CS 450: Numerical Anlaysis[1]

Interpolation

University of Illinois at Urbana-Champaign

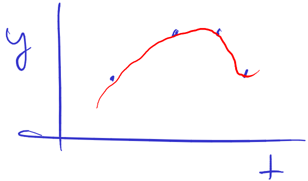
[1] *These slides have been drafted by Edgar Solomonik as lecture templates and supplementary material for the book "Scientific Computing: An Introductory Survey" by Michael T. Heath (slides).*

Interpolation

- Given $(t_1, y_1), \ldots, (t_m, y_m)$ with *nodes* $t_1 < \cdots < t_m$ an *interpolant* f satisfies:

$f(t_i) = y_i \leftarrow$ exact at the nodes

error measured at other t



interpolants are not unique

polynomial interpolants of degree $m-1$ are unique

- Interpolant is usually constructed as linear combinations of *basis functions* $\{\phi_j\}_{j=1}^n = \phi_1, \ldots, \phi_n$ so $f(t) = \sum_j x_j \phi_j(t)$.

if $n = m \Rightarrow$ interpolant is unique

$n > m \Rightarrow$ many interpolants

$n < m \Rightarrow$ may not have an interpolant

$V(\{t_i\}_{i=1}^n, \{\phi_j\}_{j=1}^n) = A$

$f(t_1) = x_1 \phi_1(t_1) + x_2 \phi_2(t_1)$

$a_{ij} = \phi_j(t_i)$

$$Ax = y \Rightarrow \begin{bmatrix} \phi_1(t_1) & \phi_2(t_1) \\ \phi_1(t_2) & \phi_2(t_2) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

Polynomial Interpolation

- The choice of *monomials* as basis functions, $\phi_j(t) = t^{j-1}$ yields a degree $n-1$ polynomial interpolant:

$$A = \begin{bmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_n & t_n^2 & \cdots & t_n^{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \Rightarrow f(t) = x_1 + x_2 t + x_3 t^2 \ldots$$

- Polynomial interpolants are easy to evaluate and do calculus on:

evaluate → use Horner's rule

$$f(t) = x_1 + t(x_2 + t(x_3 + t(x_4 + \ldots)))$$

$n-1$ adds and multiplies

to differentiate/integrate, require $O(n)$ operations

Conditioning of Interpolation

- Conditioning of interpolation matrix $\boldsymbol{A}$ depends on basis functions and coordinates $t_1, \ldots, t_m$:

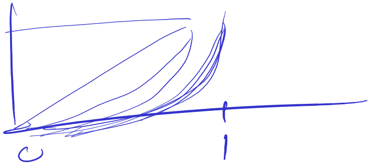
nodes

basis function ↔ column of A, so if $\varphi_i \in \mathrm{span}\{\varphi_j\}_{j \neq i}$ matrix has $\infty = \kappa(A)$

if $t_i = t_j \Rightarrow$ two rows of A are the same

monomial

- The Vandermonde matrix tends to be ill-conditioned:



Lagrange Basis

- n-points fully define the unique $(n-1)$-degree polynomial interpolant in the *Lagrange basis*:

$$\varphi_i(t) = \frac{\prod_{j \neq i} (t - t_j)}{\prod_{j \neq i} (t_i - t_j)}$$

$$a_{ij} = \varphi_i(t_j) = \begin{cases} 0 & ; \text{ if } i \neq j \\ 1 & ; \text{ if } i = j \end{cases}$$

So $A = I$

- Lagrange polynomials yield an ideal Vandermonde system, but the basis functions are hard to evaluate and do calculus on:

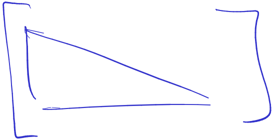
much more work to evaluate and differentiate, and this can be error-prone

Newton Basis

- The *Newton basis* functions $\phi_j(t) = \prod_{k=1}^{j-1}(t - t_k)$ with $\phi_1(t) = 1$ seek the best of monomial and Lagrange bases:

$\phi_j(t_i) = 0$ if $i < j$

to solve $Ax = y$, cost is $O(n^2)$

$A =$ 

evaluate $\phi_j(t) = \phi_{j-1}(t)(t - t_{j-1})$

- The Newton basis yields a triangular Vandermonde system:

$O(n)$ work to find $f(t)$

Cost of constructing A is $O(n^2)$ work

divided-differences that solves $Ax = y$ with $O(n^2)$ work and better stability

Orthogonal Polynomials

- Recall that good conditioning for interpolation is achieved by constructing a well-conditioned Vandermonde matrix, which is the case when the columns (corresponding to each basis function) are orthonormal. To construct robust basis sets, we introduce a notion of *orthonormal functions*:

inner product: $\langle f, g \rangle_w = \int_{-\infty}^{\infty} f(x)\, g(x)\, w(x)\, dx$

is function

weight function

f is orthogonal to g, $\langle f, g \rangle_w = 0$

$$\|f\| = \sqrt{\langle f, f \rangle_w}$$

Legendre Polynomials

- The Gram-Schmidt orthogonalization procedure can be used to obtain an orthonormal basis with the same span as any given arbitrary basis:

$e_1, \dots, e_k$ orthonormal (each $\|e_i\| = 1$)

$$e_{k+1} = f_{k+1} - \sum_{\ell=1}^{k} \langle e_\ell, f_{k+1} \rangle e_\ell$$

- The *Legendre polynomials* are obtained by Gram-Schmidt on the monomial basis, with $w(t) = \begin{cases} 1 : -1 \le t \le 1 \\ 0 : \text{otherwise} \end{cases}$ and normalized so $\boxed{\hat{\phi}_i(1) = 1}$.

weight function

$\varphi_3(t) = \dfrac{3t^2 - 1}{2}$

$e_3(1) = 1$

$f_1, f_2, f_3 = \{1, t, t^2\}$

$e_1(t) = 1$

$e_2(t) = t - \int_{-1}^{1} 1 \cdot t \, dt = t$

$e_3(t) = t^2 - \left(\underbrace{\int_{-1}^{1} 1 \cdot t^2 \, dt}_{\frac{1}{3} + \frac{1}{3}} - \underbrace{\int_{-1}^{1} t \cdot t^2 \, dt}_{0}\right)$

$= t^2 + 2/3$

Chebyshev Basis

- *Chebyshev polynomials* $\phi_j(t) = \cos((j-1)\arccos(t))$ and *Chebyshev nodes* $t_i = \cos\left(\frac{2i-1}{2n}\pi\right)$ provide a way to pick *nodes* $t_1, \ldots, t_n$ along with a basis, to yield perfect conditioning:

$$\varphi_1(t) = 1, \quad \varphi_2(t) = t, \quad \varphi_{i+1}(t) = \underbrace{2t\varphi_i(t) - \varphi_{i-1}(t)}_{\text{recurrence}}$$

$$\langle \varphi_i, \varphi_j \rangle_w = \delta_{ij} = 1 \text{ iff } i=j, \text{ 0 otherwise}$$

$$w = \begin{cases} \frac{1}{(1-t^2)^{1/2}} & ; -1 \le t \le 1 \\ 0 & ; \text{otherwise} \end{cases}$$

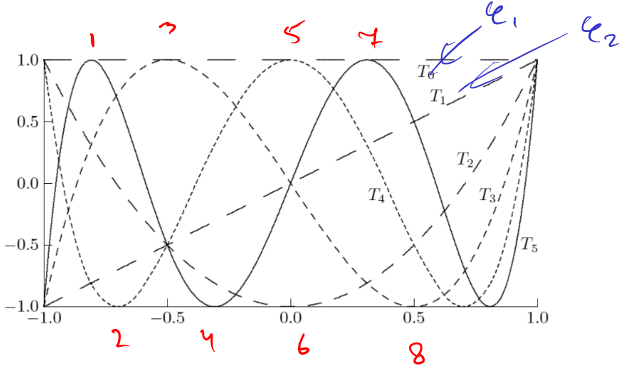
Vandermonde matrix $A = [a_1, \ldots, a_n]$

$$\langle a_i, a_j \rangle = \sum_{k=1}^{n} \varphi_i(t_k)\varphi_j(t_k)$$

$$= \sum_{k=1}^{n} \cos\left(\frac{(i-1)(2k-1)}{2n}\pi\right)\cos\left(\frac{(j-1)(2k-1)}{2n}\pi\right)$$

Chebyshev Nodes Intuition

Demo: *Jump with Chebyshev Nodes*



- Note *equi-alteration* property, successive extrema of $T_k = \phi_k$ have opposite sign.
- Set of k Chebyshev nodes of are given by zeros of T_k.

Interpolation Error Bounds

- Polynomial interpolation satisfies the following error bound:

$$|E(t)| \leq \frac{\max_{s \in [t_1, t_n]} |f^{(n+1)}(s)|}{n!} \prod_{i=1}^{n} (t - t_i) \quad \text{for} \quad t \in [t_1, t_n]$$

- Letting $h = t_n - t_1$ (often also achieve same for h as the node-spacing $t_{i+1} - t_i$), we obtain

$$|E(t)| \leq \frac{\max_{s \in [t_1, t_n]} |f^{(n+1)}(s)|}{n!} h^n = O(h^n) \quad \text{for} \quad t \in [t_1, t_n]$$

Error in Interpolation

Given degree n polynomial interpolant $\tilde{f}$ of f induction on n shows that $E(t) = f(t) - \tilde{f}(t)$ has n zeros $t_1, \ldots, t_n$ and there etist $y_1, \ldots, y_n$ such that

$$E(t) = \int_{t_1}^{t} \int_{y_1}^{w_0} \cdots \int_{y_n}^{w_{n-1}} f^{(n+1)}(w_n) dw_n \cdots dw_0 \tag{1}$$

Demo: Composite Gauss Interpolation Error

Piecewise Polynomial Interpolation

- The kth piece of the interpolant is a polynomial in $[t_i, t_{i+1}]$

- *Hermite* interpolation ensures consecutive interpolant pieces have same derivative at each *knot* t_i:

Spline Interpolation

- A *spline* is a $(k-1)$-time differentiable piecewise polynomial of degree k:

- The resulting interpolant coefficients are again determined by an appropriate generalized Vandermonde system:

B-Splines

B-splines provide an effective way of constructing splines from a basis:

- The basis functions can be defined recursively with respect to degree.
- The ith degree k polynomial piece is positive on $[t_i, t_{i+k+1}]$ and zero everywhere else
- All possible splines of degree k with notes $\{t_i\}_{i=1}^{n}$ can be represented in the basis.