Numerical Linear Algebra
Lecture 1

- Solve Linear systems of equations using Gaussian elimination with different pivoting strategies and blocking algorithms

- Solve Linear systems of equations using Gaussian elimination with different pivoting strategies and blocking algorithms
- Study and use QR decomposition and SVD decomposition

- Solve Linear systems of equations using Gaussian elimination with different pivoting strategies and blocking algorithms
- Study and use QR decomposition and SVD decomposition
- Solve eigenvalue problems based on transformation techniques for symmetric and non-symmetric matrices

- Solve Linear systems of equations using Gaussian elimination with different pivoting strategies and blocking algorithms
- Study and use QR decomposition and SVD decomposition
- Solve eigenvalue problems based on transformation techniques for symmetric and non-symmetric matrices
- Use computer algorithms, programs and software packages (BLAS/LAPACK, MATLAB, PETSC)

- Solve Linear systems of equations using Gaussian elimination with different pivoting strategies and blocking algorithms
- Study and use QR decomposition and SVD decomposition
- Solve eigenvalue problems based on transformation techniques for symmetric and non-symmetric matrices
- Use computer algorithms, programs and software packages (BLAS/LAPACK, MATLAB, PETSC)
- Solve real physical problems by modelling these problems via NLA

Notions from linear algebra

- Linear systems

Notions from linear algebra

- Linear systems
- Matrices (Identity matrix, triangular, singular, symmetric, positive definite, conjugate transpose, rank, norm)

Notions from linear algebra

- Linear systems
- Matrices (Identity matrix, triangular, singular, symmetric, positive definite, conjugate transpose, rank, norm)
- Matrix operations, inverse, transposition, scalar (inner) product, outer product

Notions from linear algebra

- Linear systems
- Matrices (Identity matrix, triangular, singular, symmetric, positive definite, conjugate transpose, rank, norm)
- Matrix operations, inverse, transposition, scalar (inner) product, outer product
- Gaussian elimination

Notions from linear algebra

- Linear systems
- Matrices (Identity matrix, triangular, singular, symmetric, positive definite, conjugate transpose, rank, norm)
- Matrix operations, inverse, transposition, scalar (inner) product, outer product
- Gaussian elimination
- Eigenvalues

Notions from linear algebra

- Linear systems
- Matrices (Identity matrix, triangular, singular, symmetric, positive definite, conjugate transpose, rank, norm)
- Matrix operations, inverse, transposition, scalar (inner) product, outer product
- Gaussian elimination
- Eigenvalues
- Norms

Notions from linear algebra

- Linear systems
- Matrices (Identity matrix, triangular, singular, symmetric, positive definite, conjugate transpose, rank, norm)
- Matrix operations, inverse, transposition, scalar (inner) product, outer product
- Gaussian elimination
- Eigenvalues
- Norms
- LU-factorization, pivoting, row echelon form

Sources

- Wikipedia
- Course Literature
  James W. Demmel: Applied Numerical Linear Algebra, SIAM 1997.
- Lapack
  http://netlib.org/lapack/
- PETSC http://www.mcs.anl.gov/petsc/

PETSc: example of Makefile

```
PETSC_DIR = /chalmers/sw/sup64/petsc-3.0.0-p12
PETSC_ARCH = /chalmers/sw/sup64/petsc-3.0.0-p12
include $PETSC_ARCH/conf/base
CC = gcc
CXX = g++
CXXFLAGS = -O3 -xHOST
BOPT = g++
MPI_INCLUDE = $PETSC_ARCH/include/mpiuni
CPPFLAGS += -I.
CPPFLAGS += -fpermissive
example : Example.o chkopts $CXX -o example Example.o
$(PETSC_LIB)
```

Linear systems

- A linear system is a mathematical model of a system which uses definition of a linear operator. Linear systems have important applications in automatic control theory, signal processing, and telecommunications. For example, the propagation medium for wireless communication systems can often be modeled by linear systems.
- A general deterministic system can be described by operator, H, that maps an input, x(t), as a function of t to an output, y(t), a type of black box description. Linear systems satisfy the properties of superposition and scaling or homogeneity. Given two valid inputs $x_1(t), x_2(t)$ as well as their respective outputs

$$y_1(t) = H\{x_1(t)\}\,; y_2(t) = H\{x_2(t)\}$$

a linear system must satisfy to the equation
$\alpha y_1(t) + \beta y_2(t) = H\{\alpha x_1(t) + \beta x_2(t)\}$
for any scalar values of α and β.

