Semester 2 Group Project

May 3

Outline

Background Information

Matrices are a set of numbers arranged in rows and columns that form a rectangular array. A 2D matrix is defined as $\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$. They can be transformed in many ways such as:

- shear $\begin{pmatrix} 1 & 0 \\ x & 1 \end{pmatrix}$ or $\begin{pmatrix} 1 & y \\ 0 & 1 \end{pmatrix}$,
- rotation $\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$,
- enlargement by scale factor a $\begin{pmatrix} ax & 0 \\ 0 & ay \end{pmatrix}$,
- reflection $\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$

Initial Task

Mathematical Concept

In the initial task,we investigate the relationship between 2×2 matrices and transformations of R^2

Here is the mathematical mechanism for the matrix transformation.

$$\begin{pmatrix} x_1' & x_2' & x_3' & x_4' \\ y_1' & y_2' & y_3' & y_4' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \end{pmatrix} \quad (1)$$

$$c_{ij} = \sum_k a_{ik} b_{kj} \quad (2)$$

Initial Task

Worked Examples

```
Here enters the point with a separate-commma list for the quadrilateral( x1,y1,x2,y2,x3,y3,x4,y4): 0,0,2,0,2,2,0,2
Here enters the point with a separate-commma list for the quadrilateral transformed matrix( x1,y1,x2,y2): 2,0,0,2
```

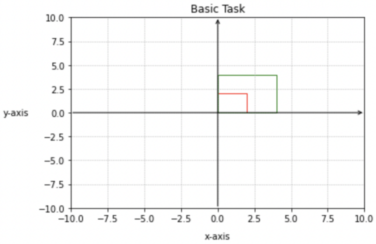


Figure: Scale

Initial Task

Worked Examples

```
Here enters the point with a separate-commma list for the quadrilateral( x1,y1,x2,y2,x3,y3,x4,y4): -2,-2,2,-2,2,2,-2,
2
Here enters the point with a separate-commma list for the quadrilateral transformed matrix( x1,y1,x2,y2): 1,2,0,1
```

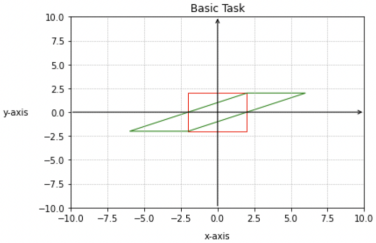


Figure: Shear

Initial Task

Worked Examples

```
Here enters the point with a separate-commma list for the quadrilateral( x1,y1,x2,y2,x3,y3,x4,y4): 0,0,2,0,2,2,0,2
Here enters the point with a separate-commma list for the quadrilateral transformed matrix( x1,y1,x2,y2): 1,0,0,-1
```

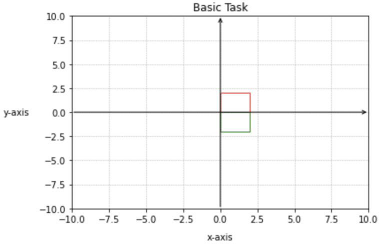


Figure: Reflection

Initial Task

Transformation Failure

Here are some reasons why the transformation is failure.

(1) $\det A = 0$
When $\det(A) = 0$, the matrix is non-invertible, which means there is no inverse matrix and the transformation is not one-to-one. The image will be stretched to a line or compressed to a point during the transformation.

(2)The input points do not satisfy the property of quadrilateral,which leads to some overlapping or missing regions in the transformed quadrilateral.Besides thisIf the four input points are not adjacent, they do not form a quadrilateral, and therefore the transformation cannot be performed.

