Lab Assignment: Matrix Operations using SymPy in Python

1 Introduction

SymPy is a Python library for symbolic mathematics. It allows for exact symbolic manipulations of matrices, making it an excellent tool for algebraic computations. This lab will focus on performing basic and advanced matrix operations using SymPy.

2 Instructions

1. Read each problem carefully and implement the required code in Python using SymPy. 2. Compare results with NumPy where necessary. 3. Submit your Python scripts along with outputs.

3 Exercises

3.1 Basic Matrix Operations

1. **Matrix Creation:** Create the following matrices using SymPy:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 0 \\ 1 & 3 \end{bmatrix} \tag{1}$$

   Print both matrices using the SymPy pprint function.

2. **Addition and Subtraction:** Compute and print $A + B$ and $A - B$.

3. **Matrix Multiplication:** Compute and print $A \times B$.

4. **Scalar Multiplication:** Compute and print $3A$.

5. **Transpose:** Compute and print the transpose of A.

6. **Determinant:** Compute and print the determinant of A.

7. **Inverse:** Compute and print the inverse of A.

3.2 Advanced Matrix Operations

1. **Symbolic Matrix Operations:** Define a symbolic matrix

$$M = \begin{bmatrix} x & 1 \\ 2 & y \end{bmatrix} \tag{2}$$

   Compute and print its determinant.

2. **Solving Matrix Equations:** Solve the system of equations represented by

$$\begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 5 \\ 6 \end{bmatrix} \tag{3}$$

   for x and y.

3. **Matrix Equation:** Solve for x and y given the equation

$$\begin{bmatrix} x+y & 2 \\ 3 & x-y \end{bmatrix} = \begin{bmatrix} 4 & 2 \\ 3 & 1 \end{bmatrix} \tag{4}$$

4. **Matrix Exponentiation using Loops:** Compute the power A^n for a given integer n using a loop instead of the built-in power function.

5. **Generating Fibonacci Sequence using Matrix Multiplication:** Implement a loop-based approach to compute the n-th Fibonacci number using matrix exponentiation.

3.3 Comparison with NumPy

1. Compare the determinant of matrix A computed using SymPy and NumPy. Explain any differences.

2. Compute the inverse of matrix A using both SymPy and NumPy and explain the output format differences.

3. Demonstrate why SymPy is more useful than NumPy for handling symbolic matrices by computing the inverse of

$$M = \begin{bmatrix} a^2 & b^3 \\ c^2 & d^4 \end{bmatrix} \tag{5}$$

   symbolically.

Lab Assignment: Matrix Operations using SymPy in Python

1 Introduction

SymPy is a Python library for symbolic mathematics. It allows for exact symbolic manipulations of matrices, making it an excellent tool for algebraic computations. This lab will focus on performing basic and advanced matrix operations using SymPy.

2 Instructions

1. Read each problem carefully and implement the required code in Python using SymPy. 2. Compare results with NumPy where necessary. 3. Submit your Python scripts along with outputs.

3 Exercises

3.1 Basic Matrix Operations

1. **Matrix Creation:** Create the following matrices using SymPy:

$$A = \begin{bmatrix} 2 & 5 \\ 8 & 10 \end{bmatrix}, \quad B = \begin{bmatrix} -2 & 5 \\ -1 & 8 \end{bmatrix}$$

   Print both matrices using the SymPy pprint function.

2. **Addition and Subtraction:** Compute and print $A + B$ and $A - B$.

3. **Matrix Multiplication:** Compute and print $A \times B$.

4. **Scalar Multiplication:** Compute and print $3A$.

5. **Transpose:** Compute and print the transpose of A.

6. **Determinant:** Compute and print the determinant of A.

7. **Inverse:** Compute and print the inverse of A.

3.2 Advanced Matrix Operations

1. **Matrix Transposition Property:** Given the matrices:

$$A = \begin{bmatrix} x & y \\ 1 & x+y \end{bmatrix}, \quad B = \begin{bmatrix} 2x+1 & y+1 \\ 2x & x+2y \end{bmatrix}$$

   Show that $(AB)^T = B^T A^T$ by computing both sides explicitly.

2. **Solving Matrix Equations:** Solve the system of equations represented by

$$\begin{bmatrix} 4 & 6 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 26 \\ 8 \end{bmatrix}$$

   for x and y.

3. **Matrix Equation:** Solve for x and y given the equation

$$\begin{bmatrix} 2x-y & 2 \\ 3 & x-y^2 \end{bmatrix} = \begin{bmatrix} 4 & 2 \\ 3 & 1 \end{bmatrix}$$

4. **Zeros of Determinant:** Find the values of x such that the determinant of matrix M is zero.

$$M = \begin{bmatrix} x^2+x-13 & 1 \\ x-12 & x^2 \end{bmatrix}$$

5. **Matrix Exponentiation using Loops:** Compute the power A^n for a given **integer** n using a loop instead of the built-in power function.

6. **Generating Fibonacci Sequence using Matrix Multiplication:** Compute the sum of the n-th and $(n-1)$-th Fibonacci numbers using matrix multiplication.