.$$

(ii) Suppose $\|\cdot\|$ is strictly convex, i.e., for $f_0, f_1 \in X, \lambda \in (0,1)$,

$$\|(1-\lambda)f_0 + \lambda f_1\| \leq (1-\lambda)\|f_0\| + \lambda\|f_1\|$$

with equality if and only if $f_0 \propto f_1$. Suppose also that Y_N is convex. Under these two conditions prove that the best approximation from (i) is unique.

(iii) Suppose that the *best approximation operator* $\Pi_N : f \mapsto p_N$ where p_N is the unique best approximation to f is well-defined (e.g. in the setting of (ii)). Prove that $\Pi_N : X \to Y_N$ is continuous.

□

Exercise 2.2 (Best Approx. in max-norms). {exr:prelims:bestapprox_maxn

(i) Consider $X = \mathbb{R}^2$ equipped with the ℓ^∞-norm. Show that this norm is *not* strictly convex. Consider the best approximation from $Y_N := \{x \in \mathbb{R}^2 | |x|_\infty \leq 1\}$. Show that

- $f = (2, 0)$, then the best-approximation is non-unique.
- $f = (2, 2)$, then the best-approximation is unique.

Hint: A geometric approach to this question is easiest.

(ii) Now consider $X = C([-1, 1])$ and

$$X_0 = Y_0 = \{x \mapsto a | a \in \mathbb{R}\}$$

i.e., approximation by constant functions. Prove that $\| \cdot \|_C = \| \cdot \|_{L^\infty}$ is *not* strictly convex, but nevertheless the best approximation problem for Y_0 has a unique solution.

Hint: An easy way to prove this is to simply construct the best approximation operator explicitly, which also helps with (iii).

(iii) Bonus: Now replace $X_0 = Y_0$ in (ii) with

$$X_1 = Y_1 = \{x \mapsto a + bx | a, b \in \mathbb{R}\},$$

i.e., best approximation by affine functions. First, construct explicitly a best approximant from that space. Try to prove it is unique. This should be difficult; at least I don't see an elementary proof. (Please share if you find one!) We will return to this problem later in the course. □

The last exercise should show that already for affine approximation the proof is not entirely trivial. We will return to best polynomial approximation in the max-norm in § 4; a classical and mathematically beautiful but in practise rather useless subject. In general, best approximation in non-Hilbert space norms is an unpleasant business. By contrast, best approximation in Hilbert spaces is straightforward, at least in theory:

Exercise 2.3 (Best Approximation in a Hilbert Space). Let X be a Hilbert space with inner product $\langle \cdot, \cdot \rangle$ and $Y_N = X_N \subset X$ an N-dimensional subspace. {exr:prelims:bestapprox_hilb

(i) Show that the best approximation p_N of $f \in X$ in X_N is characterised by the variational equation

$$\langle p_N, u \rangle = \langle f, u \rangle \qquad \forall u \in X_N.$$

Show that this has a unique solution.

(ii) Let $\Pi_N f = p_N$ denote the best approximation operator. Show that it is an orthogonal projection.

(iii) Deduce that

$$\|f - \Pi_N f\|^2 = \|f\|^2 - \|\Pi_N f\|^2.$$

(iv) **Linear Approximation:** Let $\{e_j\}_{j \in \mathbb{N}}$ be an orthonormal basis of X, i.e., $\langle e_j, e_n \rangle = \delta_{jn}$ and $\operatorname{clos}\operatorname{span}\{e_j\}_j = X$. Let $X_N := \operatorname{span}\{e_1, \dots, e_N\}$, then prove that

$$\Pi_N f = \sum_{j=1}^{N} \langle f, e_j \rangle e_j. \qquad \square$$

Exercise 2.4. {exr:prelims:inequalities}

(i) Prove (2.3) and (2.4).

(ii) Use (2.3) to prove (2.2).

(iii) Use (2.2) to prove (2.1). □

Exercise 2.5. For the following functions f, specify to which of the following spaces they belong: $C^{j,\sigma}([-1,1])$ (specify j and σ), $C^{\infty}([-1,1])$, $A([-1,1])$, $L^p(-1,1)$. No rigorous proofs are required. {exr:prelims:functions}

(i) $f(x) = x^n$, $n \in \mathbb{N}$

(ii) $f(x) = |x|$

(iii) $f(x) = |x|^3$

(iv) $f(x) = |x|^{3/2}$

(v) $f(x) = (1 + x^2)^{-1}$

(vi) $f(x) = \exp(-1/(1/2 - x))\chi_{[-1,1/2)}(x)$

(vii) $f(x) = e^{-x^2}$

(viii) $f(x) = \cos(1.23x)$ □

Exercise 2.6. Construct the analytic extensions of the following functions to a maximal set D in the complex plane, which you should specify: {exr:prelims:extensions}

(i) $f(x) = e^{-x^2}$ on $\mathbb{R}$

(ii) $f(x) = (1 + x^2)^{-1}$ on $\mathbb{R}$

(iii) $f(x) = \sum_{j=0}^{\infty} x^j$ for $x \in (-1, 1)$

(iv) $f(x) = \int_0^{\infty} e^{-t(1-x)}\,dt$ for $x < 1$

□

Part I: Univariate Approximation

3 Trigonometric Polynomials

{sec:trig}

In this chapter we consider approximation of periodic functions by trigonometric polynomials (aka Fourier spectral methods). Throughout this chapter, let $\mathbb{T} := (-\pi, \pi]$ and we identify $C^j(\mathbb{T}) = C^j_{\rm per}(\mathbb{T})$, $A_{\rm per}(\mathbb{T}) = A(\mathbb{T})$, $L^p(\mathbb{T})$ to be the spaces of 2π-periodic functions on $\mathbb{R}$ that are, respectively, j times continuously differentiable, analytic, belong to $L^p(-\pi, \pi)$. Similarly, $H^j_{\rm per}(\mathbb{T}) = H^j(\mathbb{T})$ denotes the space of 2π periodic functions on $\mathbb{R}$ such that their restriction to *any* interval $(a, a+2\pi)$ belongs to $H^j(a, a+2\pi)$.

Examples of periodic functions:

- $\sin(nx) \in A(\mathbb{T})$
- $|\sin(nx)| \in C^{0,1}(\mathbb{T})$
- $|\sin(nx)|^3 \in C^{2,1}(\mathbb{T})$
- $e^{-\cos x} \in A(\mathbb{T})$
- $(c^2 + \sin^2 x)^{-1} \in A(\mathbb{T})$
- ...

Applications:

- BVPs with periodic boundary conditions and periodic data, e.g.,

$$\begin{aligned} -(p(x)u_x)_x + q(x)u &= f(x), \qquad x \in (-\pi, \pi), \\ u(-\pi) &= u(\pi), \\ u'(-\pi) &= u'(\pi), \end{aligned}$$

  where p, q, f are 2π-periodic, then under suitable conditions on p, q, f there exists a unique solution which is also 2π-periodic.

- Functions represented in polar coordinates: $u(x, y) = v(r, \theta)$ then, for r fixed, $\theta \mapsto v(r, \theta)$ is periodic.

  There are many other examples of naturally "periodic" coordinate systems, including e.g. spherical coordinates, or the dihedral angle.

Approximation by trigonometric polynomials is based on the idea of Fourier series representation of periodic functions. Talking about Fourier series becomes much more convenient if we extend the admissible range of all functions to $\mathbb{C}$;, i.e. $f : \mathbb{R} \to \mathbb{C}$, still 2π-periodic. The following definition then becomes natural.

Definition 3.1. *A trigonometric polynomial of degree N is any function of the form*

$$\sum_{k=-N}^{N} a_k e^{ikx}$$

The space of all such polynomials is denoted by $\mathcal{T}_N$. The canonical basis is

$$\left\{ e^{ikx} \middle| k = -N, -N+1, \ldots, N \right\}$$

Definition 3.2. *Let $f \in L^1(\mathbb{T})$, then its* Fourier coefficients *are given by,*

$$\hat{f}_k := \fint_{-\pi}^{\pi} f(x) e^{-ikx}\, dx \tag{3.1}$$ {{eq:trig:fourier coeffs}}

The N-th partial sum, is a trigonometric polynomial, which we denote by

$$\Pi_N f(x) := \sum_{n=-N}^{N} \hat{f}_k e^{ikx}.$$

3.1 Approximation by L^2-projection

{sec:trig:L2}

We will initially study approximation of functions in the L^2-norm. It can then be convenient to normalise the inner product, via

$$\langle f, g \rangle_{L^2(\mathbb{T})} := \fint_{-\pi}^{\pi} f g^* \, dx.$$

Equipped with this inner product, $L^2(\mathbb{T})$ is a Hilbert space.

Theorem 3.3. {th:trig:plancherel}

(i) Convergence of Fourier Series: $\{e^{ikx} | k \in \mathbb{Z}\}$ is an orthonormal basis for $L^2(\mathbb{T})$.

(ii) Plancherel Theorem: $\mathcal{F} : L^2(\mathbb{T}; \mathbb{C}) \to \ell^2(\mathbb{Z}; \mathbb{C})$ is an isomorphism; i.e., $f \in L^2(\mathbb{T})$ then $\hat{f} \in \ell^2(\mathbb{Z})$ and

$$\sum_{k \in \mathbb{Z}} \hat{f}_k \hat{g}_k^* = \fint_{\mathbb{T}} f g^* \, dx.$$

In particular, $\|f\|_{L^2} = \|\hat{f}\|_{\ell^2}$.

Proof. This is left as an exercise. The key point is that

$$\fint_{-\pi}^{\pi} e^{-ikx} e^{i\ell x} \, dx = \fint_{-\pi}^{\pi} e^{(\ell - i)x} = \begin{cases} 1, & \ell = k, \\ 0, & \text{otherwise.} \end{cases} \tag{3.2}$$

□

Exercise 3.1. There is a general theorem that all (separable) Hilbert spaces are isometrically isomorphic to $\ell^2(\mathbb{N})$ or equivalently to $\ell^2(\mathbb{Z})$. Explain why the Plancherel theorem simply shows that the Fourier series map $f \mapsto \hat{f}$ is an the explicit construction of this isometry. □

Proposition 3.4. *Let $f \in L^2(\mathbb{T})$, then* {th:trig:PiNf-orthproj}

$$\|\Pi_N f - f\|_{L^2}^2 = \sum_{|k| > N} |\hat{f}_k|^2. \tag{3.3}$$ {{eq:trip:PiNf-orthproj}}

In particular, $\Pi_N f$ is the L^2-orthogonal projection of f onto $\mathcal{T}_N$, or equivalently, the best approximation of f from $\mathcal{T}_N$ w.r.t. $\| \cdot \|_{L^2}$.

Proof. By definition,

$$f(x) - \Pi_N f(x) = \sum_{|k|>N} \hat{f}_k e^{ikx},$$

and Plancherel's theorem then implies (3.3).

The fact that $\Pi_N f$ is the best approximation is a straightforward consequence: if $g \in \mathcal{T}_N$, then

$$\begin{aligned}
\big\| f(x) - \Pi_N f(x) - g \big\|_{L^2}^2 &= \sum_{|k|\le N} |\hat{g}_k|^2 + \sum_{|k|>N} |\hat{f}_k|^2 \\
&\geq \sum_{|k|>N} |\hat{f}_k|^2 \\
&= \big\| f(x) - \Pi_N f(x) \big\|_{L^2}^2. \qquad \square
\end{aligned}$$

The main point of Lemma 3.4 is that, exactly as in the introductory example, we can characterise the error in terms of the decay of the Fourier coefficients, so we will study this next.

3.2 Decay of Fourier Coefficients

{sec:trig:decay}

As we already saw in the introductory example, the "smoother" f is, the faster $\hat{f}_k$ decay. The following results are not difficult to generalise in several ways; see remarks below, but in the spirit of valuing simplicity over optimality, we will formulate them only for C^p regularity.

Theorem 3.5. {th:trig:decay}

(i) Let $f \in C^{p-1}(\mathbb{T})$ and $f^{(p-1)}$ be absolutely continuous, then

$$|\hat{f}_k| \leq \|f^{(p)}\|_{L^1(\mathbb{T})} |k|^{-p}.$$

(ii) Paley–Wiener Theorem: If $f \in A(\mathbb{T})$, then there exists $a > 0$ such that

$$|\hat{f}_k| \lesssim e^{-aN}.$$

Proof of Theorem 3.5(1). Consider first the case $p = 1$, i.e., f is absolutely continuous. This means that f' exists almost everywhere and satisfies an integration by parts formula. We can utilize this as follows:

$$\begin{aligned}
-ik\hat{f}_k &= \fint f(x)\big(-ike^{-ikx}\big)\,dx \\
&= \fint f(x) \frac{d}{dx} e^{ikx}\,dx \\
&= -\fint f'(x) e^{ikx}\,dx.
\end{aligned}$$

Taking the modulus on the left-hand side we obtain

$$|k\hat{f}_k| \leq \fint |f'|\,dx = \|f'\|_{L^1(\mathbb{T})}.$$

Since f' is AC, the right-hand side is finite. This yields the claim.

For $p > 1$, we can simply apply the integration by parts argument multiple times. $\square$

We postpone the proof of the Paley–Wiener Theorem to Theorem 3.9, but instead first discuss the consequences of these results. First, we consider approximation in the max-norm even though this seems counter intuitive in the Hilbert space setting.

We can also prove uniform convergence results using the decay of the Fourier coefficients established above:

Theorem 3.6. {th:trig:convergence_max}

(i) Let $f \in W^{p,1}(\mathbb{T})$ *then*

$$\|f - \Pi_N f\|_\infty \lesssim N^{1-p}$$

(ii) Let $f \in C^\infty(\mathbb{T})$*, then for each* $p > 0$ *there exists a constant* C_p *such that*

$$\|f - \Pi_N f\|_\infty \leq C_p N^{-p}.$$

(iii) If $f \in A(\mathbb{T})$*, then there exists* $a > 0$ *such that*

$$\|f - \Pi_N f\|_\infty \lesssim e^{-aN}.$$

Proof. (1) This is a straightforward calculation:

$$\begin{aligned}
\big|f(x) - \Pi_N f(x)\big| &= \bigg|\sum_{|k|>N} \hat{f}_k e^{ikx}\bigg| \\
&\leq \sum_{|k|>N} |\hat{f}_k| \\
&\lesssim \sum_{k=N+1}^{\infty} |k|^{-p} \lesssim N^{-p}.
\end{aligned}$$

(2) is an immediate consequence; (3) is proven analogously. □

Returning to convergence in L^2, a direct naive calculation shows that, if $f \in W^{p,1}(\mathbb{T})$, then

$$\|f - \Pi_N f\|_{L^2} \lesssim N^{1/2-p}.$$

Is this sharp? Depends really on whether f is *exactly* in $W^{p,1}$ and no better. The tricky part is how one can really test this. In practice one actually rarely encounters functions of this kind (though in the theory of PDEs they are phenomenally important), and it is much more convenient and natural to consiter scales of continuous and Hölder continuous functions. In particular Hölder continuity connects very naturally to the kind of singularities one typically occurs in physical models. We will get to this soon.

For now, we will briefly look at two simple extensions that can be convenient sometimes and yield slightly sharper results. The proofs are left as exercises.

Lemma 3.7. *Let* $f \in W^{p,2}(\mathbb{T})$*, then* $(\hat{f}_k |k|^p)_{k\in\mathbb{Z}} \in \ell^2(\mathbb{Z})$*.* {th:trig:fCp-coeffL2}

Theorem 3.8. *Let* $f \in W^{p,2}(\mathbb{T})$ *then* {th:trig:convergence_L2}

$$\|f - \Pi_N f\|_{L^2} \lesssim N^{-p}$$

3.2.1 Remarks

1. The algebraic convergence rates are rarely sharp, in the sense that taking an arbitrary function and showing that it is $W^{p,2}, W^{p,1}$ but no better in any way would be a rarity. The precise structure of $f^{(p)}$ is extremely relevant. One could go into fractional Sobolev spaces, to give a finer control, but we will not pursue this for now.

2. The important main message take away from these results is this: (1) $f \in C^p(\mathbb{T})$ regularity gives algebraic decay of $\hat{f}_k$; (2) $f \in C^\infty(\mathbb{T})$ gives super-algebraic decay; (3) $f \in A(\mathbb{T})$ gives exponential decay.

3. The uniform convergence estimate for analytic functions arising from the Paley–Wiener theorem is however qualitatively sharp.

4. The condition $f \in W^{p,1}$ is not sharp. One can weaken it to require only that $f^{(p-1)}$ has finite total variation. We will not look at this in generality, but consider a special case of this generalisation in Exercise 3.6.

3.3 The Paley–Wiener Theorem

{sec:trip:pw}

If f is analytic on an interval $[a, b]$, then standard theorems of complex analysis imply the it can be extended to a analytic function in a neighbourhood U of $[a, b]$. In the case of periodic functions, such a neighbourhood can be chosen to be a strip,

$$\Omega_\alpha := \{z \in \mathbb{C} | |\Im z| < \alpha\},$$

for some $\alpha > 0$. This is the starting point for a more refined version of Theorem 3.5(2).

Theorem 3.9. *Suppose that f is analytic in Ω_α with $\sup_{z \in \Omega_\alpha} |f(z)| = M_\alpha$, then* {th:trig:pw-trefversion}

$$|\hat{f}_k| \le 2\pi M_\alpha e^{-\alpha|k|}.$$

Proof. Assume $k > 0$; the case $k < 0$ is analogous. Recall that

$$\hat{f}_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) e^{-ikx}\, dx.$$

We fix some $\beta < \alpha$ and define a complex contour

$$\mathcal{C} := (-\pi, \pi] \cup (\pi, \pi + \beta i] \cup (-\pi + \beta i, \pi + \beta i] \cup (-\pi, -\pi + \beta i] = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3 \cup \mathcal{C}_4,$$

to be traversed counterclockwise. In particular $\frac{1}{2\pi i} \int_{\mathcal{C}_1} f(z) e^{ikz}\, dz = \hat{f}_k$, and periodicity of f yields

$$\sum_{j \in \{2,4\}} \int_{\mathcal{C}_j} f(z) e^{ikz}\, dz = 0.$$

Combining these observations with Cauchy's theorem yields

$$\begin{aligned}
0 &= \frac{1}{2\pi} \oint_{\mathcal{C}} f(z) e^{ikz}\, dz \\
&= \sum_{j=1}^{4} \frac{1}{2\pi} \int_{\mathcal{C}_j} f(z) e^{ikz}\, dz \\
&= \hat{f}_k + \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x+\beta i) e^{i(x+\beta i)k}\, dx.
\end{aligned}$$

Since we assumed that $k > 0$ we have $|e^{i(x+\beta i)k}| = e^{-\beta k}$, hence rearranging the previous identity yields the estimate

$$|\hat{f}_k| \le \fint_{-\pi}^{\pi} |f(x+\beta i)| e^{-\beta k}\, dx \le M_\beta e^{-\beta k}.$$

Since the upper bound valid for all $\beta < \alpha$ it also holds for $\beta = \alpha$. □

The previous theorem clarifies that, to precisely understand the best-approximation of an analytic function f by trigonometric polynomials we *must* study f not on $\mathbb{T}$ but in the complex plane. While some further generalisations are possible, we will restrict ourselves mostly to the context of Theorem 3.9 and thus look for the largest α such that f can be extended to a analytic function on Ω_α.

Suppose we have found an α such that $f \in A(\Omega_\alpha)$. If f blows up at some $x \pm i\alpha$ then we have found the maximal region of analyticity. If f is bounded in Ω_α then it is analytic at every point $z \in \partial\Omega_\alpha$ and hence we can extend f to a analytic function in a larger domain $\Omega_{\alpha'}$, $\alpha' > \alpha$. Thus, to determine the maximal region of analyticity we must find the *poles* of f. We obtain the following simple corollary of Theorem 3.9.

Corollary 3.10. *Let $f \in A(\mathbb{T}) \cap A(\Omega_\alpha)$ with α maximal, then for all $\epsilon > 0$ there exists $C_\epsilon > 0$ such that* {th:trig:pw-sharp}

$$|\hat{f}_k| \le C_\epsilon e^{-(\alpha-\epsilon)|k|}.$$

Moreover, we have the approximation error estimate

$$\|f - \Pi_N f\|_{L^\infty} \lesssim C'_\epsilon e^{-(\alpha-\epsilon)N} \qquad \forall \epsilon > 0.$$

Example 3.11 (Smeared Zig-Zag). Consider a family of periodic functions inspired by our introductory example,

$$f(x) = (1 + c^2 \sin^2 x)^{-1},$$

where $c > 0$. Then the analytic extension is still given by $f(z) = (1 + c^2 \sin^2 z)^{-1}$. To find the maximal strip of analyticity we need to compute the poles, i.e., the points $z \in C$ such that $\varepsilon^2 + \sin^2 z = 0$, or equivalently $\sin z = \pm i\varepsilon$, where $\varepsilon = 1/c$.

