MA5233 Computational Mathematics

Lecture 4: Polynomial Approximation

Simon Etter

Semester I, AY 2020/2021

Polynomial Approximation

Problem statement

Given a finite approximation interval $[a, b]$, a function $f : [a, b] \to \mathbb{R}$ and a degree $n \in \mathbb{N}$, determine a polynomial p with degree$(p) \leq n$ such that

$$p(x) \approx f(x) \qquad \text{for all } x \in [a, b].$$

Why polynomial approximation?

Polynomials are a fundamental tool in computational mathematics. There are two main reasons why this is the case.

- Polynomials are simple.

  Polynomials require only addition and multiplication, which are the basic operations both in algebra and calculus (see next slide for concrete examples).

- Polynomials are complete.

  Weierstrass approximation theorem: Any continuous function $f : [a, b] \to \mathbb{R}$ with $b - a < \infty$ can be approximated arbitrarily closely by a polynomial of large enough degree.

Polynomial Approximation

Applications

- Evaluation of scalar functions.

  Addition and multiplication are in some sense the only operations we truly know how to do on a computer.

  See slide 30 in Lecture 1, which I added after class. (Sorry about that!)

  Polynomial approximation allows us to evaluate functions like exp, sin and cos by reducing them to a sequence of additions and multiplications.

- Evaluation of matrix functions.

  The solutions to many fundamental problems in computational mathematics can be written as a matrix function times a vector. Some examples:

  - Linear systems: $Ax = b \iff x = A^{-1}b.$
  - ODEs: $\dot{y} = Ay \iff y(t) = \exp(At)\, y(0).$

  Evaluating a matrix function like $A \mapsto A^{-1}$ is difficult, but approximating $p(x) \approx x^{-1}$ with $p \in \mathcal{P}_n$ and evaluating $p(A) \approx A^{-1}$ is easy.

Polynomial Approximation

Applications (continued)

- Differentiation and integration.

  Computing $\frac{d}{dx} f(x)$ or $\int f(x)\,dx$ for a general $f(x)$ is difficult, but computing $\frac{d}{dx} p(x)$ or $\int p(x)\,dx$ for a polynomial $p(x)$ is easy.

- Representation of unknown functions.

  The solutions to ODE and PDE problems are functions. Polynomials provide a simple tool for representing these solutions.

Polynomial Approximation

Polynomial approximation vs other lectures

Lectures 1-3 introduced the foundations of computational mathematics.

- Lecture 1: Computers as function evaluators.
- Lecture 2 & 3: Notation and terminology for discussing some important properties of functions and algorithms.

All future lectures will be on applications and follow the same pattern.

1. Define a function F : input $\mapsto$ output.
2. Discuss algorithms for evaluating this function.
3. Discuss convergence.

Polynomial approximation is halfway between these two categories.

- On the one hand, there is a (more or less concrete) function $P : ([a, b], f, n) \mapsto p$ which we would like to evaluate.
- On the other hand, we are only interested in this function because it provides a tool for evaluating other functions $F(x)$.

Consequently, we will not pay much attention to the runtime of the polynomial approximation function P. Instead, we will focus on how fast $P([a, b], f, n)$ converges to f for $n \to \infty$, because this will determine the performance of the derived algorithms $F(x)$.

Polynomial Approximation

Outlook

The next three slides introduce some notation and some fundamental results to prepare for the rest of this lecture.

Def: Set of polynomials $\mathcal{P}_n$

$$\mathcal{P}_n = \{\text{polynomials } p(x) \text{ with degree}(p) \leq n\}$$
$$= \left\{ p(x) = \sum_{k=0}^{n} c_k \, x^k \mid c \in \mathbb{R}^{n+1} \right\}$$

Def: Norms of functions

The problem statement on slide 2 contained the somewhat fuzzy condition "$p(x) \approx f(x)$ for all $x \in [a, b]$".

We can use the following norms to make this condition precise.

- Supremum norm: $\|f\|_{L^\infty([a,b])} = \sup_{x \in [a,b]} |f(x)|$.
- L^2 norm: $\|f\|_{L^2([a,b])} = \sqrt{\int_a^b f(x)^2 \, dx}$.
- L^1 norm: $\|f\|_{L^1([a,b])} = \int_a^b |f(x)| \, dx$.

Polynomial Approximation

Discussion

I will only discuss approximation in the supremum norm in this lecture, and I will do so for two reasons.

- The supremum norm provides the pointwise error bound

$$|p(x) - f(x)| \leq \|p - f\|_{L^\infty([a,b])} \qquad \text{for any } x \in [a,b]$$

  which is useful in many applications. In particular, we will see on the next slide that this property allows us to bound the L^1 and L^2 norms in terms of the supremum norm.

- There is a mathematical theory which allows us to determine the asymptotic behaviour of $\inf_{p \in \mathcal{P}_n} \|f - p\|_{L^\infty([a,b])}$ for $n \to \infty$.
  In fact, this theory is one of the main topics of this lecture.

The L^1 and L^2 norms will be used in the lectures on quadrature and Krylov methods, but overall we will be using them far less often than the supremum norm. Correspondingly, I will frequently write $\|f\|_{[a,b]}$ instead of $\|f\|_{L^\infty([a,b])}$ because doing so simplifies the notation and introduces little scope for confusion.

Polynomial Approximation

Thm: On the equivalence of L^p norms

1. $\|f\|_{L^1([a,b])} \leq (b-a)\,\|f\|_{L^\infty([a,b])}$.
2. $\|f\|_{L^2([a,b])} \leq \sqrt{b-a}\,\|f\|_{L^\infty([a,b])}$.
3. There is no constant $C > 0$ such that $\|f\|_{L^\infty([a,b])} \leq C\,\|f\|_{L^1([a,b])}$.
4. There is no constant $C > 0$ such that $\|f\|_{L^\infty([a,b])} \leq C\,\|f\|_{L^2([a,b]}$.

Proof.

1. $$\begin{aligned}\|f\|_{1,[a,b]} &= \int_a^b |f(x)|\,dx \leq \int_a^b \|f\|_{L^\infty([a,b])}\,dx \\ &= \int_a^b 1\,dx\,\|f\|_{L^\infty([a,b])} = (b-a)\,\|f\|_{L^\infty([a,b])}.\end{aligned}$$
2. Analogous.
3. Counterexample: $\|x^{-1/2}\|_{L^\infty([0,1])} = \infty$ but

$$\|x^{-1/2}\|_{L^1([0,1])} = \int_0^1 x^{-1/2}\,dx = \left[2x^{1/2}\right]_0^1 = 2.$$

4. Analogous.

Polynomial Approximation

Discussion

Now that we know about the supremum norm for functions, we can sharpen the problem statement as follows.

$$\text{Given } f : [a,b] \to \mathbb{R} \text{ with } [a,b] \text{ finite,}$$
$$\text{determine } p \in \mathcal{P}_n \text{ such that } \|f-p\|_{[a,b]} \text{ is small.}$$

This statement is more precise than what we had earlier, but it still contains the somewhat vague condition "$\|f-p\|_{[a,b]}$ is small".

One way to make this statement precise would be to demand that

$$p = \operatorname*{arg\,min}_{p \in \mathcal{P}_n} \|f-p\|_{[a,b]}.$$

We will see at the end of this lecture that doing so indeed leads to well-defined function $P : (f,n) \mapsto p$, but also that this function is difficult to evaluate and that other techniques are required to estimate the rate of convergence $P(f,n) \to f$ for $n \to \infty$.

The rationale for requiring only that $\|f-p\|_{[a,b]}$ "small" but not "as small as possible" is to make up for these shortcomings of best approximation.

Polynomial Approximation

Discussion (continued)

Instead of best approximation, we will be focusing on polynomial approximation via interpolation for most of this lecture, i.e. we will determine approximations $p(x) \approx f(x)$ starting from the condition that

$$p(x_k) = f(x_k) \text{ for a given set of interpolation points } x_k.$$

We will do so following the usual outline.

1. Establish that the map $P : \left(f, (x_k)_{k=1}^{n+1}\right) \mapsto p$ is well-defined.
2. Present an algorithm for evaluating this map.
3. Study the convergence $P\left(f, (x_k)_{k=1}^{n+1}\right) \to f$.

The fundamental theorem of algebra recapitulated on the next slide provides the foundation for Step 1 in this outline.

