

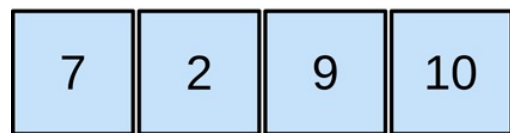
신경망 데이터 표현

김영욱

Hello AI

텐서(Tensor)

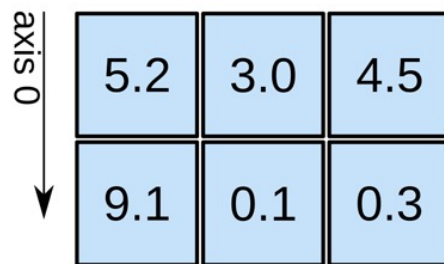
1D array



axis 0 →

shape: (4,)

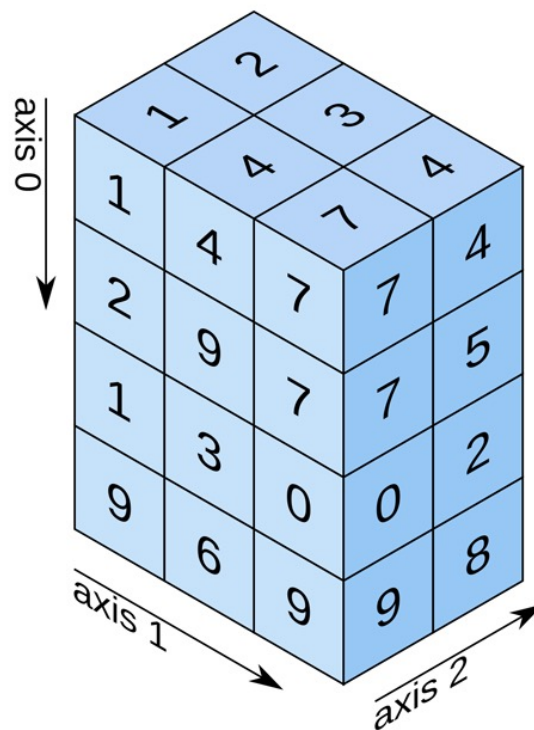
2D array



axis 1 →

shape: (2, 3)

3D array



shape: (4, 3, 2)

텐서(Tensor)

스칼라(0차원 텐서)

- 하나의 숫자를 담고 있는 텐서(tensor)
- 형상은 없음

벡터의 곱

- $A = (x_1, x_2, x_3, \dots, x_n)$
 $B = (y_1, y_2, y_3, \dots, y_n)$ 일 때,
- 원소곱
 - 같은 형상(shape)일 때, 각 원소별로 계산