To that end, we first note that

$$\sin z = \sin(x+iy) = \sin x \cosh y + i\big\{ \cos x \sinh y \big\}.$$

Thus the poles are given by the solutions to

$$\sin x \cosh y = 0, \qquad\qquad \cos x \sinh y = \pm\varepsilon.$$

Since $\cosh y \neq 0$, The first condition requires $\sin x = 0$, or, $x \in \pi\mathbb{Z}$, hence $\cos x = \pm 1$. The second condition therefore yields $\sinh y = \pm\varepsilon$, or, equivalently,

$$x \in \pi\mathbb{Z}, \qquad y = \pm \sinh^{-1}\varepsilon.$$

This characterises all the poles of $f(z)$, and in particular shows that the maximal strip of analyticity is

$$\Omega_{\sinh^{-1}\varepsilon}$$

Our theory therefore predicts (ignoring the ϵ-factors) that

$$|\hat{f}_k| \lesssim e^{-\sinh^{-1}\varepsilon |k|} \sim e^{-\varepsilon|k|} = e^{-|k|/c} \qquad \text{for } \varepsilon \sim 0$$

as well as the approximation error estimate

$$\|f - f_N\|_\infty \lesssim e^{-\sinh^{-1}\varepsilon N} \sim e^{-\varepsilon|k|} = e^{-|k|/c} \qquad \text{for } \varepsilon \sim 0.$$

After discussing trigonometric interpolation we will show numerical tests demonstrating that this is sharp. □

Finally, it is also natural to ask about the case when f is entire, i.e., $f \in A(\Omega_\alpha)$ for all $\alpha > 0$. In this case, we simply obtain Theorem 3.10 with $\alpha = \infty$:

Corollary 3.12. *Suppose that $f \in A(\mathbb{T}) \cap A(\mathbb{C})$ (i.e., f is entire), then for all $\alpha > 0$ there exists $C_\alpha > 0$ ($C_\alpha = \|f\|_{L^\infty(\Omega_\alpha)}$ such that* {th:trig:pw-entire}

$$|\hat{f}_k| \lesssim C_\alpha e^{-\alpha|k|}.$$

3.4 Approximation by convolution: Jackson's Theorem

{sec:trig:jackson}

The L^2-projection operator Π_N can be written in terms of the *Dirichlet Kernel*,

$$D_N(x) = \frac{\sin\big((N+1/2)x\big)}{\sin(x/2)}.$$

That is,

$$\Pi_N f(x) = D_N * f(x) := \fint_{-\pi}^{\pi} D_N(x-t) f(t)\, dt.$$

See Exercise 3.7 for the details. An analysis of the fine properties of the kernel D_N leads to more precise convergence estimates, in particular in norms other than L^2.

Suppose, for example that we wanted to get an optimal estimate for $\|f - \Pi_N f\|_\infty$, then we can estimate (similar as we do for estimating the interpolation error!)

$$\|f - \Pi_N f\|_\infty \leq \|f - t_N\|_\infty + \|\Pi_N(t_N - f)\|_\infty.$$

Moreover,

$$|\Pi_N g(x)| = \left| \fint D_N(x-t)g(t)\,dt \right| \le \|D_N\|_{L^2(\mathbb{T})} \|g\|_\infty,$$

and hence,

$$\|f - \Pi_N f\|_\infty \le (1 + \|D_N\|_{L^1}) \|t_N - f\|_\infty,$$

i.e. the L^2 projection $\Pi_N f$ is optimal up to a constant factor $C_N = 1 + \|D_N\|_{L^1}$. It turns out that $\|D_N\|_{L^1} \approx \log N$, which means we are really *very close* to optimal (again, see Exercise 3.7). But it is still interesting to ask whether this log-factor can be removed!

In general, we can ask whether alternative kernels could be employed to construct approximations via convolutions,

$$(K_N * f)(x) := \fint_{-\pi}^{\pi} K_N(x-t) f(t)\,dt,$$

and what advantages those different kernels might have. If $K_N(t)$ is a trigonometric polynomial, then $K_N * f$ will also be a trigonometric polynomial:

Lemma 3.13. *If $K_N \in \mathcal{T}_N$, then $K_N * f \in \mathcal{T}_N$ for all $f \in L^1(\mathbb{T})$.*

Proof.

$$\begin{aligned}
\fint_{-\pi}^{\pi} K_N(x-t) f(t)\,dt &= \sum_{k=-N}^{N} \sum_{k' \in \mathbb{Z}} \hat{K}_{N,k} \hat{f}_{k'} \fint_{-\pi}^{\pi} e^{ik(x-t)} e^{ik't}\,dt \\
&= \sum_{k=-N}^{N} \sum_{k' \in \mathbb{Z}} \hat{K}_{N,k} \hat{f}_{k'} e^{ikx} \delta_{kk'} \\
&= \sum_{k=-N}^{N} \hat{K}_{N,k} \hat{f}_k e^{ikx}.
\end{aligned}$$

□

There is considerable freedom in the choice of kernel; the only thing they all have in common is that they must approximate the Dirichlet-delta function in a suitable sense. Purely for the purpose of theoretical analysis, a great choice is the Jackson kernel,

$$J_M(x) := \gamma_M \left(\frac{\sin(Mx/2)}{\sin(x/2)} \right)^4, \qquad \fint_{\mathbb{T}} J_M(x) = 1,$$

where the second condition determines the normalisation constant γ_M. Constructing approximates via the Jackson kernel leads to elegant and sharp approximation error estimates in the max-norm in particular for Hölder continuous functions. Note already that we now have $\|J_M\|_{L^1} = 1$, i.e. we no longer have the growth of the operator norm as in the Dirichlet kernel!

We write J_M instead of J_N since the degree of J_M is *not* equal to M:

Lemma 3.14. $J_M \in \mathcal{T}_{2M-2}$.

Proof. Let $z = e^{ix/2}$, then

$$J_M(x) = \big((z^M - z^{-M})/(z - z^{-1})\big)^4.$$

Further, we have

$$\begin{aligned}\frac{z^M - z^{-M}}{z - z^{-1}} &= z^{M-1} + z^{M-3}z^{-1} + z^{M-3}z^{-2} + \cdots + zz^{-M+2} + z^{-M+1} \\ &= z^{M-1} + z^{M-3} + z^{M-5} + \cdots + z^{-M+1} = \sum_{\alpha\in\mathcal{A}} z^\alpha,\end{aligned}$$

where $\mathcal{A} := \{-M+1, -M+3, -M+5, \dots, M-1\}$. Squaring yields

$$\begin{aligned}\left(\frac{z^M - z^{-M}}{z - z^{-1}}\right)^2 &= \sum_{\alpha,\beta\in\mathcal{A}} z^\alpha z^\beta \\ &= \sum_{\alpha,\beta\in\mathcal{A}} \frac{z^{\alpha+\beta} + z^{-\alpha-\beta}}{2} \\ &= \sum_{\alpha,\beta\in\mathcal{A}} \cos\big(\tfrac{\alpha+\beta}{2}x\big).\end{aligned}$$

Since $\alpha + \beta$ is always even, it follows that $(\frac{z^M - z^{-M}}{z - z^{-1}})^2 \in \mathcal{T}_{M-1}$ and in particular $J_M \in \mathcal{T}_{2M-2}$. □

Lemma 3.15. *There exist $C_1, C_2 > 0$, independent of M such that* {th:trig:gammaM_bound}

$$C_1 M^{-3} \le \gamma_M \le C_2 M^{-3}$$

Proof. First note the geometrically evident fact that

$$x/\pi \le \sin(x/2) \le x/2.$$

To obtain a lower bound, we estimate

$$\begin{aligned}\pi/\gamma_m &= \int_0^\pi \left(\frac{\sin(Mx/2)}{\sin(x/2)}\right)^4 dx \\ &= \int_0^{\pi/M} \left(\frac{\sin(Mx/2)}{\sin(x/2)}\right)^4 dx + \int_{\pi/M}^\pi \left(\frac{\sin(Mx/2)}{\sin(x/2)}\right)^4 dx \\ &\lesssim \left[\int_0^{\pi/M} \left(\frac{Mx/2}{x/2}\right)^4 dx + \int_{\pi/M}^\pi \left(\frac{1}{x}\right)^4 dx\right] \\ &= c_1 M^3.\end{aligned}$$

Note in particular that this calculation shows that for the opposite bound we only need to consider the interval $(0, \pi/M)$. Thus, we calculate

$$
\begin{aligned}
\pi/\gamma_M &\geq \int_0^{\pi/M} \left(\frac{\sin(Mx/2)}{\sin(x/2)}\right)^4 dx \\
&\gtrsim \int_0^{\pi/M} \left(\frac{Mx/2}{x/2}\right)^4 dx \\
&= c_2 M^3.
\end{aligned}
$$

□

The next Lemma is key, and hidden in its proof is the reason that we used the fourth power to define the Jackson kernel. It quantifies the fact (already exploited above) that J_M is concentrated near the origin.

Lemma 3.16. *There exists a constant $C > 0$ such that* {th:trig:jackson_moments}

$$
\begin{aligned}
\fint J_M(x)\,dx &= 1, \qquad \textit{and} \\
\fint |x| J_M(x)\,dx &\leq C M^{-1}.
\end{aligned}
$$

Proof. The case $m = 0$ follows immediately from the normalisation of the Jackson kernel, $\int_{-\pi}^{\pi} J_M(x)\,dx = 1$.

The case $m = 1$ can be seen by a variation of the proof of Lemma 3.15:

$$
\begin{aligned}
\int_0^{\pi} x J_M(x)\,dx &= \int_0^{\pi/M} x J_M(x)\,dx + \int_{\pi/M}^{\pi} x J_M(x)\,dx \\
&\lesssim \gamma_M \Big[\int_0^{\pi/M} x M^4\,dx + \int_{\pi/M}^{\pi} x \Big(\frac{1}{x}\Big)^4 dx \Big] \\
&\lesssim M^{-3}\big[M^2 + M^2\big] \lesssim M^{-1}.
\end{aligned}
$$

□

To state the Jackson theorems we first need to adapt the notion of modulus of continuity to the torus: Namely, we require that ω is a modulus of continuous for f on all of $\mathbb{R}$. It is clear that all properties of the modulus of continuity survive, including the fact that any $f \in C(\mathbb{T})$ has such a m.o.c.

Theorem 3.17 (Jackson's Theorem). {th:trig:jackson}

1. *Let $f \in C(\mathbb{T})$ with modulus of continuity ω, then*

$$\|f - J_M * f\|_\infty \lesssim \omega(M^{-1})$$

In particular, if $f \in C^{0,\sigma}(\mathbb{T})$, then

$$\|f - J_M * f\|_\infty \lesssim N^{-\sigma},$$

and if $f \in C^1(\mathbb{T})$, then

$$\|f - J_M * f\|_\infty \lesssim M^{-1}\|f'\|_\infty. \tag{3.4}$$

{{eq:trig:jackson:C1-version

2. Let $f \in C^p(\mathbb{T})$ and $f^{(p)}$ have modulus of continuity ω, then

$$\inf_{t_N \in \mathcal{T}_N} \|f - t_N\|_\infty \lesssim N^{-p}\omega(N^{-1}).$$

Proof of Theorem 3.17(1). Recall that the polynomial degree of $J_M * f$ is $N = 2M - 2$. For N even we take $M = N/2 + 1$ while for N odd we take $M = (N+1)/2 + 1$. Either way, $N \geq 2M - 2$.

$$\begin{aligned}
\big|J_M * f(x) - f(x)\big| &= \bigg| \int_{-\pi}^{\pi} \Big(f(x-t) - f(x)\Big) J_M(t)\, dt \bigg| \\
&\leq \int_{-\pi}^{\pi} \big|f(x-t) - f(x)\big| J_M(t)\, dt.
\end{aligned}$$

Next, we can use the modulus of continuity to estimate

$$\big|f(x-t) - f(x)\big| \leq \sum_{k=1}^{K} \big|f(x - kt/K) - f(x - (k-1)t/K)\big| \leq K\omega(t/K)$$

Choosing K minimal such that $t/K \leq 1/N$ (i.e. $K = \lceil tN \rceil$) yields

$$\big|f(x-t) - f(x)\big| \leq \begin{cases} \omega(N^{-1}), & 0 \leq |t| \leq N^{-1}, \\ 2tN\omega(N^{-1}), & |t| > N^{-1}. \end{cases}$$

Using Lemma 3.16 we conclude

$$\big|J_N * f(x) - f(x)\big| \leq \omega(N^{-1}) \int_0^{1/N} J_N(t)\, dt + 2N\omega(N^{-1}) \int_{1/N}^{\pi} tJ_N(t)\, dt \lesssim \omega(N^{-1}). \tag{3.5}$$ {{eq:trig:jackson1_proof_res

The stated results follow immediately from the fact that $\omega(N^{-1}) \leq \omega(M^{-1})$. For later reference though we will also need (3.5). □

To prove Theorem 3.17 (2), we need another auxiliary results that is also of independent interest.

Lemma 3.18. *Let $E_N(f) := \inf_{t_N \in \mathcal{T}_N} \|f - t_N\|_\infty$, then for $f \in C^1(\mathbb{T})$ we have* {th:trig:jackson-auxEN}

$$E_N(f) \lesssim N^{-1} E_N(f').$$

Proof. Let $q \in \mathcal{T}_N$ such that

$$\|f' - q\|_\infty = E_N(f').$$

(Because $\mathcal{T}_N$ is finite-dimensional, we know the best approximation error is attained.) Then we can write

$$q(x) = \sum_{k=-N}^{N} \hat{q}_k e^{ikx}.$$

We wish to write $q = t'_N$, but this is in general false if $\hat{q}_0 \neq 0$. Instead, we split

$$q(x) = \hat{q}_0 + r(x),$$

then $\hat{r}_0 = 0$ and hence there exists $t \in \mathcal{T}_N$ such that $t' = r$. Moreover, we can estimate

$$|\hat{q}_0| = \left| \fint_{-\pi}^{\pi} q\, dx \right| = \left| \fint_{-\pi}^{\pi} (q - f')\, dx \right| \leq E_N(f').$$

Combining these manipulations we obtain

$$\|f' - t'\|_\infty \leq \|f' - q\|_\infty + |\hat{q}_0| \lesssim 2E_N(f').$$

Finally, since $t \in \mathcal{T}_N$ we have $E_N(f) = E_N(f - t)$ and can therefore conclude, using Jackson's first theorem,

$$E_N(f) = E_N(f - t) \lesssim N^{-1} \|f' - t'\|_\infty \lesssim N^{-1} E_N(f').$$

where we also used that $r \mapsto \|f' - t'\|_\infty r$ is the modulus of continuity for $f - t$. $\square$

Proof of Theorem 3.17 (2). According to Lemma 3.18,

$$E_N(f) \lesssim N^{-1} E_N(f') \lesssim \cdots \lesssim N^{-p} E_N(f^{(p)}),$$

and according to Jackson's first theorem, for some $M \sim N$,

$$E_N(f^{(p)}) \leq \|f^{(p)} - J_M * f^{(p)}\|_\infty \leq \omega(M^{-1}) \lesssim \omega(N^{-1}),$$

that is, $E_N(f) \lesssim N^{-p}\omega(N^{-1})$. $\square$

3.5 Interpolation

{sec:trig:interp}

We have discussed two strategies to construct approximations of functions by trigonometric polynomials: L^2-projection and convolution (e.g., with the Jackson kernel). While both appear to be constructive, they both require additional computational effort to evaluate the relevant integrals. Since this is normally done via numerical quadrature, additional errors will be introduced that need to be analysed separately. All this can be done, but it turns out that a much more practical and performant approach that gives "near-optimal" approximants (most of the time) is nodal interpolation.

To specify a trigonometric polynomial $t \in \mathcal{T}_N$ we need to determine $2N + 1$ coefficients, which should be possible using $2N + 1$ function values, i.e., we may choose $2N + 1$ nodes $x_0, \dots, x_{2N} \in (-\pi, \pi]$ and specify

$$t(x_j) = F_j,$$

with F_j some prescribed function values. If the x_j are distinct, then it is easy to prove (see below and Exercise 4.1) If $F_j = f(x_j)$ for some $f \in C(\mathbb{T})$ the we call the resulting t a *nodal interpolant*.

An important question is how we can transform the nodal values into coefficients for the trigonometric polynomial. Naively, this can be achieved by simply solving a linear system for the coefficients at $O(N^3)$ cost: Let $t(x) = \sum_{k=-N}^{N} \hat{F}_k e^{ikx}$, then

$$\sum_{k=-N}^{N} \hat{F}_k e^{i\pi x_j} = F_j. \tag{3.6}$$

{{eq:trig:pre-dft}}

The first question to ask then is how to choose the interpolation nodes. Because on the torus no part of the domain is "special", it seems intuitive that without any additional information about the target function equispaced nodes must be "generically optimal". We will prove in the remainder of this section that it is in fact optimal up to a logarithmic factor, but will also return to a more careful discussion of different choices of interpolation nodes in § 4. Moreover, equispaced nodes also lead to fast algorithms (FFT) for solvig (3.6), but this requires some specific choices and some slightly annoying, but actually natural, modification of our approximation space.

The most common implementation of trigonometric interpolation employs

$$x_j = \frac{j\pi}{N}, \qquad j \in \mathbb{Z}.$$

The x_j are called *interpolation nodes*. They depend on N of course, but we supress this dependence for the sake of simplicity of notation. Unfortunately, for a 2π-periodic function, the nodes x_j, x_{j+2N} are equivalent, i.e., there are only $2N$ independent nodes and hence only $2N$ interpolation conditions. Thus, we cannot determine the $2N+1$ parameters of a $t_N \in \mathcal{T}_N$ but must modify our approximation space. To determine a trigonometric polynomial we may, for example, drop the e^{-iNx} basis function from $\mathcal{T}_N$, which leads to interpolants of the form

$$t(x) = \sum_{k=-N+1}^{N} c_k e^{ikx}.$$

But unless $c_N = 0$, this will mean that $t(x) \notin \mathbb{R}$ even if all $f_j \in \mathbb{R}$.

But why did we drop e^{-iNx} from the basis and not e^{iNx}? It actually turns out that it doesn't matter since those two basis functions agree on the interpolation nodes $x_j = j\pi/N$:

Lemma 3.19. *Let $x_j = j\pi/N$, then $e^{iNx_j} = e^{-iNx_j}$ for all $j \in \mathbb{Z}$.* {th:trig:baby-aliasing}

Proof.

$$e^{iNx_j} = e^{i\pi j} = (-1)^j = (-1)^{-j} = e^{-i\pi j} = e^{-iNx_j}.$$ □

A simple way therefore to proceed is to replace e^{iNx}, e^{-iNx} with their mean, i.e. $\cos(Nx)$ which leads to the following modified trigonometric polynomial space

$$\mathcal{T}_N' := \mathrm{span}\Big(\mathcal{T}_{N-1} \cup \{\cos Nx\}\Big) = \Big\{ t(x) = \sum_{k=-N+1}^{N-1} c_k e^{ikx} + c_N \cos(Nx) \Big\}.$$

Note, however, that for $x = x_j$ only we can equally write

$$t(x_j) = \sum_{k=-N+1}^{N-1} c_k e^{ikx_j} + c_N \cos(Nx_j) = \sum_{k=-N+1}^{N} c_k e^{ikx_j},$$

which will be convenient in the following.

Finally, to prepare us for discussing the FFT in the next section, we will fix the interpolation condition to be nodes $x_0, \dots, x_{2N-1}$. This may seem counterintuitive given we used $(-\pi, \pi]$ as the domain until now, but it is the most common convention and therefore we shall adopt it as well.

Lemma 3.20. *Let $F = (F_j)_{j=0}^{2N-1} \in \mathbb{C}^{2N}$, then there exists a unique $t \in \mathcal{T}'_N$ such that*

$$t(x_j) = F_j, \qquad j = 0, \ldots, 2N-1.$$

Proof. According to Lemma 3.19 we need to solve

$$\begin{aligned}
& \sum_{k=-N+1}^{N} c_k e^{i\pi kj/N} = F_j \\
\Leftrightarrow \quad & \sum_{k=-N+1}^{N} c_k \left(e^{i\pi j/N}\right)^k = F_j \\
\Leftrightarrow \quad & \sum_{k=-N+1}^{N} c_k z_j^k = F_j, \\
\Leftrightarrow \quad & \sum_{k=-N+1}^{N} c_k z_j^{k+N-1} = F_j z_j^{N-1},
\end{aligned}$$

where $z_j = e^{i\pi x_j}$ are distinct complex interpolation nodes. Existence and uniqueness of algebraic polynomial interpolation gives the stated result. (cf. Exercise 4.1).

