|  |  |
| --- | --- |
|  | Francesco Cavallini  Matricola: 920835  f.cavallini8@campus.unimib.it  Corso di studi di Informatica Magistrale |
| Università degli studi Milano Bicocca  Milano, Padiglione U24  Ottobre, 2024 |  |

|  |  |
| --- | --- |
|  | |
| Easy Matrix  Metodi del Calcolo Scientifico  Relazione Progetto 1-Bis  **Consegna:**  **“““**  Si utilizzi un linguaggio di programmazione a vostra scelta: c++, fortran, java, pyton, etc. Lo scopo del progetto è quello di implementare una mini libreria che esegua i seguenti solutori iterativi, limitatamente al caso di matrici simmetriche e definite positive:  (1) metodo di Jacobi;  (2) metodo di Gauß-Seidel;  (3) metodo del Gradiente;  (4) metodo del Gradiente coniugato.  **”””**  Riferimenti : | |
| Repository git-hub: | https://github.com/VR3ED/Easy\_Matrix |
| Download libreria: | <https://www.nuget.org/packages/easy_matrix/1.0.0> |

Sommario

[0. Introduzione 3](#_Toc1)

[0.1. Obbiettivi 3](#_Toc2)

[0.2. Scelte implementative 3](#_Toc3)

[1. Parte-1: Sviluppo della libreria 4](#_Toc4)

[1.1. Requisiti: 4](#_Toc5)

[1.2: Le basi per lo sviluppo: 4](#_Toc6)

# 0. Introduzione

## 0.1. Obbiettivi

Come definito nella pagina iniziale, l’obbiettivo del progetto è quello di implementare una mini libreria che esegua una serie di solutori iterativi, limitatamente al caso di matrici simmetriche e definite positive.

## 0.2. Scelte implementative

Si sceglie di suddividere lo sviluppo del progetto in 2 parti:

1. Implementazione della libreria
2. Paragoni con librerie open source ed analisi dei risultati.

Per lo sviluppo della prima parte si sceglie di usare il linguaggio C# utilizzando il framework .Net 8.0 per i seguenti motivi:

* **Facilità di pubblicazione libreria:**

Microsoft offre la possibilità di pubblicare gratuitamente la propria libreria sviluppata usando il linguaggio C# sulla piattaforma [NuGet](https://www.nuget.org/). Da questa piattaforma la suddetta libreria potrà venire scaricata, incorporata ed utilizzata in qualsiasi progetto tramite un semplice click al seguente link: <https://www.nuget.org/packages/easy_matrix/1.0.0>

* **Gestione alta precisione:**

Il linguaggio C# permette la gestione dei numeri a virgola mobile con il tipo decimal. Il tipo decimal, seppure permetta di rappresentare solo numeri più piccoli, ha la massima precisione tra i tre tipi a virgola mobile:

|  |  |  |
| --- | --- | --- |
| **C# type/keyword** | **Approximate range** | **Precision** |
| *float* | ±1.5 x 10−45 to ±3.4 x 1038 | ~6-9 digits |
| *double* | ±5.0 × 10−324 to ±1.7 × 10308 | ~15-17 digits |
| *decimal* |  | ~28-29 digits |

Per lo sviluppo della seconda parte si sceglie di fare paragoni con performance di librerie esterne utilizzando il linguaggio python in quanto presenta un amplia gamma di librerie e (come vedremo) codice open-source che renderanno possibile il paragone con la libreria sviluppata al punto 1. Inoltre verrà utilizzato python anche per la formulazione di grafici per il paragone delle statistiche di performance tra libreria sviluppata e codice open source.

# 1. Parte-1: Sviluppo della libreria

## 1.1. Requisiti:

I requisiti che rispetteremo per lo sviluppo delle librerie sono i seguenti:

1. La libreria deve essere in grado di operare con ciascuno dei metodi iterativi sopra menzionati. Inoltre, in caso di utilizzo di librerie esterne, queste librerie devono fornire solamente la struttura dati di matrici e vettori, senza utilizzare i metodi relativi alla risoluzione dei sistemi lineari già implementati al suo interno.
2. I metodi iterativi devono partire da un vettore iniziale nullo e arrestarsi quando la k-esima iterata x(k) soddisfa la condizione:

Dove tol è una tolleranza assegnata dall’utente. Per eseguire i nostri test (che mostreremo nella parte-2) verranno usate . Alternativamente, laddove non si dovesse raggiungere la convergenza entro un numero massimo di iterazioni (maxIter, dove maxIter è un numero elevato a scelta (non inferiore a 20000), la computazione del metodo iterativo dovrà cessare e restituire in output:

convergence = false

1. La libreria deve avere un’architettura ben strutturata anziché essere una sequenza di funzioni indipendenti.

## 1.2: Le basi per lo sviluppo:

Nell’introduzione abbiamo accennato al fatto che tutte le matrici () che daremo in pasto agli algoritmi di libreria sono simmetriche e definite positive; abbiamo quindi che per ogni matrice () per essere **simmetrica e definita positiva** deve rispettare i seguenti criteri:

