

Laurea Magistrale in Informatica A.A. 2020/2021  
Università degli Studi di Milano-Bicocca  
**Riassunto Metodi del Calcolo Scientifico**

Marta Pelusi

**@Marta629**

Copyright (c) Marta629

21 giugno 2021

**Attenzione:** questo è un *riassunto* degli appunti presi durante le lezioni, quindi non saranno qui riportate:

- Le parti con conti e formule;
- Le dimostrazioni;
- Gli esempi.

Vengono *riassunte*, dunque, solamente le parti chiave teoriche del corso e alcune parti verranno date per scontate.

## Indice

<b>1</b>	<b>Algebra lineare</b>	<b>4</b>
1.1	Problema agli autovalori . . . . .	4
<b>2</b>	<b>Metodi diretti e iterativi</b>	<b>5</b>
<b>3</b>	<b>Metodi diretti</b>	<b>5</b>
3.1	Fattorizzazione di Gauss e fattorizzazione PA=LU . . . . .	5
3.2	Fattorizzazione di Cholesky . . . . .	6
<b>4</b>	<b>Metodi iterativi</b>	<b>6</b>
4.1	Metodo di Jacobi . . . . .	7
4.2	Metodo di Gauss-Seidel . . . . .	7
4.3	Criteri di arresto . . . . .	8
4.3.1	Criterio del residuo . . . . .	8
4.3.2	Criterio delle iterate successive . . . . .	8
<b>5</b>	<b>Metodi per il calcolo di autovalori</b>	<b>8</b>
5.1	Metodo delle potenze . . . . .	9
5.2	Criteri di arresto . . . . .	9
<b>6</b>	<b>Page Rank</b>	<b>9</b>
<b>7</b>	<b>Algoritmo di compressione JPEG</b>	<b>10</b>
7.1	Funzioni periodiche . . . . .	10
7.2	Funzioni pari e dispari . . . . .	11
7.3	Basi ortogonali . . . . .	11
7.4	Numeri complessi . . . . .	11
7.5	Basi $w^k$ . . . . .	12
7.6	Discrete Cosine Transform - DCT . . . . .	12
7.7	Discrete Cosine Transform 2 - DCT2 . . . . .	13
7.7.1	DCT2 e IDCT2 scalate . . . . .	14
7.8	Fenomeno di Gibbs . . . . .	14
7.9	Algoritmo JPEG (con matrice di quantizzazione) . . . . .	14
7.9.1	Compressione . . . . .	14
7.9.2	Decompressione - visualizzazione immagine . . . . .	15

# 1 Algebra lineare

Ci si è concentrati sulla risoluzione di due tipi di problemi:

- Risolvere  $Ax = b$  sistema lineare;
- Calcolo degli autovalori, ovvero trovare  $(\lambda, x)$  tale che  $Ax = \lambda x$ .

## 1.1 Problema agli autovalori

Dato  $Ax = \lambda x$ ,  $\lambda \in \mathbb{R}$ , risolvere un problema agli autovalori significa risolvere l'equazione  $(A - \lambda I)x = 0$ , ovvero

$$\det(A - \lambda I) = 0$$

dove  $A - \lambda I$  è una matrice **non invertibile** se il suo determinante è nullo.  
**Proprietà degli autovalori:**

- Gli autovettori associati ad un autovalore formano un sottospazio;
- Sia  $A$  matrice invertibile, allora gli autovalori di  $A$  sono non nulli e se  $\lambda$  è autovalore di  $A$  allora  $1/\lambda$  è autovalore di  $A^{-1}$ ;
- Se  $\lambda$  è autovalore di  $A$  allora  $\lambda^2$  è autovalore di  $A^2$ ;
- Sia  $P(x)$  un polinomio. Se  $(\lambda, x)$  è una coppia autovalore-autovettore di  $A$  allora  $(P(\lambda), x)$  è una coppia autovalore-autovettore di  $P(A)$ ;
- Sia  $D$  matrice diagonale, allora gli autovalori di  $D$  sono i suoi elementi diagonali;
- Se  $\lambda$  è autovalore di  $A$  allora  $\lambda$  è autovalore di  $A^T$ .

**Definizione (raggio spettrale)** Si dice **raggio spettrale** di  $A$  l'autovalore di modulo massimo (in modulo) di  $A$  e si indica con  $\rho(A)$ .