REMARK: the last line in the above chain was unnecessary, but we will revisit this later. □

Definition 3.21. *Let $f \in C(\mathbb{T})$ then we define $I_N f \in \mathcal{T}'_N$ to be the unique nodal interpolant of f at the nodes $x_j = \pi j/N, j \in \mathbb{Z}$, i.e., $I_N f(x_j) = f(x_j)$ for $j \in \mathbb{Z}$.*

To understand the approximation error of the $I_N f$, let $f \in C(\mathbb{T})$, $t_N \in \mathcal{T}'_N$ arbitrary, then

$$\begin{aligned}
\|f - I_N f\|_\infty &\le \|f - t_N\|_\infty + \|t_N - I_N f\|_\infty \\
&= \|f - t_N\|_\infty + \|I_N(t_N - f)\|_\infty \\
&\le (1 + \|I_N\|)\|t_N - f\|_\infty,
\end{aligned}$$

where $\|I_N\|$ is the operator norm of I_N associated with $\|\cdot\|$, defined by

$$\|I_N\| = \sup_{\substack{f \in C(\mathbb{T}) \\ \|f\|_\infty = 1}} \|I_N f\|_\infty.$$

Taking the infimum over all $t_N \in \mathcal{T}_N$ we obtain that the interpolation error deviates from the best approximation error by factor determined by the operator norm of I_N, i.e.,

$$\|f - I_N f\| \le (1 + \|I_N\|) \inf_{t_N \in \mathcal{T}'_N} \|f - t_N\| \le (1 + \|I_N\|) \inf_{t_{N-1} \in \mathcal{T}_{N-1}} \|f - t_{N-1}\|,$$

where the final inequality of course shows that the convergence rate does not change asymptotively from that in $\mathcal{T}_N$ except possibly for a constant factor.

Definition 3.22. *The interpolation operator norm is also called* Lebesgue constant*, and typically denoted by* $\Lambda_N = \|I_N\|$.

Remark 3.23. The above argument works in principle with *any* norm. But to obtain a finite bound that norm must be such that $C(\mathbb{T})$ is complete under it. If not, then $\|I_N\|$ becomes infinite. As an exercise, you may check that the L^2-operator norm, $\|I_N\|_{L(L^2)}$ is indeed infinite. This is not surprising since functions $f \in L^2$ do not even have well-defined point values. $\square$

To estimate Λ_N we wish to write $I_N f$ in terms of a *nodal basis*, i.e.,

$$I_N f(x) = \sum_{j=-N+1}^{N} f(x_j) L_j(x),$$

where $x_j = \pi j / N$, then we can simply estimate

$$\Lambda_N \leq \sup_{x \in \mathbb{T}} \sum_{j=-N+1}^{N} |L_j(x)|. \qquad (3.7)$$ {{eq:trig:LamNbound}}

An immediate observation is that, since the grid is translation invariant, the nodal basis will be translation invariant as well, i.e., $L_j(x) = L_0(x - x_j)$. This already gives us a hint what to look for.

Lemma 3.24. *The nodal basis for trigonometric interpolation (for an even number of grid points* $2N$*) is given by a modified Dirichlet kernel,*

$$L_j(x) = D'_N(x - x_j)$$

where

$$D'_N(x) = \frac{\sin(Nx)}{2N \tan(x/2)}.$$

Proof. To see the identity for D'_N simply use $\sin(\alpha + \beta) = \sin\alpha\cos\beta + \cos\alpha\sin\beta$. From the definition of D'_N it is straightforward to check that $L_j(x_i) = \delta_{ij}$. For the case $i = j$ this is a limit argument: (fill in the details!)

$$\lim_{x \to x_j} L_j(x) = 1.$$

Thus we "only" need to show that L_j is indeed a trigonometric polynomial, or equivalently, $D_N \in \mathcal{T}_N$. This is achieved by an analogous argument as for the Jackson kernel.

See also Exercise 3.7 for the Dirichlet kernel related to $\mathcal{T}_N$ and how it relates to L^2-projection. $\square$

See `[Notebook 02]`for a numerical exploration of $\Lambda_N := \|I_N\|_{\mathrm{op}}$. The numerical experiments shown there suggest that the following theorem holds.

Theorem 3.25. *The Lebesgue constant for trigonometric interpolation with respect to the* L^∞*-norm is bounded by* {th:trig:lebesgue}

$$\|I_N\| \leq \tfrac{2}{\pi} \log(N+1) + 2.$$

In particular, if $f \in C(\mathbb{T})$, then

$$\|f - I_N f\|_{L^\infty} \lesssim \log N \inf_{t_N \in \mathcal{T}_N'} \|f - t_N\|_{L^\infty}. \tag{3.8}$$ {{eq:trig:almost_best_approx

If f is real we can replace $\mathcal{T}_N'$ with $\mathcal{T}_N$ in (3.8).

Remark 3.26. This bound is close to sharp. Erdös (1961) proved that

$$\frac{2}{\pi}\log N - c_1 \le \Lambda_N \le \frac{2}{\pi}\log N + c_2. \qquad \square$$

Proof. Recall (3.7), then we need to bound

$$\Lambda_N \le \sum_{j=-N+1}^{N} |D_N'(x - x_j)|.$$

By translation invariance and reflection symmetry we only need to consider $x = -t$, $t \in (0, \frac{\pi}{2N})$ (the case $t = 0$ is trivial); in this case,

$$\begin{aligned}
\Lambda_N &\le \sum_{j=-N+1}^{N} |D_N'(x - x_j)| \\
&\le \sum_{j=0}^{N} |D_N'(x_j + t)| + \sum_{j=-N+1}^{-1} \dots \\
&\le \frac{1}{2N}\left\{ \frac{\sin(Nt)}{\tan(t/2)} + \sum_{j=1}^{N} \left| \frac{\sin\big(N(x_j+t)\big)}{\tan\big((x_j+t)/2\big)} \right| \right\} + \dots \\
&\le \frac{1}{2N}\left\{ \frac{Nt}{t} + \sum_{j=1}^{N} \frac{2}{x_j + t} \right\} + \dots \\
&\le 1 + \frac{1}{\pi}\sum_{j=1}^{N}\frac{1}{j} + \dots \\
&\le 2\Big(1 + \tfrac{1}{\pi}\log(N+1)\Big) \le 2 + \tfrac{2}{\pi}\log(N+1).
\end{aligned}$$

(Or, we could use $\sum_{j=1}^{N} \frac{1}{j} \sim \log N + \gamma + O(1/N)$ to sharpen this a bit more.)

The last statement is left as an exercise. $\square$

3.6 The Fast Fourier Transform {sec:trig:fft}

As a final topic on the theme of trigonometric polynomial approximation we will study how to work efficiently with trigonometric interpolants. This is achieved via the discrete Fourier transform and its fast implementation, the *Fast Fourier Transform*, likely one of the most important and most widely used numerical algorithms. We will assume from here on that the number of grid points is even.

Given a function $f \in C(\mathbb{T})$ we can evaluate it at grid points x_j which leads to a grid function $F_j = (f(x_j))_{j=0}^{M-1}$. Given $M \in 2\mathbb{N}$ it is common to define the DFT and FFT for the grid

$$x_j = \frac{2\pi j}{M} \qquad j = 0, \dots, M-1.$$

The assumption that $M = 2N$ is even is consistent with § 3.5.

We then ask, what are the coefficients of the trigonometric polynomial $t_N \in \mathcal{T}_N' = \mathcal{T}_{M/2}'$ such that

$$t_N(x_j) = F_j \qquad \text{for } j = 0, \dots, M-1,$$

or, written as a linear system,

$$\sum_{k=-N+1}^{N} \hat{F}_k e^{i\pi jk/N} = F_j, \qquad j = 0, \dots, 2N-1. \tag{3.9}$$ {{eq:trig:pre-dft-equi}}

The first important observation is that the system matrix is orthogonal up to a constant factor; that is, the explicit inversion formula is given by

$$\hat{F}_k = \frac{1}{2N} \sum_{j=-N+1}^{N} F_j e^{-i\pi kj/N}, \tag{3.10}$$ {{eq:trig:pre-idft}}

This is not at all surprising given that close analogy of the sums with the integrals in the definition of the Fourier coefficients.

Exercise 3.2. Prove (3.10). The key intermediate result you should extract is

$$\sum_{j=0}^{2N-1} e^{i\pi jk'/N} e^{-i\pi jk/N} = 2N\delta_{kk'}, \tag{3.11}$$ {{eq:trig:orth-dft}}

which is precisely the orthogonality of the system matrix. □

We will call the mapping $F \mapsto \hat{F}$ the discrete Fourier transform (DFT) and its inverse the IDFT: for $F \in \mathbb{C}^M$, $k = 0, \dots, M-1$,

$$\begin{aligned} \mathrm{DFT}[F] := \hat{F}, \quad \text{where} \quad \hat{F}_k &= \frac{1}{M} \sum_{j=0}^{M-1} F_j e^{-ix_j k} \\ &= \frac{1}{M} \sum_{j=0}^{M-1} F_j e^{-i2\pi jk/M}. \end{aligned} \tag{3.12}$$ {{eq:trig:dft}}

Note in particular that this is a trapezoidal rule approximation of (3.1).

Remark 3.27. Since $x_j = 2\pi j/M$ it follows that {rem:trig:k-grid}

$$e^{-ix_j(k \pm M)} = e^{-ix_j k}$$

and hence the k-grid $\{0, \dots, M-1\}$ can alternatively be interpreted as, with $N = M/2$,

$$\{0, \dots, N, -N+1, -N+2, \dots, -1\}.$$

This is the ordering normally adopted in implementations of the DFT and IDFT. □

Proposition 3.28. *Let the* IDFT *be defined by* {th:trig:dft}

$$U = \mathrm{IDFT}[\hat{U}], \quad \textit{where} \quad U_j := \sum_{k=0}^{M-1} \hat{U}_k e^{ix_j k} = \sum_{k=0}^{M-1} \hat{U}_k e^{i2\pi jk/M}, \tag{3.13}$$ {{eq:trig:idft}}

then

$$\mathrm{IDFT}\big[\mathrm{DFT}[F]\big] = F \qquad \forall F \in \mathbb{C}^M.$$

In particular, if $\hat{F} = \mathrm{DFT}[F]$*, then the two trigonometric polynomials (cf. Remark 3.27)* $t \in \mathcal{T}_N, t' \in \mathcal{T}'_N$

$$\begin{aligned} t(x) &= \sum_{k=0}^{M-1} \hat{F}_k e^{ikx} \\ t'(x) &= \sum_{k=0}^{M/2-1} \hat{F}_k e^{ikx} + \hat{F}_{M/2} \cos(M/2x) + \sum_{k=M/2+1}^{M-1} \hat{F}_k e^{ikx} \end{aligned}$$

interpolate $(x_j, F_j)_{j=0}^{M-1}$*, i.e.,*

$$t(x_j) = t'(x_j) = F_j \qquad \textit{for } j = 0, \dots, M-1.$$

Proof. Left as an exercise. □

Using expression (3.12) the cost of computing $\mathrm{DFT}[F]$ is $O(N^2)$. Indeed, this is the cost of a generic matrix-vector multiplication, i.e., applying a linear operation in $\mathbb{R}^N \to \mathbb{R}^N$ that has no special structure. Luckily the DFT has plenty of structure to exploit, which finally brings us to the FFT algorithm (specifically the radix-2 variant of Cooley–Tukey's algorithm, though the idea famously goes back to Gauss).

We begin by rewriting

$$\hat{F}_k = M^{-1} \sum_{j=0}^{M-1} F_j \omega^{kj}, \qquad \text{where} \quad \omega := e^{-i2\pi/M}.$$

Then,

$$\begin{aligned} \hat{F}_k &= \sum_{j=0}^{M/2-1} F_{2j} \omega^{2kj} + \sum_{j=0}^{M/2-1} F_{2j+1} \omega^{k(2j+1)} \\ &= \sum_{j=0}^{M/2-1} F_{2j} \omega^{2kj} + \omega^k \sum_{j=0}^{M/2-1} F_{2j+1} \omega^{2kj} \\ &=: \hat{G}_k + \omega^k \hat{H}_k. \end{aligned} \tag{3.14}$$ {{eq:trig:fft_split}}

In particular, since $\omega^2 = e^{-i2\pi/(M/2)}$, we note that $\hat{G}_k$ is the DFT of $(F_{2j})_{j=0}^{M/2-1}$, while $\hat{H}_k$ is the DFT of $(F_{2j+1})_{j=0}^{M/2-1}$.

A final remark is that, *a priori* $\hat{G}_k$ and $\hat{H}_k$ will be given only for $k = 0, \dots, M/2-1$, but the expressions are $M/2$-periodic and (3.14) allows us to recover $\hat{F}$ for all $k = 0, \dots, M-1$. Specifically, we obtain the following identity:

$$\begin{aligned} \hat{F}_k &= \hat{G}_k + \omega^k \hat{H}_k, \qquad k = 0, \dots, M/2-1, \\ \hat{F}_k &= \hat{G}_{k-M/2} - \omega^{k-M/2} \hat{H}_{k-M/2}, \qquad k = M/2, \dots, M-1. \end{aligned} \tag{3.15}$$

{{eq:trig:fft_trick}}

(We could also write ω^k instead of $\omega^{k-M/2}$; this is equivalent.)

Suppose now that $M/2$ is still divisible by 2, then the can split the computation of $\hat{F}, \hat{G}$ again into four smaller DFTs. This process can of course be iterated. If $M = 2^m$, then after $m \approx \log M$ iterations iterations we compute $\approx M$ DFTs of length $O(1)$. Combining the small DFTs into the larger DFTs requires $O(M)$ operations at each level. Since there are $O(\log M)$ levels, this means that the cost of computing the original DFT is $O(M \log M)$. Algorithms that use some variant of this strategy are called *Fast Fourier Transforms*.

3.7 Exercises

Exercise 3.3. {exr:trig:hilbert-onb}

(i) Recall the definition of a complex Hilbert space and check that $(L^2(\mathbb{T}), \langle \cdot, \cdot \rangle_{L^2(\mathbb{T})})$ is indeed a pre-Hilbert space, i.e. check all conditions except for completeness. (Completeness is a bit more involved, but it is not particularly difficult; feel free to look this up in a suitable textbook.)

(ii) Complete the proof of the Plancherel Theorem; i.e. Theorem 3.3(ii).

(iii) Using Jackson's theorem, prove also Theorem 3.3(i).

Hint: use the fact that Π_N *is an orthogonal projector and in particular has operator norm 1.*

□

Exercise 3.4. Complete the proof of Theorem 3.8. □ {exr:trig:convergence_L2}

Exercise 3.5. For the following functions f, categorize their regularity as closely as possible and estimate the rate of convergence of $\|f - \Pi_N f\|_{L^2}$. {exr:trig:functions}

(i) $f(x) = \sin(x)$

(ii) $f(x) = \sin(x/2)$

(iii) $f(x) = |\sin(x)|$

(iv) $f(x) = |\sin(x)|^3$

(v) $f(x) = (1 + c^2 \sin^2 x)^{-1}$

(vi) $f(x) = \exp(-\sin(x))$

(vii) $f(x) = \exp(-1/(1-x^2))\chi_{(-1,1)}(x)$, extended 2π-periodically to $\mathbb{R}$.

Can you sharpen your estimates after working through Exercise 3.6? $\square$

Exercise 3.6 (Gibbs Phenomenon). Consider the periodic, piecewise constant function {exr:trig:gibbs}

$$f(x) = \begin{cases} 1, & x \in (0,\pi], \\ -1, & x \in (-\pi, 0]. \end{cases}$$

(i) Prove that, there exists no sequence of trigonometric polynomials $t_N \in \mathcal{T}_N$ such that $t_N \to f$ uniformly, but that

$$\|\Pi_N f - f\|_{L^2} \to 0 \qquad \text{as } N \to \infty.$$

(ii) Show that the Fourier series for f is given by

$$\Pi_N f(x) = \frac{4}{\pi} \sum_{\substack{j=1 \\ j \text{ odd}}}^{N} \frac{\sin(jx)}{j}.$$

(iii) Deduce that

$$\|\Pi_N f - f\|_{L^2} \lesssim N^{-1/2}.$$

(iv) **Gibbs Phenomenon:** Prove that

$$\lim_{N\to\infty} \Pi_N f\big(\tfrac{\pi}{N}\big) > 1$$

You may use without proof that

$$\int_0^\pi \frac{\sin(t)}{t}\,dt \approx \frac{\pi}{2} + \pi \cdot (0.089489\dots).$$

If you plot $\Pi_N f$ you will observe oscillations around the discontinuity. This "picture" is what is commonly known as the Gibbs phenomenon. It is a special case of **ringing artefacts***, which are a common occurance when piecewise smooth data is approximated using global basis functions. This can be nicely visualised in image processing; see e.g.* `https://en.wikipedia.org/wiki/Ringing_artifacts`.

(v) Piecewise smooth functions: Make an educated guess what the rate of convergence is for $\|\Pi_N f - f\|_{L^2}$ when all derivatives up to $f^{(p-2)}$ are continuous, $f^{(p-2)}$ is Lipschitz (and in particular absolutely continuous) and $f^{(p-1)}$ is piecewise absolutely continuous with jump discontinuities at finite many points. This includes functions such as $|\sin(nx)|, |\sin(nx)|^q$ for q odd.

Adapt the proof of Theorem 3.5 to rigorously prove this. $\square$

Exercise 3.7 (Dirichlet Kernel). {exr:trig:dirichlet}

(i) Prove that

$$D_N(x) = \frac{\sin\big((N+1/2)x\big)}{\sin(x/2)} = 1 + 2\sum_{k=1}^{N} \cos(kx) = \sum_{k=-N}^{N} e^{ikx}.$$

(ii) Deduce that,

$$(D_N * e^{in\bullet})(x) = \begin{cases} e^{inx}, & -N \le n \le N, \\ 0, & \text{otherwise} \end{cases}$$

(iii) Deduce that, if $f \in L^1(\mathbb{T})$, then

$$D_N * f = \Pi_N f.$$

(iv) Show that $\|D_N\|_{L^1} \lesssim \log N$ and hence

$$\|D_N * f\|_{L^\infty} \le \|D_N\|_{L^1} \|f\|_\infty \lesssim \log N \|f\|_\infty.$$

HINT: to estimate D_N use a similar splitting into sub-intervals as in the Jackson kernel estimates.

(v) Deduce that

$$\|f - \Pi_N f\|_\infty \lesssim \log N \inf_{t_N \in \mathcal{T}_N} \|f - f_N\|_\infty,$$

and in particular, if $f \in C^p(\mathbb{T})$ and $f^{(p)}$ has modulus of continuity ω, then

$$\|f - \Pi_N f\|_\infty \log N N^{-p} \omega(N^{-1}). \qquad \square$$

Exercise 3.8. Let $f \in A(\mathbb{T})$. Prove that there exists $\alpha > 0$ such that f has an analytic extension to Ω_α. Further, show that this extension (still called f) must be 2π-periodic, i.e., {exr:trig:periodic extension

$$f(x+iy) = f(x+2\pi+iy) \qquad \forall x+iy \in \Omega_\alpha. \qquad \square$$

Exercise 3.9 (The Exponentially Convergent Trapezoidal Rule). Let $f \in A(\mathbb{T})$, and consider the trapezoidal rule approximation of $I[f] := \fint_{-\pi}^{\pi} f\,dx$; {exr:trig:trapezoidal rule}

$$Q_N[f] := \frac{1}{2N} \sum_{j=-N+1}^{N} f(x_j),$$

where $x_j := j\pi/N$.

(i) Prove that,

$$\frac{1}{2N} \sum_{j=-N+1}^{N} e^{ikx_j} = \begin{cases} 1, & k \in 2N\mathbb{Z}, \\ 0, & \text{otherwise.} \end{cases}$$

(ii) Suppose f is analytic in Ω_α, where $\alpha > 0$ is maximal. Derive a sharp convergence rate for $|Q_N[f] - I[f]|$. *(You may of course revisit our sketches from the introductory lecture.)*

(iii) *Poisson's example:* The perimeter of an ellipse with axis lengths $1/\pi, 0.6/\pi$ is given by the integral

$$I = \frac{1}{2\pi} \int_{-\pi}^{\pi} \sqrt{1 - 0.36 \sin^2 \theta}\, d\theta.$$

(You may justify this, but this is not required.)