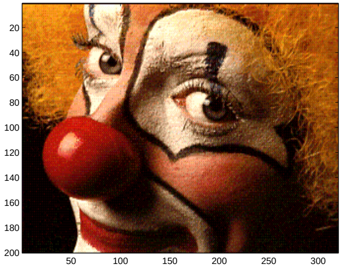
Example of application of linear systems: image compression using SVD

Definition SVD *Let A be an arbitrary m-by-n matrix with $m \geq n$. Then we can write $A = U\Sigma V^T$, where U is m-by-n and satisfies $U^T U = I$, V is n-by-n and satisfies $V^T V = I$, and $\Sigma = diag(\sigma_1, \dots, \sigma_n)$, where $\sigma_1 \geq \dots \geq \sigma_n \geq 0$. The columns $u_1, \dots, u_n$ of U are called left singular vectors. The columns $v_1, \dots, v_n$ of V are called right singular vectors. The σ_i are called singular values. (If $m < n$, the SVD is defined by considering A^T.)*

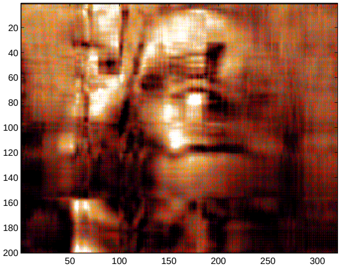
Theorem

Write $V = [\upsilon_1, \upsilon_2, \dots, \upsilon_n]$ and $U = [u_1, u_2, \dots, u_n]$, so $A = U\Sigma V^T = \sum_{i=1}^n \sigma_i u_i \upsilon_i^T$ (a sum of rank-1 matrices). Then a matrix of rank $k < n$ closest to A (measured with $||\cdot||_2$ is $A_k = \sum_{i=1}^k \sigma_i u_i \upsilon_i^T$ and $||A - A_k||_2 = \sigma_{k+1}$. We may also write $A_k = U\Sigma_k V^T$ where $\Sigma_k = diag(\sigma_1, \dots, \sigma_k, 0, \dots, 0)$.

Example of application of linear systems: image compression using SVD



a) Original image



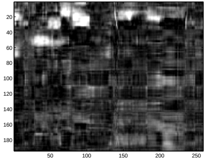
b) Rank k=20 approximation

Example of application of linear systems: image compression using SVD in Matlab

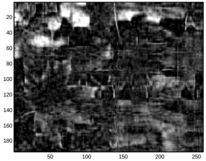
See path for other pictures:
/matlab-2012b/toolbox/matlab/demos
load clown.mat;
Size(X) $= m \times n = 320 \times 200$ pixels.
[U,S,V] = svd(X);
colormap(map);
k=20;
image(U(:,1:k)*S(1:k,1:k)*V(:,1:k)');
Now: size(U)$= m \times k$, size(V)$= n \times k$.

Example of application of linear systems: image compression using SVD in Matlab

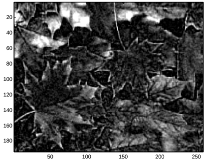
a) Original image



b) Rank k=10 approximation



c) Rank k=20 approximation



d) Rank k=50 approximation

Example of application of linear systems: image compression using SVD for arbitrary image

To get image on the previous slide, I took picture in jpg-format and loaded it in matlab like that:
A = imread('autumn.jpg');
You can not simply apply SVD to A: svd(A) Undefined function 'svd' for input arguments of type 'uint8'.
Apply type "double" to A: DA = double(A), and then perform
[U,S,V] = svd(DA);
colormap('gray');
k=20;
image(U(:,1:k)*S(1:k,1:k)*V(:,1:k)');
Now: size(U)= $m \times k$, size(V)= $n \times k$.

Example of application of linear systems: image deblurring

Original Image

Blurred Image

Figure: left: exact matrix **X**, right: approximated matrix **B**

The blurring model

Consider a grayscale image

- $\mathbf{X}$: $m \times n$ matrix representing the exact image
- $\mathbf{B}$: $m \times n$ matrix representing the blurred image

The blurring model

Consider a grayscale image

- $\mathbf{X}$: $m \times n$ matrix representing the exact image
- $\mathbf{B}$: $m \times n$ matrix representing the blurred image

Assume linear blurring.

$$\mathbf{x} = \mathrm{vec}(\mathbf{X}) = \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_n \end{bmatrix} \in \mathbb{R}^N, \quad \mathbf{b} = \mathrm{vec}(\mathbf{B}) = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_n \end{bmatrix} \in \mathbb{R}^N$$

$\mathbf{A}$ $N \times N$ matrix, with $N = m \cdot n$

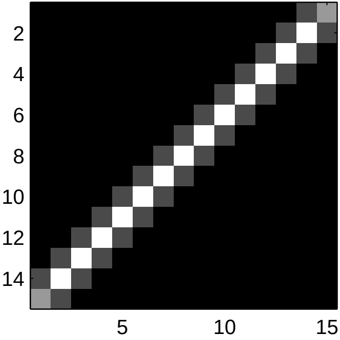
$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

Knowing $\mathbf{X}$ and $\mathbf{A}$ it is straightforward to compute the blurred image.

