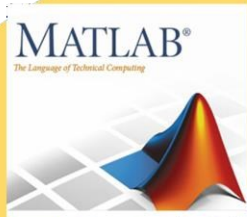


## Matlab: un fedele alleato

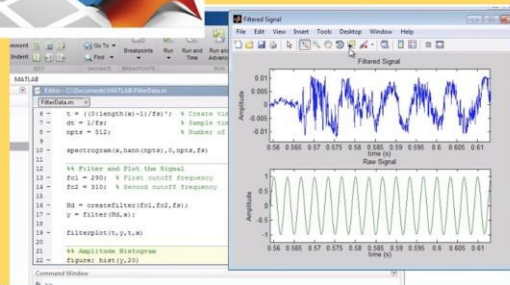


Cos'è Matlab?

- 1) un **ambiente ad alte prestazioni per il calcolo numerico**.
- 2) un **linguaggio** basato su espressioni che rende molto semplice la programmazione.

Per cosa è utile in ambito  
acquisizione e analisi segnali?

- ✓ Simulazione di segnali campionati
- ✓ Simulazione di aggiunta di rumore
  - ✓ Filtraggio digitale
  - ✓ Quantizzazione
  - ✓ Analisi in frequenza
- ✓ Acquisizione di segnali da scheda DAQ
- ✓ Generazione di segnali per scheda DAQ



# LAB. PRATICO MATLAB #1:

## Creazione di uno script MatLab per la generazione di un segnale semplice

*Misure e Acquisizione di Dati Biomedici*

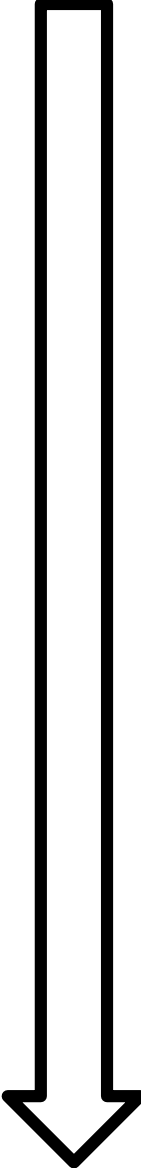
Sarah Tonello, PhD

Dipartimento di ingegneria dell'informazione

Università di Padova

# Obiettivi dell'esercitazione...

## IN PUNTI:

- 
1. definire i parametri della sinusoide;
  2. definire l'intervallo di osservazione  $T_w$ ;
  3. definire l'asse dei tempi  $t$ ;
  4. creare il segnale sinusoidale *continuo*;
  5. definire la frequenza di campionamento;
  6. calcolare l'asse dei tempi discretizzato;
  7. confrontare i due segnali sinusoidali;
  8. costruire il segnale quantizzato;
  9. determinare l'RMS;
  10. osservare le variazioni dell'RMS al variare della frequenza di campionamento;
  11. aggiungere rumore al segnale;
  12. confrontare il segnale rumoroso con quello non;
  13. calcolare SNR.

- Ricordarsi di **creare una sezione a parte in modo da avere sempre sottocchio i valori attribuiti ai parametri e poter testare il codice con diversi valori (%% nome sezione ...)**
- Commentare il codice sfruttando **%commento**
- **Attenzione alla definizione della finestra di osservazione  $T_w$ : la sua ampiezza andrà considerata in relazione alla frequenza della sinusoide, ottimale in modo da osservare qualche periodo del segnale (es.  $f_s=1000\text{Hz} \rightarrow t_s=1\text{ ms} \rightarrow T_w$  adeguate tra i 5 e i 10 ms)**

# Obiettivi dell'esercitazione...

## IN PUNTI:

1. definire i parametri della sinusoide;
2. definire l'intervallo di osservazione  $T_w$ ;
3. definire l'asse dei tempi  $t$ ;
4. creare il segnale sinusoidale *continuo*;
5. definire la frequenza di campionamento;
6. calcolare l'asse dei tempi discretizzato;
7. confrontare i due segnali sinusoidali;
8. costruire il segnale quantizzato;
9. determinare l'RMS;
10. osservare le variazioni dell'RMS al variare della frequenza di campionamento;
11. aggiungere rumore al segnale;
12. confrontare il segnale rumoroso con quello non;
13. calcolare SNR.

→ Queste indicazioni corrispondono a simulare la creazione di un segnale «continuo» poichè chiaramente ogni segnale generato con MatLab sarà sempre discreto in tempi e ampiezze.

→ In base alla frequenza della sinusoide e alla finestra di osservazione, prendendo un numero molto elevato di punti essi saranno così vicini da approssimare molto bene l'andamento continuo del segnale.