- Compute the region of analyticity for $f(\theta) = \sqrt{1 - 0.36 \sin^2 \theta}$, hence prove a rate of convergence for $Q_N[f]$. *(For this problem, you also need to estimate the prefactor!)*
- Only solve one of the following two problems:

  (OPTION 1) How many terms to you need to obtain 3, 5, 7 digits of accuracy? Using only a calculator, compute $I[f]$ to within 3 digits of accuracy. How many "non-trivial" function evaluations did you need?

  (OPTION 2) numerically demonstrate the convergence (use Julia, Matlab, Python or any language you wish.) □

Exercise 3.10. Prove Proposition 3.28. □

Exercise 3.11. **Radix-3 FFT:** Instead of M even suppose that $M = 3M'$ (you may actually still assume that M is even for consistency with our treatment of trigonometric interpolation, but this is not really relevant here). Generalise the FFT to this case, i.e., derive the analogues of (3.14) and (3.15). □

4 Algebraic Polynomials

{sec:poly}

Our second major topic concerns approximation of functions defined on an interval $f : [-1,1] \to \mathbb{R}$, without loss of generality. But contrary to § 3 we no longer assume periodicity. Instead we will approximate f by algebraic polynomials,

$$f(x) \approx p_N \in \mathcal{P}_N$$

where $\mathcal{P}_N$ denotes the space of degree N polynomials,

$$\mathcal{P}_N := \Big\{ \sum_{n=0}^{N} c_n x^n \Big| c_n \in \mathbb{R} \Big\}.$$

Note in particular that in the terms of "simplicity" these are indeed the simplest functions to evaluate numerically in that they only require addition and multiplication operations.

In terms of a basic convergence result we have the following initial proposition, which we will not prove now, but it will follow from our later work.

Proposition 4.1 (Weierstrass Approximation Theorem). *$\bigcup_{N\in\mathbb{N}} \mathcal{P}_N$ is dense in $C([-1,1])$ and by extension also in $L^p(-1,1)$ for all $p \in [1,\infty)$.* {th:poly:Weierstrass}

Indeed, as we have argued before, convergence in itself of *some* sequence of approximations is rarely useful, but we require (i) rates and (ii) explicit constructions. Much of this chapter is therefore devoted to interpolation.

It is a standard fact (and easy to prove) that for any $N+1$ distinct points $x_0, \dots, x_N \in \mathbb{R}$ and values $f_0, \dots, f_N$ there exists exactly one polynomial $p_N \in \mathcal{P}_N$ interpolating those values, i.e.,

$$p_N(x_j) = f_j, \qquad j = 0, \dots, N.$$

(Indeed, the same is even true for $x_j \in \mathbb{C}$.) These equations form a linear system for the coefficients c_n, which can be solved to obtain the interpolation polynomial, which in turn can be easily readily numerically.

A key question is how to choose the interpolation points x_j? It may seem intuitive to take equispaced nodes, $x_j = -1 + 2j/N$. We start this section by exploring precisely this approach to approximate some smooth functions on $[-1,1]$; see `[Notebook 03]`for some motivating examples. In this Julia notebook we clearly observe that this yields a divergent sequence of polynomials, but by exploring also other kinds of fits we also see that this does not preclude the possibility of computing a (very) good approximation. We therefore focus initially by deriving a "good" set of interpolation nodes. The same idea will also naturally lead to the Chebyshev polynomials.

4.1 Chebyshev Points, Chebyshev Polynomials and Chebyshev Series

We can motivate the idea of the Chebyshev points by mapping the polynomial approximation problem to the trigonometric approximation problem:

Let $f \in C([-1,1])$, then let $g \in C(\mathbb{T})$ be defined by

$$g(\theta) = f(\cos\theta).$$

Note that g "traverses" f twice!

We will later see that g inherits the regularity of f even across domain boundaries; for now let us understand the consequence of this observation. We know from § 3 that equispaced interpolation of g yields an excellent trigonometric interpolant, i.e., we choose $\theta_j = -\pi + 2\pi j/N$ and we choose coefficients $\hat{g}_k$ such that

$$t_N(\theta_j) = \sum_{-N}^{N} \hat{g}_k e^{ik\theta_j} = g(\theta_j)$$

We may ask to interpolate f at the analogous points, $x_j = \cos(\theta_j)$ but since g contains "two copies" we only take half of the nodes. This gives the Chebyshev nodes

$$x_j := \cos\big(\pi j/N\big) \qquad j = 0, \dots, N. \tag{4.1}$$ {{eq:poly:chebnodes}}

We can readily test our hypothesis that these yield much better approximations; see again `[Notebook 03]`. Thus, for future reference we define the Chebyshev interpolant $I_N f$ to be the unique function $I_f \in \mathcal{P}_N$ such that

$$I_N f(x_j) = f(x_j) \qquad \text{for } j = 0, \dots, N,$$

where x_j are the Chebyshev nodes (4.1).

Next, we ask what the analogue of the Fourier series is. We write

$$g(\theta) = \sum_{k \in \mathbb{Z}} \hat{f}_k e^{ik\theta},$$

then using that g is real and $g(-\theta) = g(\theta)$,

$$g(\theta) = \hat{g}_0 + 2 \sum_{k=1}^{N} \hat{g}_k \cos(k\theta)$$

It is therefore natural to define the *Chebyshev polynomials*

$$T_k(\cos\theta) = \cos(k\theta), \qquad k \in \mathbb{N} := \{0, 1, 2, \dots\}. \tag{4.2}$$ {{eq:poly:defn_Tk}}

A wide-ranging consequence of this definition is that

$$|T_k(x)| \leq 1 \qquad \forall k.$$

Lemma 4.2. *The functions $T_k : [-1, 1] \to \mathbb{R}$ are indeed polynomials and satisfy the recursion* {th:poly:chebpolys}

$$T_{k+1}(x) = 2x T_k(x) - T_{k-1}(x), \tag{4.3}$$ {{eq:poly:chebrecursion}}

with initial conditions $T_0(x) = 1, T_1(x) = x$.

Proof. The identities $T_0(x) = 1, T_1(x) = x$ follow immediately from (4.2). If we can prove the recursion, then the fact that T_k are polynomials follows as well.

To that end, we introduce another representation,

$$T_k\left(\tfrac{z+z^{-1}}{2}\right) = T_k(\Re z) = \Re z^k = \frac{z^k + z^{-k}}{2},$$

where $|z| = 1$. Then,

$$\begin{aligned}
&T_{k+1}(\Re z) - 2\Re z T_k(\Re z) + T_{k-1}(\Re z) \\
&\qquad = \tfrac{1}{2}\Big(z^{k+1} + z^{-k-1} - (z + z^{-1})(z^k + z^{-k}) + z^{k-1} + z^{-k+1}\Big) \\
&\qquad = \tfrac{1}{2}\Big(z^{k+1} + z^{-k-1} - z^{k+1} - z^{k-1} - z^{1-k} - z^{-1-k} + z^{k-1} + z^{-k+1}\Big) \\
&\qquad = 0.
\end{aligned}$$

□

For future reference we define the Joukowsky map

$$\phi(z) = \frac{z + z^{-1}}{2}.$$

and note that it is analytic in $\mathbb{C} \setminus \{0\}$.

We now know that $T_k(x)$ are indeed polynomials of degree k and in light of the foregoing motivating discussion, we have the following result.

Lemma 4.3. *Let $f \in C([-1,1])$ is uniformly continuous, then there exists* Chebyshev coefficients $\tilde{f}_k \in \mathbb{R}$ *such that the* Chebyshev series

$$f(x) = \sum_{k=0}^{\infty} \tilde{f}_k T_k(x) \tag{4.4}$$ {{eq:poly:chebseries}}

is absolutely and uniformly convergent.

The Chebyshev coefficients are given by the following equivalent formulas,

$$\begin{aligned}
\tilde{f}_k &= \frac{2}{\pi} \int_{-1}^{1} \frac{f(x) T_k(x)}{\sqrt{1-x^2}}\, dx \\
&= \frac{1}{2\pi i} \oint_{\mathbb{S}} \left(z^{-1+k} + z^{-1-k}\right) f(\phi(z))\, dz \\
&= \frac{1}{\pi i} \oint_{\mathbb{S}} z^{-1+k} f(\phi(z))\, dz \\
&= \frac{1}{\pi i} \oint_{\mathbb{S}} z^{-1-k} f(\phi(z))\, dz.
\end{aligned}$$

For $k = 0$ a factor $1/2$ must be applied.

Proof. If $f \in C([-1,1])$ with modulus of continuous ω, then $g \in C(\mathbb{T})$ also has a modulus of continuity and hence the Fourier series converges uniformly and equivalently, the Chebyshev series does as well.

The expressions for $\tilde{f}_k$ are simply transplanting the fourier coefficients $\hat{g}_k$ to Chebyshev coefficients $\tilde{f}_k$. □

In analogy with the truncation of the Fourier series $\Pi_N g$ (which is the $L^2(\mathbb{T})$-projection or best-approximation we define the Chebyshev projection

$$\tilde{\Pi}_N f(x) := \sum_{k=0}^{N} \tilde{f}_k T_k(x).$$

4.2 Convergence rates

{sec:poly:rates}

As we have learned in § 3, the real power of polynomials is in the approximation of analytic functions, hence we begin again with this setting.

Intuitively, the idea is that analyticity of f on $[-1,1]$ translates into analyticity of the corresponding periodic function $g(\theta) = f(\cos\theta)$. Exponential decay of the Fourier coefficients $\hat{g}_k$ then translates into exponential decay of the Chebyshev coefficients $\tilde{f}_k$. But we can prove this exponential decay directly with a relatively straightforward variation of the argument we used in § 3.3, which is interesting to see the analogies.

We begin by defining

$$F(z) := f(\Re z) = f\big(\tfrac{1}{2}(z+z^{-1})\big) = f(\phi(z)) \qquad \text{for } z \in \mathbb{S} := \{|z| = 1\}.$$

where $\phi(z) = \frac{1}{2}(z + z^{-1})$ is also called Joukowsky map. ϕ is clearly analytic in $\mathbb{C} \setminus \{0\}$. Thus, if f is analytic on $[-1,1]$ then F must be analytic on $\mathbb{S}$. Next, we note that analyticity of $g(\theta)$ on the strip Ω_α is equivalent to analyticity of F on the annulus

$$\mathbb{S}_\rho := \{z \in \mathbb{C} | \rho^{-1} \leq |z| \leq \rho\},$$

with $\rho = 1 + \alpha$. Let the corresponding *Bernstein ellipse* be the pre-image of $\mathbb{S}_\rho$ under the Joukowsky map,

$$E_\rho := \phi(\mathbb{S}_\rho),$$

then analyticity of f in E_ρ implies analyticity of F in $\mathbb{S}_\rho$.

Finally, we recall from the derivation of the Chebyshev polynomials $T_k(x)$ that they can also be written as

$$\tfrac{1}{2}(z^k + z^{-k}) = T_k(\phi(z)).$$

After these preparations, we can prove the following result.

Theorem 4.4 (Decay of Chebyshev coefficients). *Let $\rho > 1$ and $f \in A(E_\rho)$ with $M := \|f\|_{L^\infty(E_\rho)} < \infty$, then the Chebyshev coefficients of f satisfy*

$$|\tilde{f}_k| \leq 2M\rho^{-k}, \qquad k \geq 1.$$

Proof. We start with

$$\tilde{f}_k = \frac{1}{\pi i} \oint_{\mathbb{S}} z^{-1-k} F(z)\, dz$$

Since F is analytic on $\mathbb{S}_\rho$ (and hence in the neighbourhood of $\mathbb{S}_\rho$) we can expand the contour to *(Exercise: explain why this can be done using Cauchy's integral formula and a suitable sketch!)*

$$\tilde{f}_k = \frac{1}{\pi i} \oint_{|z|=\rho} z^{-1-k} F(z)\, dz$$

and hence we immediately obtain

$$|\tilde{f}_k| \leq \frac{2\pi\rho\rho^{-1-k}M}{\pi} = 2M\rho^{-k}. \qquad \square$$

Decay of Chebyshev coefficients gives the following approximation error estimates.

Theorem 4.5 (Chebyshev Projection and Interpolation Error). *Let $\rho > 1$ and $f \in A(E_\rho)$ with $M := \|f\|_{L^\infty(E_\rho)} < \infty$, then* {th:poly:err_analytic}

$$\|f - \tilde{\Pi}_N f\|_{L^\infty(-1,1)} \leq \frac{2M\rho^{-N}}{\rho - 1}, \tag{4.5}$$ {{eq:poly:projerror}}

$$\|f - I_N f\|_{L^\infty(-1,1)} \leq CM \log N \rho^{-N}, \tag{4.6}$$ {{eq:poly:interperror}}

where C is a generic constant.

Proof. For the proof of (4.5) we use the fact that $\|T_k\|_\infty \leq 1$ to estimate

$$\begin{aligned} \|f - \tilde{\Pi}_N f\|_\infty &\leq \sum_{k=N+1}^{\infty} |\tilde{f}_k| \\ &\leq 2M \sum_{k=N+1}^{\infty} \rho^{-k} \\ &= \frac{2M\rho^{-N}}{\rho - 1}. \end{aligned}$$

The estimate (4.6) follows from the bound on the Lebesgue constant

$$\|I_N\|_{L(L^\infty)} \leq C \log N,$$

which follows from the analogous bound for trigonometric interpolation given in Theorem 3.25. (For a sharp bound, it is in fact known that $\Lambda_N \leq \frac{2}{\pi} \log(N+1) + 1$.) □

Remark 4.6. One can in fact prove that

$$\|f - I_N f\|_{L^\infty(-1,1)} \leq \frac{4M\rho^{-N}}{\rho - 1},$$

using an aliasing argument; see [Tre13, Thm. 8.2], somewhat similar to the argument we used for our convergence estimate of the trapezoidal rule in Exercise 3.9. □

Example 4.7 (Fermi-Dirac Function). Consider the Fermi-Dirac function {exa:poly:fermi-dirac}

$$f_\beta(x) = \frac{1}{1 + e^{\beta x}}, \tag{4.7}$$

where $\beta > 0$.

REMARK: The Fermi–Dirac function describes the distribution of particles over energy states in systems consisting of many identical particles that obey the Pauli exclusion principle, e.g., electrons. A broad range of important algorithms in computational physics are fundamentally about approximating the Fermi–Dirac function. The parameters β is inverse proportional to temperature (that is, Fermi-temperature).

Extending f_β to the complex plane simply involves replacing x with z, i.e.,

$$f_\beta(z) = \frac{1}{1 + e^{\beta z}},$$

which is well-defined *except at the poles*

$$z_j = \pm i \frac{\pi}{\beta}.$$

In Exercise 4.5 we show that the semi-minor axis of the Bernstein ellipse E_ρ is $\frac{1}{2}(\rho - \rho^{-1})$, hence the largest ρ for which $\mathrm{int} E_\rho$ does not intersect any singularity is given by

$$\frac{1}{2}(\rho - \rho^{-1}) = \frac{\pi}{\beta} x.$$

Solving this quadratic equation for ρ yields one positive root

$$\rho = \tfrac{\pi}{\beta} + \sqrt{1 + \tfrac{\pi^2}{\beta^2}}$$

Of particular interest is the low temperature regime $\beta \to \infty$ (recall that $\beta \propto$ inverse temperature), for which we obtain

$$\rho \sim 1 + \tfrac{\pi}{\beta}.$$

In this regime we therefore expect an approximation rate close to

$$\|f_\beta - I_N f_\beta\|_\infty \lesssim \beta \big(1 + \tfrac{\pi}{\beta}\big)^{-N} \sim \beta \exp\big(- \pi \beta^{-1} N\big).$$

(Why is this not a rigorous and in fact likely false bound? You can get a rigorous reformulation from the foregoing theorems.) □

For convergence rates for $C^{j,\sigma}([-1, 1])$ and similar functions, we want to adapt the Jackson theorems. We could again "transplant" the argument from the Fourier to the Chebyshev setting, but it will be more convenient this time to simply apply the Fourier results directly. The details are carried out in Exercise 4.6. We obtain the following result.

Theorem 4.8 (Jackson's Theorem(s)). *Let $f \in C^{(j)}([-1, 1])$, $j \geq 0$, where $f^{(j)}$ has modulus of continuity ω, then* {th:poly:jackson}

$$\inf_{p_N \in \mathcal{P}_N} \|f - p_N\|_{L^\infty} \leq C N^{-j} \omega\big(N^{-1}\big). \qquad (4.8)$$ {{eq:poly:jackson1}}

Proof. See Exercise 4.6. □

We cannot yet test these predictions numerically, since we don't yet have a numerically stable way to evaluate the Chebyshev interpolants (or projections). We will remedy this in the next two sections.

4.3 Chebyshev transform

We have seen in [Notebook 03] that naive evaluation of the Chebyshev interpolant leads to highly unstable numerical results. The emphasis here is on the term "naive". Indeed, there exist at least two natural and numerically stable way to evaluate the Chebyshev interpolant.

The first approach we consider is the Discrete Chebyshev transform (DCT), an immediate analogy of the Discrete Fourier transform (DFT). As in the Fourier case, once we have transformed the polynomial to the Chebyshev basis, we can evaluate it in $O(N)$ operations. But in the Chebyshev case, this is even more efficient due to the recursion formula (4.3). Moreover, the polynomial derivatives are straightforward to compute in this case as well.

Let $F = (F_j) \in \mathbb{R}^{N+1}$ (we immagine that $F_j = f(x_j)$ are nodal values of some $f \in C([-1,1])$ at the Chebyshev nodes), then there exists a unique polynomial $p_N \in \mathcal{P}_N$ such that $p_N(x_j) = F_j$. We write $p_N(x) = \sum_{k=0}^N \tilde{F}_k T_k(x)$, then

$$\tilde{F} := \mathrm{DCT}[F] := \big(\tilde{F}_k\big)_{k=0}^N. \tag{4.9}$$ {{eq:poly:chebtransform}}

Since polynomial interpolation is linear and unique the operator is an invertible linear mapping, with inverse (obviously) given by

$$\big(\mathrm{IDCT}[\tilde{F}]\big)_j = \sum_{k=0}^N \tilde{F}_k T_k(x_j). \tag{4.10}$$

Lemma 4.9. *Let $\tilde{F} = \mathrm{DCT}[F]$, then* {th:poly:dct_explicit}

$$\tilde{F}_k = \frac{p_k}{N}\Big\{ \tfrac12\big((-1)^k F_0 + F_N\big) + \sum_{k=1}^{N-1} F_k T_k(x_j) \Big\}.$$

We won't prove Theorem 4.9 since we won't need this expression. It is only stated here for the sake of completeness. The interested reader will be able to check it by a direct computation; it is also implicitly contained in the following discussion.

A priori the cost of evaluating the DCT and IDCT is $O(N^2)$, but the connection between the Fourier and Chebyshev settings gives us an $O(N \log N)$ algorithm which we now derive. Let $F = \mathrm{IDCT}[\tilde{F}]$, then writing

$$T_k(x_j) = T_k(\cos(j\pi/N)) = \cos(kj\pi/N)$$

we obtain

$$\begin{aligned} F_j &= \sum_{k=0}^N \tilde{F}_k \cos(kj\pi/N) \qquad (4.11) \\ &= \sum_{k=0}^N \tilde{F}_k \tfrac12 \big(e^{i2\pi kj/(2N)} + e^{-i2\pi kj/(2N)}\big), \end{aligned}$$ {{eq:poly:costtransform}}

which looks *almost* like a IDFT on the grid $\{-N, \dots, N\}$. We can rewrite this a little more,

$$
\begin{aligned}
F_j &= \tilde{F}_0 + \sum_{k=1}^{N-1} \big[\tfrac{1}{2}\tilde{F}_k\big] e^{i2\pi kj/(2N)} + \tilde{F}_N \tfrac{1}{2}\big(e^{i2\pi Nj/(2N)} + e^{-i2\pi Nj/(2N)}\big) \\
&\quad + \sum_{k=-N+1}^{-1} \big[\tfrac{1}{2}\tilde{F}_{-k}\big] e^{i2\pi kj/(2N)} \\
&= \tilde{F}_0 + \sum_{k=1}^{N-1} \big[\tfrac{1}{2}\tilde{F}_k\big] e^{i2\pi kj/(2N)} + \tilde{F}_N e^{i2\pi Nj/(2N)} + \sum_{k=N+1}^{2N-1} \big[\tfrac{1}{2}\tilde{F}_{2N-k}\big] e^{i2\pi kj/(2N)} \\
&=: \sum_{k=0}^{2N-1} \hat{G}_k e^{i2\pi kj/(2N)},
\end{aligned}
$$

where we have defined

$$
\hat{G}_k := \begin{cases} \tilde{F}_k, & k = 0, \\ \frac{1}{2}\tilde{F}_k, & k = 1, \dots, N-1, \\ \tilde{F}_k, & k = N, \\ \frac{1}{2}\tilde{F}_{2N-k}, & k = N+1, \dots, 2N-1. \end{cases}
$$

Let $\hat{G}[\tilde{F}]$ be defined by this expression, then we have shown that

$$
F_j = \big(\mathrm{IDCT}[\tilde{F}]\big)_j = \big(\mathrm{IDFT}[\hat{G}[\tilde{F}]]\big)_j, \qquad j = 0, \dots, N.
$$

After determining F_j for $j = N+1, \dots, 2N-1$ we can then evaluate the DCT via the DFT. From the expression (4.11) we immediately see that

$$
\begin{aligned}
F_j &= \sum_{k=0}^{N} \tilde{F}_k \cos(kj\pi/N - 2\pi k) \\
&= \sum_{k=0}^{N} \tilde{F}_k \cos(k2\pi(j-2N)/2N) \\
&= \sum_{k=0}^{N} \tilde{F}_k \cos(k2\pi(2N-j)/2N) \\
&= F_{2N-j}
\end{aligned}
$$

That is, if we define

$$
G_j := \begin{cases} F_j, & j = 0, \dots, N, \\ F_{2N-j}, & j = N+1, \dots, 2N-1 \end{cases}
$$

then we obtain

$$
\mathrm{DFT}[G] = \hat{G},
$$

from which we can readily obtain $\tilde{F}$.