Motion blur

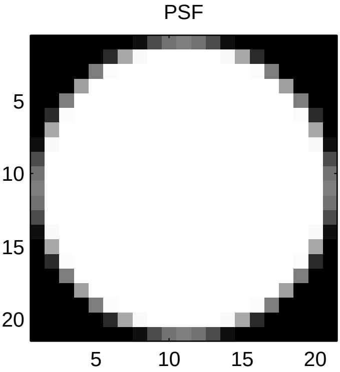
Motion Blurred Image

PSF

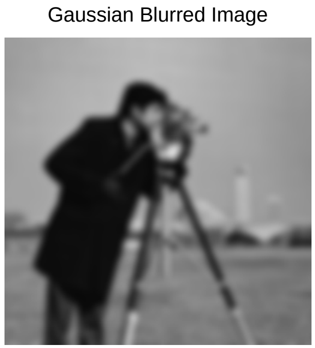


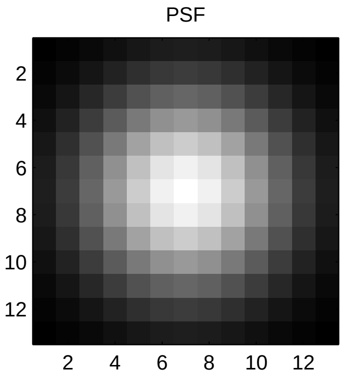
Out-of-focus blur

Blurred Image



Gaussian blur





Let H be the Hilbert space H^1 and let $\Omega \subset \mathbb{R}^m$, $m = 2, 3$, be a convex bounded domain. Our goal is to solve a Fredholm integral equation of the first kind for $x \in \Omega$

$$\int_\Omega K(x-y)z(x)dx = u(y), \tag{1}$$

where $u(y) \in L_2(\bar{\Omega})$, $z(x) \in H$, $K(x-y) \in C^k\left(\overline{\Omega}\right)$, $k \geq 0$ be the kernel of the integral equation.
Let us rewrite (1) in an operator form as

$$A(z) = u \tag{2}$$

with an operator $A : H \to L_2(\bar{\Omega})$ defined as

$$A(z) := \int_\Omega K(x-y)z(x)dx. \tag{3}$$

Ill-posed problem.

Let the function $z(x) \in H^1$ of the equation (1) be unknown in the domain Ω. Determine the function $z(x)$ for $x \in \Omega$ assuming the functions $K(x-y) \in C^k\left(\overline{\Omega}\right), k \geq 0$ and $u(x) \in L_2(\Omega)$ in (1) are known.
Let $\delta > 0$ be the error in the right-hand side of the equation (1):

$$A\left(z^*\right) = u^*, \quad \|u - u^*\|_{L_2(\sigma)} \leq \delta. \tag{4}$$

where u^* is the exact right-hand side corresponding to the exact solution z^*.
To find the approximate solution of the equation (1) we minimize the functional

$$M_\alpha\left(z\right) = \|Az - u\|_{L_2(\Omega)}^2 + \alpha\|z\|_{H_1(\Omega)}^2, \tag{5}$$

$$M_\alpha : H^1 \to \mathbb{R},$$

where $\alpha = \alpha\left(\delta\right) > 0$ is the small regularization parameter.

We consider now more general form of the Tikhonov functional (5). Let W_1, W_2, Q be three Hilbert spaces, $Q \subseteq W_1$ as a set, the norm in Q is stronger than the norm in W_1 and $\overline{Q} = W_1$, where the closure is understood in the norm of W_1. We denote scalar products and norms in these spaces as

$$(\cdot,\cdot)\,,\|\cdot\| \text{ for } W_1,$$

$$(\cdot,\cdot)_2\,,\|\cdot\|_2 \text{ for } W_2$$

$$\text{and } [\cdot,\cdot]\,,[\cdot] \text{ for } Q.$$

Let $A : W_1 \to W_2$ be a bounded linear operator. Our goal is to find the function $z(x) \in Q$ which minimizes the Tikhonov functional

$$E_\alpha(z) : Q \to \mathbb{R}, \tag{6}$$

$$E_\alpha(z) = \frac{1}{2}\|Az - u\|_2^2 + \frac{\alpha}{2}[z - z_0]^2, u \in W_2; z, z_0 \in Q, \tag{7}$$

where $\alpha \in (0,1)$ is the regularization parameter. To do that we search for a stationary point of the above functional with respect to z satisfying $\forall b \in Q$

$$E'_\alpha(z)(b) = 0. \tag{8}$$

The following lemma is well known for the case $W_1 = W_2 = L_2$.
Lemma 1. *Let* $A : L_2 \to L_2$ *be a bounded linear operator. Then the Fréchet derivative of the functional (5) is*

$$E'_\alpha(z)(b) = (A^* Az - A^* u, b) + \alpha [z - z_0, b], \forall b \in Q. \tag{9}$$

In particular, for the integral operator (1) we have

$$\begin{aligned} E'_\alpha(z)(b) = \int_\Omega b(s) \left[\int_\Omega z(y) \left(\int_\Omega K(x-y)K(x-s)dx \right) dy \right. \\ \left. - \int_\Omega K(x-s)u(x)\,dxds \right. \\ +\alpha [z - z_0, b], \forall b \in Q. \end{aligned} \tag{10}$$

Lemma 2 is also well known, since $A : W_1 \to W_2$ is a bounded linear operator. We formulate this lemma only for our specific case.