Polynomial Approximation

Fundamental theorem of algebra

The following statements are equivalent for all $p \in \mathcal{P}_n$ and all pairwise distinct $(x_k \in \mathbb{C})_{k=1}^m$. (Pairwise distinct: $\ell \neq k \implies x_\ell \neq x_k$.)

1a. $p(x_k) = 0$ for all $k \in \{1, \dots, m\}$.

2. There is a $q \in \mathcal{P}_{n-m}$ such that $p(x) = q(x) \prod_{k=1}^m (x - x_k)$.

In particular, $p(x) = 0$ if $m > n$.

Fundamental theorem of algebra with repeated points

The condition that the $(x_k)_{k=1}^m$ must be distinct can be dropped if Statement 1a is replaced with the following condition.

1b. $p^{(\ell)}(x_k) = 0$ for all $k \in \{1, \dots, m\}$ and $\ell \in \{0, \dots, n_k - 1\}$, where $n_k = \#\{\ell \mid x_\ell = x_k\}$ denotes the multiplicity of x_k.

$f^{(\ell)}(x)$ denotes the ℓth derivative of a function $f : \mathbb{R} \to \mathbb{R}$, and $f^{(0)}(x) = f(x)$.

Proofs. Undergraduate material.

Discussion: Existence and uniqueness follows easily from the fundamental theorem of algebra, see next slide.

Polynomial Approximation

Thm: Existence and uniqueness of polynomial interpolants

Let $f : \mathbb{R} \to \mathbb{R}$, and let $(x_k \in \mathbb{R})_{k=1}^{n+1}$ be $n+1$ pairwise distinct points. Then there exists a unique $p \in \mathcal{P}_n$ such that

$$p(x_k) = f(x_k) \qquad \text{for all } k \in \{1, \ldots, n+1\}.$$

Proof.

- *Uniqueness.* Assume $p, q \in \mathcal{P}_n$ are two interpolants. Then

  $$p(x) - q(x) \in \mathcal{P}_n \quad \text{and} \quad p(x_k) - q(x_k) = 0 \quad \text{for } k \in \{1, \ldots, n+1\},$$

  which by the fundamental theorem of algebra implies that $p(x) - q(x) = 0$ for all $x \in \mathbb{R}$.

- *Existence.* The interpolant is given by

  $$p(x) = \sum_{k=1}^{n+1} f(x_k)\, \ell_k(x)$$

  where $\ell_k(x)$ denote the Lagrange polynomials introduced next.

Polynomial Approximation

Def: Lagrange polynomials

Let $(x_k \in \mathbb{R})_{k=1}^{n+1}$ denote $n+1$ pairwise distinct points. The Lagrange polynomials $\big(\ell_j(x)\big)_{j=1}^{n+1}$ associated with these points are given by

$$\ell_j(x) = \prod_{i \neq j} \frac{x - x_i}{x_j - x_i}.$$

Theorem

The Lagrange polynomials are the unique polynomials satisfying

$$\ell_j(x_k) = \begin{cases} 1 & \text{if } k = j, \\ 0 & \text{otherwise.} \end{cases}$$

Proof. If $k = j$, then $\quad \ell_k(x_k) = \prod_{i \neq k} \frac{x_k - x_i}{x_k - x_i} = 1.$

If $k \neq j$, then $\quad \ell_j(x_k) = \frac{x_k - x_k}{x_j - x_k} \prod_{i \notin \{j,k\}} \frac{x_k - x_i}{x_j - x_i} = 0.$

Uniqueness follows from the uniqueness part of the theorem on the previous slide. Conversely, this result completes the proof of the existence part of that theorem.

Polynomial Approximation

Warning: Off-by-one errors

There are two natural ways to measure the "length" of a polynomial $p \in \mathcal{P}_n$, namely the degree n and the number of coefficients $n+1$. Unfortunately, it is very easy to confuse these two measures, in particular in results which involve both n and $n+1$.

Example (interpolation theorem):

$p \in \mathcal{P}_n$ is uniquely specified by its values in $n+1$ points.

My recommendation: think twice whenever you write n or $n+1$!

If you write n when you should have written $n+1$, then it is likely that your code will quietly return a wrong result, which is the worst kind of bug you can have.

An easy way to check whether a number should be n or $n+1$ is to think about the case $n = 0$. For example, it is obvious that interpolation with a constant polynomial $p \in \mathcal{P}_0$ requires 1 interpolation point; hence the numbers in the interpolation theorem must be n and $n+1$.

Polynomial Approximation

Remark: 0- vs 1-based indexing

Unlike many other authors, I index the interpolation points x_k and Lagrange polynomials $\ell_k(x)$ with

$$k \in \{1, \ldots, n+1\} \qquad \text{(1-based indexing)}$$

rather than

$$k \in \{0, \ldots, n\} \qquad \text{(0-based indexing)}.$$

Having to choose either of these conventions is a ubiquitous problem in computing and sometimes also other areas. For example, most European countries enumerate floors starting from 0 / ground floor, while the US, Canada and Singapore start with 1.

I follow 1-based indexing in this module to be consistent with Julia, even though 0-based indexing is arguably more natural for polynomials.

Implementation of interpolation via Lagrange polynomials

See `lagrange_example()`.

Polynomial Approximation

Remark

Just like the fundamental theorem of algebra can be modified to allow for repeated points, the existence and uniqueness theorem for polynomial interpolation can be extended to allow for repeated points.

Thm: Existence and uniqueness of polynomial interpolants with repeated points

Let $f : \mathbb{R} \to \mathbb{R}$ and let $(x_k \in \mathbb{R})_{k=1}^{n+1}$.
Then there exists a unique $p \in \mathcal{P}_n$ such that

$$p^{(\ell)}(x_k) = f^{(\ell)}(x_k) \qquad \text{for all } k \in \{0, \ldots, n\} \text{ and } \ell \in \{0, \ldots, n_k\}$$

where $n_k = \#\{\ell \mid x_\ell = x_k\}$ denotes the multiplicity of x_k.

Proof. Analogous to that of the theorem on slide 12.

Terminology

Interpolation in repeated points is called *Hermite interpolation*.
Interpolation in distinct points is sometimes called *Lagrange interpolation*.

Polynomial Approximation

Discussion

Recall our three-point plan.

1. Establish that the map $P : \big(f, (x_k)_{k=1}^{n+1}\big) \mapsto p$ is well-defined.
2. Present an algorithm for evaluating this map.
3. Study the convergence $P\big(f, (x_k)_{k=1}^{n+1}\big) \to f$.

We have now completed Steps 1 and 2 on this list, and it remains to tackle Step 3. We will do so using the following result.

Generalised Taylor's theorem

Assume $f : [a, b] \to \mathbb{R}$ is $n+1$ times differentiable, and denote by $p \in \mathcal{P}_n$ the interpolant to $f(x)$ in the $n+1$ points $\big(x_k \in [a, b]\big)_{k=1}^{n+1}$.
Then there exists $\xi = \xi(x) \in [a, b]$ such that

$$f(x) - p(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{k=1}^{n+1} (x - x_k).$$

Proof. Omitted since not instructive. See e.g. Burden, Faires (2011), *Numerical Analysis*, Theorem 3.3.

Polynomial Approximation

Remark

The name "generalised Taylor's theorem" is not standard, but I believe it is justified because the above result includes the more well-known Taylor's theorem

$$f(x) = \underbrace{\sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k}_{p(x)} + \frac{f^{(n+1)}(\xi)}{(n+1)!} (x - x_0)^{n+1}$$

for some $\xi \in (x_0, x)$ as the special case $(x_k = x_0)_{k=1}^{n+1}$.

Polynomial Approximation

Discussion

Note that while interpolation is well-defined for any function $f : \mathbb{R} \to \mathbb{R}$, the generalised Taylor's theorem applies only if $f(x)$ is sufficiently differentiable.

It is obvious that interpolation can only converge if $f(x)$ is "nice enough"; if $f(x)$ was allowed to be completely arbitrary, then the data $f(x_k)$ would tell us nothing about the values of $f(x)$ for $x \notin \{x_k\}$.

Differentiability is one way to make this "nice enough" requirement rigorous. Other such notions exist (e.g. Sobolev spaces), but differentiability is the easiest and most common.

I will use the notion of "smoothness" introduced on the next slide to make it easier to talk about "nice enough" functions.

Polynomial Approximation

Terminology: Smoothness

Functions with sufficiently many derivatives are called smooth. More precisely, there are two slightly different "definitions" of smoothness.

