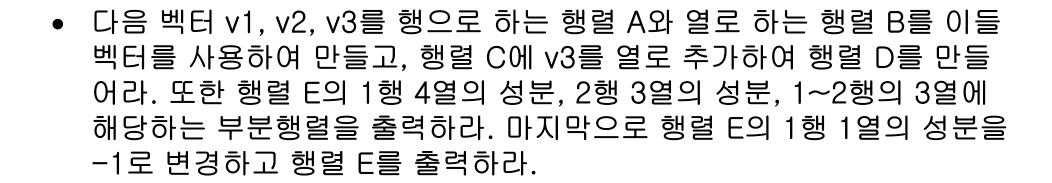
# 선형대수학

이양민(Yang Min Lee)

manson23@nate.com

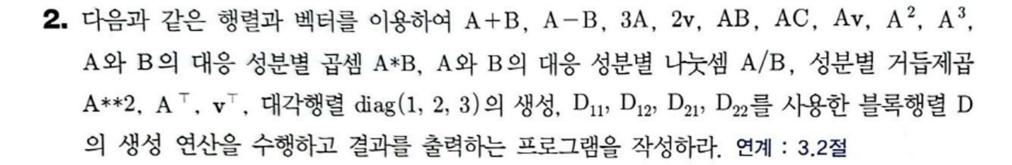
S03-301-01호실

카카오ID: yanwenry



- V1 = [1, 2, 3] V2 = [4, 5, 6], V3 = [7, 8, 9]
- C = [[1, 2], [3, 4], [5, 6]]
- $\blacksquare$  E = [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]]

```
import numpy as np
print("벡터의 결합에 의한 행렬 생성")
v1 = np.array([1, 2, 3])
v2 = np.array([4, 5, 6])
v3 = np.array([7, 8, 9])
A = np.vstack([v1, v2, v3]) # v1, v2, v3를 각각 행으로 하는 행렬 A 생성
print("A =", A)
                                              print("행렬의 성분 접근")
                                              E = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])
B = np.column_stack([v1, v2, v3])_# v1, v2, v3 =
print("B =", B)
                                              print("E[0,3] =", E[0,3]) # 1행 4열의 성분
                                              print("E[1,2] =", E[1,2]) # 2행 3열의 성분
C = np.array([[1, 2], [3, 4], [5, 6]])
                                              print("E[0:2, 2] =", E[0:2, 2]) # E의 1~2행의 3열에 해당하는 부분행렬
print("C =", C)
                                              print("E[0:2, 2:4] =", E[0:2, 2:4]) # E의 1~2행의 3~4열에 해당하는 부분행렬
                                              print("E[2, :] =", E[2, :]) # E의 3행에 해당하는 부분행렬
D = np.column_stack([C, v3]) # 행렬 C에 v3를 열로
print("D =", D)
                                              print("성분의 변경")
                                              print("E =", E)
                                              print("E[0,0] = ", E[0, 0])
                                              E[0, 0] = -1...# E의 1행 1열 성분을 -1로 변경
                                              print(E)
                                              print("E[0,0] = ", E[0, 0])
```



$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 2 \\ 1 & 3 \end{bmatrix}, \quad C = \begin{bmatrix} 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} 10 \\ 20 \end{bmatrix}$$