Lemma 2. *Let the operator $A : W_1 \to W_2$ satisfies conditions of Lemma 1. Then the functional $E_\alpha(z)$ is strongly convex on the space Q with the convexity parameter κ* such that

$$\left(E'_\alpha(x) - E'_\alpha(z), x - z\right) \geq \kappa [x - z]^2, \forall x, z \in Q. \tag{11}$$

Similarly, the functional $M_\alpha(z)$ is also strongly convex on the Sobolev space H_1:

$$\left(M'_\alpha(x) - M'_\alpha(z), x - z\right)_{H_1} \geq \kappa ||x - z||^2_{H_1}, \forall x, z \in H_1, \quad (12)$$

Find z via any gradient-like method. For example, perform usual gradient update

$$z^{k+1} = z^k + \beta E'_\alpha\left(z^k\right)(b). \quad (13)$$

until $||z^{k+1} - z^k||$ converges.

Identity matrix

The identity matrix or unit matrix of size n is the $n \times n$ square matrix with ones on the main diagonal and zeros elsewhere. It is denoted by I_n, or simply by I.

$$I_1 = \begin{bmatrix} 1 \end{bmatrix}, \; I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \; \cdots, \quad I_n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

When A has size $m \times n$, it is a property of matrix multiplication that $I_m A = A I_n = A$.
Using the notation that is sometimes used to concisely describe diagonal matrices, we can write:
$I_n = \mathrm{diag}(1, 1, ..., 1)$.
It can also be written using the Kronecker delta notation:
$(I_n)_{ij} = \delta_{ij}$.

Triangular matrix

- A square matrix is called lower triangular if all the entries above the main diagonal are zero.

$$L = \begin{bmatrix} l_{1,1} & & & & 0 \\ l_{2,1} & l_{2,2} & & & \\ l_{3,1} & l_{3,2} & \ddots & & \\ \vdots & \vdots & \ddots & \ddots & \\ l_{n,1} & l_{n,2} & \dots & l_{n,n-1} & l_{n,n} \end{bmatrix}$$

- A square matrix is called upper triangular if all the entries below the main diagonal are zero.

$$U = \begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} & \dots & u_{1,n} \\ & u_{2,2} & u_{2,3} & \dots & u_{2,n} \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & u_{n-1,n} \\ 0 & & & & u_{n,n} \end{bmatrix}$$

Triangular matrix

- A triangular matrix is one that is either lower triangular or upper triangular.
- A matrix that is both upper and lower triangular is a diagonal matrix.

$$D_n = \begin{bmatrix} d_{1,1} & 0 & \cdots & 0 \\ 0 & d_{2,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_{n,n} \end{bmatrix}$$

Singular matrix

A square matrix that does not have a matrix inverse. A matrix is singular if its determinant is 0. For example, there are 10 2×2 singular $(0, 1)$-matrices:

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$
$$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Symmetric and positive definite matrix

- A symmetric matrix is a square matrix that is equal to its transpose. Let A be a symmetric matrix. Then:

$$A = A^T.$$

  If the entries of matrix A are written as $A = (a_{ij})$, then the symmetric matrix A is such that $a_{ij} = a_{ji}$.

- An $n \times n$ real matrix M is positive definite if $z^T Mz > 0$ for all non-zero vectors z with real entries ($z \in \mathbb{R}^n$), where z^T denotes the transpose of z.

- An $n \times n$ complex matrix M is positive definite if $Re(z^* Mz) > 0$ for all non-zero complex vectors z, where z^* denotes the conjugate transpose of z and $Re(c)$ is the real part of a complex number c.

Examples

- The following matrix is symmetric:

$$\begin{bmatrix} 1 & 7 & 3 \\ 7 & 4 & -5 \\ 3 & -5 & 6 \end{bmatrix}.$$

- Every diagonal matrix is symmetric, since all off-diagonal entries are zero.

Examples

- The nonnegative matrix

$$M_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

is positive definite.

For a vector with entries

$$\mathbf{z} = \begin{bmatrix} z_0 \\ z_1 \end{bmatrix}$$

the quadratic form is

$$\begin{bmatrix} z_0 & z_1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \end{bmatrix} = \begin{bmatrix} z_0 \cdot 1 + z_1 \cdot 0 & z_0 \cdot 0 + z_1 \cdot 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \end{bmatrix} = z_0^2 + z_1^2;$$

when the entries $z0, z1$ are real and at least one of them nonzero, this is positive.

A matrix in which some elements are negative may still be positive-definite. An example is given by

$$M_1 = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}.$$

It is positive definite since for any non-zero vector

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix},$$

we have

$$
\begin{aligned}
x^{\mathrm{T}} M_1 x &= \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \\
&= \begin{bmatrix} (2x_1 - x_2) & (-x_1 + 2x_2 - x_3) & (-x_2 + 2x_3) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \\
&= 2{x_1}^2 - 2x_1x_2 + 2{x_2}^2 - 2x_2x_3 + 2{x_3}^2 \\
&= {x_1}^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + {x_3}^2
\end{aligned}
$$

which is a sum of squares and therefore nonnegative; in fact, each squared summa can be zero only when $x_1 = x_2 = x_3 = 0$, so M_1 is indeed positive-definite.

