Non-Homogeneous Equation

A **non-homogeneous equation** is a linear equation in which the sum of the product of variables and their coefficients is equal to a non-zero constant. In general, it can be represented as:

[$a_1x_1 + a_2x_2 + \cdots + a_nx_n = D$]

where ($D \neq 0$). Here, ($a_1, a_2, \ldots, a_n$) are the coefficients, and (D) is a non-zero constant.

Matrix Representation of Non-Homogeneous Equation

The system of non-homogeneous linear equations can be represented in matrix form as:

[$A\mathbf{x} = \mathbf{D}$]

where:

- (A) is a matrix of coefficients,
- ($\mathbf{x}$) is a column vector of the variables,
- ($\mathbf{D}$) is a column vector of constants.

For example, consider a system of equations:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 = D_1 \\ a_{21}x_1 + a_{22}x_2 = D_2 \end{cases}$$

The matrix form is:

[$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \end{bmatrix}$$]

Augmented Matrix

The **augmented matrix** is a way to represent a system of linear equations by combining the coefficient matrix and the constant vector into a single matrix. It is constructed by appending the vector ($\mathbf{D}$) to the coefficient matrix (A).

For the example given above, the augmented matrix is:

[$$[A \mid \mathbf{D}] = \begin{bmatrix} a_{11} & a_{12} & \mid & D_1 \\ a_{21} & a_{22} & \mid & D_2 \end{bmatrix}$$]

This form is often used to solve systems of equations using methods such as Gaussian elimination.

Homogeneous Equation

A **homogeneous equation** is a linear equation in which the sum of the product of variables and their coefficients is equal to zero. In general, it can be represented as:

[$a_1x_1 + a_2x_2 + \cdots + a_nx_n = 0$]

Here, the constant term is zero, meaning that the system always has at least one solution, known as the **trivial solution**, where all variables are equal to zero (($x_1 = x_2 = \cdots = x_n = 0$)).

Matrix Form of Homogeneous Equation

The system of homogeneous linear equations can also be represented in matrix form as:

[$A\mathbf{x} = \mathbf{0}$]

where:

- (A) is a matrix of coefficients,
- ($\mathbf{x}$) is a column vector of variables,
- ($\mathbf{0}$) is a column vector of zeros.

For the same system of equations without the constant vector:

$$[\begin{cases} a_{11}x_1 + a_{12}x_2 = 0 \\ a_{21}x_1 + a_{22}x_2 = 0 \end{cases}]$$

The matrix form is:

$$[\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}]$$

Submatrix

A **submatrix** is a matrix formed by deleting one or more rows and/or columns from a given matrix. In other words, a submatrix is derived by selecting certain rows and columns of a larger matrix.

Example

Consider a matrix (A):

$$[A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}]$$

- **Example 1**: Deleting the 1st row and 1st column results in the submatrix:

$$[\begin{bmatrix} 5 & 6 \\ 8 & 9 \end{bmatrix}]$$

- **Example 2**: Deleting the 2nd row results in the submatrix:

[

$$\begin{bmatrix} 1 & 2 & 3 \\ 7 & 8 & 9 \end{bmatrix}$$

]$

- **Example 3**: Deleting the 3rd column results in the submatrix:

$$[\begin{bmatrix} 1 & 2 \\ 4 & 5 \\ 7 & 8 \end{bmatrix}]$$

Rank of a Matrix

The **rank** of a matrix is the maximum number of linearly independent rows or columns in the matrix. It represents the dimension of the vector space spanned by its rows or columns.

In [8]:
```python
import numpy as np
```

In [12]:
```python
arr=np.array([[1,2,3],[4,5,6],[7,8,9]])
arr
```

Out[12]:
```
array([[1, 2, 3],
       [4, 5, 6],
       [7, 8, 9]])
```

In [14]:
```python
np.linalg.matrix_rank(arr)
```

Out[14]:
```
2
```

Examples

1. **Rank 1**: The rank of a matrix is 1 if every (2×2) submatrix is singular (i.e., the determinant of each (2×2) submatrix is zero).

In [11]:
```python
import numpy as np

# Example matrix of rank 1
A = np.array([[1, 2, 3],
              [2, 4, 6],
              [3, 6, 9]])
```

```python
# Check if all 2x2 submatrices are singular
def is_rank_1(matrix):
    n = matrix.shape[0]
    for i in range(n - 1):
        for j in range(n - 1):
            submatrix = matrix[i:i+2, j:j+2]
            if np.linalg.det(submatrix) != 0:
                return False
    return True

print("Rank 1:", is_rank_1(A))  # Output: True
```

```
Rank 1: False
```

Rank 2: The rank of a matrix is 2 if the matrix itself is singular, but at least one of its 2×2 submatrices is non-singular.

In [29]:

```python
# Example matrix of rank 2
B = np.array([[1, 2, 3],
              [4, 5, 6],
              [7, 8, 9]])

# Check if matrix is singular and at least one 2x2 submatrix is non-singular
def is_rank_2(matrix):
    if np.linalg.det(matrix) != 0:
        return False
    n = matrix.shape[0]
    for i in range(n - 1):
        for j in range(n - 1):
            submatrix = matrix[i:i+2, j:j+2]
            if np.linalg.det(submatrix) != 0:
                return True
    return False

print("Rank 2:", is_rank_2(B))  # Output: True
```

```
Rank 2: False
```

Rank 3: The rank of a 3×3 matrix is 3 if the matrix is non-singular (i.e., its determinant is non-zero).

In [32]:

```python
# Example matrix of rank 3
C = np.array([[1, 2, 1],
              [0, 1, 2],
              [3, 4, 5]])

# Check if matrix is non-singular
def is_rank_3(matrix):
    return np.linalg.det(matrix) != 0

print("Rank 3:", is_rank_3(C))  # Output: True
```

```
Rank 3: True
```

In [ ]:

Processing math: 100%