$\boxed{L\#18}$

Oct 27 & Nov 1
2011

Trigonometric Interpolation

In many applications we need to recover periodic function from its data, say $f(t+T)=f(t)$, $T>0$ is the period.

Def: for $n\in\mathbb{N}$ integer T_n is the set of trig polynomials

$$p(t)=\frac{a_0}{2}+\sum_{k=1}^{n}(a_k\cos kt+b_k\sin kt) \qquad T=2\pi$$

with coefficients a_j, b_j

Theorem: Given $2n+1$ distinct points in the interval $[0,2\pi)$ and $2n+1$ values $y_0,\dots,y_{2n}\in\mathbb{R}$ there is unique polynomial $q_n\in T_n$ that interpolates the data, i.e.

$$q_n(t_j)=y_j \quad j=0,\dots,2n$$

Explicit formula is

$$q_n(t)=\sum y_k \ell_k(t) \qquad \ell_k(t)=\prod_{\substack{i=0\\ i\neq k}}^{2n}\frac{\sin\frac{t-t_k}{2}}{\sin\frac{t_k-t_i}{2}}$$

Complex setting is much better.

Complex Setting of Trigonometric Interpolation

Define $E_n(x) = e^{ixn} = (e^{ix})^n$ trig poly of degree n

$$e^{ix} = \cos x + i\sin x$$

Define the set

$$T_N \equiv \{E_0(x), E_1(x), \dots, E_{N-1}(x)\} \text{ on } [0, 2\pi)$$

E_0 real, complex

Define the interpolation points (equidistant)

$$x_j = t_j = \frac{2\pi}{N} j \qquad j = 0, 1, \dots, N-1$$

<u>Interpolation Problem</u>: Find $p(x) \in T_N$,

(1) $p(x) = \sum_{k=0}^{N-1} c_k E_k(x)$, $c_k \in \mathbb{C}$ - complex numbers

such that

(2) $$p\left(\frac{2\pi}{N} j\right) = f\left(\frac{2\pi}{N} j\right) \qquad j = 0, \dots, N-1$$

Now we shall derive an explicit formula for the coefficients c_k of the interpolant (1) from the interpolation conditions (2).

Some simple facts. Consider $L^2(0,2\pi)$

$$\langle f,g\rangle = \frac{1}{2\pi}\int_0^{2\pi} f(x)\,\overline{g}(x)\,dx$$

$$\|f\| = \|f\|_{L^2} = \left(\frac{1}{2\pi}\int_0^{2\pi} |f(x)|^2 dx\right)^{1/2}$$

One can easily see that

$$\langle E_n, E_n\rangle = \delta_{mn} = \begin{cases} 1 & n=m \\ 0 & n\neq m\end{cases}$$

Next, we define a discrete inner product

$$\langle f,g\rangle_N = \frac{1}{N}\sum_{j=0}^{N-1} f(x_j)\,\overline{g}(x_j)$$

and check the properties

(a) $\langle f,f\rangle_N \geq 0$

(b) $\langle f,g\rangle_N = \overline{\langle g,f\rangle_N}$ $\Big\}$ pseudo-inner product

One can easily see that $\langle E_n, E_n\rangle_N = \delta_{nm}$. Now introduce the matrix

$$W = \begin{vmatrix} \vdots & \vdots & & \vdots \\ E_0 & E_1(x_j) & \cdots & E_{N-1}(x_j) \\ \vdots & \vdots & & \vdots \end{vmatrix} \text{ j-th row of } W$$

$W \in \mathbb{C}^{N\times N}$

if $\lambda = e^{\frac{2\pi i}{N}}$ $\quad w_{jk} = \lambda^{jk}$

Because of the property $\langle E_n, E_m \rangle_N = 0$ we have

$$\overline{W}^T W = N I \qquad I\text{-identity matrix in } \mathbb{C}^{N\times N}$$

Now, if $c = \begin{vmatrix} c_0 \\ \vdots \\ c_{N-1} \end{vmatrix}$ & $f = \begin{vmatrix} f_0 \\ \vdots \\ f_{N-1} \end{vmatrix} = \begin{vmatrix} \vdots \\ f(\frac{2\pi}{N}j) \\ \vdots \end{vmatrix}$

