Algoritmo di Gauss

Sistemi lineari

Per ogni matrice $M \in \mathbb{K}^{s,t}$ poniamo

$$\ker(M) := \{X \in \mathbb{K}^{s,1} : MX = \mathbf{0}\}.$$

In particolare il $\ker(M)$ rappresenta lo spazio delle soluzioni del sistema lineare omogeneo $MX = \mathbf{0}$.

Come notazione, fissiamo una volta per tutte $A \in \mathbb{K}^{m,n}$, $B \in \mathbb{K}^{m,1}$ ed $X = {}^T(x_1, \dots, x_n)$.

Lemma 1. *Sia $AX = B$ un sistema lineare. Allora X_0 è soluzione di $AX = B$ se e solamente se $\begin{pmatrix} X_0 \\ -1 \end{pmatrix}$ è soluzione del sistema lineare $(A|B)\overline{X} = \mathbf{0}$ ove $\overline{X} = {}^T(x_1, \dots, x_n, x_{n+1})$. In particolare,*

$$\mathscr{S} = \{X \in \mathbb{K}^n \colon AX = B\} = \\ \{(x_1, \dots, x_n) : (x_1, \dots, x_n, x_{n+1}) \in \ker(A|B) \cap \{\overline{X} = (x_1, \dots, x_{n+1}) : x_{n+1} = -1\}\}.$$

Dimostrazione. La dimostrazione è immediata e si ottiene riscrivendo il sistema come $AX - B = \mathbf{0}$ che è equivalente a $(A|B)\overline{X} = \mathbf{0}$. □

Lemma 2. *Sia $M \in \mathbb{K}^{r,s}$ una matrice siano $R_1, \dots, R_s$ le sue righe. Allora* $\ker(M)$ *dipende solamente dalla copertura lineare $\mathscr{L}(R_1, \dots, R_s)$. Tale spazio vettoriale è detto* $\mathrm{coker}(M)$.

Dimostrazione. Scriviamo $M = \begin{pmatrix} R_1 \\ R_2 \\ \vdots \\ R_s \end{pmatrix}$. Sappiamo che $X \in \ker(M)$ se e solamente se $R_1X = R_2X = \dots = R_sX = 0$. Mostriamo che per ogni combinazione lineare delle righe $\sum_{i=1}^{s} \alpha_i R_i$ si ha anche $(\sum_{i=1}^{s} \alpha_i R_i)X = \sum_{i=1}^{s} \alpha_i (R_i X) = 0$. In particolare $RX = 0$ per ogni $R \in \mathrm{coker}(M)$ e dunque

$$\ker(M) \subseteq \{X \mid \forall R \in \mathrm{coker}(M) : RX = 0\}.$$

Viceversa: se $RX = 0$ per ogni $R \in \mathrm{coker}(M)$, allora anche $R_iX = 0$ per $i = 1, \dots, s$ visto che le righe R_i generano $\mathrm{coker}(M)$ e dunque sono in esso contenute. In particolare

$$\{X \mid \forall R \in \mathrm{coker}(M) : RX = 0\} \subseteq \ker(M).$$

Queste due condizioni danno l'uguaglianza. □

Il sottospazio $\mathrm{coker}(M) \leq \mathbb{K}^{1,n}$ rappresenta le equazioni che devono essere soddisfatte dai vettori di $\ker(M) \in \mathbb{K}^{n,1}$. In particolare se $\ker(M)$ consta di vettori *colonna*, allora $\mathrm{coker}(M)$ è un sottospazio di vettori *riga*.

Corollario 1. *Supponiamo che $AX = B$ sia un sistema lineare compatibile. Allora le soluzioni di $AX = B$ dipendono solamente da* $\mathrm{coker}(A|B)$.

Dimostrazione. Per il Lemma 1 le soluzioni di $AX = B$ si ottengono come le prime n componenti dei vettori che si trovano nell'intersezione di $\ker(A|B)$ con il sottoinsieme di equazione $x_{n+1} = -1$. Per il Lemma 2 $\ker(A|B)$ è univocamente determinato da $\mathrm{coker}(A|B)$. Segue la tesi. □

Il seguente teorema è il risultato principale che ci interessa. Esso asserisce che le soluzioni di un sistema lineare (compatibile) dipendono solamente dallo spazio vettoriale generato dalle righe della matrice completa. Conseguentemente, è possibile risolvere il sistema semplicemente considerando le soluzioni di un sistema $A'X = B'$ in cui le righe di $(A'|B')$ sono una base opportuna per $\mathrm{coker}(A|B)$.

