A.A. 2010/2011

Calcolo Scientifico per l'Informatica
- Laboratory Class -
5

Davide Fugazza
fugazza@elet.polimi.it

Outline

Outline

Iterative Methods

For very large, *sparse* and *non-structured* matrices A, *direct methods* produce the so-called *fill-in* phenomenon. *Iterative methods* are instead the correct choice: A sequence of iterations is produced to approximate the solution

$$x^* = \lim_{k \to +\infty} x^{(k)},$$

which is not reached in a finite number of iteration steps as it happens for direct methods. A *tolerance* criterion will be applied: $\left\| x^{(k+1)} - x^{(k)} \right\| \leq \epsilon$.
Among them:

- Jacobi iteration method
- Gauss-Seidel iteration method
- Richardson iteration method
- Successive OverRelaxation method (SOR)
- Gradient methods

Gradient Methods

Given a s.p.d. matrix, gradient methods reformulate the linear system $A\underline{x} = \underline{b}$ in terms of an equivalent minimization problem of the quadratic form $\Phi(\underline{x})$:

$$\Phi(\underline{x}) = \frac{1}{2}\underline{x}^T A\underline{x} - \underline{x}^T \underline{b}$$

Starting from an initial $\underline{x}^{(0)}$, we construct an iteration scheme to calculate the solution:

$$\underline{x}^{(k)} = \underline{x}^{(k-1)} + \alpha^{(k-1)}\,\underline{d}^{(k-1)}.$$

In its simplest version (gradient descent method), the direction $\underline{d}^{(k)}$ at the k-th step is given by the residual $\underline{r}^{(k)} = \underline{b} - A\underline{x}^{(k)}$ and the step length $\alpha^{(k)}$ is expressed as:

$$\alpha^{(k)} = \frac{\underline{r}^{(k)\,T}\,\underline{r}^{(k)}}{\underline{r}^{(k)\,T}\,A\,\underline{r}^{(k)}}$$

Exercise 1

Given the following linear system:

$$\begin{cases} 3x + 2y = 2 \\ 2x + 6y = -8 \end{cases}$$

1. Verify that the coefficient matrix is symmetric positive definite.
2. Implement a routine in MATLAB for calculating the solution of the system by means of the gradient descent method:
   - Consider a tolerance of 10^{-4} starting from $\underline{x}^{(0)} = [-2 \quad 2]^T$.
   - Visualize the iteration steps of the method respect to the level curves of the quadratic form $\Phi(\underline{x})$ associated to the problem:
   $$\Phi(\underline{x}) = \frac{1}{2}\underline{x}^T A \underline{x} - \underline{x}^T \underline{b},$$
   where A is the coefficient matrix and $\underline{b}$ the right-hand side of the system.

Hint: for graphics see `help meshgrid` and `help contour`.

Exercise 2

Consider the following linear system $A_n \underline{x} = \underline{b}$, where A_n is obtained by discretizing the $2D$ Poisson equation from the gallery of test matrices in MATLAB:

```
>> A  = gallery('poisson',n);
>> b  = A*ones(size(A(:,1)));
```

for `n` = 3, 5, 10 and 20.

1. Calculate the solution of the system by means of the gradient descent method with a tolerance of 10^{-6} and an all-ones starting vector. Give reason of the obtained results:
    - Graphically represent the number of iterations needed for obtaining the required tolerance as a function of the order of the matrix.
    - Do you observe any relationship between the condition number of the matrix A_n and the rapidity of convergence of the method? Justify the answer.

Outline

Overdetermined Systems

Consider a linear system $(m \times n)$ in its matricial form:

$$A\underline{x} = \underline{b}.$$

where $A \in \mathbb{R}^{m \times n}$, with $m > n$, and $\underline{b} \in \mathbb{R}^m$.
We are looking for the vector $\underline{x}^*$ that minimizes the norm of the residual:

$$min_{\underline{x}} \|A\underline{x} - \underline{b}\|_2^2$$

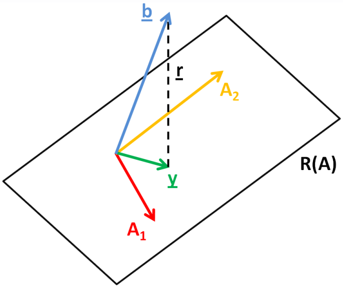
If $\exists y^*$, with $y^* = Ax^*$, so that:

$$(\underline{b} - \underline{y}^*)^T \underline{y} = 0, \qquad \forall \underline{y} \in R(A)$$

then x^* is a solution of the problem. The least-squares problem always has a solution. The solution is unique if and only if A is of full column rank.