→ Si possono utilizzare entrambi in metodi visti a lezione in questa fase, chiaramente senza fare riferimento a nessuna frequenza di campionamento, ma stabilendo a priori il numero elevato di punti se si usa linspace o la distanza che si vuole ottenere se si usa il metodo con passo costante  $T_w/N$  con comando : (0:Tw/N:Tw)

(Provare con entrambi e notare le differenze in termini di distanza tra i vari campioni)

# Obiettivi dell'esercitazione...

## IN PUNTI:

1. definire i parametri della sinusoide;
2. definire l'intervallo di osservazione  $T_w$ ;
3. definire l'asse dei tempi  $t$ ;
4. creare il segnale sinusoidale *continuo*;
5. definire la frequenza di campionamento;
6. calcolare l'asse dei tempi discretizzato;
7. confrontare i due segnali sinusoidali;
8. costruire il segnale quantizzato;
9. determinare l'RMS;
10. osservare le variazioni dell'RMS al variare della frequenza di campionamento;
11. aggiungere rumore al segnale;
12. confrontare il segnale rumoroso con quello non;
13. calcolare SNR.

- Queste indicazioni corrispondono a simulare il campionamento compiuto dal sample and hold su un segnale sorgente continuo (quello creato al punto 4).
- **Prestare attenzione alla definizione della frequenza di campionamento mettendo in pratica quanto visto a lezione considerando le richieste da teorema di Nyquist unite alle non idealità**
- In base alla **frequenza di campionamento**, calcolare il **numero di campioni che potranno essere acquisiti nella finestra di campionamento  $T_w$** , e sfruttare questo numero per creare il vettore dei tempi.

## Consiglio:

Creare il vettore con il comando «:» piuttosto che con linspace in modo **da avere diretto controllo sulla distanza tra i punti in base alla frequenza di campionamento scelta**

# Obiettivi dell'esercitazione...

## IN PUNTI:

1. definire i parametri della sinusoide;
2. definire l'intervallo di osservazione  $T_w$ ;
3. definire l'asse dei tempi  $t$ ;
4. creare il segnale sinusoidale *continuo*;
5. definire la frequenza di campionamento;
6. calcolare l'asse dei tempi discretizzato;
7. confrontare i due segnali sinusoidali;
8. costruire il segnale quantizzato;
9. determinare l'RMS;
10. osservare le variazioni dell'RMS al variare della frequenza di campionamento;
11. aggiungere rumore al segnale;
12. confrontare il segnale rumoroso con quello non;
13. calcolare SNR.

→ Utilizzare il comando **plot** per realizzare il grafico di entrambe le sinusoidi (quella «continua» e quella campionata)

→ Sfruttare il comando «**hold on**» per sovrapporre entrambi i segnali, e valutare **se la frequenza di campionamento scelta consente un'approssimazione adeguata**

## Consiglio:

Provate a variare i parametri della sinusoide e la frequenza di campionamento, per confermare quanto visto a lezione in merito alla frequenza minima e a quella ottimale

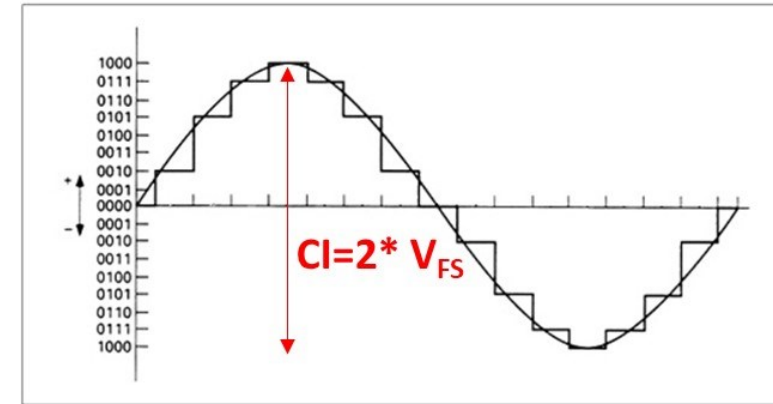
(es. provate  $f_s$  pari a 2, 3 e 5 volte la frequenza della vostra sinusoide)

# Obiettivi dell'esercitazione...

## IN PUNTI:

1. definire i parametri della sinusoide;
2. definire l'intervallo di osservazione  $T_w$ ;
3. definire l'asse dei tempi  $t$ ;
4. creare il segnale sinusoidale *continuo*;
5. definire la frequenza di campionamento;
6. calcolare l'asse dei tempi discretizzato;
7. confrontare i due segnali sinusoidali;
8. costruire il segnale quantizzato;
9. determinare l'RMS;
10. osservare le variazioni dell'RMS al variare della  $f_s$  di campionamento;
11. aggiungere rumore al segnale;
12. confrontare il segnale rumoroso con quello non;
13. calcolare SNR.