- Some authors call a function smooth if it has infinitely many derivatives.
- I (and many other authors) will use the term smooth in a more qualitative and relative sense similar to how we use hot and cold, see the example below.

Example

10°C is cold when we talk about an A/C setting, but it is hot when we talk about a fridge temperature.

Similarly, a function with 10 derivatives is smooth when we talk about approximation with quadratic polynomials, but it is not smooth when we talk about approximation with a degree 100 polynomial.

Polynomial Approximation

Discussion

The generalised Taylor's theorem is on the one hand very powerful because it provides an exact expression for the interpolation error $f(x) - p(x)$. On the other hand, it is not very useful because it does so by introducing two further quantities which are hard to estimate, namely

$$\frac{f^{(n+1)}(\xi)}{(n+1)!} \qquad \text{and} \qquad \prod_{k=1}^{n+1}(x - x_k).$$

I will discuss how we can make sense of these two quantities in due time, but first I will present a simple trick which eliminates the need to estimate the derivative factor and makes it very easy to estimate the product factor.

Polynomial Approximation

Discussion (continued)

The idea behind this trick is to note that

$$x, (x_k) \in [a, b] \qquad \Longrightarrow \qquad |x - x_k| \leq |b - a|$$

and hence

$$\begin{aligned} |f(x) - p(x)| &= \frac{|f^{(n+1)}(\xi)|}{(n+1)!} \prod_{k=1}^{n+1} |x - x_k| \\ &= O\big(|b - a|^{n+1}\big) \qquad \text{for } |b - a| \to 0. \end{aligned}$$

Of course, demanding $|b - a| \to 0$ does not make sense because the approximation interval $[a, b]$ is generally beyond our control.

However, we can always take a finite-sized interval $[a, b]$ and split it into many small intervals $[y_k, y_{k+1}] \subset [a, b]$ and achieve the analogous condition $|y_{k+1} - y_k| \to 0$ in this way.

The following slides will pursue this idea further.

Terminology

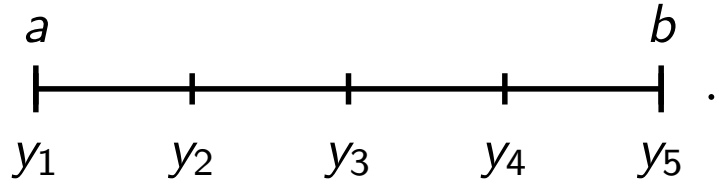
Methods which split a problem over a "large" interval $[a, b]$ into many problems over "small" intervals $[y_k, y_{k+1}]$ are called composite methods.

Polynomial Approximation

Def: Piecewise polynomials on equispaced grid

Let $(y_k)_{k=1}^{m+1}$ denote $m+1$ equispaced points in the interval $[a,b]$, i.e.

$$y_k = a + (b-a)\,\tfrac{k-1}{m} \qquad \Longleftrightarrow$$



Let us denote by $P_m\,\mathcal{P}_n([a,b])$ the space of all functions $p : [a,b] \to \mathbb{R}$ such that $p(x)$ restricted to each interval $[y_k, y_{k+1}]$ is a polynomial with degree$(p) \leq n$, i.e.

$$P_m\,\mathcal{P}_n([a,b]) = \left\{ p : [a,b] \to \mathbb{R} \mid p|_{[y_{k-1},y_k]} \in \mathcal{P}_n \right\}.$$

$P_m\,\mathcal{P}_n([a,b])$ is not standard notation.

Piecewise polynomial interpolation

We can construct an approximation $p(x) \approx f(x)$ by defining $p \in P_m\,\mathcal{P}_n([a,b])$ as the polynomial interpolant to $f(x)$ on each interval.

Code explains this better than mathematical formulae, so have a look at `piecewise_interpolation_example()`.

Polynomial Approximation

Thm: Convergence of piecewise polynomial interpolation

Assume $f : [a, b] \to \mathbb{R}$ has $n + 1$ derivatives and denote by $p \in P_m \mathcal{P}_n([a, b])$ the piecewise polynomial interpolant to $f(x)$. Then,

$$\|f(x) - p(x)\|_{[a,b]} = O\left(\left(\tfrac{b-a}{m}\right)^{n+1}\right) \qquad \text{for } m \to \infty.$$

The big O on the right is not uniform in n.

Proof. Follows immediately from the above.

Numerical demonstration

See `piecewise_interpolation_convergence()`.

Observations:

- $f(x) = \sin(x)$: The convergence is indeed $O(m^{-n-1})$.
- $f(x) = \operatorname{sign}(x)\, x^3$: The convergence is only $O(m^{-\min\{n+1,3\}})$.

  Explanation: $f(x) = \operatorname{sign}(x)\, x^3$ has only three derivatives.

  The derivatives of $f(x) = \operatorname{sign}(x)\, x^3$ are given by $f^{(k)}(x) = \operatorname{sign}(x)\, x^{3-k}$: this is obvious for $x \neq 0$, and follows for $x = 0$ by continuity.

Polynomial Approximation

Advantages of composite methods

Composite methods like piecewise interpolation are useful for the following reasons.

- They are very easy to implement for small n.

  For example, piecewise interpolation with $n = 1$ amounts to connecting the data points by straight lines.

- They perform reasonably well when applied to non-smooth functions.

  In such situations, reducing the mesh size (i.e. choosing larger m) is usually the best we can do. See `piecewise_interpolation_example()` with $f(x) = |x - \sqrt{2}|$ for illustration.

- They support adaptive approximation if $f(x)$ lacks smoothness only in a small region.

  See `adaptive_piecewise_interpolation()`.

Polynomial Approximation

Disadvantages of composite methods

However, composite methods also have some important disadvantages

- For some applications (e.g. matrix functions $p(A)$), it is important that we approximate using a proper polynomial $p \in \mathcal{P}_n$ rather than a piecewise polynomial $p \in P_m \mathcal{P}_n([a, b])$.
- Composite methods do not perform well if $f(x)$ is smooth.

  Have another look at `piecewise_interpolation_convergence()` with $f(x) = \sin(x)$ and note how the error is orders of magnitudes smaller for larger n and similar m.

These disadvantages suggest that we return to the generalised Taylor's theorem

$$f(x) - p(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{k=1}^{n+1} (x - x_k)$$

and study how this error behaves in the limit $n \to \infty$.

Polynomial Approximation

Discussion

Observations:

- The generalised Taylor's theorem only tells us that $\xi \in [a, b]$; hence we have to work with the worst-case bound

$$|f(x) - p(x)| \leq \frac{\|f^{(n+1)}\|_{[a,b]}}{(n+1)!} \prod_{k=1}^{n+1} |x - x_k|.$$

- This bound separates into a derivative factor which depends only on $f(x)$ but not on x and (x_k), and a product factor which depends only on x and (x_k) but not on $f(x)$.

Conclusions:

- We can choose "good" interpolation points (x_k) without knowing anything about $f(x)$.

  Choosing (x_k) independently of $f(x)$ is the exact opposite of what we did for adaptive approximation.

- (x_k) are "good" if they minimise $\max_{x \in [a,b]} \prod_{k=1}^{n+1} |x - x_k|$.

We will next look into how to obtain such (x_k), but before we can do so we first need to introduce some notation.

Polynomial Approximation

Terminology: Node polynomial

$$\ell(x) = \prod_{k=1}^{n+1}(x - x_k) \in \mathcal{P}_{n+1}$$

is called the *node polynomial* associated with the points $(x_k)_{k=1}^{n+1}$.

Discussion: Approximation interval

Many quantities introduced in the following will depend on the approximation interval $[a, b]$, and the formulae for these quantities can be quite complicated if we allow arbitrary $[a, b]$.

To avoid these complications, I will exclusively consider $[a, b] = [-1, 1]$ in the following. The results on the next two slides show that doing so does not reduce generality.

Polynomial Approximation

Lemma

Let $p \in \mathcal{P}_n$ be the unique interpolant to $f(x)$ with interpolation points $(x_k \in \mathbb{R})_{k=1}^{n+1}$, and let $\phi \in \mathcal{P}_1$ be bijective (i.e. $\phi(x) \neq \text{const}$).
Then, $p \circ \phi \in \mathcal{P}_n$ is the unique polynomial interpolant to $f\big(\phi(x)\big)$ with interpolation points $\big(\phi^{-1}(x_k)\big)_{k=1}^{n+1}$.

