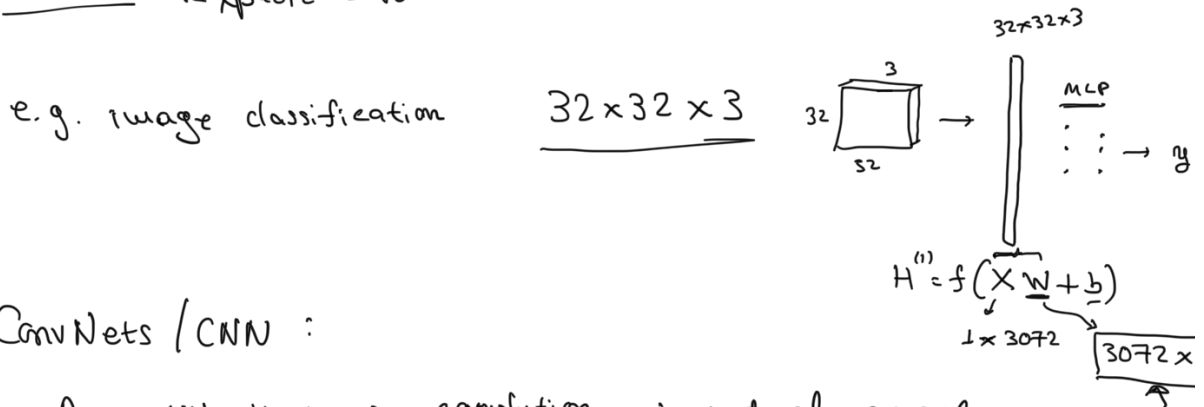


# Convolutional Neural Networks

Motivation: Address the unfavorable complexity of MLPs for high-dimensional inputs/outputs.

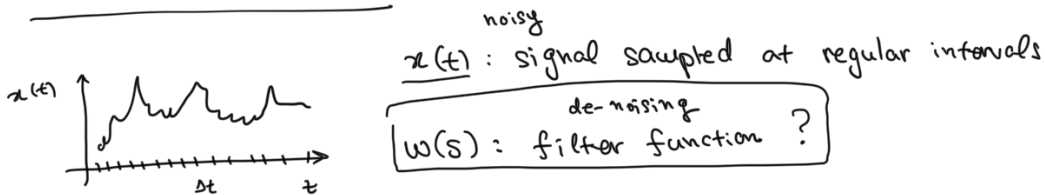
Idea: Exploit structure in the data (e.g. data lives on "grid").



ConvNets / CNN:

Are NN that use convolution instead of general matrix multiplication in at least one of their layers.

Definition in 1D:

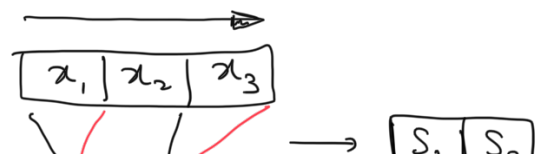


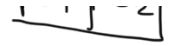
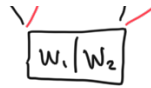
$$S(t) = \int_{-\infty}^{+\infty} x(s) w(t-s) ds = \int_{-\infty}^{+\infty} w(s) x(t-s) ds$$

$$S(t) = (x * w)(t)$$

Discrete convolution in 1D:

$$S(t) = \sum_{s=-N}^{N/2} x(s) w(t-s)$$





$$\underline{S_1} = x_1 w_1 + x_2 w_2$$

$$S_2 = w_1 x_2 + w_2 x_3$$

Discrete convolution in 2D:

$$I(i, j) = \sum_m \sum_n I(m, n) K(i-m, j-n)$$

$$= \sum_m \sum_n K(m, n) I(i-m, j-n)$$