Overdetermined Systems

Solving a linear system in the *least-squares sense* is equivalent to find a vector $\underline{x}$ whose image, by means of A, is the vector $\underline{y}$ orthogonal projection of the vector $\underline{b}$ over the column space of A.



Numerical methods for *least-squares problems*:

- Normal Equation Method and Moore-Penrose pseudoinverse
- QR decomposition method
- Singular Value Decomposition (SVD) method

In MATLAB see `help mldivide` also for least-squares solution of lin. systems.

Normal Equations

$$A^T A\underline{x}^* = A^T\underline{b}$$

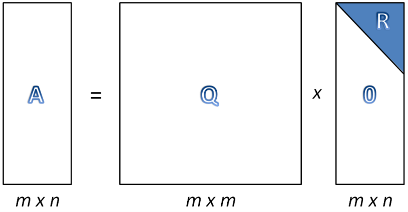
$$\underline{x}^* = (A^T A)^{-1}A^T\underline{b} = A^+\underline{b}$$

where $A^+ = (A^T A)^{-1}A^T$ is the *Moore-Penrose* pseudoinverse matrix.

```
>> help pinv
 PINV   Pseudoinverse.
    X = PINV(A) produces a matrix X of the same dimensions as A'
    so that A*X*A = A, X*A*X = X and A*X and X*A are Hermitian.
    The computation is based on SVD(A) and any singular values
    less than a tolerance are treated as zero. The default
    tolerance is MAX(SIZE(A)) * NORM(A) * EPS(class(A)).
    PINV(A,TOL) uses the tolerance TOL instead of the default.
```

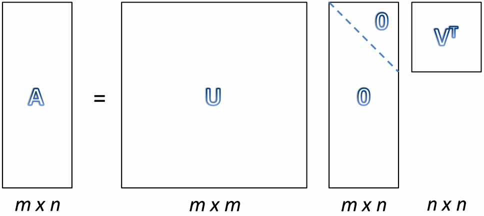
Take a look also at the examples!

QR Decomposition




```
>> help qr
 QR     Orthogonal-triangular decomposition.
    [Q,R] = QR(A), where A is m-by-n, produces an m-by-n upper
    triangular matrix R and an m-by-m unitary matrix Q so that A = Q*R.
...
    If A is full:
    [Q,R,E] = QR(A) produces unitary Q, upper triangular R and a
    permutation matrix E so that A*E = Q*R. The column permutation E
    is chosen so that ABS(DIAG(R)) is decreasing.
...
```

SVD Decomposition




```
>> help svd
 SVD    Singular value decomposition.
    [U,S,V] = SVD(X) produces a diagonal matrix S, of the same
    dimension as X and with nonnegative diagonal elements in
    decreasing order, and unitary matrices U and V so that
    X = U*S*V'.
...
```

Exercise 3

Given the linear system $Ax = b$, where:

$$A = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -3 \\ 0 & 1 & 1 \\ 0 & -1 & 1 \end{bmatrix}, \qquad b = \begin{bmatrix} 4 \\ 6 \\ -1 \\ 2 \end{bmatrix},$$

1. Starting from the normal equations, find the least-squares solutions. Verify that the residual vectors are orthogonal to $R(A)$.
2. Can the Cholesky factorization of A be used to solve the normal equation?
3. Calculate the least-squares solution by means of the singular value decomposition (SVD) of A.
4. Calculate the QR decomposition of A, with Q orthogonal matrix and R upper triangular matrix, and then solve the system.
5. Calculate the least-squares solution using the pseudoinverse matrix of A.
6. Use the `backslash` command. How does MATLAB calculate the solution?

Exercise 4 (From [4])

Given the linear system $Ax = b$, where:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \\ 13 & 14 & 15 \end{bmatrix}, \qquad b = \begin{bmatrix} 16 \\ 17 \\ 18 \\ 19 \\ 20 \end{bmatrix}.$$

1. Starting from the normal equations, find the least-squares solutions.
2. Find a null vector of A.
3. Calculate the least-squares solution with minimum norm: Compare the results obtained with `pinv` and `backslash` MATLAB commands. Which one is better?
4. Find the least square solution by means of the singular value decomposition of the matrix A.

Hint: see `help pinv`, `help mldivide`.

Outline

Application: Image compression

Import the image $Sunset.jpg$ in MATLAB. Associate it to variable A.