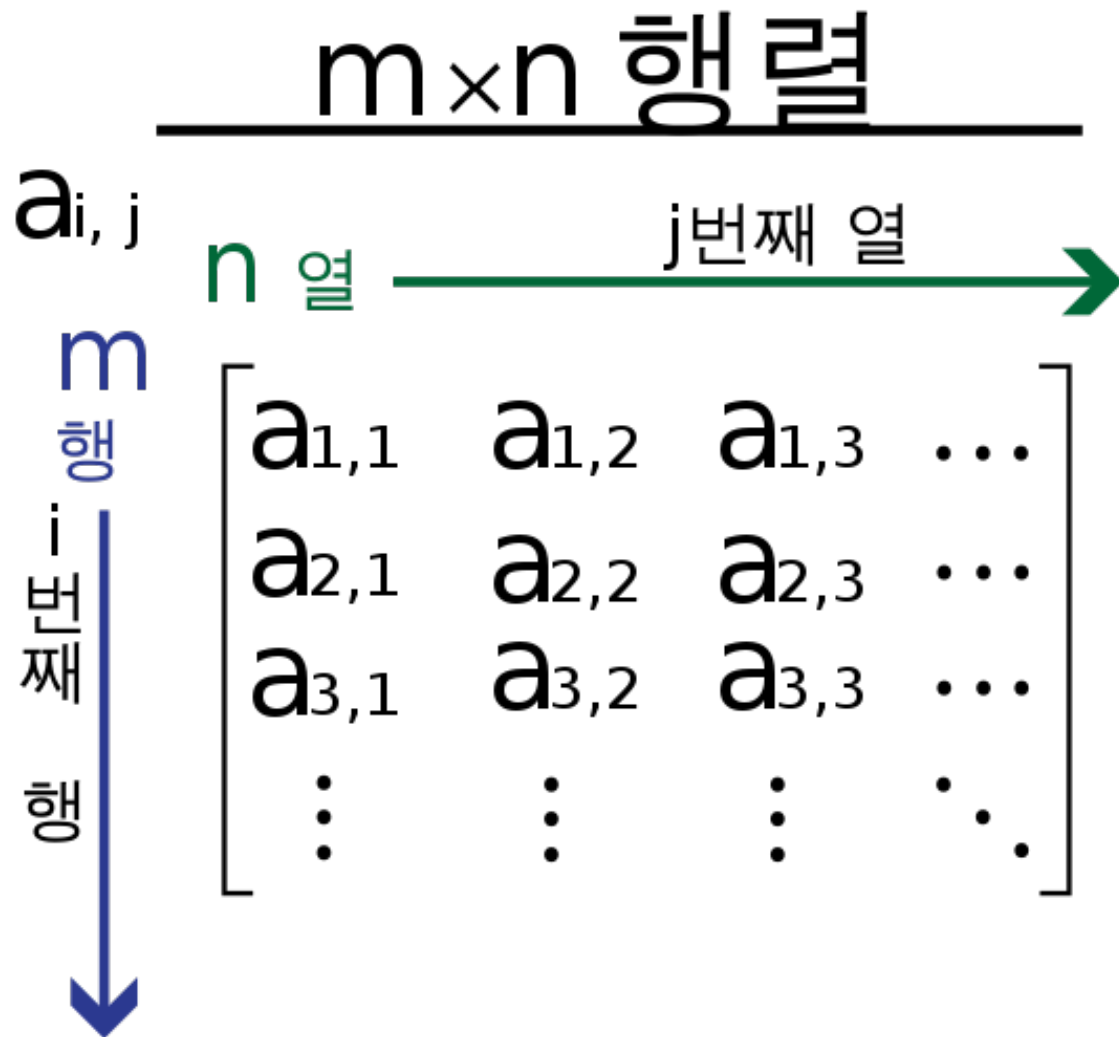
$$\begin{aligned} A \times B &= (x_1, x_2, x_3, \dots, x_n) \times (y_1, y_2, y_3, \dots, y_n) \\ &= (x_1 y_1, x_2 y_2, x_3 y_3, \dots, x_n y_n) \end{aligned}$$

- 벡터곱(product, dot)
 - 두 1차원 벡터가 있을 때 각각의 성분끼리의 곱을 모두 더하는 계산

$$\begin{aligned} A \bullet B \Rightarrow A \times B^T &= (x_1, x_2, x_3, \dots, x_n) \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_n \end{pmatrix} \\ &= (x_1 y_1 + x_2 y_2 + x_3 y_3 + \dots + x_n y_n) \end{aligned}$$

2차원 텐서(행렬)