**Definizione (vettori ortogonali)** Due vettori  $u, v$  sono **ortogonali** se

$$u \cdot v = 0$$

**Definizione (norma euclidea)** Il prodotto scalare tra due vettore induce una **norma** su  $\mathbb{R}^n$

$$\|u\| = \sqrt{u \cdot u} = \sqrt{\sum_{i=1}^n u_i^2}$$

Si può dimostrare che per ogni matrice  $A$  si ha

$$\|A\| = \sqrt{\rho(A \cdot A^T)}$$

La norma di una matrice  $A$   $n \times n$  è definita come

$$\|A\| = \max_{x \in \mathbb{R}^{n \times n}, x \neq 0} \frac{\|Ax\|}{\|x\|}$$

Si può dimostrare che

$$\|A \cdot B\| \leq \|A\| \cdot \|B\|$$

**Definizione (numero di condizionamento)** Sia  $A \in \mathbb{R}^{n \times n}$  invertibile. Si dice **numero di condizionamento** di  $A$

$$COND(A) = \|A\| \cdot \|A^{-1}\|, \quad COND(A) \geq 1$$

Una matrice si dice **malcondizionata** se il numero di condizionamento è molto grande (circa dell'ordine di  $10^{10} - 10^{12}$ ). Il numero di condizionamento dice quanto è difficile risolvere un determinato sistema lineare. Solitamente una matrice è malcondizionata quando i suoi autovalori sono molto distanti tra loro.

## 2 Metodi diretti e iterativi

**Definizione (metodo diretto)** Un metodo si dice **diretto** se, in aritmetica esatta, siamo in grado di trovare la vera soluzione  $x$ , ovvero viene risolta esattamente l'equazione

$$Ax = b$$

**Definizione (metodo indiretto)** Un metodo si dice **indiretto o iterativo** se genera una successione  $\{x^{(k)}\}_{k \in \mathbb{N}}$  che convergerà ad  $x$ .

Tramite i metodi indiretti non si otterrà una soluzione esatta, ma si otterrà una soluzione approssimata.

L'idea è generare una successione  $\{x^{(k)}\}_{k \in \mathbb{N}}$  che converga alla soluzione esatta  $x^*$ . I metodi indiretti, quindi, in aritmetica esatta non forniscono la soluzione esatta in un numero finito di operazioni, ma occorre troncare la successione mediante opportuni criteri.

## 3 Metodi diretti

### 3.1 Fattorizzazione di Gauss e fattorizzazione PA=LU

Sia l'algoritmo di Gauss che la fattorizzazione PA=LU consistono nell'operare su un sistema lineare in modo da ottenere un sistema triangolare superiore.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases} \longrightarrow \begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ 0 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots \\ 0 + 0 + \dots + 0 + a_{nn}x_n = b_n \end{cases}$$

Un sistema lineare è equivalente ad un altro ottenuto scambiando le righe o cambiando un'equazione con la stessa più il multiplo di un'altra equazione del sistema.

Gli elementi  $a_{ii}^{(i-1)}$  sono detti **pivot** e la strategia migliore (chiamata **strategia**

**del pivot parziale**) consiste nel prendere dal passo  $i - 1$ -esimo il pivot sulla colonna  $i$  di **modulo massimo** in modo da minimizzare i rapporti

$$\frac{a_{ji}^{(i-1)}}{a_{ii}^{(i-1)}}, \quad j = i - 1, \dots, n$$

Il costo computazionale dell'algoritmo di eliminazione di Gauss è

$$O(n^3)$$

L'obiettivo della fattorizzazione PA=LU è fattorizzare la matrice  $A$  come

$$A = LU$$

### 3.2 Fattorizzazione di Cholesky

**Definizione (matrice definita positiva)** Una matrice  $A$  si dice **definita positiva** se

- $A$  è simmetrica;
- $(Ax, x) > 0 \quad \forall x \in \mathbb{R}^n, \quad x \neq 0$ .

**Proprietà 1** Gli autovalori di una matrice definita positiva sono tutti positivi.

**Proprietà 2 (criterio di Sylvester)** Una matrice  $A$  è definita positiva se e solo se tutte le sottomatrici di  $A$  "compatibili" con la diagonale hanno determinate positivo.

