SINGULAR VALUE DECOMPOSITION

Dissertion submitted in the partial fulfillment of the requirement for the

MASTER'S DEGREE IN MATHEMATICS

By

AARDRA RAJESH

Reg.no.200111014834

Under the Guidance of

Dr.LAKSHMI C

Department of Mathematics

BHARATA MATA COLLEGE,THRIKAKARA

(Affiliated to Mahatma Gandhi University,Kottayam)

2021-2023

DECLARATION

I Aardra Rajesh hereby declare that this project entitled **'SINGULAR VALUE DECOMPOSITION'** is a bonafide record of work done by me under the guidence of Dr.Lakshmi C, Associate Professor,Department of Mathematics,Bharata Mata College,Thrikkakara and this work has not previously formed by the basis for the award of any academic qualification,fellowship or other similar title of any other University or Board.

Aardra Rajesh
Department of Mathematics
Bharata Mata College,Thrikkakara

Place:Thrikkakara

Date:

CERTIFICATE

This is to certify that the project entitled **'SINGULAR VALUE DECOMPOSITION'** submitted for the partial fulfilment requirement of Master's Degree in Mathematics in the original work done by Aardra Rajesh during the period of the study in the Department of Mathematics,Bharata Mata College,Thrikkakara under my guidance and has not been included in any other project submitted previously for the award of any degree.

Dr.Lakshmi C
Supervisor

Place:Thrikkakara
Date:

ACKNOWLEDGEMENT

First and foremost I express my sincere gratitude towards the abiding presence and grace of God Almighty.

I sincerely express my gratitude to Dr.Lakshmi C ,Associate Professor, Department of Mathematics,Bharata Mata College,Thrikkakara for guiding and helping me to complete this work.

I also express my profound gratitude to Dr.Seethu Varghese,Head of the Department of Mathematics for her wholehearted cooperation and valuable advices.

Finally,I am obliged to all our teachers of the Mathematics Department, Bharata Mata College for the encouragement for completing this work.

Aardra Rajesh

CONTENTS

INTRODUCTION

Matrix decomposition,also known as matrix factorization,is a technique used in linear algebra to break down a matrix into simpler components.There are several types of matrix decompositions,each with its own set of applications and benefits.A common decomposition method is SINGULAR VALUE DECOMPOSITION(SVD) is discussed in this project.

The computation of SVD involves factorizing a matrix into three matrices,a diagonal matrix and two orthogonal matrix that contain the left and right singular vectors.One of the most significant properties of SVD is that it provides a way to extract the most important information from a matrix.SVD is also useful for computing low-rank approximation of a matrix and to solve linear system of equations for computing pseudoinverse of a matrix.

Many popular algorithms have been implemented in Python,a programming language icluding sorting algorithm,searching algorithm,machine learning algorithm.There are so many libraries available in Python that provide implementations of advanced algorithms of matrix operations,numerical analysis etc.

SVD has numerical application in various fields including data analysis,image compression and machine learning.The application of SVD are vast and varied,making it a powerful tool for analysing and manipulating complex data in many different fields.The main goal of this project is to apply singular value decompositon ,its algorithm to a given problem and use the results to extract useful information or solve a specific task.

Chapter 1

PRELIMINARIES

ORTHOGONAL MATRIX:

It is a square matrix whose columns and rows are orthogonal unit vectors.that is,the dot product of any two different rows or columns is zero,and the dot product of any row or column with itself is one.

DIAGONAL MATRIX:

It is a square matrix whose entries off the main diagonal are zero,and the entries on the main diagonal can be any real or complex numbers.

NORMAL MATRIX:

It is a square matrix that commutes with its conjugate transpose.That is,a matrix A is said to be normal if it satisfies the equation

$$AA^H = A^H A$$

where A^H,the conjugate transpose of A.
The Conjugate Transpose of a matrix is obtained by taking the transpose of the matrix and taking the complex conjugate of each entry.

