

# ESERCITAZIONE

## ALGORITMI ITERATIVI DI RICOSTRUZIONE DELLE IMMAGINI DI TOMOGRAFIA AD EMISSIONE

Maximum Likelihood Expectation Maximization (MLEM)  
Ordered Subset Expectation Maximization (OSEM)

### MATERIALE

- Fantoccio cerebrale 2D (111x111 pixel) preparato come file **brain.mat**
- Funzione *MATLAB* **Calcolo\_A.m** già utilizzata nell'esercitazione sulle tecniche di ricostruzione analitiche
- Scheletro dell'esercitazione da svolgere con:
  1. predisposizione dei parametri base (liberamente modificabili) per ottenere risultati comparabili con quelli qui proposti;
  2. implementazione (già sviluppata) della parte di simulazione del sinogramma e delle sorgenti artefattuali di rumore (attenuazione, scattering e conteggi random) come già svolto in una delle esercitazioni precedenti;
  3. descrizione del codice da implementare e dei risultati da visualizzare.
- Scheletro della funzione **Calcolo\_Hblock.m** con descrizione di quale debba essere la sua struttura e il suo funzionamento.

### PUNTI DA SVOLGERE

#### 1. Ricostruzione ML-EM

- (a) Implementare l'algoritmo ML-EM per la ricostruzione del sinogramma rumoroso generato ai punti precedenti.
- (b) Valutare la qualità della ricostruzione a seconda che vengano corretti o meno i disturbi simulati.
- (c) Salvare l'immagine intermedia ricostruita ad ogni iterazione in un vettore 3D (N,N,iter\_mlem).

#### 2. Ricostruzione OSEM: calcolo dei blocchi della matrice di sistema

- (a) Creare una funzione esterna (partire dal file **Calcolo\_Ablock.m** fornito) per l'estrazione dei blocchi della matrice di sistema con cui ricostruire i singoli subset del sinogramma.
- (b) La descrizione della funzione è fornita nel file dedicato: è importante assicurarsi che restituisca in output 'nblock' segmenti della matrice di sistema A e, per ciascuno di essi, tenga traccia delle proiezioni che fanno parte del subset a cui è associato un determinato blocco.

### 3. Ricostruzione OS-EM

- (a) Implementare l'algoritmo OS-EM per la ricostruzione del sinogramma rumoroso generato ai punti precedenti.
- (b) Valutare la qualità della ricostruzione a seconda che vengano corretti o meno i disturbi simulati.
- (c) Salvare l'immagine intermedia ricostruita ad ogni iterazione in un vettore 3D ( $N, N, \text{iter\_osem} * \text{nblock}$ ).

### 4. Analisi della relazione tra numero di subset e numero di iterazioni OSEM

- (a) Visualizzare il risultato della ricostruzione OSEM usando diverse combinazioni di valori per il numero di subset (fattore di accelerazione) e il numero di iterazioni di ricostruzione.

### 5. Valutazione dell'accelerazione ottenuta grazie all'algoritmo OS-EM

- (a) Misurare i tempi di esecuzione di una ricostruzione ML-EM (totale, non delle singole sub-iterazioni) e di una ricostruzione OS-EM e verificare che ci sia una velocizzazione del processo di ricostruzione a parità di iterazioni utilizzate

#### **NOTA BENE:**

- mettersi nella condizione  $\text{iter\_mlem} = \text{iter\_osem} * \text{nblock}$
- verificare tutte e 3 le combinazioni possibili (es.  $\text{iter\_mlem}=40$ ;  $\text{iter\_osem}=2$   $\text{nblock}=20$ ;  $\text{iter\_osem}=20$   $\text{nblock}=2$ )
- assicurarsi di misurare l'effettivo tempo di ricostruzione, privo di sovraccarichi legati a salvataggio di ricostruzioni intermedie e visualizzazione degli stessi

## FACOLTATIVO

### 1. Ricostruzione MAP-OSL con prior quadratico - smoothing -

- (a) Partire (a scelta) dal codice MLEM o OSEM precedentemente implementati
- (b) Individuare il punto dell'algoritmo in cui inserire il prior facendo riferimento alle formule discusse a lezione
- (c) Il kernel 3x3 fornito è tale da calcolare  $dE(f)/df$  tramite una semplice conv2 (attenzione a cosa è 'f', quale dimensione ha normalmente, quale dimensione deve avere per poter essere convoluta con il kernel, e se è necessario fare un reshape prima di inserire il prior stimato nell'algoritmo di ricostruzione)
- (d) verificare come cambia il comportamento al variare di beta (peso del prior) e come per valori di beta troppo alti il denominatore della MLEM diventa negativo e la ricostruzione non converge più)