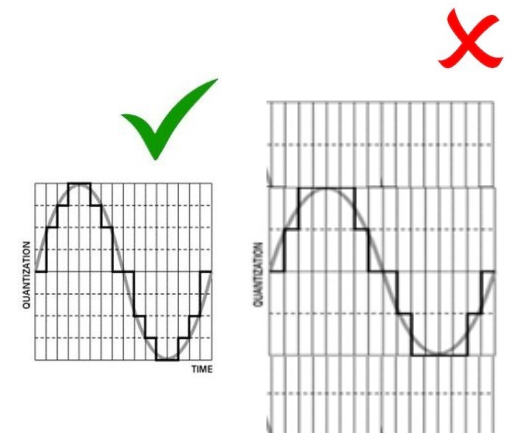
→ Questo punto corrisponde al passare da un segnale continuo nelle ampiezze, ad un segnale che può assumere un **numero limitato di valori di ampiezze, simulando quello che viene fatto da un ADC**



→ Per definire il numero di livelli disponibili sarà necessario stabilire i parametri del quantizzatore (**ADC simulato**), ovvero:

**N\_bit** → il numero di bit, ossia di **cifre binarie disponibili per la conversione** (es della figura 4 bit)

**V\_fs** → **valore di fondo scala**, ossia il valore di fondo scala del quantizzatore in base al quale verranno calcolati i livelli del quantizzatore (N.B. Ricordando quanto accennato nella parte di condizionamento del segnale, attenzione che per garantire la maggiore risoluzione possibile bisognerà fare in modo che il valore di fondo scala sia il più vicino possibile al valore di picco della sinusoide)



# Obiettivi dell'esercitazione...

## IN PUNTI:

1. definire i parametri della sinusoide;
2. definire l'intervallo di osservazione  $T_w$ ;
3. definire l'asse dei tempi  $t$ ;
4. creare il segnale sinusoidale *continuo*;
5. definire la frequenza di campionamento;
6. calcolare l'asse dei tempi discretizzato;
7. confrontare i due segnali sinusoidali;
8. costruire il segnale quantizzato;
9. determinare l'RMS;
10. osservare le variazioni dell'RMS al variare della frequenza;
11. aggiungere rumore al segnale;
12. confrontare il segnale rumoroso con quello non;
13. calcolare SNR.

Una volta noti i parametri del quantizzatore  $N_{bit}$  e  $V_{fs}$

→ il **numero dei livelli** sarà dato da  $N_{lev} = 2^{N_{bit}}$

→ l' **ampiezza del passo di quantizzazione può essere calcolata come:**

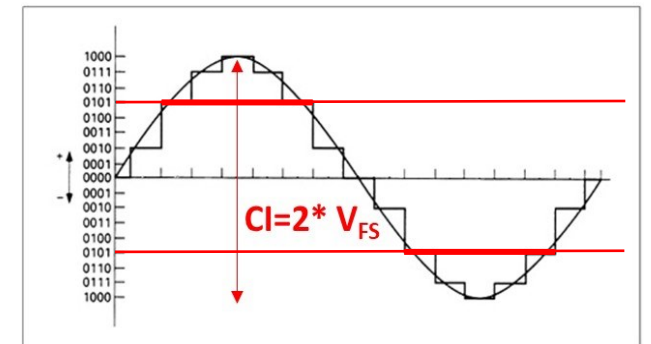
1)  $\Delta q = (2 * V_{fs}) / N_{lev}$  (come visto a lezione, accettando asimmetria tra le due parti)

oppure 2)  $\Delta q = (2 * V_{fs}) / (N_{lev} - 1)$  (questo permetterà di occupare con gli 8 livelli tutto il campo disponibile, ma non avremo livello corrispondente allo zero)

Una volta definite queste grandezze, è possibile calcolare per ciascun campione il **livello di quantizzazione**, arrotondando il rapporto tra valore non quantizzato e passo di quantizzazione all'intero più vicino con «round»:  **$q = \text{round}(x / \Delta q)$** ; **(perchè in questo caso stiamo realizzando quantizzatore per arrotondamento)**

→ Al di fuori del campo di funzionamento ( $x \geq V_{fs}$  e  $x \leq -V_{fs}$ ) è necessario **impostare le condizioni di saturazione**

$$\begin{aligned} \text{se } q(i) > 2^{N_{bit}-1} - 1 & \text{ allora } q(i) = 2^{N_{bit}-1} - 1 \\ \text{se } q(i) < -2^{N_{bit}-1} & \text{ allora } q(i) = -2^{N_{bit}-1} \end{aligned}$$



**Infine, il segnale quantizzato lo si ottiene:  $x_q = q * \Delta q$**

# Obiettivi dell'esercitazione...

## IN PUNTI:

1. definire i parametri della sinusoide;
2. definire l'intervallo di osservazione  $T_w$ ;
3. definire l'asse dei tempi  $t$ ;
4. creare il segnale sinusoidale *continuo*;
5. definire la frequenza di campionamento;
6. calcolare l'asse dei tempi discretizzato;
7. confrontare i due segnali sinusoidali;
8. costruire il segnale quantizzato;
9. determinare l'RMS;
10. osservare le variazioni dell'RMS al variare della frequenza di campionamento;
11. aggiungere rumore al segnale;
12. confrontare il segnale rumoroso con quello non;
13. calcolare SNR.