**Fattorizzazione di Cholesky** Una matrice  $A$  simmetrica è definita positiva se e solo se esiste una matrice  $R$  triangolare superiore tale che

$$A = R^T \cdot R$$

$R$  è **unica** se si impone che i suoi elementi diagonali siano tutti positivi.

## 4 Metodi iterativi

I metodi iterativi si introducono per ovviare al fenomeno del **fill-in** (riempimento): facendo la fattorizzazione PA=LU di una matrice  $A$  sparsa a bande, si ottengono delle matrici  $L$  ed  $U$  con un numero di elementi non nulli maggiore rispetto a quello di  $A$ .

Dato un sistema lineare  $Ax = b$  con soluzione esatta  $x^*$ , si vuole generare una successione  $\{x^{(k)}\}_{k \in \mathbb{N}}$  tale che

$$\lim_{k \rightarrow +\infty} x^{(k)} = x^*$$

Criteri di convergenza per vettori:

- $\lim_{k \rightarrow +\infty} \|x^{(k)} - x^*\| = 0$ ;
- $\lim_{k \rightarrow +\infty} x_i^{(k)} = x_i^*, \quad i = 1, \dots, n$ .

## 4.1 Metodo di Jacobi

Il **metodo di Jacobi** si esprime tramite la seguente formula di calcolo della sequenza

$$x^{(k)} = D^{-1} [b - (L + U) x^{(k-1)}]$$

Con questo metodo non si hanno garanzie circa la convergenza di una successione, però si sa che se  $x^{(k)}$  converge, allora sicuramente converge alla soluzione esatta  $x^*$ .

Il metodo di Jacobi ha costo computazionale  $O(n^2)$ , che può essere abbassato se la matrice considerata è sparsa.

## 4.2 Metodo di Gauss-Seidel

Il **metodo di Gauss-Seidel** si esprime tramite la seguente formula di calcolo della sequenza

$$x^{(k)} = (D + L)^{-1}(b - Ux^{(k-1)})$$

Anche in questo caso non si sa con certezza se  $x^{(k)}$  converge, ma sicuramente se converge, allora converge alla soluzione esatta  $x^*$ .

In generale, quindi, un metodo iterativo lo si può scrivere nella forma

$$x^{(k)} B x^{(k-1)} + d \quad (\star)$$

- Jacobi:  $B = -D^{-1}(L + U)$ ,  $d = D^{-1}b$ ;
- Gauss-Seidel:  $B = -(D + L)^{-1}U$ ,  $d = (D + L)^{-1}b$ .

È possibile dimostrare che il metodo iterativo  $(\star)$  converge (e quindi converge anche alla soluzione esatta  $x^*$ ) se e solo se

$$\rho(B) < 1$$

**Definizione (predominanza per righe)** Una matrice  $A$  si dice a **predominanza diagonale per righe** se

$$\sum_{j=1, j \neq i}^n |a_{ij}| < |a_{ii}|$$

**Definizione (predominanza per colonne)** Una matrice  $A$  si dice a **predominanza diagonale per colonne** se

$$\sum_{j=1, j \neq i}^n |a_{ji}| < |a_{ii}|$$

È possibile dimostrare che i metodi di Jacobi e Gauss-Seidel convergono per qualsiasi scelta del vettore iniziale  $x^{(0)}$  se la matrice  $A$  è a predominanza diagonale o per righe o per colonne. Il vettore iniziale  $x^{(0)}$  è il punto di partenza per generare

la successione  $x^{(k)}$ .

Come detto inizialmente, i metodi iterativi vogliono generare una successione che converge alla soluzione esatta del sistema. Ma qual è il momento giusto per stoppare l'algoritmo?

Idealmente si stoppa l'algoritmo quando

$$\frac{\|x^{(k+1)} - x^*\|}{\|x^*\|} < TOLL$$

ma non si conosce la soluzione esatta  $x^*$ , quindi vengono usati i criteri di arresto.

## 4.3 Criteri di arresto

### 4.3.1 Criterio del residuo

L'idea è chiedersi quanto  $x^{(k+1)}$  sia "lontana" dal risolvere il sistema. La formula su cui si basa questo criterio è

$$\frac{\|x^{(k+1)} - x^*\|}{\|x^*\|} < COND(A) \cdot TOLL$$

dove

- $r^{(k+1)} = Ax^{(k+1)} - b$ , se  $r^{(k+1)} = 0 \Rightarrow x^{(k+1)} = x^*$ ;
- $\frac{\|r^{(k+1)}\|}{\|b\|} < TOLL$  è l'errore relativo nella perturbazione;
- $TOLL$  è detta tolleranza.

