5.6.7

Abhiram Reddy-AI25BTECH11021

September 28,2025

Question

Let

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

Verify that

$$A^3 - 6A^2 + 9A - 4I = 0, \tag{1}$$

by using the Cayley–Hamilton theorem, and hence find A^{-1}.

Step 1: Characteristic Polynomial

$$\chi_A(\lambda) = \det(\lambda I - A)$$
$$= \det \begin{pmatrix} \lambda - 2 & 1 & -1 \\ 1 & \lambda - 2 & 1 \\ -1 & 1 & \lambda - 2 \end{pmatrix}. \tag{2}$$

Expanding gives

$$\chi_A(\lambda) = (\lambda - 4)(\lambda - 1)^2 = \lambda^3 - 6\lambda^2 + 9\lambda - 4. \tag{3}$$

Step 2: Cayley–Hamilton Theorem

By Cayley–Hamilton, A satisfies its characteristic equation:

$$A^3 - 6A^2 + 9A - 4I = 0, \tag{4}$$

which proves the required identity.

Step 3: Formula for A^{-1}

Multiplying (**??**) on the right by A^{-1}:

$$A^2 - 6A + 9I - 4A^{-1} = 0. \tag{5}$$

Thus,

$$A^{-1} = \frac{1}{4}\left(A^2 - 6A + 9I\right). \tag{6}$$

Step 4: Determinant and Adjugate

From (**??**), eigenvalues are $4, 1, 1$. Hence

$$\det(A) = 4. \tag{7}$$

The adjugate matrix:

$$\operatorname{adj}(A) = \det(A)\, A^{-1} = 4A^{-1}.$$

So,

$$A^{-1} = \frac{1}{4}\begin{pmatrix} 3 & 1 & -1 \\ 1 & 3 & 1 \\ -1 & 1 & 3 \end{pmatrix}. \tag{8}$$

C Code (Part 1)

```c
#include <stdio.h>

int main() {
    int i, j, k;
    int A[3][3] = {
        {2, -1, 1},
        {-1, 2, -1},
        {1, -1, 2}
    };

    int A2[3][3] = {0};
    int temp[3][3] = {0};
    float Ainv[3][3];

    for(i = 0; i < 3; i++) {
        for(j = 0; j < 3; j++) {
            for(k = 0; k < 3; k++) {
                A2[i][j] += A[i][k] * A[k][j];
            }
```

```
    for(i = 0; i < 3; i++) {
        for(j = 0; j < 3; j++) {
            temp[i][j] = A2[i][j] - 6*A[i][j];
            if(i == j) temp[i][j] += 9;
        }
    }

    for(i = 0; i < 3; i++) {
        for(j = 0; j < 3; j++) {
            Ainv[i][j] = temp[i][j] / 4.0;
        }
    }

    printf(The inverse matrix A^{-1} is:\n);
    for(i = 0; i < 3; i++) {
        for(j = 0; j < 3; j++) {
            printf(%6.2f , Ainv[i][j]);
        }
        printf(\n);
```

Python Code (Part 1)

```
import numpy as np

A = np.array([
    [2, -1, 1],
    [-1, 2, -1],
    [1, -1, 2]
], dtype=float)

# Compute A^2
A2 = np.dot(A, A)

# Compute A^2 - 6A + 9I
temp = A2 - 6*A + 9*np.eye(3)

# Divide by determinant (4) to get inverse
Ainv = temp / 4.0

print(The inverse matrix A^{-1} is:)
print(Ainv)
```