Lecture 22

Friday, February 28, 2020

Last time, we saw that the function $f(x) = \dfrac{1}{4x^2+1}$ is not well approximated by interpolation polynomials (with equally spaced sample points). The reason is that the higher derivative $f^{(n)}$ grows too quickly as $n \to \infty$. To explain this, we need to compute $f^{(n)}$. It is quite messy to compute derivatives of f from the given formula

$$f'(x) = \frac{-8x}{(4x^2+1)^2}$$

$$f''(x) = \frac{-8(4x^2+1)^2 - (-8x)\,2\,(8x)(4x^2+1)}{(4x^2+1)^4}$$

$$\cdots$$

getting complicated quickly

We need to find a better way to compute the derivatives of f. Let us consider another function that looks somewhat similar to f.

$$g(x) = \frac{1}{4x^2-1}$$

How to find the derivatives of g? If we differentiate g from the given formula, $g^{(n)}$ gets complicated very quickly. This is the same problem we had with f. Instead, one can rewrite g using partial fraction decomposition:

$$g(x) = \frac{1}{(2x-1)(2x+1)} = \frac{1}{2}\left(\frac{1}{2x-1} - \frac{1}{2x+1}\right).$$

The derivative of each fraction is easy to compute. For example,

$$\frac{1}{2x-1} = (2x-1)^{-1}$$

$$\left(\frac{1}{2x-1}\right)' = (-1)\,2\,(2x-1)^{-2}$$

$$\left(\frac{1}{2x-1}\right)'' = (-1)(-2)\,2^2\,(2x-1)^{-3}$$

$$\left(\frac{1}{2x-1}\right)^{(n)} = (-1)^n n!\, 2^n (2x-1)^{-n-1}$$

Similarly,

$$\left(\frac{1}{2x+1}\right)^{(n)} = (-1)^n n!\, 2^n (2x+1)^{-n-1}$$

Therefore,

$$g^{(n)}(x) = \frac{1}{2}\left[\left(\frac{1}{2x-1}\right)^{(n)} - \left(\frac{1}{2x+1}\right)^{(n)}\right]$$

$$= \frac{1}{2}(-1)^n n!\, 2^n \left[(2x-1)^{-n-1} - (2x+1)^{-n-1}\right].$$

The reason why we can find this formula for $g^{(n)}$ is that we can split g using partial fraction decomposition. It is not quite clear how to do likewise to function f. However, with the help of complex numbers, we can write partial decomposition for function f.

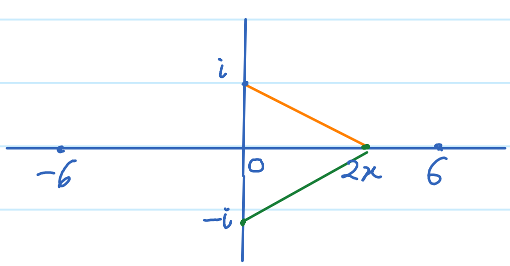
$$f(x) = \frac{1}{4x^2+1} = \frac{1}{(2x-i)(2x+i)} = \frac{1}{2i}\left(\frac{1}{2x-i} - \frac{1}{2x+i}\right).$$

From here we can take derivatives of f as we did with g:

$$f^{(n)}(x) = \frac{1}{2i}(-1)^n n!\, 2^n \left[\left(\frac{1}{2x-i}\right)^{-n-1} - \left(\frac{1}{2x+i}\right)^{-n-1}\right]$$

Let us take the absolute value of both sides:

$$|f^{(n)}(x)| \leq \frac{1}{2} n!\, 2^n \left(\frac{1}{|2x-i|^{n+1}} + \frac{1}{|2x+i|^{n+1}}\right)$$



Recall that the modulus of the complex number $a+bi$ is

$$|a+bi| = \sqrt{a^2+b^2}.$$

The modulus of $2x-i$ is the length of the orange bar. The modulus of $2x+i$ is the length of the green bar.