### 4.3.2 Criterio delle iterate successive

Se la successione converge da un certo punto in poi, ovvero vale

$$\frac{\|x^{(k+1)} - x^{(k)}\|}{\|x^{(k)}\|} < TOLL$$

allora si può dimostrare che

$$\|x^{(k+1)} - x^*\| \leq \frac{\rho(B)}{1 - \rho(B)} \cdot \|x^{(k+1)} - x^{(k)}\|$$

## 5 Metodi per il calcolo di autovalori

Data  $A \in \mathbb{R}^{n \times n}$  si vuole trovare  $x \in \mathbb{R}^n$ ,  $x \neq 0$ ,  $\lambda \in \mathbb{R}$  tale che

$$Ax = \lambda x$$

Calcolare gli autovalori significa trovare gli zeri dell'equazione

$$\det(A - \lambda I) = 0$$



## 5.1 Metodo delle potenze

Sia  $A \in \mathbb{R}^{n \times n}$  e siano  $\lambda_1, \dots, \lambda_n$  gli autovalori associati agli autovettori  $x_1, \dots, x_n$  con  $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$ , allora

$$Ax_i = \lambda_i x_i, \quad i = 1, \dots, n$$

Supponendo che  $A$  abbia un solo autovalore di modulo massimo, l'obiettivo è calcolare l'autovettore  $\lambda_1$  di modulo massimo (associato all'autovettore  $x_1$ ).

Il **metodo delle potenze** consiste nel generare una successione  $\{q^{(k)}\}_{k \in \mathbb{N}}$  di vettori e una successione  $\{\lambda_1^{(k)}\}_{k \in \mathbb{N}}$  di scalari tale che

$$\lim_{k \rightarrow +\infty} \lambda_1^{(k)} = \lambda_1 \text{ e } \lim_{k \rightarrow +\infty} q^{(k)} = q$$

con  $q$  autovettore di  $A$  relativo a  $\lambda_1$  di norma unitaria.

**Definizione (molteplicità)**  $\lambda$  autovalore  $\Leftrightarrow \det(A - \lambda I) = 0$ . Si dice che  $\lambda$  ha **molteplicità**  $k$  se  $\text{rank}(A - \lambda I) = n - k$ .

Per matrici sparse il costo computazionale è di  $O(n)$ .

## 5.2 Criteri di arresto

Anche in questo caso esistono due criteri di arresto:

- **Criterio del residuo**, che consiste nello stimare  $\|Aq^{(k)} - \lambda_1^{(k)} q^{(k)}\|$ ;
- **Criterio delle iterate successive**, che consiste nello stimare  $|\lambda_1^{(k)} - \lambda_1^{(k-1)}|$

## 6 Page Rank

Il **page rank** consiste di ordinare le pagine web secondo un ordine di importanza. Si presentano, però, una serie di problematiche:

- Le pagine sono *tantissime*
- Come si fa a stabilire quale pagina è più importante di un'altra?

Si può vedere il web come una rete sociale, ovvero più persone si conoscono, più link si hanno e più si è importanti. Le pagine web, infatti, si "parlano" attraverso i link.

Il modello associato al page rank di Google è un problema agli autovalori

$$V \cdot w = \lambda \cdot w, \quad \lambda = 1$$

**Teorema (di Perron-Frobenius)** Sia  $A \in \mathbb{R}^{n \times n}$  con elementi strettamente positivi, allora:

- (a)  $A$  ammette un solo autovalore di modulo massimo;
- (b) Tale autovalore ha un solo autovettore associato (con molteplicità 1);
- (c) L'autovettore ha componenti non negative.

Il **metodo delle potenze** per il calcolo degli autovalori si riassume in:

- Scegliere  $w^{(0)}$  tale che  $(w^{(0)}) \cdot e = 1$ ;
- Calcolare  $w^{(k+1)} = A^T w^{(k)}$ .