- 2차원 텐서는 행렬로 생각할 수 있음
 - (m, n) 형상의 배열



전치행렬

- 행과 열을 바꾼 배열의 형태

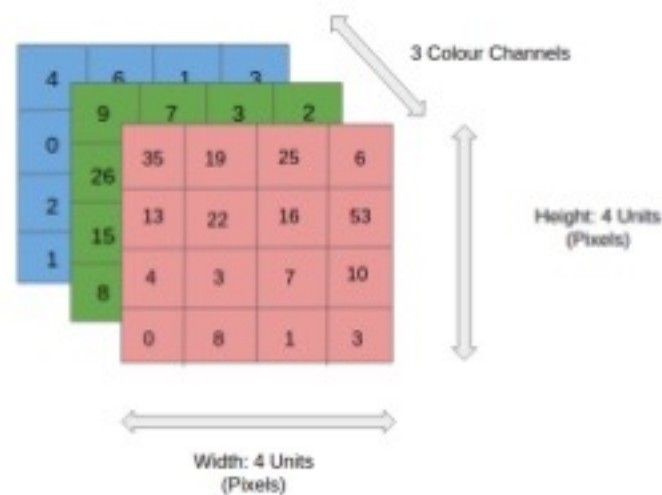
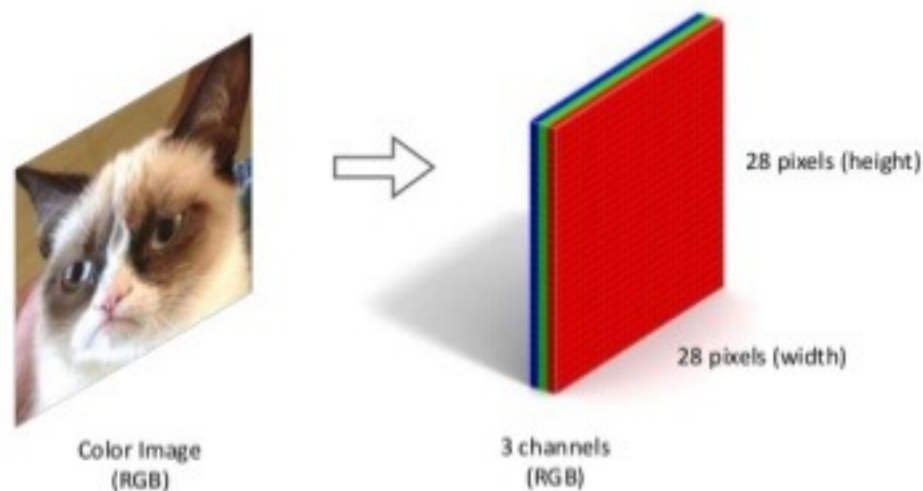
A

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

3차원 텐서

- 보통 이미지를 나타낼 때 사용되는 텐서
 - (width, height, channels)
 - 일반적으로 Numpy array로 표현

color image is 3rd-order tensor

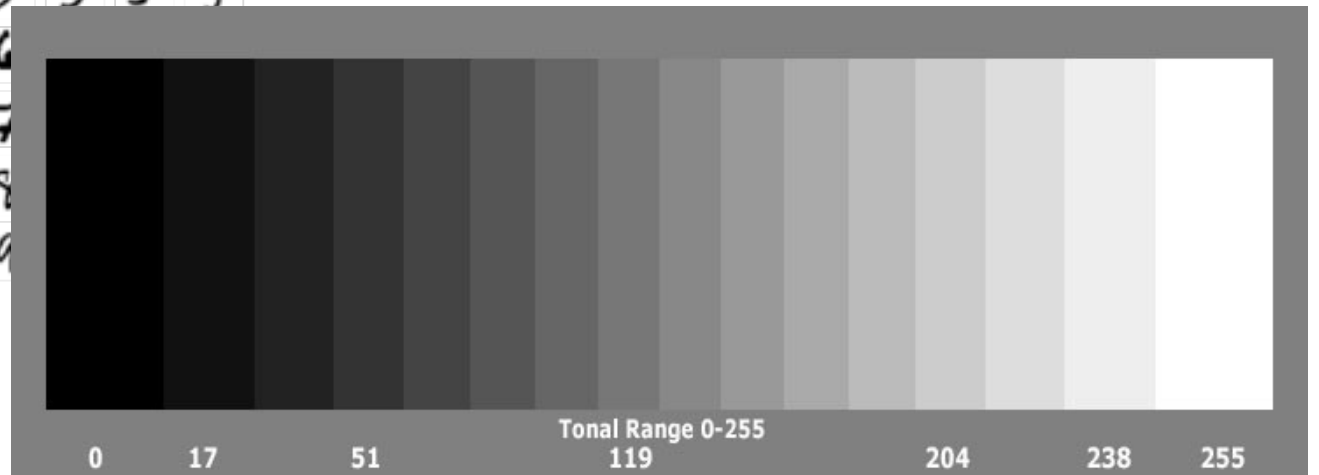


3차원 텐서 활용 예시(이미지)