HERMITIAN MATRIX:

It is a matrix that is equal to its conjugate transpose.That is,

$$A = A^H$$

where A^H is the conjugate transpose of A.

ORTHONORMAL MATRIX:

It is a square matrix where all the columns are orthonormal,meaning that each column has length of 1 and is perpendicular to every other column in the matrix.

EIGENVALUES AND EIGENVECTORS:

Let A be an $n \times n$ matrix.
An eigen vector of A is a nonzero vector v in R^n such that $Av = \lambda v$,for some scalar λ.
An eigen value of A is a scalar λ such that the equation $Av = \lambda v$ has a nontrivial solution.

POSITIVE DEFINITE MATRIX:

A positive definite matrix is a symmetric matrix whose every eigen values is positive.

EIGENVALUE DECOMPOSITION:

Given a square matrix A, the eigenvalue decomposition expresses it as a product of three matrices:

$$A = PDP^{-1}$$

where:

A is the square matrix to be decomposed,
P is a matrix whose columns are the eigenvectors of A,
D is a diagonal matrix whose diagonal entries are the corresponding eigenvalues of A.

RANK OF A MATRIX:

Let A is an $m \times n$ matrix.Then Rank(A) is defined as the dim(Range(A)) .It is the number of linearly independent rows (or columns) of matrix A.
For example,

$$A_1 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ having rank 3.}$$

$$A_2 = \begin{bmatrix} 2 & 1 & -1 \\ 1 & 0 & 1 \\ 1 & 1 & -2 \end{bmatrix} \text{ having rank 2.}$$

Chapter 2

INTRODUCTION TO SINGULAR VALUE DECOMPOSITION

The Singular Value Decomposition(SVD) is a fundamental matrix factorization technique in linear algebra. It was first introduced by Eugenio Beltrami in 1873, but its modern form and importance in numerical linear algebra emerged later.

The term "Singular Value Decomposition " was coined and popularized by Golub and Kahan in their seminar paper " Calculating the Singular Values and Pseudo-Inverse of a Matrix" published in 1965. Their paper provided efficient algorithms for computing the SVD and emphasized its importance in various numerical applications, including least-squares solutions and eigenvalue problems.

Since then, the SVD has become a fundamental concept in numerical linear algebra and found applications in diverse fields, including signal processing, data compression, image processing, statistics, machine learning, and more. Its significance lies in providing a powerful tool for understanding the properties of matrices, data analysis, and solving various mathematical problems in practical and efficient ways.

2.1 SINGULAR VALUE DECOMPOSITION:

Singular value decomposition(SVD) is a matrix factorization technique that decomposes a matrix into three matrices.
Given an $m \times n$ matrix A ,SVD factorises it into three component matrices:

$$A = U\Sigma V^T$$

where U is an $m \times m$ Orthogonal matrix.ie,$UU^T = I$.
Σ is an $m \times n$ diagonal matrix with diagonal elements on the first r rows are singular values and rest of the entries zero.
and V^T is the transpose of an $n \times n$ Orthogonal matrix V.ie,$VV^T = I$

ie,An $m \times n$ matrix A can be decomposed into three components:

$$\text{A}=\begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ u_1 & u_2 & \cdots & u_m \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{m\times m} \begin{bmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \sigma_r & 0 \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}_{m\times n} \begin{bmatrix} \cdots & v_1^T & \cdots \\ \cdots & v_2^T & \cdots \\ & \vdots & \\ \cdots & v_n^T & \cdots \end{bmatrix}_{n\times n}$$

What are those factors U,V and Σ ?

Let A be an $m \times n$ matrix .$\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq ....\lambda_n \geq 0$ be the eigen values of $A^T A$.

Let $\sigma_i = \sqrt{\lambda_i}$, then $\sigma_1 \geq \sigma_2 \geq \sigma_3......\sigma_n \geq 0$.These σ_i's are called Singular values of A.

