Approssimazione trigonometrica, interpolazione trigonometrica e FFT

30 gennaio 2009

1 Approssimazione con funzioni trigonometriche

Consideriamo una funzione periodica $f \in L^2([0,2\pi])$, $f : \mathbb{R} \to \mathbb{C}$, e cerchiamo la miglior approssimazione del tipo

$$f(x) \approx s_n(x) := \sum_{j=-n}^{n} c_j \exp(+i\,j\,x) \tag{1}$$

dove come al solito i è la costante immaginaria. Osserviamo fin d'ora che in alcuni manuali, si studia il problema di cercare un'approssimante del tipo

$$s_n(x) := \sum_{j=-n}^{n} c_j \exp(-i\,j\,x)$$

il che corrisponde a un ordine diverso dei coefficienti c_j. I coefficienti c_j sono detti di *Fourier*.
Di seguito poniamo

$$(f,g)_2 = \int_0^{2\pi} f(x)\,\overline{g(x)}\,dx \tag{2}$$

dove $\overline{g(x)}$ è il coniugio di $g(x)$.
L'insieme $L^2_\pi([0,2\pi])$ delle funzioni di $L^2([0,2\pi])$ periodiche una volta dotato del prodotto interno $(\cdot,\cdot)_2$ è uno spazio euclideo, e

$$\mathcal{T}_n = \operatorname{span}(\exp(-i\,n\,x),\ldots,\exp(+i\,n\,x))$$

è in effetti un suo sottospazio vettoriale di dimensione finita, in cui

$$\exp(-i\,n\,x),\ldots,\exp(+i\,n\,x)$$

è in particolare una sua base ortogonale. Vediamo perchè .

Che la funzione s_n sia periodica è conseguenza dell'uguaglianza di Eulero

$$\exp(i\,t) = \cos(t) + i\sin(t),\ t \in \mathbb{R} \tag{3}$$

Infatti

$$\begin{aligned} \exp(i\,k(x+2\pi)) &= \cos(k\,x+2\,k\pi)+i\sin(k\,x+2\,k\pi) \\ &= \cos(k\,x)+i\sin(k\,x) \end{aligned} \tag{4}$$

e la somma finita di funzioni periodiche è periodica. Quindi in effetti $\mathcal{T}_n$ è un sottoinsieme di $L^2_\pi([0,2\pi])$. Senza molte difficoltà si vede che in effetti è un sottospazio vettoriale di $L^2_\pi([0,2\pi])$.
Passiamo a vedere la proprietà di ortogonalità tra elementi di $\mathcal{T}_n$ del tipo $\exp(+i\,j\,\cdot)$ con $j=-n,\ldots,n$. Osserviamo che

$$(\exp(i\,j\,\cdot),\exp(i\,k\,\cdot))_2=\int_0^{2\pi}\exp(i(j-k)x)\,dx$$

e che se $j=k$

$$\int_0^{2\pi}\exp(i(j-k)x)\,dx=\int_0^{2\pi}1\,dx=2\pi$$

altrimenti,

$$\begin{aligned} \int_0^{2\pi}\exp(i(j-k)x)\,dx &= \int_0^{2\pi}\cos((j-k)x)\,dx+i\int_0^{2\pi}\sin((j-k)x)\,dx \\ &= \frac{1}{(j-k)}\int_0^{2\pi(j-k)}\cos(t)\,dt+\frac{i}{(j-k)}\int_0^{2\pi(j-k)}\sin(t)\,dt \\ &= 0 \end{aligned} \tag{5}$$

Poniamo $\phi_1(x):=\exp(-inx),\ldots,\phi_{2n+1}(x):=\exp(inx)$, $N=2n+1$. Il teorema di miglior approssimazione in spazi euclidei, stabilisce che l'elemento di miglior approssimazione di $f\in L^2_\pi([0,2\pi])$ è $s^*_N=\sum_{k=1}^N\tilde{c}_k\phi_k(x)$
dove

$$\tilde{c}_j=\frac{(f,\phi_j)}{(\phi_j,\phi_j)},\ \ j=1,\ldots,N.$$

Di conseguenza essendo

$$(\phi_j,\phi_j)_2=2\pi,\ \ j=1,\ldots,N.$$

si ha

$$c_k=\tilde{c}_{k+n+1}=\frac{(f,\phi_{k+n+1})}{(\phi_{k+n+1},\phi_{k+n+1})}=\frac{1}{2\pi}(f,\exp(+i\,k\,\cdot))_2,\ k=-n,\ldots,n. \tag{6}$$

Ora integriamo numericamente

$$(f,\exp(+i\,k\,\cdot))_2=\int_0^{2\pi}f(x)\exp(-i\,k\,\cdot)\,dx$$

mediante la formula dei trapezi

$$\int_0^{2\pi} g(x)\,dx \approx T_L(g) := \frac{h}{2}g(t_0) + h\sum_{j=1}^{L-1} g(t_j) + \frac{h}{2}g(t_L),\ h = \frac{2\pi}{L},\ t_j = j\,h.$$

Dalla periodicità di f, essendo $t_0 = 0$, $t_L = 2\pi$, abbiamo che

$$T_L(f) := h\sum_{j=0}^{L-1} f(t_j) = h\sum_{j=1}^{L} f(t_j)$$

e quindi per $k = -n, \ldots, n$

$$\begin{aligned} c_k &= \frac{1}{2\pi}(f, \exp(i\,k\,\cdot))_2 \approx \frac{1}{2\pi}T_L(f\,\exp(-i\,k\,\cdot)) \\ &= \frac{1}{2\pi}\frac{2\pi}{L}\sum_{j=0}^{L-1} f(t_j)\exp(-i\,k\,t_j) \\ &= \frac{1}{L}\sum_{j=0}^{L-1} f(t_j)\exp\left(-i\,k\,j\,\frac{2\pi}{L}\right) \\ &= \frac{1}{L}\sum_{s=1}^{L} f(t_{s-1})\exp\left(-i\,k\,(s-1)\,\frac{2\pi}{L}\right) \end{aligned} \tag{7}$$

In generale tutti i coefficienti c_k sono calcolati tramite un famoso algoritmo, la FFT (cioè la *Fast Fourier Transform*) (cf. [1, p.181], [4, p.277], [9, p.398]).

