

INSEGNAMENTO DI ANALISI DI DATI BIOLOGICI
LM in BIOINGEGNERIA – A.A.2023/2024

HOMEWORK 2

COGNOME: CASARIN	NOME: ALESSANDRO	MATRICOLA: 2086857	DATA: 09/02/2024
e-mail: alessandro.casarin.4@studenti.unipd.it		allegati: casarin_h2_14.zip	

ESERCIZIO H2_14

Il vettore y_s fornito nel file `glucose_data_fm.mat` contiene campioni, relativi al vettore di tempi t_s , del segnale glicemico in un paziente diabetico di tipo 1 misurato da un sensore CGM. Le misure in y_s sono espresse in mg/dl, i tempi in t_s in min. Il rumore di misura è bianco, a media nulla e varianza $\sigma_v^2 = 9 \text{ mg}^2/\text{dl}^2$. Tramite deconvoluzione regolarizzata (da implementare, a scelta, con SVD o gradiente coniugato) ottenere, simultaneamente, una versione del segnale glicemico depurata dal rumore di misura e una stima della sua derivata prima. Calcolare le stime dei due segnali su una griglia virtuale a passo unitario. Per non fare assunzioni sulla condizione iniziale non nulla, usare la tecnica (suggerita nel Lab. 5) che realizza un burn in impostando l'inizio della griglia virtuale da un tempo negativo. Porre attenzione alla struttura delle matrici in gioco (es. quella di covarianza del rumore) al fine di gestire il codice in un modo ragionevolmente efficiente.

Suggerimento: impostare il problema di deconvoluzione osservando che un segnale causale è interpretabile come l'integrazione della sua derivata prima, e che, in termini modellistici, l'integrazione di un segnale è a sua volta interpretabile come l'uscita di un sistema LTI avente come ingresso la derivata prima del segnale e come risposta impulsiva il gradino.

SVOLGIMENTO

Introduzione e metodologia

Per la risoluzione del quesito, è necessario innanzitutto impostare il problema di deconvoluzione regolarizzata e definire la funzione di trasferimento del sistema LTI. Seguendo il suggerimento riportato nella consegna, la risposta impulsiva del sistema è il gradino unitario e quindi la sua funzione di trasferimento sarà la seguente:

$$G(z) = \frac{1}{1 - z^{-1}}$$

Quindi l'ingresso del sistema LTI sarà un segnale u , da stimare, rappresentante la derivata prima del segnale glicemico; mentre l'uscita sarà una versione del segnale glicemico depurata dal suo rumore di misura, questa verrà calcolata tramite riconvoluzione del segnale stimato \hat{u} .

Dal momento che il segnale y_s non è causale (il glucosio è presente nell'organismo in qualsiasi momento), se si applicasse l'algoritmo fornendo in ingresso i dati campionati senza alcun accorgimento, si otterrebbe un segnale che al tempo $t = 0$ indica una concentrazione nulla. Per risolvere questo problema, si realizza un burn in iniziale traslando il vettore dei tempi di campionamento di una certa quantità e si costruisce la griglia virtuale a partire dal nuovo vettore dei tempi, in modo da avere nel primo segmento temporale un certo numero di istanti temporali svincolati dal segnale campionato, che svolgono la funzione di innesco per il segnale approssimante.

La deconvoluzione regolarizzata è stata implementata con il metodo del gradiente coniugato, per ogni valore di γ "candidato" viene generata una successione convergente in al più N iterazioni, dove N è la dimensione della griglia virtuale.

Per determinare γ^0 tra tutti i valori "candidati" è stato utilizzato il criterio di consistenza 1.

Presentazione del codice

Per realizzare quanto richiesto, ho realizzato un M-file, *h2_14.m*, disponibile nell'archivio *casarin_h2_14.zip*.

