CS5016
Computational Methods and Applications

112001051
VM Sreeram

08/03/23

Least Square Function Approximations

Sribe 05

Interpolation vs curve fitting

In interpolation, we're given n points and have to find a curve that fits them.

For m points there's a unique polynomial of degree m-1 that fits them. Say there are 2 functions f and g such that $f(x_i) = g(x_i) = y_i$ $\forall i$ in $\{1, \ldots, n+1\}$. The degree of the polynomial $f-g$ is upper bounded by n. If $f \neq g$, then $f-g$ has at most n roots. But we know $f-g$ has at least $n+1$ roots. So $f=g$. Hence there is only one unique polynomial.

Say we want to find a cubic poly that fits 5 points. We cannot discard one point as we donot know if that point is indeed correct or has error/noise.

We need a method to measure loss/fitness of a particular curve to given set of points.

Parameterized functions and measure of fitness

Let f_θ be a fn param by θ - scalar / vector / finite or countable sequence.

eg.

$f_\theta(x) = \sin(\theta x)$ or $f_\theta(x) = \sum_{i=0}^{k} \theta_i x^i$

Natural measure of fit is

$$\sum_{i=1}^{m} (y_i - f_\theta(x_i))^2$$

We can find best fit curve as f_θ

$$\theta = \arg\min_{\theta} \sum_{i=1}^{m} (y_i - f_\theta(x_i))^2$$

Best fit line (error is)

$$\min_{a_0, a_1} \sum_{i=1}^{m} [y_i - (a_0 + a_1 x_i)]^2$$

for Best fit poly of degree n error is

$$E = \min_{a_0, a_1, \dots a_n} \sum_{i=1}^{m} \left(y_i - \left(\sum_{j=0}^{n} a_j x_i^j \right) \right)^2$$

Taking partial derivative wrt a_k

$$\frac{\partial E}{\partial a_k} = 2 \sum_{i=1}^{m} \left(\left(y_i - \sum_{j=0}^{n} a_j x^j \right) \cdot (-x_i^k) \right)$$

Since

$$\frac{\partial E}{\partial a_k} = 0,$$

$$\sum_{i=1}^{m} y_i x_i^k = \sum_{j=0}^{n} a_j \left(\sum_{i=1}^{m} x_i^{j+k} \right) \quad \forall k \in \{0, 1, \dots, n\}$$

Least square approx. of a fn using monomial poly.

Given a function $f(x)$ continuous on $[a,b]$, find a poly. $P_n(x)$ of degree at most n

$$P_n(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$$

such that integral of square of error is minimized, i.e.,

$$\min_{a_0, \dots, a_n} \int_a^b (f(x) - P_n(x))^2 \, dx$$

Normal equation

$$\sum_{k=0}^{n} a_k \int_a^b x^{j+k} \, dx = \int_a^b x^j f(x) \, dx \quad \forall j \in \{0, 1, \dots, n\}$$

The above normal eqn has matrix form $Sa = b$. S is often ill conditioned

→ when a small change in coefficients can cause a large change in the solution matrix is said to be ill-conditioned.

- If some matrix is ill conditioned, then solving using numerical methods becomes hard.

It can be made computationally effective using orthogonal polynomials.

Orthogonal Functions

A set of functions $\{\phi_1, \phi_2, \ldots, \phi_n\}$ in $[a,b]$ is orthogonal w.r.t. $w(x)$ (weight fn) if

$$\int_a^b w(x)\,\phi_i(x)\,\phi_j(x)\,dx = \begin{cases} 0 & \text{if } i \neq j \\ C_j & \text{otherwise} \end{cases}$$

where c_j is a positive real number. If $C_j = 1 \;\forall j$ then it is called an orthonormal set.

Using Orthogonal Functions

We are interested in finding least square approximation of $f(x)$ on $[a,b]$ by means of a poly. of the form

$$Q_n(x) = \sum_{i=0}^{n} a_i \phi_i(x)$$

where $\{\phi_i\}_{i=0}^{n}$ is a set of orthogonal polynomials on $[a,b]$ such that the least sq error is minimised, i.e.,

$$\min_{a_0, a_1, \ldots, a_n} \int_a^b w(x)\cdot(f(x) - Q_n(x))^2\,dx$$

Setting partial derivatives to 0, we get

$$\int_a^b w(x)\,\phi_j(x)\,f(x)\,dx = \int_a^b w(x)\,\phi_j(x)\left(\sum_{i=0}^{x} \phi_i(x)\right)dx = C_j a_j$$

Or,

$$a_j = \frac{1}{C_j}\int_a^b w(x)\,\phi_j(x)\,f(x)\,dx \qquad \forall j \in \{0, 1, \ldots, n\}$$

where

$$C_j = \int_a^b w(x)\,\phi_j^2(x)\,dx$$

Legendre Polynomial

$$L_n(x) = \frac{1}{2^n\, n!} \frac{d^n}{dx^n} (x^2-1)^n$$

The above polynomials are orthogonal in the interval $[-1,1]$ wrt weight function $w(x)=1$.