1.1 Facoltativo: Velocità di convergenza della formula dei trapezi composta per funzioni periodiche

Consideriamo la formula dei trapezi composta

$$T_n(g) := \frac{h}{2}g(x_0) + h\sum_{j=1}^{L-1} g(x_j) + \frac{h}{2}g(x_n),\ h = \frac{b-a}{n},\ x_j = a + j\,h$$

che nel caso di funzioni periodiche, cioè tali che $g(x_0) = g(x_n)$ assume la forma

$$T_n(g) := h\sum_{j=0}^{n-1} g(x_j),\ h = \frac{b-a}{n},\ x_j = a + j\,h.$$

Ci si chiede se

$$I(g) = \int_a^b$$

quanto valga l'errore $E_n(g) = I(g) - T_n(g)$. Si dimostra che (cf. [1, p.253])

Teorema 1.1 *Se $g \in C^{(2)}([a,b])$ allora per un certo $\eta \in (a,b)$ si ha*

$$E_n(g) = \frac{-(b-a)\cdot h^2 \cdot g^{(2)}(\eta)}{12}$$

Questa stima mostra che la velocità di convergenza è quadratica rispetto al passo h e quindi che se il passo si dimezza e $g^{(2)}$ è circa costante allora $E_{2n}(g) \approx \frac{E_n(g)}{4}$.

Nel caso di funzioni periodiche le cose possono cambiare radicalmente (cf. [1, p.285]). La formula di *Eulero-Maclaurin* stabilisce che

Teorema 1.2 *Si supponga $m \geq 0$, $n \geq 1$ e sia $h = (b-a)/n$, $x_j = a + jh$ per $j = 0, 1, \ldots, n$. Si assuma inoltre $g \in C^{(2m+2)}([a,b])$ per qualche $m \geq 0$. Allora*

$$\begin{aligned} E_n(g) &= -\sum_{k=1}^{m} \frac{B_{2k} h^{2k}}{2k!} \left(g^{(2k-1)}(b) - g^{(2k-1)}(a) \right) \\ &+ \frac{h^{2m+2}}{(2m+2)!} \int_a^b \overline{B}_{2m+2}\left(\frac{x-a}{h}\right) g^{(2m+2)}(x)\, dx \end{aligned} \tag{8}$$

dove i numeri di Bernoulli sono definiti implicitamente dalla formula

$$\frac{t}{\exp(t)-1} = \sum_{j=0}^{\infty} B_j \frac{t^j}{j!}$$

e $\overline{B}_j(x)$ sono l'estensione periodica dei polinomi di Bernoulli grado j, diciamo $B_j(x)$, ottenuti tramite la funzione generatrice

$$\frac{t(\exp(xt)-1)}{\exp(t)-1} = \sum_{j=1}^{\infty} B_j(x) \frac{t^j}{j!}.$$

Per un asserto meno complicato per quanto riguarda l'ultimo termine, ma sotto ipotesi più restrittive, si consideri [**?**, p.990].

A parte la complicatezza dell'asserto, se $g^{(2k-1)}(b) = g^{(2k-1)}(a)$ per $k = 1, \ldots, m$, allora si ha

$$E_n(g) = \frac{h^{2m+2}}{(2m+2)!} \int_a^b \overline{B}_{2m+2}\left(\frac{x-a}{h}\right) g^{(2m+2)}(x)\, dx$$

Se il termine $\int_a^b \overline{B}_{2m+2}\left(\frac{x-a}{h}\right) g^{(2m+2)}(x)\, dx$ è circa costante allora dimezzandosi il passo h allora $E_{2n}(g) \approx \frac{E_n(g)}{2^{2m+2}}$ con una riduzione dell'errore molto più elevata di quella prevista dal teorema generale (1.1).

Osserviamo ora che nel teorema relativo al calcolo dei coefficienti del polinomio trigonometrico di miglior approssimazione (in norma $\|g\|^2 = \int_0^{2\pi} |g(x)|^2 dx$) dobbiamo calcolare per $f : [0, 2\pi] \to \mathbb{R}$ l'integrale

$$\int_0^{2\pi} f(x) \exp(-i\,k\,\cdot)\, dx = \int_0^{2\pi} f(x) \cos(k\,x)\, dx - i \int_0^{2\pi} f(x) \sin(k\,x)\, dx$$

il che comporta che la complessità è essenzialmente dovuta al calcolo dei due integrali

$$\int_0^{2\pi} f(x) \cos(k\,x)\, dx,$$

$$\int_0^{2\pi} f(x) \sin(k\,x)\, dx.$$

Il teorema precedente stabilisce, in virtù della periodicità delle integrande, la velocità di convergenza della formula dei trapezi composta per il calcolo dell'integrale $\int_0^{2\pi} f(x)\exp(-i\,k\cdot)\,dx$. Interessante il fatto che essendo periodiche per $k \in \mathbb{Z}$ tutte le derivate di $\cos(k\,x)$ e $\sin(k\,x)$ allora dalla formula di derivazione di un prodotto di funzioni si deduce facilmente che se $f^{(2k-1)}(a) = f^{(2k-1)}(b)$ per $k = 1,\ldots,m$ allora l'errore compiuto è dell'ordine di h^{2m+2}.