- MNIST Dataset
- 28x28 사이즈의 gray scale 이미지들로 구성

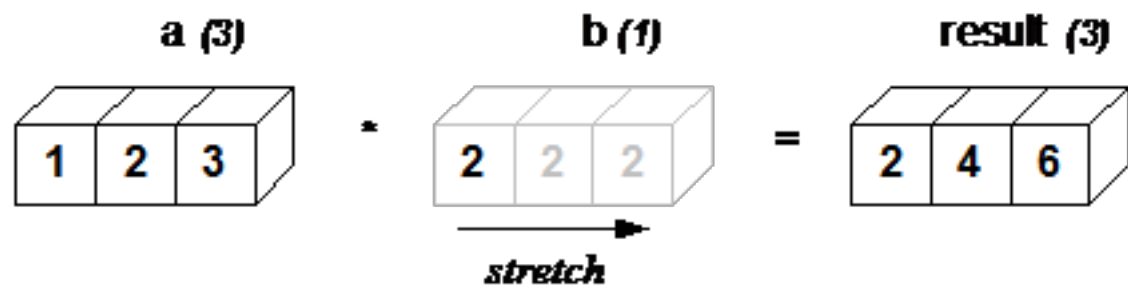


gray scale: 0~255의 값을 통해 밝기를
표현 0으로 갈수록 어두워지고, 255로
갈수록 밝아짐

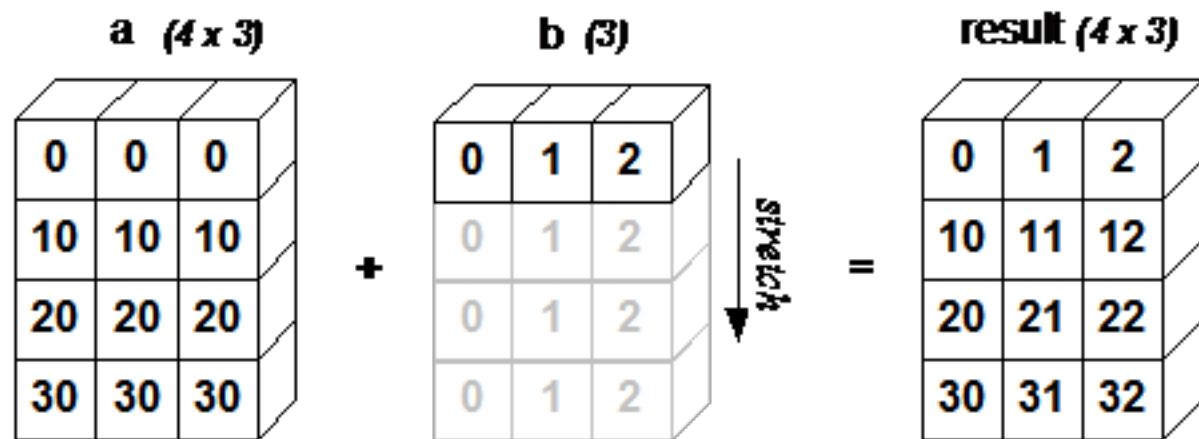


브로드캐스팅(broadcasting)

- 넘파이에서 다른 형상(shape)끼리 계산 가능
- 1차원 텐서



- 2차원 텐서



4, 5차원 텐서

- Color Image Datasets(4차원)
 - (samples, height, width, channels) (Keras, Tensorflow)
 - (samples, channels, height, width) (Pytorch)
- 동영상(5차원)
 1. (samples, frames, height, width, channels)
 2. (samples, frames, channels, height, width)
 - 예시 1) (4, 300, 1920, 1080, 3)
 - 1920x1080 사이즈 3채널의 300프레임 수를 가진 배치 4개