Lagrange interpolation: What nodes?

The setting: Lagrange interpolation in the xy-plane with nodes $x_0 < x_1 < \ldots < x_n$ and knots (x_i, y_i), $\;i = 0\!:\!n$. In many situations the knots are given as input data, and we have no choice. But in some cases we do have such freedom. A common such case is where you have a function subroutine, say f.m, which returns $y = f(x)$ given an input value x (in Matlab: y = feval('f',x)). Another common case is an experimental measurement, where the nodes are free to be chosen in the experiment's setup (e.g. "where do I place my sensors?").

Here we think about good nodes and bad nodes. To make things simple we assume $x_j \in [-1, 1]$, typically called the *unit interval.* If $x_0 = -1$ and $x_n = 1$, we say the nodes are *closed* in $[-1, 1]$. If your nodes, say $\hat{x}_j$, are in the interval $[a, b]$, then we can linearly map from $[-1, 1]$ to $[a, b]$ by $\hat{x}_j = (a + b + (b - a)x_j)/2$.

Recall the Lagrange interpolator in Barycentric form

$$P(x) = \left(\sum_{i=0}^{n} \frac{y_j w_j}{x - x_j} \right) \Big/ \left(\sum_{i=0}^{n} \frac{w_j}{x - x_j} \right), \qquad \text{where} \qquad 1/w_j = \prod_{i=0, i \neq j}^{n} (x_j - x_i).$$

It turns out that equidistant nodes (a 'uniform grid') are a relatively bad choice for interpolation: the relative magnitudes of the w_j can differ by up to about 2^n. With equidistant nodes, as n increases small changes in y (say $y(1 + \epsilon)$) can give large (think $\epsilon 2^n$) changes in P: *interpolation on a uniform grid is an ill-conditioned problem.*

Rounding errors alone are enough to give big spurious oscillations near the ends of the grid even for $n < 20$. We *will* use Lagrange interpolation on uniform grids in this course, but we will keep $n \lesssim 5$. For comparison later, here is the $n\!=\!10$ closed uniform grid and w:

$$\begin{aligned} x &= [-1,\ -.80,\ -.60,\ -.40,\ -.20,\ 0,\ .20,\ .40,\ .60,\ .80,\ 1] \\ w &\approx [2.7,\ -27,\ 121,\ -323,\ 565,\ -678,\ 565,\ -323,\ 121,\ -269,\ 2.7] \end{aligned}$$

This isn't true for all node choices for a given interval. There are (infinitely) many sets of nodes on the unit interval for which Lagrange interpolation is well conditioned, even for n in the thousands. These 'nice' nodes all cluster near the endpoints, and the relative magnitudes between the w_j are not large. Chebyshev nodes are a common example (we will use a closed set of Chebyshev nodes). These are the real parts of a closed uniform grid on the upper half of the unit circle: $\quad x_j = -cos(j\pi/n), \quad j = 0\!:\!n.$
For example, if $n = 10$ we have the 11 Chebyshev (type 2) nodes and w:

$$\begin{aligned} x &\approx [-1,\ -.95,\ -.81,\ -.58,\ -.31,\ 0,\ .31,\ .58,\ .81,\ .95.\ 1] \\ w &= [25.6,\ -51.2,\ -51.2,\ -51.2,\ -51.2,\ -51.2,\ -51.2,\ -51.2,\ -51.2,\ -51.2,\ 25.6] \end{aligned}$$

If we think of Lagrange interpolation input data as an interval and a function f on that interval, then node selection is an important decision which can strongly effect the quality of the approximation. Choosing 'nice' nodes makes our approximation to f mostly depend on the smoothness of f. If f is smooth, then its frequency content (how 'wiggly' it is) puts a lower bound on n (see Nyquist frequency, or 'aliasing').