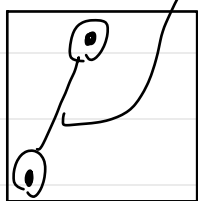


# Rede Neural Convolucional

↳ Até o momento temos como referência as MLP que tem camadas totalmente conectadas. O problema dessa abordagem é que ela não leva em consideração a estrutura espacial das imagens.

Por exemplo:



↳ **Pixels**: pixels que estão distantes um do outro serão tratados da **mesma maneira**.

↳ Então é aí que as CNN's entram em ação. Elas conseguem tomar proveito da estrutura espacial.

↳ **Funções convolucionais**

- Funções muito utilizadas na engenharia que transformam uma função do domínio do tempo  $\rightarrow$  Frequência

- Na matéria de Sinais e Sistemas vemos que trabalham no domínio da frequência por ser mais barato computacionalmente. Pois integrais ( $\int$ ) são mais comuns.

$$\int_{-\infty}^{+\infty} f(\tau) g(x - \tau) d\tau$$

↳ Digamos a seguinte imagem:

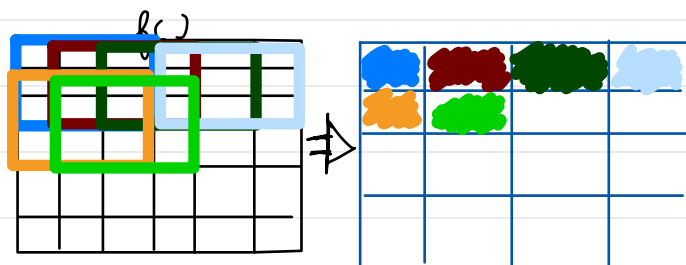
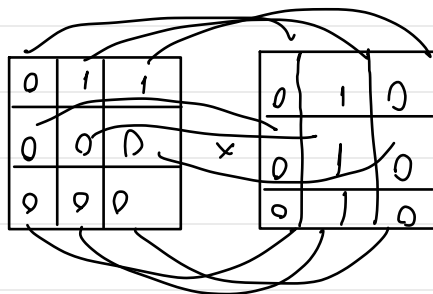
0	1	1	1	1	0
0	0	0	1	1	0
0	0	0	0	1	0
0	0	0	0	1	0
0	0	0	0	1	0
0	0	0	0	1	0

↳ Realizar a convolução é basicamente fazer uma transformação matemática na matriz usando uma matriz

filtro:

0	1	0
0	1	0
0	1	0

e gerar uma matriz menor



## Técnicas de Melhoria de Resultados:

**Padding:** para que a matriz resultante da convolução destaque melhor os padrões completos a matriz original para ela ter uma dimensão maior

0	1	1	1	1	0
0	0	0	0	1	0
0	0	0	0	1	0
0	0	0	0	1	0
0	0	0	0	1	0
0	0	0	0	1	0

⇒

0	0	0	0	0	0	0	0
0	0	1	1	1	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0

**Max Pooling:** serve para comprimir a imagem. Normalmente utilizada na seguinte ordem **Padding → Convolução → Max Pooling**

0	1	1	2	1	0
0	0	0	0	1	0
0	0	0	0	1	0
0	0	0	0	1	0
0	0	0	0	1	0
0	0	0	0	1	0

⇒

1	2	1
0	0	3
8	0	4

**Uls:** basicamente pega o maior valor dentro de cada cell

**Flatten:** é basicamente achatar a matriz em uma linha