Conjugate transpose matrix

The conjugate transpose, Hermitian transpose, Hermitian conjugate, or adjoint matrix of an m-by-n matrix A with complex entries is the n-by-m matrix A^* obtained from A by taking the transpose and then taking the complex conjugate of each entry (i.e., negating their imaginary parts but not their real parts). The conjugate transpose is formally defined by

$$(\mathbf{A}^*)_{ij} = \overline{\mathbf{A}_{ji}}$$

where the subscripts denote the i,j-th entry, and the overbar denotes a scalar complex conjugate. (The complex conjugate of $a + bi$, where a and b are reals, is $a - bi$.)
This definition can also be written as

$$\mathbf{A}^* = (\overline{\mathbf{A}})^{\mathrm{T}} = \overline{\mathbf{A}^{\mathrm{T}}}$$

where $\mathbf{A}^{\mathrm{T}}$ denotes the transpose and $\overline{\mathbf{A}}$, denotes the matrix with complex conjugated entries.

The conjugate transpose of a matrix A can be denoted by any of these symbols:

$$\mathbf{A}^* \text{ or } \mathbf{A}^{\mathrm{H}},$$

commonly used in linear algebra.
Example
If

$$\mathbf{A} = \begin{bmatrix} 3+i & 5 & -2i \\ 2-2i & i & -7-13i \end{bmatrix}$$

then

$$\mathbf{A}^* = \begin{bmatrix} 3-i & 2+2i \\ 5 & -i \\ 2i & -7+13i \end{bmatrix}$$

Basic remarks

- A square matrix A with entries a_{ij} is called Hermitian or self-adjoint if $A = A^*$, i.e., $a_{ij} = \overline{a_{ji}}$.
- normal if $A^* A = AA^*$.
- unitary if $A^* = A^{-1}$. a unitary matrix is a (square) $n \times n$ complex matrix A satisfying the condition $A^* A = AA^* = I_n$, where I_n is the identity matrix in n dimensions.
- Even if A is not square, the two matrices $A^* A$ and AA^* are both Hermitian and in fact positive semi-definite matrices.
- Finding the conjugate transpose of a matrix A with real entries reduces to finding the transpose of A, as the conjugate of a real number is the number itself.

Row echelon form

In linear algebra a matrix is in row echelon form if

- All nonzero rows (rows with at least one nonzero element) are above any rows of all zeroes [All zero rows, if any, belong at the bottom of the matrix]
- The leading coefficient (the first nonzero number from the left, also called the pivot) of a nonzero row is always strictly to the right of the leading coefficient of the row above it.
- All entries in a column below a leading entry are zeroes (implied by the first two criteria).

This is an example of 3×4 matrix in row echelon form:

$$\left[\begin{array}{ccc|c} 1 & a_1 & a_2 & a_3 \\ 0 & 2 & a_4 & a_5 \\ 0 & 0 & -1 & a_6 \end{array}\right]$$

Row echelon form

A matrix is in reduced row echelon form (also called row canonical form) if it satisfies the additional condition: Every leading coefficient is 1 and is the only nonzero entry in its column, like in this example:

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & b_1 \\ 0 & 1 & 0 & b_2 \\ 0 & 0 & 1 & b_3 \end{array}\right]$$

Note that this does not always mean that the left of the matrix will be an identity matrix. For example, the following matrix is also in reduced row-echelon form:

$$\left[\begin{array}{cccc|c} 1 & 0 & 1/2 & 0 & b_1 \\ 0 & 1 & -1/3 & 0 & b_2 \\ 0 & 0 & 0 & 1 & b_3 \end{array}\right]$$

Rank

- Column rank of a matrix A is the maximum number of linearly independent column vectors of A. The row rank of a matrix A is the maximum number of linearly independent row vectors of A. Equivalently, the column rank of A is the dimension of the column space of A, while the row rank of A is the dimension of the row space of A.
- A result of fundamental importance in linear algebra is that the column rank and the row rank are always equal. It is commonly denoted by either $rk(A)$ or $rank\ A$. Since the column vectors of A are the row vectors of the transpose of A (denoted here by A^T), column rank of A equals row rank of A is equivalent to saying that the rank of a matrix is equal to the rank of its transpose, i.e. $rk(A) = rk(A^T)$.
- The rank of an $m \times n$ matrix cannot be greater than m nor n. A matrix that has a rank as large as possible is said to have full rank; otherwise, the matrix is rank deficient.

Cofactor

In linear algebra, the cofactor (sometimes called adjunct, see below) describes a particular construction that is useful for calculating both the determinant and inverse of square matrices. Specifically the cofactor of the (i, j) entry of a matrix, also known as the (i, j) cofactor of that matrix, is the signed minor of that entry.

Informal approach to minors and cofactors

Finding the minors of a matrix A is a multi-step process:

- Choose an entry a_{ij} from the matrix.
- Cross out the entries that lie in the corresponding row i and column j.
- Rewrite the matrix without the marked entries.
- Obtain the determinant M_{ij} of this new matrix.

If $i + j$ is an even number, the cofactor C_{ij} of a_{ij} coincides with its minor: $C_{ij} = M_{ij}$.
Otherwise, it is equal to the additive inverse of its minor: $C_{ij} = -M_{ij}$.

Formal definition of cofactor

If A is a square matrix, then the minor of its entry a_{ij}, also known as the (i, j) minor of A, is denoted by M_{ij} and is defined to be the determinant of the submatrix obtained by removing from A its i-th row and j-th column.
It follows: $C_{ij} = (-1)^{i+j} M_{ij}$ and C_{ij} is called the cofactor of a_{ij}.