First 3 Legendre Poly:

$$L_1(x) = \frac{1}{2^1 . 1!} \frac{d}{dx} (x^2-1) = x$$

$$L_2(x) = \frac{1}{2^2 . 2!} \frac{d^2}{dx^2} (x^2-1)^2 = \frac{1}{8} . (12x^2-4) = \frac{3}{2}x^2 - \frac{1}{2}$$

$$L_3(x) = \frac{1}{2^3 . 3!} \frac{d^3}{dx^3} (x^2-1)^3 = \frac{5}{2}x^3 - \frac{3}{2}x$$

Orthogonality verification

$$\int_{-1}^{1} L_1(x) L_2(x)\, dx = \int_{-1}^{1} \left(\frac{3}{2}x^3 - \frac{1}{2}x\right) dx$$

$$= \int_{-1}^{1} \left(\frac{3}{2}(-x)^3 - \frac{1}{2}(-x)\right) dx \qquad \left[\int_a^b f(x)\,dx = \int_a^b f(b+a-x)\,dx\right]$$

adding,

$$= 0$$

$$\int_{-1}^{1} L_1(x) L_3(x)\, dx = \int_{-1}^{1} \left(\frac{5}{2}x^4 - \frac{3}{2}x^2\right) dx = \left.\frac{x^5 - x^3}{2}\right|_{-1}^{1} = 0.$$

$$\int_{-1}^{1} L_2(x) L_3(x)\, dx = \int_{-1}^{1} \left(\frac{5}{2}x^3 - \frac{3}{2}x\right)\left(\frac{3}{2}x^2 - \frac{1}{2}\right) dx = \left.\frac{10x^6 - 29x^4 + 24x^2}{16}\right|_{-1}^{1} = 0.$$

Chebyshev Polynomial

$$T_n(x) = \cos(n \cos^{-1}(x))$$

Is $T_n(x)$ really a polynomial?

Yes. It may be proved by induction

$T_0(x) = 1$

$T_1(x) = x$

$T_{n+1}(x) = 2x T_n(x) - T_{n-1}(x)$

They are orthogonal in the interval $[-1, 1]$ wrt wt fn $w(x) = \frac{1}{\sqrt{1-x^2}}$

Fourier Series

For any +ve int n,

$\{\cos(0), \cos(x), \dots \cos(nx), \sin(0), \dots, \sin(nx)\}$ is orthogonal in the interval $[-\pi, \pi]$ wrt wt fn $w(x) = 1$.

Let

$$S_n(x) = \frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos(kx) + \sum_{k=1}^{n} b_k \sin(kx)$$

such that least sq error is min

$$\min \int_{-\pi}^{\pi} (f(x) - S_n(x))^2 \, dx$$

equating partial derivatives to 0,

$$a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(kx) \, dx$$

$$b_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(kx) \, dx$$

Discrete Fourier Transform

We have $2m$ data points x_k, y_k where

$x_k = -\pi + \frac{k\pi}{m}$ and $y_k = f(x_k)$ $k \in \{0, 1, \dots, 2m-1\}$

It does the foll:

$$\min \sum_{k=0}^{2m-1} (S_n(x_k) - y_k)^2$$

Lemma

1) If r is not a multiple of $2m$

$$\sum_{k=0}^{2m-1} \cos(r x_k) = \sum_{k=0}^{2m-1} \sin(r x_k) = 0$$

2) If $r \neq 0$ is a multiple of m

$$\sum_{k=0}^{2m-1} \cos^2(r x_k) = \sum_{k=0}^{2m-1} \sin^2(r x_k) = m$$

3) If $r \neq l$ and $r+l$ is not a multiple of $2m$,

$$\sum_{k=0}^{2m-1} \cos(r x_k) \cos(l x_k) = \sum_{k=0}^{2m-1} \sin(r x_k) \sin(l x_k) = 0$$

$$\sum_{k=0}^{2m-1} \cos(r x_k) \sin(l x_k) = \sum_{k=0}^{2m-1} \sin(r x_k) \cos(l x_k) = 0$$

Then for any $n < m$ the best approx is

$$S_n(x) = \frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos(kx) + \sum_{k=1}^{n} b_k \sin(kx)$$

Due to prev 3 lemmas

$$a_k = \frac{1}{m} \sum_{j=0}^{2m-1} y_j \cos(k x_j) \qquad b_k = \frac{1}{m} \sum_{j=0}^{2m-1} y_j \sin(k x_j)$$

choose $n = m-1$

$$\{y_j\}_{j=0}^{2m-1} \xrightarrow{DFT} \{(a_k, b_k)\}_{k=0}^{m-1}$$

Issue with $n \geq m$: we may not have sufficient eqns to solve for all diff values of a_k and b_k since every value of m gave one pair of vals for a_k, b_k.

Fast Fourier Transform (FFT)

FFT computes these coefficients in $O(m \log_2(m))$. This is fast compared to $O(m^2)$ (naive method).