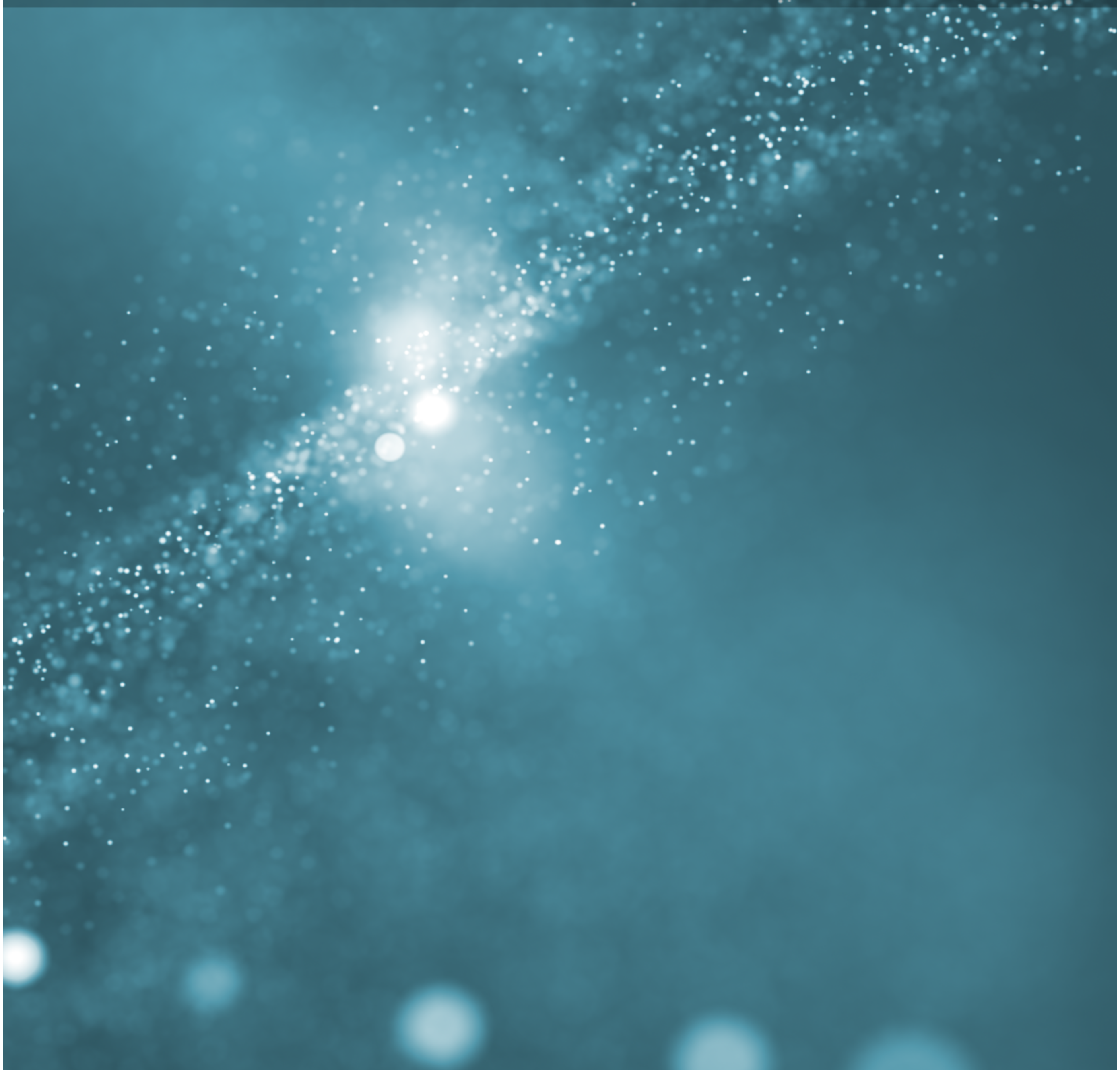
Linear Algebra Lecture Notes

For MATH 1554 at the Georgia Institute of Technology

Greg Mayer

Version 0.4, Compiled April 23, 2021



Preface

These lecture notes are intended for use in a Georgia Tech undergraduate level linear algebra course, MATH 1554. In this first edition of the notes, the focus is on some of the topics not already covered in the Interactive Linear Algebra text.

The document was created using LaTeX and was last compiled on April 23, 2021.

The cover image was obtained from pixabay.com in April 2021 under the Pixabay License.

Contents

Chapter 1

Applications of Matrix Algebra

1.1 Block Matrices

A **block matrix** is a matrix that is interpreted as having been broken into sections called **blocks**, or **submatrices**. Intuitively, a block matrix can be interpreted as the original matrix that is partitioned into a collection of smaller matrices. For example, the matrix

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 2 & 2 & 2 \end{pmatrix}$$

can also be written as a 2×2 partitioned (or block) matrix:

$$A = \begin{pmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{pmatrix}$$

where the entries of A are the blocks

$$A_{1,1} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}, \quad A_{1,2} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad A_{2,1} = \begin{pmatrix} 0 & 0 \end{pmatrix}, \quad A_{2,2} = \begin{pmatrix} 2 & 2 & 2 \end{pmatrix}$$

We partitioned our matrix into four blocks, each of which have different dimensions. But the matrix could also, for example, be partitioned into five 4×1 blocks, or four 1×5 blocks. Indeed, matrices can be partitioned into blocks in many different ways, and depending on the application at hand, there can be a partitioning that is useful or needed.

For example, when solving a linear system $A\vec{x} = \vec{b}$ to determine $\vec{x}$, we can construct and row reduce an augmented matrix of the form

$$X = \begin{pmatrix} A & \vec{b} \end{pmatrix}$$

The augmented matrix X consists of two sub-matrices, A and $\vec{b}$, meaning that it can be viewed as a block matrix. Another application of a block matrix arises when using the SVD, which is a popular tool used in data science. The SVD uses a matrix, Σ, of the form

$$\Sigma = \begin{pmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}$$

Matrix D is a diagonal matrix, and each $\mathbf{0}$ is a zero matrix. Representing Σ in terms of sub-matrices helps us see what the structure of Σ is. Another block matrix arises when introducing a procedure for computing the inverse of an $n \times n$ matrix. To compute the inverse of matrix A, we construct and row reduce the matrix

$$X = \begin{pmatrix} A & I \end{pmatrix}$$

This is an example of a block matrix used in an algorithm. In order to use block matrices in other applications we need to define matrix addition and multiplication with partitioned matrices.

1.1.1 Block Matrix Addition

If $m \times n$ matrices A and B are partitioned in exactly the same way, then the entries of their sum is the sum of their blocks. For example, if A and B are the block matrices

$$A = \begin{pmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{pmatrix}, \quad B = \begin{pmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{pmatrix}$$

then their sum is the matrix

$$A + B = \begin{pmatrix} A_{1,1} + B_{1,1} & A_{1,2} + B_{1,2} \\ A_{2,1} + B_{2,1} & A_{2,2} + B_{2,2} \end{pmatrix}$$

As long as A and B are partitioned in the same way the addition is calculated block by block.

1.1.2 Block Matrix Multiplication

Recall the **row column** method for matrix multiplication.

Theorem

Let A be $m \times n$ and B be $n \times p$ matrix. Then, the (i, j) entry of AB is

$$\operatorname{row}_i A \cdot \operatorname{col}_j B.$$

This is the Row Column Method for matrix multiplication.

Partitioned matrices can be multiplied using this method, as if each block were a scalar provided each block has appropriate dimensions so that products are defined.

1.1.3 Example 1: Computing A^2

Block matrices can be useful in cases where a matrix has a particular structure. For example, suppose A is the $n \times n$ block matrix

$$A = \begin{pmatrix} X & \mathbf{0} \\ \mathbf{0} & Y \end{pmatrix}$$

where X and Y are $p \times p$ matrices, $\mathbf{0}$ is a $p \times p$ zero matrix, and $2p = n$. Then

$$A^2 = AA = \begin{pmatrix} X & \mathbf{0} \\ \mathbf{0} & Y \end{pmatrix} \begin{pmatrix} X & \mathbf{0} \\ \mathbf{0} & Y \end{pmatrix} = \begin{pmatrix} X^2 & \mathbf{0} \\ \mathbf{0} & Y^2 \end{pmatrix}$$

Computation of A^2 only requires computing X^2 and Y^2. Taking advantage of the block structure A leads to a more efficient computation than it otherwise would have been with a naive row-column method that does not take advantage of the structure of the matrix.

1.1.4 Example 2: Computing AB

A and B are the matrices

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \end{pmatrix}$$

$$B = \begin{pmatrix} 2 & -1 \\ 0 & -1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} B_{11} \\ B_{21} \end{pmatrix}$$

where

$$A_{11} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad A_{12} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad B_{11} = \begin{pmatrix} 2 & -1 \\ 0 & -1 \end{pmatrix}, \quad B_{21} = \begin{pmatrix} 0 & 1 \end{pmatrix}$$

If we compute the matrix product using the given partitioning we obtain

$$AB = \begin{pmatrix} A_{11} & A_{12} \end{pmatrix} \begin{pmatrix} B_{11} \\ B_{21} \end{pmatrix} = \begin{pmatrix} A_{11}B_{11} + A_{12}B_{21} \end{pmatrix}$$

where

$$A_{11}B_{11} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & -1 \\ 0 & -1 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ 0 & -1 \end{pmatrix}$$

$$A_{12}B_{21} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}$$

Therefore

$$AB = A_{11}B_{11} + A_{12}B_{21} = \begin{pmatrix} 2 & -1 \\ 0 & -1 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$$

Computing AB with the row column method confirms our result.

$$AB = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 2 & -1 \\ 0 & -1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 2+0+0 & -1+0+1 \\ 0+0+0 & 0-1+1 \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$$

1.1.5 Block Matrix Inversion

In some cases, matrix partitioning can be used to give us convenient expressions for the inverse of a matrix. Recall that the inverse of $n \times n$ matrix A is a matrix B, that has the same dimensions as A and satisfies

$$AB = BA = I$$

where I is the $n \times n$ identity matrix. As we will see in the next example, we can use this equation to construct expressions for the inverse of a matrix.

1.1.6 Example 3: Expression for Inverse of a Block Matrix

Recall, using our formula for a 2×2 matrix,

$$\begin{pmatrix} a & b \\ 0 & c \end{pmatrix}^{-1} = \frac{1}{ac}\begin{pmatrix} c & -b \\ 0 & a \end{pmatrix} = \begin{pmatrix} 1/a & -b/(ac) \\ 0 & 1/c \end{pmatrix} \tag{1.1}$$

provided that $ac \neq 0$. Suppose A, B, and C are invertible $n \times n$ matrices. Suppose we wish to construct an expression for the inverse of the matrix

$$P = \begin{pmatrix} A & B \\ 0 & C \end{pmatrix}$$

To construct the inverse of P, we can write

$$PP^{-1} = P^{-1}P = I_n$$

where P^{-1} is the matrix we seek. If we let P^{-1} be the block matrix

$$P^{-1} = \begin{pmatrix} W & X \\ Y & Z \end{pmatrix}$$

we can determine P^{-1} by solving $PP^{-1} = I$ or $P^{-1}P = I$. Solving $PP^{-1} = I$ gives us:

$$\begin{aligned}
I_n &= PP^{-1} \\
\begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} &= \begin{pmatrix} A & B \\ 0 & C \end{pmatrix} \begin{pmatrix} W & X \\ Y & Z \end{pmatrix} \\
\begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} &= \begin{pmatrix} AW + BY & AX + BZ \\ CY & CZ \end{pmatrix}
\end{aligned}$$

The above matrix equation gives us a set of four equations that can be solved to determine W, X, Y, and Z. The block in the second row and first column gives us $CY = 0$. It was given that C is an invertible matrix, so Y is a zero matrix because

$$\begin{aligned}
CY &= 0 \\
C^{-1}CY &= C^{-1}0 \\
IY &= 0 \\
Y &= 0
\end{aligned}$$

Likewise the block in the second row and second column yields $CZ = I$, so

$$\begin{aligned}
CZ &= I \\
C^{-1}CZ &= C^{-1}I \\
Z &= C^{-1}
\end{aligned}$$

Now that we have expressions for Y and Z we can solve the remaining two equations for W and X. Solving for X gives us the following expression.

$$\begin{aligned}
AX + BZ &= 0 \\
AX + BC^{-1} &= 0 \\
AX &= -BC^{-1} \\
A^{-1}AX &= -A^{-1}BC^{-1} \\
X &= -A^{-1}BC^{-1}
\end{aligned}$$

Solving for W:

$$\begin{aligned} AW + BY &= I \\ AW + B0 &= I \\ A^{-1}AW &= A^{-1}I \\ W &= A^{-1} \end{aligned}$$

We now have our expression for P^{-1}:

$$P^{-1} = \begin{pmatrix} W & X \\ Y & Z \end{pmatrix} = \begin{pmatrix} A^{-1} & -A^{-1}BC^{-1} \\ 0 & C^{-1} \end{pmatrix}$$

Note that in the special case where $n = 2$ that each of the blocks are scalars and our expression is equivalent to Equation (1.1).

1.1.7 Summary

In this section we used partitioned matrices to solve problems regarding matrix invertibility and matrix multiplication. Partitioned matrices can be multiplied using this method, as if each block were a scalar provided each block has appropriate dimensions so that products are defined. They can be used for example when dealing with large matrices that have a known structure where it is more convenient to describe the structure of a matrix in terms of its blocks. Although not part of this text, matrix partitioning can be used to help derive new algorithms because they give a more concise representation of a matrix and of operations on matrices.

1.1.8 Exercises

1. Suppose $A = \begin{pmatrix} Y & X \end{pmatrix} \begin{pmatrix} X & 0 \\ Y & Z \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix}$. Which of the following could A be equal to?

   (a) $A = YX^2 + XYX + XZY$

   (b) $A = 2X + XZY$

(c) $A = YX^2 + X + Z$

2. A, B, and C are $n \times n$ invertible matrices. Construct expressions for X and Y in terms of A, B, and C.

$$\begin{pmatrix} 0 & X & 0 \\ A & 0 & Y \end{pmatrix} \begin{pmatrix} B & 0 \\ 0 & A \\ A & 0 \end{pmatrix} = \begin{pmatrix} 0 & B \\ A & 0 \end{pmatrix}$$

3. Suppose A, B and C are invertible $n \times n$ matrices, and

$$P = \begin{pmatrix} A & 0 \\ B & C \end{pmatrix}$$

Give an expression for P^{-1} in terms of A, B, and C.

1.2 The LU Factorization

To solve a linear system of the form $A\vec{x} = \vec{b}$ we could use row reduction or, in theory, calculate A^{-1} and use it to determine $\vec{x}$ with the equation

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

But computing A^{-1} requires the computation of the inverse of an $n \times n$ matrix, which is especially difficult for large n. It is more practical to solve $A\vec{x} = \vec{b}$ with row reductions (i.e. - Gaussian Elimination). But it turns out that there are more efficient methods, especially when n is large.

One method for solving linear systems that relies on what is referred to as a matrix factorizations. A **matrix factorization**, or **matrix decomposition** is a factorization of a matrix into a product of matrices. Factorizations can be useful for solving $A\vec{x} = \vec{b}$, or for understanding the properties of a matrix.

In this section, we factor a matrix into **lower** and into **upper** triangular matrices to construct what is known as the LU factorization that is used to solve linear systems in a systematic and efficient method. Before we introduce the LU factorization, we will first need to introduce lower and upper triangular matrices.

1.2.1 Triangular Matrices

Before we introduce the LU factorization, we need to first define upper and lower triangular matrices.

Upper and Lower Triangular Matrices

Suppose that the entries of $m \times n$ matrix A are $a_{i,j}$. Then A is **upper triangular** if $a_{i,j} = 0$ for $i > j$. Matrix A is **lower triangular** if $a_{i,j} = 0$ for $i < j$.

As an example, all of the matrices below are in upper triangular form.

$$\begin{pmatrix} 1 & 5 & 0 \\ 0 & 2 & 4 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 2 & 1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Notice how all of the entries below the main diagonal are zero, and the entries on and above the main diagonal can be anything. Likewise, examples of lower triangular matrices are below.

$$\begin{pmatrix} 1 & 0 & 0 \\ 3 & 2 & 0 \end{pmatrix}, \quad \begin{pmatrix} 3 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 \\ 1 & 4 \\ 0 & 1 \\ 2 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Again, note that our definition for an upper triangular matrix does not specify what the entries on or above the main diagonal need to be. Some or all of the entries above the main diagonal can, for example, be zero. Likewise the entries on and below the main diagonal of a lower triangular matrix do not have to have specific values.

1.2.2 The LU Factorization

After stating a theorem that gives the LU decomposition, we will give an algorithm for constructing the LU factorization. We will then see how we can use the factorization to solve a linear system.

Theorem: The LU Factorization

If A is an $m \times n$ matrix that can be row reduced to echelon form without row exchanges, then $A = LU$, where L is a lower triangular $m \times m$ matrix with 1's on the diagonal, and U is an echelon form of A.

Proof

To prove the theorem above we will first show that we can write $A = LU$ where L is an invertible matrix, and U is an echelon form of A.

Suppose that $m \times n$ matrix A can be reduced to echelon form U with p elementary row operations that only add a multiple of a row to another row that is below it. Then each row operation can be performed by multiplying A with p elementary matrices.

$$E_p E_{p-1} \cdots E_3 E_2 E_1 A = U \tag{1.2}$$

If we let $L^{-1} = E_p E_{p-1} \cdots E_3 E_2 E_1$, then

$$L^{-1} A = U \tag{1.3}$$

Note that $L^{-1} = E_p E_{p-1} \cdots E_3 E_2 E_1$ is invertible because elementary matrices are invertible. Therefore L^{-1} can be reduced to the identity with a sequence of row operations. Moreover, if we multiply Equation (1.3) by L we obtain:

$$LL^{-1}A = LU \quad \Rightarrow \quad A = LU$$

Therefore A has the decomposition $A = LU$ where U is an echelon form of A and L is an invertible $m \times m$ matrix. To show that L is lower triangular, recall from equations (1.2) and (1.3) that

$$L^{-1} = E_p E_{p-1} \cdots E_3 E_2 E_1$$

Each elementary matrix E_i is lower triangular because to reduce A to U we only used one type of row operation: adding a multiple of a row to a row below it, so each E_i is a lower triangular matrix. It can also be shown that the product of two lower-triangular matrices is a lower triangular matrix, and the inverse of a lower triangular matrix is lower triangular. This implies that both L^{-1} and L will be lower-triangular.

■

1.2.3 Constructing the LU Factorization

To construct the LU factorization of a matrix we must first apply a sequence of row operations to A in order to reduce A to U. Equation (1.3) gives us that

$$E_p E_{p-1} \cdots E_3 E_2 E_1 A = L^{-1} A = U, \quad \text{where } L^{-1} = E_p E_{p-1} \cdots E_3 E_2 E_1$$

But if $L^{-1}L = I$, then then the sequence of row operations that reduce A to U will reduce L to I. This gives us an algorithm for constructing the LU factorization.

