



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

Dipartimento di Matematica

Laboratorio di Calcolo Numerico  
Laboratorio 12: Operazioni matrice-vettore.  
Iterazioni di Jacobi e Gauss-Seidel

Damiano Pasetto

E-mail: [pasetto@math.unipd.it](mailto:pasetto@math.unipd.it)

Dispense: [http://dispense.dmsa.unipd.it/putti/calcolo\\_ambientale/index.html](http://dispense.dmsa.unipd.it/putti/calcolo_ambientale/index.html)

5 Giugno 2014

A matrice di dimensione  $n \times m$  (Amat),  $b$  vettore di dimensione  $n$  (bvet),  
 $x$  vettore di dimensione  $m$  (xvet),  $y$  vettore di dimensione  $m$  (yvet)

## Letture e stampa di matrici e vettori

- Lettura e stampa del vettore  $x$  (xvet):

```
read(iunit,*) (xvet(i),i=1,m) | write(ounit,*) (xvet(i),i=1,m)
```

- Lettura e stampa della matrice Amat:

```
do i=1,n  
  read(iunit,*) (Amat(i,j),j=1,m)  
end do | do i=1,n  
  write(ounit,*) (Amat(i,j),j=1,m)  
end do
```

## Prodotto vettore-vettore

Prodotto scalare,  $c = x^T y = \sum_{j=1}^m x_j y_j$ :

```
sum=0.0
do j=1,m
    sum=sum+xvet(j)*yvet(j)
end do
c=sum
```

## Prodotto matrice-vettore

Prodotto matrice-vettore,  $b = Ax$  (in componenti:  $b_i = \sum_{j=1}^m A_{i,j} x_j$ )

```
do i=1,n
    bvet(i)=0.0
    do j=1,m
        bvet(i)=bvet(i)+Amat(i,j)*xvet(j)
    end do
end do
```

## Prodotto matrice-matrice

Prodotto matrice-matrice,  $C = AB$ , dove  $A$  matrice  $n \times m$  e  $B$  matrice  $m \times l$  (in componenti:  $c_{i,j} = \sum_{k=1}^m A_{i,k}B_{k,j}$ )

```
do i=1,n
  do j=1,l
    Cmat(i,j)=0.0
    do k=1,m
      Cmat(i,j)=Cmat(i,j)+Amat(i,k)*Bmat(k,j)
    end do
  end do
end do
```

## Note di implementazione dell'algoritmo di Jacobi

Iterazione di Jacobi per risolvere un sistema lineare  $Ax = b$ :

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{j=1}^{i-1} a_{i,j} x_j^{(k)} - \sum_{j=i+1}^n a_{i,j} x_j^{(k)} \right)$$

Condizione di terminazione:  $\|x^{(k+1)} - x^{(k)}\| < \text{TOLL}$

```
do while( scarto<TOLL and iter<IMAX)
```

```
1. k←k+1
```

```
2. do i=1,n
```

```
   (a) som1=  $\sum_{j=1}^{i-1} a_{i,j} x_j^k$ 
```

```
   (b) som2=  $\sum_{j=i+1}^n a_{i,j} x_j^k$ 
```

```
   (c)  $x_i^{(k+1)} = \frac{1}{a_{ii}} (b_i - \text{som1} - \text{som2})$ 
```

```
3. end do
```

```
4. scarto= $\|x^{(k+1)} - x^{(k)}\|$ 
```

```
5.  $x^{(k)} \leftarrow x^{(k+1)}$ 
```

```
end do
```

## Note di implementazione dell'algoritmo di Gauss-Seidel

Iterazione di Gauss-Seidel per risolvere un sistema lineare  $Ax = b$ :

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{j=1}^{i-1} a_{i,j} x_j^{(k+1)} - \sum_{j=i+1}^n a_{i,j} x_j^{(k)} \right)$$

Condizione di terminazione:  $\|x^{(k+1)} - x^{(k)}\| < \text{TOLL}$

```
do while( scarto<TOLL and iter<IMAX)
```

```
1. k←k+1
```

```
2. do i=1,n
```

```
(a) som1=  $\sum_{j=1}^{i-1} a_{i,j} x_j^{k+1}$ 
```

```
(b) som2=  $\sum_{j=i+1}^n a_{i,j} x_j^k$ 
```

```
(c)  $x_i^{(k+1)} = \frac{1}{a_{ii}} (b_i - \text{som1} - \text{som2})$ 
```

```
3. end do
```

```
4. scarto= $\|x^{(k+1)} - x^{(k)}\|$ 
```

```
5.  $x^{(k)} \leftarrow x^{(k+1)}$ 
```

```
end do
```