Better conditioning: Orthogonal polynomials

- What caused monomials to have a terribly conditioned Vandermonde?
- What's a way to make sure two vectors are *not* like that?
- But polynomials are functions!
- But how can I practically compute the Legendre polynomials?

$$(f,g) = \int_a^b f(x)\, g(x)\, \underline{w(x)}\, dx$$

Orthogonal $(f,g)=0$

$(P_n, P_m) = 0 \quad n \neq m$

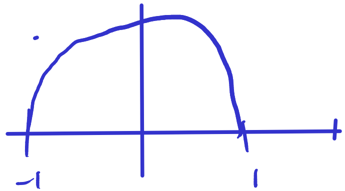
$1, x, x^2 \ldots$ —Gram-Schmidt→ Legendre polynomials

$P_0(x) = 1 \qquad P_1(x) = x$

$$(n+1)\, P_{n+1}(x) = (2n+1)\, x\, P_n(x) - P_{n-1}(x)$$

$x^2 + \sqrt{1-x^2}^2 = 1$

Another family of orthogonal polynomials: Chebyshev



Three equivalent definitions:

$y^2 + x^2 = 1$

y

- Result of Gram-Schmidt with weight $1/\sqrt{1-x^2}$
    - What is that weight?

    (Again, you won't exactly get the standard normalization if you do this.)
- $T_k(x) = \cos(k \cos^{-1}(x))$
- $T_k(x) = 2x\, T_k(x) - T_{k-1}(x)$ ←

    $+1$

Demo: Chebyshev interpolation part I

- What is the Vandermonde matrix for Chebyshev polynomials?

$x_i = \cos(\ \ldots\ )$ $\qquad (0, \pi)$

$\frac{i}{k}\pi \quad (i = 0 \ldots k)$

$$V_{ij} = \cos\left(j \underbrace{\cos^{-1}\left(\cos\left(\frac{i}{k}\pi\right)\right)}\right)$$

$$= \cos\left(\frac{ji}{k}\pi\right)$$

FFT

FCT

can be applied in $O(n \log n)$

Chebyshev nodes

Might also consider zeros (instead of roots) of T_k:

$$x_i = \cos\left(\frac{2i+1}{2k}\pi\right) \quad (i = 1\ldots, k).$$

The Vandermonde for these (with T_k) can be applied in $O(N \log N)$ time, too.

It turns out that we were still looking for a good set of interpolation nodes.

- We came up with the criterion that the nodes should bunch towards the ends. Do these do that?

Demo: Chebyshev interpolation part II

- Summary?

  - High-order poly interp? → Chebyshev polynomials w/ Chebyshev nodes

Error Result

intp. error

$$IE = \overbrace{f(x) - \underbrace{p_{n-1}}_{\text{degree } n-1}(x)} = \frac{f^{(n)}(\xi)}{n!}(x - x_1)(x - x_2)\cdots(x - x_n)$$

M

- Why does Chebyshev-like 'bunching' work?

It keeps the intp. error under control at more closely spaced points towards the interval edges

Assume: $|f^{(n)}(x)| \leq M$ for $x \in [x_1, \dots, x_n]$

$h = |x_n - x_1| \rightsquigarrow |x - x_i| \leq h$ for all $x \in [x_1, x_n]$

$$\max_{x \in [x_1, x_n]} |f(x) - p_{n-1}(x)| \leq C \cdot M \cdot h^n$$

\# nodes

$E(h) = O(h^n)$

$E(h) = 0.1$

Using cubic interpolation

$n-1 = 3$

$n = 4$

$E(h/2) \approx \left(\frac{h}{2}\right)^n = \left(\frac{h}{2}\right)^4$

$= \left(\frac{1}{2}\right)^4 \cdot \underbrace{h^4}_{E(h)}$

$= \left(\frac{1}{2}\right)^4 \cdot E(h)$

- Boil the error result down to a simpler form.

Going piecewise: Simplest Case

Construct a piecwelse linear interpolant at four points.

x_0, y_0		x_1, y_1		x_2, y_2		x_3, y_3
|	$f_1 = a_1 x + b_1$	|	$f_2 = a_2 x + b_2$	|	$f_3 = a_3 x + b_3$	|
|	2 unk.	|	2 unk.	|	2 unk.	|
|	$f_1(x_0) = y_0$	|	$f_2(x_1) = y_1$	|	$f_3(x_2) = y_2$	|
|	$f_1(x_1) = y_1$	|	$f_2(x_2) = y_2$	|	$f_3(x_3) = y_3$	|
|	2 eqn.	|	2 eqn.	|	2 eqn.	|

- Why three intervals?

Piecewise Cubic ('Splines')

Now do the same accounting for piecewise cubic at four points:

$$\begin{array}{ccccccc} x_0, y_0 & & x_1, y_1 & & x_2, y_2 & & x_3, y_3 \\ | & f_1 = a_1x^3 + b_1x^2 + c_1x + d_1 & | & f_2 = a_2x^3 + b_2x^2 + c_2x + d_2 & | & f_3 = a_3x^3 + b_3x^2 + c_3x + d_3 & | \end{array}$$