Interpolation

Def: An interpolating approximation to a function $f(x)$ is an expression $P_{n-1}(x)$ (polynomial) whose $n$ degrees of freedom are determined by the requirement that $P_{n-1}(x) = f(x)$ at a set of interpolation pts.

i.e. $P_{n-1}(x_i) = f(x_i)$ for $i = 1, \ldots, n$.

The pts $\{x_i\}_{i=1}^n$ are called interpolation or collocation pts.

How do we find $P_{n-1}(x)$?

Option 1:

We can write $P_{n-1}(x) = a_0 + a_1 x + \ldots + a_{n-1} x^{n-1}$.

Then use the fact that $P_{n-1}(x_i) = f(x_i)$ to make a linear system to solve for $a_0, a_1, \ldots, a_{n-1}$.

$$
\begin{bmatrix}
1 & x_1 & x_1^2 & \ldots & x_1^{n-1} \\
1 & x_2 & x_2^2 & \ldots & x_2^{n-1} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & x_n & x_n^2 & \ldots & x_n^{n-1}
\end{bmatrix} \begin{bmatrix}
a_0 \\
a_1 \\
\vdots \\
a_{n-1}
\end{bmatrix} = 
\begin{bmatrix}
f(x_1) \\
f(x_2) \\
\vdots \\
f(x_n)
\end{bmatrix}
$$

Then $\sqrt{\mathbf{a}} = \mathbf{f}$.

The matrix is called the Vandermonde matrix.

It is very ill-conditioned.
Option 2: We can write
\[ P_n(x) = \sum_{j=1}^{n} f(x_j) L_j(x) \]
where \( L_j(x) \) is the **Lagrange polynomial** satisfying \( L_j(x_i) = \delta_{ij} \).

This means
\[ L_j(x) = \prod_{j \neq i}^{n} \frac{x-x_j}{x_i-x_j} \]

Note: The Lagrange Polynomial is sometimes called the **Cardinal function**.

Q: How do we choose interpolation pts?

Ex: let's approximate
\[ f(x) = \exp(x) \text{ on } [-5, 5] \]
\[ g(x) = \frac{1}{1+x^2} \text{ on } [-5, 5] \]
using **equispaced pts**.

What do you observe?
To correct the problems we observe we will look to some theorems.

**Thm (Cauchy interpolation or Modified Taylor)**

Let \( f(x) \) have at least \( n+1 \) derivatives on the interval of interest \( I \) let \( P_n(x) \) be a Lagrange interpolant of degree \( n \). Then
\[
f(x) - P_n(x) = \frac{1}{(n+1)!} \sum_{i=0}^{n} \frac{f^{(n+1)}(\zeta)}{(n+1)!} (x-x_i)^{n+1}
\]

For some \( \zeta \) in the interval containing \( x \in [a, b] \) where \( x \in [x_i, x_{i+1}] \). 

**Notice:** The magnitude of the error will depend on \( \prod_{i=0}^{n} (x-x_i) \) i.e. it depends on the interpolation pts.

**Thm (Chebyshev minimal Amplitude Thm)**

Of all the polynomials of degree \( n \) with leading coefficient equal to 1, the unique polynomial which has the smallest maximum on \( [-1, 1] \) is
\[
T_n(x) = \frac{\sin((n+1)\theta)}{\sin(n\theta)}
\]

i.e. the \( n^{th} \) Chebyshev polynomial divided by \( 2^{n-1} \).

In other words, all polynomials of the same degree with leading coefficient of 1 satisfy
\[
\max_{x \in [-1, 1]} |P_n(x)| \geq \max_{x \in [-1, 1]} \left| T_n(x) \right| = \frac{1}{2^n}
\]

Since polynomials are unique,
\[
T_n(x) = \sum_{i=0}^{n} a_i (x-x_i)^i
\]
where \(x_i\) are the roots of \(T_n(x)\).

These are the optimal interpolation pts.
Note: other optimal choices exist depending on the choice of basis.

Important facts about Chebyshev polynomials:
- \(T_n(x) = \cos(n \cos^{-1}(x))\)
- Can be written as a 3 term recursion:
  \[
  \begin{align*}
  T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x) \\
  T_0 &= 1 \\
  T_1(x) &= x
  \end{align*}
  \]
- \(\int_{-1}^{1} \frac{T_n(x) T_m(x)}{\sqrt{1-x^2}} \, dx = 0\) if \(m \neq n\)

We can choose our polynomial approximation to be:
\[
P_n(x) = \sum_{i=0}^{n} a_i T_i(x)
\]

What if we do not want to evaluate \(P_n(x)\) to find the approximate solution? Value at a pt \(x^*\)?
- Roundoff in computing coefficients
- Roundoff in add. evaluating \(P(x)\) etc.
We can rewrite $P_n(x)$ so that we are solving for the approximate solution at the interpolation pts.

Let $\phi_{m+1}(x)$ denote the $(m+1)$ orthogonal polynomial in a basis. (Chebyshev, Legendre, etc.)

Let $\sum_{j=0}^{n} x_j$ denote the roots of $\phi_{m+1}$

In the vicinity of $x_j$, Taylor tells us

$$\phi_{m+1}(x) \approx \phi_{m+1}(x_j) + \frac{d\phi_{m+1}(x_j)}{dx}(x-x_j) - O((x-x_j)^2)$$

We can normalize the basis function by dividing it by $\phi_{m+1}(x_j)(x-x_j)$

Define $C_j(x) = \frac{\phi_{m+1}(x)}{\phi_{m+1}(x_j)(x-x_j)}$ to be the $j^{th}$ cardinal function.

Note $(j(x_i)) = \sum_{j=0}^{n} 1$ if $i=j$.

$$P_n(x) = \sum_{j=0}^{n} u_j C_j(x)$$

This works for Chebyshev, Legendre, Laguerre, Hermite, etc.
If you want to include endpoints \( x = \pm 1 \)
i.e. have a laballo grid. The cardinal functions
are given by
\[
\phi_j(x) = \frac{(1-x^2) \phi_j(x)}{\left( x-x_j \right)^2 \phi_{n+1}(x_j)} (x-x_j)
\]
\( j = 0, \ldots, n \)