Tensor Layouts

Array/Vector (1D)

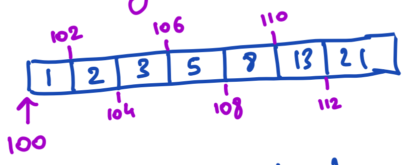
Logical view

1	2	3	5	8	13	21

Shape: [7]

Stride: [1]

Physical view



In C, we get a pointer to the location in memory where the pointer starts and we use it to access all the rest of the elements

Matrix (2D)

Logical view

1	2	3
5	8	13

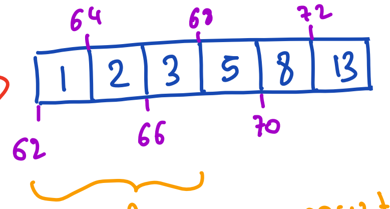
= T

Shape: [2,3]

Stride: [3,1]

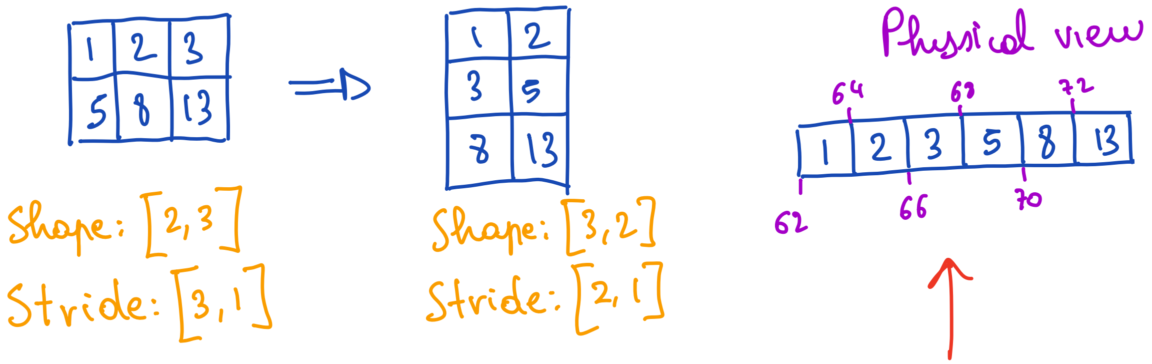
Note: this is known as a row-major layout.

Physical view



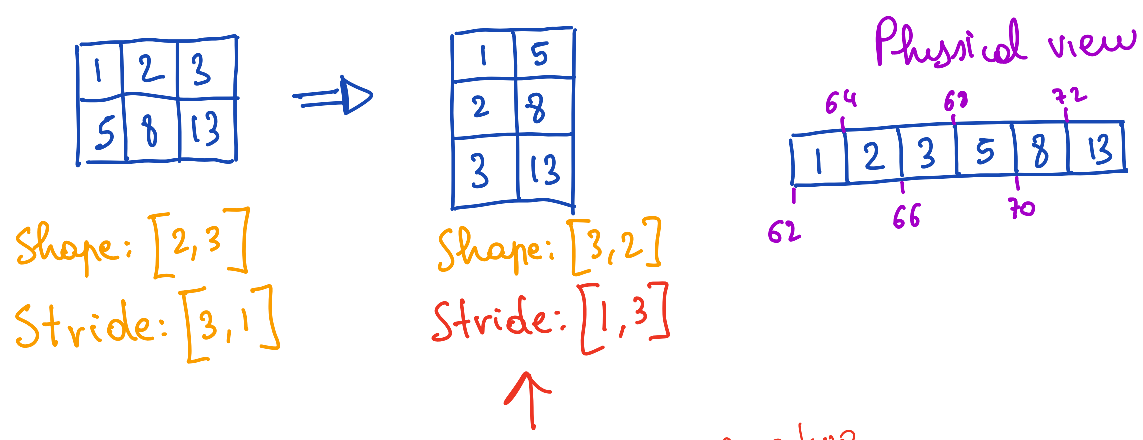
To go from one row to the next, we need to skip three elements

Matrix (2D) - Reshape



Note: physical arrangement does not change!

Matrix (2D) - Transpose



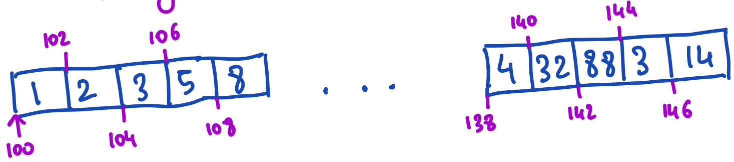
To transpose along two dimensions, we need to swap the shape and stride of the two dimensions ... and that's it!

Note: after transposing, the tensor is no longer "contiguous". that's why in PyTorch you can't "view" but you need to "reshape".

Tensor (3D)

$$\left[\begin{array}{c}\begin{array}{|c|c|c|}\hline 1 & 2 & 3 \\ \hline 5 & 8 & 13 \\ \hline 21 & 34 & 55 \\ \hline 9 & 11 & 13 \\ \hline\end{array} \\ \\ \begin{array}{|c|c|c|}\hline 72 & 42 & 2 \\ \hline 31 & 1 & 92 \\ \hline 7 & 4 & 32 \\ \hline 88 & 3 & 14 \\ \hline\end{array}\end{array}\right]$$

Physical view:



Shape: [2, 4, 3]

Stride: [12, 3, 1]

$$\text{Stride} = \begin{cases} \text{Stride}[i] = \prod_{j=i+1}^{N} \text{shape}[j] \\ \\ \text{Stride}[N] = 1 \end{cases}$$