## RISULTATI ATTESI

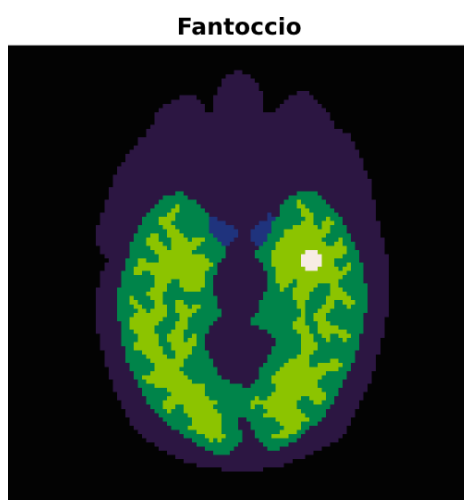
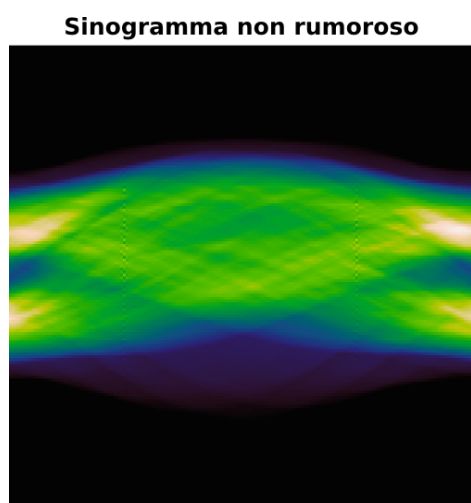
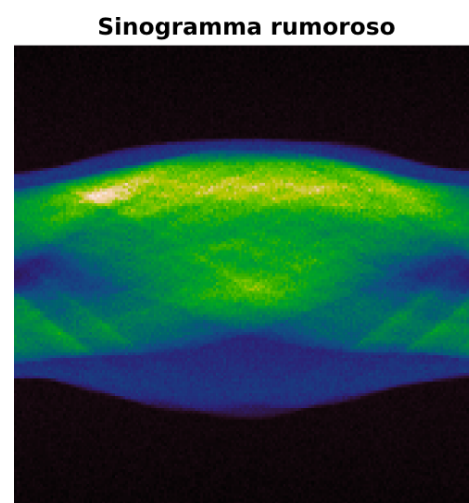


Figura 1: Fantoccio cerebrale



(a) Sinogramma ideale non rumoroso



(b) Sinogramma con artefatti e rumore Poissoniano

Tabella 1: VALUTAZIONE DELL'ACCELERAZIONE OTTENUTA CON ALGORITMO OS-EM

<b>MLEM 360 iter</b>	3.609439e+00
<b>OSEM 9 subset 40 iter</b>	4.635321e-01
<b>OSEM 40 subset 9 iter</b>	1.287901e-01

