Corso di Laurea in Matematica, SAPIENZA Università di Roma

Corsi di ISTITUZIONI DI ANALISI NUMERICA e di FONDAMENTI DI ANALISI NUMERICA
Esercitazioni in Laboratorio

Foglio 3: ALGEBRA LINEARE NUMERICA - TERZA PARTE

A. Scrivere i codici **GSarnoldi + arnoldialg**, **GSlanczos + lanczosalg**, **houshess**, **qrshift** che implementano i seguenti metodi:

1. Algoritmo di proiezione di Arnoldi nel sottospazio di Krylov K_m
2. Algoritmo di proiezione di Lanczos nel sottospazio di Krylov K_m
3. Metodo di Householder di riduzione in forma di Hessenberg
4. Metodo QR

Alcuni codici possono essere trascritti dal manuale Matematica Numerica di Quarteroni, Sacco, Saleri, dopo aver apportato le modifiche discusse in classe.

B. Laboratorio MATLAB.

1. Considerare la matrice simmetrica definita positiva generata con le istruzioni

   ```
   m=22;
   A=delsq(numgrid('S',m));
   ```

   Studiare l'help delle functions **eig**, **eigs** e **sort**. A partire da un vettore random v, usare l'algoritmo di proiezione di Arnoldi per proiettare A nei sottospazi di Krylov $K_{15}(A;v)$, $K_{25}(A;v)$ e $K_{50}(A;v)$. In ognuno dei tre casi:

   (a) usare la function **eig** per calcolare gli autovalori della matrice H;

   (b) ordinare gli autovalori in ordine crescente e salvarli in un vettore;

   (c) visualizzare in **format long e** i primi due e gli ultimi due elementi del vettore e confrontare con l'output fornito dalla function **eigs** applicata ad A nella modalità 'be';

   (d) approssimare il numero di condizionamento in norma 2 di A e confrontare con l'output della function **cond** applicata alla matrice A - dopo averla convertita in formato 'full'. Accertarsi che si tratta di una approssimazione per difetto.

   Fare analogamente usando l'algoritmo di proiezione di Lanczos.

   Confrontare i tempi di esecuzione dei due algoritmi di proiezione in $K_{75}(A;v)$ e visualizzare la struttura (*pattern*) delle matrici H e T costruite (usare **spy**). Commentare i risultati ottenuti.

2. Usare l'algoritmo di Arnoldi per proiettare A nei sottospazi di Krylov $K_{15}(A; v)$ e $K_{400}(A; v)$. Accertarsi che in entrambi i casi sia stata generata effettivamente una base ortonormale del sottospazio e che la matrice di Hessenberg superiore ottenuta rappresenti effettivamente una proiezione di A nel sottospazio. Fare analogamente usando l'algoritmo di proiezione di Lanczos. Commentare i risultati ottenuti.

3. Ridurre la matrice A in forma di Hessenberg superiore facendo uso del codice **houshess**. Verificare se la matrice ottenuta è effettivamente simmetrica, se è effettivamente tridiagonale e se la trasformazione di similitudine è effettivamente ortogonale. Commentare i risultati ottenuti.

4. Considerare la matrice generata con le istruzioni MATLAB

   ```
   rng('default')
   A=sprand(delsq(numgrid('S',22)))+speye(400);
   ```

   A partire da un vettore random, usare l'algoritmo di Arnoldi per proiettare A nel sottospazio di Krylov $K_{50}(A; v)$. Usare la function **eig** per calcolare spettro e autovettori della matrice di Hessenberg superiore H ottenuta. Produrre un grafico per visualizzare gli autovalori nel piano complesso.

   Creare due vettori dove gli autovalori siano ordinati rispettivamente per parte reale crescente e per modulo crescente. Confrontare l'output fornito dalla function **eigs** applicata ad A nelle modalità 'lr', 'sr' e 'lm' con l'autovalore di H, rispettivamente

   (a) di massima parte reale,
   (b) di minima parte reale,
   (c) di massimo modulo.

5. Considerare la matrice simmetrica definita positiva generata con le istruzioni

   ```
   A=gallery('lehmer',25);
   ```

   Calcolarne gli autovalori facendo uso della function **eig**. Fare analogamente con il codice **qrshift** con dieci cifre significative esatte e calcolare la norma dell'errore. Accertarsi che la matrice degli autovettori fornita da **eig** risulti ortogonale.