Consider $AA^T = (U\Sigma V^T)V\Sigma^T U^T = U\Sigma^2 U^T$ and

$$A^T A = (V\Sigma^T U^T)U\Sigma V^T) = V\Sigma^2 V^T$$

The matrix AA^T and $A^T A$ is symmetric positive definite .So it has non-negative real eigen values and orthogonal eigen vectors.

The columns of U are orthogonal unit vectors $u_1, u_2, u_3, ...u_m$ of AA^T and they are left singular vectors of A.

Similarly The columns of V are the orthogonal unit vectors $v_1, v_2, v_3, ...v_m$ of $A^T A$ and they are called right singular vectors of A.

If $m > n$, then the Matrix Σ has diagonal structure upto n rows and consists zeros from n+1 to m.

$$\Sigma = \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_n \\ 0 & \cdots & 0 \\ \vdots & \cdots & 0 \\ 0 & \cdots & 0 \end{bmatrix}$$

If $m < n$,then the Matrix Σ has Diagonal structure upto column m and zeros consists from m+1 to n.

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \cdots & & \\ 0 & 0 & \cdots & \sigma_m & 0 & \cdots & 0 \end{bmatrix}$$

It is important to note that SVD can be performed on any matrix,including rectangular and singular matrices,while eigenvalue decomposition can only be done on square matrices,this property makes SVD more versatile and widely used matrix factorisation technique.

2.1.1 TRUNCATED SVD

It is a variant of the full SVD that approximates a given matrix by keeping only the most significant singular values and their corresponding singular vectors.
In Truncated SVD,we keep only the top k singular values and their corresponding singular vectors, effectively reducing the dimensionality of the original matrix A. The truncated SVD is represented as:

$$A \approx U_k \Sigma_k V_k^T$$

Where:
U_k is an $m \times k$ matrix, containing the first k columns of U.
Σ_k is a $k \times k$ diagonal matrix, containing the top k singular values of A.
V_k^T is the transpose of an n × k matrix, containing the first k columns of V^T.

$$A=\begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ u_1 & u_2 & \cdots & u_k \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{m\times k} \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \sigma_k \end{bmatrix}_{k\times k} \begin{bmatrix} \cdots & v_1^T & \cdots \\ \cdots & v_2^T & \cdots \\ & \vdots & \\ \cdots & v_n^T & \cdots \end{bmatrix}_{n\times k}$$

Truncated SVD is particularly useful when dealing with high-dimensional data and reducing computational complexity while retaining most of the essential information of the original matrix.
It has applications in various fields, including data compression, and feature extraction in machine learning. The choice of the parameter k determines the level of compression and the amount of information preserved.

2.2 GEOMETRICAL INTERPRETATION OF SVD

In the Geometrical interpretation of Singular Value Decomposition (SVD) on the unit sphere,SVD factors can be related to transformations that can be visualised in terms of rotation and stretching.
An m $\times$ n matrix A maps a unit sphere in $\mathbb{R}^n$ to an ellipsoid in $\mathbb{R}^m$.
SVD decomposes a matrix into three fundamental components:
U,Σ and V^T.
Since U and V are orthogonal, applying V and U results in two rotations without distorting the shape

While application of Σ stretches the circle along the coordinate axes to form an ellipse.

For higher dimensions, with rank(A) = r, the unit sphere is transformed to an r-dimensional ellipsoid with semi-axes in the direction of the left singular vectors u_i of magnitude σ_i