Infine si ottiene

$$A^T w^{(k)} = \gamma \left[ H^T y^{(k)} + \left( (y^{(k)})^T \cdot u \right) e \right] + \frac{1 - \gamma}{N} \cdot e$$

## 7 Algoritmo di compressione JPEG

Il JPEG è un **formato di compressione di immagini** ed è un formato di tipo **lossy**, ovvero porta alla perdita di alcune informazioni.

Se l'immagine presenta un forte contrasto, allora comprimerla in JPEG non è la cosa migliore da fare. La compressione in JPEG risulta ottimale se l'immagine presenta sfumature tra le figure.

L'idea che sta alla base della compressione in JPEG è quella di tagliare delle frequenze alte impercettibili all'uomo.

### 7.1 Funzioni periodiche

**Definizione (funzione periodica)**  $f : \mathbb{R} \rightarrow \mathbb{R}$  è una **funzione periodica di periodo  $p$**  se vale

$$f(x) = f(x + p), \quad \forall x \in \mathbb{R}$$

In generale tutte le funzioni del tipo:

- $f(x) = \sin(2\pi kx)$ , con valori compresi tra  $[-1, 1]$  che si ripetono in  $[0, 1]$ ;
- $f(x) = \cos(2\pi kx)$ .

sono **periodiche di periodo 1**. Il numero di oscillazioni è dato da  $k$ , che è detto **frequenza** di  $f$ .

**Proprietà** Sia  $f : \mathbb{R} \rightarrow \mathbb{R}$  una funzione periodica di periodo  $p$ , allora  $\forall a \in \mathbb{R}$

$$\int_a^{a+p} f(x) \, dx = \int_0^p f(x) \, dx$$

Data  $f : \mathbb{R} \rightarrow \mathbb{R}$  periodica di periodo  $p$ , si vuole scrivere la seguente uguaglianza chiamata **serie di Fourier**

$$f(x) = a_0 + \sum_{k=1}^{+\infty} \left[ a_k \cdot \sin\left(\frac{2\pi kx}{p}\right) + b_k \cdot \cos\left(\frac{2\pi kx}{p}\right) \right]$$

La serie di Fourier equivale a scrivere la funzione  $f$  come **combinazione lineare** degli elementi dell'insieme  $\left\{1, \cos\left(\frac{2\pi kx}{p}\right), \sin\left(\frac{w\pi kx}{p}\right)\right\}$ ,  $k \geq 1$ . La serie di Fourier permette di **decomporre  $f$  in frequenze**.

**Nota:** la serie di Fourier si può avere anche per funzioni non periodiche.

**Teorema (di Dirichlet)** Sia  $f : \mathbb{R} \rightarrow \mathbb{R}$  periodica di periodo  $p$ . Assumiamo che  $f$  sia  $\mathbb{C}^1$  a tratti (convergenza veloce). Allora se  $f$  è continua in  $x \in \mathbb{R}$  la serie  $f_N(x)$  converge a  $f(x)$ , altrimenti  $f_N(x)$  converge alla media dei valori destro e sinistro.

## 7.2 Funzioni pari e dispari

**Definizione (funzione pari)**  $f : \mathbb{R} \rightarrow \mathbb{R}$  si dice **pari** (simmetrica rispetto all'asse delle  $y$ ) se

$$f(x) = f(-x)$$

**Definizione (funzione dispari)**  $f : \mathbb{R} \rightarrow \mathbb{R}$  si dice **dispari** (simmetrica rispetto all'origine) se

$$f(x) = -f(-x)$$

## 7.3 Basi ortogonali

**Definizione (base)** Consideriamo uno spazio vettoriale  $\mathbb{R}^N$ . L'insieme  $\mathcal{B} = \{\underline{w}_1, \dots, \underline{w}_N\} \subseteq \mathbb{R}^N$  costituisce una **base** di  $\mathbb{R}^N$  se ogni vettore  $\underline{v} \in \mathbb{R}^N$  si può scrivere come

$$\underline{v} = c\underline{w}_1 + \dots + c_N\underline{w}_N$$

**Definizione (base ortogonale)** Sia  $\mathcal{B} = \{\underline{w}_1, \dots, \underline{w}_N\}$  una base di  $\mathbb{R}^N$ . La base  $\mathcal{B}$  si dice **ortogonale** se

$$\underline{w}_j \cdot \underline{w}_k = 0, \quad j \neq k$$

Avendo  $\underline{v} \in \mathbb{R}^N$  si vuole scrivere  $\underline{v} = c\underline{w}_1 + \dots + c_N\underline{w}_N$ , quindi si vogliono calcolare i coefficienti  $c_1, \dots, c_N \in \mathbb{R}$

$$c_k = \frac{\underline{v} \cdot \underline{w}_k}{\|\underline{w}_k\|^2}, \quad k = 1, \dots, N$$

