Chebyshev Picard Iteration

Fixed Point IVP Integration

A.C.

September 22, 2024

Outline

Starting Point

Consider:

$$\cos n\theta$$

How does this relate to $\cos\theta$?

$n+1$

$$
\begin{aligned}
\cos 0 &= 1 \\
\cos\theta &= \cos\theta \\
\cos 2\theta &= 2\cos^2\theta - 1 \\
&\vdots \\
\cos(n+1)\theta &= 2\cos\theta\cos n\theta - \cos(n-1)\theta
\end{aligned}
$$

See notes for proof.

Recurrence

Let $T_n(\cos\theta) := \cos n\theta$ and $x := \cos\theta$ and observe the following recurrence

$$
\begin{aligned}
T_0 &= 1 \\
T_1 &= x \\
T_{n+1} &= 2xT_n - T_{n-1}
\end{aligned}
$$

In closed form: $T_n(x) = \cos n \cos^{-1} x$

Polynomial

$$T_{n+1} = 2xT_n - T_{n-1}$$

$2x$ multiplier increases degree of each subsequent T_n by exactly one. Easy to see that $\{T_0, T_1, \ldots, T_n\}$ spans forms a basis set for $\mathbb{P}_n$.

More directly, we can find coefficients $c_0, c_1, \ldots, c_n$ for any polynomial p_n s.t.

$$p_n = c_0 T_0 + c_1 T_1 + \cdots + c_n T_n$$

Domain

The recurrence form of Chebyshev polynomials (of the first kind) applies across the whole real line.

The cosine form requires we restrict ourselves to the interval $[-1, 1]$.

We will take it as a given that we are in the range $[-1, 1]$ throughout.

Integral

$$\int T_k = \frac{1}{k+1} T_{k+1} - \frac{1}{k-1} T_{k-1} + C$$

See notes for proof.

Key Result

If we have a polynomial in Chebyshev-space, then we can integrate it easily without changing basis (e.g. to the "usual" basis $1, x, x^2, \ldots$, where integration is even simpler).

Lagrange Interpolation

Recall: a polynomial of degree n is uniquely determined by $n+1$ points $(x_0, y_0), \ldots, (x_n, y_n)$.

Langrange interpolation is a standard method for constructing such a polynomial:

$$p_n(x) = \sum_i y_i \prod_{j \neq i} \frac{x - x_j}{x_i - x_j}$$

What if we would like the polynomial in terms of the Chebyshev basis?

System of Equations

Given points $(x_0, y_0), \ldots, (x_n, y_n)$, can write and solve a system of equations

$$\begin{aligned}
c_0 T_0(x_0) + c_1 T_1(x_0) + \cdots + c_n T_n(x_0) &= y_0 \\
c_0 T_0(x_1) + c_1 T_1(x_1) + \cdots + c_n T_n(x_1) &= y_1 \\
&\vdots \\
c_0 T_0(x_n) + c_1 T_1(x_n) + \cdots + c_n T_n(x_n) &= y_n
\end{aligned}$$

Fairly heavyweight computation, would prefer a cheaper solution.

Orthogonality

Define

$$\langle f, g\rangle = \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} fg \, dx$$

It can be shown that

$$\langle T_j, T_k\rangle = \begin{cases} 0, & \text{if } j \neq k \\ \pi, & \text{if } j = k = 0 \\ \frac{\pi}{2}, & \text{if } j = k > 0 \end{cases}$$

That is, Chebyshev polynomials form an *orthogonal* basis over $\mathbb{P}_n$. See notes for proof.

Inner Product Interpolation

Consider the equation

$$p_n = c_0 T_0 + c_1 T_1 + \cdots + c_n T_n$$

Take the inner product against T_k on each side to see

$$\langle p_n, T_k \rangle = c_k \langle T_k, T_k \rangle$$

If we can compute $\langle p_n, T_k \rangle$ efficiently, then we have a fast procedure for acquiring the polynomial in Chebyshev form.

Quadrature

It is far from obvious but Lobatto quadrature at Chebyshev nodes $x_j = \cos\frac{j\pi}{n}$ computes $\langle p_n, T_k \rangle$ exactly, giving us the formula

$$\langle p_n, T_k \rangle = \frac{\pi}{n} \sum_j{}'' \, p_n\left(\cos\frac{j\pi}{n}\right) T_k\left(\cos\frac{j\pi}{n}\right)$$

See notes for proof.

