FF505

Computational Science

Systems of Linear Equations

Marco Chiarandini (marco@imada.sdu.dk)

Department of Mathematics and Computer Science (IMADA)
University of Southern Denmark

Outline

1. Solving Linear Systems

Systems of Linear Equations

How many solutions have these linear systems? Find it out using the graphical approach.

$$\begin{aligned} 6x - 10y &= 2 \\ 3x - 4y &= 5 \end{aligned}$$

```
% plot functions in implicit form
ezplot
```

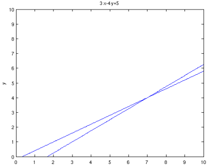
$$\begin{aligned} 3x - 4y &= 5 \\ 6x - 8y &= 10 \end{aligned}$$

$$\begin{aligned} 3x - 4y &= 5 \\ 6x - 8y &= 3 \end{aligned}$$

Systems of Linear Equations

$$\begin{aligned} 6x - 10y &= 2 \\ 3x - \ \ 4y &= 5 \end{aligned}$$

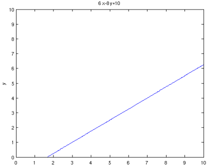
```
% plot functions in implicit form
ezplot('6*x-10*y=2',[0 10 0 10]),
hold,
ezplot('3*x-4*y=5',[0 10 0 10])
```



has one single solution

$$\begin{aligned} 3x - 4y &= \ \ 5 \\ 6x - 8y &= 10 \end{aligned}$$

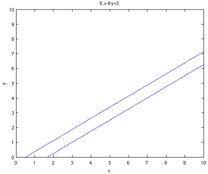
```
ezplot('3*x-4*y=5',[0 10 0 10]),
hold,
ezplot('6*x-8*y=10',[0 10 0 10])
```



has infinite solutions

$$\begin{aligned} 3x - 4y &= 5 \\ 6x - 8y &= 3 \end{aligned}$$

```
ezplot('3*x-4*y=5',[0 10 0 10]),
hold,
ezplot('6*x-8*y=3',[0 10 0 10])
```



has no solution

Matrix Form

The linear system:

$$\begin{aligned} 2x_1 + \quad 9x_2 &= \quad 5 \\ 3x_1 - \quad 4x_2 &= \quad 7 \end{aligned}$$

can be expressed in vector-matrix form as:

$$\begin{bmatrix} 2 & 9 \\ 3 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 5 \\ 7 \end{bmatrix}$$

In general, a set of m equations in n unknowns can be expressed in the form $A\mathbf{x} = \mathbf{b}$, where A is $m \times n$, $\mathbf{x}$ is $n \times 1$ and $\mathbf{b}$ is $m \times 1$.

The inverse of A is denoted A^{-1} and has property that

$$A^{-1}A = AA^{-1} = I$$

Hence the solution to our system is:

$$\mathbf{x} = A^{-1}\mathbf{b}$$

Inverse and Determinant

Compute the inverse and the determinant of this matrix in Matlab:

```
>> A=[3 -4; 6 -8];
```

Has the system solutions?
What about the system in the previous slide? What are its solutions?

A matrix is singular if $\det(A) = |A| = 0$

Inverse of a square matrix A is defined only if A is nonsingular.

If A is singular, the system has no solution

```
>> A=[3 -4; 6 -8];
>> det(A)
ans =
     0
>> inv(A)
Warning: Matrix is singular to working precision.
ans =
   Inf  Inf
   Inf  Inf
```

For a 2×2 matrix

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

the matrix inverse is

$$A^{-1} = \frac{1}{|A|} \begin{bmatrix} d & -c \\ -b & a \end{bmatrix}^T = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

For a 3×3 matrix

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

the matrix inverse is

$$A^{-1} = \frac{1}{|A|} \begin{bmatrix} +\begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} & -\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} & +\begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \\ -\begin{vmatrix} a_{12} & a_{13} \\ a_{32} & a_{33} \end{vmatrix} & +\begin{vmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{vmatrix} & -\begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} \\ +\begin{vmatrix} a_{12} & a_{13} \\ a_{22} & a_{23} \end{vmatrix} & -\begin{vmatrix} a_{11} & a_{13} \\ a_{21} & a_{23} \end{vmatrix} & +\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \end{bmatrix}^T$$

Calculating the inverse

$$A^{-1} = \frac{1}{|A|}\mathrm{adj}(A)$$

$\mathrm{adj(A)}$ is the adjugate matrix of A:

1. Calculate the (i,j) minor of A, denoted M_{ij}, as the determinant of the $(n-1)\times(n-1)$ matrix that results from deleting row i and column j of A.
2. Calculate the cofactor matrix of A, as the $n\times n$ matrix C whose (i,j) entry is the (i,j) cofactor of A

$$C_{ij} = (-1)^{i+j}M_{ij}$$

3. set $\mathrm{adj(A)}_{ij} = C_{ji}$

Left Division Method

- $\mathbf{x} = A^{-1}\mathbf{b}$ rarely applied in practice because calculation is likely to introduce numerical inaccuracy

- The inverse is calculated by LU decomposition, the matrix form of Gaussian elimination.

```
% left division method
x = A\b
```

$$A = LU$$
$$PA = LU$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

- for a matrix A, $n \times n$, $\det(A) \neq 0 \Leftrightarrow$ rank of A is n
  - for a system $A\mathbf{x} = \mathbf{b}$ with m equations and n unknowns a solution exists iff $rank(A) = rank([A\mathbf{b}]) = r$
    - if $r = n \rightsquigarrow$ unique
    - if $r < n \rightsquigarrow$ infinite sol.
  - for a homogeneous system $A\mathbf{x} = \mathbf{0}$ it is always $rank(A) = rank([A\mathbf{b}])$ and there is a nonzero solution iff $rank(A) < n$

• `A\b` works for square and nonsquare matrices. If nonsquare ($m < n$) then the system is underdetermined (infinite solutions). `A\b` returns one variable to zero
• `A\b` does not work when $\det(A) = 0$.

```
>> A=[2, -4,5;-4,-2,3;2,6,-8];
>> b=[-4;4;0];
>> rank(A)
ans =
    2
>> rank([A,b])
ans =
    2
>> x=A\b
Warning: Matrix is singular to working
     precision.
x =
   NaN
   NaN
   NaN
```

However since

$$rank(A) = rank([A\mathbf{b}])$$

an infinite number of solutions exist (underdetermined system).
`x=pinv(A)b` solves with pseudoinverse and `rref([A,b])` finds the reduced row echelon form

Overdetermined Systems

An overdetermined system is a set of equations that has more independent equations than unknowns ($m > n$).

For such a system the matrix inverse method will not work because the A matrix is not square.

However, some overdetermined systems have exact solutions, and they can be obtained with the left division method `x = A \ b`

If a solution does not exist, the left-division answer is the least squares solution. We need to check the ranks of A and $[A\mathbf{b}]$ to know whether the answer is the exact solution.

Flowchart for Linear System Solver