Alexander Belyaev (belyaev@mpi-sb.mpg.de, http://www.mpi-sb.mpg.de/∼belyaev/gm06)

Interpolation and Approximation by Polynomials

Taylor Series Approximation

Consider a smooth function $f(x)$ defined on the segment $[a, b]$. Then for $x_0 \in [a, b]$ we have

$$f(x) = P_n(x) + E_n(x), \tag{1}$$

where $P_n(x)$ is the n-degree Taylor polynomial of $f(x)$ at $x = x_0$

$$P_n(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k. \tag{2}$$

The error function $E_n(x)$ is given by

$$E_n(x) = \frac{f^{(n+1)}(c)}{(n+1)!}(x - x_0)^{n+1} \tag{3}$$

where c is between x and x_0. The Taylor series approximation (1),(2),(3) generalizes the Mean Value Theorem

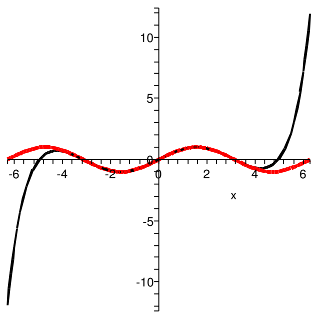
$$\frac{f(b) - f(a)}{b - a} = f'(c)$$

and (3) is called the *Lagrange form of the remainder.*

One can use (1) and (2) to derive finite-difference formulas approximations of derivatives of function $f(x)$:

$$\begin{aligned} f'(x) &\approx \frac{f(x+h) - f(x)}{h}, \\ f''(x) &\approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}, \quad \text{etc.} \end{aligned}$$

The main limitation of (1),(2) is that it delivers a local approximation only. For example, the figure below shows plots of $\sin x$ and its Taylor polynomial $P_9(x) = x - x^3/3! + x^5/5! - x^7/7! + x^9/9!$



Lagrange Interpolation

A classical interpolation problem addressed below is as follows. Let $n+1$ distinct points (nodes) $x_0, x_1, \ldots, x_n$ be given together with values $f_0, f_1, \ldots, f_n$, at the nodes. We want to construct function $f(x)$ which interpolates the given data:

$$f(x_i) = f_i, \qquad i = 0, \ldots, n.$$

Probably the simplest solution to the above interpolation problem consists of constructing a polynomial passing through given points $(x_0, f_0), (x_1, f_1), \ldots, (x_n, f_n)$. These $n+1$ points are sufficient to define uniquely a polynomial

$$P(x) = c_0 + c_1 x + \ldots + c_n x^n$$

of degree n passing through $x_0, x_1, \ldots, x_n$. Indeed we have $n+1$ equations for $n+1$ unknowns $c_0, c_1, \ldots, c_n$. However, solving the corresponding system of linear equations is computationally expensive. A closed-form solution described below is due to Lagrange.

Lagrange interpolating polynomial. Let us define the polynomials

$$L_k(x) = \prod_{i=0,\, i\neq k}^{n} \frac{(x - x_i)}{(x_k - x_i)}, \qquad L_k(x_j) = \delta_{kj} = \begin{cases} 1 & j = k \\ 0 & j \neq k \end{cases}$$

Now the Lagrange interpolating polynomial is given by

$$L(x) = \sum_{j=0}^{n} f_j L_j(x). \tag{4}$$

For example,

$$L(x) = A\frac{(x-b)(x-c)}{(a-b)(a-c)} + B\frac{(x-c)(x-a)}{(b-c)(b-a)} + C\frac{(x-a)(x-b)}{(c-a)(c-b)}$$

interpolates the points (a, A), (b, B), (c, C).

Barycentric form of Lagrange interpolating polynomial. One can rewrite (4) as

$$L(x) = l(x)\sum_{j=0}^{n} \frac{w_j}{x - x_j} f_j, \quad l(x) = (x - x_0)(x - x_1)\ldots(x - x_n),$$

where

$$w_j = 1 \Bigg/ \prod_{k=0,\, k\neq j}^{n} (x_j - x_k) \,.$$

Thus

$$L_j(x) = l(x)\frac{w_j}{x - x_j}.$$

Since

$$1 = \sum_{j=0}^{n} L_j(x) = l(x)\frac{w_j}{x - x_j}$$

we arrive at the following *barycentric* form

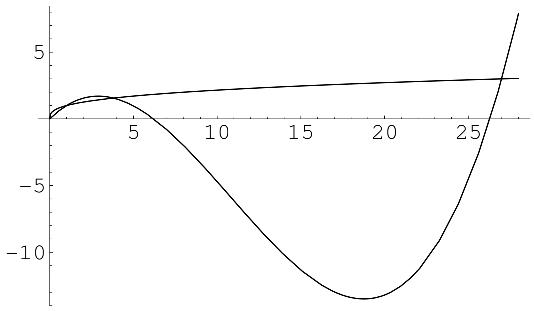
$$L(x) = \sum_{j=0}^{n} \frac{w_j}{x - x_j} f_j \Bigg/ \sum_{j=0}^{n} \frac{w_j}{x - x_j}.$$

Example of Lagrange interpolation. Consider the curve $y = x^{1/3}$ and the control points $(0, 0)$, $(1, 1)$, $(8, 2)$, and $(27, 3)$. The Lagrange interpolation yields

$$L(x) = \frac{x(x-8)(x-27)}{182} - \frac{x(x-1)(x-27)}{532} + \frac{x(x-1)(x-8)}{4446}.$$

A comparison between $y = x^{1/3}$ and $y = L(x)$ shows that the interpolation polynomial delivers a poor approximation of the original function.