These lengths are at least equal to 1. Thus,

$$\max_{[-3,3]} |f^{(n)}(x)| \approx \frac{1}{2} n! \, 2^n (1+1) = n! \, 2^n.$$

We have

$$\max_{[-3,3]} |f(x) - P(x)| \leq \frac{1}{n}\left(\frac{b-a}{n-1}\right)^n \max |f^{(n)}(x)|$$

$$\sim \frac{1}{n}\left(\frac{6}{n-1}\right)^n n! \, 2^n$$

$$= \left(\frac{12}{n-1}\right)^n (n-1)!$$

$$\sim \frac{12}{n-1}\left(\frac{12}{n-1}\right)^{n-1} (n-1)!$$

$$\sim \frac{12}{n-1}\left(12 \, \frac{\sqrt[n-1]{(n-1)!}}{n-1}\right)^{n-1}$$

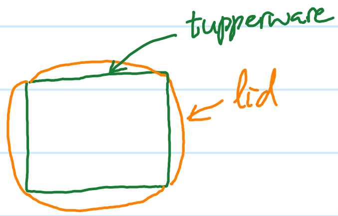
By Stirling's approximation, $\dfrac{\sqrt[n-1]{(n-1)!}}{n-1} \approx \dfrac{1}{e}$. Thus,

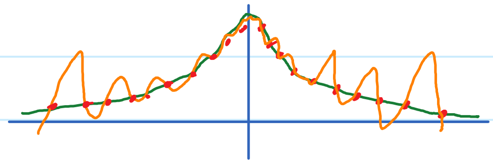
$$\max_{[-3,3]} |f(x) - P(x)| \lesssim \underbrace{\frac{12}{n-1}\left(\frac{12}{e}\right)^{n-1}}_{\text{goes to } \infty \text{ as } n \to \infty}$$

* Spline interpolation:

The advantage of fitting a given set of data point by a polynomial is that the polynomial is easy to find Indeed, by using Lagrange's formula, one immediately has an explicit form of the polynomial without any further computations.

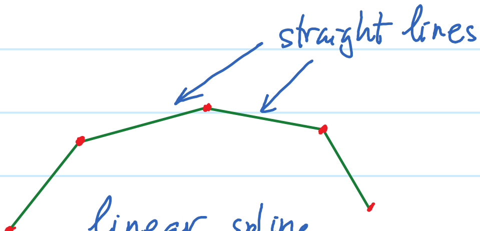
The drawback is that the polynomial curve may have unnatural oscilations near the ending points. One can liken this phenomenon to attempting to close a tupperware by a lid of different size. One may be able to fit the lid on the tupperware at two corners, maybe three. But the last corner is very difficult to fit, unless one is willing to deform the whole lid.



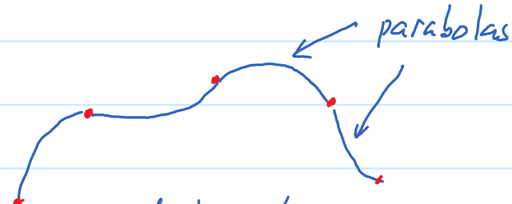


The function $\frac{1}{4x^2+1}$ is not a polynomial and doesn't behave like a polynomial. (A nonzero polynomial never decays as $x \to \pm\infty$.) One is able to "fit" a polynomial to this function at some given points, but the polynomial will largely deviate from other points. The larger the number of data points the larger the degree of the polynomial. The large degree causes the polynomial grows quickly near the ending points. This observation motivates another type of interpolation technique, called spline interpolation.

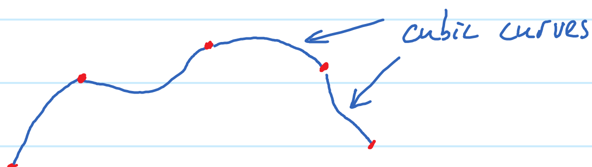
The idea of spline interpolation is that: instead of fitting n data point by a single polynomial, one fits any pair of consecutive points by a polynomial of small degree. Then find ways to joint these curves together.



linear spline



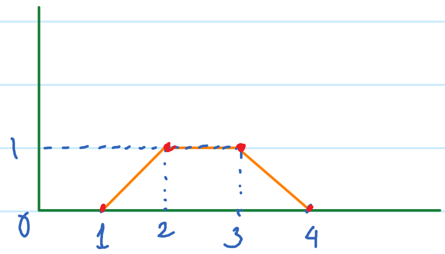
quadratic spline



cubic spline

The idea of spline interpolation is not new. For example, one can naturally interpolates the set of n points by connecting them by straight lines. This is called linear spline. The plot command of Matlab connects the given point by straight lines. For example,

```
x = [1, 2, 3, 4]
y = [0, 1, 1, 0]
plot(x,y)
```



Linear spline is simple. However, the sudden change of slope across each data point makes the curve look unnatural. If we allow the degree of each local curve to be 2 (i.e. parabola) then we can make the slope continuous across each data point. If we allow each local curve to have degree 3 (i.e. cubic curve) then we can make both first and second derivatives continuous across each data point.

Cubic splines are popular in applications. We will learn how to find quadratic splines next time. The computation of cubic splines is very similar.