Algorithm: Constructing the LU Factorization of a Matrix

Suppose A is an $m \times n$ matrix that can be row reduced to echelon form without row exchanges. To construct the LU factorization:

1. reduce A to an echelon form U by a sequence of row replacement operations, if possible

2. place entries in L such that the sequence of row operations that reduces A to U will reduce L to I

Note that the above procedure will work for any $m \times n$ matrix that can be reduced to echelon form without row exchanges. Meaning that we do not need A to be square or invertible to construct its LU factorization.

1.2.4 Example 1: LU of a 3×2 Matrix

In this example we construct LU factorizations of the following matrix.

$$A = \begin{pmatrix} 1 & 3 \\ 2 & 10 \\ 0 & 12 \end{pmatrix}$$

Because A is a 3×2 matrix, the LU factorization has the form

$$A = LU = \begin{pmatrix} 1 & 0 & 0 \\ * & 1 & 0 \\ * & * & 1 \end{pmatrix} \begin{pmatrix} * & * \\ 0 & * \\ 0 & 0 \end{pmatrix} \tag{1.4}$$

Each $*$ represents an entry that we need to compute the value of. To reduce A to U we apply a sequence of row replacement operations as shown below.

$$A = \begin{pmatrix} 1 & 3 \\ 2 & 10 \\ 0 & 12 \end{pmatrix} \sim \begin{pmatrix} 1 & 3 \\ 0 & 4 \\ 0 & 12 \end{pmatrix} \sim \begin{pmatrix} 1 & 3 \\ 0 & 4 \\ 0 & 0 \end{pmatrix} = U$$

Matrix U is the echelon form of A that we need for the LU factorization. We next construct L so that the row operations that reduced A to U will reduce L to I. Our row operations were:

$$R_2 - 2R_1 \to R_2 \quad \text{and} \quad R_3 - 3R_2 \to R_3$$

With these two row operations, we see that L must be the matrix:

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix}$$

Note that the row operations $R_2 - 2R_1 \to R_2$ and $R_3 - 3R_2 \to R_3$ applied to L will give us the identity. The LU factorization of A is

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} \begin{pmatrix} 1 & 3 \\ 0 & 4 \\ 0 & 0 \end{pmatrix}$$

1.2.5 Solving Linear Systems with the LU Factorization

Our motivation for introducing the LU factorization was to introduce an efficient method for solving linear systems. Given rectangular matrix A and vector $\vec{b}$, we wish to use the LU factorization of A to solve $A\vec{x} = \vec{b}$ for $\vec{x}$. A procedure for doing so is below.

Algorithm

To solve $A\vec{x} = \vec{b}$ for $\vec{x}$:

1. Construct the LU decomposition of A to obtain L and U.
2. Set $U\vec{x} = \vec{y}$. Forward solve for $\vec{y}$ in $L\vec{y} = \vec{b}$.
3. Backwards solve for $\vec{x}$ in $U\vec{x} = \vec{y}$.

1.2.6 Example 2: Solving a Linear System With LU

In this example we will solve the linear system $A\vec{x} = \vec{b}$ given the LU decomposition of A.

$$A = LU = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 3 & 1 \end{pmatrix} \begin{pmatrix} 1 & 4 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix}, \quad \vec{b} = \begin{pmatrix} 2 \\ 3 \\ 2 \\ 0 \end{pmatrix}$$

We first set $U\vec{x} = \vec{y}$ and solve $L\vec{y} = \vec{b}$. Reducing the augmented matrix $(L \,|\, \vec{b})$ gives us:

$$\left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 1 & 1 & 0 & 0 & 3 \\ 0 & 2 & 1 & 0 & 2 \\ 0 & 0 & 3 & 1 & 0 \end{array}\right) \sim \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 2 & 1 & 0 & 2 \\ 0 & 0 & 3 & 1 & 0 \end{array}\right) \sim \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 3 & 1 & 0 \end{array}\right) \sim \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{array}\right)$$

Therefore, $\vec{y}$ is the vector

$$\vec{y} = \begin{pmatrix} 2 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

We now solve $U\vec{x} = \vec{y}$.

$$\left(\begin{array}{ccc|c} 1 & 4 & 1 & 2 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right) \sim \left(\begin{array}{ccc|c} 1 & 4 & 0 & 2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right) \sim \left(\begin{array}{ccc|c} 1 & 0 & 0 & -2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right)$$

The solution to the linear system, $\vec{x}$, is the vector

$$\vec{x} = \begin{pmatrix} -2 \\ 1 \\ 0 \end{pmatrix}$$

1.2.7 Final Notes on The LU Factorization

In our treatment of the LU factorization we constructed the LU decomposition using the following process.

1. reduce A to an echelon form U by a sequence of row replacement operations, if possible
2. place entries in L such that the same sequence of row operations reduces L to I

There is much more to the LU factorization than what was presented in this section. There are for example other methods for constructing $A = LU$ that you may encounter in future courses or project you are working on. In our approach, the only row operation we use to construct L and U is to replace a row with a multiple of a row above it. Multiplying a row by a non-zero scalar is not needed, but more importantly, we cannot swap rows. More advanced linear algebra and numerical analysis courses would address this significant limitation.

1.2.8 Exercises

1. Construct the LU Factorizations for the following matrices.

   (a) $A = \begin{pmatrix} -1 & 5 & 3 \\ 1 & -10 & -3 \end{pmatrix}$

   (b) $A = \begin{pmatrix} 1 & 5 \\ 2 & 10 \\ 0 & 60 \end{pmatrix}$

   (c) $A = \begin{pmatrix} 2 & 1 & 0 \\ 4 & 3 & 1 \\ 0 & -1 & 2 \end{pmatrix}$

2. Show that the product of two $n \times n$ lower triangular matrices is lower triangular.

3. Show that the inverse of an $n \times n$ lower triangular matrix is also $n \times n$ and lower triangular.

1.3 The Leontif Input-Output Model

Input–output models are used in economics to model the inter-dependencies between different sectors of an economy. Wassily Leontief (1906–1999) is credited with developing the type of analysis that we explore in this chapter. His work on this model earned a Nobel Prize in Economics.

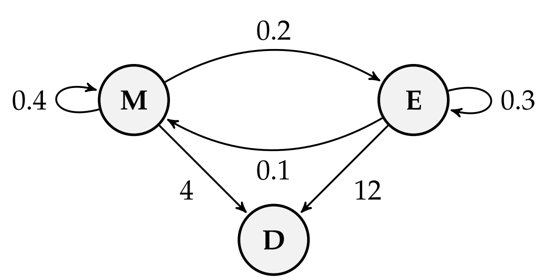
The input-output model assumes that there are sectors in an economy that produce a set of desired products to meet an external demand. The model also assumes that the sectors themselves will also demand a portion of the output that the sectors produce. If the sectors produce exactly the number of units to meet the external demand, then we have the equation

$$(\text{sector output}) - (\text{internal consumption}) = (\text{external demand})$$

In this section we will see that this equation is a linear system that can be solved to determine the output the economy needs to produce to meet the external demand.

1.3.1 Example 1: The Internal Consumption Matrix

Suppose an economy that has two sectors: manufacturing (M) and energy (E). Both of the sectors produce an output to meet an external demand (D) for their products. Sectors M and E also require output from each other to produce their output. The way in which they do so is described in the diagram below.



The numbers in the above diagram can be interpreted as follows.

- For every 100 units that sector M creates, M requires 40 units from M and 10 units from E.
- For every 100 units that sector E creates, E requires 20 units from M and 30 units from E.
- An external demand (D) requires 4 units from M and 12 units from E.

In other words, if M were to create x_M units, then M would consume $0.4x_M$ units from M and $0.1x_M$ units from E. The consumption from sector M could be represented with a vector.

$$\text{consumption from M} = \begin{pmatrix} 0.4x_M \\ 0.1x_M \end{pmatrix} = \frac{x_M}{10}\begin{pmatrix} 4 \\ 1 \end{pmatrix}$$

Likewise, the consumption from sector E would be

$$\text{consumption from E} = \frac{x_E}{10}\begin{pmatrix} 2 \\ 3 \end{pmatrix}$$

Adding these vectors together gives us the total internal consumption from both sectors.

$$\begin{aligned} \text{total internal consumption} &= \frac{x_M}{10}\begin{pmatrix} 4 \\ 1 \end{pmatrix} + \frac{x_E}{10}\begin{pmatrix} 2 \\ 3 \end{pmatrix} \\ &= \frac{1}{10}\begin{pmatrix} 4 & 2 \\ 1 & 3 \end{pmatrix}\begin{pmatrix} x_M \\ x_E \end{pmatrix} \\ &= C\vec{x}, \quad \text{where } C = \frac{1}{10}\begin{pmatrix} 4 & 2 \\ 1 & 3 \end{pmatrix}, \quad \vec{x} = \begin{pmatrix} x_M \\ x_E \end{pmatrix} \end{aligned}$$

Matrix C is called the **consumption matrix**. Typically its entries are between 0 and 1, and the sum of the entries in each column of C will be less than 1. Vector $\vec{x}$ is the output of the sectors. If the sectors produce exactly the number of units to meet the external demand, then we have the equation

$$(\text{sector output}) - (\text{internal consumption}) = (\text{external demand}) \tag{1.5}$$

$$\vec{x} - C\vec{x} = \vec{d} \tag{1.6}$$

In our example, vector $\vec{d} = \begin{pmatrix} 4 \\ 12 \end{pmatrix}$, and $\vec{x} - C\vec{x} = (I - C)\vec{x}$. This simplifies Equation (1.6) to

$$(I - C)\vec{x} = \vec{d} \tag{1.7}$$

$$\left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \frac{1}{10} \begin{pmatrix} 4 & 2 \\ 1 & 3 \end{pmatrix} \right) \begin{pmatrix} x_M \\ x_E \end{pmatrix} = \begin{pmatrix} 4 \\ 12 \end{pmatrix} \tag{1.8}$$

$$\begin{pmatrix} 0.6 & -0.2 \\ -0.1 & 0.7 \end{pmatrix} \begin{pmatrix} x_M \\ x_E \end{pmatrix} = \begin{pmatrix} 4 \\ 12 \end{pmatrix} \tag{1.9}$$

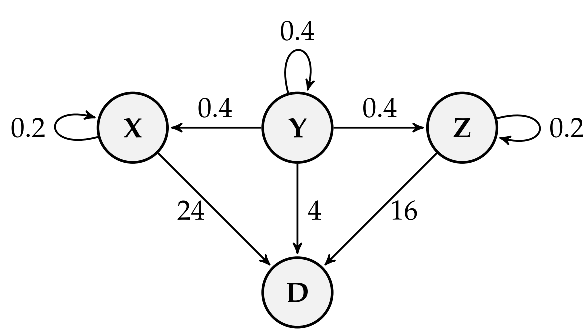
This is a linear system with two equations, whose solution gives us the output vector that balances production with demand. Expressing the system as an augmented matrix and using row operations yields the solution as shown below.

$$\left(\begin{array}{cc|c} .6 & -.2 & 4 \\ -0.1 & 0.7 & 12 \end{array} \right) \sim \left(\begin{array}{cc|c} -1 & 7 & 120 \\ 6 & -2 & 40 \end{array} \right) \sim \left(\begin{array}{cc|c} -1 & 7 & 120 \\ 0 & 40 & 760 \end{array} \right) \sim \left(\begin{array}{cc|c} 1 & 0 & 13 \\ 0 & 1 & 19 \end{array} \right)$$

The unique solution to this linear system is $\vec{x} = \begin{pmatrix} 13 \\ 19 \end{pmatrix}$. This is the output that sectors M and E would need to produce to meet the external demand exactly.

1.3.2 Example 2: An Economy with Three Sectors

Suppose an economy that has three sectors: X, Y, and Z. Each of these sectors produce an output to meet an external demand (D) for their products. The way in which they do so is described in the diagram below.



The external demand, D, is requiring 24 units from X, 4 units from Y, and 16 units from Z. Our goal is to determine how many units the sectors need to produce in order to satisfy this demand, while also accounting for internal consumption.

If Sector X were to create x_X units, then it would consume $0.2x_X$ units from X and $0.4x_X$ units from Y. This consumption could be represented by the vector

$$\text{consumption from Sector X} = \begin{pmatrix} 0.2x_X \\ 0.4x_X \\ 0x_X \end{pmatrix} = \frac{x_X}{10}\begin{pmatrix} 2 \\ 4 \\ 0 \end{pmatrix}$$

Likewise, the consumption from the other two sectors are

$$\text{consumption from Sector Y} = \frac{x_Y}{10}\begin{pmatrix} 0 \\ 4 \\ 0 \end{pmatrix}$$

$$\text{consumption from Sector Z} = \frac{x_Z}{10}\begin{pmatrix} 0 \\ 4 \\ 2 \end{pmatrix}$$

Adding these three vectors together gives us the total internal consumption from all sectors and the consumption matrix C.

$$\begin{aligned} \text{total internal consumption} &= \frac{x_X}{10}\begin{pmatrix} 2 \\ 4 \\ 0 \end{pmatrix} + \frac{x_Y}{10}\begin{pmatrix} 0 \\ 4 \\ 0 \end{pmatrix} + \frac{x_Z}{10}\begin{pmatrix} 0 \\ 4 \\ 2 \end{pmatrix} \\ &= \frac{1}{10}\begin{pmatrix} 2 & 0 & 0 \\ 4 & 4 & 4 \\ 0 & 0 & 2 \end{pmatrix}\begin{pmatrix} x_X \\ x_Y \\ x_Z \end{pmatrix} \\ &= C\vec{x}, \quad \text{where} \quad C = \frac{1}{10}\begin{pmatrix} 2 & 0 & 0 \\ 4 & 4 & 4 \\ 0 & 0 & 2 \end{pmatrix}, \quad \vec{x} = \begin{pmatrix} x_X \\ x_Y \\ x_Z \end{pmatrix} \end{aligned}$$

Each of the sectors in our economy are producing units to satisfy an external demand. The difference between the output and the internal consumption will represent the number of units produced to meet external demand.

$$\begin{aligned} \text{remaining units to meet demand} &= (\text{sector output}) - (\text{internal consumption}) \\ &= \vec{x} - C\vec{x} \\ &= (I - C)\vec{x} \end{aligned}$$

If the sectors are to meet the needs of the external demand exactly, the demand would need to equal the number of units produced after internal consumption is taken into account. That is, we need that

$$(I - C)\vec{x} = \vec{d}$$

This is a linear system that can be solved for the output vector, $\vec{x}$. This could be computed using an augmented matrix.

$$\left(\begin{array}{c|c} I - C & \vec{d} \end{array} \right) = \left(\begin{array}{ccc|c} 0.8 & 0 & 0 & 24 \\ -0.4 & 0.6 & -0.4 & 4 \\ 0 & 0 & 0.8 & 16 \end{array} \right)$$

$$\sim \left(\begin{array}{ccc|c} 8 & 0 & 0 & 240 \\ -4 & 6 & -4 & 40 \\ 0 & 0 & 8 & 160 \end{array} \right)$$

$$\sim \left(\begin{array}{ccc|c} 1 & 0 & 0 & 30 \\ -4 & 6 & -4 & 40 \\ 0 & 0 & 1 & 20 \end{array} \right)$$

$$\sim \left(\begin{array}{ccc|c} 1 & 0 & 0 & 30 \\ 0 & 1 & 0 & 40 \\ 0 & 0 & 1 & 20 \end{array} \right)$$

A helpful trick when reducing these matrices by hand is to multiply each row by 10 to make the algebra a bit less tedious. The above augmented matrix is in row reduced echelon form, and indicates that the desired output is

$$\vec{x} = \begin{pmatrix} 30 \\ 40 \\ 20 \end{pmatrix}$$

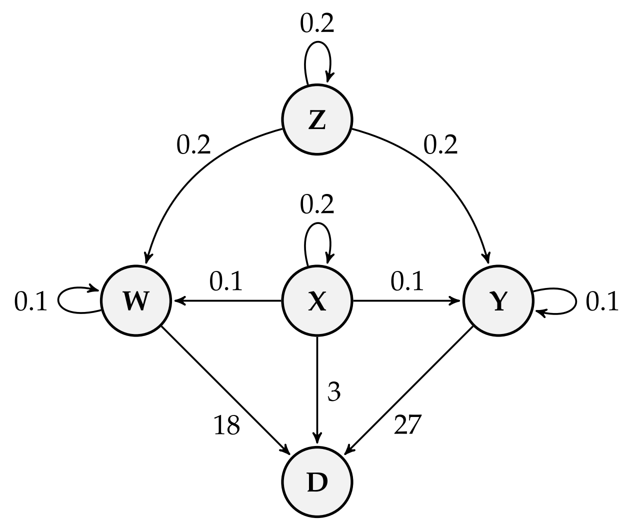
1.3.3 Exercises

1. Consider the production model $\vec{x} = C\vec{x} + \vec{d}$ for an economy with two sectors, where $C = \begin{pmatrix} .0 & .5 \\ .6 & .2 \end{pmatrix}$, and $\vec{d} = \begin{pmatrix} 5 \\ 3 \end{pmatrix}$.

(a) Construct the augmented matrix that can be used to calculate $\vec{x}$.

(b) Solve your linear system for $\vec{x}$.

2. A model for an economy consists of four sectors, W, X, Y, and Z, and an external demand, D. The relationships between them are given in the diagram below.



Sector Z provides resources to the other sectors internally. There is no external demand from D for the output from Z.

(a) Construct the augmented matrix which can be used to solve the system for the output that would meet the external demand exactly while accounting for internal consumption between the four sectors.

(b) Solve your augmented matrix to determine the desired output vector.

1.4 2D Computer Graphics

Linear transformations are often used in computer graphics to simulate the motion of an object. They can be modeled with a matrix-vector product of the form

$$T(\vec{x}) = A\vec{x}$$

where $\vec{x}$ is a vector that represents a point that is transformed to the vector $A\vec{x}$. The matrix-vector product $A\vec{x}$ is a transformation that acts on the vector $\vec{x}$ to produce a new vector, $\vec{b} = A\vec{x}$, and if we set the function $T(\vec{x})$ to be

$$T(\vec{x}) = A\vec{x} = \vec{b}$$

then T maps the vector $\vec{x}$ to vector $\vec{b}$. The nature of the transform is described by matrix A.

Translations are a type of transformation needed in computer graphics. But translations are not a linear transformation because they do not leave the origin fixed. How might we use matrix multiplication in order to perform such transformations? In this section we answer this question by introducing homogeneous coordinates, which allow for more general transformations to be computed with linear algebra.

1.4.1 Homogeneous Coordinates

Homogeneous coordinates are a tool that can be used to model translations.

Definition: Homogeneous Coordinates in $\mathbb{R}^2$

Each point (x, y) in $\mathbb{R}^2$ can be identified with the point $(x, y, 1)$, on the plane in $\mathbb{R}^3$ that lies 1 unit above the xy-plane.

For example, a translation of the form $(x, y) \to (x + h, y + k)$ is a transformation. The parameters h and k adjust the location of the point (x, y) after the transformation. This transform can be represented as a matrix multiplication with homogeneous coordinates in the following way.

$$\begin{pmatrix} 1 & 0 & h \\ 0 & 1 & k \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x+h \\ y+k \\ 1 \end{pmatrix}$$

The first two entries can be extracted from the output of the transform to obtain the coordinate of the translated point. The following examples demonstrate how homogeneous coordinates can be used to create more general transforms.

