

MCS 2020

Algebra lineare numerica

Sistemi lineari con matrici sparse simmetriche e definite positive

Silva Edoardo 816560, Zhigui Bryan 816335, Marchetti Davide 815990

21/05/2020

Abstract

Lo scopo di questo progetto è di studiare l'implementazione in ambienti di programmazione open source del metodo di Choleski per la risoluzione sistemi lineari per matrici sparse, simmetriche e definite positive, e di confrontarli con l'implementazione di MATLAB.

Il confronto avverrà in termini di tempo, accuratezza, impiego della memoria e anche facilità d'uso tramite documentazione sia in ambiente Linux che Windows, eseguendo il codice con input **SuiteSparse Matrix Collection** che colleziona matrici sparse derivanti da applicazioni di problemi reali.

1 Librerie

1.1 descrizione libreria matlab

cholesky.matlab chol: libreria ufficiale MatLab per la fattorizzazione di Choleski.

Syntax:

- **$R = \text{chol}(A, [\text{triangle}])$:** Fattorizza la matrice simmetrica A definita positiva in una matrice triangolare R tale che $A=R'*R$;

R di default è 'upper', ma può essere specificata diversamente con $R = \text{chol}(A, \text{'lower'})$. se A è asimmetrica, allora chol la tratta come se fosse simmetrica prendendo solo la triangolare superiore.

- **[R,flag] = chol():** ritorna in output 'flag' che indica se A è simmetrica e definita positiva. Inoltre calcola R, ma non ritorna errori; flag = 0 -> la fattorizzazione è avvenuta con successo, altrimenti indica il punto di fallimento.
- **[R,flag,P] = chol(S, [outputForm]):** come [R,flag] = chol(), ma ritorna anche la matrice di permutazione P, ossia il preordinamento della matrice S obtained by amd. se flag = 0, allora S è simmetrica e definita positiva e R è la matrice triangolare superiore $R^*R = P^*S^*P$. outputForm è specificabile per ritornare P come matrice o vettore ('matrix' di default).

1.1.1 manutentore

Rilasciata con l'aggiornamento R2013a MatLab.

1.2 descrizione libreria c++

Eigen: Libreria con l'obbiettivo di essere leggera offrendo supporto alle operazioni su matrici e vettori tramite overloads delle operazioni aritmetiche comuni C++ quali +, -, *, o tramite funzioni speciali come dot(), cross(),

Per la classe Matrix (matrici e vettori), gli operatori sono stati sovrascritti per supportare solo operazioni di algebra lineare.

- **Eigen::loadMarket(A, filename):** Carica matrice Eigen::SparseMatrix<T> A dal file 'filename'. **[OLD non più supportato]**
- **Eigen::VectorXd::Ones(A.rows()):** dichiara matrice di dimensioni fissate (prese dalle dimensioni della matrice A), package 'VectorXd' usato per le operazioni su matrici dinamiche di double.
- **Eigen::SimplicialCholesky<SpMat> chol(A):** Pacchetto creato per gestire matrici di grandi dimensioni con pochi elementi diversi da 0. Implementa uno schema di rappresentazione e gestione dei valori diversi da 0 con uso di poca memoria e alte prestazioni.
Il metodo chol(A) implementa la fattorizzazione di Cholesky della matrice A.
- **Eigen::VectorXd x_ap = chol.solve(b):** Applicazione del risolutore iterativo per risolvere la fattorizzazione.

1.2.1 manutentore

Eigen è un software gratis, open-source sotto la licenza Mozilla Public License 2.0 (MPL2: simple weak copyleft license) dalla versione 3.1.1.

Precedentemente sotto GNU Lesser General Public License.

2 HardWare

2.1 Windows

2.1.1 Tempo

2.1.2 Errore relativo

2.1.3 Memoria

2.2 Linux

2.2.1 Tempo

2.2.2 Errore relativo

2.2.3 Memoria

3 Conclusioni

4 Code