* La matrice () è simmetrica (ossia )
* La matrice A è definita positiva (ossia per ogni vettore colonna , soddisfa la seguente condizione:

Il metodo di Cholesky permette di scomporre una matrice simmetrica e definita positiva () in

Dove ed sono rispettivamente una matrice triangolare inferiore e superiore

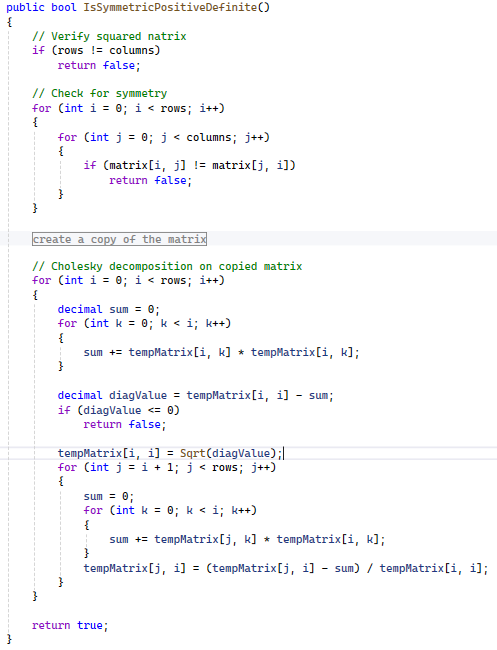
**NOTA**: Metodo di Cholesky

Nella libreria, all’iterno della classe AccurateMatrix (che serve per rappresentare una matrice in formato decimal), è stata infatti inserita una funzione apposta per verificare se le matrici lette in input sono simmetriche e definite positive, ma al posto che controllare se la matrice è definita positiva, si prova invece ad applicare Cholesky, se non ci si riesce allora vuol dire che la matrice non è simmetrica e definita positiva:

Possiamo ora calcolare:

* R[i,i] =

In questo modo si calcola Cholesky per ogni iterazione



Verifica se l'elemento diagonale calcolato nella decomposizione di Cholesky è positivo, ossia prima si calcola poi si calcola il quadrato del valore diagonale: (se questo valore è >0 possiamo procedere)

Verifica se la matrice è simmetrica comparando

con dentro un doppio loop

In C# non esistono metodi per leggere un file .mtx ed assegnare il valore ad un tipo decimal, quindi, in verità, prima di definire se una matrice è simmetrica e definita positiva, la prima sfida implementativa è stata trovare una soluzione per permettere alla libreria di poter leggere i file.

Questa sfida è stata poi superata utilizzando il multi-threading per leggere il file riga per riga come fosse un file txt e creare un parser custom che permettesse di convertire la stringa letta in un valore di tipo decimal (e poi memorizzare il valore nella cella corretta della matrice).   
  
Questo metodo di lettura della matrice non verrò però approfondito in questa relazione in quanto si reputa essere fuori dallo scope della relazione stessa, ma si possono trovare più informazioni su questo metodo sulla repository di git-hub ([qui](https://github.com/VR3ED/Easy_Matrix))

**NOTA IMPLEMENTATIVA**: Lettura delle matrici da file .mtx

Svilupperemo quindi i nostri metodi iterativi

|  |  |
| --- | --- |
|  | VanArsdel, Ltd. |
| Taylor Phillips  5678 Main St  New York, NY 90210  September 16, 20XX |  |

|  |
| --- |
|  |
| Dear Taylor Phillips,  Write the body of your letter here. To update any of the letter's information, select the text, and start typing.  Want to change fonts? Go to the Home tab and choose Fonts. You can use a built-in font combination or select one of your own.  You can also change the colors of the template to match your personal taste. Go to the Design tab and choose a color palette from the Colors menu. Hovering over the different palettes will show you what your document would look like with the new palette.  To change the color or font formatting back to the original settings, go to the Design tab, and select the Theme menu. From there, choose the option to reset the original template theme.  Warm Regards,  Jordan Mitchell  CEO  5678 Main St.  New York, NY 90210  212-555-0199 |
| www.vanarsdelltd.com | jordan@vanarsdelltd.com |

|  |  |
| --- | --- |
|  | VanArsdel, Ltd. |
|  | 5678 Main St.  New York, NY 90210  212-555-0199 |
| Taylor Phillips  5678 Main St  New York, NY 90210  September 16, 20XX |  |

|  |
| --- |
|  |
| Dear Taylor Phillips,  Write the body of your letter here. To update any of the letter's information, select the text, and start typing.  Want to change fonts? Go to the Home tab and choose Fonts. You can use a built-in font combination or select one of your own.  You can also change the colors of the template to match your personal taste. Go to the Design tab and choose a color palette from the Colors menu. Hovering over the different palettes will show you what your document would look like with the new palette.  To change the color or font formatting back to the original settings, go to the Design tab, and select the Theme menu. From there, choose the option to reset the original template theme.  Warm Regards,  Jordan Mitchell  CEO |
| www.vanarsdelltd.com | jordan@vanarsdelltd.com |