## 7.4 Numeri complessi

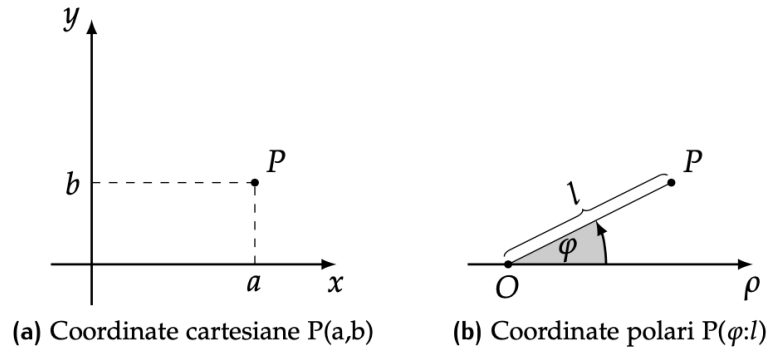
Sia  $\mathbb{R}$  l'insieme dei numeri reali.  $\sqrt{-1} \notin \mathbb{R}$  perchè non può essere calcolato in  $\mathbb{R}$ .  $i = \sqrt{-1}$  viene chiamata **unità immaginaria**.

**Definizione (numeri complessi)** Definiamo l'insieme

$$\mathbb{C} = \{z \text{ t.c.} : z = a + ib, \text{ con } a, b \in \mathbb{R}\}$$

come l'insieme dei numeri complessi, in cui  $z = a + ib$  con  $a$  **parte reale** di  $z$  e  $b$  **parte immaginaria** di  $z$ .

I numeri complessi possono essere rappresentati sia in forma algebrica che in forma grafica che in forma trigonometrica.



**Formula di Moivre** Siano  $\theta \in \mathbb{R}$  e  $k \in \mathbb{N}$ ,  $j \geq 1$ , allora

$$\cos(\theta j) + i \sin(\theta j) = (\cos \theta + i \sin \theta)^j$$

## 7.5 Basi $w^k$

Considerando lo spazio  $\mathbb{R}^N$  e  $\underline{v} = [v_1, \dots, v_N] \in \mathbb{R}^N$  un generico vettore, si vogliono introdurre dei vettori  $\underline{w}^k$  con  $k = 0, 1, \dots, N-1$ . Questi vettori costituiscono una base di  $\mathbb{R}^N$ .

L'obiettivo è definire i vettori  $\underline{w}^k$  e quindi definire  $(\underline{w}^k)_j$  come

$$(\underline{w}^k)_j = \cos(k\pi x_j) = \cos\left(k\pi \cdot \frac{2j-1}{2N}\right), \quad j = 1, \dots, N, \quad k = 0, \dots, N-1$$

Un'altro obiettivo è mostrare che i vettori  $\underline{w}^k$  rappresentano una **base ortogonale** di  $\mathbb{R}^N$ .

**Proposizione:** sia  $z \in \mathbb{C}$ ,  $z \neq 1$ , allora

$$\sum_{j=0}^{N-1} z^j = \frac{1 - z^N}{1 - z}$$

## 7.6 Discrete Cosine Transform - DCT

Dopo aver introdotto  $\mathcal{B}_N = \{\underline{w}^0, \dots, \underline{w}^{N-1}\} \subseteq \mathbb{R}^N$  base e  $\underline{w}^k \in \mathbb{R}^N$ ,  $k = 0, \dots, N-1$ , si sa che

$$(\underline{w}^k)_j = \cos\left(k\pi \cdot \frac{2j-1}{2N}\right)$$

Si sa anche che  $\mathcal{B}_N$  è una **base ortogonale**, ovvero

$$\underline{w}^k \cdot \underline{w}^q = \begin{cases} 0 & k \neq q \\ N & k = q = 0 \\ \frac{N}{2} & k = q \neq 0 \end{cases}$$