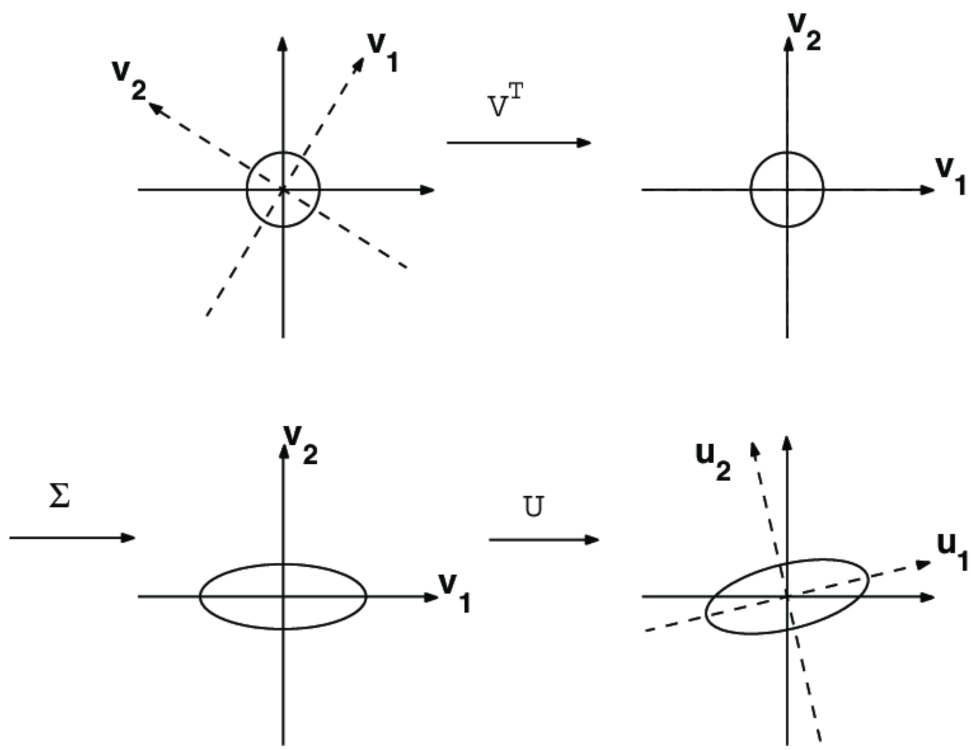
A 2×2 matrix is visualised in the figure.
Let $v = (v_1, v_2)$ be two unit vectors in $\mathbb{R}^2$.
SVD of A $=U\Sigma V^T$

$$\Rightarrow Av = (\text{U } \Sigma V^T)v$$

$$\Rightarrow \text{U } \Sigma V^T(v)$$

Here, V^T rotates and aligns the vector v with the principal axes.Since V^T is an orthogonal matrix,it preserves the length of v but may change its orientation.

$\Rightarrow$ U$\Sigma(V^T v)$
Here,the matrix Σ scales the transformed vector $V^T v$ by the singular values.Since Σ is diagonal,it scales the vector differently along the principal axes.
$\Rightarrow$U ($\Sigma V^T v$)
Here the matrix U further transforms the scalar vector $\Sigma V^T v$ into the output space of $\mathbb{R}^2$.Since U is orthogonal,it preserves the length and orientation of the transformed vector.
So,the SVD of the 2-dimensional matrix A allows us to decompose any input vector v into three successive transformations:
Rotation,Scaling,another Rotation.



Chapter 3

COMPUTING SVD

The computation of Singular Value Decomposition (SVD) involves several steps.

1. Compute $A^T A$ to obtain the $m \times m$ symmetric positive semi-definite matrix $A^T A$.

2. Find the eigenvalues and eigenvectors of $A^T A$
Compute the eigenvalues $(\lambda_1, \lambda_2, ..., \lambda_m)$ and corresponding eigenvectors $(v_1, v_2, ..., v_m)$ of $A^T A$. The eigenvalues will be non-negative real numbers, and the eigenvectors will be m-dimensional unit vectors.

3. Sort and normalize the eigenvectors:
Sort the eigenvalues and corresponding eigenvectors in descending order based on the magnitude of the eigenvalues. Normalize each eigenvector to make them unit vectors.

4. Form matrix V:
The matrix V is an $m \times m$ matrix whose columns are the normalized eigenvectors obtained in step 3. The i-th column of V is the normalized eigenvector corresponding to the i-th largest eigenvalue.