Proof. We have for all $k \in \{1, \dots, n+1\}$ that

$$\big(p \circ \phi\big)\big(\phi^{-1}(x_k)\big) = p(x_k) = f(x_k) = \big(f \circ \phi\big)\big(\phi^{-1}(x_k)\big);$$

thus $p \circ \phi$ indeed interpolates $f\big(\phi(x)\big)$ in the specified points, and we have $p \circ \phi \in \mathcal{P}_n$ since $p_1 \in \mathcal{P}_{n_1}, p_2 \in \mathcal{P}_{n_2} \implies p_1 \circ p_2 \in \mathcal{P}_{n_1\, n_2}$.

Lemma

Let $[a_1, b_1]$ and $[a_2, b_2]$ be two finite intervals. Then there exists a unique linear and bijective function $\phi : [a_1, b_1] \to [a_2, b_2]$.

Proof. It follows from the monotonicity of linear functions that $\phi \in \mathcal{P}_1$ is a bijective map $[a_1, b_1] \to [a_2, b_2]$ iff $\phi(a_1) = a_2$ and $\phi(b_1) = b_2$.
Existence and uniqueness then follow from the existence and uniqueness of polynomial interpolants.

Polynomial Approximation

Lemma

For any $f : [a_2, b_2] \to \mathbb{R}$ and any surjective $\phi : [a_1, b_1] \to [a_2, b_2]$, we have

$$\|f\|_{[a_2,b_2]} = \|f \circ \phi\|_{[a_1,b_1]}.$$

Proof. Since $\phi(x_1)$ is surjective, we have

$$\begin{aligned}\|f\|_{[a_2,b_2]} &= \sup_{x_2 \in [a_2,b_2]} |f(x_2)| = \sup_{x_2 \in \phi([a_1,b_2])} |f(x_2)| \\ &= \sup_{x_1 \in [a_1,b_1]} |f(\phi(x_1))| = \|f \circ \phi\|_{[a_1,b_1]}\end{aligned}$$

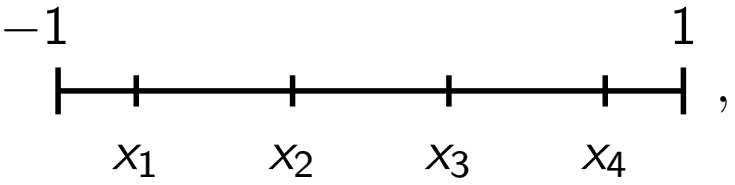
Polynomial Approximation

Discussion

We now return to the problem of choosing interpolation points (x_k) such that $\ell(x)$ is as small as possible on $[-1, 1]$.

You might expect that this goal is achieved if we distribute the interpolation points (x_k) uniformly over $[-1, 1]$, i.e. if we set

$$x_k = -1 + \tfrac{2k-1}{m} \qquad \Longleftrightarrow$$



because clearly $\ell(x)$ is small if x is close to an interpolation point x_k where $\ell(x_k) = 0$, and distributing the point uniformly minimises the maximal distance to the nearest interpolation point, i.e. the above choice of x_k minimises

$$\max_{x \in [-1,1]} \quad \min_{k \in \{1,\ldots,n+1\}} \quad |x - x_k|.$$

However, it turns out this choice of interpolation points leads to a node polynomial $\ell(x)$ which is much larger for $x \approx \pm 1$ than for $x \approx 0$, see `node_polynomial()`.

Polynomial Approximation

Discussion (continued)

By increasing n, we further observe that uniform distribution of the interpolation points leads to an $\ell(x)$ which tends to 0 for all x, but it does so much more slowly at the ends of $[-1, 1]$ than it does in the middle.

According to the generalised Taylor's theorem, this means that the interpolant $p(x)$ converges more slowly for $x \approx \pm 1$, and we observe in `interpolant()` that $p(x)$ may even fail to converge for $x \approx \pm 1$.

On second thought, it becomes clear that choosing uniformly distributed interpolation points corresponds to minimising the smallest factor $(x - x_k)$ in $\ell(x)$, but this may not be enough if many other factors are large. Instead of minimising the smallest factor, we should hence aim to make as many factors $(x - x_k) < 1$ as possible.

Heuristically speaking, this means that we should place more (x_k) near the ends of $[-1, 1]$, because near the ends we can be "close" to only half the (x_k) while near the middle we can be "close" to all (x_k).

This heuristic is further supported by the shape of $\ell(x)$ observed in `node_polynomial()`. Unfortunately, it is only a heuristic, and making this idea precise requires some more elaborate mathematical analysis which will be the topic of the following slides.

Polynomial Approximation

Generalised Taylor's theorem, supremum norm version

Assume $f : [-1,1] \to \mathbb{R}$ is $n+1$ times differentiable, and denote by $p \in \mathcal{P}_n$ the interpolant to $f(x)$ in the $n+1$ points $\left(x_k \in [-1,1]\right)_{k=1}^{n+1}$. Then,

$$\|f - p\|_{[-1,1]} \leq \frac{1}{(n+1)!} \|f^{(n+1)}\|_{[-1,1]} \|\ell\|_{[-1,1]}.$$

Proof. Immediate consequence of the generalised Taylor's theorem. The main reason for stating this result it to introduce the supremum norm notation for the product factor.

Lemma

The node polynomial $\ell(x) = \prod_{k=1}^{n+1}(x - x_k)$ is monic, i.e. $\ell(x) \in \mathcal{P}_n^{\text{monic}}$.

Proof. Obvious.

Def: Monic polynomials

- $p \in \mathcal{P}_n$ is called *monic* if $p(x) = x^n + q(x)$ for some $q \in \mathcal{P}_{n-1}$.
- I denote the set of monic polynomial by $\mathcal{P}_n^{\text{monic}} = x^n + \mathcal{P}_{n-1}$.

Polynomial Approximation

Thm: Equioscillating monic polynomials

We have $p = \arg\min_{p_{\text{trial}} \in \mathcal{P}_n^{\text{monic}}} \|p_{\text{trial}}\|_{[-1,1]}$ if there are $n+1$ points

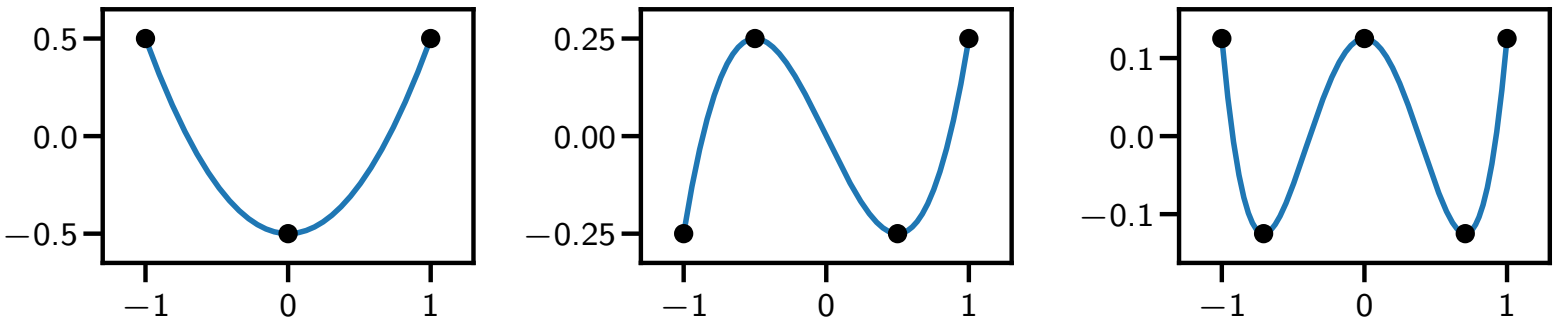
$$-1 \leq x_1 < x_2 < \ldots < x_n < x_{n+1} \leq 1$$

such that

$$p(x_k) = (-1)^{n+1-k} \|p\|_{[-1,1]} \qquad \text{for all } k \in \{1, \ldots, n+1\}.$$

A function $p(x)$ satisfying this property is said to *equioscillate*.

Examples



$n = 2$ $n = 3$ $n = 4$

Polynomial Approximation

Proof. Assume for the sake of contradiction that $p \in \mathcal{P}_n^{\text{monic}}$ equioscillates and there exists $q \in \mathcal{P}_n^{\text{monic}}$ with $\|q\|_{[-1,1]} < \|p\|_{[-1,1]}$.