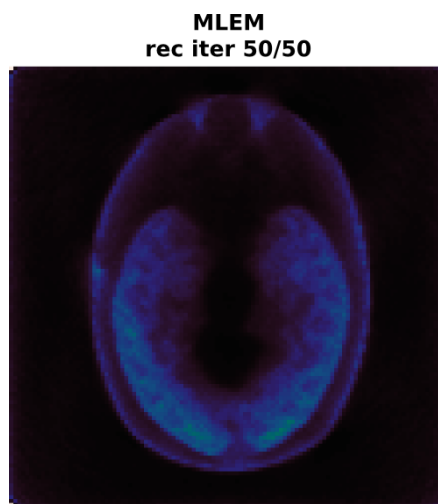


Figura 2: Ricostruzione MLEM senza correzioni

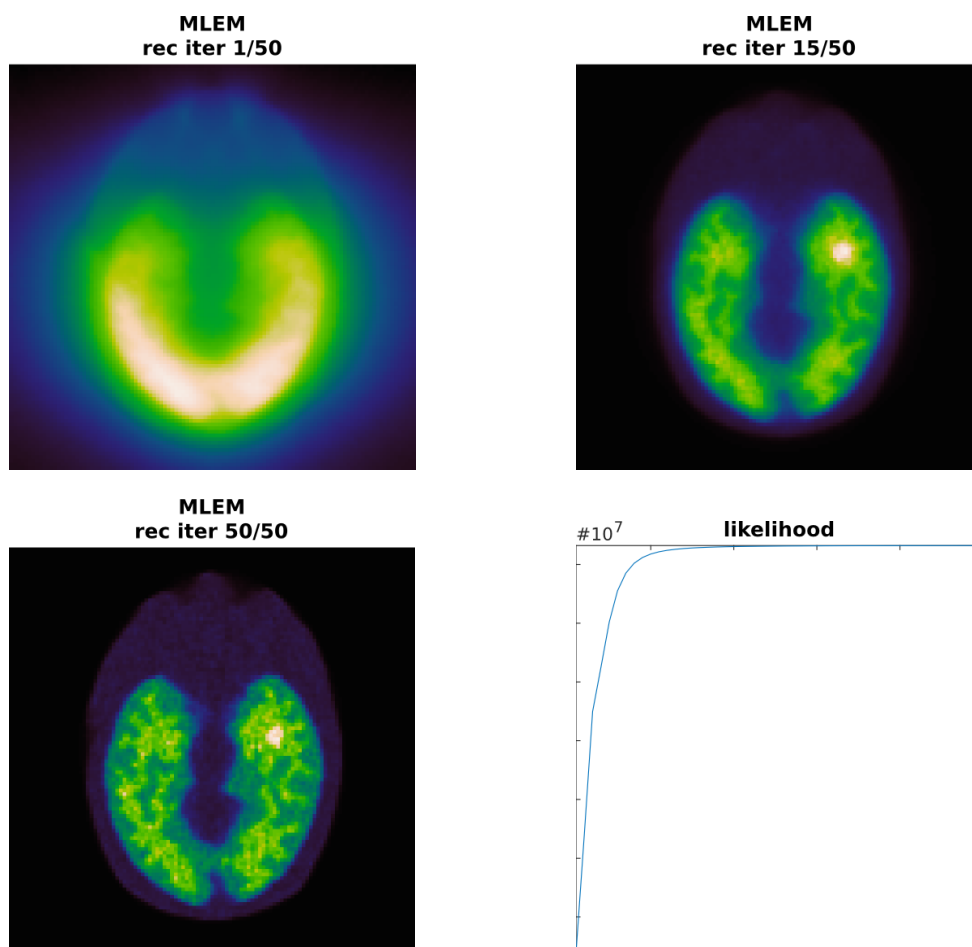


Figura 3: Ricostruzione MLEM

**OSEM  
rec iter 2/2  
subset 25/25**

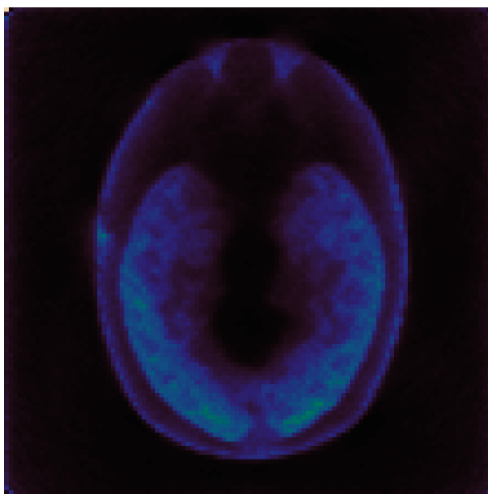
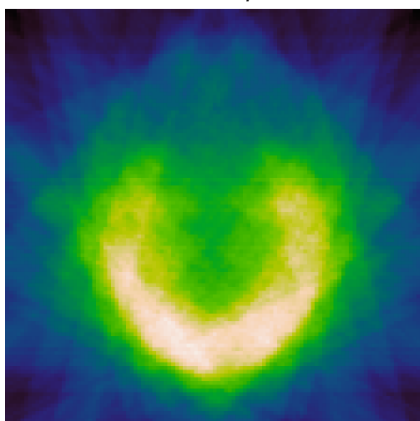
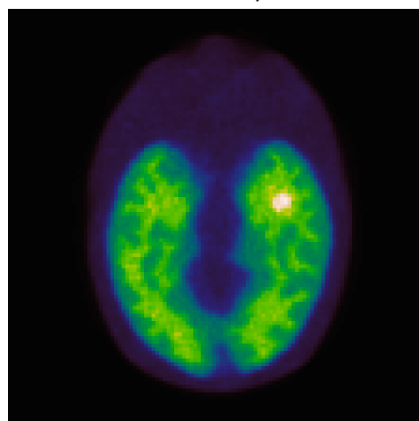