Usually polynomial interpolation works well for low number of control points. High-degree interpolation polynomials often have undesirable oscillations.



One can overcome the problem of unwanted oscillations by using a low-degree piecewise polynomial curve, i. e. a composite interpolation curve assembled from low-degree polynomials in a piecewise manner.

Shepard's Interpolation Scheme.

The general problem we are addressing here is as follows. Given a set of n data points $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ in $\mathbb{R}^N$ with associated data values $f_0, f_1, \ldots, f_n$, we seek an interpolating function $f(\boldsymbol{x})$ such that $f(\boldsymbol{x}_i) = f_i$.

An elegant and simple solution to the above interpolation problem was given by D. Shepard in 1968. Let us construct $f(\boldsymbol{x})$ as a weighted sum of f_i:

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} w_i(\boldsymbol{x}) f_i \Big/ \sum_{i=1}^{n} w_i(\boldsymbol{x})$$

where positive weighting functions $w_i(\boldsymbol{x})$ become infinite at their corresponding data points:

$$w_i(\boldsymbol{x}) \to \infty \quad \text{as } \boldsymbol{x} \to \boldsymbol{x}_i.$$

A typical choice of the weighting functions is

$$w_i(\boldsymbol{x}) = 1/|\boldsymbol{x} - \boldsymbol{x}_i|^\alpha, \quad \alpha > 0.$$

Hermite Interpolation

Hermite interpolating polynomials are used when, in addition to to values $f_0, f_1, \ldots, f_n,$, the gradients (tangent directions) $f_i' = (df/dx)_i$ at the nodes $x_0, x_1, \ldots, x_n$ are specified. The Hermite interpolating polynomials $H_k(x)$ and $\bar{H}_k(x)$ have the following properties:

$$H_k(x_j) = \delta_{kj}, \quad H_k'(x_j) = 0, \quad \bar{H}_k(x_j) = 0, \quad \bar{H}_k'(x_j) = \delta_{kj}.$$

One can express the Hermite interpolating polynomials via Lagrange interpolation polynomials

$$\begin{aligned} H_k(x) &= \left[1 - 2L_k'(x_k)(x - x_k)\right][L_k(x)]^2, \\ \bar{H}_k(x) &= (x - x_k)[L_k(x)]^2. \end{aligned}$$

The Hermite polynomial interpolation is given by

$$H(x) = \sum_{k=0}^{n} f_k H_k(f) + \sum_{k=0}^{n} f_k' \bar{H}_k(f).$$

The Hermite cubic polynomials for the segment $0 \le x \le 1$ are

$$\begin{array}{ll} H_0(x) = 2x^3 - 3x^2 + 1 & H_1(x) = -2x^3 + 3x^2 \\ \bar{H}_0(x) = x^3 - 2x^2 + x & \bar{H}_0(x) = x^3 - x^2 \end{array}$$

Least Square Fitting

Fitting a line to data. Suppose we have data points

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \tag{5}$$

and want to find a straight line

$$y = ax + b$$

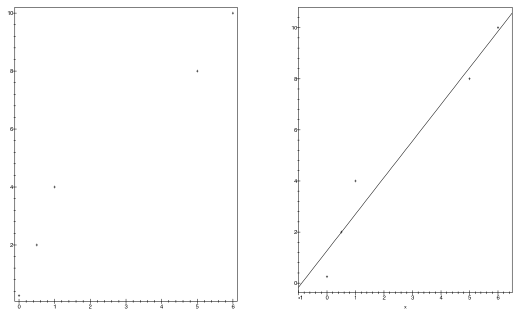
that best fit the data. Let us define the error function

$$E(a, b) = \sum_{i=1}^{n} (ax_i + b - y_i)^2$$

and search for the values of a and b where the error $E(a, b)$ achieves a minimum. It leads to the following two linear equations

$$\frac{\partial E}{\partial a}(a, b) = 0, \quad \frac{\partial E}{\partial b}(a, b) = 0$$

from which the desired values of a and b can be easily found.



Fitting a circle to data. The general equation of a circle centered at (a, b) with a radius R is

$$(x - a)^2 + (y - b)^2 = R^2$$

One reasonable measure of the distance between given points (5) and a circle is

$$E(a, b, R) = \sum_{i=1}^{n} \left[(x_i - a)^2 + (y_i - b)^2 - R^2\right]^2 \tag{6}$$

One difficulty with this is that E is not quadratic in a, b, and R. However if we let $c = a^2 + b^2 - R^2$, then the function $E(a, b, c)$ is quadratic in a, b, and c. Thus the system

$$\frac{\partial E}{\partial a}(a, b, c) = 0, \quad \frac{\partial E}{\partial b}(a, b, c) = 0, \quad \frac{\partial E}{\partial c}(a, b, c) = 0$$

is a system of linear equations.

Instead of (6) one can consider

$$E(a, b, R) = \sum_{i=1}^{n} w_i \left[(x_i - a)^2 + (y_i - b)^2 - R^2\right]^2$$

where the weights w_i indicate which points should be followed more closely, which less.

Problems

1. The quadratic equation

$$\frac{(x-b)(x-c)}{(a-b)(a-c)} + \frac{(x-c)(x-a)}{(b-c)(b-a)} + \frac{(x-a)(x-b)}{(c-a)(c-b)} = 1$$

has three real roots: $x = a$, $x = b$, and $x = c$. Do we have a contradiction?

2.

3.

4.