→ Dalla teoria, ci si ricorda che il **Valore efficace o RMS di un segnale** equivale al **valore che avrebbe un segnale costante di pari potenza media.**

→ Calcolare quindi il valore con la formula

$$V_{RMS} = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} v^2(nT_s)}$$

→ Calcolarlo per i) il segnale continuo, ii) il segnale campionato e iii) il segnale quantizzato. Confrontare i valori ottenuti con quello ottenuto su un segnale **continuo teorico ( $V_{eff}=V_{picco}/\sqrt{2}$ )**

## CONSIGLIO:

Provare a ripetere lo stesso calcolo modificando alcuni valori di frequenza di campionamento (es. 3 e 5 volte la frequenza massima della sinusoide)



# Obiettivi dell'esercitazione...

## IN PUNTI:

1. definire i parametri della sinusoide;
2. definire l'intervallo di osservazione  $T_w$ ;
3. definire l'asse dei tempi  $t$ ;
4. creare il segnale sinusoidale *continuo*;
5. definire la frequenza di campionamento;
6. calcolare l'asse dei tempi discretizzato;
7. confrontare i due segnali sinusoidali;
8. costruire il segnale quantizzato;
9. determinare l'RMS;
10. osservare le variazioni dell'RMS al variare della frequenza di campionamento;
11. aggiungere rumore al segnale;
12. confrontare il segnale rumoroso con quello non;
13. calcolare SNR.

→ Mettere in pratica quanto visto nell'introduzione in merito al SNR e alla possibilità di simulare aggiunta di rumore

→ Ricordando che per SNR espresso in forma lineare si intende  $SNR = V_{RMS,signal}^2 / V_{RMS,noise}^2$

una volta definito un SNR a piacere si avrà che

$$V_{RMS,noise}^2 = V_{RMS,signal}^2 / SNR$$

e quindi estraendo la radice quadrata di questo valore otterrò il  $V_{RMS,noise}$  utile per scalare il rumore.

→ Potrò quindi generare il rumore come distribuzione gaussiana come

$$noise = \text{sqrt}(V_{RMS,noise}^2) * \text{randn}(1, N\_sample)$$

e aggiungerlo al segnale

**CONSIGLIO:** Per confrontare il segnale rumoroso con quello no, oltre a mostrarli sullo stesso plot nel tempo, sfruttare il comando «**periodogram**» che fornisce un'indicazione immediata del contenuto in frequenza

# Obiettivi dell'esercitazione...

## IN PUNTI:

1. definire i parametri della sinusoide;
2. definire l'intervallo di osservazione  $T_w$ ;
3. definire l'asse dei tempi  $t$ ;
4. creare il segnale sinusoidale *continuo*;
5. definire la frequenza di campionamento;
6. calcolare l'asse dei tempi discretizzato;
7. confrontare i due segnali sinusoidali;
8. costruire il segnale quantizzato;
9. determinare l'RMS;
10. osservare le variazioni dell'RMS al variare della frequenza di campionamento;
11. aggiungere rumore al segnale;
12. confrontare il segnale rumoroso con quello non;
13. calcolare SNR.

→ Confrontare il valore teorico di SNR con quello ottenuto in realtà nel segnale rumoroso simulato sfruttando la funzione di MatLab «**snr**»  
**SNR\_mat=snr(Signal\_sampled\_noisy)** che restituisce il valore di SNR espresso in dB (attenzione a questo nel confrontare il risultato che si ottiene con quello teorico che si era imposto ma che era in forma lineare!)

$$\text{SNR dB} = 10 \cdot \log_{10}(\text{SNR})$$

# Scaricate la guida da Moodle e ...buon lavoro!!

## MATERIALE ESPERIENZE DI LABORATORI MATLAB

LABORATORIO INTRODUZIONE A MATLAB (7 NOVEMBRE)

Spunta come completato

LABORATORIO MATLAB 1.0 (14 novembre): Creazione di uno script MatLab per la generazione di un segnale semplice

## LABORATORIO MATLAB 1.0 (14 novembre): Creazione di uno script MatLab per la generazione di un segnale semplice

Guida alla prima esercitazione Matlab.pdf

Introduzione\_ese1\_matlab.pdf

Scarica cartella

Modifica