Corso di Laurea in Matematica, SAPIENZA Università di Roma

Corso di ISTITUZIONI DI ANALISI NUMERICA
Esercitazioni in Laboratorio - A.A. 2018/19

Foglio 3: Algebra lineare numerica - terza parte

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

1. Iterazione di Arnoldi
2. Metodo GMRES
3. Metodo di Arnoldi
4. Iterazione di Lanczos
5. Metodo di Lanczos
6. Metodo di Householder di riduzione in forma di Hessenberg superiore
7. Metodo QR con shift

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

B. Laboratorio MATLAB.

1. Considerare il sistema lineare generato dalle istruzioni

   ```
   n=400; rng('default');
   A=sprand(gallery('poisson',20))+2*speye(n); b=sum(A,2);
   ```

   Fissare tolleranza a 10 cifre significative esatte e vettore iniziale a componenti nulle.

   Risolvere il sistema lineare con il metodo GMRES. Confrontare l'output ottenuto con l'output fornito dalla function **gmres** del MATLAB.

   Costruire il sistema precondizionato con il precondizionatore $P_1 = L_{\text{inc}}U_{\text{inc}}$ e risolvere con il metodo GMRES. Confrontare l'output ottenuto con l'output fornito da **gmres** a cui siano passati in input A, b, L_{inc}, e U_{inc}. Fare analogamente con $P_2 = \text{diag}(\text{diag}(A))$, passando in questo caso a **gmres** in input A, b, e P_2.

2. Considerare la matrice simmetrica definita positiva generata dall'istruzione

   ```
   A=gallery('poisson',20);
   ```

(a) A partire da un vettore random v, usare il metodo di Arnoldi per proiettare A in $K_{15}(A; v)$, $K_{25}(A; v)$ e $K_{50}(A; v)$. Per ogni $k \in \{15, 25, 50\}$, usare le function **eig** e **sort** del MATLAB per calcolare gli autovalori della matrice H_k, ordinarli in ordine crescente, e salvarli nel vettore $\ell_k \in \mathbb{R}^k$.

Facendo uso dei vettori ℓ_k, con $k \in \{15, 25, 50\}$, dare delle approssimazioni del numero di condizionamento in norma spettrale di A.

Fare analogamente usando l'algoritmo di proiezione di Lanczos.

(b) Confrontare i tempi di esecuzione dei due algoritmi di proiezione in $K_{75}(A; v)$ e verificare se è stata generata effettivamente una base ortonormale del sottospazio e visualizzare la struttura delle matrici H_{75} e T_{75} ottenute, facendo uso della function **spy** del MATLAB.

(c) Confrontare le prestazioni dei due algoritmi nel generare una base ortonormale per $K_{400}(A; v)$.

3. Considerare di nuovo la matrice generata al punto 1. Partendo da un vettore random v, usare l'algoritmo di Arnoldi per proiettare A nel sottospazio di Krylov $K_{50}(A; v)$. Eseguire n iterazioni di **qrshift** per approssimare gli autovalori della matrice di Hessenberg superiore H_{50} ottenuta. Determinare gli autovalori

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

   facendo uso delle functions **real**, **abs**, **max**, e **min** del MATLAB. Produrre un grafico per visualizzare lo spettro nel piano complesso. Eseguire le istruzioni

```
eigs(A,1,mod)
```

   con mod uguale a 'lr', 'sr', 'lm', e 'sm', rispettivamente. Confrontare con i quattro i valori di Ritz precedentemente calcolati.

4. Considerare la matrice simmetrica definita positiva generata dall'istruzione

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

   Memorizzare autovalori e autovettori di A calcolati facendo uso di **eig**. Calcolare il numero di condizionamento spettrale del problema computazionale. Ordinare gli autovalori in ordine decrescente, facendo uso di **sort**. Fissando la tolleranza a 10 cifre significative esatte, e facendo variare il massimo numero N di iterazioni, $N = 10 : 5 : 35$, approssimare gli autovalori di A con il codice **qrshift** e calcolare per ogni N la norma dell'errore nell'approssimazione dello spettro.