2 Interpolazione trigonometrica

Cominciamo stabilendo il problema dell'interpolazione trigonometrica ed un risultato di esistenza e unicità .

Teorema 2.1 *Sia f una funzione continua e periodica nell'intervallo $[0,2\pi]$ e si pongano*

$$t_j = j\cdot\frac{2\pi}{2n+1},\ j = 0,\ldots,2n.$$

Il problema dell'interpolazione trigonometrica, consistente nel calcolare i coefficienti $\tilde{c}_k$, $k = -n,\ldots,n$ tali che

$$\sum_{k=-n}^{n} \tilde{c}_k \exp(+i\,k\,t_j) = f(t_j),\ j = 0,\ldots,2n, \tag{9}$$

ha una e una sola soluzione $\tilde{c} = (\tilde{c}_k)_{k=-n,\ldots,n}$.

Dimostrazione facoltativa. Dimostriamo che effettivamente il polinomio trigonometrico

$$p_n(x) = \sum_{k=-n}^{n} c_k \exp(i\,k\,x)$$

interpolante le coppie $(t_k, f(t_k))$ con $k = 0.\ldots,2n$ esiste ed è unico. Vediamo di riscrivere questo problema di interpolazione trigonometrica come un certo problema di interpolazione polinomiale in campo complesso. Posto $s = k+n$, e notato che

$$(\exp(i\,t_j))^{-n} \neq 0,\ j = 0,\ldots,2n$$

si ha

$$\begin{aligned} f(t_j) &= \sum_{k=-n}^{n} c_k \exp(i\,k\,t_j), \\ &= \sum_{k=-n}^{n} c_k (\exp(i\,t_j))^k, \\ &= \sum_{s=0}^{2n} c_{s-n} (\exp(i\,t_j))^{s-n}, \\ &= (\exp(i\,t_j))^{-n} \sum_{s=0}^{2n} c_{s-n} (\exp(i\,t_j))^s, \end{aligned} \tag{10}$$

e quindi il problema trigonometrico è ricondotto a quello di calcolare il polinomio algebrico

$$q_{2n}(z) = \sum_{s=0}^{2n} c_{s-n} z^s,\ z \in \mathbb{C}$$

di grado $2n$ (in campo complesso!) tale che posto $z_j = i\,t_j$ per $j = 0\ldots,2n$ si abbia

$$q_{2n}(z_j) = f(t_j)\,(\exp(i\,t_j))^{+n} = f(t_j)\,(\exp(i\,n\,t_j)).$$

Che il polinomio algebrico interpolatore $q_{2n} : \mathbb{C} \to \mathbb{C}$ esista unico è una ovvia estensione di quanto visto nell'interpolazione con polinomi algebrici in $\mathbb{R}$, in quanto il determinante della matrice di Vandermonde (le cui componenti sono numeri complessi)

$$V = \begin{bmatrix} 1 & t_0 & t_0^2 & \cdots & t_0^n \\ 1 & t_1 & t_1^2 & \cdots & t_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & t_{2n} & t_{2n}^2 & \cdots & t_{2n}^n \end{bmatrix}$$

è

$$\prod_{0 \le l < m \le 2n} (t_m - t_l),$$

evidentemente non nullo in quanto $t_m \neq t_l$ se $m \neq l$ (cf. [8]).

Vista l'equivalenza dei due problemi e l'esistenza e unicità di un polinomio in $\mathbb{C}$ di grado $2n$ interpolante le coppie $(t_k, f(t_k))$, deduciamo che pure il polinomio trigonometrico $p_n(x) = \sum_{k=-n}^{n} c_k \exp(ikx)$ tale che $p_n(t_k) = f(t_k)$ esiste ed è unico. ∎

Mostriamo ora che

Teorema 2.2 *I coefficienti $\tilde{c}_k$, $k = -n,\ldots,+n$ del polinomio trigonometrico*

$$\sum_{k=-n}^{n} \tilde{c}_k \exp(+i\,k\,t) \tag{11}$$

interpolante le coppie $(t_j = j \cdot \frac{2\pi}{2n+1},\ j = 0,\ldots,2n, f(t_j))$, sono uguali a

$$\begin{aligned} \tilde{c}_k &= \frac{1}{2n+1} \cdot \sum_{s=0}^{2n} f(t_s) \exp(-i\,k\,t_s) \\ &= \frac{1}{2n+1} \cdot \sum_{l=1}^{2n+1} f(t_{l-1}) \exp(-i\,k\,t_{l-1}) \\ &= \frac{1}{2n+1} \cdot \sum_{l=1}^{2n+1} f((l-1)\frac{2\pi}{2n+1}) \exp(-i\,k\,(l-1)\frac{2\pi}{2n+1}) \end{aligned} \tag{12}$$

con $k = -n,\ldots,n$.