Then the interpolation problem (1), (2) becomes

$$p(x_j) = f(x_j) \qquad \sum_{k=0}^{N-1} c_k E_k(x_j) = f(x_j) \quad j = 0, 1, \dots, N-1$$

could be written in matrix form

(4) $$W c = f$$

Now multiply (4) by $\overline{W}^T$ from left to get

$$\underbrace{\overline{W}^T W}_{NI} c = \overline{W}^T f \implies \boxed{c = \frac{1}{N} \overline{W}^T f}$$

$$\begin{array}{ll} & \overline{W}^T = \begin{vmatrix} \text{---} & \overline{E_0}(x_j) & \text{---} \\ \text{---} & \overline{E_1}(x_j) & \text{---} \\ \text{---} & \overline{E_k}(x_j) & \text{---} \\ \text{---} & \overline{E_{N-1}}(x_j) & \text{---} \end{vmatrix} \\ k\text{-th row} & \end{array} \qquad c_k = (\overline{W}^T f)_k = \frac{1}{N} \sum_{j=0}^{N-1} \overline{E_k}(x_j) f(x_j)$$

$$\equiv \langle f, E_k \rangle_N$$

Thus we get the formula

$$\boxed{(5) \quad p(x) = \sum_{k=0}^{N-1} c_k E_k(x), \quad c_k = \langle f, E_k \rangle_N}$$

Nov 1, 2011

L# 18

Trigonometric Interpolation (continue, FFT)

The problem find $p(x) = \sum_{k=0}^{N-1} c_k E_k(x)$, $E_k(x) = e^{ixk}$

s.t. $p(x_j) = f(x_j) \equiv f_j$ $x_j = \frac{2\pi}{N} j$, $j = 0, \dots, N-1$ $f = \begin{pmatrix} f_0 \\ \vdots \\ f_{N-1} \end{pmatrix}$ $c = \begin{pmatrix} c_0 \\ \vdots \\ c_{N-1} \end{pmatrix}$

has unique solution which is expressed in the following closed form

$$C = \frac{1}{N} \overline{W}^T f \quad \text{or equiv} \quad c_k = \frac{1}{N} \sum$$

or equivalently

$$\boxed{\begin{aligned} c_k &= \frac{1}{N} \sum_{j=0}^{N-1} f(x_j) \overline{E}_k(x_j) \equiv \langle f, E_k \rangle_N \\ p(x) &= \sum_{k=0}^{N-1} c_k E_k(x) \quad \text{- Fourier transform} \end{aligned}}$$

Straightforward implementation of these formula will produce the coefficients c_k, $k = 0, \dots, N-1$ for N^2 long arithmetic operations.

FFT

Nov 1, 2011

From the formula (5) we see that if we precompute the matrix W we find each coefficient c_k by just N multiplications and N-additions So for all c_k we need N^2 multiplications.

However there is a clever implementation of this process that needs only $N\log_2 N$ mult/div.
Let us have a comparison

N	N^2	$N\log_2 N$
$1024 = 2^{10}$	$\sim 10^6$	$\sim 10^4$
$16384 = 2^{14}$	$\sim 2.7\,10^8$	$\sim 2.3\,10^5$

How this is done? It is done by recursion
Consider $N = 2n$ (in general, $N = 2^m$)

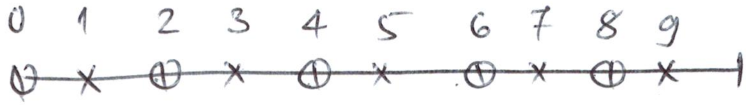
<u>Theorem:</u> Let $p(x)$ & $q(x)$ be trigonometric polynomials of degree $\leq n-1$ s.t. for $x_j = \frac{2\pi j}{2n}$ $j = 0, 1, \dots, 2n-1$ interpolate the data $f(x_j)$ $j=0,1,\dots,2n$ in the following way

$$p(x_{2j}) = f(x_{2j}) \quad q(x_{2j}) = f(x_{2j+1}) \quad j = 0, \dots, n-1$$

i.e. $p(x)$ interpolates the data with even index $q(x)$ interpolates the data with odd index