Una volta realizzato il burn in iniziale aggiungendo una quantità Tburn alla griglia di campionamento, è stata creata la griglia virtuale con passo di 1 minuto e sono stati mappati i campioni virtualmente mancanti.

A partire dalla funzione di trasferimento riportata nel paragrafo precedente, è stata creata la matrice G_v e successivamente la matrice G.

È stata poi creata la matrice F relativa alle differenze con ordine $m = 2$ e le matrici/vettori relativi alla varianza dell'errore di misura v.

A questo punto, creati tutti i vettori e le matrici necessarie per l'algoritmo, si procede con la deconvoluzione regolarizzata impostando un range (molto grande) nel quale si è certi essere contenuto il valore ottimo per γ , tramite il metodo di bisezione questo range verrà ridotto ad ogni iterazione fino ad ottenere un valore molto vicino a quello desiderato. Per ogni iterazione viene applicato il metodo del gradiente coniugato per risolvere il sistema $Qx = b$, dove $Q = (G^T B^{-1} G + \gamma F^T F)$, mentre $b = G^T B^{-1} y$. Dopo N vettori della successione si arriva ad un vettore x che rappresenta la stima \hat{u} , questo viene utilizzato per calcolare i parametri utilizzati nel criterio di consistenza 1 e modificare il limite superiore o inferiore del range di valori di γ "candidati". L'algoritmo generico del gradiente coniugato è schematizzato in Figura 1.

```

 $x_0 = \text{vettore iniziale composto da zeri}$ 
 $r_0 = b - Qx_0$ 
 $p_0 = r_0$ 
for  $k = 0, \dots, N$ 
     $\alpha_k = \frac{p_k^T r_k}{p_k^T Q p_k}$ 
     $x_{k+1} = x_k + \alpha_k p_k$ 
     $r_{k+1} = r_k - \alpha_k Q p_k$ 
     $\beta_k = -\frac{r_{k+1}^T Q p_k}{p_k^T Q p_k}$ 
     $p_{k+1} = r_{k+1} + \beta_k p_k$ 
     $k = k + 1$ 
end

```

Figura 1: schema generale per la soluzione mediante metodo del gradiente coniugato, la successione converge in al più N iterazioni (N è la dimensione della griglia virtuale).

In questo caso specifico, la griglia virtuale è uniforme e il sistema è LTI, questo significa che F è di Toeplitz e G è quasi di Toeplitz; inoltre la funzione di trasferimento del sistema è razionale. È possibile sfruttare queste informazioni per rendere l'algoritmo del gradiente coniugato più efficiente, riducendo la complessità di un'iterazione a $O(N)$ operazioni e quindi complessivamente $O(N^2)$, per ogni valore di γ .

Nel dettaglio:

- Moltiplicazioni del tipo Fx possono essere svolte facendo la convoluzione tra la prima colonna della matrice F e il vettore x .
- Moltiplicazioni del tipo $F^T x$ possono essere svolte facendo la convoluzione tra la prima colonna della matrice F e il vettore x ribaltato, ribaltando infine il vettore risultante.
- Moltiplicazioni del tipo $B^{-1}x$ possono essere svolte moltiplicando i valori sulla diagonale della matrice B^{-1} per il vettore x , essendo B una matrice diagonale.
- Moltiplicazioni del tipo Gx possono essere svolte calcolando il prodotto $G_v x$ e rimuovendo successivamente le righe relative ai campioni virtualmente mancanti.
- Moltiplicazioni del tipo $G^T x$ possono essere svolte “riempiendo” il vettore x di zeri nelle posizioni relative ai campioni virtualmente mancanti e calcolando il prodotto tra G_v e il vettore x ribaltato, infine ribaltando il vettore risultante.

Infine, una volta trovata la stima \hat{u} calcolata con γ^0 , questa viene utilizzata per fare una riconvoluzione ed ottenere una predizione sul segnale glicemico ripulito dal rumore di misura.