Teorema 1 (Sistema principale equivalente). *Supponiamo che $R_1, \dots, R_k$ siano righe che formano una base dello spazio delle righe della matrice completa $(A|B)$ associata ad un sistema lineare compatibile $AX = B$. Allora il sistema dato è equivalente al sistema $A'X = B'$ ove $(A'|B')$ è la matrice formata dalle sole righe $R_1, \dots, R_k$.*

Dimostrazione. È conseguenza diretta del Corollario 1, visto che $R_1, \dots, R_k$ generano coker$(A|B)$. □

Un sistema $A'X = B'$ ottenuto nel teorema precedente è detto *sistema principale equivalente* ad $AX = B$.

Forma RREF

L'algoritmo di Gauss per la risoluzione dei sistemi lineari consiste nel trovare una base particolarmente semplice per coker$(A|B)$, da cui sia immediato ricostruire le soluzioni desiderate. A tal fine definiamo innanzi tutto di quali proprietà deve godere tale base.

Definizione 1. *Una matrice $M = ((m_{ij})) \in \mathbb{K}^{m,n}$ è detta* in forma ridotta per righe a scalini *(RREF) se, posto $i_s(M) := \min\{\{\infty\} \cup \{j : m_{sj} \neq 0\}\}$*

1. *Per ogni riga s, la prima entrata non nulla di ogni riga è 1; ovvero, $i_s(M) < \infty$ implica $m_{sj} = 1$.*
2. *La colonna che contiene la prima entrata non nulla di una riga di M' contiene solamente un elemento uguale ad 1 e tutti gli altri sono 0;*
3. *Per ogni coppia di righe R_s e R_t si ha che $s < t$ implica $i_s(M) < i_t(M)$.*

Se la matrice $(A'|B') \in \mathbb{K}^{k,n+1}$ è in forma RREF, allora è possibile leggere le soluzioni del sistema associato come segue:

1. Se $(A'|B')$ contiene una riga la cui unica entrata non nulla è nella colonna $n + 1$, allora il sistema contiene una equazione equivalente a $0 = 1$ ed è dunque incompatibile.
2. Supponiamo ora che il sistema sia compatibile. Consideriamo come *incognite* tutte le indeterminate che corrispondono ai *coefficienti iniziali non nulli* delle righe; le restanti indeterminate le trattiamo come parametri.
3. Spostiamo i parametri a destra del segno di uguale insieme al vettore dei termini noti. Questo, al variare in tutti i modi possibili dei parametri in $\mathbb{K}$ fornisce tutte le soluzioni.

Descriviamo ora nel dettaglio la procedura di eliminazione gaussiana.

Teorema 2 (Eliminazione gaussiana). *Data una matrice $M = ((m_{ij})) \in \mathbb{K}^{r,s}$ è sempre possibile ottenere manipolando le righe di M matrice $M' = ((m'_{ij})) \in \mathbb{K}^{r,s}$ in forma RREF, tale che* coker(M') = coker(M).

Dimostrazione. A meno di permutare le righe possiamo supporre che per tutti gli $j > 1$ si abbia $i_j(M) \geq i_1(M)$. A questo punto dividiamo la prima riga R_1 per m_{1i_1}, ponendo

$$R'_1 \leftarrow m_{1i_1}^{-1} R_1$$

e per ogni riga R_j con $j \neq 1$ poniamo

$$R'_j \leftarrow R_j - m_{ji_1} R'_1.$$

Otteniamo in questo modo una matrice M'. Per ogni riga successiva la prima procediamo in modo analogo; in particolare, a meno di permutare le righe $R'_2 \dots, R'_r$ possiamo supporre che per tutti gli $j > 2$ si abbia $i_j(M') \geq i_2(M')$. Come prima dividiamo la seconda riga per m'_{2i_2} e poi sottraiamo da tutte le rimanenti con $j \neq 2$ tale riga divisa moltiplicata per m'_{ji_2}. Si continua in questo modo sino a che non si arriva all'ultima riga. □

```
"Riduzione in forma RREF di una matrice"
function RREF(m; sort=true)
    n=copy(m)
    rows,cols=size(n)
    # Per ogni indice di riga
    for i = 1:rows
        # Determinare la prima entrata non nulla
        k=findfirst(x->x!=0,n[i,:])
        # Se risulta =0, passare alla riga successiva
        k nothing && continue
        # Normalizzare la riga
        n[i,:] /= n[i,k]
        # Per ogni altra riga fare in modo
        # che le entrate in posizione k siano 0
        for j in 1:rows
            i==j && continue
            n[j,:]-=n[i,:]*n[j,k]
        end
    end
    # Riordinare le righe (opzionale)
    sort && (n=sortslices(n,dims=1,rev=true))
    n
end
```