In `Julia` code an $O(N \log N)$ scaling Chebyshev transform might look as follows:

```
"fast Chebyshev transform"
function fct(F)
   N = length(F)-1
   G = [ F; F[N:-1:2] ]
   Ghat = real.(fft(F))
   return [Ghat[1]; 2 * Ghat[2:N]; Ghat[N+1]]
end

"fast inverse Chebyshev transform"
function ifct(Ftil)
   N = length(Ftil)-1
   Ghat = [Ftil[1]; 0.5 * Ftil[2:N]; Ftil[N+1]; 0.5*Ftil[N:-1:2]]
   G = real.(ifft(Ghat))
   return G[1:N+1]
end
```

Remark 4.10. The expression (4.11) is in fact another kind of well-known transform, the *Discrete Cosine Transform* (one of several variants). A practical implementation of the fast Chebyshev transform should therefore use an efficient implementation of the fast cosine transform rather than the FFT. For the sake of simplicity (to avoid studying yet another transformation) we did not study this transform in detail, but there is plenty of literature and software available on this topic. □

4.4 Barycentric interpolation formula

{sec:poly:bary}

The second method we discuss is the *barycentric interpolation formula*. After precomputing some "weights" it gives another $O(N)$ method to evaluate the Chebyshev interpolant (or indeed *any* polynomial interpolant) in a numerically stable manner. This method entirely avoids the transformation to the Chebyshev basis. (This section is taken almost verbatim from [Tre13]; see also [Tre13, Ch. 5] for a more detailed, incl historical, discussion).

We begin with the usual Lagrange formula for the nodal interpolant. Let $p(x_j) = f_j, j = 0, \dots, N$ where $p \in \mathcal{P}_N$, then

$$p(x) = \sum_{j=0}^{N} f_j \ell_j(x), \qquad \text{where} \quad \ell_j(x) = \frac{\prod_{n \neq j}(x - x_n)}{\prod_{n \neq j}(x_j - x_n)}.$$

This formula has the downside that it costs $O(N^2)$ to evaluate p at a single point x.

But we observe that $\ell_j(x)$ have a lot of terms in common. This can be exploited by defining the *node polynomial*

$$\ell(x) := \prod_{n=0}^{N} (x - x_n),$$

then we obtain

$$\ell_j(x) = \ell(x) \frac{\lambda_j}{x - x_j} \qquad \text{where} \qquad \lambda_j = \frac{1}{\prod_{n \neq j}(x_j - x_n)}. \tag{4.12}$$

{{eq:poly:bary_weights}}

The "weights" λ_j still cost $O(N^2)$, but they are independent of x and can therefore be precomputed (Moreover, for various important sets of nodes there exist fast algorithms. For Chebyshev nodes there is an explicit experession; see below.). Since the common factor $\ell(x)$ does not depend on j we can now evaluate all $\ell_j(x), j = 0, \dots, N$ at $O(N)$ cost and thus obtain the *first form of the barycentric interpolation formula*,

$$p(x) = \ell(x) \sum_{j=0}^{n} \frac{\lambda_j}{x - x_j} f_j. \tag{4.13}$$ {{eq:poly:bary1}}

Once the weights λ_j have been precomputed, the cost of evaluating $p(x)$ becomes $O(N)$. However, (4.13) has a different shortcoming: in floating point arithmetic it is prone to overflow or underflow. Specifically, suppose that $x = -1$ and we compute $\ell(x)$ with x_j ordered decreasingly as defined in (4.1), then after approximately the first $M \approx N/4$ terms we have evaluated

$$\bigg| \prod_{n=0}^{M} (x - x_j) \bigg| \geq \big(\tfrac{3}{4}\big)^{M+1}$$

which quickly becomes very large. The issue is also reflected in the coefficients λ_j, which for Chebyshev points are $O(2^N)$ (cf. Exercise 4.8). In practise, one typically gets overflow beyond 100 or so grid points.

This can be avoided with the second form of the barycentric formula: observing that $\sum_{j=0}^N \ell_j \equiv 1$ we obtain

$$1 = \ell(x) \sum_{j=0}^{N} \frac{\lambda_j}{x - x_j},$$

and hence arrive at the second form of the barycentric interpolation formula:

Theorem 4.11 (Barycentric interpolation formula). *Let $p \in \mathcal{P}_N$, with $p(x_j) = f_j$ at $N+1$ distincts points $\{x_j\}$ then* {th:poly:bary}

$$p(x) = \frac{\sum_{j=0}^N \frac{\lambda_j f_j}{x - x_j}}{\sum_{j=0}^N \frac{\lambda_j}{x - x_j}}, \qquad \text{where} \qquad \lambda_j = \frac{1}{\prod_{n \neq j}(x_j - x_n)},$$

with the special case $p(x_j) = f_j$.

Theorem 4.12 (Barycentric interpolation formula in Chebyshev Points). *Let $\{x_j\}$ be the Chebyshev points (4.1), then the barycentric weights λ_j from Theorem 4.11 may be chosen as* {th:poly:barycheb}

$$\lambda_j = \begin{cases} (-1)^j, & j = 1, \dots, N-1, \\ \frac{1}{2}(-1)^j, & j = 0, N. \end{cases}$$

Proof. See Exercise 4.8. □

4.4.1 Numerical stability of barycentric interpolation

{sec:poly:barystab}

While the DFT is matrix multiplication with an othogonal matrix, and the FFT an algorithm that even reduced the number of operations it is natural to expect that these algorithms are numerically stable. By contrast, this is not at all obvious *a priori* for the barycentric formula. We will therefore spend a little time discussing this. To simplify this discussion we will only analyse the numerical stability of the *first* barycentric formula (4.13). Understanding stability of the second barycentric formula is slightly more involved; see [Hig04] for the details.

We have to begin by explaining the standard model of floating point arithmetic. Let $\otimes \in \{+, -, *, /\}$ be one of the standard four floating point operations, then applying the operation $a \otimes b$ to two floating point numbers will give an error, which we express as

$$\mathrm{fl}\big(a \otimes b\big) = (a \otimes b)(1 + \delta),$$

where $|\delta| \leq \varepsilon$ and ε denotes machine precision (typically 10^{-6}). That is, floating point arithmetic controls the *relative error*. For more on this topic, in particular additional subtleties that we are sweeping under the carpet here, see [Hig02].

For example, consider the evaluation of an inner product of two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^2$,

$$\begin{aligned}
\mathrm{fl}(\mathbf{a} \cdot \mathbf{b}) &= \mathrm{fl}\big(\mathrm{fl}(a_1 b_1) + \mathrm{fl}(a_2 b_2)\big) \\
&= \mathrm{fl}\big(a_1 b_1 (1 + \delta_1) + a_2 b_2 (1 + \delta_2)) \\
&= \big(a_1 b_1 (1 + \delta_1) + a_2 b_2 (1 + \delta_2))(1 + \delta_3) \\
&= a_1 b_1 (1 + \delta_1)(1 + \delta_3) + a_2 b_2 (1 + \delta_2)(1 + \delta_3).
\end{aligned}$$

Upon setting

$$\tilde{a_1} = a_1(1 + \delta_1), \quad \tilde{b_1} = b_1(1 + \delta_3), \quad \tilde{a_2} = a_2(1 + \delta_2), \quad \tilde{b_2} = b_2(1 + \delta_3),$$

we obtain

$$\mathrm{fl}(\mathbf{a} \cdot \mathbf{b}) = \tilde{\mathbf{a}} \cdot \tilde{\mathbf{b}},$$

where $\|\mathbf{a} - \tilde{\mathbf{a}}\| = O(\varepsilon)$ and $\|\mathbf{b} - \tilde{\mathbf{b}}\| = O(\varepsilon)$. This is called *backward stability*: the numerically evaluated quantity is the exact quantity for an exact computation with perturbed data.

We can now turn to the first barycentric formula. First we consider the evalutation of a weight $\ell(x)$,

$$\mathrm{fl}\Big(\prod_{n=0}^{N} (x - x_n)\Big) = \mathrm{fl}\bigg(\mathrm{fl}\bigg(\prod_{n=0}^{N-1} (x - x_n)\bigg) * \mathrm{fl}(x - x_N)\bigg) \tag{4.14}$$

$$= \mathrm{fl}\bigg(\prod_{n=0}^{N-1} (x - x_n)\bigg) * (x - x_N)(1 + \delta_1)(1 + \delta_2), \tag{4.15}$$

and by induction

$$\mathrm{fl}\bigg(\prod_{n=0}^{N} (x - x_n)\bigg) = \ell(x) \prod_{m=1}^{2N+1} (1 + \delta_m). \tag{4.16}$$

The argument for λ_j is of course analogous, hence we obtain with a little extra work:

Proposition 4.13. *Let* {th:poly:barystab}

$$\tilde{p}_N(x) := \mathrm{fl}\bigg(\ell(x) \sum_{j=0}^{N} \frac{\lambda_j}{x - x_j}\bigg)$$

be the numerically evaluated polynomial in the standard model of floating point arithmetic, then

$$\tilde{p}_N(x) = \ell(x) \sum_{j=0}^{N} \frac{\lambda_j f_j}{x - x_j} \prod_{m=1}^{5N+5} (1 + \delta_{jm}).$$

Proof. This is a straightforward continuation of the calculations above. □

The key point of Theorem 4.13 is that this is a *backward stability* result, i.e., let $\tilde{f}_j = f_j \prod_{m=1}^{5N+5}(1 + \delta_{jm})$, then $\hat{p}_N$ is interpolates the values $\tilde{f}_j$. In particular, the error in the floating point polynomial $\hat{p}_N(x)$ is no larger than if we had small errors in the nodal values f_j, which we will normally have anyhow.

Finally, for the second barycentric formula, the numerical stability result is weaker, but one can still show that for interpolation nodes with moderate Lebesgue constant, and reasonable functions f that we are interpolating, the numerical stability is of no concern; see [Hig04] for more details.

4.5 Applications

The following applications of the theory in this chapter will be covered in `[Notebook 03]`.

- Evaluating special functions
- Approximating a Matrix function
- Spectral methods for BVPs; see also [Tre13, Sec. 21]

Further applications that could be explored in self-directed reading:

- Chebyshev filtering
- Conjugate gradients and other Krylov methods
- Quadrature: [Tre13],
- Richardson extrapolation: [Tre13], p. 258
- ...

4.6 Exercises

Exercise 4.1 (Interpolation: Existence and Uniqueness). Prove that for any collection of nodes $z_0, \dots, z_N \subset \mathbb{C}$ with $x_i \neq z_j$ for $i \neq j$, and nodal values f_j, there exists a unique interpolant $p \in \mathcal{P}_N$ such that $p(z_j) = f_j$. □ {exr:poly:interpunique}

Exercise 4.2 (Runge Phenomenon). For a partial explanation of the Runge phenomenon (cf `[Notebook 03]`) consider the following steps: {exr:poly:Runge Phenomenon}

(i) Suppose $f \in C^{N+1}([-1,1])$. Prove that there exists $\xi \in (-1,1)$ such that

$$f(x) - I_N f(x) = \frac{f^{(N+1)}(\xi)}{(N+1)!} \ell_N(x),$$

where $\ell_N(x)$ is the node polynomial for the interpolation points.

Hint: Let $e(t) = f(t) - I_N f(t)$ and show that $y(t) = e(t) - e(x)\ell(t)/\ell(x)$ has $N+2$ roots. What does this imply about the roots of $y^{(N+1)}$?

(ii) Prove that for equispaces nodes, $\|\ell_N\|_\infty \geq \frac{1}{4}(N/2)^{-N-1}(N-1)!$.

(iii) For $f(x) = 1/(1+25x^2)$ (The Witch of Agnesi), prove that $\|f^{(N+1)}\|_\infty \|\ell_N\|_\infty/(N+1)! \to \infty$ as $N \to \infty$. *[HINT: $(1+c^2x^2)^{-1} = (1+cix)^{-1} + (1-cix)^{-1}$]*

(Note this does not prove divergence but proved a strong hint why divergence occurs.)

□

Exercise 4.3 (Clenshaw's Algorithm). Let $p \in \mathcal{P}_N$, $N \geq 1$, be given in the Chebyshev basis, with coefficients $(\tilde{f}_k)_{k=0}^N$ and let $x \in [-1,1]$. Show that $p(x)$ can be evaluated by Clenshaw's algorithm: {exr:poly:clenshaw}

$$\begin{aligned}
&u_{N+1} = 0, \qquad u_N = \tilde{f}_N; \\
&u_n = 2x u_{n+1} - u_{n+2} + \tilde{f}_n, n = N-1, N-2, \dots, 0; \\
&p(x) = \tfrac{1}{2}(\tilde{f}_0 + u_0 - u_2).
\end{aligned}$$

What is the purpose of the Clenshaw algorithm, i.e., the potential advantage over simply summing over the Chebyshev basis? □

Exercise 4.4 (Orthogonality of T_k). Consider the weighted space

$$L^2_C := \big\{ f : (-1,1) \to \mathbb{R}, \text{ measurable}, \|f\|_{L^2_C} < \infty \big\}, \qquad \text{where}$$
$$\|f\|^2_{L^2_C} := \int_{-1}^1 \frac{|f|^2}{\sqrt{1-x^2}}\, dx.$$

Prove that L^2_C is a Hilbert space and show that the Chebyshev polynomials are (up to scaling) and orthonormal basis of this space.

Thus, conclude that the Chebyshev projection $\tilde{\Pi}_N$ is in fact that best-approximation with respect to the $\|\cdot\|_{L^2_C}$-norm. □

Exercise 4.5 (Bernstein Ellipse). Prove that the Bernstein Ellipse E_ρ, $\rho > 1$ is indeed an ellipse with centre $z = 0$, foci ± 1, semi-major axis $\frac{1}{2}(\rho + \rho^{-1})$ and semi-minor axis $\frac{1}{2}(\rho - \rho^{-1})$. □ {exr:poly:ellipse}

Exercise 4.6 (Convergence Bounds). {exr:poly:convergence}

(i) Complete the proof of (4.6) by proving

$$\|I_N\|_{L(L^\infty)} \le C \log N,$$

where I_N is the Chebyshev nodal interpolation operator.

(ii) In preparation for the proofs of the best approximation error estimates for differentiable (non-analytic) functions, prove that, if $f \in C([-1, 1])$ with modulus of continuity ω, then $g \in C(\mathbb{T})$ and it has the same modulus of continuity.

(iii) Prove Theorem 4.8, case $j = 0$.

(iv) Let $E_N(f) := \inf_{p \in \mathcal{P}_N} \|f - p\|_\infty$. Prove that

$$E_N(f) \le C N^{-1} E_{N-1}(f'),$$

where C is independent of N and try to quantify C.

(v) Complete the proof of Theorem 4.8 (general j). Indeed, you should obtain a more precise formula.

□

Exercise 4.7. {exr:poly:examplefunctions}

(i) For the following functions give bounds on the rate of polynomial best approximation in the max-norm, as sharp as you can manage:

(a) $f(x) = |\sin(5x)|$
(b) $f(x) = \sqrt{|x|}$
(c) $f(x) = x(1 + 1000(x - 1/2)^2)^{-1/2}$
(d) $f(x) = e^{-\cos(3x)}$
(e) $f(x) = x^{100}$
(f) $f(x) = e^{-x^2}$
(g) $f(x) = \mathrm{sign}(x)$

(ii) and for the following two functions also in the L^2-norm:

- $f(x) = \mathrm{sign}(x)$
- $f(x) = \sqrt{|x|}$ □

Exercise 4.8 (Barycentric Chebyshev Interpolation). Let x_j be the Chebyshev points on $[-1, 1]$. {exr:poly:bary}

(i) In general (not only for Chebyshev points), demonstrate that the barycentric weights satisfy $\lambda_j = 1/\ell'(x_j)$.

(ii) Prove that the node polynomial satisfies

$$\ell(x) = 2^{-N}\big(T_{N+1}(x) - T_{N-1}(x)\big)$$

(iii) Show that

$$T'_{N+1}(x_j) - T'_{N-1}(x_j) = \begin{cases} 2N(-1)^j, & 1 \le j \le N-1, \\ 4N(-1)^j, & j = 0, N. \end{cases}$$

(iv) Deduce that, if λ_j is given by (4.12), then

$$\lambda_j = \frac{2^{N-1}}{N}(-1)^j, \qquad j = 1, \dots, N-1,$$

and suitably adjusted for $j = 0, N$. Explain why we can rescale the weights λ_j without changing the validity of the barycentric formula, and hence complete the proof of Theorem 4.12.

WARNING: it turns out, this exercise needs more material than I realised, namely aliasing of Chebyshev coefficients. It is still very interesting so I will leave it here for now. An interested reader should follow to [Tre13, Sec. 5] to derive this formula. □

NOTE: The last exercise is a bit tedious; it needs to be redesigned a bit. Maybe best leave it for now.

Exercise 4.9 (Coordinate Tranformations). The purpose of this exercise is to investigate how the choice of coordinate systems can expand the range of approximable functions, as well as have an affect on the rate of convergence. {exr:poly:coordinates}

The basic idea is to considier functions $F : [a, b] \to \mathbb{R}$ and via a coordinate transformation $f(x) = F(\xi(x))$ transform them to functions $f : [-1, 1] \to \mathbb{R}$. This can can multiple consequences, including: (1) we can represent functions on an arbitrary interval (inclusinf $\mathbb{R}$); (2) we can transform functions in such a way to increase the region of analyticity and thus accelerate convergence.

(i) Consider the Morse potential $F(y) = e^{-2\alpha y} - 2e^{-\alpha y}$, then $F(y) = f(e^{-\alpha y})$ where $f(x) = x^2 - 2x$ is a quadratic polynomial. Suppose though that this "optimal" coordinate transform $x = e^{-y}$ is not known.

Instead, consider the Morse coordinate transformation $\xi^{-1}(y) = 2e^{-y} - 1$ and the transformed function $f(x) = F(\xi(x))$.

coordinate transformation $x = 2/(1+y) - 1 = \xi^{-1}(y)$, that is, $\xi^{-1}(0) = 1, \xi^{-1}(\infty) = -1$, and let $f(x) = F(\xi(x))$.

(a) Establish an upper bound (as sharp as you can manage) for approximation by Chebyshev projection and interpolation of f on $[-1, 1]$ in the max-norm.

(b) Convert this bound to an approximation result for $F(y)$ on $[0, \infty)$.

(c) Can you give a simpler / more direct characterisation of the effective approximation space for functions on $[0, \infty)$ that you used here?

(ii) Now consider the function $F(y) = (\varepsilon^2 + y^2)^{-1/2}$ on $[-1, 1]$. Recall the rate of convergence of Chebyshev projection and Chebyshev interpolation.

Now consider a coordinate transformation

$$\xi^{-1}(y) = \frac{\arctan(x/\eta)}{\arctan(1/\eta)},$$

and explicitly compute its inverse. Show that $\xi, \xi^{-1} : [-1, 1] \to [-1, 1]$ are bijective.