Risultati e discussione

Inizialmente, sono stati plottati i campioni di glucosio acquisiti in funzione del tempo. Come visibile in Figura 2, il segnale è affetto da rumore di misura.

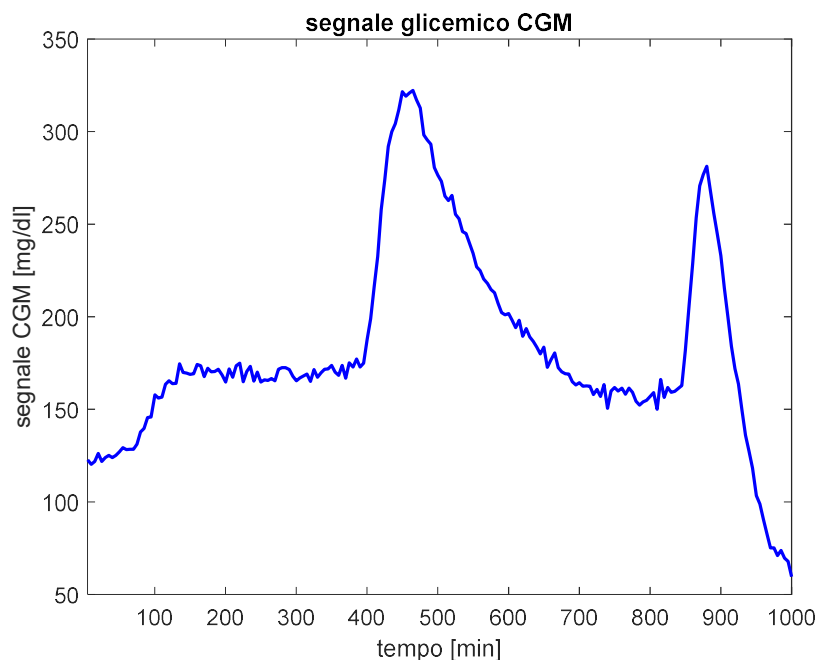


Figura 2: segnale glicemico in un paziente affetto da diabete di tipo 1 misurato da un sensore CGM.

I risultati ottenuti dopo la deconvoluzione regolarizzata sono soddisfacenti, il segnale glicemico stimato è molto più regolare rispetto alla versione con rumore, i residui normalizzati sono in linea con quanto atteso e la stima della derivata prima è corretta. I segnali stimati sono visibili in Figura 3 e Figura 4.

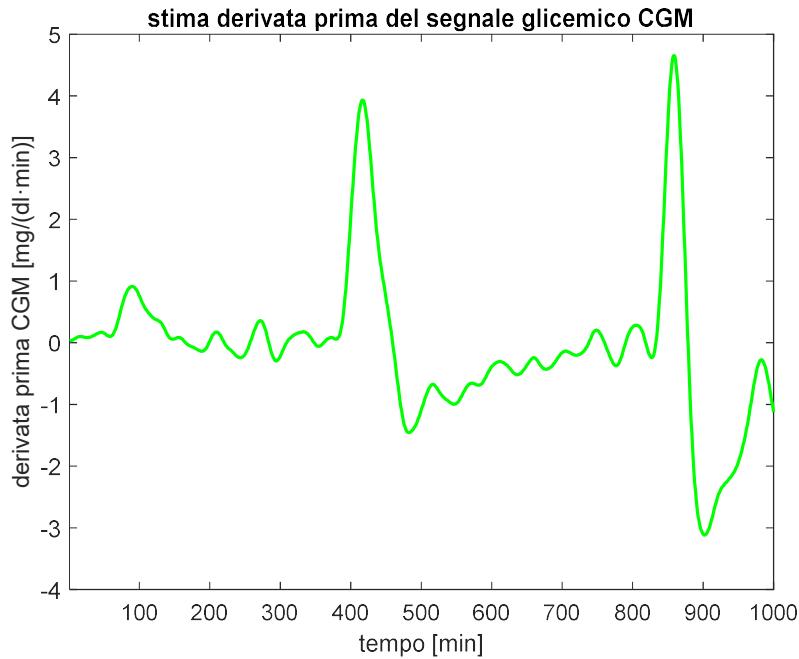


Figura 3: stima della derivata prima del segnale glicemico.