1.4.2 Example 1: A Composite Transform with Translation

Suppose the transformation $T(\vec{x})$ reflects points in $\mathbb{R}^2$ across the line $x_2 = x_1$ and then translates them by 2 units in the x_1 direction and 3 units in the x_2 direction. In this example we will use homogeneous coordinates to construct a matrix A so that $T = A\vec{x}$.

With homogeneous coordinates the point (x, y) may be represented by the vector

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Points in $\mathbb{R}^2$ can be reflected across the line $x_2 = x_1$ using the standard matrix

$$A_r = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

With homogeneous coordinates our point is represented with a vector in $\mathbb{R}^3$, so we use the block matrix

$$A_1 = \begin{pmatrix} A_r & \mathbf{0} \\ \mathbf{0} & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The symbol **0** denotes a matrix of zeroes. In this case, either a 1×2 matrix or a 2×1 matrix. Then the matrix-vector product below produces the needed transformation.

$$A_1\vec{x} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} y \\ x \\ 1 \end{pmatrix}$$

Note that the x_1 and x_2 coordinates have been swapped, as required for the reflection through the line $x_2 = x_1$. The matrix below will perform the translation we need.

$$A_2 = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix}$$

The product below will apply the translation, of 2 units in the x_1 direction and 3 units in the x_2 direction, to the reflected point.

$$T(\vec{x}) = A_2(A_1\vec{x}) = A_2A_1\vec{x} = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y \\ x \\ 1 \end{pmatrix} = \begin{pmatrix} y+2 \\ x+3 \\ 1 \end{pmatrix}$$

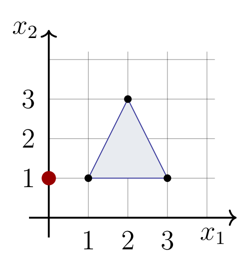
Therfore, our standard matrix is

$$A = A_2A_1 = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 3 \\ 0 & 0 & 1 \end{pmatrix}$$

1.4.3 Example 2: Rotation About the Point $(0, 1)$

Triangle S is determined by the points $(1, 1), (2, 3), (3, 1)$. Transform T rotates these points by $\pi/2$ radians counterclockwise about the point $(0, 1)$. Our goal is to use matrix multiplication to determine the image of S under T.

A sketch of the triangle before and after the rotation is in the diagram below.



We need a way to calculate the locations of the points after the transformation. The rotation can be calculated by first representing each point by a vector in homogeneous coordinates, and then multiplying the vectors by a sequence of matrices that perform the needed transformation. The transformations will first shift the points in a way so that the rotation point is about the origin. We will then rotate about the origin by the desired about. And then we move the rotated points up by one unit to account for the initial translation.

Step 1: Shift Points Down by 1 Unit

In homogeneous coordinates our three points can be represented by the vectors below.

$$\vec{a} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad \vec{b} = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix}, \quad \vec{c} = \begin{pmatrix} 3 \\ 1 \\ 1 \end{pmatrix}$$

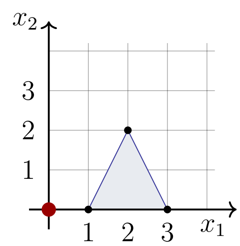
Multiplying each vector by the matrix

$$A_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix}$$

shifts the points down by one unit.

$$\begin{aligned}
\vec{a} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} &\to A_1\vec{a} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \\
\vec{b} = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} &\to A_1\vec{b} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 1 \end{pmatrix} \\
\vec{c} = \begin{pmatrix} 3 \\ 1 \\ 1 \end{pmatrix} &\to A_1\vec{c} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \\ 1 \end{pmatrix}
\end{aligned}$$

Note the difference between the input and output vectors. The second entry of the output vectors is one less than their corresponding entries in the input vectors. Our translated triangle and rotation point is shown below.



With this transform, the rotation point also moves down one unit, from $(0, 1)$ to the origin $(0, 0)$.

Step 2: Rotate About $(0, 0)$

Rotating the translated points by $\pi/2$ radians about the origin can be calulated by multiplying the three vectors by the matrix

$$A_2 = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

This gives us three new points.

$$\vec{a} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \to A_2A_1\vec{a} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$$

$$\vec{b} = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} \to A_2A_1\vec{b} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} -2 \\ 2 \\ 1 \end{pmatrix}$$

$$\vec{c} = \begin{pmatrix} 3 \\ 1 \\ 1 \end{pmatrix} \to A_2A_1\vec{c} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 3 \\ 1 \end{pmatrix}$$

Finally, to undo the initial translation that placed the rotation point at the origin, we need to translate our points up by one unit.

Step 3: Translate Points Up One Unit

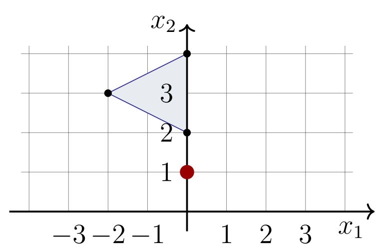
Translating the data up by one unit can be accomplished by multiplying the three vectors by the matrix

$$A_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

This gives us three new points.

$$\vec{a} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \to A_3A_2A_1\vec{a} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ 1 \end{pmatrix}$$
$$\vec{b} = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} \to A_3A_2A_1\vec{b} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} -2 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} -2 \\ 3 \\ 1 \end{pmatrix}$$
$$\vec{c} = \begin{pmatrix} 3 \\ 1 \\ 1 \end{pmatrix} \to A_3A_2A_1\vec{c} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 3 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 4 \\ 1 \end{pmatrix}$$

Our rotated and translated triangle is shown below.



Therefore the standard matrix that performs a rotation by $\pi/2$ degrees about $(0, 1)$ is the matrix

$$A = A_3A_2A_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & -1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

Our result can be verified by calculating $A\vec{a}$, $A\vec{b}$, or $A\vec{c}$.

1.4.4 Example 3: A Reflection Through The Line $x_2 = x_1 + 3$

In this example we construct the 3×3 standard matrix, A, that uses homogeneous coordinates to reflect points in $\mathbb{R}^2$ across the line $x_2 = x_1 + 3$. We will confirm that our results are correct by calculating $T(\vec{x}) = A\vec{x}$ for any point $\vec{x}$ that uses homogeneous coordinates.

The standard matrix A will be the product of three matrices that translate and reflect points using homogeneous coordinates. The first matrix will translate points in some way so that the line about which we are reflecting will pass through the origin. We can use

$$A_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -3 \\ 0 & 0 & 1 \end{pmatrix}$$

This matrix will shift points down three units so that the line $x_2 = x_1 + 3$ will pass through the origin. Note that at this point we could have also used a matrix that, for example, shifts to the right by three units. The second matrix will reflect points through the shifted line, which is $x_2 = x_1$. Recall that the matrix

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

will reflect vectors in $\mathbb{R}^2$ through the line $x_2 = x_1$. This is because any point with coordinates (x_1, x_2) can be represented with the vector

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

and

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ x_1 \end{pmatrix}$$

The point (x_1, x_2) is mapped to (x_2, x_1), which is a reflection through the line $x_2 = x_1$ in $\mathbb{R}^2$. The standard matrix for this transformation in homogeneous coordinates is

$$A_2 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Our final transformation shifts points back up by three units to undo the initial translation.

$$A_3 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 3 \\ 0 & 0 & 1 \end{pmatrix}$$

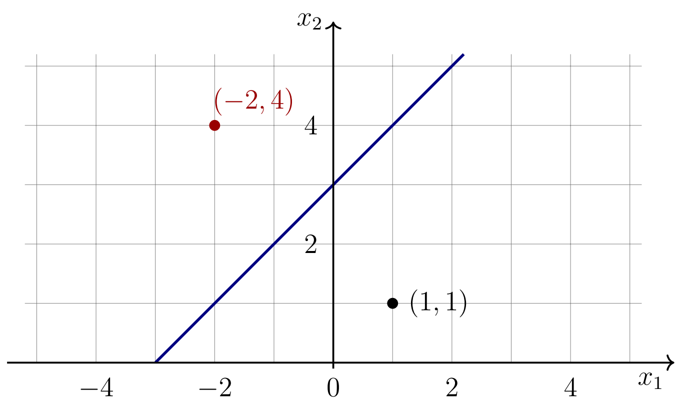
The standard matrix for the transformation that reflects points in $\mathbb{R}^2$ across the line $x_2 = x_1 + 3$ is

$$A = A_3 A_2 A_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 3 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -3 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & -3 \\ 1 & 0 & 3 \\ 0 & 0 & 1 \end{pmatrix}$$

We can check whether our work is correct by transforming any point (x_1, x_2) with the above standard matrix. For example, the point $(1, 1)$ is transformed by calculating

$$T(\vec{x}) = A\vec{x} = \begin{pmatrix} 0 & 1 & -3 \\ 1 & 0 & 3 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -2 \\ 4 \\ 1 \end{pmatrix}$$

The reflected point is $(-2, 4)$. The line of reflection, initial point, and the reflected point are shown below.

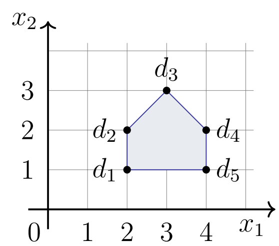


1.4.5 The Data Matrix

The examples in this section have only involved a small number points that need to be transformed. For problems involving many points, it may be more convenient to represent the points in what we refer to as a **data matrix**. For example, the shape in the figure below is determined by five points, or vertices, $d_1, d_2, \ldots, d_5$. Their respective homogeneous coordinates can be stored in the columns of a matrix, D.

$$D = \begin{pmatrix} \vec{d}_1 & \vec{d}_2 & \vec{d}_3 & \vec{d}_4 & \vec{d}_5 \end{pmatrix} = \begin{pmatrix} 2 & 2 & 3 & 4 & 4 \\ 1 & 2 & 3 & 2 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

For our purposes, the order in which the points are placed into D is arbitrary.



In the previous examples we applied a transform with a matrix-vector multiplication. With a data matrix we can use a similar approach. Recall that the product of two matrices A and D, is defined as

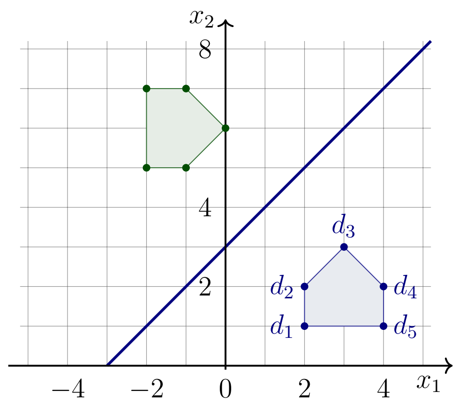
$$AD = A \begin{pmatrix} \vec{d}_1 & \vec{d}_2 & \cdots & \vec{d}_p \end{pmatrix} = \begin{pmatrix} A\vec{d}_1 & A\vec{d}_2 & \cdots & A\vec{d}_p \end{pmatrix}$$

where $\vec{d}_1, \vec{d}_2, \cdots, \vec{d}_p$ are the columns of D. In other words, can perform the transformation on our data by computing AD, which transforms each column independently of the others.

For example, applying the transform in the previous example will reflect our shape through the line $x_2 = x_1 + 3$. The transformation is found by computing

$$AD = \begin{pmatrix} 0 & 1 & -3 \\ 1 & 0 & 3 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 2 & 3 & 4 & 4 \\ 1 & 2 & 3 & 2 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} -2 & -1 & 0 & -1 & -2 \\ 5 & 5 & 6 & 7 & 7 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Extracting the first two entries of each column of the result gives us the transformed points (green), as shown in the figure below.



1.4.6 Exercises

1. Construct the standard matrices for the following transforms.

   (a) The standard matrix of the transform $\vec{x} \to A\vec{x}$ that reflects points in $\mathbb{R}^2$ across the line $x_1 = k$.

   (b) The standard matrix of the transform $\vec{x} \to A\vec{x}$ that rotates points in $\mathbb{R}^2$ about the point $(1, 1)$ and then reflects points through the the line $x_2 = 1$.

1.5 3D Computer Graphics

Results from the previous section on 2D graphics have a natural extension to three dimensions. In this section we extend the data matrix and homogeneous coordinates to three dimensions. This will allow us to model translations and composite transforms involving many points with matrix multiplication.

1.5.1 Rotations in 3D

Rotations about the origin are linear transforms. Because they are linear they can be expressed in the form $T(\vec{x}) = A\vec{x}$ where A is a 3×3 matrix, and we can obtain the columns of matrix A by transforming the standard vectors

$$\vec{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \vec{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad \vec{e}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

We will use the convention that a positive rotation is in the counterclockwise direction when looking toward the origin from the positive half of the axis of rotation. For example, rotating $\vec{e}_1$ about the x_3-axis by θ radians results in the vector

$$T(\vec{e}_1) = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix}$$

Transforming the first standard vector $\vec{e}_1$ yields the first column of A. Likewise the remaining columns can be found by transforming the other standard vectors.

$$T(\vec{e}_2) = \begin{pmatrix} -\sin\theta \\ \cos\theta \\ 0 \end{pmatrix}, \quad T(\vec{e}_3) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

The third standard vector does not change under this transformation because it is parallel to the rotation axis. The standard matrix for a rotation about the x_3-axis is

$$A = \begin{pmatrix} T(\vec{e}_1) & T(\vec{e}_2) & T(\vec{e}_3) \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

A similar analysis gives us the standard matrices for rotations about the x_1 and the x_2 axes. Results are summarized in Table 1.1. The standard matrices in the table can be multiplied together to model transforms that perform multiple transformations. The next example demonstrates this application.

rotation axis	**standard matrix**
x_1-axis	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix}$
x_2-axis	$\begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix}$
x_3-axis	$\begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Table 1.1: Standard matrices for 3D rotations about the coordinate axes.

1.5.2 Example 1: 3D Rotations

Suppose that the transform $\vec{x} \to A\vec{x}$ first rotates points in $\mathbb{R}^3$ about the x_2-axis by $\pi/2$ radians and then rotates points about the x_1-axis by π radians. We can determine the standard matrix, A, for this transform in a few different ways. One approach is to use the standard matrices in Table 1.1. The standard matrix, A, is the product of two rotation matrices.

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\pi & -\sin\pi \\ 0 & \sin\pi & \cos\pi \end{pmatrix} \begin{pmatrix} \cos(\pi/2) & 0 & -\sin(\pi/2) \\ 0 & 1 & 0 \\ \sin(\pi/2) & 0 & \cos(\pi/2) \end{pmatrix} = \begin{pmatrix} 0 & 0 & -1 \\ 0 & -1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

Note that the rotation about the x_2-axis is applied before the rotation about the x_1-axis, which determines the multiplication order. The standard matrix for the first transformation is placed in the rightmost position.

We could also obtain the same result by transforming the standard vectors, because $A = \begin{pmatrix} T(\vec{e}_1) & T(\vec{e}_2) & T(\vec{e}_3) \end{pmatrix}$. The first standard vector gives us the first column of A.

$$\vec{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \to \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \to \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}$$

This result agrees with our result obtained above by multiplying rotation matrices together. Note also that our convention is that a positive rotation is in the counterclockwise direction when looking toward the origin from the positive half of the axis of rotation.

1.5.3 The Data Matrix for 3D Transforms

Similar to the 2D case, for problems involving many points it is convenient to represent the points a data matrix. Analogous to our approach in 2D, points in $\mathbb{R}^3$ can be represented in a matrix whose columns are vectors that correspond to the points we wish to transform. We may transform this matrix with a matrix-vector multiplication. Recall that the product of two matrices A and D, is defined as

$$AD = A \begin{pmatrix} \vec{d}_1 & \vec{d}_2 & \cdots & \vec{d}_p \end{pmatrix} = \begin{pmatrix} A\vec{d}_1 & A\vec{d}_2 & \cdots & A\vec{d}_p \end{pmatrix}$$

where $\vec{d}_1, \vec{d}_2, \cdots, \vec{d}_p$ are the columns of D. In other words, can perform the transformation on our data by computing AD, which transforms each column independently of the others. The following example demonstrates this approach.

1.5.4 Example 2: A Projection in 3D with the Data Matrix

Data in Table (1.2) define a cube in $\mathbb{R}^3$ with side length 1. Suppose the linear transform $T(\vec{x})$ projects points in $\mathbb{R}^3$ onto the x_1x_2-plane. In this example we will construct the matrix, A, that is the standard matrix of the transformation $T(\vec{x}) = A\vec{x}$.

x_1	x_2	x_3
1	1	1
1	2	1
2	2	1
2	1	1
1	1	2
1	2	2
2	2	2
2	1	2

Table 1.2: Corners of a cube with side length 1.

The data in Table (1.2) (blue) and its projection (green) are shown Figure (1.2).

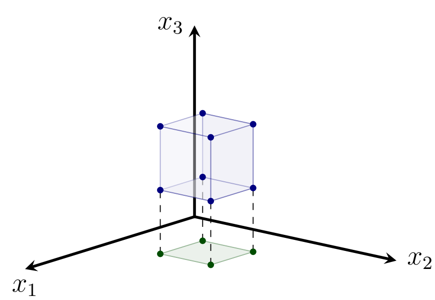


Figure 1.1: Data from Table (1.2) and its projection onto the x_1x_2-plane.

Because the given transform that we are dealing with in this example is linear, we can express the transform in the form of a matrix-vector product

$$T(\vec{x}) = A\vec{x}$$

where A is a 3×3 matrix. Moreover, because we are working with a linear transform, each column of A is equal to the product

$$A\vec{e}_i, \quad i = 1, 2, 3$$

and $\vec{e}_i$ is a standard vector. For example, the first column of A can be found using

$\vec{e}_1$, which is the vector

$$\vec{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Projecting $\vec{e}_1$ onto the x_1x_2-plane does not change the vector, because the vector is already in that plane.

$$\vec{e}_1 \to A\vec{e}_1 = \vec{e}_1 = \text{first column of } A$$

The first column of A is $\vec{e}_1$. Likewise, the second column of A is $\vec{e}_2$, becuase $\vec{e}_2$ is also already in the x_1x_2-plane.

$$\vec{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \to A\vec{e}_2 = \vec{e}_2 = \text{second column of } A$$

The last column of A is the projection of $\vec{e}_3$ onto the plane, which is the zero vector.

$$\vec{e}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \to A\vec{e}_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \text{third column of } A$$

Combining our results for each column of A gives us the standard matrix.

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Now that we have the standard matrix for this transform, we can use it to transform the data in Table 1. Representing each point as a vector in $\mathbb{R}^3$ and placing the vectors in a data matrix, D, will allow us to compute the projection using a matrix multiplication. Our matrix D is

$$D = \begin{pmatrix} 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 \end{pmatrix}$$

The transformed points can be computed as follows.

$$AD = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Extracting the columns of the product gives us the projected points.

1.5.5 3D Homogeneous Coordinates

Homogeneous coordinates in 3D are analogous to the homogeneous 2D coordinates we introduced in the previous section.