(a) For any $\eta > 0$ establish an upper bound (as sharp as you can manage) for approximation of $f(x) = F(\xi(x))$ by Chebyshev projection and Chebyshev interpolation.

(b) Discuss which choices of η appear to be particularly good. Visualise the effect of ξ on the function f as well as on the singularities in the complex plane.

$\square$

Part II: Further Topics in Univariate Approximation

5 Splines

{sec:splines}

In this very short chapter we will briefly introduce and explore some consequences of piecewise polynomial approximation (as opposed to global polynomial approximation as in § 4). The basic results will be very easy to obtain. For lack of time we will skip the more interesting algorithmic aspects, in particular B-Splines (we will briefly define them and show some examples, but we won't go into the implementation details, at least not this year).

5.1 Motivation

{sec:splines:motivation}

Let us motivate the idea of splines as follows: consider the function $f(x) = \sqrt{x}$ on $[0, 1]$. After rescaling to $[-1, 1]$ we can approximate it with polynomials to obtain the convergence rate (cf. Jackson's Theorem 4.8)

$$\inf_{p \in \mathcal{P}_N} \|f - p\|_{L^\infty(0,1)} \lesssim N^{-1/2}.$$

This is a very slow rate of convergence, purely caused by the singularity at $x = 0$. But in $[1/2, 1]$ f is analytic and on that interval we would expect

$$\inf_{p \in \mathcal{P}_N} \|f - p\|_{L^\infty(1/2,1)} \lesssim \rho^{-N},$$

for some $\rho > 1$. We can then prescribe a second polynomial on $[1/4, 1/2]$, and so forth, thus obtaining a piecewise polynomial approximation. The subintervals $[1/2, 1], [1/4, 1/2], \dots$ are called a mesh and the flexibility in choosing these sub-intervals can lead to very strong results. We will later see that in this particular case we obtain almost exponential convergence.

5.2 Splines for C^j functions

{sec:splines:Cj}

To work with splines we will need to construct polynomial approximations on arbitrary subintervals $[a, b] \subset \mathbb{R}$. The Chebyshev nodes on $[a, b]$ are simply the rescaled nodes

$$x_j^{[a,b]} = a + \frac{(x_j + 1)(b - a)}{2},$$

where x_j are the Chebyshev nodes on $[-1, 1]$. The resulting interpolation operator is denoted by $I_N^{[a,b]}$.

We can now quantify the effect of domain size with the following lemma.

Lemma 5.1. *Let $f \in C^{p-1,1}([a, b])$ where $a < b$, and $N \leq p$, then*

$$\|f - I_N^{[a,b]} f\|_{L^\infty(a,b)} \leq \frac{c^N \log N}{N!} \big(b - a\big)^N \|f^{(N)}\|_{L^\infty(a,b)},$$

where c is a generic constant.

Proof. Let $g(y) = f(\xi(y))$ where $\xi(y) = a + (b - a)(1 + y)/2$, i.e.,

$$\xi : [-1, 1] \to [a, b]$$

is affine and bijective. Then according to Jackson's theorem (the sharp version; cf. Exercise 4.6),

$$\|f - I_N^{[a,b]} f\|_{L^\infty(a,b)} = \|g - I_N g\|_{L^\infty(-1,1)} \le \frac{c_1^N \log N}{N!} \|g^{(N)}\|_{L^\infty(-1,1)}.$$

Next, since ξ is affine it is easy to show that

$$g'(y) = f'(\xi(y))\xi'(y) = f'(\xi(y))\tfrac{b-a}{2},$$

and hence

$$g^{(j)}(y) = f^{(j)}(\xi(y))\Big(\tfrac{b-a}{2}\Big)^j.$$

Combining this with the interpolation error estimate for g yields the stated result. $\square$

Thus we see that we now have two parameters to control the approximation error: the polynomial degree N and the interval lengths $(b-a)$. This extra freedom is what can make splines a powerful alternative to polynomials.

Definition 5.2. *Let $y_0 < y_1 < \cdots < y_M$ be a partition of an interval $[y_0, y_M]$, then we define the space of splines (piecewise polynomials) of degree N on that partition to be*

$$\mathcal{S}_N(\{y_i\}) := \big\{ s : [y_0, y_M] \to \mathbb{R}, \quad s|_{[y_{m-1}, y_m]} \in \mathcal{P}_N \text{ for all } m = 1, \dots, M \big\}$$

Splines are of course C^∞ in each interval $[y_{j-1}, y_j]$, but sometimes it is also interesting to require that splines have a certain regularity on the entire interval $[y_0, y_M]$. We therefore define

$$\mathcal{S}_N^p(\{y_i\}) := \mathcal{S}_N(\{y_i\}) \cap C^p([y_0, y_M]).$$

It is worth nothing that $s \in \mathcal{S}_N^p$ implies in fact that $s \in C^{p,1}$.

Remark 5.3. It is of course also possible to define splines with varying polynomial degree, i.e. in each subinterval $[y_{j-1}, y_j]$ we might impose a degree N_j. This has advantages for some applications but we will not consider it here. $\square$

It takes a bit more work to construct splines of regularity $p = 1$ or higher, but $\mathcal{S}_N^0$ splines are obtained by simply taking Chebyshev interpolants on each sub-interval. We call the resulting interpolant $I_{N,M}$,

$$I_{N,M} f(x) := I_N^{[y_{m-1}, y_m]} f(x) \qquad \text{for } x \in [y_{m-1}, y_m].$$

We then obtain the following basic approximation error estimates.

Theorem 5.4. *Let $f \in C^p([a,b])$ and $a = y_0 < \cdots < y_M = b$ a partition of $[a,b]$, and let $h_m := y_m - y_{m-1}$) be the mesh size, and $N \le p$, then* {th:splines:convergence_Cj}

$$\|f - I_{N,M} f\|_{L^\infty(a,b)} \le C_N \max_{m=1,\dots,M} h_m^N \|f^{(N)}\|_{L^\infty(y_{m-1}, y_m)},$$

where $C_N = \frac{c^N \log N}{N!}$. In particular, if the partition is uniform, $y_m = a + hm$ where $h = (b-a)/M$ then

$$\|f - I_{N,M} f\|_{L^\infty(a,b)} \le C_N h^N \|f^{(N)}\|_{L^\infty(a,b)}.$$

Proof. Left as an exercise. $\square$

5.3 Splines for functions with singularities

{sec:splines:sing}

We will demonstrate how splines can be used to effectively resolve singular behaviour using the example from the beginning of this chapter,

$$f(x) = \sqrt{x} \qquad \text{on } x \in [0, 1]$$

A possible analytic continuation is given by

$$f(re^{i\varphi}) = \sqrt{r}e^{i\varphi/2},$$

which is analytic in $\mathbb{C} \setminus (-\infty, 0]$. Moreover, we have $|f(z)| = \sqrt{|z|}$ which will make it easy to estimate $\|f\|_{L^\infty(E_\rho)}$ where E_ρ will be some suitable Bernstein ellipsi.

Our strategy will be to use a partition

$$0, 2^{-M}, 2^{-M+1}, \dots, 2^{-1}, 1.$$

Since f is analytic in each subinterval $[2^{-m}, 2^{-m+1}]$ we will be able to use the exponential convergence rates from Theorem 4.5.

Let us therefore consider f on $[2^{-m}, 2^{-m+1}]$. We rescale

$$g(y) = f\big(2^{-m} + 2^{-m-1}(1 + y)\big),$$

then the singularity $x = 0$ maps to $y = -3$, hence g in analytic in $\Re z > -3$. In particular taking $\rho = 4$ we have $a = \frac{1}{2}(\rho + \rho^{-1}) < 3$ and

$$\begin{aligned}
\|g\|_{L^\infty(E_\rho)} &\le g(a) \le f(2^{-m} + 2^{-m-1}(1 + a)) \\
&\le f(2^{-m} + 2^{-m+1}) \\
&\le \sqrt{2^{-m+2}} \\
&= 2^{-m/2+1}.
\end{aligned}$$

Thus, we obtain

$$\|f - I_N^{[2^{-m},2^{-m+1}]} f\|_{L^\infty(2^{-m},2^{-m+1})} = \|f - I_N g\|_{L^\infty(-1,1)} \le C4^{-N}2^{-m/2}$$

To make our life a little easier we can just estimate

$$\|f - I_N^{[2^{-m},2^{-m+1}]} f\|_{L^\infty(2^{-m},2^{-m+1})} \le C4^{-N} \qquad \text{for } m = M, M-1, \dots, 1;$$

that is,

$$\|f - I_{N,M} f\|_{L^\infty(2^{-M},1)} \le CN^{-4}.$$

Finally, we address the first interval $[0, 2^{-M}]$. We rescale again as before, but now the singularity becomes part of the domain $[-1, 1]$, i.e., $g \in C^{0,1/2}([-1, 1])$ and no better. Jackson's theorem therefore tells us the

$$\|g - I_N g\|_{L^\infty(0,2^{-M})} \le C\omega_g(N^{-1}) = CN^{-1/2}.$$

But the constant matters here! Specifically, we can show that

$$\omega_g(r) = c2^{-M/2}\sqrt{r},$$

that is, we even have

$$\|f - I_N^{[0,2^{-M}]} f\|_{L^\infty(0,2^{-M})} \le C 2^{-M/2} N^{-1/2}.$$

Let us again make our life a little easier and ignore the $N^{-1/2}$ term, then we want to balance $2^{-M/2} = 4^{-N}$; that is,

$$M = 4N.$$

With this choice, we finally obtain

$$\|f - I_{N,M} f\|_{L^\infty(0,1)} \le C 4^{-N}.$$

To conclude we convert this into a cost estimate. The cost of evaluating $I_{N,M} f$ at a single point in space is the same as evaluating a polynomial of degree N, that is

$$\mathrm{COST} - \mathrm{EVAL}(I_{N,M} f) = O(N)$$

and in particular, we obtain the very nice exponential convergence result

$$\|f - I_{N,M} f\|_{L^\infty(0,1)} \le C \rho^{-\mathrm{COST}-\mathrm{EVAL}},$$

for some $\rho > 0$. The cost to "build and store" $I_{N,M} f$ is the cost of evaluating f at $M \cdot N$ points, i.e., $O(N^2)$ so this cost is a little higher, but still very attractive.

This example is intended to demonstrate the power of adapting the spline grid to the features of the function to be approximated. Automating this process is of great interest but goes beyond the scope of this module.

Remark 5.5. We can do slightly better by balancing the two terms in

$$\|f - I_N^{[2^{-m}, 2^{-m+1}]} f\|_{L^\infty(2^{-m}, 2^{-m+1})} \le C 4^{-N_m} 2^{-m/2} = C 4^{-N_m - m/4},$$

i.e., choosing $N_m + m/4 = N = \text{const}$. But one can easily check that this only gives an improvement in some constants, but not qualitatively. □

5.4 Exercises

Exercise 5.1. Prove Theorem 5.4 □ {exr:splines:}

Exercise 5.2.

(i) Suppose you are given a function $f \in C^{p-1,1}([-1,1])$. For simplicity, assume even that in each subinterval $[a,b] \subset [-1,1]$ the regularity of f is no better than $C^{p-1,1}$. Assume you discretise $[-1,1]$ with a uniform grid. How would you optimally balance the grid spacing h against the polynomial degree N? (i.e. minimise the error against the number of function evaluations you need to specify the approximant)

(ii) Now suppose that $f \in A([-1,1])$; how would you balance h against N now?

(iii) For the following functions compare the performance of global polynomial versus $\mathcal{S}_N^0$ approximation on a uniform grid:

- $f(x) = |x|$
- $f(x) = |x + \pi|$
- $f(x) = |\sin(x/2)|$
- $f(x) = (1 + 25x^2)^{-1}$
- $f(x) = x \sin(1/x)$

□

Exercise 5.3. For the following functions $f : [-1, 1] \to \mathbb{R}$, design a spline approximation with quasi-optimal rate of convergence in $\| \cdot \|_{L^\infty(-1,1)}$ in terms of evaluation cost.

- $f(x) = |x|$
- $f(x) = |\sin(x/2)|$
- $f(x) = (1 + 25x^2)^{-1}$
- $f(x) = x \sin(1/x)$

□

Exercise 5.4 (Linear Splines). Show that for we can write continuous linear spline interpolations, i.e. $s \in \mathcal{S}_1^0(\{y_m\})$ in terms of a nodal basis,

$$s(y) = \sum_{m=0}^{M} f(y_m)\phi_m(y),$$

where ϕ_m are "hat-functions" that you should specify explicitly. □

Exercise 5.5 (Hermite Interpolation with Cubic Splines). Let $y_0 < \cdots < y_M$ be a grid and let f_m, f'_m be function and derivative values at those grid points. Show that there exists a unique cubic spline $s \in \mathcal{S}_3^1(\{y_m\})$ such that

$$s(y_m) = f_m, \quad \text{and} \quad s'(y_m) = f'_m \quad \text{for } m = 0, \dots, M.$$

HINT: in each interval $[y_m, y_{m+1}]$ write $s(x) = f_m + f'_m(x - x_m) + a_m(x - x_m)^2 + b_m(x - x_m)^3$ and show that there exist unique a_m, b_m such that $s(x_{m+1}) = f_{m+1}, s'(x_{m+1}) = f'_{m+1}$. You may wish to derive explicit expressions for a_m, b_m in preparation for the next exercise. □

Exercise 5.6 (B-Splines). Depending on regularity requirements of an application it is sometimes advantageous to require higher regularity of the approximant, i.e., we should consider $\mathcal{S}_N^p$, $p > 0$. The case $\mathcal{S}_N^{N-1}$ turns out to be particularly natural; these are alled the B-splines. And amongst those, the cubic splines enjoy particular polularity.

(i) Suppose for the moment that $s \in \mathcal{S}_3^2(\{y_m\})$ with $s(y_m) = f_m$ where f_m are some nodal values. Prove that, for *any* $g \in C^2[a, b]$ with $g(y_m) = f_m$,

$$\int_a^b |s''(x)|^2 \, dx \le \int_a^b |g''(x)|^2 \, dx,$$

provided that s satisfies a condition at the end-points $a = y_0, b = y_M$, which you should derive.

Thus, s'' with this end-point condition minimises curvature amongst all C^2 functions satisfying the nodal interpolation conditions. These splines are therefore called natural splines.

HINT: Consider $\int_a^b |s''|^2 + 2s''(g'' - s'') + |s'' - g''|^2 \, dx$ and show that the middle term vanishes if the correct end-point condition is applied.

(ii) Given $(f_m)_{m=0}^M \in \mathbb{R}^{M+1}$, prove that there exists a unique $s \in \mathcal{S}_3^2(\{y_m\})$ satisfying the nodal interpolation conditions $s(y_m) = f_m$ and the end-point conditions found in part (ii). For the sake of simplicity you may wish to assume that the nodes are equispaced, i.e. $y_m = y_0 + hm$.

HINT: Prescribe artificial derivative values f'_m, then derive a tridiagonal linear system for $(f'_m)_{m=0}^M$ and show that it has a unique solution. Note that this system can be solved in $O(M)$ time. $\square$

6 Least Squares Methods

6.1 Motivation

We first describe least squares methods in abstract terms. Let $[a, b]$ be an interval and $b_1, \dots, b_N \in C([a, b])$ be N linearly independent basis functions for an approximation space

$$A_N := \mathrm{span}\big\{b_1, \dots, b_N\big\}.$$

Given $w \in C(a, b) \cap L^1(a, b)$ (note the open interval!) we can define a weighted L^2-inner product

$$\langle f, g \rangle_{L^2_w} := \int_a^b w(x) f(x) g(x)^* \, dx,$$

with associated norm $\|f\|_{L^2_w} := \langle f, f \rangle_{L^2_w}^{1/2}$. The best approximation of a function $f \in C([a, b])$ with respect to this weighted norm is then given by

$$p_N \in \min_{p \in A_N} \big\|f - p\big\|_{L^2_w}^2. \tag{6.1}$$ {{eq:lsq:ctslsq}}

We call this a continuous least squares problem.

Computationally, we typically need to discretise (6.1). To that end, we choose points $x_1, \dots, x_M \in [a, b]$ and weights $w_1, \dots w_M$ and define the discrete inner product

$$\langle f, g \rangle_{\ell^2_w} := \sum_{m=1}^M w_m f(x_m) g(x_m)^* \tag{6.2}$$ {{eq:lsq:disip}}

with associated norm $\|f\|_{\ell^2_w} := \langle f, f \rangle_{\ell^2_w}^{1/2}$. This gives the discrete least squares problem

$$p_N \in \min_{p \in A_N} \big\|f - p\big\|_{\ell^2_w}. \tag{6.3}$$ {{eq:lsq:dislsq}}

This is the typical kind of least squares problem encountered in real applications.

We distinguish two scenarios:

1. **User Chooses Data:** In this scenario the "user" is given a function f to be approximated. She may then choose the points x_m, weights w_m and evaluations $f(x_m)$ in order to fine-tune and optimise the fit p_N. For example it is then feasible to start from (6.1) and design a discrete LSQ system (6.3) that approximates (6.1) in a suitable sense. An arbtirary amount of data $(x_m, f(x_m))$ may then be generated to ensure a good fit.

2. **Data is provided:** Some data has been collected outside the control of the person ("user") designing the fit. Given the data points $(x_m, f(x_m))$ (possibly subject to noise, i.e. $y_m = f(x_m) + \eta_m$ is then provided) one then needs to choose an appropriate approximation space A_N, approximation degree N and weights w_m to ensure a good fit in a sense dictated by the application.

We will study both scenarios but note that the second one is the more typical in applications.

6.2 Solution methods

{sec:lsq:soln}

We convert (6.3) into a linear algebra problem. By writing

$$Y_m := f(x_m), \sqrt{W} := \mathrm{diag}(\sqrt{w_1}, \dots, \sqrt{w_m}) \in \mathbb{R}^{M \times M}$$

and

$$p(x_m) = \sum_{n=1}^{N} c_n b_n(x_m) = Ac, \quad \text{where } A_{mn} = b_n(x_m),$$

then $A \in \mathbb{R}^{M \times N}$ and we obtain

$$\sum_{m=1}^{M} w_m |p(w_m) - f(x_m)|^2 = \big\|\sqrt{W}Ac - \sqrt{W}Y\big\|^2,$$

where $\|\cdot\|$ denotes the standard Euclidean norm in $\mathbb{R}^M$. We write $\tilde{A} := \sqrt{W}A, \tilde{Y} := \sqrt{W}Y$ and write the least squares functional equivalently as

$$\Phi(c) := \big\|\sqrt{W}Ac - \sqrt{W}Y\big\|^2 = c^T \tilde{A}^T \tilde{A} c - 2c^T \tilde{A}^T \tilde{Y} + \|\tilde{Y}\|^2.$$

A minimiser must satisfy $\nabla\Phi(c) = 0$, which gives the linear system

$$\tilde{A}^T \tilde{A} c = \tilde{A}^T \tilde{Y}. \tag{6.4}$$

{{eq:lsq:normaleqns}}

These are called the normal equations, which can be solved using the LU or Cholesky factorisation.

It turns out that they are often (though not always) ill-conditioned. An alternative approach is therefore to perform the (numerically stable) *thin QR factorisation*

$$\tilde{A} = QR,$$

where $R \in \mathbb{R}^{N \times N}$ is upper triangular and $Q \in \mathbb{R}^{M \times N}$ has ortho-normal columns, i.e., $Q^T Q = I \in \mathbb{R}^{N \times N}$. With the QR factorisation in hand the normal equations can be rewritten as

$$\begin{aligned} \tilde{A}^T \tilde{A} c &= \tilde{A}^T \tilde{Y} \\ \Leftrightarrow \qquad R^T Q^T Q R c &= R^T Q^T \tilde{Y} \\ \Leftrightarrow \qquad R c &= Q^T \tilde{Y}, \end{aligned}$$

provided that R is invertible (which is equivalent to $A^T A$ being invertible and to A having full rank). Thus, the solution of the least squares problem becomes

$$Rc = Q^T \sqrt{W} Y, \qquad \text{where} \qquad \sqrt{W} A = QR. \tag{6.5}$$

{{eq:lsq:qr}}

It is worthwhile comparing the computational cost of the two approaches.

1. The assembly of thenormal equations requires the multiplication $A^T A$ which requires $O(MN^2)$ operations, followed by the Cholesky factorisation of $A^T A$ which requires $O(N^3)$ operiations. Thus the cost of solving (6.4) is $O(MN^2)$.

2. The cost of the QR factorisation in (6.5) is $O(MN^2)$ as well, while the inversion of R is only $O(N^2)$ and the multiplication with Q^T is $O(NM)$.