1. Calculate the SVD decomposition of the matrix $A = UDV^T$.
2. Called $\sigma_1, \sigma_2, \cdots, \sigma_k$ the singular values of A, approximate the image with a growing number of terms in the SVD:

$$A_r \approx \sum_{i=1}^{r} \sigma_i u_i v_i^T, \text{ with } r < k$$

3. Comment on the sparsity plot of D.

Outline

Eigenvalues and Eigenvectors

Given a matrix A $(n \times n)$, $\lambda \in \mathbb{C}$ is an *eigenvalue* of A if $\exists \underline{x} \in \mathbb{C}$, $\underline{x} \neq \underline{0}$ such that:

$$A\underline{x} = \lambda \underline{x}$$

where $\underline{x}$ is the *eigenvector* corresponding to the *eigenvalue* λ.
In order to be an eigenvalue of A, λ must satisfy:

$$det(A - \lambda I) = 0$$

that corresponds to calculate the roots of the characteristic polynomial $\varphi(\lambda)$.

Different approaches for the *eigenvalue problem*:

- Eigenvalue localization through Gerschgorin theorem
- Local methods: power iteration method and inverse power iteration method
- Global methods: QR.

In MATLAB see `help eig`.

Exercise 5 (From [2])

Consider the following linear system $Ax = b$:

$$A = \begin{bmatrix} 3 & 1 & 1 & 0 & 0 \\ 0 & 4 & 1 & 0.5 & 0.5 \\ 2 & 2 & 5 & 0 & 0 \\ 2 & 2 & 1 & 9 & 1 \\ 0 & 0 & 2 & 3 & 9 \end{bmatrix}, \qquad b = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}.$$

1. Verify that A is not singular by means of Gerschgorin theorem.
2. Validate the results by means of `eig` MATLAB command.
3. Write down the iteration matrix B_J of the Jacobi method and use Gerschgorin theorem to prove that the method converge to the solution of the linear system $Ax = b$.

Hint: Use `gersch` function to ...

Outline

Homework

Exercise H5.1 (Ex 4.5 Cont'd)

Given the linear system obtained with the following instructions:

```
>> B = rand(5) + diag(10*ones(5,1));
>> A = B*B';
```

- √ Analyze the convergence of Jacobi and Gauss-Seidel methods.
- √ Given the (stationary) preconditioned Richardson iterative method, with a diagonal precondition matrix P, whose diagonal is the same one of A, numerically determine the values of the relaxation parameter $\alpha \in (0, 2)$ for which the method is convergent and, in particular, the optimal value α_{opt}.

3. Calculate the solution of the system by means of the gradient descent method.

Homework

Exercise H5.2

Given the linear system $Ax = b$, where:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \\ 4 & 5 & 6 \end{bmatrix}, \qquad b = \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix},$$

1. Does the system has a unique solution?
2. What does it mean finding the least-squares solution of a given linear system?
3. Find the least-squares solution by means of the SVD decomposition of A.
4. Is it possible to use the Moore-Penrose pseudoinverse to find the minimum norm solution?

Hint: see `help pinv`, `help mldivide`, `help svd` and `help qr`.

Homework

Exercise H5.3 (From [2])

Given the linear system $Ax = b$, where:

$$A = \begin{bmatrix} 857375 & 9025 & 95 & 1 \\ 12167 & 529 & 23 & 1 \\ 216000 & 3600 & 60 & 1 \\ 110592 & 2304 & 48 & 1 \\ 704969 & 7921 & 89 & 1 \end{bmatrix}, \qquad b = \begin{bmatrix} 81450625 \\ 279841 \\ 12960000 \\ 5308416 \\ 62742241 \end{bmatrix}$$

1. Find the least-squares solution by applying `inv` MATLAB command to the normal equation, and `mldivide` and `svd` MATLAB commands to matrix A.
2. Verify that the residual vectors are orthogonal to the column space of A.
3. Calculate the 2-norm of the residual vectors and assess, also relying on the results obtained at the previous point: which one of the three solutions is the less accurate?

Hint: see `help pinv`, `help mldivide`, `help svd` and `help qr`.

Homework

Exercise H5.4

Given the following linear system $A\underline{x} = \underline{b}$:

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ 1 \\ 1 \end{bmatrix}.$$

1. What is the Euclidean norm of the minimum residual vector?
2. What is the solution vector $\underline{x}^*$?

Homework

Exercise H5.5

Given the matrices:

$$A_1 = \begin{bmatrix} 4 & -1 & 1 & 0 & 0 \\ 1 & 3 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 1 & 8 \end{bmatrix}, \qquad A_2 = \begin{bmatrix} 5 & 4 & 1 & 1 \\ 4 & 5 & 1 & 1 \\ 1 & 1 & 4 & 2 \\ 1 & 1 & 2 & 4 \end{bmatrix},$$

and the matrix A_3 obtained through the following instructions in MATLAB:

```
>> d = diag([1 7 45]); c = rand(3); A = c'*d*c;
```

use Gerschgorin theorem in order to locate their eigenvalues.

Exercise H5.6 * (From [4])

Try the following MATLAB function, available in the `demos` directory, and try to give reason of the obtained graphs:

```
>> eigshow
```