Figura 4: Ricostruzione OSEM senza correzioni

**OSEM  
rec iter 1/2  
subset 1/25**



**OSEM  
rec iter 1/2  
subset 25/25**



**OSEM  
rec iter 2/2  
subset 25/25**

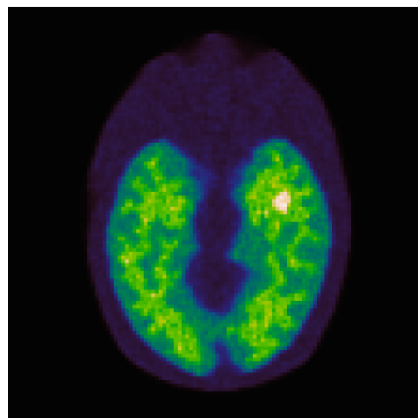


Figura 5: Ricostruzione OSEM

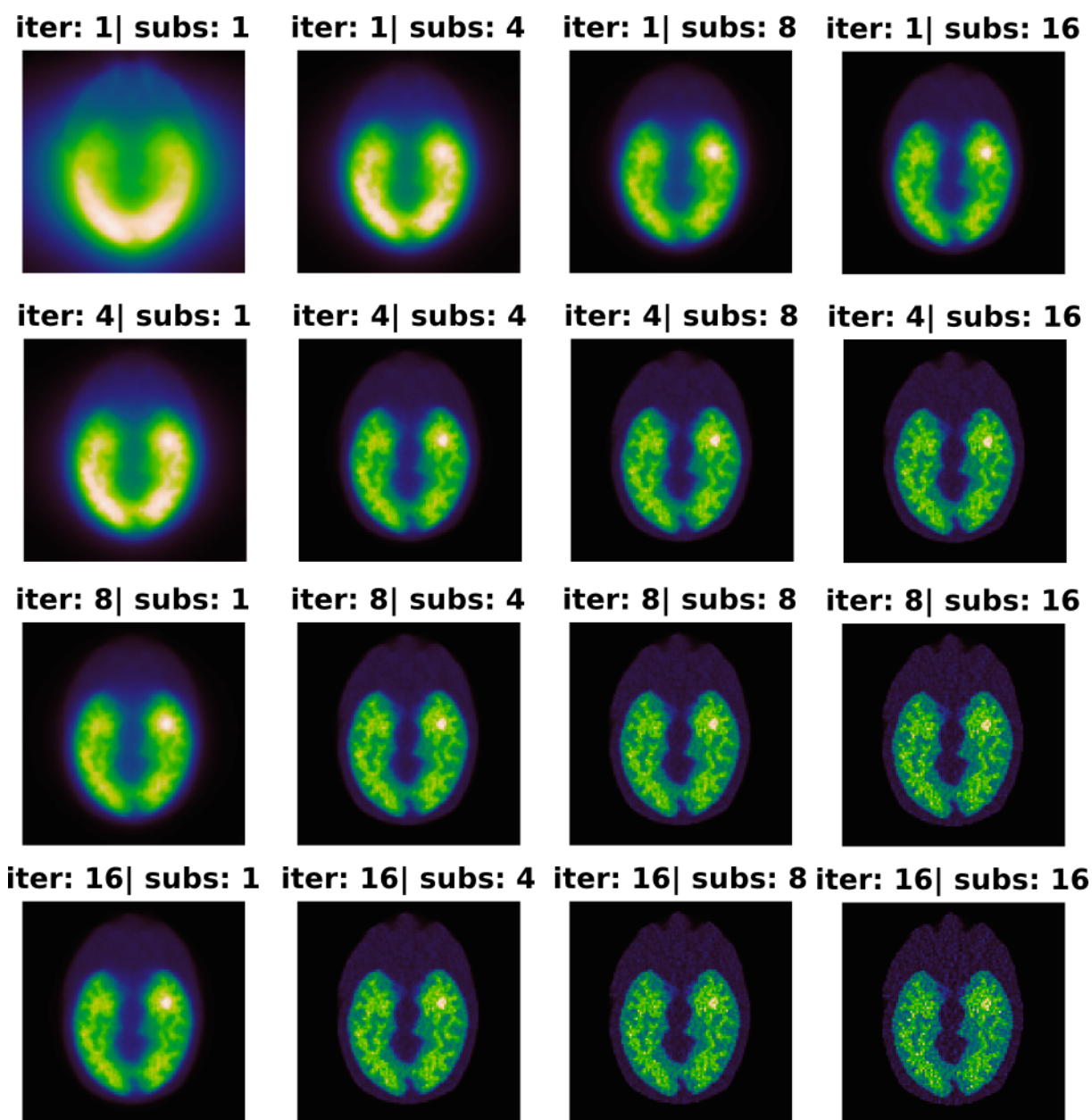


Figura 6: Relazione tra numero di subset e numero di iterazioni di algoritmo OSEM