Homogeneous Coordinates in $\mathbb{R}^3$

$(X, Y, Z, 1)$ are homogeneous coordinates for (x, y, z) in $\mathbb{R}^3$

A translation of the form $(x, y, z) \to (x + h, y + k, z + l)$ can be represented as a matrix multiplication with homogeneous coordinates:

$$\begin{pmatrix} 1 & 0 & 0 & h \\ 0 & 1 & 0 & k \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x+h \\ y+k \\ z+l \\ 1 \end{pmatrix}$$

Example 3: A Translation in 3D

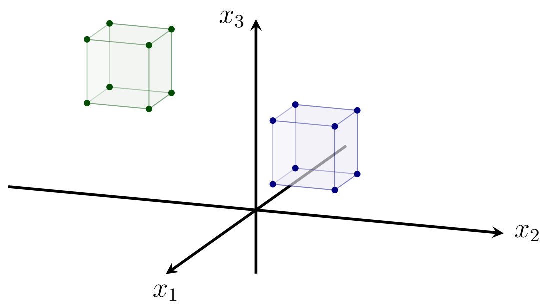
The data in Table (1.2) can be translated using a homogeneous coordinate system. The data matrix D_n in homogeneous coordinates would be

$$D_h = \begin{pmatrix} 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

The transform that, for example, shifts the data by -3 units in the x_2 direction and by 1 unit in the x_3-direction is

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} D_h = \begin{pmatrix} 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 \\ -2 & -1 & -1 & -2 & -2 & -1 & -1 & -2 \\ 2 & 2 & 2 & 2 & 3 & 3 & 3 & 3 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

The figure below shows the original data (blue) and its translated version (green).



1.5.6 Exercises

1. Construct the standard matrices for the following transforms.

   (a) The 4×4 standard matrix of the transform $\vec{x} \to A\vec{x}$ that uses homogeneous coordinates to reflect points in $\mathbb{R}^3$ across the plane $x_3 = k$, where k is any real number.

   (b) The 3×3 standard matrix of the transform $\vec{x} \to A\vec{x}$ that reflects points in $\mathbb{R}^3$ across the plane $x_1 + x_2 = 0$.

   (c) The 3×3 standard matrix of the transform $\vec{x} \to A\vec{x}$ that first rotates points in $\mathbb{R}^3$ about the x_3-axis by an angle θ and then projects them onto the x_2x_3-plane.

2. Line L passes through the point $(1, 0, 0)$ and is parallel to the vector $\vec{v}$, where

$$\vec{v} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

   Construct the 4×4 matrix that uses homogeneous coordinates to rotate points in $\mathbb{R}^3$ about line L by an angle θ.

Chapter 2

Symmetric Matrices and the SVD

2.1 Orthogonal Diagonalization

Many algorithms rely on a type of matrix that is equal to its transpose. If matrix A satisfies $A = A^T$, then A is **symmetric**. A common example of a symmetric matrix is the product A^TA, where A is any $m \times n$ matrix. We use A^TA when, for example, constructing the normal equations in least-squares problems. One way to see that A^TA is symmetric for any matrix A is to take the transpose of A^TA.

$$(A^TA)^T = A^TA^{TT} = A^TA$$

A^TA is equal to its transpose so it must be symmetric. But another way to see that A^TA is symmetric is that for any rectangular matrix A with columns $a_1, \ldots, a_n$, is to express the matrix product using the row-column rule for matrix multiplication.

$$A^TA = \begin{pmatrix} -- & a_1^T & -- \\ -- & a_2^T & -- \\ \vdots & \vdots & \vdots \\ -- & a_n^T & -- \end{pmatrix} \begin{pmatrix} | & | & \cdots & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & \cdots & | \end{pmatrix} = \underbrace{\begin{pmatrix} a_1^Ta_1 & a_1^Ta_2 & \cdots & a_1^Ta_n \\ a_2^Ta_1 & a_2^Ta_2 & \cdots & a_2^Ta_n \\ \vdots & \vdots & \ddots & \vdots \\ a_n^Ta_1 & a_n^Ta_2 & \cdots & a_n^Ta_n \end{pmatrix}}_{\text{entries are dot products of columns of } A}$$

Note that $a_i^Ta_j$ is the dot product between a_i and a_j. And because dot products commute, in other words

$$a_i^T a_j = a_i \cdot a_j = a_j \cdot a_i = a_j^T a_i$$

$A^T A$ is symmetric.

One of the reasons that symmetric matrices are found in many algorithms is that they posses several properties that we can use to make useful or efficient calculations. In this section we investigate some of these properties that symmetric matrices have. In later sections of this chapter we will use these properties to develop and understand algorithms and their results.

2.1.1 Properties of Symmetric Matrices

In this section we give three theorems that characterize symmetric matrices.

1) Symmetric Matrices Have Orthogonal Eigenspaces

The eigenspaces of symmetric matrices have a useful property that we can use when, for example, diagoanlizing a matrix.

> **Theorem**
>
> If A is a symmetric matrix, with eigenvectors $\vec{v}_1$ and $\vec{v}_2$ corresponding to two distinct eigenvalues, then $\vec{v}_1$ and $\vec{v}_2$ are orthogonal.

More generally this theorem implies that eigenspaces associated to distinct eigenvalues are orthogonal subspaces.

Proof

Our approach will be to show that if A is symmetric then any two of its eigenvectors $\vec{v}_1$ and $\vec{v}_2$ must be orthogonal when their corresponding eigenvalues λ_1 and

λ_2 are not equal to each other.

$$\begin{aligned}
\lambda_1\vec{v}_1 \cdot \vec{v}_2 &= A\vec{v}_1 \cdot \vec{v}_2 && \text{using } A\vec{v}_i = \lambda_i\vec{v}_i \\
&= (A\vec{v}_1)^T\vec{v}_2 && \text{using the definition of the dot product} \\
&= \vec{v}_1{}^T A^T\vec{v}_2 && \text{property of transpose of product} \\
&= \vec{v}_1{}^T A\vec{v}_2 && \text{given that } A = A^T \\
&= \vec{v}_1 \cdot A\vec{v}_2 && \text{again using the definition of the dot product} \\
&= \vec{v}_1 \cdot \lambda_2\vec{v}_2 && \text{using } A\vec{v}_i = \lambda_i\vec{v}_i \\
&= \lambda_2\vec{v}_1 \cdot \vec{v}_2
\end{aligned}$$

Rearranging the equation yields

$$0 = \lambda_2\vec{v}_1 \cdot \vec{v}_2 - \lambda_1\vec{v}_1 \cdot \vec{v}_2 = (\lambda_2 - \lambda_1)\vec{v}_1 \cdot \vec{v}_2$$

But $\lambda_1 \neq \lambda_2$ so $\vec{v}_1 \cdot \vec{v}_2 = 0$. In other words, eigenvectors corresponding to distinct eigenvalues must be orthogonal.

■

This theorem can be sometimes be used to quickly identify the eigenvectors of a matrix. For example, if A is a 2×2 matrix and we know that $\vec{v}_1$ is an eigenvector of A, then we can find any non-zero vector orthogonal to $\vec{v}_1$ to identify the eigenvector for the other eigenspace.

2) The Eigenvalues of a Symmetric Matrix are Real

Theorem

If A is a real symmetric matrix then all eigenvalues of A are real.

A proof of this result is in Appendix 1.2.

3) The Spectral Theorem

It turns out that every real symmetric matrix can always be diagonalized using an orthogonal matrix, which is a result of the spectral theorem.

The Spectral Theorem

An $n \times n$ matrix A is symmetric if and only if the matrix can be orthogonally diagonalized.

A proof of this theorem is beyond the scope of these notes, but there are several important consequences of this theorem. All symmetric matrices can not only be diagonalized, but they can be diagonalized with an orthogonal matrix. Moreover, the only matrices that can be diagonalized orthogonally are symmetric, and that if a matrix can be diagonalized with an orthogonal matrix, then it is symmetric.

2.1.2 Examples

Example 1: Orthogonal Diagonalization of a 2×2 Matrix

Suppose A is the symmetric matrix below.

$$A = \begin{pmatrix} 0 & -2 \\ -2 & 3 \end{pmatrix}, \quad \lambda_1 = 4, \quad \lambda_2 = -1$$

The eigenvalues of A are given. In this example we will diagonalize A using an orthogonal matrix, P. For eigenvalue $\lambda_1 = 4$ we have

$$A - \lambda_1 I = \begin{pmatrix} -4 & -2 \\ -2 & -1 \end{pmatrix}$$

A vector in the null space of $A - \lambda_1 I$ is the eigenvector

$$\vec{v}_1 = \begin{pmatrix} 1 \\ -2 \end{pmatrix}$$

A vector orthogonal to $\vec{v}_1$ is

$$\vec{v}_2 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

which must be an eigenvector for λ_2 because A is symmetric.

Dividing each of the eigenvectors by their respective length, and then collecting these unit vectors into a single matrix, P, we obtain an orthogonal matrix. In other words,$P^{-1}P = PP^{-1} = I$. This convenient property gives us a convenient way to compute P^{-1} should it be needed.

Placing the eigenvalues of A in the order that matches the order used to create P, we obtain the factorization

$$A = PDP^T, \quad D = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}, \quad P = \frac{1}{\sqrt{5}} \begin{pmatrix} 1 & 2 \\ -2 & 1 \end{pmatrix}$$

Example 2: Orthogonal Diagonalization of a 3×3 Matrix

In this example we will diagonalize a matrix, A, using an orthogonal matrix, P.

$$A = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \lambda = -1, 1$$

The eigenvalues of A are given. For eigenvalue $\lambda_1 = -1$ we have

$$A - \lambda_1 I = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

A vector in the null space of $A - \lambda_1 I$ is the eigenvector

$$\vec{v}_1 = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

For eigenvalue $\lambda_2 = 1$ we have

$$A - \lambda_2 I = \begin{pmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{pmatrix} \sim \begin{pmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

By inspection, two vectors in the null space of $A - \lambda_2 I$ are

$$\vec{v}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad \vec{v}_3 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$

There are many other choices that we could make but the above two vectors will suffice. Note that $\vec{v}_2$ and $\vec{v}_3$ happen to be orthogonal to each other. If they happened to not be orthogonal, one could use the Gram-Schmidt procedure to make them so.

Dividing each of the three eigenvectors by their respective length, and then collecting these unit vectors into a single matrix, P, we obtain an orthogonal matrix. This will give us a matrix whose inverse is equal to its transpose. In other words, P is an orthogonal matrix, and $P^{-1}P = PP^{-1} = I$. This convenient property gives us a convenient way to compute P^{-1} should it be needed.

Placing the eigenvalues of A in the order that matches the order used to create P, we obtain the factorization

$$A = PDP^T, \quad D = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad P = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 \\ 0 & \sqrt{2} & 0 \\ -1 & 0 & 1 \end{pmatrix}$$

2.1.3 Summary

In this section we explored how we might construct an orthogonal diagonalization of a symmetric matrix, $A = PDP^T$. Note that when a symmetric matrix has a repeated eigenvalue, Gram-Schmidt may be needed when eigenvalues are repeated to construct a full set of orthonormal eigenvectors that span $\mathbb{R}^n$. The theorems we introduced in this section gives us that

- all eigenvalues of A are real
- eigenspaces of A are mutually orthogonal
- A can be diagonalized as $A = PDP^T$

2.1.4 Exercises

1. Suppose A and C are $n \times n$ matrices, $\vec{x} \in \mathbb{R}^n$, and C is symmetric. Which of the following products are equal to a symmetric matrix?

   (a) AA^T

   (b) $\vec{x}\vec{x}^T$

   (c) C^2

2. If $A = PDP^T$ where D is a diagonal matrix and $P^T = P^{-1}$, then is A symmetric?

2.2 Quadratic Forms

Does this inequality hold for all real values of x and y?

$$5x^2 + 8y^2 - 4xy \geq 0$$

Were it not for the $-4xy$ term we could immediately tell that this statement is true. Because if the expression was $5x^2 + 8y^2 \geq 0$, we would more easily see that this can never be negative for any real values of x and y. But could their sum ever be less than $4xy$? Because if it is, then $5x^2 + 8y^2 - 4xy$ would be negative and the inequality would not be true.

After introducing some theory and procedures we will circle back to this motivating problem. Our first step will be to draw a connection to quadratic forms, which allow us to study the above inequality in more general context.

2.2.1 Quadratic Forms

A **quadratic form** is a function $Q : \mathbb{R}^n \to \mathbb{R}$, given by

$$Q(\vec{x}) = \vec{x}^T A \vec{x} = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{12} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Matrix A is $n \times n$ and symmetric and $\vec{x}$ is a vector of variables. If we represent quadratic forms using a symmetric matrix, we can take advantage of their properties to solve problems like the one given at the start of this article. First lets explore a few example of quadratic forms so that we have a better understanding of what they are.

Example 1: Quadratic forms in $\mathbb{R}^2$

In this example we consider the general quadratic form in two variables, $Q(x, y) = \vec{x}^T A\vec{x}$, with

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \quad \vec{x} = \begin{pmatrix} x \\ y \end{pmatrix}$$

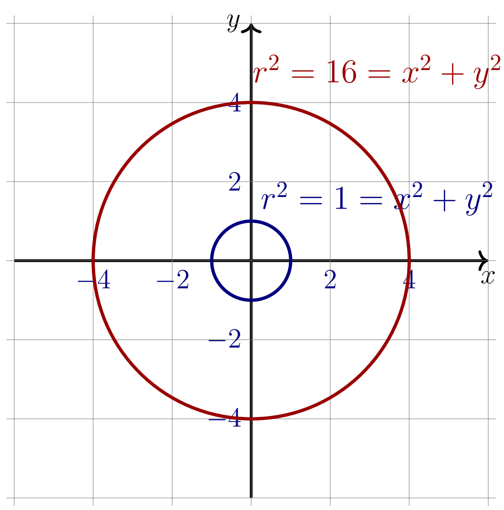
A is symmetric, so we have $a_{12} = a_{21}$, and

$$Q(x, y) = a_{11}x^2 + a_{22}y^2 + 2a_{12}xy$$

A particular example of a quadratic form familiar to many reading this section would be

$$Q(x, y) = \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = x^2 + y^2$$

Setting $Q = r^2$ equal to a constant generates a set of points that create a circle with radius r. Two examples are shown in the diagram below.



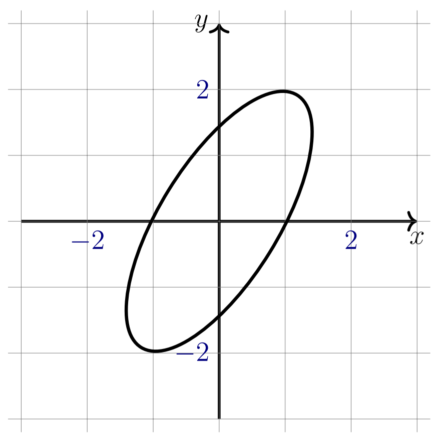
Other choices of Q and the entries in A will create other curves in $\mathbb{R}^2$. For example, $Q = \vec{x}^T A\vec{x}$ with

$$A = \begin{pmatrix} 4 & -2 \\ -2 & 2 \end{pmatrix}$$

generates a set of equations the form $Q = 4x^2 + 2y^2 - 4xy$, because

$$\begin{aligned} Q &= \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} 4 & -2 \\ -2 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ &= \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} 4x - 2y \\ -2x + 2y \end{pmatrix} \\ &= 4x^2 - 2yx - 2xy + 2y^2 \\ &= 4x^2 + 2y^2 - 4xy \end{aligned}$$

If we set $Q = 4$ we obtain the ellipse below.



Example 2: A Quadratic Form

In this example we express $Q = x^2 - 6xy + 9y^2$ in the form $Q = \vec{x}^T A\vec{x}$, where $\vec{x} \in \mathbb{R}^2$ and $A = A^T$. Placing coefficients of x^2 and y^2 on the main diagonal, and dividing coefficient of xy by 2, we obtain

$$x^2 - 6xy + 9y^2 = \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} 1 & -3 \\ -3 & 9 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

We can verify this result by multiplying $\vec{x}^T A\vec{x}$.

Example 3: Quadratic Form in Three Variables

Write Q in the form $\vec{x}^T A\vec{x}$ for $\vec{x} \in \mathbb{R}^3$.

$$Q(\vec{x}) = 5x_1^2 - x_2^2 + 3x_3^2 + 6x_1x_3 - 12x_2x_3$$

Note that we can write Q as

$$Q = 5x_1^2 - x_2^2 + 3x_3^2 + 6x_1x_3 - 12x_2x_3 + 0x_1x_2$$

Taking a similar approach to the previous exercise, we obtain

$$Q = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} \begin{pmatrix} 5 & 0 & 3 \\ 0 & -1 & -6 \\ 3 & -6 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

Again, we can verify this result by multiplying $\vec{x}^T A\vec{x}$.

2.2.2 Principle Axes Theorem

One of the problems we will explore later in this course involves determining the points on a curve of the form $1 = \vec{x}^T A\vec{x}$ that are closest or furthest from the origin. This particular problem will be aided with the Principal Axes Theorem.

Theorem

If A is a symmetric matrix then there exists an orthogonal change of variable $\vec{x} = P\vec{y}$ that transforms $\vec{x}^T A\vec{x}$ to $\vec{y}^T D\vec{y}$ with no cross-product terms.

The proof of this theorem relies on the fact that A is a symmetric matrix and therefore can be diagonalized using an orthogonal matrix.

Proof

Given $Q(\vec{x}) = \vec{x}^T A\vec{x}$, where $\vec{x} \in \mathbb{R}^n$ is a variable vector and A is a real $n \times n$ symmetric matrix. Then we can write

$$A = PDP^T$$

where P is an $n \times n$ orthogonal matrix. A **change of variable** can be represented as

$$\vec{x} = P\vec{y}, \quad \text{or} \quad \vec{y} = P^{-1}\vec{x}$$

With this change of variable, the quadratic form $\vec{x}^{\,T}A\vec{x}$ becomes

$$\begin{aligned} Q = \vec{x}^{\,T}A\vec{x} &= (P\vec{y})^T A(P\vec{y}) \\ &= \vec{y}^{\,T}P^T AP\vec{y} \\ &= \vec{y}^{\,T}D\vec{y}, \quad \text{using } A = PDP^T \end{aligned}$$

Thus, Q is expressed without cross-product terms because D is a diagonal matrix.

2.2.3 Example 4: Change of Variable

Consider the quadratic form

$$Q = \vec{x}^{\,T}A\vec{x}, \quad A = \begin{pmatrix} 5 & 2 \\ 2 & 8 \end{pmatrix}$$

The eigenvalues and eigenvectors of A are given below.

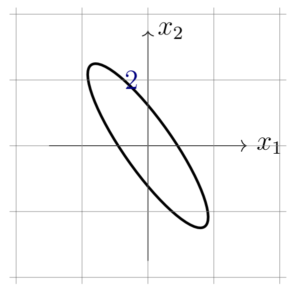
$$\lambda_1 = 9, \quad \lambda_2 = 4, \quad \vec{v}_1 = \begin{pmatrix} 2 \\ -1 \end{pmatrix}, \quad \vec{v}_2 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

We will identify a change of variable that removes the cross-product term. Our change of variable is

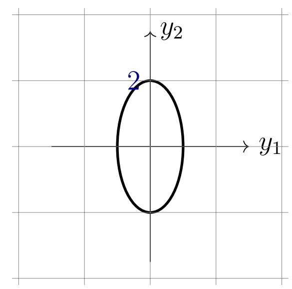
$$\vec{x} = P\vec{y}, \quad P = \frac{1}{\sqrt{5}}\begin{pmatrix} 2 & 1 \\ -1 & 2 \end{pmatrix}$$

Using this change of variable, $Q = \vec{x}^{\,T}A\vec{x} = \vec{y}^{\,T}D\vec{y} = 9y_1^2 + 4y_2^2$.

