MatGeo Assignment 4.13.76

AI25BTECH11007

September 28, 2025

Question

Use elementary column operation $C_2 \to C_2 + 2C_1$ in the following matrix equation

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

Solution

We have

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

The column operation $C_2 \to C_2 + 2C_1$ is represented by the elementary matrix

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

since post-multiplication by E performs the same column operation. Thus, by matrix theory,

$$AE = MNE,$$

where

$$AE = \begin{pmatrix} 2 & 5 \\ 2 & 4 \end{pmatrix}, \qquad NE = \begin{pmatrix} 1 & 2 \\ -1 & -1 \end{pmatrix}.$$

Hence,

$$\boxed{\begin{pmatrix} 2 & 5 \\ 2 & 4 \end{pmatrix} = \begin{pmatrix} 3 & 1 \\ 2 & 0 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ -1 & -1 \end{pmatrix}}.$$

C code

```c
        #include <stdio.h>

#define N 2

// Function to multiply two matrices
void multiply(int A[N][N], int B[N][N], int result[N][N]) {
    for (int i = 0; i < N; i++) {
        for (int j = 0; j < N; j++) {
            result[i][j] = 0;
            for (int k = 0; k < N; k++) {
                result[i][j] += A[i][k] * B[k][j];
            }
        }
    }
}
```

C code

```c
        // Function to print matrix
void printMatrix(int A[N][N]) {
    for (int i = 0; i < N; i++) {
        for (int j = 0; j < N; j++) {
            printf("%d ", A[i][j]);
        }
        printf("\n");
    }
    printf("\n");
}

int main() {
    int A[N][N] = {{2, 1}, {2, 0}};
    int M[N][N] = {{3, 1}, {2, 0}};
    int Nmat[N][N] = {{1, 0}, {-1, 1}};
    int E[N][N] = {{1, 2}, {0, 1}}; // elementary matrix

    int AE[N][N], NE[N][N], MNE[N][N];
```

C code

```
    // Compute AE and NE
    multiply(A, E, AE);
    multiply(Nmat, E, NE);

    // Compute M * (NE)
    multiply(M, NE, MNE);

    printf("AE = \n");
    printMatrix(AE);

    printf("NE = \n");
    printMatrix(NE);

    printf("M * NE = \n");
    printMatrix(MNE);

    return 0;
}
```

Python code

```
        import numpy as np

# Define matrices
A = np.array([[2, 1], [2, 0]])
M = np.array([[3, 1], [2, 0]])
N = np.array([[1, 0], [-1, 1]])

# Elementary matrix for C2 -> C2 + 2*C1
E = np.array([[1, 2], [0, 1]])

# Compute AE, NE, and M * NE
AE = A @ E
NE = N @ E
MNE = M @ NE

print("AE =\n", AE)
print("\nNE =\n", NE)
print("\nM * NE =\n", MNE)
```