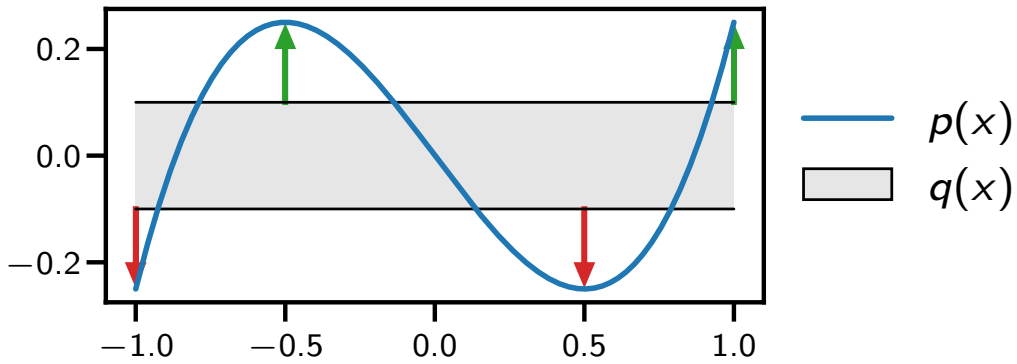
We observe:

- $p - q \in \mathcal{P}_{n-1}$ since p and q are monic.
- $\|q\|_{[-1,1]} < \|p\|_{[-1,1]}$ implies that

$$\begin{cases} p(x_k) - q(x_k) > 0 & \text{if } n+1-k \text{ is even}, \\ p(x_k) - q(x_k) < 0 & \text{if } n+1-k \text{ is odd}, \end{cases}$$

see the illustration below. Thus $p(x) - q(x)$ has at least n zeroes according to the intermediate value theorem.

It follows from these observations and the fundamental theorem of algebra that $p(x) - q(x) = 0$, which contradicts our initial assumption.



Polynomial Approximation

Discussion

Recall that our aim is to find interpolation points (x_k) such that $\|\ell\|_{[-1,1]}$ is as small as possible.

The above theorem then says that $\ell(x)$ must be of the form $\ell(x) = p(x)/c_{n+1}$ where $p \in \mathcal{P}_{n+1}$ is any polynomial which equioscillates on $[-1, 1]$, and $c_{n+1} \in \mathbb{R}$ is the coefficient of the highest-degree monomial in $p(x)$.

Determining polynomials $p(x)$ which have roughly equal magnitude throughout some $D \subset \mathbb{C}$ is trivial if $D = \{|z| = 1\}$ is the unit circle. In this case, all monomials $p(x) = x^k$ satisfy $|p(x)| = 1$ for all $x \in D$.

The following slides will show that polynomials $p(x)$ equioscillating on $[-1, 1]$ can be obtained by "projecting" these monomials from the unit circle $\{|z| = 1\}$ to $[-1, 1]$.

Polynomial Approximation

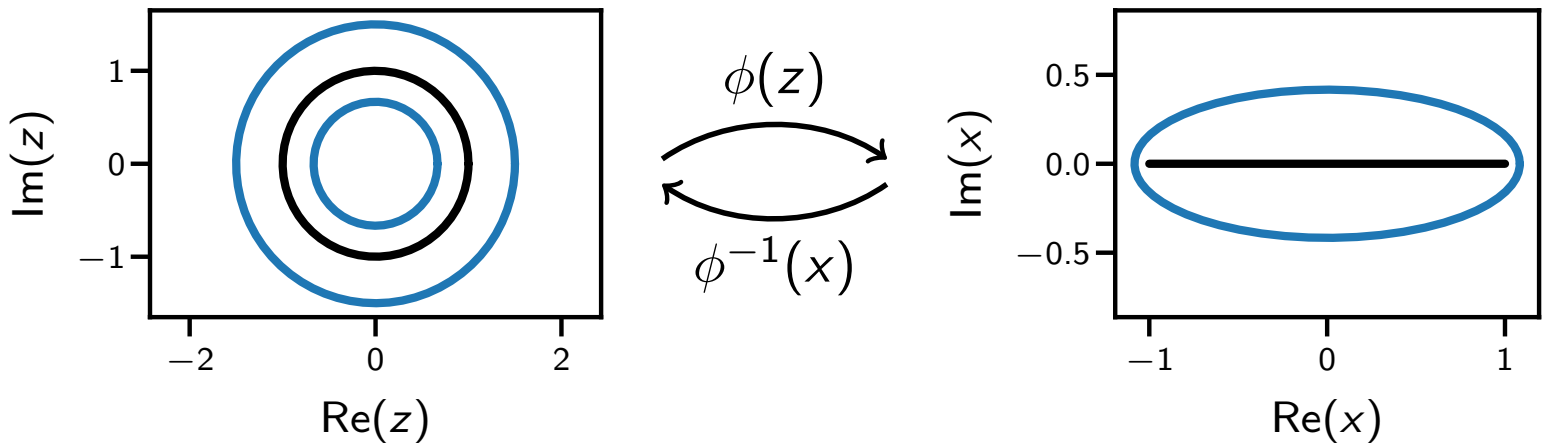
Thm: Properties of Joukowsky map

Consider the function $\phi : \mathbb{C} \setminus \{0\} \to \mathbb{C}$ given by

$$\phi(z) = \frac{z + z^{-1}}{2}, \qquad \phi_{\pm}^{-1}(x) = x \pm \sqrt{x^2 - 1}.$$

This function is called the *Joukowsky map* and has the following properties.

1. $\phi(z) = \text{real}(z)$ if $|z| = 1$ and hence $\phi\big(\{|z| = 1\}\big) = [-1, 1]$.
2. The two branches of $\phi^{-1}(x)$ satisfy $\phi_{+}^{-1}(x) = \big(\phi_{-}^{-1}(x)\big)^{-1}$



Polynomial Approximation

Proof of inverse.

$$\frac{z+z^{-1}}{2} = x \iff z^2 - 2zx + 1 = 0 \iff z = x \pm \sqrt{x^2-1}.$$

Proof of Property 1. We have $z^{-1} = \frac{\bar{z}}{|z|^2}$ where $\bar{z} = \mathrm{real}(z) - i\ \mathrm{imag}(z)$ denotes the complex conjugate; hence for $|z| = 1$ we obtain

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

Proof of Property 2. Immediate consequence of

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

Polynomial Approximation

Thm: Properties of Chebyshev polynomials

Consider the functions $T_n : \mathbb{C} \to \mathbb{C}$ given by $T_n(x) = \phi\big(\phi^{-1}(x)^n\big)$.

Note that $\phi(\phi_+^{-1}(x)^n)) = \phi\big(\phi_-^{-1}(x)^{-n}\big) = \phi\big(\phi_-^{-1}(x)^n\big)$, so it does not matter which branch of $\phi^{-1}(x)$ we use in the definition of $T_n(x)$.

These functions are called *Chebyshev polynomials* and have the following properties.

1. $T_n \in \mathcal{P}_n$ and T_n satisfies the recurrence relations

$$T_0(x) = 1, \qquad T_1(x) = x, \qquad T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x).$$

2. Equioscillation: We have $\|T_n\|_{[-1,1]} = 1$ and

$$T_n(x_k) = (-1)^{n+1-k} \qquad \text{for all} \qquad \Big(x_k = \cos\big(\pi \tfrac{k-1}{n}\big)\Big)_{k=1}^{n+1}.$$

   These points are called the *Chebyshev extreme points*.

3. $T_n(x_k) = 0$ for $\big(x_k = \cos\big(\pi \tfrac{2k-1}{2n}\big)\big)_{k=1}^{n}$.
   These points are called the *Chebyshev zeroes*.

Polynomial Approximation

Proof of Property 1.

The formulae for $n = 0$ and $n = 1$ can be verified by straightforward calculations:

$$T_0(x) = \phi\left(\phi^{-1}(x)^0\right) = \phi(1) = \tfrac{1+1^{-1}}{2} = 1,$$
$$T_1(x) = \phi\left(\phi^{-1}(x)^1\right) = x.$$

To verify the recurrence relation, I introduce

$$z = \phi^{-1}(x) \qquad \Longleftrightarrow \qquad x = \phi(z) = \tfrac{z+z^{-1}}{2}$$

and compute

$$\begin{aligned} 2x\,T_n(x) &= \frac{1}{2}\left(z + z^{-1}\right)\left(z^n + z^{-n}\right) \\ &= \frac{z^{n+1} + z^{-n-1}}{2} + \frac{z^{n-1} + z^{-n+1}}{2} \\ &= T_{n+1}(x) + T_{n-1}(x). \end{aligned}$$

Finally, $T_n \in \mathcal{P}_n$ follows immediately from the above.