2.1 Una dimostrazione sul calcolo dei c_k

Osserviamo che

$$\sum_{j=0}^{2n} \exp(+i(k-l)t_j) = \begin{cases} 0 & k \neq l \\ 2n+1 & k = l \end{cases}$$

Il caso in cui $k = l$ è ovvio e quindi poniamo attenzione al caso in cui $k \neq l$. Si vede subito che posto $w = \exp(+i(k-l)\cdot\frac{2\pi}{2n+1})$, essendo $\exp(i\,s2\pi) = 1$ per un intero s,

$$\begin{aligned} \sum_{j=0}^{2n} \exp(+i(k-l)t_j) &= \sum_{j=0}^{2n} \exp(+i(k-l)j\cdot\frac{2\pi}{2n+1}) \\ &= \sum_{j=0}^{2n} w^j = (w^{2n+1}-1)/(w-1) \\ &= \left(\exp\left(i(k-l)\cdot\frac{2\pi}{2n+1}\right)^{2n+1} - 1\right)/(w-1) \\ &= \left(\exp(i(k-l)\cdot 2\pi) - 1\right)/(w-1) \\ &= 0/(w-1) = 0 \end{aligned} \tag{13}$$

Moltiplicando ambo i membri di (17) per $\exp(-i\,l\,t_j)$ e sommando in j, otteniamo

$$\begin{aligned} \sum_{j=0}^{2n} f(t_j)\exp(-i\,l\,t_j) &= \sum_{j=0}^{2n} \exp(-i\,l\,t_j) \sum_{k=-n}^{n} \tilde{c}_k \exp(+i\,k\,t_j) \\ &= \sum_{k=-n}^{n} \tilde{c}_k \sum_{j=0}^{2n} \exp(i(k-l)t_j) \\ &= (2n+1)\,\tilde{c}_l \end{aligned} \tag{14}$$

da cui

$$\tilde{c}_l = \frac{1}{2n+1}\sum_{j=0}^{2n} f(t_j)\exp(-i\,l\,t_j).$$

Una prima lettura non banale consiste nel fatto che i coefficienti c_k sono noti esplicitamente, senza risolvere numericamente il sistema lineare del tipo $Vc = f$ dove

$$V_{jk} = \exp(i\,k\,t_j),\ f_j = f(t_j).$$

Osserviamo ora che in (7), per $k = -n,\ldots,n$ avevamo approssimato numericamente c_k con la formula dei trapezi fornendo

$$c_k \approx \frac{1}{N}\sum_{s=1}^{N} f(t_{s-1})\exp\left(-i\,k\,(s-1)\,\frac{2\pi}{N}\right) \tag{15}$$

Nel caso speciale in cui $N = 2n+1$, posto $j = s-1$, abbiamo essendo $t_j = j\cdot\frac{2\pi}{2n+1}$

$$c_k \approx \frac{1}{2n+1}\sum_{j=0}^{2n} f(t_j)\exp\left(-i\,k\,j\,\frac{2\pi}{2n+1}\right) = \tilde{c}_k,\ s = k = -n,\ldots,n. \tag{16}$$

cioè l'interpolante polinomiale ha quali coefficienti $\tilde{c}_{k=-N}^{N}$ quelli ottenuti calcolando numericamente i coefficienti di Fourier c_k con una formula dei trapezi avente $2N+1$ punti.

2.2 Una seconda dimostrazione sul calcolo dei c_k (più difficile della prima)

Se $f \in C([0,2\pi])$ è una funzione periodica e se

$$t_j = j \cdot \frac{2\pi}{2n+1},\ j = 0,\dots,2n,$$

da quanto detto precedentemente il problema dell'interpolazione trigonometrica, consistente nel calcolare i coefficienti $\tilde{c}_k$, $k = -n,\dots,n$ tali che

$$\sum_{k=-n}^{n} \tilde{c}_k \exp(+ik t_j) = f(t_j),\ j = 0,\dots,2n. \tag{17}$$

ha una e una sola soluzione $(\tilde{c}_k)_{k=-n,\dots,n}$. Sia

$$\mathbf{w}^{(k)} = (w_j^{(k)})_{j=0,\dots,2n} \in \mathbb{C}^{2n+1}$$

dove

$$w_j^{(k)} = \exp(i\,k\,t_j).$$

Allora il problema iniziale si può riscrivere come quello di determinare un vettore $\tilde{\mathbf{c}} = (\tilde{c}_k)_{k=-n,\dots,n} \in \mathbb{R}^{2n+1}$ per cui

$$\sum_{k=-n}^{n} \tilde{c}_k w_j^{(k)} - f(t_j) = 0,\ j = 0,\dots,2n. \tag{18}$$

Dotato $\mathbb{C}^{2n+1}$ dell'usuale prodotto scalare

$$(\mathbf{u},\mathbf{v}) = \sum_{s=0}^{2n} u_s \cdot \overline{v_s},\ \mathbf{u} = (u_s)_{s=0,\dots,2n},\ \mathbf{v} = (v_s)_{s=0,\dots,2n}$$

e definita la norma

$$\|\mathbf{u}\|^2 := (u,u)$$

se $\tilde{\mathbf{c}} = (\tilde{c}_k)_k$ risolve (18), allora automaticamente è l'unica soluzione del problema di miglior approssimazione

$$\min_{\mathbf{c}\in\mathbb{C}^{2n+1}} \left\| \sum_{k=-n}^{n} c_k w_j^{(k)} - f(t_j) \right\| = \min_{\mathbf{c}\in\mathbb{C}^{2n+1}} \left\| \sum_{k=-n}^{n} c_k \exp(i\,k\,t_j) - f(t_j) \right\| \tag{19}$$

in quanto da (17)

$$\left\| \left(\sum_{k=-n}^{n} \tilde{c}_k w_j^{(k)} - f(t_j) \right)_{j=0,\dots,2n} \right\| = 0. \tag{20}$$