$n=5 \quad 2n-1=9$ last index

Ex



$\circ - p(x)$
$\times - q(x)$

Then the polynomial $P(x)$ of degree $2n-1$ that interpolates f at $x_0, x_1, \dots, x_{2n-1}$ is

$$P(x) = \frac{1}{2}\left(1 + e^{inx}\right)p(x) + \frac{1}{2}\left(1 - e^{+inx}\right)q\left(x - \frac{\pi}{n}\right)$$

We just need to check whether $p(x_j) = f(x_j)$ $j=0, \dots, 2n-1$

Check First consider even points $x_{2j} = \frac{2\pi j 2}{2n} = \frac{2\pi j}{n}$

$$e^{inx_{2j}} = e^{i\frac{n 2\pi j}{n}} = e^{2\pi j i} = 1$$

$$P(x_{2j}) = \frac{1}{2}(1+1)\,p(x_{2j}) = p(x_{2j}) = f(x_{2j}) \quad \text{OK}$$

Next consider odd points $x_{2j+1} = \frac{2\pi(2j+1)}{2n} = \frac{2\pi j}{n} + \frac{\pi}{n}$

$$e^{inx_{2j+1}} = e^{i(2\pi j + \pi)} = e^{i\pi} = -1$$

$$P(x_{2j+1}) = \frac{1}{2}(1+1)\,q(x_{2j}) = f(x_{2j+1}) \text{ by construct}$$

Thus the polynomial is computed using two polynomials of degree twice smaller. This gives the general idea for the computations of the interpolation coefficients.

Theorem: Let $p(x)=\sum_{j=0}^{n-1}\alpha_j E_j(x)$, $q(x)=\sum_{j=0}^{n-1}\beta_j E_j(x)$ and $P(x)=\sum_{j=0}^{2n-1}\gamma_j E_j(x)$. If α_j & β_j are available then γ_j are computed by

$$(1)\quad \boxed{\begin{aligned} \gamma_j &= \tfrac{1}{2}\alpha_j + \tfrac{1}{2}e^{-i\frac{\pi}{n}}\beta_j \\ \gamma_{j+n} &= \tfrac{1}{2}\alpha_j - \tfrac{1}{2}e^{-i\frac{\pi}{n}}\beta_j \end{aligned} \qquad j=0,1,\dots,n-1}$$

Proof: Indeed since $E_n(x)E_j(x)=E_{n+j}(x)$ we

$$e^{inx}e^{ijx}=e^{i(n+j)x}$$

rewrite the above formula

$$P(x)=\frac{1}{2}(1+E_n(x))\sum_{j=0}^{n-1}\alpha_j E_j(x)+\frac{1}{2}(1-E_n(x))\sum_{j=0}^{n-1}\beta_j E_j(x)e^{-\frac{\pi}{n}i}$$

$$P(x)=\frac{1}{2}\sum_{j=0}^{n-1}\left(\alpha_j E_j(x)+\beta_j e^{-\frac{\pi i}{n}}E_j(x)\right)$$

$$+\frac{1}{2}\sum_{j=0}^{n-1}\left(\alpha_j E_{n+j}(x)-\beta_j e^{-\frac{\pi}{n}i}E_{n+j}(x)\right)=\sum_{j=0}^{2n-1}\gamma_j E_j(x)$$

By comparing the coefficients in front of E_j we get the desired formulas (1).

Now how you work with these and what is the operation count?

First, you precompute the numbers $\frac{1}{2}e^{-i\frac{\pi}{n}} = c$

Then you see that to find the coefficients γ_j you need to

① multiply $\alpha_j * \frac{1}{2}$ $j=0,\dots,n-1$ n

② multiply $\beta_j * c$ $j=0,\dots,n-1$ n

total $2n$ long operations

Let us denote by $R(2n)$ the cost of computing the coefficients of trig polynomial of degree $2n$. Then obviously the count of long arithmetic operation is

$$R(2n) = 2R(n) + 2n$$

the cost of computing two polynomials p & q of degree n

cost of formula (1)

Now take $N = 2^m$ $m = \log_2 N$

$$R(2^m) = 2R(2^{m-1}) + 2^m =$$

$$= 2\left[2R(2^{m-2}) + 2^{m-1}\right] + 2^m = 2^2 R(2^{m-2}) + 2 \cdot 2^m$$

$$\dots = 2^m R(1) + m \cdot 2^m \approx m 2^m \approx \boxed{N \log_2 N}$$