Polynomial Approximation

Proof of Property 2.

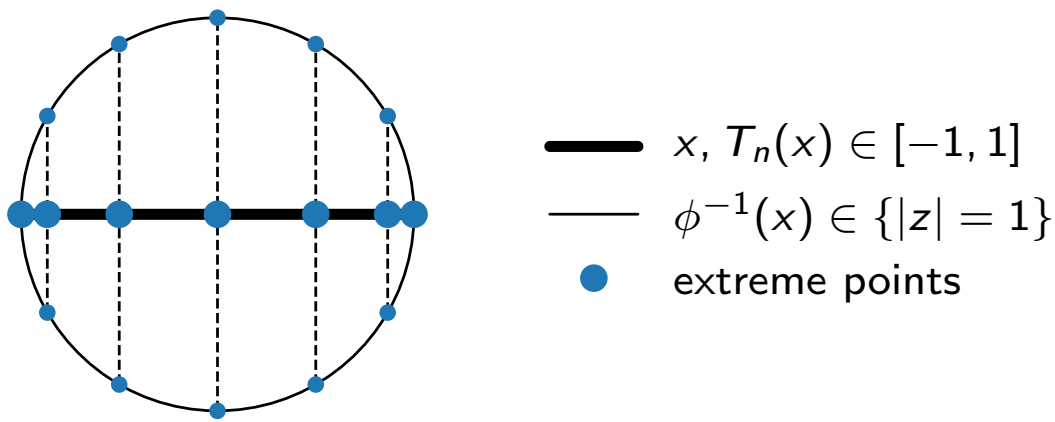
We know from Property 1 on slide 37 that

$$|\phi^{-1}(x)| = 1 \qquad \text{and} \qquad T_n(x) = \text{real}\big(\phi^{-1}(x)^n\big) \qquad \text{for all } x \in [-1,1].$$

These properties yield (see picture below) that $\|T_n\|_{[-1,1]} \leq 1$ and that this upper bound is attained if

$$\begin{aligned} \phi^{-1}(x_k)^n = \pm 1 \quad &\Longleftrightarrow \quad \phi^{-1}(x_k) = e^{\pi i \frac{k-1}{n}} \\ &\Longleftrightarrow \quad x_k = \phi\Big(e^{\pi i \frac{k-1}{n}}\Big) = \text{real}\Big(e^{\pi i \frac{k-1}{n}}\Big) = \cos\big(\pi \tfrac{k-1}{n}\big). \end{aligned}$$

Illustration for $n = 6$:



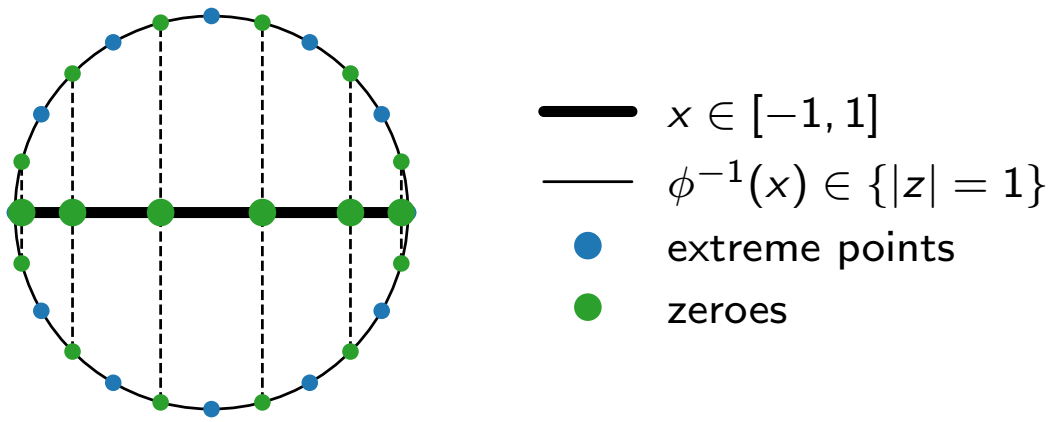
Polynomial Approximation

Proof of Property 3.

Analogous as above, we conclude that $T_n(x_k) = \text{real}\big(\phi^{-1}(x_k)\big) = 0$ if

$$\phi^{-1}(x_k)^n = \pm i \iff \phi^{-1}(x_k) = e^{\pi i \frac{2k-1}{2n}}$$
$$\iff x_k = \phi\Big(e^{\pi i \frac{2k-1}{2n}}\Big) = \text{real}\Big(e^{\pi i \frac{2k-1}{2n}}\Big) = \cos\big(\pi \tfrac{2k-1}{2n}\big).$$

Illustration for $n = 6$:



Polynomial Approximation

Discussion

Considering the generalised Taylor's theorem, the optimality theorem for equioscillating monic polynomials and the theorem on the properties of Chebyshev polynomials, we conclude that Chebyshev zeroes should be good interpolation points for polynomial interpolation.

Indeed, we see by uncommenting the "Chebyshev" line in `node_polynomial()` that Chebyshev zeroes lead to an equioscillating $\ell(x)$, and doing likewise in `interpolant()` shows that interpolation in Chebyshev zeroes converges everywhere.

In fact, one can rigorously prove that interpolation in Chebyshev zeros converges and quantify the rate of convergence, see the next slide.

Terminology: Chebyshev interpolation

Interpolation in Chebyshev zeroes is called Chebyshev interpolation.

Polynomial Approximation

Thm: Chebyshev interpolation error for differentiable functions

Assume $f : [-1, 1] \to \mathbb{R}$ is k times differentiable and denote by $p_n \in \mathcal{P}_n$ the interpolant to $f(x)$ in the $n + 1$ Chebyshev zeroes.

Then there exists a $C > 0$ independent of n and f such that

$$\|f - p_n\|_{[-1,1]} \leq C \, \|f^{(k)}\|_{[-1,1]} \, n^{-k}.$$

Proof. Beyond the scope of this module. See Trefethen (2013), Approximation Theory and Approximation Practice, Theorem 7.2.

Examples

- $f(x) = |x|$ (`convergence_abs()`):
  $f(x)$ is once differentiable, so convergence is $O(n^{-1})$.
- $f(x) = |\sin(4\pi x)|^3$ (`convergence_sinabs3()`):
  $f(x)$ is three times differentiable, so convergence is $O(n^{-3})$.

We further observe that in the second example, the error stagnates for a while before reaching the $O(n^{-3})$ asymptotics. This is because we must first have enough interpolation points to resolve the oscillations of $\sin(4\pi x)$, see `convergence_sinabs3()`.

Polynomial Approximation

Discussion

If $f(x)$ has infinitely many derivatives, then the above theorem says that the Chebyshev interpolation error is $O(n^{-k})$ for any arbitrary $k \geq 0$.

On the one hand, this statement is quite powerful because it says that Chebyshev interpolation converges very quickly. On the other hand, this statement is not very satisfying because it does not give us a precise big O estimate for the speed of convergence.

It turns out that we can obtain precise big O estimates for infinitely differentiable functions, but to do so we must assume that $f(x)$ is analytic on $[-1, 1]$.

You should have heard about analytic function during your undergraduate studies, but I suspect many of you will have forgotten at least parts of the theory because it is not used very often in applied mathematics.

I will therefore briefly summarise the theory relevant for this module on the following slides before stating the convergence result for Chebyshev interpolation applied to analytic functions.

Polynomial Approximation

Thm: Taylor series and complex differentiability

Assume $f : \mathbb{C} \to \mathbb{C}$ and $z_0 \in \mathbb{C}$.

Then the following statements are equivalent.

- The Taylor series of $f(z)$ around z_0 converges to $f(z)$, i.e. there exists a $\delta > 0$ such that for all $z \in \mathbb{C}$ with $|z - z_0| < \delta$ we have

$$f(z) = \lim_{n \to \infty} \sum_{k=0}^{n} \frac{f^{(k)}(z_0)}{k!} (z - z_0)^k.$$

- $f(z)$ is complex differentiable at z_0, i.e.

$$f'(z_0) = \lim_{z \to z_0} \frac{f(z) - f(z_0)}{z - z_0}$$

  exists and assumes the same value for any sequence $z \to z_0$.