Si vuole introdurre la **DCT**: si vuole riscrivere  $\underline{v} = v_1 e_1 + v_2 e_2 + \dots + v_N e_N$  come combinazione lineare della base  $\mathcal{B}_N$ , ovvero

$$\underline{v} = c_0 \underline{w}^0 + c_1 \underline{w}^1 + \dots + c_{N-1} \underline{w}^{N-1}$$

Richiamando quanto detto [prima](#) si sa che

$$c_k = \frac{\underline{w}^k \cdot \underline{v}}{\underline{w}^k \underline{w}^k}, \quad \alpha_k = \underline{w}^k \cdot \underline{w}^k = \begin{cases} \frac{N}{2} & k \neq 0 \\ N & k = 0 \end{cases}$$

Facendo qualche calcolo si ottiene

$$c_k = \frac{1}{\alpha_k} \cdot \sum_{j=1}^N \cos \left( k\pi \cdot \frac{2j-1}{2N} \right) v_j$$

La **DCT** è un'applicazione che fa quanto segue:

$$\begin{aligned} [v_1, v_2, \dots, v_N] &\xrightarrow{DCT} [c_0, c_1, \dots, c_{N-1}] \\ \underline{v} = \sum_{j=1}^N v_j e_j &\xrightarrow{DCT} \underline{v} = \sum_{j=0}^{N-1} c_k \underline{w}^k \end{aligned}$$

Si può definire anche la trasformazione inversa, chiamata **Inverse Discrete Cosine Transform - IDCT**:

$$\begin{aligned} [c_0, c_1, \dots, c_{N-1}] &\xrightarrow{IDCT} [v_1, v_2, \dots, v_N] \\ \underline{v} = \sum_{j=0}^{N-1} c_k \underline{w}^k &\xrightarrow{IDCT} \underline{v} = \sum_{j=1}^N v_j e_j \end{aligned}$$

Si possono riassumere le due formule:

$$DCT : \quad c_k = \frac{1}{\alpha_k} \cdot \sum_{j=1}^N v_j \cos \left( k\pi \cdot \frac{2j-1}{2N} \right), \quad k = 0, \dots, N-1$$

$$IDCT : \quad v_j = \sum_{k=0}^{N-1} c_k \cos \left( k\pi \cdot \frac{2j-1}{2N} \right), \quad j = 1, \dots, N$$

## 7.7 Dicrete Cosine Transform 2 - DCT2

La DCT2 è l'applicazione della DCT monodimensionale a matrici (o immagini) bidimensionali. Le formule generali si possono riassumere nel seguente modo:

$$DCT2 : \quad c_{kq} = \frac{1}{\alpha_k \cdot \alpha_q} \cdot \sum_{i=1}^N \sum_{j=1}^N m_{ij} \cos \left( k\pi \cdot \frac{2i-1}{2N} \right) \cos \left( q\pi \cdot \frac{2j-1}{2N} \right), \quad k, q = 0, \dots, N-1$$

$$IDCT2 : \quad m_{ij} = \sum_{k=0}^{N-1} \sum_{q=0}^{N-1} c_{kq} \cos \left( k\pi \cdot \frac{2i-1}{2N} \right) \cos \left( q\pi \cdot \frac{2j-1}{2N} \right), \quad i, j = 1, \dots, N$$

Il **costo computazionale** della DCT è di  $O(N^2)$ , mentre quello della DCT2 è di  $O(N^3)$  operazioni.

La **Fast Fourier Transform - FFT** calcola la DCT in un tempo di  $O(N \cdot \log N)$  operazioni, mentre calcola la DCT2 in un tempo di  $O(N^2 \cdot \log N)$  operazioni.