$$D_{11} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad D_{12} = \begin{bmatrix} 5 \\ 6 \end{bmatrix}, \quad D_{21} = \begin{bmatrix} 7 & 7 \end{bmatrix}, \quad D_{22} = \begin{bmatrix} 8 \end{bmatrix}, \quad D = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix}$$

```
pprint( msg: "A+B", A+B) # 행렬의 합 A+B
import numpy as np
                                                          pprint( msg: "A-B", A-B) # 행렬의 차 A-B
# 행렬 A를 출력하는 함수
                                                          pprint( msg: "3*A ", 3*A) # 행렬의 스칼라배 3A
                                                          pprint( msg: "2*v ", 2*v) # 벡터의 스칼라배 2v
def pprint(msg, A):
    print("---", msg, "---")
                                                          pprint( msg: "matmul(A,B)", np.matmul(A,B)) # 행렬의 곱 AB
                                                          pprint( msg: "matmul(A,C)", np.matmul(A,C)) # 행렬의 곱 AC
    (n,m) = A.shape
                                                          pprint( msg: "A*v", A*v) # 행렬과 벡터의 곱 Av
    for i in range(0, n):
        line = ""
                                                          pprint( msg: "matrix_power(A, 2)", np.linalg.matrix_power(A, n: 2))
        for j in range(0, m):
                                                          pprint( msg: "matrix_power(A, 3)", np.linalg.matrix_power(A, n: 3))
            line += \{0:.2f\}".format(A[i,j]) + "\t"
                                                          pprint( msg: "A*B", A*B) # 행렬의 성분별 곱셈 A*B
        print(line)
                                                          pprint( msg: "A/B", A/B) # 행렬의 성분별 나눗셈 A/B
    print("")
                                                          pprint( msg: "A**2 == A*A", A**2) # 행렬의 성분별 거듭제곱 A**2
A = np.array([[1., 2.], [3., 4.]])
                                                          pprint( msg: "A.T", A.T) # 행렬의 전치 AT
B = np.array([[2., 2.], [1., 3.]])
                                                          pprint( msg: "v.T", v.T) # 벡터의 전치 vT
C = np.array([[4., 5., 6.], [7., 8., 9.]])
                                                          M = np.diag([1, 2, 3]) # 대각행렬 diag(1,2,3) 생성
v = np.array([[10.], [20.]])
                                                          pprint( msg: "diag(1,2,3) =", M)
```

```
D11 = np.array([[1, 2], [3, 4]])
D12 = np.array([[5], [6]])
D21 = np.array([[7, 7]])d
D22 = np.array([[8]])
D = np.block([[D11, D12], [D21, D22]]) # 블록행렬 D 생성
pprint( msg: "block matrix", D)
```



**11.**  $A = \begin{bmatrix} 2 & -5 \\ 3 & -2 \end{bmatrix}$ 에 대하여 다음 행렬 연산을 하라.

(a) 
$$3I_2 - A$$

(b) 
$$(3I_2)A$$

**12.** 주어진 행렬 A에 대하여 다음 행렬 연산을 하라.

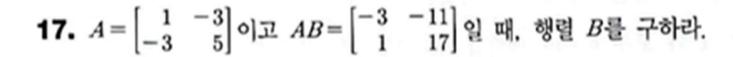
$$A = \begin{bmatrix} 5 & -1 & 3 \\ -4 & 3 & -6 \\ -3 & 1 & 2 \end{bmatrix}$$

(a) 
$$A - 5I_3$$

(b) 
$$(5I_3)A$$

13. 주어진 행렬 A, B에 대하여 AB = BA를 만족하는 k 값을 구하라.

$$A = \begin{bmatrix} 2 & 3 \\ -1 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 9 \\ -3 & k \end{bmatrix}$$



18. 다음 각 행렬의 역행렬을 구하라.

(a) 
$$\begin{bmatrix} 5 & 7 \\ 2 & 4 \end{bmatrix}$$

(b) 
$$\begin{bmatrix} 2 & 6 \\ 1 & 7 \end{bmatrix}$$

(a) 
$$\begin{bmatrix} 5 & 7 \\ 2 & 4 \end{bmatrix}$$
 (b)  $\begin{bmatrix} 2 & 6 \\ 1 & 7 \end{bmatrix}$  (c)  $\begin{bmatrix} 1 & 5 \\ -1 & 2 \end{bmatrix}$  (d)  $\begin{bmatrix} 1 & 3 \\ 2 & 5 \end{bmatrix}$ 

(d) 
$$\begin{bmatrix} 1 & 3 \\ 2 & 5 \end{bmatrix}$$

19. [연습문제 18]의 결과와 [정리 3-8]을 이용하여 다음 행렬의 역행렬을 구하라.