Example

Given the matrix

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

suppose we wish to find the cofactor C_{23}. The minor M_{23} is the determinant of the above matrix with row 2 and column 3 removed.

$$M_{23} = \begin{vmatrix} b_{11} & b_{12} & \square \\ \square & \square & \square \\ b_{31} & b_{32} & \square \end{vmatrix} \text{ yields } M_{23} = \begin{vmatrix} b_{11} & b_{12} \\ b_{31} & b_{32} \end{vmatrix} = b_{11} b_{32} - b_{31} b_{12}$$

Using the given definition it follows that

$$C_{23} = (-1)^{2+3}(M_{23})$$
$$C_{23} = (-1)^{5}(b_{11} b_{32} - b_{31} b_{12})$$
$$C_{23} = b_{31} b_{12} - b_{11} b_{32}.$$

Invertible matrix

- In linear algebra an n-by-n (square) matrix A is called invertible (some authors use nonsingular or nondegenerate) if there exists an n-by-n matrix B such that $\mathbf{AB} = \mathbf{BA} = \mathbf{I}_n$, where I_n denotes the n-by-n identity matrix and the multiplication used is ordinary matrix multiplication. If this is the case, then the matrix B is uniquely determined by A and is called the inverse of A, denoted by A^{-1}. It follows from the theory of matrices that if $\mathbf{AB} = \mathbf{I}$ for finite square matrices A and B, then also $\mathbf{BA} = \mathbf{I}$.
- Non-square matrices (m-by-n matrices which do not have an inverse). However, in some cases such a matrix may have a left inverse or right inverse. If A is m-by-n and the rank of A is equal to n, then A has a left inverse: an n-by-m matrix B such that $BA = I$. If A has rank m, then it has a right inverse: an n-by-m matrix B such that $AB = I$.
- A square matrix that is not invertible is called singular or degenerate. A square matrix is singular if and only if its determinant is 0.

Methods of matrix inversion

- Gaussian elimination
- Gauss-Jordan elimination is an algorithm that can be used to determine whether a given matrix is invertible and to find the inverse. An alternative is the LU decomposition which generates upper and lower triangular matrices which are easier to invert. For special purposes, it may be convenient to invert matrices by treating m·n-by-m·n matrices as m-by-m matrices of n-by-n matrices, and applying one or another formula recursively (other sized matrices can be padded out with dummy rows and columns). For other purposes, a variant of Newton's method may be convenient. Newton's method is particularly useful when dealing with families of related matrices: sometimes a good starting point for refining an approximation for the new inverse can be the already obtained inverse of a previous matrix that nearly matches the current matrix. Newton's method is also useful for "touch up" corrections to the Gauss-Jordan algorithm which has been contaminated by small errors due to imperfect computer arithmetic.

Eigendecomposition

Let A be a square $n \times n$ matrix. Let $q_1 ... q_k$ be an eigenvector basis, i.e. an indexed set of k linearly independent eigenvectors, where k is the dimension of the space spanned by the eigenvectors of A. If $k = n$, then A can be written

$$\mathbf{A} = \mathbf{QUQ}^{-1}$$

where Q is the square $n \times n$ matrix whose i-th column is the basis eigenvector q_i of A, and U is the diagonal matrix whose diagonal elements are the corresponding eigenvalues, i.e. $U_{ii} = \lambda_i$.

Properties

Let A be an $n \times n$ matrix with eigenvalues $\lambda_i, i = 1, 2, \ldots, n$. Then

- Trace of A

$$\text{tr}(A) = \sum \lambda_i = \lambda_1 + \lambda_2 + \cdots + \lambda_n.$$

- Determinant of A

$$\det(A) = \prod \lambda_i = \lambda_1 \lambda_2 \cdots \lambda_n.$$

- Eigenvalues of A^k are $\lambda_1^k, \ldots, \lambda_n^k$.
  These first three results follow by putting the matrix in upper-triangular form, in which case the eigenvalues are on the diagonal and the trace and determinant are respectively the sum and product of the diagonal.
- If $A = A^H$, i.e., A is Hermitian, every eigenvalue is real.
- Every eigenvalue of a Unitary matrix has absolute value $|\lambda| = 1$.

Example

We take a 2×2 matrix

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix}$$

and want it to be decomposed into a diagonal matrix. First, we multiply to a non-singular matrix

$$\mathbf{B} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, [a, b, c, d] \in \mathbb{R}.$$

Then

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} x & 0 \\ 0 & y \end{bmatrix},$$

for some real diagonal matrix

$$\begin{bmatrix} x & 0 \\ 0 & y \end{bmatrix}.$$

Shifting B to the right hand side:

$$\begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x & 0 \\ 0 & y \end{bmatrix}$$

The above equation can be decomposed into 2 simultaneous equations:

$$\begin{cases} \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} a \\ c \end{bmatrix} = \begin{bmatrix} ax \\ cx \end{bmatrix} \\ \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} b \\ d \end{bmatrix} = \begin{bmatrix} by \\ dy \end{bmatrix} \end{cases}$$