Di conseguenza, invece di studiare il problema di interpolazione (10) posso considerare l'equivalente problema di miglior approssimazione (20). I vettori

$$\mathbf{w}^{(-n)},\dots,\mathbf{w}^{(n)}$$

formano un sistema ortogonale. Infatti essendo

$$\overline{\exp(i\,s\,x)} = \overline{\cos(s\,x) + i\sin(s\,x)} = \cos(s\,x) - i\sin(s\,x) = \cos(-s\,x) + i\sin(-s\,x) = \exp(-i\,s\,x)$$

abbiamo per $l, k = -n, \ldots, n$

$$\begin{aligned}
(\mathbf{w}^{(l)}, \mathbf{w}^{(k)}) &= \sum_{s=0}^{2n} \mathbf{w}_s^{(l)} \cdot \overline{\mathbf{w}}^{(k)}{}_s \\
&= \sum_{s=0}^{2n} \mathbf{w}^{(l)} \cdot \overline{\mathbf{w}}^{(k)} \\
&= \sum_{s=0}^{2n} \exp(i\,j\,t_s) \cdot \overline{\exp(i\,k\,t_s)} \\
&= \sum_{s=0}^{2n} \exp(i\,(j-k)\,t_s) \qquad (21)
\end{aligned}$$

Se $l = k$ allora

$$(\mathbf{w}^{(l)}, \mathbf{w}^{(k)}) = \sum_{s=0}^{2n} \exp(i\,(l-k)\,t_s) = \sum_{s=0}^{2n} \exp(i \cdot 0 \cdot t_s) = 2n+1$$

altrimenti posto

$$\xi = \exp\left(+i(k-l) \cdot \frac{2\pi}{2n+1}\right)$$

ed essendo $\exp(i\,s\,2\pi) = 1$ per un qualsiasi numero intero s,

$$\begin{aligned}
(\mathbf{w}^{(l)}, \mathbf{w}^{(k)}) &= \sum_{j=0}^{2n} \exp(+i(k-l)\,t_j) \\
&= \sum_{j=0}^{2n} \exp(+i(k-l)\,j \cdot \frac{2\pi}{2n+1}) \\
&= \sum_{j=0}^{2n} \xi^j = (\xi^{2n+1} - 1)/(\xi - 1) \\
&= \left(\exp\left(i(k-l) \cdot \frac{2\pi}{2n+1}\right)^{2n+1} - 1\right)/(\xi - 1) \\
&= \left(\exp(i(k-l) \cdot 2\pi) - 1\right)/(\xi - 1) \\
&= 0/(\xi - 1) = 0 \qquad (22)
\end{aligned}$$

Di conseguenza, essendo $\mathbb{C}^{2n+1}$ uno spazio euclideo, dal teorema di miglior approssimazione in spazi euclidei, posto $\mathbf{f} = (f(t_s))_s$, per $k = -n, \ldots, n$

$$\tilde{c}_k = c_k^* = \frac{(\mathbf{f}, \mathbf{w}^{(k)})}{(\mathbf{w}^{(k)}, \mathbf{w}^{(k)})} = \frac{1}{2n+1} \cdot \sum_{s=0}^{2n} f(t_s) \exp(-i\,k\,t_s)$$

3 FFT

Il problema consiste nel calcolare i coefficienti di Fourier $\{c_j\}_{j=0,\ldots,N-1}$ per una funzione $f \in L^2_\pi([0,2\pi])$ i cui valori sono noti nei punti $\frac{2\pi\beta}{N}$ con $\beta = 0,1,\ldots,N-1$. Per prima cosa vediamo che ci sono più formulazioni in qualche modo equivalenti.

3.1 Varie formulazioni equivalenti dei coefficienti di interpolazione

Ci si domanda quale sia il legame di una formulazione del tipo

$$\mathscr{X}(\tau) = \sum_{n=1}^{N} x(n) \exp\left(\frac{-i\,2\,\pi\,(\tau-1)\,(n-1)}{N}\right),\ \tau = 1,\ldots,N. \tag{23}$$

come vedremo prevista in Matlab, con l'approssimazione dei coefficienti $\tilde{c}_k$ del polinomio di interpolazione $k = -n,\ldots,n$

$$\begin{aligned} \tilde{c}_k &= \frac{1}{2n+1} \cdot \sum_{s=0}^{2n} f(t_s) \exp(-i\,k\,t_s) \\ &= \sum_{j=1}^{2n+1} \frac{f(t_{j-1})}{2n+1} \exp(-i\,k\,t_{j-1}) \end{aligned} \tag{24}$$

Notiamo che Matlab calcola $\mathscr{X}(\tau)$ per $\tau = 1,\ldots,N$ mentre $\tilde{c}_k$ ha indici $k = -n,\ldots,n$. Riportiamoci a una notazione comune.