5. Compute the singular values :
The singular values $(\sigma_1, \sigma_2, ..., \sigma_n)$ of the original matrix A can be obtained as the square root of the non-zero eigenvalues $(\lambda_1, \lambda_2, ..., \lambda_n)$,

6. Construct the Σ matrix:
Create the Σ matrix, which is an $m \times n$ diagonal matrix containing non-zero singular values obtained in step 5.

Once you have computed V and Σ, you can find the matrix U using the relation:
$U = AV\Sigma^{-1}$

Now,Let's look at an **example** of computing SVD for a 2×3 matrix.

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

The SVD of the matrix A is of the form $A_{2\times3} = U_{2\times2}\Sigma_{2\times3}V_{3\times3}$
Compute A^TA

$$A^T = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Then $A^TA = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Now to find the eigenvalues and eigenvectors of A^TA by solving the characteristic equation det($A^TA - \lambda I) = 0$,where $\lambda$is the eigen value and I is the identity matrix.

Here we get the Characterisation equation as,

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

Solving the equation we get the eigenvalues λ as 0,1 and 2.

Now we find eigenvectors corresponding to these eigen values.Sort the eigenvalues in descending order.Then the corresponding singular values are the square roots of this eigen values.

For $\lambda_1 = 2$,$\lambda_2 = 1$,$\lambda_3 = 0$

Then the singular values are $\sigma_1 = \sqrt{2}, \sigma_2 = 1, \sigma_3 = 0$

The corresponding eigen vectors are,

for $\lambda_1 = 2, X_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$

for $\lambda_2 = 1, X_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

for $\lambda_3 = 0, X_3 = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}$

Normalise these eigen vectors,

$v_1 = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix}$; where $\sqrt{2} = \sqrt{1^2 + 1^2 + 0^2}$

Similarly, $v_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ and $v_3 = \begin{bmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix}$

$\therefore$ V=$\begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 1 & 0 \end{bmatrix}$

Then $\Sigma = \begin{bmatrix} \sqrt{2} & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$

To find U ,

$$u_1 = \frac{1}{\sigma_1} A v_1 = \frac{1}{\sqrt{2}} \times \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$u_2 = \frac{1}{\sigma_2} A v_2 = \frac{1}{1} \times \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$\therefore$ U=$\begin{bmatrix} u_1 & u_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Now we can easily verify that $U \Sigma V^T = A$

$$U \Sigma V^T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \\ \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{bmatrix}$$

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

3.1 SVD USING PYTHON

In Python,libraries are collection of pre-written code and functions that serve specific purposes.Libraries can easily imported into python projects.
NumPy is the most fundamental and widely used libraries in python for numerical computing.
Here,we utilised the power of NumPY to compute the Singular Value Decomposition of a given matrix.In NumPy,the **'numpy.linalg'** module is used to solve various linear algebra problems.
To compute SVD,we used the built-in **'numpy.linalg.svd()'** function from NumPY,which is specially designed to perform the SVD of a matrix efficiently.The function calculates the left singular vectors,singular values,the transpose of the right singular vectors of the matrix.

lets look at an example.

```
matrix=np.array([[1,2],[3,4]])
U,singularValues,Vt=svd(matrix)
print(" U matrix:")
print(U)
print("Singular Values:")
print(singularValues)
print("V^T matrix:")
print(Vt)
```

lets look at how the function **svd()** defined in Python.

```
import numpy as np
def svd(A):
    # Compute A^T A
    ATA=np.dot(A.T,A)
    # Find eigenvalues and eigenvectors of ATA
    eigenvalues, eigenvectors = np.linalg.eigh(ATA)
    #Sort eigenvalues and corresponding eigenvectors in
descending order
    idx = np.argsort(eigenvalues)[::-1]
    eigenvalues = eigenvalues[idx]
    eigenvectors = eigenvectors[:, idx]
    # Calculate singular values σ_i
    singularValues = np.sqrt(np.maximum(eigenvalues, 0))
    # Compute matrix U
    U = np.dot(A, eigenvectors)
    U /= np.linalg.norm(U, axis=0)
    # Compute matrix V^T
    V = eigenvectors / singularValues
        return U, singularValues, V.T
```

