INNER Product and OUTER Product

By: Vivek Vittal Biragoni 211AI041

09/05/23

(i) vector and vector

```
# Define test vectors of different sizes
v1 = [1, 2, 3]
v2 = [4, 5, 6]
v3 = [1, 2, 3, 4]
v4 = [4, 5, 6]

# Compute the inner product of the two vectors
if len(v1) == len(v2):
    inner_product = sum([v1[i] * v2[i] for i in range(len(v1))])
    print("Inner product:", inner_product)
else:
    print("Error: Vectors must be of the same length for inner product calculation")

# Compute the outer product of the two vectors
outer_product = [[0]*len(v2) for i in range(len(v1))]
if len(v1) == len(v2):
    for i in range(len(v1)):
        for j in range(len(v2)):
            outer_product[i][j] = v1[i] * v2[j]
    print("Outer product:")
    for row in outer_product:
        print(row)
else:
    print("Error: Vectors must be of the same length for outer product calculation")

# Compute the inner and outer product of v3 and v4
if len(v3) == len(v4):
    inner_product = sum([v3[i] * v4[i] for i in range(len(v3))])
    print("Inner product:", inner_product)
else:
    print("Error: Vectors must be of the same length for inner product calculation")

outer_product = [[0]*len(v4) for i in range(len(v3))]
if len(v3) == len(v4):
    for i in range(len(v3)):
        for j in range(len(v4)):
            outer_product[i][j] = v3[i] * v4[j]
    print("Outer product:")
    for row in outer_product:
        print(row)
else:
    print("Error: Vectors must be of the same length for outer product calculation")
```

```
Inner product: 32
Outer product:
[4, 5, 6]
[8, 10, 12]
[12, 15, 18]
Error: Vectors must be of the same length for inner product calculation
Error: Vectors must be of the same length for outer product calculation
```

In this code, we define four different vectors of different sizes for testing. The vectors v1 and v2 have the same size, so both the inner and outer products are computed and printed without errors. The vectors v3 and v4 have different sizes, so the computation of both the inner and outer products results in errors

To compute the inner product, the program uses a loop to multiply corresponding elements of the two vectors and sum the products. If the vectors are of different lengths, the program raises an error message.

To compute the outer product, the program first creates a two-dimensional array of zeros with the same dimensions as the resulting outer product. It then uses nested loops to multiply each element of one vector by each element of the other vector, storing the results in the corresponding positions of the two-dimensional array. If the vectors are of different lengths, the program raises an error message.

The program includes additional test cases to demonstrate how it handles vectors of different lengths and produces error messages when appropriate.

(ii) vector and matrix

```
def inner_product(vector, matrix):
    if len(vector) == len(matrix[0]):
        inner_product = [sum([vector[i] * matrix[j][i] for i in range(len(vector))]) for j in range(len(matrix))]
        return inner_product
    else:
        return "Error: The length of the vector must be the same as the number of columns in the matrix for inner product calculation"

def outer_product(vector, matrix):
    if len(vector) == len(matrix[0]):
        outer_product = [[vector[i] * matrix[j][i] for j in range(len(matrix))] for i in range(len(vector))]
        return outer_product
    else:
        return "Error: The length of the vector must be the same as the number of columns in the matrix for outer product calculation"

# Test cases
v1 = [1, 2, 3]
v2 = [4, 5, 6, 7]
M1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
M2 = [[1, 2], [3, 4], [5, 6]]
M3 = [[1, 2, 3], [4, 5, 6], [7, 8, 9], [10, 11, 12]]

# Compute the inner product of the vector and each row of the matrix
print(inner_product(v1, M1))
print(inner_product(v2, M1))

# Compute the outer product of the vector and the matrix
print(outer_product(v1, M1))
print(outer_product(v2, M1))
print(outer_product(v1, M2))
print(outer_product(v1, M3))
```

```
[14, 32, 50]
Error: The length of the vector must be the same as the number of columns in the matrix for inner product calculation
[[1, 4, 7], [4, 10, 16], [9, 18, 27]]
Error: The length of the vector must be the same as the number of columns in the matrix for outer product calculation
Error: The length of the vector must be the same as the number of columns in the matrix for outer product calculation
[[1, 4, 7, 10], [4, 10, 16, 22], [9, 18, 27, 36]]
```

This implementation defines two functions, `inner_product()` and `outer_product()`, that calculate the inner and outer product, respectively, between a vector and a matrix. The code includes checks to ensure that the length of the vector matches the number of columns in the matrix to properly perform the calculations. Additionally, the implementation includes test cases for both successful and failure cases. The inner product function calculates the dot product between a vector and each row of the matrix and returns the resulting vector. The outer product function calculates the outer product between a vector and a matrix and returns the resulting matrix. Overall, this implementation provides a straightforward and efficient way to compute the inner and outer product of a vector and matrix.