   Thus both algorithms scale like $O(MN^2)$.

6.3 Orthogonal Polynomials

{sec:lsq:orthpolys}

We have so far encountered orthogonal polynomials in the context of the Chebyshev basis, which arise naturally due to their connection to trigonometric polynomials. More generally, we can consider orthogonal polynomials with respect to *any* inner product $\langle \cdot, \cdot \rangle$. For simplicity we will continue to work on the domain $[-1, 1]$. In the context of least squares problems, we can think of (6.1) or (6.3) and the inner continuous or discrete products associated with these least squares problems.

The main result we want to discuss here is that the three-point recursion (4.3) for the Chebyshev basis is not special, but that all families of orthogonal polynomials satisfy such a recursion. That is, given an inner product $\langle \cdot, \cdot \rangle$ we will construct sequences of coefficients, A_k, B_k, C_k such that the sequence of polynomials given by

$$\phi_{k+1}(x) := (x - B_k)\phi_k(x) - C_k \phi_{k-1}(x) \tag{6.6}$$

{{eq:lsq:general_3ptrec}}

are orthogonal. By construction, we immediately see that the leading term in ϕ_k is x^k; hence they also span the space of all polynomials.

Taking the inner product of (6.6) with ϕ_k and then ϕ_{k-1} we obtain

$$\begin{aligned} 0 &= \langle x\phi_k, \phi_k \rangle - B_k \langle \phi_k, \phi_k \rangle, \\ 0 &= \langle \phi_k, x\phi_{k-1} \rangle - C_k \langle \phi_{k-1}, \phi_{k-1} \rangle, \end{aligned}$$

which gives expressions for B_k, C_k,

$$\begin{aligned} B_k &:= \frac{\langle x\phi_k, \phi_k \rangle}{\langle \phi_k, \phi_k \rangle}, \\ C_k &:= \frac{\langle \phi_k, x\phi_{k-1} \rangle}{\langle \phi_{k-1}, \phi_{k-1} \rangle}. \end{aligned} \tag{6.7}$$

{{eq:lsq:coeffs_3pt}}

It is worth noting that this construction is simply the Gram-Schmidt procedure, but truncated at a three-term recursion rather than the full recursion to ϕ_0. In particular, by construction, we have that $\phi_{k+1} \perp \phi_k, \phi_{k-1}$ and it only remains to show that they it is also orthogonal to $\phi_0, \dots, \phi_{k-2}$. Concretely we obtain the following result.

Proposition 6.1. *Suppose that $\langle \cdot, \cdot \rangle$ is an inner product on the space of polynomials such that the operator $p \mapsto x \cdot p$ is self-adjoint, (i.e., $\langle xp, q \rangle = \langle p, xq \rangle$ for all polynomials p, q). Suppose, moreover, that $\phi_0 = 1, \phi_1 = x - \langle 1, x \rangle$, and that $\phi_k, k \geq 2$, is given by the three-point recursion (6.6) with coefficients (6.7). Then $\{\phi_k : k \in \mathbb{N}\}$ is a basis of the space of polynomials which is orthogonal with respect to $\langle \cdot, \cdot \rangle$.*

Proof. By construction we have that $\phi_1 \perp \phi_0$ and that $\phi_{k+1} \perp \phi_k, \phi_{k-1}$ for $k \geq 2$. Thus, it only remains to prove that, for $k \geq 2$, $\phi_{k+1} \perp \phi_j$ for $j = 0, \dots, k-2$.

By induction we may assume that $\langle \phi_j, \phi_i \rangle = 0$ for $i \neq j$ and $i \leq k$. Then, we have

$$\begin{aligned} \langle \phi_{k+1}, \phi_j \rangle &= \langle x\phi_k, \phi_j \rangle - B_k \langle \phi_k, \phi_j \rangle - C_k \langle \phi_{k-1}, \phi_j \rangle \\ &= \langle \phi_k, x\phi_j \rangle, \end{aligned}$$

where we also used self-adjointness of multiplication by x. Since the degree of $x\phi_j$ is at most $k-1$ and, again by induction, ϕ_k is orthogonal to $\phi_0, \dots, \phi_{k-1}$ it follows that $\langle \phi_k, x\phi_j \rangle = 0$. This completes the proof. $\square$

Exercise 6.1. Derive a recursion for an *orthonormal* basis of the form

$$\begin{aligned}
A_0\phi_0 &= 1, \\
A_1\phi_1 &= x - B_1, \\
A_k\phi_k &= (x - B_k)\phi_{k-1} - C_k\phi_{k-2}.
\end{aligned}$$

Make sure to prove that all A_k are non-zero. $\square$

Exercise 6.2. Consider the inner product

$$\langle p, q\rangle = \int_{-1}^{1} pq + p'q'\,dx;$$

prove that the multiplication operator $p \mapsto xp$ is not self-adjoint.

If we were to construct a sequence of orthogonal polynomials by the Gram-Schmidt procedure, would we again obtain a three-term recursion?

Hint: The involved calculations are somewhat boring. You may wish to use a computer algebra system to explore this question. $\square$

Remark 6.2. A discrete inner product of the form (6.2) is not strictly an inner product on the space of all polynomials, but depending on the summation points it may be an inner product on a subspace $\mathcal{P}_N$. In this case the recursion formula can simply be terminated at degree $k = N$ to obtain an orthogonal (or orthonormal) basis of $\mathcal{P}_N$. $\square$

6.4 Accuracy and Stability I: Least Squares and Nodal Interpolation

Consider fitting trigonometric polynomials $A_{2N} = \mathbb{T}'_N$ with equispaced grid points $x_n = \pi n/N$ and uniform weights $w_n = 1$. Then the least squares fit

$$\min \sum_{n=0}^{2N-1} |f(x_n) - t(x_n)|^2$$

is equivalent to trigonometric interpolation, for which we have sharp and error estimates that predict a close to optimal rate of approximation.

We could leave it at this, but it is still interesting to observe what happens to the least squares system. The matrix A is now given by

$$A_{nk} = e^{ikx_n}$$

and the entries in the normal equation by

$$[A^*A]_{kk'} = \sum_n e^{-ikx_n}e^{ik'x_n} = 2N\delta_{kk'}$$

according to Exercise 3.9(i). This is due to the fact that the discrete inner product (6.2) (up to a constant factor) identical to the $L^2(\mathbb{T})$-inner product on the space $\mathbb{T}'_N$, that is,

$$\langle f, g\rangle_{\ell^2} = 2N \fint_{-\pi}^{\pi} fg^*\,dx \qquad \forall f, g \in \mathbb{T}'_N.$$

No QR factorisation is needed and the lsq fit is given by

$$c = (2N)^{-1} A^* Y,$$

where the operation $(2N)^{-1} A^T Y$ can be performed at $O(N \log N)$ computational cost using the FFT.

Analogous observations are of course true for connecting least squares methods and algebraic polynomials.

6.5 Accuracy and Stability II: Random data

{sec:lsq:rand}

The situation gets more interesting when we are not allowed to optimise the points at which to fit the approximant. There is an infinite variety of different situations that can occur when the provided data is application driven, which goes far beyond the scope of this module. Here, we will assume that the points x_m are distributed according to some probability law. That is, they are random. This is in fact a rather common situation in applications as well. Note also that we are now in the Case-2 situtation where we are given a fixed amount of data $(x_m, f(x_m))_{m=1}^M$ and should choose N ensure the best possible fit given the data we have. In particular this means that we should not choose N too large!

Specifically, we shall assume throughout this section that

$$x_m \sim w\,dx, \qquad \text{are iid}, \quad \text{for } m = 1, \dots, M. \tag{6.8}$$

{{eq:lsw:wm_law}}

(identically and independently distributed) and without loss of generality that $\int_a^b w\,dx = 1$ as this can always be achieved by rescaling. We also assume that $w \in C(a,b) \cap L^1(a,b)$ as before. In this case, we can construct a sequence of orthogonal polynomials as in § 6.3 with respect to the L^2_w-inner product and we will target best approximation with respect to the same inner product.

We will discuss two fundamental results due to Cohen, Davenport and Leviatan [CDL13], but we won't prove them. The first result concerns the *stability* of the normal equations. Specifically, we will show that if we use an L^2_w-orthogonal basis then $A^T A$ will be close to identity (and in particular invertible) for a sufficiently large number of sample points x_m.

Theorem 6.3 (Stability). *Let $\phi_1, \dots, \phi_N$ be L^2_w-orthonormal, $A_{mk} := \phi_k(x_m)$, then* {th:lsq:randstab}

$$\mathbb{P}\Big[\|A^* A - I\|_{\rm op} > \delta\Big] \leq 2N \exp\Big(-\tfrac{C_\delta M}{K(N)}\Big),$$

where $C_\delta = (1+\delta)\log(1+\delta) - \delta$ and

$$K(N) = \sup_{x \in [a,b]} \sum_{k=1}^N |\phi_k(x)|^2.$$

Let us specifically focus on the Chebyshev measure where $w(x) = (1-x^2)^{-1/2}$, the Chebyshev basis $T_k(x)$ on the interval $[a,b] = [-1,1]$. Since $|T_k(x)| \leq 1$ it readily follows that $K(N) \leq N$. Moreover, the recursion formulat for T_k implies that $T_k(1) = 1$ for all k, hence this bound is sharp, i.e., $K(N) = N$ in this case.

To make $N\exp(-\frac{C_\delta M}{N})$ small, we therefore need to choose $N \le \alpha M/\log M$. With this choice,

$$N\exp(-\tfrac{C_\delta M}{N}) = N\exp\big(-\alpha C_\delta \log M\big) \le M^{1-\alpha C_\delta}/\log M$$

and by choosing α sufficiently large we can ensure that this value tends to zero as $M \to \infty$. (the case of sufficiently large amounts of data). Conversely, if α is too small, then $M^{1-\alpha C_\delta}/\log M \to \infty$ as $M \to \infty$ which shows that the choice $N \le \alpha M/\log M$ is sharp. This is a very mild restriction!

The next result we discuss concerns the approximation $p_{NM} = \sum_n c_n\phi_n$ we obtain by solving the least squares problem $\|f - p_{NM}\|^2_{\ell^2(\{x_m\})} \to \min$.

Theorem 6.4. *There exists a constant c such that, if* {th:lsq:randerr}

$$K(N) \le \frac{c}{1+r}\frac{M}{\log M},$$

then,

$$\mathbb{E}\big[\|f - p_{NM}\|^2_{L^2_w}\big] \le (1+\epsilon(M))\|f - \Pi_N f\|^2_{L^2_w} + 8\|f\|^2_{L^\infty(a,b)}M^{-r},$$

where $\epsilon(M) \to 0$ as $M \to \infty$, and Π_N denotes the best-approximation operator with respect to the L^2_w-norm.

As a first comment, we observe that our restriction $N \le \alpha M/\log M$ for sufficiently small α re-appears. (In fact, this prerequisite is required to be able to apply Theorem 6.3.)

We can now ask what consequence the choice $N = \alpha M/\log M$ has on the error. In this case, $\alpha = c/(1+r)$, or equivalently, $r = c/\alpha - 1$ hence (sacrifycing just a log-factor)

$$M^{-r} \le N^{-r} = N^{1-c/\alpha} =: N^{-\alpha'},$$

where $\alpha' > 0$ provided that α is chosen sufficiently small. In this case, we can conclude that

$$\mathbb{E}\big[\|f - p_{NM}\|^2_{L^2_w}\big] \lesssim \|f - \Pi_N f\|^2_{L^2_w} + N^{-\alpha'}$$

for some $\alpha' > 0$. Thus, for differentiable functions f, such a choice is quasi-optimal.

However, for analytic functions the rate in the error estimate is reduced. Let us assume that $f \in A(E_\rho)$, then $\|f - \Pi_N f\|_{L^2_w} \lesssim \rho^{-N}$ hence we must balance the two constributions

$$\rho^{-N} + M^{-r}.$$

We have already seen that $N = \alpha M/\log M$ leads to $\rho^{-N} \ll M^{-r}$, hence we instead attempt to choose $N = a(M/\log M)^\alpha$ for some $0 < \alpha < 1$, which gives

$$r = c'(M/\log M)^{1-\alpha}.$$

Thus, we wish to balance

$$\begin{aligned}&\exp\Big[-(\log\rho)N\Big] + \exp\Big[-r\log M\Big]\\ &= \exp\Big[-c'' M^\alpha(\log M)^{-\alpha}\Big] + \exp\Big[-cM^{1-\alpha}(\log M)^{-\alpha}\Big]\end{aligned}$$

We can now see that the two terms are balanced when $\alpha = 1/2$, that is, the quasi-optimal choice of N appears to be

$$N = a\big(M/\log M\big)^{1/2}.$$

This is also somewhat consistent with the observation that in the C^j case we need to decrease α for increasing j.

That said, we should remember that we have balanced an error estimate and not the actual error. At this point, it is highly advisable to test these predictions numerically, which is done in `[Notebook 06]`, where we see — for some limited examples — that the stability condition $N \lesssim M/\log M$ appears to be crucial but the stronger requirement $N \lesssim (M/\log M)^{1/2}$ seems to not be required.

In summary, the foregoing analysis is intended to demonstrate how different competing contributions to approximation by fitting from data can be balanced at least in principle but also the limitations of analysis.

6.6 Exercises

{sec:lsq:exercises}

7 Nonlinear Approximation

{sec:nonlin}

7.1 Best polynomial approximation

{sec:poly:bestapprox}

Best approximation in Hilbert spaces is a linear operation, indeed an orthogonal projection. By contrast best approximation max-norms is far less trivial. This section concerns the best approximation of continuous functions with polynomials in the L^∞-norm. Although this norm is not strictly convex, it turns out that the best-approximant is still unique. Moreover, its characterisation leads to an algorithm (the Remez algorithm). The high cost of the Remez algorithm an together with the fact that Chebyshev interpolation (or projection) typically gives accuracy very close to best-approximation means this is rarely used in practise, however the mathematics is still interesting and worth studying. Moreover, this is our first non-trivial example of a *non-linear approximation algorithm.*

Theorem 7.1. *Let $f \in C([-1,1])$, then there exists a unique best approximation $p \in \mathcal{P}_N$ such that $\|f-p\|_\infty \le \|f-q\|_\infty$ for all $q \in \mathcal{P}_N$.* {th:poly:bestapprox}

A polynomial $p \in \mathcal{P}_N$ is the best approximation if and only if it equioscillates at (at least) $N+2$ points $y_0 < \cdots < y_{N+1}$; that is,

$$(f-p)(y_j) = \pm(-1)^j \|f-p\|_\infty.$$

Proof. test *1. Existence:* This is covered in Exercise 2.1. Let $E := \inf_{p\in\mathcal{P}_N} \|f-p\|_\infty$.

2. Equi-oscillation implies optimality: Suppose p satisfies the equi-oscillation property and $q \in \mathcal{P}_N$ such that $\|f-q\|_\infty < \|f-p\|_\infty$. Without loss of generality, we then have

$$\begin{aligned} (f-q)(y_j) &< (f-p)(x_j), \qquad j \text{ even}, \\ (f-q)(y_j) &> (f-p)(x_j), \qquad j \text{ odd}, \end{aligned}$$

and hence

$$\begin{aligned} (p-q)(y_j) &> 0, \qquad j \text{ odd}, \\ (p-q)(y_j) &< 0, \qquad j \text{ even}. \end{aligned}$$

Consequently $p-q$ has at least $N+1$ roots, which means that $p-q=0$.

3. Optimality implies equi-oscillation: Let $p \in \mathcal{P}_N$ and suppose there exist *at most* $M < N+2$ points $y_1 < \dots y_M$ at which $f-p$ equi-oscillates. Without loss of generality, assume that $(f-p)(y_1) = -E$, then we can find points

$$z_1 \in (-1, y_1), z_2 \in (y_1, y_2), \dots, z_M \in (y_{M-1}, y_M), z_{M+1} \in (y_M, 1)$$

such that

$$(f-p) < E, \qquad \text{in } [-1, z_1], [z_2, z_3], \dots (f-p) \qquad > E, \qquad \text{in } [z_1, z_2], [z_3, z_4], \dots$$

Now, let

$$\delta p(x) := (z_1 - x)(z_2 - x) \cdots (z_{M+1} - 1),$$

then we readily see that

$$\|f - (p + \varepsilon \delta p)\|_\infty < \|f - p\|_\infty \qquad \text{for } \varepsilon \text{ sufficiently small.}$$

Thus, p was not optimal.

4. Uniqueness: Suppose that p, q are both best approximations, then $r := (p+q)/2$ is a best approximation as well. Let y_j be the equi-oscillation points. $|r(y_j)| = E$ is only possible if $p(y_j) = q(y_j) = \pm E$. Thus q, p agree at $N+2$ points and are therefore equal. $\square$

Interestingly, then proof is semi-constructive and with a bit of immagination gives rise to the following (not quite an) algorithm:

Remez Algorithm: Input: f, N

1. Choose initial interpolation nodes $x_0 < \cdots < x_N$. E.g., Chebyshev nodes are a canonical choice.

2. Solve the system

$$b_0 + b_1 x_j + \cdots + b_N x_j^N + (-1)^j E = f(x_j), \qquad j = 0, \dots, N+1$$

for the $n+2$ unknowns b_i, E.

3.

7.2 Rational Approximation by Example

7.3 Rational Approximation by Iteratively Reweighted Least Squares

7.4 The AAA Algorithm

Part III: Approximation in High Dimension

8 Tensor Products and Sparse Grids

{sec:sparse}

8.1 Introduction and Motivation

{sec:sparse:intro}

We now consider the approximation of functions $f : [a,b]^d \to \mathbb{R}$. We will explore (1) how to construct "good" approximations and (2) what effect the dimension d has on the rates of approximation.

We will build multi-variate approximations from tensor products of uni-variate basis functions. For example, a polynomial in two dimensions can be written as

$$p(x_1, x_2) = \sum_{k_1=0}^{N} \sum_{k_2=0}^{N} c_{k_1 k_2} x_1^{k_1} x_2^{k_2}.$$

The function $(x_1, x_2) \mapsto x_1^{k_1} x_2^{k_2}$ is called the tensor product between the two functions $x_j \mapsto x_j^{k_j}$.

We know of course from our univariate theory that the Chebyshev basis has many advantageous algorithmic and approximation properties, thus we may prefer to write $p(x_1, x_2)$ in the form

$$p(x_1, x_2) = \sum_{k_1=0}^{N} \sum_{k_2=0}^{N} \tilde{c}_{k_1 k_2} T_{k_1}(x_1) T_{k_2}(x_2).$$

In a general dimension $d > 1$ we will write

$$\begin{aligned}
\boldsymbol{x} &= (x_1, \dots, x_d), \\
\boldsymbol{k} &= (k_1, \dots, k_d), \\
T_{\boldsymbol{k}}(\boldsymbol{x}) &= \prod_{\alpha=1}^{d} T_{k_\alpha}(x_\alpha),
\end{aligned}$$

and consider polynomials

$$p(\boldsymbol{x}) = \sum_{\boldsymbol{k} \in \mathcal{K}} c_{\boldsymbol{k}} T_{\boldsymbol{k}}(\boldsymbol{x}),$$

where $\mathcal{K} \subset \mathbb{N}^d$ is a suitable index set.

The definition of $T_{\boldsymbol{k}}$ can also equivalently be written as

$$T_{\boldsymbol{k}} = T_{k_1} \otimes T_{k_2} \otimes \cdots \otimes T_{k_d}$$

However we write these multi-variate polynomials we can see the "curse of dimensionality" creep in already: the number of coefficients to represent a d-variate polynomial of degree N is clearly $(1+N)^d$. Suppose our computational budget is one million coefficients (quite a lot!) and we are working in 10 dimensions. What is the maximal degree we may choose? But we will see that the notion of degree has no unique extension to higher dimensions and this will sometimes save us.

Before we proceed, we need to introduce some minimal amount of additional notation.