Chapter 4

APPLICATIONS OF SVD

How many pictures you take on your smartphone every day?
Now, imagine trying to share these pictures with friends or family, but you have limited data or storage space.
That's when SVD becomes your trusty sidekick! It takes these high-resolution images, captures their essential features, and creates a compact representation that preserves the most critical details.
SVD helps to save space and data without losing the beauty of your memories.
Now, let's dive into the world of online shopping. Imagine you're an e-commerce giant with millions of users and an endless inventory of products. How do you make sure your customers find what they love?
That's where SVD comes to the rescue! With collaborative filtering, SVD analyzes user interactions, identifies hidden connections between users and items, and recommends products they might adore.

In this chapter, let's explore applications of Singular Value Decomposition (SVD). We will unlock the potential of SVD in image compression and personalized recommendations .

4.1 LOW RANK APPROXIMATION

Let A be an $m \times n$ matrix and $A = U\Sigma V^T$ be its Singular value decomposition.
Σ is an $m \times n$ diagonal matrix with entries $\sigma_1 \geq \sigma_2 \geq ...... \geq \sigma_r \geq 0$ where r=min(m,n).
Let $\Sigma = \Sigma_1 + \Sigma_2 + \Sigma_3 + ... + \Sigma_r$,
where Σ_j is the diagonal matrix with σ_j on the diagonal,but all the other entires are zero.

$$\text{Then } U\Sigma_j V^T = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ u_1 & u_2 & \cdots & u_m \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \Sigma_j \begin{bmatrix} \cdots & v_1^T & \cdots \\ \cdots & v_2^T & \cdots \\ & \vdots & \\ \cdots & v_n^T & \cdots \end{bmatrix}$$

$= \sigma_j u_j v_j^T$,is a rank 1 $m \times n$ matrix.

Then A=$U\Sigma V^T$
=$U(\Sigma_1 + \Sigma_2 + ... + \Sigma_r)V^T$
=$\sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + ...\sigma_r u_r v_r^T$
Thus the Matrix A decomposes into the sum of rank 1 matrices.

Because of the ordering $\sigma_1 \geq \sigma_2 \geq .... \geq \sigma_r$ >0,the rank 1 pieces appear in the descending order of importance.
So $\sigma_1 u_1 v_1^T$ is the best rank 1 approximation to A.
$\sigma_1 u_1 v_1^T + \sigma_2 u_1 v_2^T$ is the best rank 2 approximation etc.
Then $\sum_{j=1}^{k} \sigma_j u_j v_j^T$ will be a good approximation for some value of k that is much smaller than the rank of the matrix A.
This leads to the idea of low rank approximation.

MATRIX NORMS:

Matrix norms are mathematical measures that define the "size" or "magnitude" of a matrix. Two commonly used matrix norms are the 2-norm (also known as the spectral norm) and the Frobenius norm.

2-Norm (Spectral Norm):

The 2-norm of a matrix is the maximum singular value of the matrix. It is defined as follows:

$$||A||_2 = max(\sigma),$$

where σ represents the singular values of the matrix A.

Example: Let's consider a 2×2 matrix A:

$A = \begin{bmatrix} 3 & 1 \\ 1 & 4 \end{bmatrix}$

To find the 2-norm of A, we first need to find its singular values. The singular values of A can be obtained by finding the square root of the eigenvalues of $A^T A$. After calculating the eigenvalues, we find the singular values:

Eigenvalues of $A^T A$:
$\lambda_1 = (7 + \sqrt{5})/2$
$\lambda_2 = (7 - \sqrt{5})/2$

Singular values of A:
$\sigma_1 = \sqrt{\lambda_1} \approx 1.9817$
$\sigma_2 = \sqrt{\lambda_2} \approx 1.7516$

The 2-norm of A is the maximum singular value:

$||A||_2 = max(\sigma_1, \sigma_2) \approx 1.9817$

Frobenius Norm:

The Frobenius norm of a matrix is the square root of the sum of the squares of all its elements. It is defined as follows:

$$||A||_F = \sqrt{(\Sigma(A_{ij})^2)},$$

where A_{ij} are the elements of the matrix A.