(iii) matrix and matrix

```
def inner_product(matrix1, matrix2):
    if len(matrix1[0]) == len(matrix2):
        inner_product = [[sum([matrix1[i][k] * matrix2[k][j] for k in range(len(matrix1[0]))]) for j in range(len(matrix2[0]))] for i in rang
        return inner_product
    else:
        return "Error: The number of columns in matrix 1 must be the same as the number of rows in matrix 2 for inner product calculation"

def outer_product(matrix1, matrix2):
    if len(matrix1[0]) == len(matrix2[0]):
        outer_product = [[matrix1[i][j] * matrix2[k][l] for l in range(len(matrix2[0]))] for k in range(len(matrix2)) for i in range(len(matr
        outer_product = [[outer_product[i + j][j] for j in range(len(matrix2[0]))] for i in range(0, len(matrix1) * len(matrix2), len(matrix2
            return outer_product
        else:
            return "Error: The number of columns in matrix 1 must be the same as the number of columns in matrix 2 for outer product calculation"

    # Test cases
    M1 = [[1, 2], [3, 4], [5, 6]]
    M2 = [[7, 8, 9], [10, 11, 12]]

    # Compute the inner product of the two matrices
    print(inner_product(M1, M2))

    # Compute the outer product of the two matrices
    print(outer_product(M1, M2))


    # Test case 1 - valid input for inner product
    M1 = [[1, 2], [3, 4]]
    M2 = [[5, 6], [7, 8]]
    # Compute the inner product of the two matrices
    print(inner_product(M1, M2))

    # Compute the outer product of the two matrices
    print(outer_product(M1, M2))

    # Test case 2 - valid input for outer product
    M1 = [[1, 2], [3, 4]]
    M2 = [[5, 6], [7, 8]]
    # Compute the inner product of the two matrices
    print(inner_product(M1, M2))

    # Compute the outer product of the two matrices
    print(outer_product(M1, M2))

    # Test case 3 - invalid input for inner product (unequal number of columns and rows)
    M1 = [[1, 2, 3], [4, 5, 6]]
    M2 = [[7, 8, 9], [10, 11, 12]]
    # Compute the inner product of the two matrices
    print(inner_product(M1, M2))

    # Compute the outer product of the two matrices
    print(outer_product(M1, M2))

    # Test case 4 - invalid input for outer product (unequal number of columns)
    M1 = [[1, 2], [3, 4]]
    M2 = [[5, 6, 7], [8, 9, 10]]
    # Compute the inner product of the two matrices
    print(inner_product(M1, M2))

    # Compute the outer product of the two matrices
    print(outer_product(M1, M2))

    # Test case 5 - invalid input for inner product (empty matrix)
    M1 = []
    M2 = [[1, 2, 3], [4, 5, 6]]
    # Compute the inner product of the two matrices
    print(inner_product(M1, M2))

    # Compute the outer product of the two matrices
    print(outer_product(M1, M2))
```

```
[[27, 30, 33], [61, 68, 75], [95, 106, 117]]
Error: The number of columns in matrix 1 must be the same as the number of columns in matrix 2 for oute
[[19, 22], [43, 50]]
[[5, 12], [15, 24]]
[[19, 22], [43, 50]]
[[5, 12], [15, 24]]
Error: The number of columns in matrix 1 must be the same as the number of rows in matrix 2 for inner p
[[7, 16, 27], [28, 40, 54]]
[[21, 24, 27], [47, 54, 61]]
Error: The number of columns in matrix 1 must be the same as the number of columns in matrix 2 for oute
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
```

The code includes functions for inner and outer matrix product calculations. The inner_product() function returns the product of two matrices if they have the correct dimensions for matrix multiplication, otherwise it returns an error message. The outer_product() function returns the outer product of two matrices if they have the same number of columns, otherwise it returns an error message.

Five test cases are provided:

1. valid input for both inner and outer product
2. valid input for both inner and outer product
3. invalid input for inner product due to unequal number of columns and rows
4. invalid input for outer product due to unequal number of columns
5. invalid input for inner product due to empty matrix.

These test cases provide a comprehensive testing for the matrix product functions.