(a) 
$$4\begin{bmatrix} 5 & 7 \\ 2 & 4 \end{bmatrix}$$

(b) 
$$\begin{bmatrix} 5 & -3 \\ -2 & 1 \end{bmatrix}$$

20. [연습문제 18]의 결과와 [정리 3-8]을 이용하여 우변의 행렬의 역행렬을 구하라.

(a) 
$$\begin{bmatrix} 1 & 5 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 2 & 5 \end{bmatrix} = \begin{bmatrix} 11 & 28 \\ 3 & 7 \end{bmatrix}$$

(a) 
$$\begin{bmatrix} 1 & 5 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 2 & 5 \end{bmatrix} = \begin{bmatrix} 11 & 28 \\ 3 & 7 \end{bmatrix}$$
 (b)  $\begin{bmatrix} 1 & 3 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 2 & 5 \end{bmatrix} = \begin{bmatrix} 7 & 18 \\ 12 & 31 \end{bmatrix}$ 

**25.** 행렬 
$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 5 & 6 \\ 1 & 3 & 8 \end{bmatrix}$$
에 대하여 다음 행렬 연산을 하라.

(a) 
$$diag(1, 2, 4) \cdot A$$

(b) 
$$A \cdot diag(1, 2, 4)$$

(c) 
$$diag(1, 1, 1) \cdot A$$

(d) 
$$diag(1, 3, 5, 2) \cdot diag(-1, 2, 3, 1)$$

**26.** 주어진 행렬  $A \sim E$ 에 대하여 다음 행렬 연산을 하라.

$$A = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 5 \end{bmatrix}, \ B = \begin{bmatrix} 1 & 0 & 0 \\ 4 & 2 & 0 \\ -1 & 3 & 4 \end{bmatrix}, \ C = \begin{bmatrix} 10 & 0 & 0 \\ 1 & 0 & 0 \\ 2 & 1 & 5 \end{bmatrix}, \ D = \begin{bmatrix} 2 & 4 & 1 \\ 0 & 3 & 4 \\ 0 & 0 & 7 \end{bmatrix}, \ E = \begin{bmatrix} 9 & 6 & 3 \\ 0 & -2 & 4 \\ 0 & 0 & 11 \end{bmatrix}$$

- (a) *BA*
- (b) *BC*

- (c) DE (d) ABC

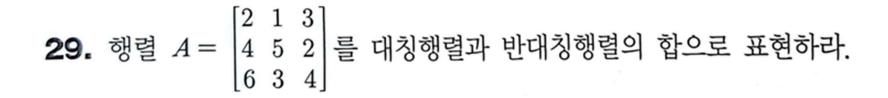
(e)  $E^{\mathsf{T}}$ 

**27.** 다음 열벡터 u와 행벡터 v의 곱 uv를 계산하라.

$$\boldsymbol{u} = \begin{bmatrix} 2\\4\\1 \end{bmatrix}, \quad \boldsymbol{v} = \begin{bmatrix} 1 & 3 & 2 \end{bmatrix}$$

**28.** 행렬 A가 다음과 같이 부분행렬들로 분할될 때, 부분행렬  $A_{12}$ 를 구하라.

$$A = \begin{bmatrix} 1 & 5 & 3 & 13 \\ 7 & 4 & 2 & -1 \\ \hline 11 & 1 & 3 & -2 \end{bmatrix}$$



35. 주어진 행렬에 대하여 다음 각 블록행렬의 곱을 계산하라.

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad E = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad 0 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

$$B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}, \quad B_{11} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \quad B_{12} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \quad B_{21} = \begin{bmatrix} 3 & 1 \\ 3 & 2 \end{bmatrix}, \quad B_{22} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}$$

(a) 
$$\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$
 (b)  $\begin{bmatrix} C & 0 \\ 0 & C \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$  (c)  $\begin{bmatrix} D & I \\ I & E \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$