If, for example, we set $Q = 1$, we obtain two curves in $\mathbb{R}^2$. One curve is x_1x_2-plane, the other in the y_1y_2-plane.



$Q = 5x^2 + 4xy + 8y^2 = 1$



$Q = 9y_1^2 + 4y_2^2 = 1$

Our change of variable can simplify our analysis. For example, in the y_1y_2-plane we can more easily identify points on the ellipse that are closest/furthest from the origin, and determine whether Q can take on negative/positive values.

2.2.4 Example 5: Inequality

We can now return to our motivating question from the start of this section. Does $x^2 - 6xy + 9y^2 \geq 0$ hold for all x, y?

To answer this question we set $Q = 5x^2 - 4xy + 8y^2$.

$$Q = 5x^2 - 4xy + 8y^2 = \vec{x}^T A\vec{x}, \quad A = \begin{pmatrix} 5 & -2 \\ -2 & 8 \end{pmatrix}$$

The characteristic polynomial is $(\lambda - 5)(\lambda - 8) - 4 = (\lambda - 9)(\lambda - 4)$. The eigenvalues therefore are $\lambda_1 = 4$ and $\lambda_2 = 9$. Note that to quickly check that these numbers are, in fact, the eigenvalues of A, we could check whether $A - \lambda_1 I$ and $A - \lambda_2 I$ are singular.

Knowing the eigenvalues of A, we find that

$$Q = \vec{y}^T D\vec{y} = 4y_1^2 + 9y_2^2$$

We see that Q can be zero when $y_1 = y_2 = 0$, but Q is never negative. So the inequality is true.

2.2.5 Summary

We saw how we can express quadratic forms in the form $Q(\vec{x}) = \vec{x}^T A\vec{x}$, for $\vec{x} \in \mathbb{R}^n$. In this section we introduced a representation of quadratic forms with symmetric matrices. We saw how we can express quadratic forms in the form $Q(\vec{x}) = \vec{x}^T A\vec{x}$, for $\vec{x} \in \mathbb{R}^n$ without cross-product terms. We gave a change of variable to represent quadratic forms without cross-product terms and used the Principle Axis Theorem to investigate inequalities involving quadratic forms.

Another one of the reasons we are interested in quadratic forms is because they can be used to describe linear transforms. Consider the transform $\vec{x} \to A\vec{x} = \vec{y}$. The squared length of the vector $\vec{y} = A\vec{x}$ is a quadratic form.

$$\|\vec{y}\| = \|A\vec{x}\|^2 = (A\vec{x}) \cdot (A\vec{x}) = \vec{x}^T A^T A\vec{x}$$

Because $A^T A$ is symmetric, we can use symmetric matrices and their properties to characterize linear transforms. Later in this course we will explore this connection.

2.3 Quadratic Surfaces

In a previous section of these notes we encountered situations where we want to minimize or maximize a quadratic function of the form

$$Q = \vec{x}^T A \vec{x} \tag{2.1}$$

where $A \in \mathbb{R}^{2\times 2}$ is symmetric. Then the set of $\vec{x}$ that satisfies Equation (2.1) We were also interested in additional constraints on what $\vec{x}$ could be. These sorts of problems are encountered, for example, when constructing the singular value decomposition of a matrix, which we will get to soon. Either way, to help us understand these constrained optimization problems it can be helpful to have a geometric interpretation of what Equation (2.1) represents. The interpretations and terminology we introduce in this section can help us describes the shape of Q and solve optimization problems related to it.

2.3.1 Example 1: A Quadratic Surface in $\mathbb{R}^3$

For a fixed Q, Equation (2.1) will define a curve in $\mathbb{R}^2$. For example, if

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

then the points that satisfy

$$Q = \vec{x}^T \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \vec{x} = 2x^2 + 2y^2 + 2xy$$

generates a curve in $\mathbb{R}^2$. The diagram below shows a set of curves for Q equal to 2 and to 8. As we vary the value of Q, the size of our curve will change. In general, when we increase the value of Q, the curve gets larger and points on the curve get further away from the origin. As we decrease Q, the opposite happens: the curve gets smaller and points on the curve get closer to the origin.

If we consider many values of Q we would generate many more curves in $\mathbb{R}^2$. The curves could also be displayed in $\mathbb{R}^3$, with one of the axes corresponding to

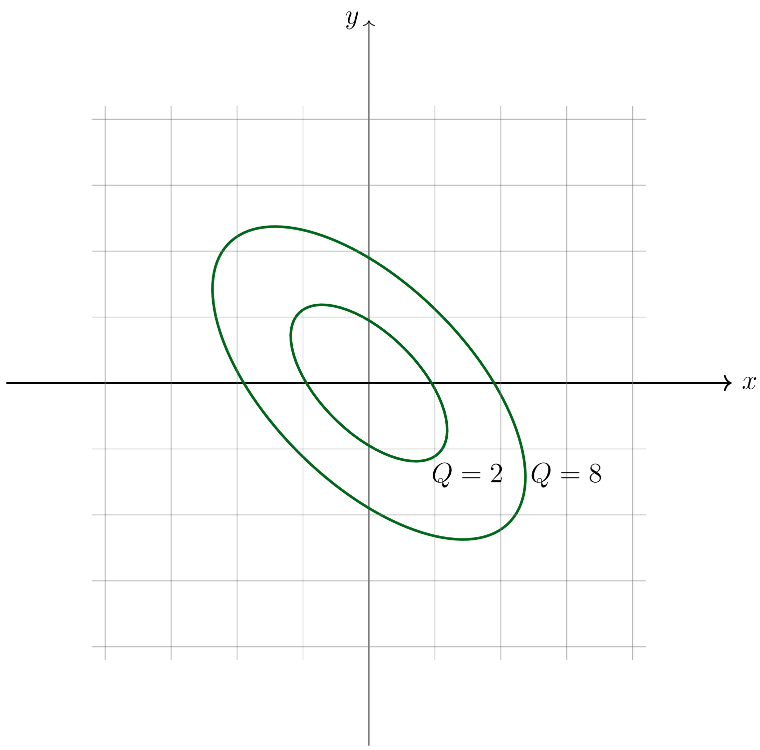


Figure 2.1: Curves generated by $Q = 2x^2 + 2y^2 + 2xy$.

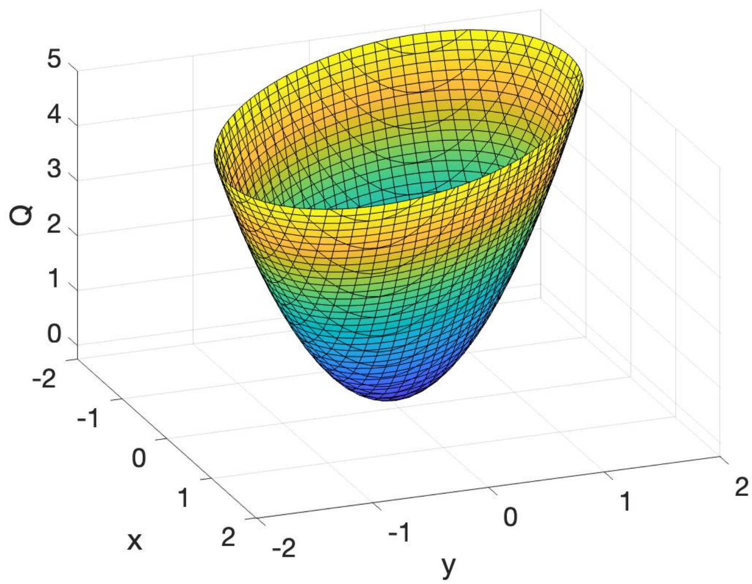


Figure 2.2: The surface $Q = z = 2x^2 + 2y^2 + 2xy$.

Q. In fact, if we allow Q to vary continuously, $Q = 2x^2 + 2y^2 + 2xy$ would give us a surface in $\mathbb{R}^3$, which is shown in Figure (2.2).

Those familiar with MATLAB may be surprised that the above surface can be generated using only a few lines of code. The script that was used to create Figure (2.2) is below. The code uses the fimplicit3 function.

MATLAB Script

```
fimplicit3(@(x,y,Q) Q-2*y.^2-2*x.^2-2*x.*y)
xlabel('x')
ylabel('y')
zlabel('Q')
set(gcf,'color','w'); % sets background color to white
set(gca,'FontSize',18) % increases font size to 18
```

Most of the code above was used to format the diagram. The MATLAB fimplicit3 function plots the three dimensional implicit function defined by $f(x, y, z) = 0$

over a default interval of $[-5, 5]$ for input values of x, y, z. By rearranging Equation (2.1) we can obtain $Q - 2y^2 - 2x^2 - 2xy = 0$ which is the form that MATLAB needs for fimplicit3.

2.3.2 Example 2: Quadratic Surfaces

The entries of A in Equation (2.1) will determine the shape of a quadratic surface that it creates. Several examples are shown in the figures below.

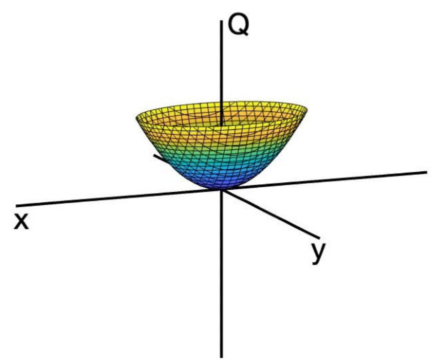


Figure 2.3: $Q = x^2 + y^2$

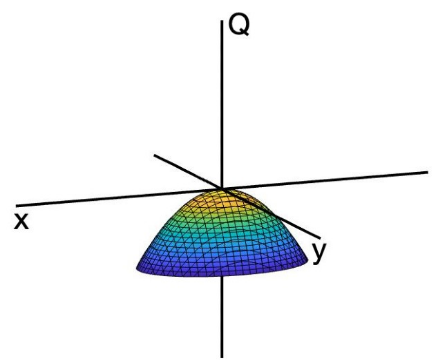


Figure 2.4: $Q = -x^2 - y^2$

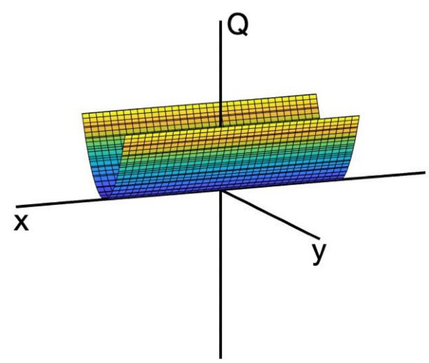


Figure 2.5: $Q = y^2$

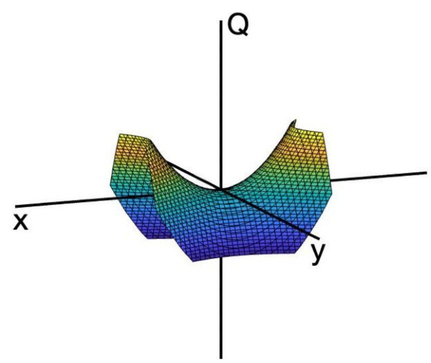


Figure 2.6: $Q = x^2 - y^2$

Notice how some surfaces will have a maximum or minimum value. Figures (2.3) and (2.5) have a minimum value of $Q = 0$. Whereas the form shown in Figure (2.4) has a maximum value $Q = 0$.

2.3.3 Classifying Quadratic Forms

Quadratic functions of the form $Q = \vec{x}^T A\vec{x}$ can be classified based on the values that Q can have.

Definition

A quadratic form Q is

- **positive definite** if $Q > 0$ for all $\vec{x} \neq \vec{0}$.
- **negative definite** if $Q < 0$ for all $\vec{x} \neq \vec{0}$.
- **positive semidefinite** if $Q \geq 0$ for all $\vec{x}$.
- **negative semidefinite** if $Q \leq 0$ for all $\vec{x}$.
- **indefinite** if Q takes on positive and negative values for $\vec{x} \neq \vec{0}$.

That these categories are not mutually exclusive. A form can, for example, be both positive definite and positive semidefinite. The following theorem allows us to classify a form based on the eigenvalues of the matrix of the quadratic form.

Theorem

If A is a symmetric matrix with eigenvalues λ_i, then $Q = \vec{x}^T A\vec{x}$ is

- **positive definite** when all eigenvalues are positive
- **positive semidefinite** when all eigenvalues are non-negative
- **negative definite** when all eigenvalues are negative
- **negative semidefinite** when all eigenvalues are non-positive
- **indefinite** when at least one eigenvalue is negative and at least one eigenvalue is positive

Proof

If A is symmetric, we can write $A = PDP^T$ and set $\vec{y} = P^T\vec{x}$, so $\vec{x} = P\vec{y}$, and

$$
\begin{align}
Q &= \vec{x}^{\,T} A\vec{x} \tag{2.2}\\
&= (P\vec{y})^T A(P\vec{y}), \quad \text{using } \vec{x} = P\vec{y} \tag{2.3}\\
&= \vec{y}^{\,T} P^T AP\vec{y} \tag{2.4}\\
&= \vec{y}^{\,T} D\vec{y}, \quad \text{using } A = PDP^T \tag{2.5}\\
&= \sum \lambda_i y_i^2, \quad \text{because } D \text{ is diagonal} \tag{2.6}
\end{align}
$$

The entries of $\vec{y}$ are y_i. Note that y_i^2 is always non-negative, so for $\vec{y} \neq 0$, the sign of $Q = \sum \lambda_i y_i^2$ will only depend on the values of λ_i. This implies, for example that when $\lambda_i > 0$ for all i, that Q is positive definite.

2.3.4 Example 3: Quadratic Forms and Eigenvalues

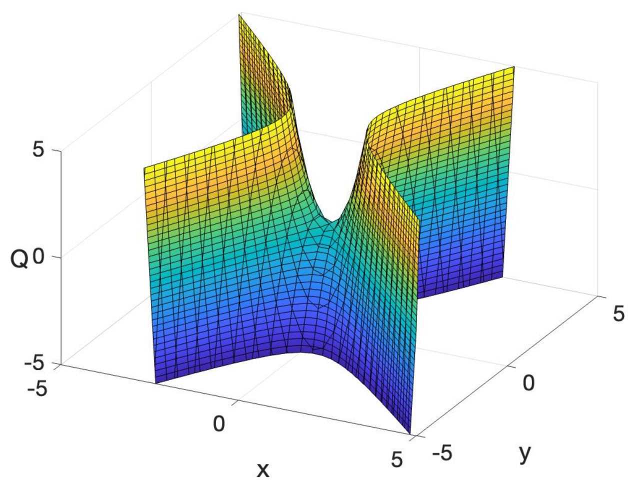
Consider the quadratic form

$$Q = 4x^2 + 2xy - 2y^2 = \vec{x}^{\,T} A\vec{x}$$

The matrix of this quadratic form is

$$A = \begin{pmatrix} 4 & 1 \\ 1 & -2 \end{pmatrix}$$

Calculating its eigenvalues reveals that $\lambda = 1 \pm \sqrt{10}$. Because the eigenvalues are both positive and negative, our quadratic form is indefinite. Indeed, when we plot this surface using MATLAB, we see that the surface does have values that are both positive and negative.



2.3.5 Summary

In this section we explored geometric interpretations of the quadratic form

$$Q = \vec{x}^T A \vec{x} \tag{2.7}$$

where $A \in \mathbb{R}^{2\times 2}$ is symmetric. Then the set of $\vec{x}$ that satisfies this equation create a surface. The surface, Q, could have a minimum or maximum value that may or may not be unique. If all the eigenvalues of A are known, we have seen how we can characterize the extreme values of a quadratic form give by $Q = \vec{x}^T A \vec{x}$.

Those students who have encountered quadratic surfaces in a multivariable calculus course may have already seen the forms discussed in this section from a different perspective. In such a course students may also consider more general quadratic surfaces of the form

$$Q^2 = \vec{x}^T A \vec{x}$$

Such forms can be used to create ellipsoids, cylinders, and other useful shapes that are studied in calculus, but go beyond the scope of this course.

2.4 Constrained Optimization

Symmetric matrices can be found in certain optimization problems involving quadratic functions. By applying some of the properties that symmetric matrices have, we can develop algorithms and theorems to better understand and also solve these optimization problems. The following example demonstrates how symmetric matrices might arise in an optimization problem.

2.4.1 Example 1: Temperature on a Unit Sphere

Suppose that the temperature, Q, on the surface of the sphere, whose radius is one, is given by

$$Q(\vec{x}) = 9x_1^2 + 4x_2^2 + 3x_3^2, \quad \text{where } x_1^2 + x_2^2 + x_3^2 = ||\vec{x}||^2 = 1$$

Our goals are to determine the location of the largest and smallest values of Q on the surface of the sphere, and what the temperature is at these points. To do this, we can first identify the largest value of Q on the sphere.

$$\begin{aligned}
Q(\vec{x}) &= \vec{x}^{\,T} \begin{pmatrix} 9 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 3 \end{pmatrix} \vec{x} \\
&= 9x_1^2 + 4x_2^2 + 3x_3^2 \\
&\leq 9x_1^2 + 9x_2^2 + 9x_3^2 \\
&= 9(x_1^2 + x_2^2 + x_3^2) \\
&= 9\|\vec{x}\|^2 \\
&= 9
\end{aligned}$$

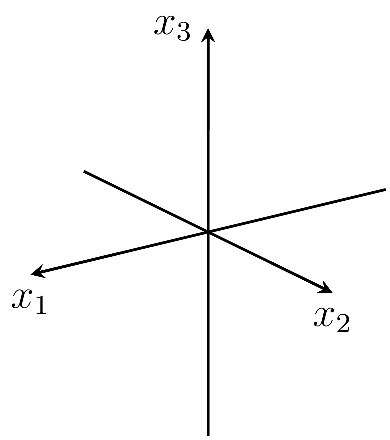
Note that we are only considering points on the surface of the sphere, so $\|\vec{x}\|^2 = 1$. Notice also, by inspection, that Q is equal to 9 at the points $(\pm 1, 0, 0)$. We now have both the maximum value of Q and where the maximum values of Q are located. Therefore,

$$\max\{Q(\vec{x}) : ||\vec{x}|| = 1\} = 9, \text{ and max occurs at } \vec{x} = \begin{pmatrix} \pm 1 \\ 0 \\ 0 \end{pmatrix}$$

A similar analysis yields the minimum value of Q.

$$\min\{Q(\vec{x}) : ||\vec{x}|| = 1\} = 3, \text{ and min occurs at } \vec{x} = \begin{pmatrix} 0 \\ 0 \\ \pm 1 \end{pmatrix}$$

The diagram below shows our unit sphere, colored in a way that gives the temperature of the sphere, with the hottest points being red, and the coldest points being blue.



Note that the hottest points are given by the points $(\pm 1, 0, 0)$ and the coldest points at $(0, 0, \pm 1)$. You may have also noticed that the maximum and minimum values of Q coincide with the eigenvalues of A. We will explore this connection in the next section.

2.4.2 A Constrained Optimization Problem

We will now turn our attention to a more general problem of optimizing a function, Q, on a unit sphere. That is, we wish to identify the maximum or minimum values of

$$Q(\vec{x}) = \vec{x}^T A\vec{x}, \quad \vec{x} \in \mathbb{R}^n, \quad A \in \mathbb{R}^{n\times n},$$

subject to $||\vec{x}|| = 1$. This is an example of a **constrained optimization** problem. Also note that we may also want to know where these extreme values are obtained. The following theorem gives us some insight on how these values can be obtained.