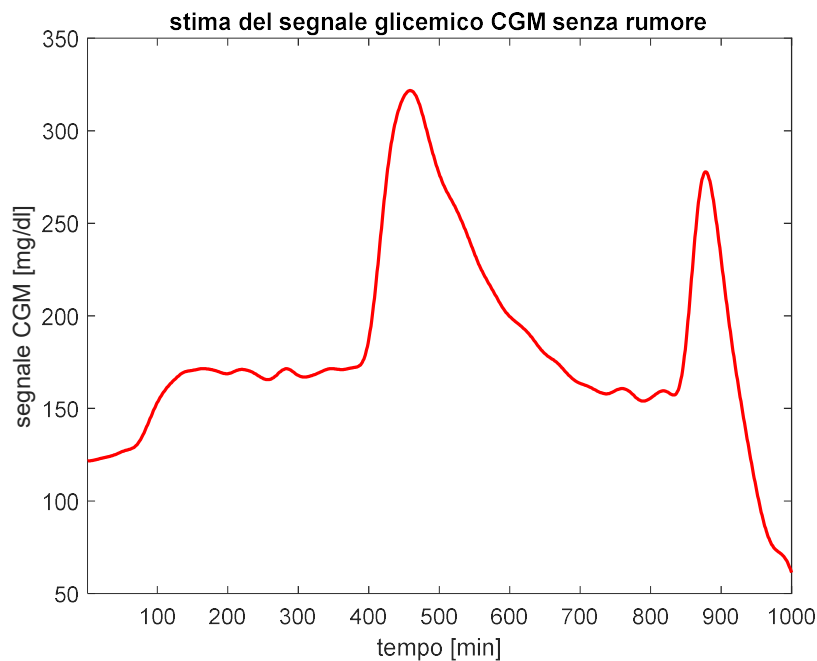


Figura 4: stima del segnale glicemico ottenuto come riconvoluzione del segnale stimato \hat{u} corrispondente alla derivata prima del segnale glicemico.

Il valore scelto per m è 2, quindi la matrice F è relativa alle differenze seconde. Altri valori di m sono stati testati e hanno fornito risultati peggiori rispetto a questi.

È importante fare una precisazione riguardo la convergenza della successione generata dal metodo del gradiente coniugato. Precedentemente è stato scritto che questo algoritmo converge sicuramente in N iterazioni, questo è vero in aritmetica esatta ma nella realtà non è così. Per rimediare alla non-idealità delle operazioni al calcolatore, tipicamente si lascia operare l'algoritmo fino a $3N$ iterazioni e si fissa nel codice un criterio d'arresto legato alla norma di r_k . In questo caso ho preferito fare N cicli ed "accontentarmi" del vettore finale x_N come soluzione per alleggerire il codice e rendere il programma più rapido.

Come ultimo passaggio prima di plottare i risultati è importante ricordarsi di rimuovere il burn in iniziale riducendo la dimensione della griglia virtuale e sottraendo da entrambe le griglie temporali la quantità T_{burn} che era stata aggiunta inizialmente.

I “take-home messages” da evidenziare sono i seguenti:

- In questo tipo di analisi in cui i campioni da stimare sono molti (2000 in questo caso) è importante utilizzare delle accortezze sulla struttura delle matrici in gioco in modo da rendere il codice più efficiente.
- Nonostante il costo computazionale sia stato ridotto, il programma risulta comunque pesante perché N è elevato. Diventa quindi necessario ridurre il numero di iterazioni quanto basta per ottenere dei risultati soddisfacenti in un tempo contenuto.