8.2 The Curse of Dimensionality

{sec:sparse:curse}

As a first attempt at constructing high-dimensional approximations we apply the one-dimensional techniques in each coordinate direction. For example, let us assume that $f : [-1,1]^2 \to \mathbb{R}$ has any regularity we may later need, then we can first apply a Chebyshev interpolant in the x_1-direction which we denote by $I_N^{(1)}$ which becomes a polynomial in x_1 but a general function in x_2, i.e.,

$$I_N^{(1)} f(x_1, x_2) = \sum_{k_1=0}^{N} c_{k_1}(x_2) T_{k_1}(x_1).$$

We then apply an interpolant in the x_2-direction to obtain

$$I_N^{(2)} I_N^{(1)} f(x_1, x_2) = \sum_{k_1=0}^{N} I_N^{(2)} c_{k_1}(x_2) T_{k_1}(x_1) = \sum_{k_1,k_2=0}^{N} c_{k_1 k_2} T_{k_1(x_1)} T_{k_2}(x_2) = \sum_{\boldsymbol{k} \in \{0 \dots N\}^2} c_{\boldsymbol{k}} T_{\boldsymbol{k}}(\boldsymbol{x}).$$

More generally, we define the d-dimensional Chebyshev interpolation operator to be

$$I_N^{(1..d)} := I_N^{(d)} I_N^{(d-1)} \cdots I_N^{(1)}$$

To estimate the error committed, let us assume that $x_\alpha \mapsto f(\boldsymbol{x}) \in C^j([-1,1])$ for all α, then we can bound

$$\begin{aligned}
\big\| f - I_N^{(d)} I_N^{(d-1)} \cdots I_N^{(1)} f \big\|_\infty &\le \big\| f - I_N^{(d)} f \big\|_\infty + \big\| I_N^{(d)} f - I_N^{(d)} I_N^{(d-1)} \cdots I_N^{(1)} f \big\|_\infty \\
&\le C(\log N) N^{-j} \|\partial_{x_d}^j f\|_\infty + \big\| I_N^{(d)} \big[f - I_N^{(d-1)} \cdots I_N^{(1)} f \big] \big\|_\infty \\
&\le C(\log N) \Big\{ N^{-j} \|\partial_{x_d}^j f\|_\infty + \big\| f - I_N^{(d-1)} \cdots I_N^{(1)} f \big\|_\infty \Big\}.
\end{aligned}$$

Arguing by induction we obtain the following result.

Theorem 8.1. *(1) Let $x_\alpha \mapsto f(\boldsymbol{x}) \in C^j([-1,1])$ for $\alpha = 1, \dots, d$, then* {th:sparse:curse}

$$\big\| f - I_N^{(1..d)} f \big\|_\infty \le C(\log N)^d N^{-j} \sum_{\alpha=1}^{d} \|\partial_{x_\alpha}^j f\|_\infty$$

(2) Let $x_\alpha \mapsto f(\boldsymbol{x}) \in A(E_\rho)$ for some $\rho > 1$ and for all $\boldsymbol{x} \in [-1,1]^d - 1$, then

$$\big\| f - I_N^{(1..d)} f \big\|_\infty \le C M_f (\rho - 1)^{-d} (\log N)^d \rho^{-N} \sum_{\alpha=1}^{d} M^{(\alpha)},$$

where $M^{(\alpha)} = \|f\|_{L^\infty(E_\rho^{(\alpha)})}$, where $E_\rho^{(\alpha)} = [-1,1] \times \cdots \times [-1,1] \times E_\rho \times [-1,1] \times \cdots \times [-1,1]$ and the position of the Bernstein ellipse is in the αth coordinate.

Let us translate these estimates into cost-error relations. The number of basis functios for a d-dimensional tensor product basis of degree N is $(1+N)^d$ and this is directly proportional to the associated computational cost. Further, even though the terms $(\log N)^d$ are significiant they arise from sub-optimal estimates of the interpolation error (cf. [Tre13] for sharp estimates) hence we ignore them. Letting $\epsilon := \|f - I_N^{(1..d)} f\|_\infty$, we therefore obtain

$$\text{Cost} \approx N^d \lesssim \begin{cases} \epsilon^{d/j}, & \text{case (1);} \\ (\log \epsilon)^d, & \text{case (2).} \end{cases}$$

The exponential dependence of these estimates on d is what we call the "curse of dimensionality": without new ideas and new information it becomes exponentially harder to approximate functions in high dimension.

8.3 Chebyshev series and greedy approximation

{sec:sparse:chebseries}

We begin by making precise the notion of a multi-variate Chebyshev series, which has already been at the back of our mind since the beginning of § 8. To that end, we define the d-dimensional Chebvyshev space

$$L^2_{\rm C}([-1,1]^d) := \big\{ f : [-1,1]^d \text{measurable, and } \|f\|_{L^2_{\rm C}} < \infty \big\}, \qquad \text{where}$$

$$\|f\|_{L^2_{\rm C}}^2 := \int_{[-1,1]^d} |f(x)|^2 \prod_{\alpha=1}^d (1 - x_\alpha^2)^{-1/2} \, d\boldsymbol{x}.$$

Note how the weight in this integral is simply the tensor product of the univariate weights. Because of this, we have the orthogonality (Exercise: check this!)

$$\big\langle T_{\boldsymbol{k}}, T_{\boldsymbol{k}'} \big\rangle_{L^2_{\rm C}} = \delta_{\boldsymbol{k}\boldsymbol{k}'}. \tag{8.1}$$

{{eq:sparse:dDorth}}

That is, $\{T_{\boldsymbol{k}}\}_{\boldsymbol{k}\in\mathbb{N}^d}$ is an ortho-normal subset of $L^2_{\rm C}$. The result from the previous section moreover indicates density of their linear combinations (polynomials) and after making this precise we will obtain the following result:

Theorem 8.2. *(i) Let $f \in L^2_{\rm C}([-1,1]^d)$, then there exist coefficients $\tilde{f}_{\boldsymbol{k}}, \boldsymbol{k} \in \mathbb{N}^d$ such that*

{th:sparse:mvchebseries}

$$f = \sum_{\boldsymbol{k}\in\mathbb{N}^d} \tilde{f}_{\boldsymbol{k}} T_{\boldsymbol{k}},$$

where the convergence of the series in is the $L^2_{\rm C}$-norm.

(ii) Moreover, we have a multi-variate Plancherel Theorem

$$\|f\|_{L^2_{\rm C}}^2 = \sum_{\boldsymbol{k}\in\mathbb{N}^d} \big|\tilde{f}_{\boldsymbol{k}}\big|^2. \tag{8.2}$$

{{eq:sparse:plancherel}}

(iii) If $(\tilde{f}_{\boldsymbol{k}})_{\boldsymbol{k}\in\mathbb{N}^d} \in \ell^1(\mathbb{N}^d)$, then f is continuous and the convergence is in the max-norm (i.e., uniform in $[-1,1]^d$).

We can now translate the Hilbert-space best approximation results to the multivariate setting. For every finite set $\mathcal{K} \subset \mathbb{N}^d$ we have a Chebyshev series truncation

$$p_{\mathcal{K}} := \tilde{\Pi}_{\mathcal{K}} f := \sum_{\boldsymbol{k}\in\mathcal{K}} \tilde{f}_{\boldsymbol{k}} T_{\boldsymbol{k}}$$

The resulting error is

$$\|f - p_{\mathcal{K}}\|_{L^2_{\mathrm{C}}}^2 = \sum_{\boldsymbol{k} \in \mathbb{N}^d \setminus \mathcal{K}} |\tilde{f}_{\boldsymbol{k}}|^2$$

which we can minimise as follows:

1. Compute the coefficients $\tilde{f}_{\boldsymbol{k}}$ and order them by magnitude,

$$|\tilde{f}_{\boldsymbol{k}_1}| \geq |\tilde{f}_{\boldsymbol{k}_2}| \geq \dots$$

2. Given a "budget" M, let $\mathcal{K} := \{\boldsymbol{k}_1, \dots, \boldsymbol{k}_M\}$

3. Then, $p_{\mathcal{K}}$ is the *best M-term approximation* to f in the L^2_{C}-norm.

Remark 8.3. It is not clear at all that minimising the number of terms in $\mathcal{K}$ optimises the computational cost to evaluate $p_{\mathcal{K}}$ for a given target error, due to the fact that the basis functions are computed via a recursion formula, but there are far more signifiicant issues with computing with best M-term approximations, hence we will not explore this any further. □

By a similar mechanism we can also get an L^∞-error bound,

$$\|f - p_{\mathcal{K}}\|_\infty \leq \sum_{\boldsymbol{k} \in \mathbb{N}^d \setminus \mathcal{K}} |\tilde{f}_{\boldsymbol{k}}|,$$

using the fact that $|T_{\boldsymbol{k}}(\boldsymbol{x})| \leq 1$. However, it is not at all obvious yet whether this bound is close to optimal.

Proof of Theorem 8.2. We will use, without proof, the fact that continuously differentiable functions are dense in L^2_{C}. This can be proven e.g. by using a multi-variate Jackson-type approximation. Then Theorem 8.2 implies that polynomials are dense in L^2_{C}. The remaining statements are straightforward. □

8.4 Sparse Grids

{sec:sparse:sparse}

In practise it is rare (though not impossible) to have the explicit coefficients $\tilde{f}_{\boldsymbol{k}}$ available, which makes it exceedingly difficult to develop "greedy algorithms"; however, see [DeV98] for an extensive review article. However, one may have some more generic qualitative information, such as genuine multi-variate versions of C^j regularity or analyticity.

Before we motivate the next definition let us assume that f has not only univariate partial derivatives $\partial_{x_\alpha}^j f$ but also mixed derivatives, e.g., $\partial_{x_1}\partial_{x_2} f$. Unfortunately, this idea is difficult to motivate in the Chebyshev basis, so we temporarily switch to multi-variate trigonometric polynomials. That is,

$$f \in L^2(\mathbb{T}^d)$$

and the multi-variate Fourier series is given by

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{k} \in \mathbb{Z}^d} \hat{f}_{\boldsymbol{k}} e^{i\boldsymbol{k}\cdot\boldsymbol{x}},$$

where we note that

$$e^{i\boldsymbol{k}\cdot\boldsymbol{x}} = e^{i\sum_{\alpha=1}^{d} k_\alpha x_\alpha} \prod_{\alpha=1}^{d} e^{ik_\alpha x_\alpha},$$

i.e. this is again precisely the same setting as before.

Assuming that $f : \mathbb{T}^2 \to \mathbb{R}$ has two mixed derivatives, then just calculating formally,

$$\begin{aligned}
\partial_{x_1}\partial_{x_2} f &= \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \hat{f}_{\boldsymbol{k}} \partial_{x_1} e^{ik_1x_1} \partial_{x_2} e^{ik_2x_2} \\
&= \sum_{\boldsymbol{k}\in\mathbb{Z}^2} -k_1k_2 \hat{f}_{\boldsymbol{k}} e^{i\boldsymbol{k}\cdot\boldsymbol{x}}.
\end{aligned}$$

Thus, if $\partial_{x_1}\partial_{x_2} f \in L^2(\mathbb{T}^2)$, then we have that

$$\sum_{\boldsymbol{k}\in\mathbb{Z}^2} |k_1|^2|k_2|^2|\hat{f}_{\boldsymbol{k}}|^2 < \infty.$$

This gives us some first information about the decay of $\hat{f}$ that we can exploit. Specifically, the best information we have is that we should choose index sets of the form

$$\mathcal{K}_N^{\mathrm{hc}} = \big\{|k_1||k_2| \leq N\big\}.$$

This is an example of the "hyperbolic cross" approximation. We will return to this idea again in a moment.

The foregoing example shows how genuine multi-variate regularity relates directly to decay of Fourier coefficients, and one can show more for $f : [-1,1]^d \to \mathbb{R}$ that it relates to the decay of Chebyshev coefficients. It is therefore expedient to define the following function classes:

$$\begin{aligned}
\mathcal{A}_\omega^{(2)} &:= \Big\{ f \in L^2_{\mathrm{C}} \Big| \sum_{\boldsymbol{k}\in\mathbb{N}^d} \omega(\boldsymbol{k})^2 |\tilde{f}_{\boldsymbol{k}}|^2 < \infty \Big\} \\
\mathcal{A}_\omega^{(\infty)} &:= \Big\{ f \in L^2_{\mathrm{C}} \Big| \sum_{\boldsymbol{k}\in\mathbb{N}^d} \omega(\boldsymbol{k}) |\tilde{f}_{\boldsymbol{k}}| < \infty \Big\}.
\end{aligned}$$

Informally, the "faster" $\omega(\boldsymbol{k})$ grows as $|\boldsymbol{k}| \to \infty$ the "smoother" is f. In the following we will focus on $\mathcal{A}_\omega^{(\infty)}$ but analogous results also hold for $\mathcal{A}_\omega^{(2)}$.

There are or course limitless possibilities, we will consider two specific cases:

$$(1) \quad \omega(\boldsymbol{k}) = \prod_{\alpha=1}^{d} (1+k_\alpha)^j, \qquad \text{and} \qquad (2) \quad \omega(\boldsymbol{k}) = \prod_{\alpha=1}^{d} \rho^{k_\alpha} = \rho^{\sum_\alpha k_\alpha},$$

where $j \geq 1$ and $\rho > 1$. The case (1) is related to mixed C^j regularity, while the case (2) corresponds to analyticity of f in E_ρ^d. Both cases can be extended in obvious ways to more anisotropic conditions, e.g., $\omega(\boldsymbol{k}) = \prod_{\alpha=1}^{d} \rho_\alpha^{k_\alpha}$ and similar.

Warning: Case (1) is in fact difficult to obtain practise except with very strongly dimension-dependent constants. In particular the hyperbolic cross approximation we derive from this below is not how it is used in practise. However, (1) is similar in spirit to $\omega(\boldsymbol{k}) = \prod_\alpha \max(1, k_\alpha)^j$, which leads to the *true* hyperbolic cross approximation.

As Chebyshev coefficients we choose the indices that are in the sublevel of ω, i.e.,

$$\mathcal{K}_N^{\mathrm{hc}} := \Big\{ \boldsymbol{k} \in \mathbb{N}^d \Big| \prod_{\alpha=1}^{d} (1 + k_\alpha) \le N \Big\}, \qquad \text{in Case (1);}$$

$$\mathcal{K}_N^{\mathrm{tot}} := \Big\{ \boldsymbol{k} \in \mathbb{N}^d \Big| \sum_{\alpha=1}^{d} k_\alpha \le N \Big\}, \qquad \text{in Case (2).}$$

Case (1) is called the *hyperbolic cross approximation* scheme, while Case (2) is called the *sparse grid approximation.* $\sum_\alpha k_\alpha$ is also called the *total degree* of the polynomial $T_{\boldsymbol{k}}$.

Theorem 8.4. *(1) Suppose that $f \in \mathcal{A}_\omega^{(\infty)}$ with $\omega(\boldsymbol{k}) = \prod_\alpha (1 + k_\alpha)^j$, then* {th:sparse:grids}

$$\|f - \tilde{\Pi}_{\mathcal{K}_N^{\mathrm{hc}}} f\|_\infty \le M_f N^{-j}, \qquad \textit{while} \tag{8.3}$$

{eq:sparse:hc_error}

$$\#\mathcal{K}_N^{\mathrm{hc}} \le N \log^{d-1} N. \tag{8.4}$$

{eq:sparse:hc_nbasis}

where $M_f := \sum_{\boldsymbol{k} \in \mathbb{N}^d} \omega(\boldsymbol{k}) |\tilde{f}_{\boldsymbol{k}}|$.

(2) Suppose that $f \in \mathcal{A}_\omega^{(\infty)}$ with $\omega(\boldsymbol{k}) = \rho^{|\boldsymbol{k}|_1}$ where $\rho > 1$, then

$$\|f - \tilde{\Pi}_{\mathcal{K}_N^{\mathrm{hc}}} f\|_\infty \le M_f \rho^{-N}, \qquad \textit{while} \tag{8.5}$$

{eq:sparse:sp_error}

$$\#\mathcal{K}_N^{\mathrm{tot}} = \binom{N+d}{d}, \tag{8.6}$$

{eq:sparse:sp_nbasis}

where $M_f := \sum_{\boldsymbol{k} \in \mathbb{N}^d} \omega(\boldsymbol{k}) |\tilde{f}_{\boldsymbol{k}}|$.

Let $\epsilon := \|f - \tilde{\Pi}_{\mathcal{K}_N^*} f\|_\infty$, then in case (1), hyperbolic cross, we obtain

$$\text{Cost} \approx \#\mathcal{K}_N^{\mathrm{hc}} \lesssim \epsilon^{1/j} \log^{d-1} \epsilon^{1/j}.$$

In case (2), sparse grid, the cost-error estimate is a little more involved. First, we use Stirling's formula to estimate

$$\begin{aligned}\binom{N+d}{d} &\approx \sqrt{\frac{2\pi(N+d)}{2\pi N 2\pi d}} \frac{((N+d)/e)^{N+d}}{(N/e)^N (d/e)^d} \\ &\lesssim \Big(1 + \frac{d}{N}\Big)^N \Big(1 + \frac{N}{d}\Big)^d.\end{aligned}$$

We distinguish two cases, $d \ll N$ and $N \ll d$:

$$\#\mathcal{K}_N^{\mathrm{tot}} \lesssim \begin{cases} e^d (1 + N/d)^d, & N \gg d, \\ e^N (1 + d/N)^N, & N \ll d. \end{cases}$$

In the $N \gg d$ case, which is the one more relevant for moderately high dimension we can now readily obtain

$$\text{Cost} \approx \#\mathcal{K}_N^{\mathrm{tot}} \lesssim \Big(\frac{c|\log \epsilon|}{d}\Big)^d,$$

which does not entirely remove the curse of dimensionality, but it substantially ameliorates it. Contrast this with the estimate for a tensor product basis, Cost $\approx |\log \epsilon|^d$.

It is somewhat striking that the "good" case when f is analytic uses less agressive sparsification and also ameliorates the curse of dimensionality to a lesser degree. That said, the rate of approximation is still better of course, and this can be seen in the occurance of $\log \epsilon$ instead of $\epsilon^{1/j}$ in the estimate.

8.4.1 Proofs

Throughout the following proofs we write $p_N := \tilde{\Pi}_{\mathcal{K}} f$ where $\mathcal{K} = \mathcal{K}_N^{\text{tot}}$ or $\mathcal{K} = \mathcal{K}_N^{\text{hc}}$. Further, we define Let $M_{Nd} := \#\mathcal{K}$ in dimension d.

Proof of (8.3).

$$
\begin{aligned}
\|f - p_N\|_\infty &\leq \sum_{\prod(1+k_\alpha)>N} |\tilde{f}_{\boldsymbol{k}}| \\
&\leq N^{-j} \sum_{\prod(1+k_\alpha)>N} \prod_\alpha (1+k_\alpha)^j |\tilde{f}_{\boldsymbol{k}}| \\
&\leq M_f N^{-j}.
\end{aligned}
$$

□

Proof of (8.4). For $d = 1$ we have $M_{N,1} = N + 1$. For $d > 1$ we can create a recursion

$$
\begin{aligned}
M_{N,d} &= \sum_{k_d=0}^{N} M_{\lfloor N/(1+k_d)\rfloor, d-1} \\
&\leq \sum_{k_d=0}^{N} \log^{d-1}\left[(N+2)/(1+k_d)\right] \frac{N+1}{1+k_d} \\
&\leq (N+1)\log^{d-2}(N+2) \sum_{k_d=0}^{N} \frac{1}{1+k_d} \\
&\leq (N+1)\log^{d-1}(N+2).
\end{aligned}
$$

□

Proof of 8.5. This is analogous as (8.3),

$$
\begin{aligned}
\|f - p_N\|_\infty &\leq \sum_{\sum k_\alpha > N} |\tilde{f}_{\boldsymbol{k}}| \\
&\leq \rho^{-N} \sum_{\sum k_\alpha > N} \rho^{|\boldsymbol{k}|_1} |\tilde{f}_{\boldsymbol{k}}| \\
&\leq M_f \rho^{-N}.
\end{aligned}
$$

□

Proof of (8.6). This is a simple combinatorics problem: The set $\mathcal{K}_N^{\text{tot}}$ can be interpreted as the set of all d-element multi-sets from $\{0, \ldots, N\}$, which gives the stated expression. □

References

[CDL13] Albert Cohen, Mark A Davenport, and Dany Leviatan. On the stability and accuracy of least squares approximations. *Found. Comut. Math.*, 13(5):819–834, October 2013.

[DeV98] Ronald A DeVore. Nonlinear approximation. *Acta Numer.*, 7:51–150, January 1998.

[Hig02] Nicholas J Higham. *Accuracy and Stability of Numerical Algorithms: Second Edition.* SIAM, January 2002.

[Hig04] Nicholas J Higham. The numerical stability of barycentric lagrange interpolation. *IMA J. Numer. Anal.*, 24(4):547–556, October 2004.

[Pow81] M J D Powell. *Approximation Theory and Methods.* Cambridge University Press, March 1981.

[Tre00] Lloyd N Trefethen. *Spectral Methods in MATLAB.* SIAM, January 2000.

[Tre13] Lloyd N Trefethen. *Approximation Theory and Approximation Practice.* SIAM, January 2013.