**Sinogramma rumoroso**

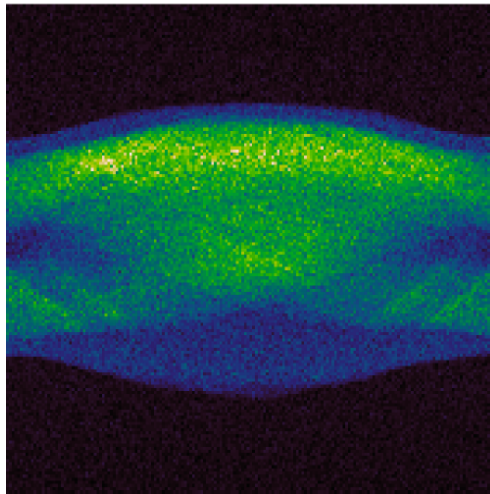
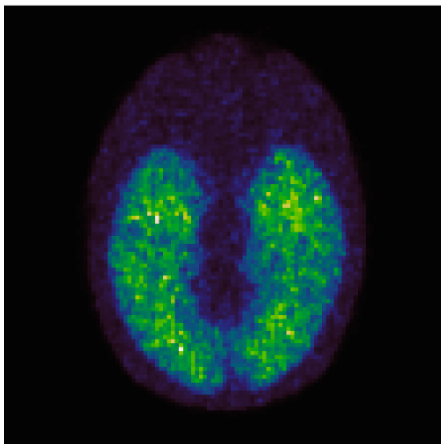
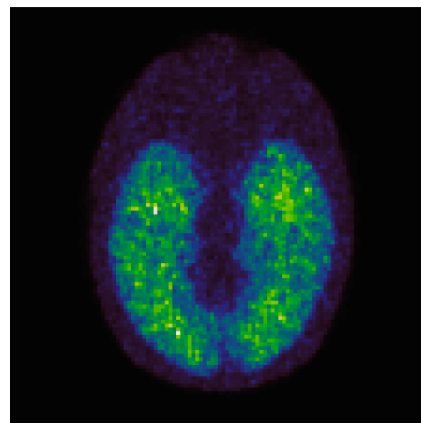


Figura 7: Riduciamo i conteggi nel sinogramma per enfatizzare l'effetto del prior

**MLEM**  
rec iter 30/30



**OSEM**  
rec iter 2/2  
subset 15/15



**MAP**  
rec iter 10/10  
subset 15/15

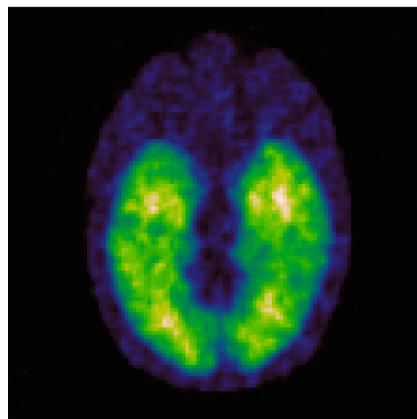


Figura 8: Ricostruzione MAP-OSL