Factoring out the eigenvalues x and y:

$$\begin{cases} \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} a \\ c \end{bmatrix} = x \begin{bmatrix} a \\ c \end{bmatrix} \\ \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} b \\ d \end{bmatrix} = y \begin{bmatrix} b \\ d \end{bmatrix} \end{cases}$$

Letting

$$\overrightarrow{a} = \begin{bmatrix} a \\ c \end{bmatrix}, \overrightarrow{b} = \begin{bmatrix} b \\ d \end{bmatrix},$$

this gives us two vector equations:

$$\begin{cases} A\overrightarrow{a} = x\overrightarrow{a} \\ A\overrightarrow{b} = y\overrightarrow{b} \end{cases}$$

And can be represented by a single vector equation involving 2 solutions as eigenvalues:

$\mathbf{Au} = \lambda\mathbf{u}$

where λ represents the two eigenvalues x and y, $\mathbf{u}$ represents the vectors $\overrightarrow{a}$ and $\overrightarrow{b}$.

Shifting $\lambda\mathbf{u}$ to the left hand side and factorizing $\mathbf{u}$ out

$(\mathbf{A} - \lambda\mathbf{I})\mathbf{u} = 0$

Since $\mathbf{B}$ is non-singular, it is essential that $\mathbf{u}$ is non-zero. Therefore,

$(\mathbf{A} - \lambda\mathbf{I}) = \mathbf{0}$

Considering the determinant of $(\mathbf{A} - \lambda\mathbf{I})$,

$$\begin{bmatrix} 1-\lambda & 0 \\ 1 & 3-\lambda \end{bmatrix} = 0$$

Thus

$(1-\lambda)(3-\lambda) = 0$

Giving us the solutions of the eigenvalues for the matrix $\mathbf{A}$ as $\lambda = 1$ or $\lambda = 3$, and the resulting diagonal matrix from the eigendecomposition of $\mathbf{A}$ is thus
$\begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix}$.
Putting the solutions back into the above simultaneous equations

$$\begin{cases} \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} a \\ c \end{bmatrix} = 1 \begin{bmatrix} a \\ c \end{bmatrix} \\ \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} b \\ d \end{bmatrix} = 3 \begin{bmatrix} b \\ d \end{bmatrix} \end{cases}$$

Solving the equations, we have $a = -2c, a \in \mathbb{R}$ and $b = 0, d \in \mathbb{R}$
Thus the matrix $\mathbf{B}$ required for the eigendecomposition of $\mathbf{A}$ is
$\begin{bmatrix} -2c & 0 \\ c & d \end{bmatrix}$, [c, d]$\in \mathbb{R}. i.e.$:

$$\begin{bmatrix} -2c & 0 \\ c & d \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} -2c & 0 \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix}, [c, d] \in \mathbb{R}$$

Methods of matrix inversion

- Eigendecomposition

  If matrix A can be eigendecomposed and if none of its eigenvalues are zero, then A is nonsingular and its inverse is given by $\mathbf{A}^{-1} = \mathbf{Q}\Lambda^{-1}\mathbf{Q}^{-1}$.

  Furthermore, because U is a diagonal matrix, its inverse is easy to calculate: $\left[\Lambda^{-1}\right]_{ii} = \frac{1}{\lambda_i}$.

- Cholesky decomposition

  If matrix A is positive definite, then its inverse can be obtained as $\mathbf{A}^{-1} = (\mathbf{L}^*)^{-1}\mathbf{L}^{-1}$, where L is the lower triangular Cholesky decomposition of A.

Methods of matrix inversion

- Analytic solution

  Writing the transpose of the matrix of cofactors, known as an adjugate matrix, can also be an efficient way to calculate the inverse of small matrices, but this recursive method is inefficient for large matrices. To determine the inverse, we calculate a matrix of cofactors:

$$\mathbf{A}^{-1} = \frac{1}{|\mathbf{A}|} \left(\mathbf{C}^{\mathrm{T}}\right)_{ij} = \frac{1}{|\mathbf{A}|} \left(\mathbf{C}_{ji}\right) = \frac{1}{|\mathbf{A}|} \begin{pmatrix} \mathbf{C}_{11} & \mathbf{C}_{21} & \cdots & \mathbf{C}_{n1} \\ \mathbf{C}_{12} & \mathbf{C}_{22} & \cdots & \mathbf{C}_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{1n} & \mathbf{C}_{2n} & \cdots & \mathbf{C}_{nn} \end{pmatrix}$$

  where $|A|$ is the determinant of A, C_{ij} is the matrix of cofactors, and C^T represents the matrix transpose.

Inversion of 2×2 matrices

The cofactor equation listed above yields the following result for 2×2 matrices. Inversion of these matrices can be done easily as follows:

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}.$$

This is possible because $1/(ad - bc)$ is the reciprocal of the determinant of the matrix in question, and the same strategy could be used for other matrix sizes.