Example:
Let's use the same 2×2 matrix A:
$A = \begin{bmatrix} 3 & 1 \\ 1 & 4 \end{bmatrix}$

To find the Frobenius norm of A, we compute:

$$||A||_F = \sqrt{((3^2 + 1^2) + (1^2 + 4^2))} = \sqrt{(10 + 17)} \approx \sqrt{27} \approx 5.1962$$

The Frobenius norm gives a measure of the overall magnitude of the matrix, taking into account all its elements, while the 2-norm focuses on the largest singular value, providing a measure of the maximum scaling effect when the matrix is applied to a vector.

Theorem:Eckart Young

Let $A \in \mathbb{R}^{m\times n}$ .Then for all k $\leq$rank(A),the truncated Singular value decomposition

$$A_k = \sum_{j=1}^{k} \sigma_j u_j v_j^T$$

is the best rank k approximation to A,in the sense that it minimize the difference between the original and truncated matrix.ie,

$$||A - A_k||_F = \min_{rank(A)=k} ||A - X||_F \ ,$$

where $||.||_F$ is the Frobenius norm.

IMAGE COMPRESSION USING SVD

Image Compression provides the most visually appearing application of the low rank approximation of SVD.

A grayscale $m \times n$ pixel image can be represented as a $m \times n$ matrix A where $A_{i,j}$ is the intensity of pixel $p_{i,j}$. In most cases, the intensity lies in the range [0, 255] where 0 is black and 255 is white.
When the original image takes up $O(mn)$ space, the truncated SVD of the image matrix takes up $O(k(m+n+1))$ space. Since k(m+n+1) very less than mn there will be significant savings in storage ,thus giving an effective compression of A.
We have seen that the truncated SVD is the best k-rank approximation for the original matrix,minimising the difference between the original matrix and truncated matrix.

Lets look at an exmaple how effective this image compression can be. The original image is 480 × 640, and so the space needed is reduced for all k ¡ 480.
Lets examine the resulting image $A_k = U_k \Sigma_k V_k^T$ for increasing values of k when we compress the image using SVD.

The following figure shows the original picture with rank 480.

Original (Rank 480)

For k = 288, we can barely notice any difference between the original image and the compressed image.
For k=48 the quality of the image degrades a bit.
Even for k=15,you can still tell that the image shows a city.

Rank 288 approximation

Rank 48 approximation

Rank 15 approximation

Then how's the image for k=1?
So here we are retaining the top 288 singular values and their corresponding singular vectors for compressing the image.

4.2 MOVIE RATING SYSTEM USING SVD

Have you ever wondered how Netflix figures out which movie you want to watch netflix? They have their own recommender systems to do so.There are some fundamental ideas in mathematics that will help us understand how those recommender systems work,One such idea is the SVD.

Consider a simplified movie rating matrix with four users (A, B, C, D) and four movies (X, Y, Z, W). The matrix could look like this:

	X	Y	Z	W
A	5	4	–	2
B	–	3	4	5
C	1	–	2	3
D	4	–	3	–

SVD decomposes the original matrix into three matrices: U, Σ and V^T. The resulting matrices capture the underlying latent features that drive user preferences and movie characteristics.

U: User-feature matrix (left singular vectors): Each row represents a user's affinity for the latent features.
Σ: Diagonal matrix with singular values: These values represent the importance of each latent feature, arranged in descending order.
V^T: Item-feature matrix (right singular vectors): Each row represents a movie's strength in the latent features.
SVD often involves truncating the number of latent features by selecting the top k singular values and their corresponding vectors
Those matrices are given below.