### 7.7.1 DCT2 e IDCT2 scalate

$$DCT2 : \quad c_{kq} = \frac{1}{\sqrt{\alpha_k \cdot \alpha_q}} \cdot \sum_{i=1}^N \sum_{j=1}^N m_{ij} \cos \left( k\pi \cdot \frac{2i-1}{2N} \right) \cos \left( q\pi \cdot \frac{2j-1}{2N} \right), \quad k, q = 0, \dots, N-1$$

$$IDCT2 : \quad m_{ij} = \frac{1}{\sqrt{\alpha_k \cdot \alpha_q}} \sum_{k=0}^{N-1} \sum_{q=0}^{N-1} c_{kq} \cos \left( k\pi \cdot \frac{2i-1}{2N} \right) \cos \left( q\pi \cdot \frac{2j-1}{2N} \right), \quad i, j = 1, \dots, N$$

L'idea che sta alla base della compressione JPEG è:

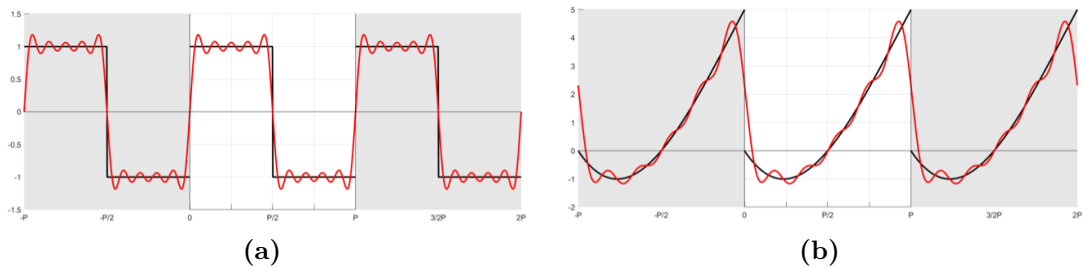
1. Leggere l'immagine come matrice di valori;
2. Trasformare l'immagine, tramite la DCT2, in una matrice di frequenze;
3. Lavorare sulla matrice delle frequenze eliminando quelle più alte impercettibili all'occhio umano per alleggerirne le dimensioni;
4. Utilizzare la IDCT2 per trasformare la matrice delle frequenze in una nuova matrice di valori che sarà poi l'immagine compressa.

## 7.8 Fenomeno di Gibbs

Il fenomeno di Gibbs è un fenomeno che denota la **presenza di forti oscillazioni** nei polinomi di Fourier che vogliono approssimare una funzione con **discontinuità** di prima specie. Si osserva che queste anomalie, presenti soprattutto vicino ai punti di discontinuità, non sembrano diminuire aumentando il grado del polinomio.

Questo è il motivo del fatto che in immagini fortemente contrastate non è ottimale usare la compressione in JPEG.

Sono mostrati in seguito due esempi del fenomeno di Gibbs.



## 7.9 Algoritmo JPEG (con matrice di quantizzazione)

### 7.9.1 Compressione

1.  $A \in \mathbb{R}^{8 \times 8}$  macroblocco con  $a_{ij} \in \mathbb{Z}$  in  $[0, 255]$ ;
2.  $A^1 = A - 128$  spostato di 128 gli elementi della matrice  $A$ , ovvero  $a_{ij}^1 \in \mathbb{Z}$  in  $[-128, 127]$ ;

3.  $C = DCT2(A)$ ,  $C \in \mathbb{R}^{8 \times 8}$ ;
4.  $Q \in \mathbb{N}^{8 \times 8}$  con  $C^1 = C./Q$ ,  $c_{ij}^1 = \frac{c_{ij}}{q_{ij}}$ ;
5.  $C^2 = \text{round}(C^1)$ .

### 7.9.2 Decompressione - visualizzazione immagine

1.  $C^3 = C^2 \cdot * Q$ ,  $c_{ij}^3 = c_{ij}^2 \cdot q_{ij}$ ;
2.  $A^2 = IDCT2(C^3)$ ;
3.  $A^3 = A^2 + 128$ ;
4.  $A^4$  aggiustando gli elementi di  $A^3$  come interi in  $[0, 255]$ .

La **quality**  $q$  è un parametro scelto dall'utente per determinare la qualità della compressione:  $q \in \mathbb{Z}$  in  $[1, 100]$ . Definiamo  $q_f(q)$  come una funzione continua come segue

$$q_f(q) = \begin{cases} \frac{200-2 \cdot q}{100} & 50 \leq q \leq 99 \\ \frac{50}{q} & 1 \leq q \leq 50 \end{cases}$$

Scegliendo una quality bassa si applica la **quantizzazione** come segue: ad esempio scegliendo una quality pari a  $q = 1 \Rightarrow q_f(1) = 50$  si applica una quantizzazione di  $50 \cdot Q$ .