Theorem 2.4.1 Constrained Optimization

If $Q = \vec{x}^T A\vec{x}$, A is a real $n \times n$ symmetric matrix, with eigenvalues

$$\lambda_1 \geq \lambda_2 \ldots \geq \lambda_n$$

and associated normalized eigenvectors $\vec{u}_1, \vec{u}_2, \ldots, \vec{u}_n$. Then, subject to the constraint $||\vec{x}|| = 1$, the **maximum** value of $Q(\vec{x})$ is λ_1, which is attained at $\vec{x} = \pm\,\vec{u}_1$. The **minimum** value of $Q(\vec{x})$ is λ_n, which is attained at $\vec{x} = \pm\,\vec{u}_n$.

Proof

Suppose λ_1 is the largest eigenvalue of A and $\vec{u}_1$ is the corresponding unit eigenvector.

$$\begin{aligned}
Q = \vec{x}^T A\vec{x} = \vec{y}^T D\vec{y}, &\quad \text{using } A = PDP^T,\ \vec{x} = P\vec{y} \\
&= \sum \lambda_i y_i^2, \quad \text{because } D \text{ is diagonal} \\
&\leq \sum \lambda_1 y_i^2, \quad \text{because } \lambda_1 \text{ is the largest eigenvalue} \\
&= \lambda_1 \sum y_i^2 \\
&= \lambda_1\, \|\vec{y}\|^2 = \lambda_1, \quad \text{because } \|\vec{y}\|^2 = 1
\end{aligned}$$

This means that Q is at most λ_1. But $Q = \lambda_1$ at $\pm\vec{u}_1$ because

$$Q(\pm\vec{u}_1) = \vec{u}_1{}^T A\vec{u}_1 = \vec{u}_1{}^T(\lambda_1\vec{u}_1) = \lambda_1$$

Example 2: Constrained Optimization with a Repeated Eigenvalue

In this example we will calculate the maximum and minimum values of $Q(\vec{x}) = \vec{x}^T A\vec{x} = x_1^2 + 2x_2x_3$, $\vec{x} \in \mathbb{R}^3$, subject to $||\vec{x}|| = 1$, and identify points where these values are obtained.

For $Q(\vec{x}) = x_1^2 + 2x_2x_3$, we have

$$Q = \vec{x}^{\,T} A \vec{x}, \quad A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

By inspection, A has eigenvalues $\lambda \pm 1$ (don't forget that an eigenvalue, λ, is a number that makes $A - \lambda I$ singular). For $\lambda = 1$,

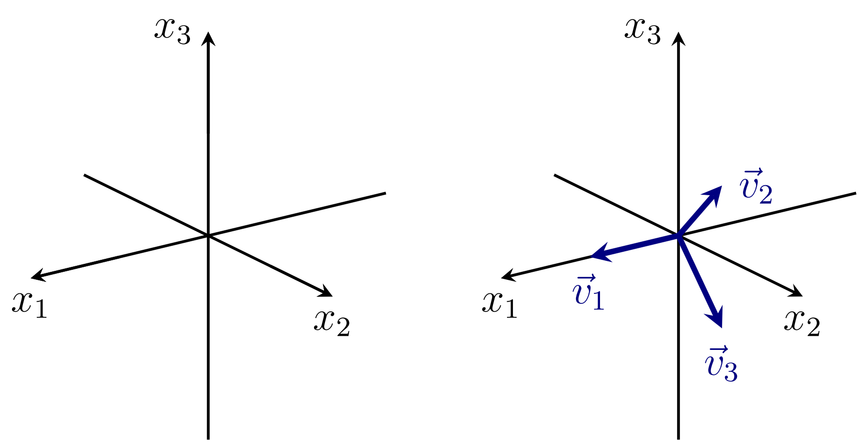
$$A - I = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 1 & -1 \end{pmatrix} \Rightarrow \vec{v}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \vec{v}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$$

Because A is symmetric, the eigenvector for eigenvalue $\lambda = -1$ must be orthogonal to $\vec{v}_1$ and $\vec{v}_2$. So by inspection

$$\vec{v}_3 = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix}$$

Therefore, the minimum value of Q is -1, and is obtained at $\pm\vec{v}_3$. The maximum value of Q is $+1$, which is obtained at any unit vector in the span of $\vec{v}_1$ and $\vec{v}_2$.

The image below is the unit sphere whose surface is colored according to the quadratic from the previous example. Notice the agreement between our solution and the image.



2.4.3 Orthogonality Constraints

Another useful constraint that we will use when constructing the singular value decomposition of a matrix involves orthogonality.

Theorem 2.4.2 Optimization with an Orthogonality Constraint

Suppose $Q = \vec{x}^T A\vec{x}$, where $A \in \mathbb{R}^{n\times n}$ is symmetric and has eigenvalues $\lambda_1 \geq \lambda_2 \ldots \geq \lambda_n$ and associated normalized eigenvectors $\vec{u}_1, \vec{u}_2, \ldots, \vec{u}_n$. Then, subject to the constraints $||\vec{x}|| = 1$ and $\vec{x} \cdot \vec{u}_1 = 0$, the maximum value of $Q(\vec{x})$ is λ_2, which is attained at $\vec{x} = \vec{u}_2$. The minimum value of $Q(\vec{x})$ is λ_n, which is attained at $\vec{x} = \vec{u}_n$.

A proof would go beyond the scope of what we need for these notes, but it could use a similar approach to the theorem that gives the maximum (or minimum) of Q subject to $\|\vec{x}\| = 1$. We could start the proof with a change of variable so that we could express Q using a diagonal matrix and an orthonormal basis for $\mathbb{R}^n$. We could then identify the second largest eigenvalue. The associated eigenvector would be orthogonal to the eigenvector associated with the largest eigenvalue.

Example 3: Optimization with an Orthogonality Constraint

In this example our goal is to identify the maximum value of

$$Q(\vec{x}) = \vec{x}^T A\vec{x}, \quad \text{where } A = A^T, \quad \text{subject to } ||\vec{x}|| = 1 \text{ and } \vec{x} \cdot \vec{u}_1 = 0$$

where $\vec{x} \in \mathbb{R}^3$, A and its eigenvalues are

$$A = \begin{pmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{pmatrix}, \quad \lambda_1 = 0,\ \lambda_2 = -1,\ \lambda_3 = -3$$

and $\vec{u}_1$ is the eigenvector associated with λ_1, where

$$\vec{u}_1 = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

The eigenvector associated with λ_2 is found using the usual process of finding a vector in the nullspace of $A - \lambda_2 I$.

$$A - \lambda_2 I = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

By inspection, a vector in the nullspace will be

$$\begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

But we need to satisfy the constraint $\|\vec{x}\| = 1$, so we will need to normalize our eigenvector to ensure that it has length one. Our unit eigenvector is

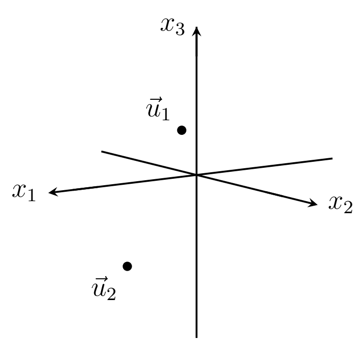
$$\vec{u}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

This eigenvector gives one location where the maximum value of Q is obtained, with the constraints

$$||\vec{x}|| = 1 \text{ and } \vec{x} \cdot \vec{u}_1 = 0$$

The two locations where this maximum are obtained are $(1, 0, -1)$ and $(-1, 0, 1)$. Evaluating Q at either of these points will give us $Q = \lambda_2 = -1$.

The image below is the unit sphere whose surface is colored according to the quadratic from this example.



The set of unit vectors that are orthogonal to $\vec{u}_1$ creates a circle, and the point on that circle with the largest value of Q corresponds to $\vec{u}_2$.

Example 4: Optimization with an Orthogonality Constraint, a Repeated Eigenvalue Case

In this example we will identify the maximum value of $Q(\vec{x}) = \vec{x}^T A\vec{x}$, $\vec{x} \in \mathbb{R}^3$, subject to $||\vec{x}|| = 1$ and to $\vec{x} \cdot \vec{u}_1 = 0$, where

$$Q(\vec{x}) = x_1^2 + 2x_2x_3, \qquad \vec{u}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Noting that this example uses the same quadratic as in Example 2, we know that $\vec{u}_1$ is an eigenvector associated with the largest eigenvalue, $\lambda = 1$. The next largest eigenvalue is $\lambda = -1$, which was a repeated eigenvalue. Any unit vector in the span of $\vec{u}_2$ and $\vec{u}_3$ is also an eigenvector with eigenvalue $\lambda_2 = -1$. If, for example, $\vec{w}$ is a unit vector in the span of $\vec{u}_2$ and $\vec{u}_3$, then $Q(\vec{w}) = \lambda_2 = -1$ and $\vec{w}$ will be orthogonal to $\vec{u}_1$. Therefore the maximum value of $Q(\vec{x}) = \vec{x}^T A\vec{x}$, subject to $||\vec{x}|| = 1$ and to $\vec{x} \cdot \vec{u}_1 = 0$ is $\lambda_2 = -1$.

2.4.4 Summary

In this section we introduced two constrained optimization problems. Our first problems was to identify the maximum/minimum values (and where they are located) of

$$Q(\vec{x}) = \vec{x}^T A\vec{x}, \quad \text{where } A = A^T, \quad \text{subject to } ||\vec{x}|| = 1.$$

We saw that the maximum/minimum values are given by eigenvalues of A, and that the locations of these extreme values are given by the unit eigenvectors of A.

We then explored a related constrained optimization problem. We extended our results from the previous constrained optimization problem to identify the maximum/minimum values of Q, and where they are located, subject to two constraints. If, for example, $\vec{u}_1$ is the eigenvector corresponding to the largest eigenvalue, then our goal was to optimize

$$Q(\vec{x}) = \vec{x}^T A\vec{x}, \quad \text{where } A = A^T, \quad \text{subject to } ||\vec{x}|| = 1 \text{ and } \vec{x} \cdot \vec{u}_1 = 0$$

Again we saw that the maximum/minimum values are given by eigenvalues of A. And we saw that the corresponding **locations** of these extreme values are given by unit eigenvectors of A, but that we needed to use the second largest eigenvalue. These results could then be extended to handle cases with repeated eigenvalues, to add additional orthogonality constraints, or to identify minimum values of Q with orthogonality constraints.

Students who have already completed a multivariable calculus course may recognize constrained optimization problems when working with **Lagrange Multipliers**, which would give a more general framework to approach constrained optimization problems. Lagrange Multipliers are not explored in this particular course. But we will make use of the results in this section when developing the singular value decomposition (SVD) of a matrix.

2.5 Singular Values

If A is any real $m \times n$ matrix, what is the maximum that $\|A\vec{x}\|$ could be equal to, given that $\vec{x}$ has to be a unit vector? It turns out that the answer to this question reveals an application of a constrained optimization problem that we explored in a previous sections, and in the process of answering it we will introduce what are known as the singular values of a matrix and their properties. Singular values are at the heart of many applications of linear algebra. Indeed, singular values play an important role in the singular value decomposition of a matrix, which has many applications.

We will introduce the SVD in a later section of these notes. In this section we motivate their definition and properties by exploring examples involving linear transforms.

2.5.1 Example 1: A Linear Transform on the Unit Circle

Consider the linear transform $\vec{x} \to A\vec{x}$ where

$$A = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 2\sqrt{2} & 0 \\ 0 & \sqrt{2} \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ 2 & 1 \end{pmatrix}$$

The set of all unit vectors in $\mathbb{R}^2$ will form a circle, and this particular transform will map these unit vectors to points on another curve, as shown in Figure (2.7). To help illustrate the problem we are working on, the diagram also shows how a unit vector, $\vec{v}$ is mapped to $A\vec{v}$, where

$$\vec{v} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \to \quad A\vec{v} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 3 \end{pmatrix}$$

There are two questions we want to ask. Is there a unit vector, $\vec{v}$, that will maximize the length of $A\vec{v}$? And what would $A\vec{v}$ be equal to for that particular vector? In other words, which unit vector, $\vec{v}$, maximizes $||A\vec{v}||$, and what is $||A\vec{v}||$ equal to?

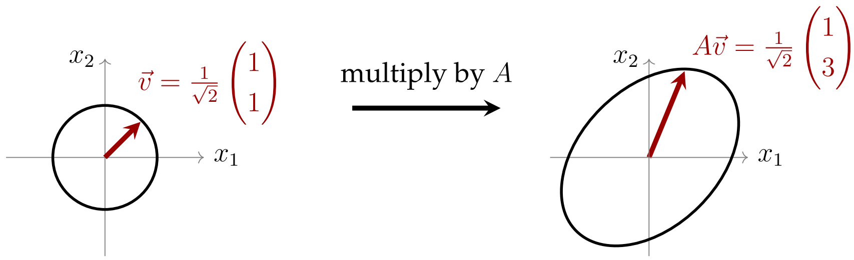


Figure 2.7: The linear transform $\vec{v} \to A\vec{v}$ maps the unit circle to a curve in $\mathbb{R}^2$. An example of one unit vector, $\vec{v}$, and its image, $A\vec{v}$, are also shown.

To answer these questions, it is helpful to use the idea that location of the maximum of $\|A\vec{v}\|$ will be at the same location as the maximum of $\|A\vec{v}\|^2$, subject to our constraint that $\vec{x}$ must be a unit vector. Using this idea, we can then write the squared length as

$$\|A\vec{v}\|^2 = \vec{v}^T A^T A\vec{v}.$$

But $A^T A$ is symmetric. Which means that we are now working with a familiar optimization problem. We know from a previous section that because A is symmetric that we can use the eigenvalues and eigenvectors of $A^T A$ to 1) identify the maximum value of $\|A\vec{v}\|^2$, and 2) identify where the maximum is located. First need to compute $A^T A$.

$$A^T A = \begin{pmatrix} 2 & 2 \\ -1 & 1 \end{pmatrix}\begin{pmatrix} 2 & -1 \\ 2 & 1 \end{pmatrix} = \begin{pmatrix} 8 & 0 \\ 0 & 2 \end{pmatrix} \quad \Rightarrow \quad \lambda = 8, 2$$

In this case, $A^T A$ happens to be diagonal, so the eigenvalues can be obtained by inspection. The largest eigenvalue is 8.

$$\max_{\|\vec{v}\|=1} \|A\vec{v}\|^2 = 8$$

Taking the square root of this gives us the maximum value that we need, which we denote by σ_1.

$$\sigma_1 = \max_{\|\vec{v}\|=1} \|A\vec{v}\| = \sqrt{8}.$$

But what is the vector $\vec{v}$ that corresponds to this maximum? Let the unit eigenvector corresponding to the largest eigenvalue is the vector be $\vec{v}_1$, which maximizes

$\|A\vec{v}\|$ over all unit vectors $\vec{v}$.

$$A^TA - \lambda I = \begin{pmatrix} 0 & 0 \\ 0 & -6 \end{pmatrix} \quad \Rightarrow \quad \vec{v}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Thus the maximum value we found is obtained at the point $(1, 0)$. But of course we could also use $(-1, 0)$, either point will maximize the length of $A\vec{v}$.

If we also wanted to determine the smallest value of $\|A\vec{v}\|$ subject to $\|\vec{v}\| = 1$, we would use the eigenvector that is associated with the smallest eigenvalue of A^TA, which is

$$\vec{v}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Therefore, the minimum value of $\|A\vec{v}\|$ is

$$\sigma_2 = \min_{\|\vec{v}\|=1} \|A\vec{v}\| = \sqrt{2}, \quad \vec{v}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$\|A\vec{v}_2\|$ is the square root of the smallest eigenvalue of A^TA, which is $\sqrt{2}$.

The maximum and minimum lengths of $\|A\vec{v}\|$ are denoted by the Greek letter σ, so that $\sigma_1 = \sqrt{\lambda_1} = \sqrt{8}$, and $\sigma_2 = \sqrt{\lambda_2} = \sqrt{2}$. They are known as the **singular values** of A, and Figure (2.8) shows how they are related to the range of the linear transform $\vec{x} \to A\vec{x}$. We give a definition of the singular values of a matrix in the next section.

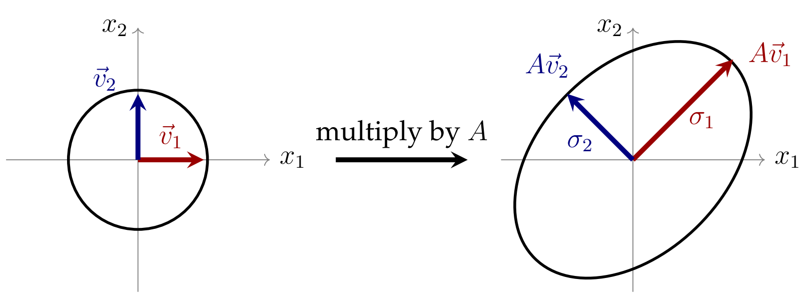


Figure 2.8: The singular values σ_1 and σ_2 are lengths. They give the largest and smallest values of $\|A\vec{v}\|$ subject to $\|\vec{v}\| = 1$, respectively. The vectors $A\vec{v}_1$ and $A\vec{v}_2$ give the locations of these extreme values.

2.5.2 Singular Values

Definition

The singular values, σ_j, of any $m \times n$ real matrix A are the square roots of the eigenvalues of $A^T A$, so that $\sigma_i = \sqrt{\lambda_i}$ for $1 \leq i \leq n$, where λ_i is an eigenvalue of $A^T A$. Singular values are also ordered from largest to smallest, so that

$$\sigma_1 = \sqrt{\lambda_1} \geq \sigma_2 = \sqrt{\lambda_2} \geq \cdots \geq \sigma_n = \sqrt{\lambda_n}.$$

Because we are relying on a square root to define the singular values of a matrix, we might wonder whether the eigenvalues of $A^T A$ could be negative, which would imply that singular values can be complex. But, it turns out that the eigenvalues of $A^T A$ can never be negative because of the following theorem.

Theorem

The eigenvalues of $A^T A$ are real and non-negative.

Proof: We have already shown that the eigenvalues of any symmetric matrix are real (see Appendix (3.1)). Also recall that $\vec{v}_j^T \vec{v}_j = \vec{v}_j \cdot \vec{v}_j = \|\vec{v}_j\|^2 = 1$ because $\vec{v}_j$ are **unit** eigenvectors of $A^T A$. Then

$$\|A\vec{v}_j\|^2 = (A\vec{v}_j)^T A\vec{v}_j = \vec{v}_j A^T A\vec{v}_j = \lambda_j \vec{v}_j^T \vec{v}_j = \lambda_j \geq 0.$$

Therefore the eigenvalues of $A^T A$ must be real and non-negative. And the singular values of A, which are the square roots of the eigenvalues, must also be real and non-negative. ■

2.5.3 Singular Values Represent Lengths

We saw in Example 1 how the singular values of the 2×2 matrix A represented the lengths of $A\vec{v}_1$ and $A\vec{v}_2$. We can extend this concept to any $m \times n$ matrix.