Se $k \le 0$, da $t_j = j \cdot \frac{2\pi}{2n+1}$, $\exp(i\,s\cdot 2\pi) = 1$ per $s \in \mathbb{Z}$, abbiamo

$$\begin{aligned} \tilde{c}_{k+2n+1} &= \frac{l=1}{2n+1} \sum_{l=1}^{2n+1} f(t_{l-1}) \exp(-i\,(k+2n+1)\,t_{l-1}) \\ &= \frac{l=1}{2n+1} \sum_{l=1}^{2n+1} f(t_{l-1}) \exp\left(-i\,(k+2n+1)\,(l-1)\cdot\frac{2\pi}{2n+1}\right) \\ &= \frac{l=1}{2n+1} \sum_{l=1}^{2n+1} f(t_{l-1}) \exp\left(-i\,k\,(l-1)\cdot\frac{2\pi}{2n+1}\right) \cdot \exp\left(-i\,(2n+1)\,(l-1)\cdot\frac{2\pi}{2n+1}\right) \\ &= \frac{l=1}{2n+1} \sum_{l=1}^{2n+1} f(t_{l-1}) \exp\left(-i\,k\,(l-1)\cdot\frac{2\pi}{2n+1}\right) \\ &= \tilde{c}_k \end{aligned} \tag{25}$$

Di conseguenza possiamo ricostruire il vettore $(\tilde{c}_k)_{k=-n,\ldots,n}$ dal vettore shiftato

$$\tilde{c}^{(\mathrm{sh})}_{k+2n+1} = (\tilde{c}_0,\ldots,\tilde{c}_n,\tilde{c}_{-n},\ldots,\tilde{c}_{-1})$$

cioè $\tilde{c}^{(\mathrm{sh})}_\tau = \tilde{c}_{\tau-1}$ se $\tau \in 0,\ldots,n$ altrimenti, se $\tau \in n+1,\ldots,2n+1$ si ha $\tilde{c}^{(\mathrm{sh})}_\tau = \tilde{c}_{\tau-2n-2}$.

Si vede che

$$\tilde{c}^{(\mathrm{sh})}_\tau = \sum_{l=1}^{2n+1} \frac{f(t_{l-1})}{2n+1} \exp\left(\frac{-i(\tau-1)\,2\pi\,(l-1)}{2n+1}\right),\ \tau = 1,\ldots,2n+1$$

per cui posto $N = 2n+1$, $x(l) = \frac{f(t_{l-1})}{2n+1}$ si ha che $\mathscr{X}(\tau) = \tilde{c}_\tau^{(\mathrm{sh})}$.

Posto $\beta = n-1$, $j = \tau - 1$, $x(\beta+1) = \frac{1}{N} f\left(\frac{2\pi\beta}{N}\right)$ è immediato vedere che la formulazione

$$\mathscr{X}(\tau) = \sum_{n=1}^{N} x(n) \exp\left(\frac{-i\,2\,\pi\,(\tau-1)\,(n-1)}{N}\right),\ \tau = 1,\ldots,N. \tag{26}$$

è equivalente a

$$c_j = \frac{1}{N}\sum_{\beta=0}^{N-1} f\left(\frac{2\pi\,\beta}{N}\right)\exp\left(\frac{-i\,j2\pi\,\beta}{N}\right),\ j = 0,\ldots,N-1. \tag{27}$$

3.2 Algoritmo FFT

Per quanto visto, nella sottosezione precedente, non è restrittivo considerare il problema del calcolo dei coefficienti c_j definiti

$$c_j = \frac{1}{N}\sum_{\beta=0}^{N-1} f\left(\frac{2\pi\,\beta}{N}\right)\exp\left(\frac{-i\,j2\pi\,\beta}{N}\right),\ j = 0,\ldots,N-1. \tag{28}$$

Si ponga

$$w = \exp\left(\frac{-2\pi\,i}{N}\right),\ a_\beta = \frac{1}{N} f\left(\frac{2\pi\,\beta}{N}\right)$$

Possiamo quindi riscrivere il problema come segue: per $j = 0,\ldots,N-1$, calcolare

$$c_j = \sum_{\beta=0}^{N-1} a_\beta w^{j\beta} \text{ dove } w^N = 1 \tag{29}$$

Se per operazione intendiamo una moltiplicazione e una somma in campo complesso, allora usando lo schema di Horner, si può risolvere questo problema in N^2 operazioni. L'algoritmo noto come FFT è diventato popolare dopo uno studio di Cooley and Tukey del 1966 (cf. [2]), ma pare fosse già noto in forma limitata a Gauss (1805), permette di risolvere il problema con una complessità inferiore. Consideriamo il caso $N = 2^k$. Sia $\beta = 2\beta_1$ quando β è pari, e $\beta = 2\beta_1 + 1$ quando β è dispari. Da (29) abbiamo

$$\begin{aligned} c_j &= \sum_{\beta_1=0}^{\frac{N}{2}-1} a_{2\beta_1} w^{2j\beta_1} + \sum_{\beta_1=0}^{\frac{N}{2}-1} a_{2\beta_1+1} w^{2j\beta_1+1}. \\ &= \sum_{\beta_1=0}^{\frac{N}{2}-1} a_{2\beta_1} (w^2)^{j\beta_1} + \sum_{\beta_1=0}^{\frac{N}{2}-1} a_{2\beta_1+1} (w^2)^{j\beta_1} w^j. \end{aligned} \tag{30}$$

Sia $j = \frac{q\cdot N}{2} + j_1$. Allora, poichè $w^N = 1$ (e quindi $(w^N)^{\alpha\beta_1} = 1$)

$$(w^2)^{j\beta_1} = (w^2)^{\frac{qN\beta_1}{2}} (w^2)^{j_1\beta_1} = (w^N)^{q\beta_1} (w^2)^{j_1\beta_1} = (w^2)^{j_1\beta_1}.$$