Polynomial Approximation

Def: Analytic functions

A function for which the above statements hold is called analytic at z_0.

If $f(z)$ is analytic at every point $z \in \Omega \subseteq \mathbb{C}$, then we say "$f(z)$ is analytic on Ω" or "$f : \Omega \to \mathbb{C}$ is analytic".

Note that some authors write "holomorphic" instead of "analytic".

Examples

- real(z) is not analytic at any point $z_0 = x_0 + iy_0$.

  Proof. We have

  $$\lim_{x \to x_0} \frac{\text{real}(x + iy_0) - \text{real}(x_0 + iy_0)}{(x + iy_0) - (x_0 + iy_0)} = \lim_{x \to x_0} \frac{x - x_0}{x - x_0} = 1$$

  but

  $$\lim_{y \to y_0} \frac{\text{real}(x_0 + iy) - \text{real}(x_0 + iy_0)}{(x_0 + iy) - (x_0 + iy_0)} = \lim_{y \to y_0} \frac{x_0 - x_0}{y - y_0} = 0.$$

- The identity function $z \mapsto z$ is analytic on $\mathbb{C}$.

  Proof. We have $\lim_{z \to z_0} \frac{z - z_0}{z - z_0} = 1$.

Polynomial Approximation

Complex differentiation rules

Assume $f(z), g(z)$ are analytic at z_0, and $\hat{f}(w)$ is analytic at $w_0 = g(z_0)$. Then, all of the function $h(z)$ given below are analytic at z_0 and have the given derivative.

$$
\begin{aligned}
h(z) = f(z) + g(z) \quad &\Longrightarrow \quad h'(z) = f'(z) + g'(z), \\
h(z) = f(z)\, g(z) \quad &\Longrightarrow \quad h'(z) = f'(z)\, g(z) + f(z)\, g'(z), \\
h(z) = f(z)^{-1} \quad &\Longrightarrow \quad h'(z) = -f'(z)/f(z)^2, \\
h(z) = \hat{f}(g(z)) \quad &\Longrightarrow \quad h'(z) = \hat{f}'(g(z))\, g'(z).
\end{aligned}
$$

Important analytic functions

- Polynomials $p \in \mathcal{P}_n$ are analytic on $\mathbb{C}$.
- Rationals $r(z) = p(z)/q(z)$ with $p, q \in \mathcal{P}$, are analytic for all $z \in \mathbb{C}$ such that $q(z) \neq 0$.
- $\exp(z)$, $\sin(z)$, $\cos(z)$ are analytic on $\mathbb{C}$.

Note that the analyticity of polynomials and rationals are immediate consequences of the sum, product and inverse rules listed above.

Polynomial Approximation

Thm: Chebyshev interpolation error for analytic functions

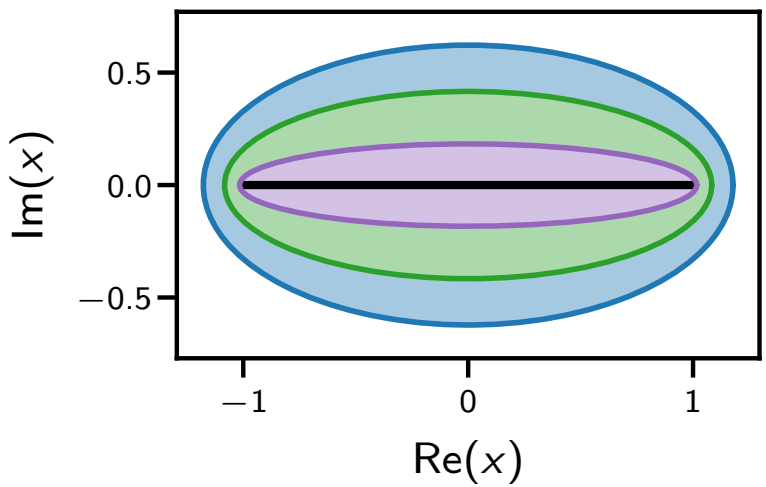
Assume $f(x)$ is analytic on the *Bernstein ellipse*

$$E(r) := \left\{ x \in \mathbb{C} \mid \tfrac{1}{r} < |\phi_{\pm}^{-1}(x)| < r \right\}, \qquad r \geq 1,$$

and denote by $p \in \mathcal{P}_n$ the interpolant to f in $n+1$ Chebyshev zeroes. Then, there exists a $C > 0$ independent of f and n such that

$$\|f - p\|_{[-1,1]} \leq C \, \|f\|_{E(r)} \, r^{-n}.$$

Example: Bernstein ellipses for radii $1 = r_1 < r_1 < r_2 < r_3$.



Polynomial Approximation

Examples

- $f(x) = \exp\left(-\frac{1}{|x|}\right)$ (`convergence_expinvx()`)

  This function is infinitely differentiable but not analytic at $x = 0$ because its Taylor series around $x = 0$ is $0 \neq f(x)$ (you can check this yourself).

  The convergence is therefore super-algebraic but sub-exponential.

- $f(x) = \sin(4\pi x)$ (`convergence_sin()`)

  This function is analytic on $\mathbb{C}$, so it converges super-exponentially.

  Furthermore, we again observe that the error stagnates for a while before we start seeing the asymptotic behaviour. This is exactly the same phenomenon as we observed for `convergence_sinabs3()`.

Polynomial Approximation

Examples (continued)

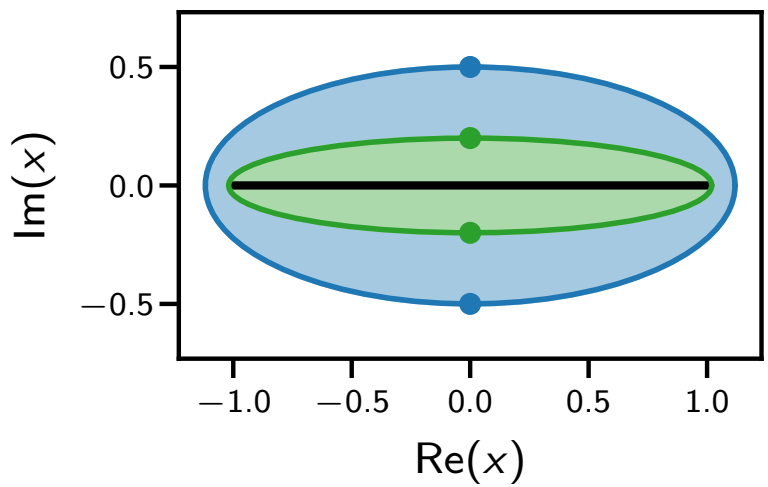
- $f_1(x) = \frac{1}{1+4x^2}$ and $f_2(x) = \frac{1}{1+25x^2}$ (`convergence_runge()`)

  We observe:

  - $f_1(x)$ is analytic on $D_1 = \mathbb{C} \setminus \left\{\pm\frac{i}{2}\right\}$.
  - $f_2(x)$ is analytic on $D_2 = \mathbb{C} \setminus \left\{\pm\frac{i}{5}\right\}$.

  Hence the convergence is exponential in both cases, but with a faster rate for $f_1(x)$ because we can fit a larger Bernstein ellipse into D_1, see the picture below.

  Moreover, we observe that the rate of convergence precisely matches the predictions of the above theorem.



Polynomial Approximation

Remark: Density of Chebyshev points

I mentioned earlier that minimising $\|\ell\|_{[-1,1]}$ requires placing more interpolation points near the ends of $[-1, 1]$. Now that we know about Chebyshev polynomials and points, we can make this statement rigorous.

Warning: The following assumes some advanced knowledge of measure theory. Ignore anything that confuse you.

The theorem regarding the properties of Chebyshev polynomials shows that the Chebyshev zeroes are given by $x = \cos(t)$ where t is uniformly distributed over $[0, \pi]$.