When showing that the eigenvalues of $A^T A$ are non-negative, we saw that

$$||A\vec{v}_i||^2 = \lambda_i$$

Therefore,

$$||A\vec{v}_i|| = \sigma_i$$

This is an important point: the singular value σ_i is the length of $A\vec{v}_i$ for $i = 1, 2, \ldots, n$. Moreover, our proof relied on the fact that because the matrix $A^T A$ is symmetric with non-negative eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$, the eigenvectors of $A^T A$, the set $\{\vec{v}_1, \ldots, \vec{v}_n\}$, forms an orthogonal basis for $\mathbb{R}^n$. In other words, not only is each σ_i is the length of $A\vec{v}_i$, but the lengths are in orthogonal directions.

For example, the largest singular value of A gives us the maximum length of $A\vec{v}$ subject to $\|\vec{v}\| = 1$.

$$\sigma_1 = \max_{\|\vec{v}\|=1} \|A\vec{v}\|$$

The second largest eigenvalue of a symmetric matrix gives the maximum of $\|A\vec{v}\|$ subject to

$$\|\vec{v}\| = 1, \quad \vec{v} \cdot \vec{u}_1 = 0$$

The second largest eigenvalue is σ_2. Thus, σ_2. Likewise with the remaining eigenvalues.

Example 2: Singular Values of a Matrix with Orthogonal Columns

Suppose $T(\vec{x}) = A\vec{x}$ is a linear transform and A is the matrix

$$A = \begin{pmatrix} 2 & 0 & 0 \\ 0 & \sqrt{2}/2 & \sqrt{2}/4 \\ 0 & \sqrt{2}/2 & -\sqrt{2}/4 \end{pmatrix}.$$

The maximum and minimum values of $\|A\vec{x}\|$ are determined from the eigenvalues of $A^T A$. And $A^T A$ is the matrix

$$A^T A = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/4 \end{pmatrix}$$

This matrix happens to be diagonal because A has orthogonal columns. So the eigenvalues can be determined by inspection and are $\lambda_1 = 4$, $\lambda_2 = 1$, and $\lambda_3 = 1/4$. Their square roots are the singular values of A, which are

$$\sigma_1 = \sqrt{\lambda_1} = 2, \quad \sigma_2 = \sqrt{\lambda_2} = 1, \quad \sigma_1 = \sqrt{\lambda_3} = 1/2$$

Shown below, on the left, is the unit sphere in $\mathbb{R}^3$. The vectors that make up the unit sphere are transformed by T: each unit vector $\vec{v}$ is transformed to $A\vec{v}$. The output of the transform is shown on the right.

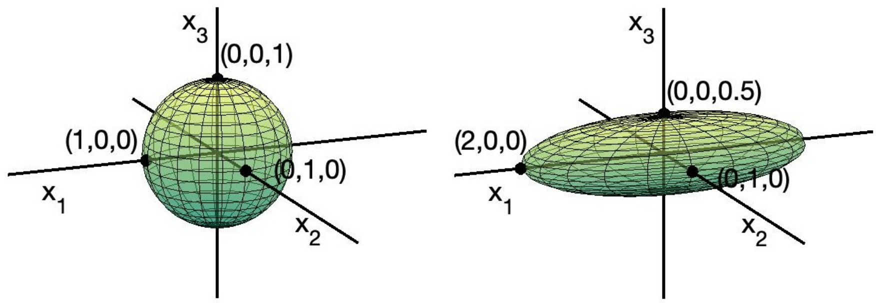


Figure 2.9: The transform of the unit sphere creates an ellipsoid in $\mathbb{R}^3$, whose size is described by the singular values of A.

The maximum value of $\|A\vec{v}\|$ subject to $\|\vec{v}\|^2 = 1$ is $\sigma_1 = 2$, which is why the ellipsoid on the right intersects the x_1 axis at $(2, 0, 0)$ and at $(-2, 0, 0)$. Notice we have chosen the first eigenvalue to be the largest and the last to be the smallest, which is consistent with the convention that the singular values are arranged in decreasing order.

2.5.4 Summary

In this section we saw how the singular values of any $m \times n$ real matrix A are the square roots of the eigenvalues of $A^T A$. They are real and non-negative, arranged in decreasing order, and are related to the lengths of $\|A\vec{x}\|$ for $\|\vec{x}\| = 1$. In the next section we will see how they can be used to construct the singular value decomposition.

2.6 The SVD

The Singular Value Decomposition (SVD) is a factorization that expresses a matrix as a product of three matrices. The decomposition gives us useful information about the subspaces that its rows and columns span, which are used in important applications in data science. In this section, we build on the previous section that introduced singular values to define the singular vectors of a matrix, and show howe can use them to construct the SVD of a matrix.

Earlier in the course we introduced the four fundamental subspaces of a matrix. Recall that for any $A \in \mathbb{R}^{m \times n}$, the orthogonal complement of Row$A$ is Null$A$, and the orthogonal complement of Col$A$ is NullA^T. The relationships between these subspaces in described in Figure (2.10).

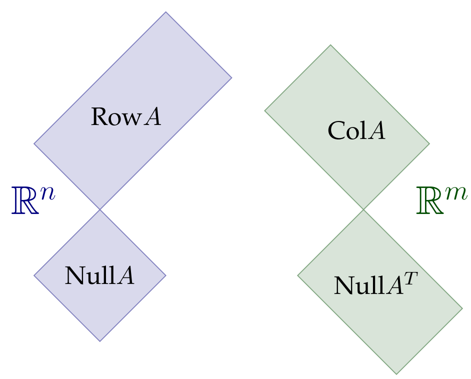


Figure 2.10: The four fundamental subspaces of an $m \times n$ matrix A.

Our introduction to the SVD will begin with showing how the the eigenvectors of A^TA can be used construct an orthonormal bases for Null$A$ and its orthogonal compliment, Row$A$. With a little more work, we can also show how we can use the eigenvectors can be used to calculate bases for Col$A$ and its orthogonal compliment. But we will first look at the bases for Null$A$ and Row$A$, which will give us the right-singular vectors.

2.6.1 Orthogonal Bases for Row$A$ and Null$A$

Theorem: The Right Singular Vectors

Suppose A is an $m \times n$ matrix, and $\vec{v}_i$ are n orthonormal eigenvectors of $A^T A$, ordered so that their corresponding eigenvalues satisfy $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. Suppose also that A has r non-zero singular values, $r \leq n$. Then the set of vectors

$$\{\vec{v}_{r+1}, \vec{v}_{r+2}, \dots, \vec{v}_n\}$$

is an orthonormal basis for Null$A$, and the set

$$\{\vec{v}_1, \vec{v}_2, \dots, \vec{v}_r\}$$

is an orthonormal basis for Row$A$, and rank$A = r$. The vectors $\{\vec{v}_i\}$ for $i \leq n$ are the **right singular vectors** of A.

Proof

First we will show that the right singular vectors $\vec{v}_i$ will form an orthonormal basis for Null$A$ if $i > r$. Recall that for a set of vectors to form an orthogonal basis for a subspace that they must be in that space, span the space, be independent, and mutually orthogonal.Each $\vec{v}_i$ is an eigenvector of $A^T A$. Eigenvectors cannot be zero vectors, so it is possible for them to linearly independent. Moreover, $\vec{v}_i$ must be orthogonal and span $\mathbb{R}^n$ because they are eigenvectors of a symmetric matrix, $A^T A$. Eigenvectors of symmetric matrices that correspond to distinct eigenvalues are orthogonal. Moreover if any eigenvalues of $A^T A$ are repeated then we can use Gram-Schmidt to construct an orthogonal basis for the eigenspace. So the entire set of right singular vectors form an orthogonal basis for $\mathbb{R}^n$.

If the rank of A is less than n, then there must be non-zero vectors $\vec{x}$ so that $A\vec{x} = \vec{0}$. Recall also that the singular values are real, non-negative, arranged in decreasing order, and are the lengths of $A\vec{v}_i$.

$$\|A\vec{v}_i\| = \sigma_i$$

So if $\|A\vec{v}_i\| = 0$ for $i > r$, then $\vec{v}_i \in$ Null$A$ for $i > r$. And if $\|A\vec{v}_i\| \neq 0$ for $i \leq r$, then $\vec{v}_i$ cannot be in Null$A$ for $i \leq r$, they must be in (Null$A$)$^\perp$ = Row$A$, because $\{\vec{v}_i\}$ is an orthonormal set.

Thus, our basis for Null$A$ is the set

$$\{\vec{v}_{r+1}, \vec{v}_{r+2}, \ \dots \ , \vec{v}_n\}$$

and our basis for Row$A$ is the set

$$\{\vec{v}_1, \vec{v}_2, \ \dots \ , \vec{v}_r\}$$

We should also explain why rank$A = r$. There are r vectors in our basis for Row$A$. The number of vectors in a basis for a subspace is the dimension of the subspace. And recall that $\dim(\text{Row}A) = \dim(\text{Col}A) = \text{rank}A$. Thus, rank$A$ is the number of non-zero singular values, r. ■

We will next look at the bases for Col$A$ and $(\text{Col}A)^{\perp}$, which will give us the left singular vectors.

Orthogonal Bases for Col$A$ and NullA^T

Theorem: The Left Singular Vectors

Suppose $\{\vec{v}_i\}$ are the n orthonormal eigenvectors of A^TA, ordered so that their corresponding eigenvalues satisfy $\lambda_1 \geq \lambda_2 \geq \dots \ \geq \lambda_n$. Suppose also that A has r non-zero singular values. Then

$$\{A\vec{v}_1, A\vec{v}_2, \ \dots \ , A\vec{v}_r\}$$

are an orthogonal basis for Col$A$. The vectors $\{\vec{u}_i\}$ for $i \leq m$ are the **left singular vectors** of A.

Proof

The product $A\vec{v}_i$ is just a linear combination of the columns of A weighted by the entries of $\vec{v}_i$, so $A\vec{v}_i$ is a vector in Col$A$. $A\vec{v}_i$ and $A\vec{v}_j$ are orthogonal for $i \neq j$.

$$(A\vec{v}_i) \cdot (A\vec{v}_j) = \vec{v}_i^T A^T A\vec{v}_j = \lambda_j \vec{v}_i \cdot \vec{v}_j = 0$$

For $i \leq r = \text{rank}A$, $A\vec{v}_i$ are orthogonal and non-zero. So they must also independent, and thus they must form an orthogonal basis for Col$A$. Note that for $i > r$, $A\vec{v}_i = \vec{0}$ because $\vec{v}_i \in \text{Null}A$ for $i > r$.

Summary: The Four Fundamental Spaces

Suppose $\vec{v}_i$ are orthonormal eigenvectors for A^TA, and

$$\vec{u}_i = \frac{1}{\sigma_i}A\vec{v}_i \text{ for } i \leq r = \text{rank}A,\ \sigma_i = \|A\vec{v}_i\|.$$

Then we have the following orthogonal bases for any $m \times n$ real matrix A.

- Row$A$: the vectors $\vec{v}_1, \ldots, \vec{v}_r$ are an orthonormal basis for Row$A$. They can be constructed by identifying the eigenvectors of A^TA for $1 \leq i \leq r \leq n$.
- Null$A$: if rank$A < n$, then $\vec{v}_{r+1}, \ldots, \vec{v}_n$ is an orthonormal basis for Null$A$. They can be found by computing the eigenvectors of A^TA for $r < i \leq n$.
- Col$A$: the vectors $\vec{u}_1, \ldots, \vec{u}_r$ are an orthonormal basis for Col$A$. They can be computed using $\sigma_i\vec{u}_i = A\vec{v}_i$ for $1 \leq i \leq r$.
- NullA^T: the vectors $\vec{u}_{r+1}, \ldots, \vec{u}_n$ are an orthonormal basis for NullA^T.

To construct a basis for $(\text{Col}A)^\perp$, we could identify any $m - r$ independent nonzero vectors in $(\text{Col}A)^\perp$ and then use Gram-Schmidt to create an orthogonal basis from them. We will explore that process in an example later on in this section.

2.6.2 The SVD

Having now defined singular values and singular vectors, we are now ready to give a definition of the SVD. Applications of the SVD are given in Section (2.7).

Theorem: Singular Value Decomposition

Suppose A is an $m \times n$ matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$ and $m \geq n$. Then A has the decomposition $A = U\Sigma V^T$ where

$$\Sigma = \begin{pmatrix} D \\ \mathbf{0}_{m-n,n} \end{pmatrix}, \; D = \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_n \end{pmatrix}$$

U is a $m \times m$ orthogonal matrix, and V is a $n \times n$ orthogonal matrix. If $m < n$, then $\Sigma = \begin{pmatrix} D & \mathbf{0}_{m,n-m} \end{pmatrix}$ with everything else the same.

The proof that we can factor any real matrix as $A = U\Sigma V^T$ is similar to one often used to prove that any $n \times n$ matrix with n linearly independent eigenvectors can be diagonalized. We first construct the matrix V from the right singular vectors, by placing them into a matrix as follows.

$$V = (\vec{v}_1 \; \vec{v}_2 \; \dots \vec{v}_n)$$

Then AV becomes

$$AV = A(\vec{v}_1 \; \vec{v}_2 \; \dots \; \vec{v}_n) = (A\vec{v}_1 \; A\vec{v}_2 \; \dots A\vec{v}_n)$$

But $\sigma_i \vec{u}_i = A\vec{v}_i$, and $\sigma_i = \|A\vec{v}_i\|$. So

$$AV = \begin{pmatrix} \sigma_1 \vec{u}_1 & \sigma_2 \vec{u}_2 & \cdots & \sigma_n \vec{u}_n \end{pmatrix} = (\vec{u}_1 \; \vec{u}_2 \; \dots \; \vec{u}_n) \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{pmatrix} = U\Sigma$$

Thus, $AV = U\Sigma$, or $A = U\Sigma V^T$. ■

2.6.3 A Procedure for Constructing the SVD of A

Putting together our definitions for the singular values and singular vectors of a matrix we arrive at a procedure for computing the SVD of a matrix. Suppose A is $m \times n$ and has rank r.

1. Compute the squared singular values of A^TA, which are σ_i^2. Use them to construct the $m \times n$ matrix Σ.

2. Compute the right singular vectors of A, which are $\vec{v}_i$ for $i = 1, 2, \ldots, n$. Then use these vectors to form V.

3. Compute an orthonormal basis for Col$A$ using

$$\vec{u}_i = \frac{1}{\sigma_i} A\vec{v}_i, \quad i = 1, 2, \ldots r$$

   If $r < m$, extend the set $\{\vec{u}_i\}$ to form an orthonormal basis for $\mathbb{R}^m$ and use the basis to form U.

A process for extending the orthonormal basis for $\mathbb{R}^m$ is explored in the examples below.

Example 1: Constructing the SVD of a 4×2 Matrix with Independent Columns

In this example we will construct the singular value decomposition for

$$A = \begin{pmatrix} 2 & 0 \\ 0 & -3 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

We first need to identify the singular values of A and construct Σ. The singular values of A are the eigenvalues of A^TA.

$$A^TA = \begin{pmatrix} 4 & 0 \\ 0 & 9 \end{pmatrix} \quad \Rightarrow \quad \lambda_1 = 9, \quad \lambda_2 = 4$$

The positive square roots of the eigenvalues are the singular values.

$$\sigma_1 = 3,\ \sigma_2 = 2$$

Don't forget that, by convention, σ_1 is the largest singular value of A. Using the singular values of A we can then construct Σ.

$$\sigma_1 = 3,\ \sigma_2 = 2 \ \Rightarrow\ \Sigma = \begin{pmatrix} 3 & 0 \\ 0 & 2 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Next we construct the right-singular vectors $\{\vec{v}_i\}$ so that we can form matrix V.

$$A^TA - \lambda_1 I = \begin{pmatrix} -5 & 0 \\ 0 & 0 \end{pmatrix} \quad \Rightarrow \quad \vec{v}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$
$$A^TA - \lambda_2 I = \begin{pmatrix} 0 & 0 \\ 0 & 5 \end{pmatrix} \quad \Rightarrow \quad \vec{v}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

With the right singular vectors $\vec{v}_1$ and $\vec{v}_2$ we can form V.

$$V = \begin{pmatrix} \vec{v}_1 & \vec{v}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Next we construct left-singular vectors $\{\vec{u}_i\}$ using $\vec{u}_i = \frac{1}{\sigma_i}A\vec{v}_i$ for $i = 1, 2, \ldots r$. Each $\vec{u}_i$ will be a unit vector in $\mathbb{R}^4$.

$$\vec{u}_1 = \frac{1}{\sigma_1}A\vec{v}_1 = \frac{1}{3}\begin{pmatrix} 2 & 0 \\ 0 & -3 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \end{pmatrix}$$
$$\vec{u}_2 = \frac{1}{\sigma_2}A\vec{v}_2 = \frac{1}{2}\begin{pmatrix} 2 & 0 \\ 0 & -3 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

To construct the SVD of A, we must construct the last two columns of U. In this example, A has rank $r = 2$ and U will be a 4×4 orthogonal matrix. Because the columns of U must be orthonormal, and $\vec{u}_1$ and $\vec{u}_2$ were standard vectors, by inspection we can set the last two columns to be

$$\vec{u}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad \vec{u}_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Note that $\vec{u}_3$ and $\vec{u}_4$ are unit vectors, and that $\{\vec{u}_i\}$ are orthonormal. We could have chosen other vectors for $\vec{u}_3$ and $\vec{u}_4$.

We have the SVD of A.

$$A = \begin{pmatrix} 2 & 0 \\ 0 & -3 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 3 & 0 \\ 0 & 2 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Example 2: The SVD of a 3×2 Matrix with Rank 1

In this example we will construct the singular value decomposition of

$$A = \begin{pmatrix} 1 & -1 \\ -2 & 2 \\ 2 & -2 \end{pmatrix}$$

The singular values are found by computing the eigenvalues of $A^T A$.

$$A^T A = \begin{pmatrix} 1 & -2 & 2 \\ -1 & 2 & -2 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ -2 & 2 \\ 2 & -2 \end{pmatrix} = \begin{pmatrix} 9 & -9 \\ -9 & 9 \end{pmatrix}$$

Recall that singular matrices have eigenvalue 0 and the trace of a matrix is the sum of its eigenvalues. So $\lambda_1 = 9 + 9 = 18$ and $\lambda_2 = 0$. The positive square roots of the eigenvalues are the singular values. σ_1 is the largest singular value, and $\sigma_1 = \sqrt{18}$, $\sigma_2 = 0$.