Key Result

If we choose to interpolate a function f at $n+1$ Chebyshev nodes, then the Chebyshev form of the resulting polynomial may be efficiently computed through simple sums.

Weierstrass Approximation Theorem

Given any smooth function f on a compact interval $[a, b]$ and an arbitrary error tolerance ϵ, there exists a polynomial p s.t.

$$||p - f||_{\infty} \leq \epsilon$$

That is, we may approximate any smooth function arbitrarily well on a finite interval with *some* polynomial.

It seems a fair guess that interpolating f at a large number of points should yield such a polynomial.

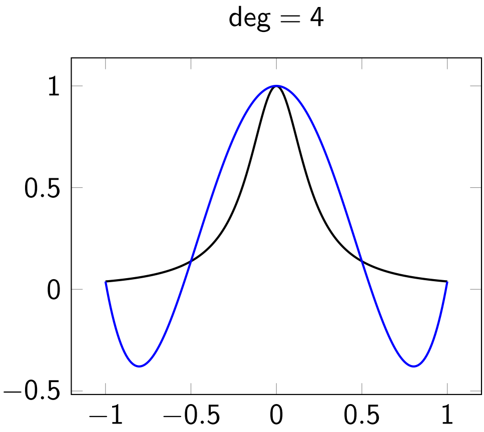
Runge's Phenomenon

Let us consider the behavior of polynomial interpolants on the following function

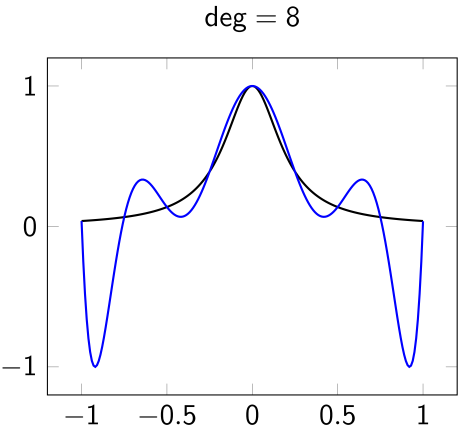
$$y = \frac{1}{1 + 25x^2}$$

We'll do the obvious thing, and select equally spaced points for evaluation.

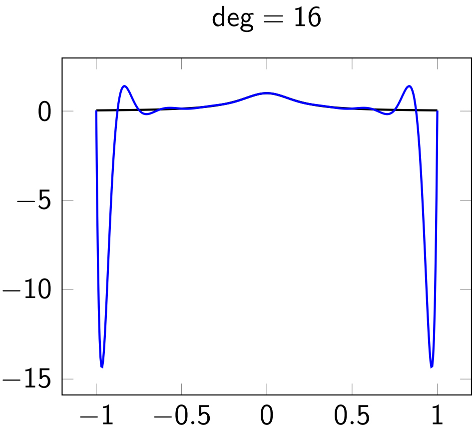
Runge's Phenomenon



Runge's Phenomenon



Runge's Phenomenon



Lagrange Error Formula

Let f be a smooth function on a compact interval, and p_n be the polynomial interpolating it at points $x_0, x_1, \ldots, x_n$. Then

$$f(x) = p_n(x) + \frac{f^{(n+1)}(\eta_x)}{(n+1)!} \prod (x - x_j)$$

See notes for proof.

Controlling Approximation Error

Our error term has two components – a derivative part, and a product part.

It is difficult to fathom a mechanism to control the former, but the latter should be directly impacted by our choice of interpolating points.

Chebyshev Node Error

Set $x_0, \ldots, x_n$ to the zeros of the n-th Chebyshev polynomial. We observe the following bounds on the "product" component of the Lagrange error, which no other choice of nodes can beat.

$$\max_{[-1,1]} \prod (x - x_j) = \frac{1}{2^n}$$

The extrema of the n-th Chebyshev polynomial are actually of more practical interest to us, but the result for them is the same, up to a constant factor. See notes for proof.

Key Result

Interpolating through Chebyshev nodes minimizes a key term underpinning L_∞ error.

We have not shown that Chebyshev nodes are always best (in fact, there exists no choice of nodes with is best for all polynomial-approximable functions), but it is very good, very often.

Outline

Outline