The volume element dx of Chebyshev zeroes thus satisfies

$$dx = -\sin(t)\,dt = -\sin\left(\cos^{-1}(x)\right)\,dt \quad \Longleftrightarrow \quad -\frac{1}{\sqrt{1-x^2}}\,dx = dt,$$

where for the expression on the right-hand side I used the identity

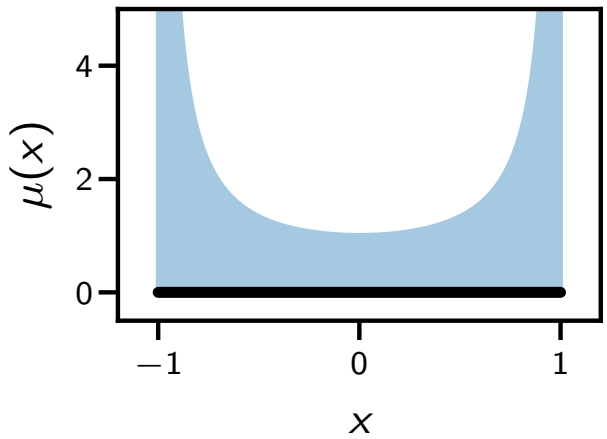
$$\sin(x)^2 + \cos(x)^2 = 1 \quad \Longleftrightarrow \quad \sin(x) = \sqrt{1-\cos(x)^2}.$$

Polynomial Approximation

Remark: Density of Chebyshev points (continued)

It follows from the above that the density of Chebyshev points is given by

$$\mu_{\text{Cheb}}(x) = \frac{1}{\sqrt{1-x^2}} \qquad \Longleftrightarrow$$



As expected, Chebyshev zeroes are much denser near the ends of $[-1, 1]$ than in the centre. In fact, one can show that it is the square-root-type singularity of $\mu(x)$ for $x \approx \pm 1$ which makes polynomial interpolation reliable and fast, i.e. any of the above results also apply to other families of interpolation points (x_k) as long as these points are asymptotically distributed according to $\mu_{\text{Cheb}}(x)$.

In particular, many software packages use the Chebyshev extreme points rather than the Chebyshev zeroes for interpolation, because the extreme points include the endpoints $x = \pm 1$ which is useful e.g. to include boundary conditions in PDEs.

Polynomial Approximation

Discussion

Recall: At the beginning of this lecture, I spent quite a few words motivating why we settle for interpolation and "$\|f - p\|_{[a,b]}$ small" rather than $p = \arg\min_{p \in \mathcal{P}} \|f - p\|_{[a,b]}$.

Now that we are experts on polynomial interpolation, we have all the necessary tools to return to this question.

Polynomial Approximation

Equioscillation theorem

Assume $f : [a, b] \to \mathbb{R}$ is continuous. Then, $\arg\min_{p \in \mathcal{P}_n} \|f - p\|_{[a,b]}$ exists and is unique, and we have

$$p = \arg\min_{p \in \mathcal{P}_n} \|f - p\|_{[a,b]}$$

if and only if there are $n + 2$ points

$$a \leq x_1 < x_2 < \ldots < x_{n+1} < x_{n+2} \leq b$$

such that for some $s \in \{-1, 1\}$ we have

$$f(x_k) - p(x_k) = s\,(-1)^k \|f - p\|_{[a,b]} \qquad \text{for all } k \in \{1, \ldots, n+2\}.$$

Proofs. Similar to the proof on slide 35. See Trefethen (2013), Approximation Theory and Approximation Practice, Theorem 10.1 for details.

Polynomial Approximation

Example

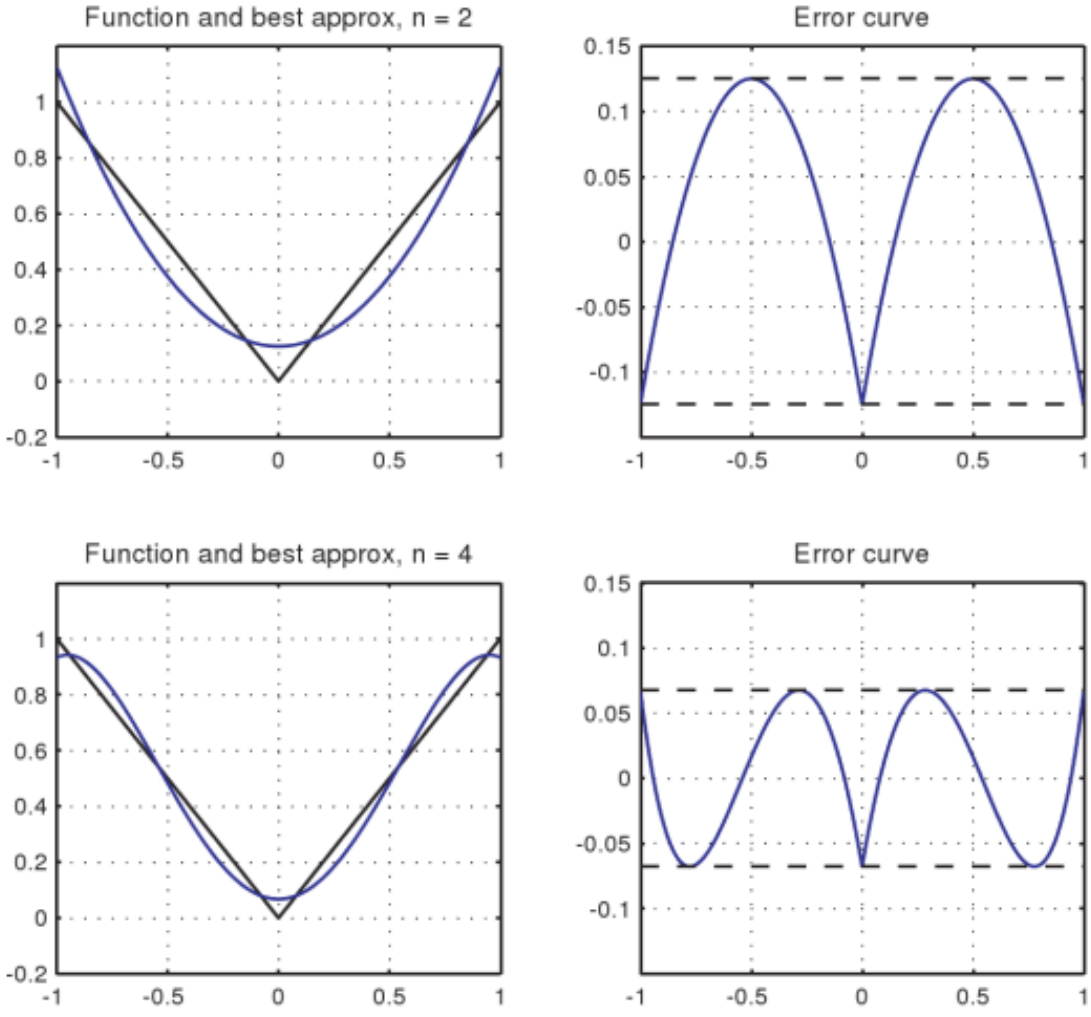


Figure copied from Trefethen (2013), Approximation Theory and Approximation Practice.

Note that these best approximation errors equioscillate in $n+3$ points rather than $n+2$ points as required by the equioscillation theorem.

Polynomial Approximation

Discussion

The equioscillation theorem in particular implies that the best approximation $p \in \mathcal{P}_n$ interpolates the original function $f(x)$ in at least $n+1$ points. The best approximation can hence be determined by tweaking the interpolation points until we achieve equioscillation.

There exist concrete algorithms for doing so (e.g. Remez algorithm), but they are generally expensive and may fail to converge. Moreover, one can show that the Chebyshev interpolant $p_{\text{Cheb}} \in \mathcal{P}_n$ and best approximation $p_{\text{best}} \in \mathcal{P}_n$ are related by

$$\|f - p_{\text{Cheb}}\|_{[-1,1]} \leq \left(2 + \tfrac{2}{\pi} \log(n+1)\right) \|f - p_{\text{best}}\|_{[-1,1]},$$

i.e. best approximation is at most a logarithmic factor better than interpolation in Chebyshev zeros.

Ref: Trefethen (2013), Approximation Theory and Approximation Practice, Thm. 16.1.

For these reasons, best approximation is rarely used except in some special cases where $p_{\text{best}}(x)$ can be determined explicitly.

Polynomial Approximation

Summary

- Existence and uniqueness of polynomial interpolants
- Convergence theory for piecewise polynomial interpolation
- Chebyshev polynomials
- Convergence theory for interpolation in Chebyshev points
- Equioscillation theorem for best approximation

Further reading

- Trefethen (2013). *Approximation Theory and Approximation Practice*.