Matrici sparse

Si dice che una matrice è sparsa se ha *tanti* coefficienti uguali a zero. (Una matrice tridiagonale è un esempio di matrice sparsa).

Se una matrice è sparsa è conveniente memorizare solo dove si trovano gli elementi non nulli e il loro valore.

- Il comando `sparse` di Matlab memoriza una matrice in questo modo.
- Il comando `nnz` restituisce il numero di elementi non nulli in una matrice.
- il comando `spy` *disegna* la struttura di una matrice.

I metodi diretti "riempiono" le matrici sparse

Metodi iterativi classici

$$A = P - N \qquad P \textit{ facilmente} \text{ invertibile}$$

$$A\mathbf{x} = \mathbf{b} \qquad P\mathbf{x} = \mathbf{b} + N\mathbf{x} \qquad \mathbf{x} = P^{-1}N\mathbf{x} + P^{-1}\mathbf{b}$$

$$A\mathbf{x} = \mathbf{b} \Leftrightarrow \mathbf{x} = B\mathbf{x} + \mathbf{c}$$

Il metodo iterativo costruisce una sucessione di vettori $\{\mathbf{x}^{(k)}\}$ nel seguente modo

$\mathbf{x}^{(0)}$ assegnato
Per $k \geq 0$
$\quad \mathbf{x}^{(k+1)} = B\mathbf{x}^{(k)} + \mathbf{c}$

Se questa succession converge a un limite (che chiamiamo $\mathbf{x}^{\infty}$) questo limite è la soluzione del sistema lineare

$$A\mathbf{x}^{\infty} = \mathbf{b}$$

Esempi di metodi iterativi classici

Metodo di Jacobi `P=diag(diag(A))`

$$\text{for } i = 1 : n$$

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

Metodo di Gauss-Seidel `P=tril(A)`

$$\text{for } i = 1 : n$$

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

Esercicio

- Escrivere una funzione di Matlab che implementi il metodo di Jacobi. Fermare l'iterazioni quando $\|\mathbf{r}^{(k)}\| < toll * \|\mathbf{b}\|$.

  (`norm(X,p)` "Matrix or vector norm.")
- Data la matrice simmetrica definita positiva $A = M * M'$ con

$$M = \begin{bmatrix} 1 & 0 & -1 & 1 & 2 \\ -2 & 3 & 1 & 2 & 1 \\ 1 & 0 & 0 & -2 & 3 \\ 1 & 2 & 3 & -1 & 0 \\ -1 & -2 & -1 & -1 & 3 \end{bmatrix}$$

  usando i comandi di Matlab calcolare il raggio spettrale della matrice d'iterazione del metodo di Jacobi e del metodo di Gauss-Seidel.

Numero di condizionamento

`cond(X,p)`	"Condition number" `norm(X,p) * norm(inv(X),p)`
`hilb(n)`	"Hilbert matrix" $h_{i,j} = 1/(i+j-1)$.

<u>Esercizio</u>

Sia A la matrice di Hilbert 8×8 sia $\mathbf{b} = A * ones(8,1)$ e $\mathbf{c} = \mathbf{b} + 1.e - 10 * rand(8,1)$.

- Usando i comandi di Matlab risolvere $A\mathbf{x} = \mathbf{b}$ e $A\mathbf{y} = \mathbf{c}$.
- Calcolare $\frac{\|\mathbf{x}-\mathbf{y}\|}{\|\mathbf{x}\|}$ e confrontare con $\frac{\|\mathbf{b}-\mathbf{c}\|}{\|\mathbf{b}\|}$.
- Calcolare il numero di condizionamento di A.
- Commentare i risultati.

Metodo del gradiente

$$A\mathbf{x} = \mathbf{b}$$

con A una matrice simmetrica definita positiva

$\mathbf{x}^{(0)}$ assegnato
for $k \geq 0$

$$\mathbf{r}^{(k)} = b - A\mathbf{x}^{(k)}$$

$$\alpha_k = \frac{[\mathbf{r}^{(k)}]^T \mathbf{r}^{(k)}}{[\mathbf{r}^{(k)}]^T A\mathbf{r}^{(k)}}$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{r}^{(k)}$$

Metodo del gradiente coniugato

$\mathbf{x}^{(0)}$ assegnato, $\quad \mathbf{r}^{(0)} = b - A\mathbf{x}^{(0)}$, $\quad \mathbf{d}^{(0)} = \mathbf{r}^{(0)}$,

for $k \geq 0$

$$\alpha_k = \frac{[\mathbf{d}^{(k)}]^T \mathbf{r}^{(k)}}{[\mathbf{d}^{(k)}]^T A \mathbf{d}^{(k)}}$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$$

$$\mathbf{r}^{(k+1)} = b - A\mathbf{x}^{(k+1)}$$

$$\beta_k = -\frac{[\mathbf{d}^{(k)}]^T A \mathbf{r}^{(k+1)}}{[\mathbf{d}^{(k)}]^T A \mathbf{d}^{(k)}}$$

$$\mathbf{d}^{(k+1)} = \mathbf{r}^{(k+1)} + \beta_k \mathbf{d}^{(k)}$$