$U =$

	Feature 1	Feature 2
A	0.88	0.48
B	0.23	0.82
C	0.53	0.31
D	0.69	0.04

$$\Sigma = \begin{bmatrix} 9.02 & 0 \\ 0. & 5.81 \end{bmatrix}$$

$$V^T = \begin{array}{c|cc} & \text{Feature 1} & \text{Feature 2} \\ \hline X & 0.89 & 0.48 \\ Y & 0.41 & 0.88 \\ Z & 0.57 & 0.08 \\ W & 0.00 & 0.00 \end{array}$$

To recommend movies to a specific user, we find the movies with high predicted ratings based on the user's preferences and the movie characteristics .
Now, to recommend movies to user C, we calculate the predicted ratings for each movie based on their preferences:

Predicted rating for movie X =$0.53 \times 0.89 + 0.31 \times 0.48 \approx 0.71$
Predicted rating for movie Y =$0.53 \times 0.41 + 0.31 \times 0.88 \approx 0.40$
Predicted rating for movie Z = $0.53 \times 0.57 + 0.31 \times 0.08 \approx 0.34$
Predicted rating for movie W = $0.53 \times 0.00 + 0.31 \times 0.00 \approx 0$
Based on these predicted ratings, we can recommend movie X to user C, as it has the highest predicted rating.

SVD provides a foundation for collaborative filtering methods and has proven to be effective in many recommendation systems, including movie rating systems.

CALCULATING PSEUDOINVERSE

Let A be an $m \times n$ matrix. Then an $n \times m$ matrix X is the pseudoinverse of A if X satisfies the following properties:
(i) $AXA =$ A
(ii)$XAX =$ X
(iii)$(AX)^T =$ AX
(iv) $(XA)^T =$ XA
The pseudoinverse of A is denoted by A^+
Now, when A $= U\Sigma V^T$ then,
$A^+ = (U\Sigma V^T)^+$
$\Rightarrow(\mathrm{V}^T)^{-1}\Sigma^+U^{-1}$
$\Rightarrow (V^{-1})^{-1}\Sigma^+U^T$
$\Rightarrow V\Sigma^+U$
ie,$\mathbf{A}^+ = V\Sigma^+U$
where $\Sigma^+ = diag(\sigma_i^+)$ and

$$\sigma_i^+ = \begin{cases} 1/\sigma_i \ , & \text{if}\sigma_i > 0 \\ & \text{if}\sigma_i = 0 \end{cases}$$

Example

Let A$=\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
In chapter 2,we have seen the SVD of A is,

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \\ \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{bmatrix}$$

The Pseudo inverse A^+ is,

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 \\ 0 & 1 \end{bmatrix}$$

In this chapter,we have seen three applications of SVD.
SVD plays a crucial role in data analysis,dimensionality reduction,collabarative filtering.The practical relevance of SVD in solving real life problems make Singular value decomposition significant.

CONCLUSION

Singular value decomposition (SVD) is a method of representing a matrix as a series of linear approximations that expose the underlying meaning-structure of the matrix.

The singular value decomposition hold for any matrices ,while eigen value only holds for square matrices that are diagonalizable.This makes Singular value decomposition a better tool.

Singular value decomposition is effective in devoloping low rank approximation to A than Eigen value Decomposition .This is because,the singular values are non-negative real numbers whose ordering

$$\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq .....\sigma_r > 0$$

gives a natural way to understand how much the rank 1 matrices $\sigma_j u_j v_j^T$ contribute to A.

Mathematical application of SVD involves computing the Pseudoinverse,matrix approximation,and determining rank,nullspace,range of a matrix.

Singular value Decomposition is also useful in the field of science,engineering, ans statistics such as signal processing,least squares of fitting of data and process control.

REFERENCE

1.Mathematics for Machine Learning
Marc Peter Deisenorth,A.Aldo Faisal,Cheng Soon Ong
2.Singular Value Decomposition and its Applications
Zecheng Kuang ,Undergraduate Honor Thesis.
3.JW.Demmel,Applied Numerical Linear Algebra,Soc.for industrial and applied Math.
4.Computation of Singular Value Decomposition
Alan Kaylor Clinie,Inderjit S.Dhillion.