Using the singular values we can construct Σ.

$$\sigma_1 = \sqrt{18} = 3\sqrt{2}, \quad \sigma_2 = 0 \qquad \Rightarrow \qquad \Sigma = \begin{pmatrix} 3\sqrt{2} & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Next we construct the right-singular vectors $\{\vec{v}_i\}$ and form V.

$$A^T A - \lambda_1 I = \begin{pmatrix} -9 & -9 \\ -9 & -9 \end{pmatrix} \quad \Rightarrow \quad \vec{v}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

$$A^T A - \lambda_2 I = \begin{pmatrix} 9 & -9 \\ -9 & 9 \end{pmatrix} \quad \Rightarrow \quad \vec{v}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Thus V is the matrix

$$V = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$$

Next we construct left-singular vectors $\{\vec{u}_i\}$. The rank of A is $r = 1$, so we may use

$$\vec{u}_i = \frac{1}{\sigma_i} A\vec{v}_i$$

for $i = 1$. Vector $\vec{u}_1$ will be a unit vector in $\mathbb{R}^3$.

$$\vec{u}_1 = \frac{1}{\sigma_1} A\vec{v}_1 = \frac{1}{3\sqrt{2}} \begin{pmatrix} 1 & -1 \\ -2 & 2 \\ 2 & -2 \end{pmatrix} \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix} = \frac{1}{3 \cdot 2} \begin{pmatrix} 2 \\ -4 \\ 4 \end{pmatrix} = \frac{1}{3} \begin{pmatrix} 1 \\ -2 \\ 2 \end{pmatrix}$$

How can we construct the remaining left-singular vectors to construct U? In this example, A has rank $r = 1$, and U will be a 3×3 orthogonal matrix. By inspection, two vectors orthogonal to $\vec{u}_1$ are

$$\vec{x}_2 = \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix}, \quad \vec{x}_3 = \begin{pmatrix} -2 \\ 0 \\ 1 \end{pmatrix}$$

Because $\vec{u}_1 \in \mathrm{Col}A$, these two vectors are in $(\mathrm{Col}A)^\perp$, but $\vec{x}_2$ and $\vec{x}_3$ are not orthogonal. U is an orthogonal matrix, so how might we create an orthogonal basis for $(\mathrm{Col}A)^\perp$ using $\vec{x}_2$ and $\vec{x}_3$? The Gram-Schmidt procedure.

$$\bar{u}_2 = \vec{x}_2 = \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix}$$

$$\bar{u}_3 = \vec{x}_3 - \frac{\vec{x}_3 \cdot \bar{u}_2}{\bar{u}_2 \cdot \bar{u}_2} \bar{u}_2 = \begin{pmatrix} -2 \\ 0 \\ 1 \end{pmatrix} - \frac{-4}{5} \begin{pmatrix} -2 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{5} \begin{pmatrix} -2 \\ 4 \\ -5 \end{pmatrix}$$

Normalizing these vectors yields the remaining left singular vectors.

$$\vec{u}_2 = \frac{1}{\sqrt{5}} \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix}, \quad \vec{u}_3 = \frac{1}{\sqrt{45}} \begin{pmatrix} -2 \\ 4 \\ -5 \end{pmatrix}$$

Thus, $A = U\Sigma V^T$, where

$$U = \begin{pmatrix} 1/3 & 2/\sqrt{5} & -2/\sqrt{45} \\ -2/3 & 1/\sqrt{5} & 4/\sqrt{45} \\ 2/3 & 0 & -5/\sqrt{45} \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 3\sqrt{2} & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

$$V = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$$

2.6.4 Summary

In this section we introduced the left and right singular vectors and gave a definition of the full SVD for any $m \times n$ matrix with real entries. A procedure for computing the SVD was given, which can require applying Gram-Schmidt to construct a basis for $\text{Col}A^{\perp}$ to construct U.

2.7 Applications of The SVD

The SVD has been applied to many modern applications in CS, engineering, and mathematics. In this section we will use the SVD to construct bases for the four fundamental subspaces of a matrix, characterize how errors in the entires of A could lead to errors when solving $A\vec{x} = \vec{b}$, and construct a spectral decomposition of a matrix.

2.7.1 The Condition Number of a Matrix

In some applications of linear algebra, entries of A and contain errors. The condition number of a matrix describes the sensitivity that any approach to determining solutions to $A\vec{x} = \vec{b}$ might have to errors in A. These errors could arise from the way the entries of A are stored, or from the way that the entries of A were determined.

Definition

Suppose A is an invertible $n \times n$ matrix. The ratio

$$\kappa = \frac{\sigma_1}{\sigma_n}$$

is the **condition number** of A, where σ_1 is the largest singular value of A, and σ_n is the smallest.

The larger the condition number, the more sensitive the system is to errors.

Example 1: The Condition Number of a 2×2 Matrix

Suppose $A = \begin{pmatrix} 2 & -1 \\ 2 & 1 \end{pmatrix}$. We found in an earlier example that $\sigma_1 = \sqrt{8}, \sigma_2 = \sqrt{2}$. Therefore, the condition number of A is

$$\kappa = \frac{\sigma_1}{\sigma_n} = \frac{\sqrt{8}}{\sqrt{2}} = 2$$

Example 2: The Condition Number of a Matrix with Orthogonal Columns

Suppose $h > 0$ is any positive real number and A is the matrix

$$A = \begin{pmatrix} 2 & h & 1 \\ 1 & 0 & -4 \\ 2 & -h & 1 \end{pmatrix}$$

This matrix happens to have orthogonal columns, so $A^T A$ will be diagonal.

$$A^T A = \begin{pmatrix} 9 & 0 & 0 \\ 0 & 2h^2 & 0 \\ 0 & 0 & 18 \end{pmatrix}$$

Then by inspection the eigenvalues of $A^T A$ are $18, 9$ and $2h^2$. If h is a small positive number close to zero, the condition number is the ratio

$$\kappa = \frac{\sigma_1}{\sigma_3} = \frac{\sqrt{18}}{\sqrt{2h^2}} = \frac{3}{h}$$

As h tends to zero σ_3 also tends to zero, and κ tends to infinity. This suggests that when h is very small, the process of solving the linear system $A\vec{x} = \vec{b}$ would be sensitive to any errors made in calculating the solution. And that there could be a relationship between how close A is to a singular matrix and the sensitivity to errors in calculating the solution to $A\vec{x} = \vec{b}$. A more detailed discussion would certainly go beyond the scope of our discussion.

2.7.2 The SVD and the Four Fundamental Subspaces

Suppose $\vec{v}_i$ are orthonormal eigenvectors for $A^T A$, and

$$\vec{u}_i = \frac{1}{\sigma_i} A\vec{v}_i \ \text{ for } i \leq r = \text{rank}A, \ \sigma_i = \|A\vec{v}_i\|.$$

Then we have the following bases for any $m \times n$ real matrix A.

- $\vec{v}_1, \dots, \vec{v}_r$ is an orthonormal basis for Row$A$.
- $\vec{v}_{r+1}, \dots, \vec{v}_n$ is an orthonormal basis for Null$A$.
- $\vec{u}_1, \dots, \vec{u}_r$ is an orthonormal basis for Col$A$.
- $\vec{u}_{r+1}, \dots, \vec{u}_m$ is an orthonormal basis for Col$A^\perp$ = NullA^T.

Example 3: Constructing Orthonormal Bases for the Fundamental Subspaces

Suppose A is a 4×5 matrix whose SVD is given below.

$$A = U\Sigma V^T = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 5 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & \sqrt{0.8} & 0 & -\sqrt{0.2} \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & \sqrt{0.8} & 0 & \sqrt{0.2} \end{pmatrix}$$

There are exactly three non-zero singular values, so $r = \text{rank}A = 3$. The first three columns of V are a basis for Row$A$ and the last two columns of V (or rows of V^T) are a basis for Null$A$. The first three columns of U are a basis for Col$A$, and the remaining column of U are a basis for $(\text{Col}A)^\perp$.

2.7.3 The Spectral Decomposition of a Matrix

The SVD can be also used to approximate any $m \times n$ real matrix with what is defined as the spectral decomposition. Note that a similar decomposition for symmetric matrices that uses the orthogonal decomposition of a matrix is described in Appendix 3.2.

Spectral Decomposition

For any real $m \times n$ matrix A with rank r and SVD $A = U\Sigma V^T$ we can write

$$A = U\Sigma V^T = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^{\,T} = \sigma_1 \vec{u}_1 \vec{v}_1^T + \sigma_2 \vec{u}_2 \vec{v}_2^T + \ldots + \sigma_r \vec{u}_r \vec{v}_r^T$$

Vectors $\vec{u}_i, \vec{v}_i$ are the i^{th} columns of U and V respectively.

Here we give a short explanation on why A has the decomposition

$$A = \lambda_1 \vec{u}_1 \vec{u}_1^T + \cdots + \lambda_n \vec{u}_n \vec{u}_n^T = \sum_{i=1}^{n} \lambda_i \vec{u}_i \vec{u}_i^T$$

If the columns of Σ are $\vec{s}_1, \vec{s}_2, \cdots, \vec{s}_n$, then, using the definition of matrix multiplication,

$$U\Sigma = U\begin{pmatrix}\vec{s}_1 & \vec{s}_2 & \cdots & \vec{s}_n\end{pmatrix} = \begin{pmatrix}U\vec{s}_1 & U\vec{s}_2 & \cdots & U\vec{s}_n\end{pmatrix}$$

Recall that a matrix times a vector is a linear combination of the columns of the matrix weighted by the entries of the vector. Column i of the product $U\Sigma$ is

$$U\Sigma_i = U\begin{pmatrix}0\\ \vdots \\ 0 \\ \lambda_i \\ 0 \\ \vdots\end{pmatrix} = 0 + 0 + \ldots + 0 + \lambda_i\vec{u}_i + 0 + \ldots = \lambda_i\vec{u}_i$$

Therefore, the columns of PD are $\lambda_i\vec{u}_i$. We can now simplify our expression for $A = PDP^T$ to a product of two $n \times n$ matrices. A can be expressed as

$$A = U\Sigma V^T = \begin{pmatrix}\sigma_1\vec{u}_1 & \sigma_2\vec{u}_2 & \cdots & \sigma_n\vec{u}_n\end{pmatrix}\begin{pmatrix}\vec{v}_1^T \\ \vec{v}_2^T \\ \vdots \\ \vec{v}_n^T\end{pmatrix}$$

Using the **column-row expansion** for the product of two matrices, this becomes

$$A = \sigma_1\vec{u}_1\vec{v}_1^T + \cdots + \sigma_n\vec{u}_n\vec{v}_n^T = \sum_{i=1}^{n}\sigma_i\vec{u}_i\vec{v}_i^T$$

The row-column expansion for the product of two matrices is a way of defining matrix multiplication. Note that each term in this sum is a rank 1 matrix. In applications of linear algebra, σ_i can become sufficiently small, allowing us to approximate A with a small number of rank 1 matrices.

Example 4: The Spectral Decomposition of a 4×2 Matrix

Suppose A has the following SVD.

$$A = \begin{pmatrix}2 & 0 \\ 0 & -3 \\ 0 & 0 \\ 0 & 0\end{pmatrix} = \begin{pmatrix}0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1\end{pmatrix}\begin{pmatrix}3 & 0 \\ 0 & 2 \\ 0 & 0 \\ 0 & 0\end{pmatrix}\begin{pmatrix}0 & 1 \\ 1 & 0\end{pmatrix}$$

The spectral decomposition of A is as follows.

$$\begin{aligned}
A &= \sum_{s=1}^{r} \sigma_s \vec{u}_s \vec{v}_s^T \\
&= 3 \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \end{pmatrix} + 2 \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \end{pmatrix} \\
&= 3 \begin{pmatrix} 0 & 0 \\ 0 & -1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} + 2 \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}
\end{aligned}$$

Example 5: The Spectral Decomposition of a 4×5 Matrix

In this example we construct and discuss the spectral decomposition of the 4×5 matrix

$$A = \begin{pmatrix} 1 & -2 & 2 & -2 & -4 \\ 0 & 2 & 0 & 2 & 4 \\ 2 & 1 & 4 & 1 & 2 \\ 2 & 0 & 4 & 0 & 0 \end{pmatrix}$$

With some help from MATLAB we obtain the eigenvalues of $A^T A$ and the singular values of A.

$$\begin{aligned}
\lambda_1 &= 54 \quad \Rightarrow \quad \sigma_1 = \sqrt{54} = 3\sqrt{6} \approx 7.3485 \\
\lambda_2 &= 45 \quad \Rightarrow \quad \sigma_2 = \sqrt{45} = 3\sqrt{5} \approx 6.7082
\end{aligned}$$

The script used to compute the eigenvalues are below.

$$\vec{u}_1 = \frac{1}{3} \begin{pmatrix} 2 \\ -2 \\ -1 \\ 0 \end{pmatrix}, \quad \vec{u}_2 = \frac{1}{3} \begin{pmatrix} -1 \\ 0 \\ -2 \\ -2 \end{pmatrix}, \quad \vec{v}_1 = \frac{1}{\sqrt{6}} \begin{pmatrix} 0 \\ -1 \\ 0 \\ -1 \\ -2 \end{pmatrix}, \quad \vec{v}_2 = \frac{1}{\sqrt{5}} \begin{pmatrix} -1 \\ 0 \\ -2 \\ 0 \\ 0 \end{pmatrix}$$

The MATLAB script that computes the eigenvalues of $A^T A$ and the SVD of A is shown below.

MATLAB Script

```
A = [1 -2 2 -2 -4;0 2 0 2 4;2 1 4 1 2;2 0 4 0 0];
AA = A'*A;  % compute A^TA
L = eig(AA); % L will be a vector containing the eigenvalues of A^TA
[U,S,V] = svd(A); % this will give us the SVD of matrix A
```

Using the SVD of A we can construct the spectral decomposition of A is as follows. Note that in this case that the number of non-zero singular values is $r = \text{rank}A = 2$.

$$\begin{aligned}
A &= \sum_{s=1}^{r} \sigma_s \vec{u}_s \vec{v}_s^T \\
&= \sigma_1 \vec{u}_1 \vec{v}_1^T + \sigma_2 \vec{u}_2 \vec{v}_2^T \\
&= \frac{3\sqrt{6}}{3\sqrt{6}} \begin{pmatrix} 2 \\ -2 \\ -1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 & -1 & 0 & -1 & -2 \end{pmatrix} + \frac{3\sqrt{5}}{3\sqrt{5}} \begin{pmatrix} -1 \\ 0 \\ -2 \\ -2 \end{pmatrix} \begin{pmatrix} 1 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 2 \\ -2 \\ -1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 & -1 & 0 & -1 & -2 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \\ -2 \\ -2 \end{pmatrix} \begin{pmatrix} -1 & 0 & -2 & 0 & 0 \end{pmatrix}
\end{aligned}$$

Notice how this representation of A only requires two vectors with four entries and two vectors with 5 entries each, for a total of 18 numbers. The original matrix is 4×5, which requires 20 numbers. The spectral decomposition happened to give us a slightly more concise representation of A. But we were only able to obtain this result because there a few non-zero singular values. Had there been three or four non-zero singular singular values, the spectral decomposition would have not been as concise. More generally, the reduction in the amount of data needed to represent a matrix will be greater with larger matrices that also have a small number of non-zero singular values.

2.7.4 Summary

In this section we introduced three applications of the SVD and singular values.

- The four fundamental subspaces of a matrix can be calculated using the left and right singular vectors.
- The condition number of a square invertible matrix, $\kappa = \frac{\sigma_1}{\sigma_n}$, characterizes the sensitivity to solving $A\vec{x} = \vec{b}$ to errors in A.
- The spectral decomposition of a matrix can be calculated using its SVD.

A similar decomposition for symmetric matrices that uses the orthogonal decomposition of a matrix is described in Appendix 3.2.

While there are numerous applications of the SVD, many of them would be beyond the scope of this course and would require introducing topics and problems that would go beyond the scope of a linear algebra course.

Chapter 3

Appendices

3.1 Symmetric Matrices Have Real Eigenvalues

First we will show that when A is a real symmetric matrix that for any $x \in \mathbb{C}^n$ that the quantity $Q = \bar{x}^T A x$ is real.

To show that Q is real, we will make use of the complex conjugate. We use the overbar notation to denote complex conjugate, so if a and b are real, then the complex number $z = a + ib$ has complex conjugate $\overline{z} = a - ib$. Moreover, if $\overline{z} = z$, then $b = 0$ which implies that z must be a real number. We use this idea to show that $\overline{Q} = Q$.

$$\begin{aligned} \overline{Q} &= \overline{\overline{x}^T A x} \\ &= \overline{\overline{x}}^T\, \overline{Ax} \\ &= x^T\, \overline{Ax} \\ &= x^T A \overline{x} \end{aligned}$$

The last step uses the assumption that A is real, so that $\overline{A} = A$. But $\overline{Q}$ is a number, so $\overline{Q}^T = \overline{Q}$ and

$$\overline{Q} = (\overline{Q})^T = (x^T A \overline{x})^T = \overline{x}^T (x^T A)^T = Q$$

Because $\overline{Q} = Q$, we have shown that $Q = \overline{x}^T Ax$ is real for any $x \in \mathbb{C}^n$ and real symmetric $n \times n$ matrix A.

Next we use this result to show that when $x = v_j$ is an eigenvector of A that Q is equal to an eigenvalue of A.

$$Q = \overline{x}^T Ax = \overline{v}_j{}^T Av_j = \overline{v}_j{}^T(\lambda_j v_j) = \lambda_j v_j \cdot v_j$$

But $v_j \cdot v_j$ is real and Q is real, so λ_j must also be real.

3.2 The Spectral Decomposition of a Symmetric Matrix

We have seen how any symmetric matrix can be diagonalized as $A = PDP^T$, where

$$A = PDP^T = \begin{pmatrix} \vec{u}_1 & \cdots & \vec{u}_n \end{pmatrix} \begin{pmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{pmatrix} \begin{pmatrix} \vec{u}_1^T \\ \vdots \\ \vec{u}_n^T \end{pmatrix}$$

The columns of P are the eigenvectors of A, and the entries on the main diagonal of D are the corresponding eigenvalues. Following the same proof for the spectral decomposition of a matrix using the SVD, it can be shown that A has the decomposition

$$A = \lambda_1 \vec{u}_1 \vec{u}_1^T + \cdots + \lambda_n \vec{u}_n \vec{u}_n^T = \sum_{i=1}^{n} \lambda_i \vec{u}_i \vec{u}_i^T$$

We will give a brief explanation on why A has the decomposition given below.

$$A = \lambda_1 \vec{u}_1 \vec{u}_1^T + \cdots + \lambda_n \vec{u}_n \vec{u}_n^T = \sum_{i=1}^{n} \lambda_i \vec{u}_i \vec{u}_i^T$$

We assume that we can write $A = PDP^T$. If the columns of D are $d_1, d_2, \ldots, d_n$, then, using the definition of matrix multiplication,

$$PD = \begin{pmatrix} Pd_1 & Pd_2 & \ldots Pd_n \end{pmatrix}$$

Recall that a matrix times a vector is a linear combination of the columns of the matrix weighted by the entries of the vector. Column i of PD is

$$Pd_i = P \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \lambda_i \\ 0 \\ \vdots \end{pmatrix} = 0 + 0 + \ldots + 0 + \lambda_i \vec{u}_i + 0 + \ldots = \lambda_i \vec{u}_i$$

Therefore, the columns of PD are $\lambda_i \vec{u}_i$. We can now simplify our expression for $A = PDP^T$ to a product of two $n \times n$ matrices.

Thus, A can be expressed as follows.

$$A = PDP^T = \begin{pmatrix} \lambda_1 \vec{u}_1 & \lambda_2 \vec{u}_2 & \cdots & \lambda_n \vec{u}_n \end{pmatrix} \begin{pmatrix} \vec{u}_1^T \\ \vec{u}_2^T \\ \vdots \\ \vec{u}_n^T \end{pmatrix}$$

Using the **column-row expansion** for the product of two matrices, this becomes

$$A = \lambda_1 \vec{u}_1 \vec{u}_1^T + \cdots + \lambda_n \vec{u}_n \vec{u}_n^T = \sum_{i=1}^{n} \lambda_i \vec{u}_i \vec{u}_i^T$$

The row-column expansion for the product of two matrices is a way of defining matrix multiplication.