Così se poniamo

$$\phi(j_1) = \sum_{\beta_1=0}^{\frac{N}{2}-1} a_{2\beta_1} (w^2)^{j_1\beta_1}, \ (j_1 = 0,1,\ldots,\frac{N}{2}-1) \tag{31}$$

$$\psi(j_1) = \sum_{\beta_1=0}^{\frac{N}{2}-1} a_{2\beta_1+1} (w^2)^{j_1\beta_1} \ (j_1 = 0,1,\ldots,\frac{N}{2}-1) \tag{32}$$

$$\tag{33}$$

abbiamo

$$c_j = \phi(j_1) + w^j \psi(j_1), \ j = 0,\ldots,N-1.$$

Il punto chiave dell'algoritmo è osservare che il calcolo di $\phi(j_1)$ e $\psi(j_1)$ significa che vengono compiute due analisi di Fourier con $\frac{N}{2} = 2^{k-1}$ termini invece di una con 2^k. Ora applichiamo la stessa idea alle due analisi di Fourier. Si hanno così 4 analisi di Fourier, ognuna ha 2^{k-2} termini; queste possono essere ulteriormente divise in 8 analisi di Fourier con 2^{k-3} termini, eccetera. Il numero di operazioni richieste per calcolare $\{c_j\}$ quando $\{\phi(j_1)\}$ e $\{\psi(j_1)\}$ sono stati calcolati, è al più $2 \cdot 2^k$ (un numero maggiore di calcoli può essere evitato se le potenze di w sono immagazzinate in memoria).
Sia p_k il numero totale di operazioni necessarie per calcolare i coefficienti quando $N = 2^k$. Come prima, abbiamo

$$p_k \leq 2p_{k-1} + 2 \cdot 2^k, \ k = 1,2,\ldots$$

Osserviamo che $p_0 = 1$, in quanto corrisponde ad un analisi di Fourier di ordine $N = 1$, cioè al calcolo di un coefficiente di Fourier il cui costo computazionale è pari a 1 moltiplicazione. si ha per induzione, essendo $N = 2^k$ e quindi $k = \log_2(N)$ e $2^k \gg 1$

$$\begin{aligned} p_k &\leq 2p_{k-1} + 2 \cdot 2^k \leq 2\left(2p_{k-2} + 2 \cdot 2^{k-1}\right) + 2 \cdot 2^k \\ &\leq 4\,p_{k-2} + 2 \cdot 2 \cdot 2^k = 4\left(2p_{k-3} + 2 \cdot 2^{k-2}\right) + 4 \cdot 2^k \\ &= 2^3 p_{k-3} + 8 \cdot 2^{k-2} + 4 \cdot 2^k = 2^3 p_{k-3} + 2 \cdot 2^k + 2^2 \cdot 2^k \\ &= 2^3 p_{k-3} + 2 \cdot 3 \cdot 2^k \leq 2^k p_0 + 2k \cdot 2^k \\ &= 2^k + 2k \cdot 2^k \\ &= 2k \cdot (2^k + 1) \approx 2k \cdot 2^k = 2N \cdot \log_2 N \end{aligned} \tag{34}$$

da paragonarsi con N^2 operazioni del metodo di base.

Per capire i vantaggi, facciamo qualche conto con Matlab, mostrando per alcune potenze di 2 (prima colonna), il numero di operazioni per il metodo di base (seconda colonna) e dell' FFT (terza colonna):

```
>> N=2.^(1:10);
>> N=N';
>> N2=N.^2;
>> Nlog=2*N.*log2(N);
>> [N N2 Nlog]

ans =

           2           4           4
           4          16          16
           8          64          48
          16         256         128
          32        1024         320
          64        4096         768
         128       16384        1792
         256       65536        4096
         512      262144        9216
        1024     1048576       20480

>>
```

4 FFT in Matlab

Vediamo come utilizzare in Matlab la FFT partendo dal relativo help

```
>> help fft

 FFT Discrete Fourier transform.
    FFT(X) is the discrete Fourier transform (DFT) of vector X.  For
    matrices, the FFT operation is applied to each column. For N-D
    arrays, the FFT operation operates on the first non-singleton
    dimension.

    FFT(X,N) is the N-point FFT, padded with zeros if X has less
    than N points and truncated if it has more.

    FFT(X,[],DIM) or FFT(X,N,DIM) applies the FFT operation across the
    dimension DIM.

    For length N input vector x, the DFT is a length N vector X,
    with elements
                     N
       X(k) =       sum  x(n)*exp(-j*2*pi*(k-1)*(n-1)/N), 1 <= k <= N.
                    n=1
    The inverse DFT (computed by IFFT) is given by
                     N
       x(n) = (1/N) sum  X(k)*exp( j*2*pi*(k-1)*(n-1)/N), 1 <= n <= N.
                    k=1
```

```
    See also IFFT, FFT2, IFFT2, FFTSHIFT.

 Overloaded methods
    help qfft/fft.m
```

Si capisce che tale procedura calcola, a partire da un vettore $x = (x(j))_{j=1,\ldots,N}$, il vettore $\mathscr{X} = (\mathscr{X}(k))_{k=1,\ldots,N}$ definito da

$$\mathscr{X}(\tau) = \sum_{n=1}^{N} x(n) \exp\left(\frac{-i\,2\pi\,(\tau-1)\,(n-1)}{N}\right), \tau = 1,\ldots,N. \tag{35}$$

(si noti che per errore nell'help di Matlab hanno indicato con j la costante immaginaria i).