Inversion of 3×3 matrices

A computationally efficient 3×3 matrix inversion is given by

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & k \end{bmatrix}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A & B & C \\ D & E & F \\ G & H & K \end{bmatrix}^{T} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A & D & G \\ B & E & H \\ C & F & K \end{bmatrix}$$

where the determinant of A can be computed by applying the rule of Sarrus as follows:
$\det(\mathbf{A}) = a(ek - fh) - b(kd - fg) + c(dh - eg)$.
If the determinant is non-zero, the matrix is invertible, with the elements of the above matrix on the right side given by

$$\begin{array}{lll} A = (ek - fh) & D = (ch - bk) & G = (bf - ce) \\ B = (fg - dk) & E = (ak - cg) & H = (cd - af) \\ C = (dh - eg) & F = (gb - ah) & K = (ae - bd). \end{array}$$

Eigenvalues and eigenvectors

- The vector x is an eigenvector of the matrix A with eigenvalue λ (lambda) if the following equation holds: $\mathbf{Ax} = \lambda \mathbf{x}$.

  If the eigenvalue $\lambda > 1, x$ is stretched by this factor. If $\lambda = 1$, the vector x is not affected at all by multiplication by A. If $0 < \lambda < 1$, x is shrunk (or compressed). The case $\lambda = 0$ means that x shrinks to a point (represented by the origin), meaning that x is in the kernel of the linear map given by A. If $\lambda < 0$ then x flips and points in the opposite direction as well as being scaled by a factor equal to the absolute value of λ.
- As a special case, the identity matrix I is the matrix that leaves all vectors unchanged: $I\mathbf{x} = 1\mathbf{x} = \mathbf{x}$.,
- Every non-zero vector x is an eigenvector of the identity matrix with eigenvalue $\lambda = 1$.

Eigenvalues and eigenvectors

- The eigenvalues of A are precisely the solutions λ to the equation $\det(A - \lambda I) = 0$.

  Here *det* is the determinant of the matrix formed by $A - \lambda I$. This equation is called the characteristic equation of A. For example, if A is the following matrix (a so-called diagonal matrix):

$$A = \begin{bmatrix} a_{1,1} & 0 & \cdots & 0 \\ 0 & a_{2,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & a_{n,n} \end{bmatrix},$$

  then the characteristic equation reads

$$
\det(A-\lambda I) = \det\left(\begin{bmatrix} a_{1,1} & 0 & \cdots & 0 \\ 0 & a_{2,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & a_{n,n} \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\right)
$$

$$
= \det \begin{bmatrix} a_{1,1}-\lambda & 0 & \cdots & 0 \\ 0 & a_{2,2}-\lambda & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & a_{n,n}-\lambda \end{bmatrix}
$$

$$
= (a_{1,1}-\lambda)(a_{2,2}-\lambda)\cdots(a_{n,n}-\lambda) = 0.
$$

The solutions to this equation are the eigenvalues $\lambda i = ai, i(i = 1, ..., n)$. The eigenvalue equation for a matrix A can be expressed as
$A\mathbf{x} - \lambda I\mathbf{x} = \mathbf{0}$,
which can be rearranged to $(A - \lambda I)\mathbf{x} = \mathbf{0}$.

A criterion from linear algebra states that a matrix (here: $A\lambda I$) is non-invertible if and only if its determinant is zero, thus leading to the characteristic equation.

Example

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}.$$

The characteristic equation of this matrix reads

$$\det(A - \lambda I) = \det \begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix} = 0.$$

Calculating the determinant, this yields the quadratic equation $\lambda^2 - 4\lambda + 3 = 0$, whose solutions (also called roots) are $\lambda = 1$ and $\lambda = 3$. The eigenvectors for the eigenvalue $\lambda = 3$ are determined by using the eigenvalue equation, which in this case reads

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 3 \begin{bmatrix} x \\ y \end{bmatrix}.$$

This equation reduces to a system of the following two linear equations:

$$\begin{aligned} 2x + y &= 3x, \\ x + 2y &= 3y. \end{aligned}$$

Example

Both equations reduce to the single linear equation $x = y$. Or any vector of the form (x, y) with $y = x$ is an eigenvector to the eigenvalue $\lambda = 3$. However, the vector $(0, 0)$ is excluded. A similar calculation shows that the eigenvectors corresponding to the eigenvalue $\lambda = 1$ are given by non-zero vectors (x, y) such that $y = -x$. For example, an eigenvector corresponding to $\lambda = 1$ is

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

whereas an eigenvector corresponding to $\lambda = 3$ is

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

The singular values, or s-numbers of a compact operator $T : X \to Y$ acting between Hilbert spaces X and Y, are the square roots of the eigenvalues of the nonnegative self-adjoint operator $T^*T : X \to X$ (where T^* denotes the adjoint of T).
The singular values are nonnegative real numbers, usually listed in decreasing order $(s_1(T), s_2(T), ...)$. If T is self-adjoint, then the largest singular value $s1(T)$ is equal to the operator norm of T.
In the case of a normal matrix A (or $A^*A = AA^*$, when A is real then $A^TA = AA^T$), the spectral theorem can be applied to obtain unitary diagonalization of A as $A = U\Lambda U^*$. Therefore, $\sqrt{A^*A} = U|\Lambda|U^*$ and so the singular values are simply the absolute values of the eigenvalues.

- Gaussian elimination
- Norms
- LU-factorization, pivoting