Algoritmo di risoluzione con metodo di Gauss

Dato un sistema lineare $AX = B$:

1. Ridurre la matrice $(A|B)$ in forma RREF (non è indispensabile riordinare le righe).
2. Osservare che se la matrice $(A|B)$ contiene una riga in cui l'unico coeff. non nullo corrisponde alla posizione $n + 1$, allora il sistema non è compatibile;
3. Se il sistema è compatibile, eliminare dalla matrice $(A|B)$ le righe nulle. Si ottiene in questo modo un sistema lineare equivalente a quello iniziale descritto da una matrice $(A'|B')$.
4. Interpretare le incognite che *non* corrispondono a coefficienti $i_s(A'|B')$ come parametri.
5. Riscrivere il sistema aggiungendo per ogni parametro x_t una equazione del tipo $x_t = x_t$, ottenendo ora un nuovo sistema $(A''|B'')$ di n equazioni in n incognite e con rango n.
6. Osservare che il vettore $-B''$ corrisponde ad una soluzione particolare del sistema. I vettori che corrispondono alle colonne dei parametri invece sono una base dello spazio vettoriale $\ker(A)$ delle soluzioni del sistema lineare omogeneo associato.

Osserviamo che in pratica tutte le informazioni necessarie a costruire le soluzioni di un sistema lineare $AX = B$ sono facilmente deducibili dalla matrice `RREF`$(A|B)$. La funzione seguente riscrive questi dati restituendo una soluzione particolare `sol` ed una base per lo spazio delle soluzioni del sistema omogeneo associato $AX = 0$.

```
"""
Stampa/restituisce soluzioni di un sistema lineare.
Il parametro n deve essere la riduzione in forma RREF della
matrice completa del sistema.
"""
```

```
function printsolution(n)
    rows,cols = size(n)
    sol=zeros(cols-1)
    gen=zeros(cols-1,cols-1)
    fix=[]
    for i = 1:rows
        k=findall(x->x!=0,n[i,:])
        isempty(k) && continue
        if first(k)==cols
            print("Sistema non compatibile\n")
            return false,false
        end
        append!(fix,first(k))
        print("x(",first(k),")=",n[i,cols])
        sol[first(k)]=n[i,cols]
        for j in k[2:end]
            if j<cols
                print("+",-n[i,j],"*x(",j,")")
                gen[first(k),j]=-n[i,j]
            end
        end
        print("\n")
    end

    for i = 1:cols-1
        i in fix && continue
        print("x(",i,")=x(",i,")\n")
        gen[i,i]=1
        sol[i]=0
    end
    gen = hcat([ x for x in eachcol(gen) if !iszero(x) ]...)
    print("\n")
    return transpose(sol), transpose(gen)
end
```

Il programma seguente si utilizza per verificare le soluzioni ottenute sopra.

```
"""
Verifica soluzioni

m=matrice (A|B) completa del sistema
s=soluzione particolare
gen=generatori dello spazio delle soluzioni sistema omogeneo AX=0
"""
function checksolution(m,s,gen)
    rows, cols = size(m)
    print("Matrice incompleta A:\n")
    display(m[:,1:cols-1])
    print("\n Termini noti B:\n")
    display(m[:,cols])
    print("Soluzione particolare S:\n")
    display(s)
    print("Generatori ker(A):\n")
    isempty(gen) && (gen=zeros(cols-1)')
    display(gen)
    r0 = m[:,1:cols-1]*transpose(s)-m[:,cols]
    print("AS-B=\n")
    display(r0)
    print("Verifica ker:")
```

```
    r1=m[:,1:cols-1]*transpose(gen)
    print("AZ=\n")
    display(r1)
    r0,r1
end
```