Initial Task

Worked Examples

```
Here enters the point with a separate-commma list for the quadrilateral( x1,y1,x2,y2,x3,y3,x4,y4): -2,-2,2,-2,2,2,-2,
2
Here enters the point with a separate-commma list for the quadrilateral transformed matrix( x1,y1,x2,y2): 3,3,3,3
```

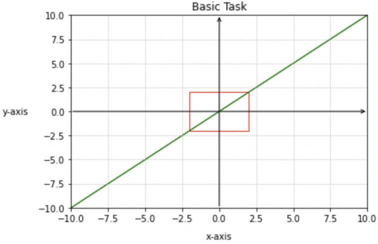


Figure: Failure 1

Initial Task

Worked Examples

```
Here enters the point with a separate-commma list for the quadrilateral( x1,y1,x2,y2,x3,y3,x4,y4): -2,-4,2,4,-2,4,2,-
4
Here enters the point with a separate-commma list for the quadrilateral transformed matrix( x1,y1,x2,y2): 0,1,1,0
```

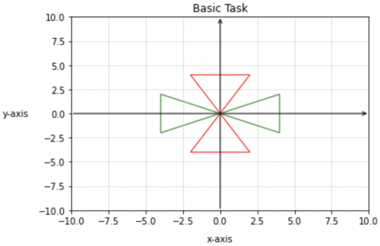


Figure: Failure 2

Composition and Matrix Multiplication

Commutativity of Matrix Multiplication

When multiplication is commutative, it means that the order of the elements being multiplied won't affect the product i.e., $AB = BA$. Matrix Multiplication is not commutative which means that $AB \neq BA$. You can very quickly prove this by finding examples using this code:

```
import numpy as np
def com_check(): #commutativity check function
    A = input("Please enter values with a separate-commma list for
matrix A ( x1,y1,x2,y2): ")
    A = np.array([float(x) for x in A.split(",")]).reshape(2, 2)
    B = input("Please enter valuesin the same way for the matrix B:")
    B = np.array([float(x) for x in B.split(",")]).reshape(2, 2)
    C = A@B
    D = B@A
    if np.mean(C==D):#compares matrices
        print(C)
        print("These matrices happen to be commutative, try again")
        com_check()#repeats till one is found
    else:
        print(C)
        print(D)
        print("These are not equal and therefore non-commutative for

com_check()
```

Composition and Matrix Multiplication

When is Matrix Multiplication Commutative?

Although most matrices are not commutative, there is a specific set if matrices that are. Matrices are commutative when they simultaneously diagonalizable. This means they are of the form: $\begin{pmatrix} x & y \\ y & x \end{pmatrix}$. This is easy to prove algebraically if you set $X = \begin{pmatrix} a & b \\ b & a \end{pmatrix}$ and $Y = \begin{pmatrix} c & d \\ d & c \end{pmatrix}$

$XY = \begin{pmatrix} (ac + bd) & (ad + bc) \\ (bc + ad) & (bd + ac) \end{pmatrix}$. and $YX = \begin{pmatrix} (ca + db) & (cb + da) \\ (da + cb) & (db + ca) \end{pmatrix}$.

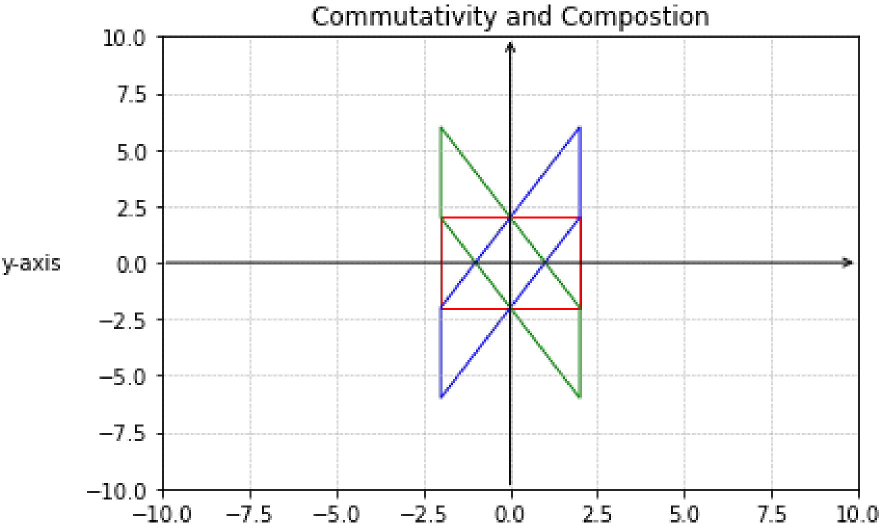
Composition and Matrix Multiplication

Composition of Transformations

Matrix multiplication being non-commutative is significant for transposition of transformations as it means that the order in which you do each transformation is important for the final outcome. For example, If we star with the quadrilateral $\begin{pmatrix} -2 & 2 & 2 & -2 \\ -2 & -2 & 2 & 2 \end{pmatrix}$ and apply a shear parallel to the x-axis of factor 2, $S = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}$, and then reflect it in the x-axis, $R = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$, it should produce a different quadrilateral when the transformations are applied in the opposite order. This is seen in the following figure where green represents the shear then reflection, $R \circ S$, blue represents reflection then shear, $S \circ R$ and red represents the original quadrilateral:

Composition and Matrix Multiplication

Composition of Transformations



Complex Numbers

Matrix Form

To write complex numbers in matrix form we first look at the rotation matrix $e^{i\theta} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$ which can be simplified to $\cos\theta = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and $\sin\theta = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ and $\cos\theta + i\sin\theta$ so

$$1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad i = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}.$$

Therefore a complex number of the form $a + bi$ can be written in matrix form

$$a + bi = \begin{pmatrix} a & -b \\ b & a \end{pmatrix} \quad \text{which is standardised to} \quad z = \begin{pmatrix} Re(z) & -Im(z) \\ Im(z) & Re(z) \end{pmatrix}.$$

Complex Numbers

Using the following code some of the basic properties of matrices can be produced in python:

```
det = np.linalg.det(matrix)
transpose = np.transpose(matrix)
inverse = np.linalg.inv(matrix)
```

and when z is set to a complex number in the form $a + bi$ the following code is used to calculate some properties:

```
z_conj = np.conj(z)
mod = (a**2 + b**2)**0.5
z_rep = 1/z
```

Complex Numbers

Worked Example

When $z = 6 + 8i$ is used the corresponding matrix is $\begin{pmatrix} 6 & -8 \\ 8 & 6 \end{pmatrix}$.

$$det(A) = 100 \quad transpose = \begin{pmatrix} 6 & 8 \\ -8 & 6 \end{pmatrix} \quad inverse = \begin{pmatrix} 0.06 & -0.08 \\ 0.08 & 0.06 \end{pmatrix}.$$

$$|z| = 10 \quad z^* = 6 - 8i \quad \frac{1}{z} = 0.06 + 0.08i$$

From these values we can see that:

- $det(A) = |z|^2$.
- $A^T = z^*$.
- $A^{-1} = \frac{1}{z}$.

Complex Numbers can also undergo transformations.

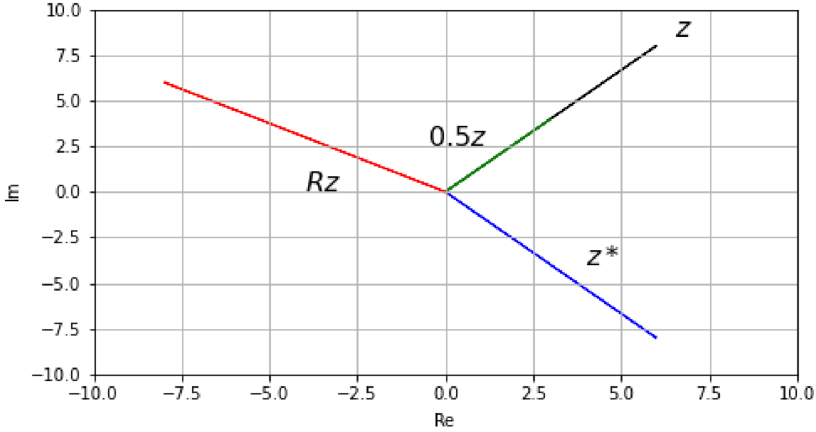


Figure: Complex Transformations

Reference

1. https://www.britannica.com/science /matrix-mathematics
2. *Linear Algebra step by step* **Kuldeep Singh** (*OUP*)
3. https://github.com/xinychen/awesome-beamer
4. https://www.overleaf.com/learn/latex/Matrices
5. https://math.libretexts.org/Bookshelves/ Differential_Equations/Applied_Linear_Algebra_and_Differential_Equations_(Chasnov)/02%3A_II._Linear_Algebra/01%3A_Matrices/1.06%3A_Matrix_Representation_of_Complex_Numbers

Thanks for your participation !