4.1 Esempio 1

Vediamo un esempio in Matlab, in cui analizziamo per $N = 8$ la funzione

$$f(x) = 5\exp(2\,i\,x)$$

in cui evidentemente $c_0 = c_1 = c_3 = \ldots = c_8 = 0$ e $c_2 = 5$. Quindi ci aspettiamo un vettore $\tau = \{\tau_k\}$ in cui poichè $\tau_k = c_{k-1}$ si ha che $\tau_3 = 5$ e tutte le altre componenti nulle.
Salviamo nel file `esempio_FFT`

```
clear all;
format long;

n=8;
f=inline('5*exp(i*2*t)');

h=2*pi/n;
t=(0:n-1)'*h;
ft=feval(f,t);

c_fft=fft(ft,n)*h/(2*pi);
```

e otteniamo come prevedibile

```
>> c_fft
c_fft =
 -0.00000000000000 + 0.00000000000000i
 -0.00000000000000 + 0.00000000000000i
  5.00000000000000 - 0.00000000000000i
  0.00000000000000 + 0.00000000000000i
  0.00000000000000 + 0.00000000000000i
  0.00000000000000 + 0.00000000000000i
                 0 + 0.00000000000000i
 -0.00000000000000 + 0.00000000000000i
```

4.2 Esempio 2

Ora consideriamo

$$f(x) = 10\exp(-i\,2\,x)$$

e osserviamo che il risultato è

```
>> esempio_FFT
>> c_fft
c_fft =
 -0.00000000000000 - 0.00000000000000i
 -0.00000000000000 - 0.00000000000000i
                 0 - 0.00000000000000i
  0.00000000000000 - 0.00000000000000i
  0.00000000000000 - 0.00000000000000i
  0.00000000000000 - 0.00000000000000i
 10.00000000000000 + 0.00000000000000i
 -0.00000000000000 - 0.00000000000000i
```

A prima vista la cosa è sorprendente, visto che il coefficiente uguale a 10 appare alla settima componente, quella che dovrebbe essere di c_6 e non c_{-2}. Vediamo perchè . Si ha in generale per $f(x) = \lambda\exp(-i\,k\,x)$

$$\begin{aligned}
\mathscr{X}(\tau) &= \sum_{n=1}^{N} x(n)\exp\left(\frac{-i\,2\,\pi\,(\tau-1)\,(n-1)}{N}\right) \\
&= \frac{1}{N}\sum_{n=1}^{N}\lambda\exp\left(-i\,k\,\frac{2\pi(n-1)}{N}\right)\exp\left(\frac{-i\,2\,\pi\,(\tau-1)\,(n-1)}{N}\right) \\
&= \frac{1}{N}\sum_{n=1}^{N}\lambda\exp\left(-i\,2\,\frac{2\pi(\tau-1+k)(n-1)}{N}\right) \qquad (36)
\end{aligned}$$

sommatoria che vale λ se $\tau-1+k$ è multiplo di N (con τ avente valori tra 1 e N) e 0 altrimenti. Quindi nell'esempio precedente in cui $\lambda = 10$, $k = 2$, $N = 8$, la sommatoria vale 10 se $\tau - 1 + 2 = \tau + 1$ è multiplo di 8 (cioè $\tau = 7$) e 0 altrimenti.

4.3 Esempio 3

Per convincersi di quanto visto nel secondo esempio, posto $N = 8$ applichiamo la FFT a

$$f(x) = 10\exp(-6\,i\,x) + 20\exp(2\,i\,x).$$

Ci aspettiamo che il primo termine contribuisca alla sommatoria, cosicche' la terza componente valga 10 e il secondo termine contribuisca alla sommatoria, cosicche' la terza componente valga 20. Per linearità avremo quindi una terza componente uguale a 30, in quanto in generale

$$\sum_{n=1}^{N}(x(n)+y(n))\exp(\ldots) = \sum_{n=1}^{N}x(n)\exp(\ldots) + \sum_{n=1}^{N}y(n)\exp(\ldots).$$

```
>> esempio_FFT
>> c_fft
c_fft =
 -0.00000000000000 - 0.00000000000000i
  0.00000000000000 - 0.00000000000000i
 30.00000000000000 + 0.00000000000000i
 -0.00000000000000 - 0.00000000000000i
  0.00000000000000 - 0.00000000000000i
 -0.00000000000000 + 0.00000000000000i
                 0 - 0.00000000000000i
  0.00000000000000 + 0.00000000000000i
>>
```

5 Online

Citiamo alcuni siti web che espongono parte dell'argomento:

http://en.wikipedia.org/wiki/Cooley-Tukey_FFT_algorithm
http://en.wikipedia.org/wiki/Fast_Fourier_transform
http://it.wikipedia.org/wiki/Rappresentazione_spettrale_dei_segnali

References

[1] K. Atkinson, *An Introduction to Numerical Analysis*, Wiley, (1989).

[2] J.W. Cooley e J.W. Tukey, *An algorithm for the machine calculation of complex Fourier series*, Math. Comput. 19, p. 297-301, (1965).

[3] G. Dahlquist e A. Bjorck , *Numerical methods*, Dover, (2003).

[4] S.D. Conte e C. de Boor, *Elementary Numerical Analysis, An Algorithmic Approach*, McGraw-Hill, (1981).

[5] G. Gilardi *Analisi Due, seconda edizione*, McGraw-Hill, (1996).

[6] D.H. Griffel, *Applied functional analysis*, Dover publications, 2002.

[7] A.N. Kolmogorov e S.V. Fomin, *Introductory Real Analysis*, Dover publications, 1970.

[8] Planet Math (Proof of determinant of the Vandermonde matrix) http://planetmath.org/encyclopedia/PrrofOfDeterminantOfTheVandermondeMatrix.html

[9] A. Quarteroni, R. Sacco e F. Saleri *Matematica Numerica*, Springer, (1998).

[10] Wikipedia, (Serie di Fourier), http://it.wikipedia.org/wiki/Serie_di_Fourier.