

Lista 8 - Exercícios Redes Neurais:

[Prof. Dr. Jodavid Ferreira](#)

Discente:

* Gabriel D'assumpção de Carvalho

Data: 10/09/2024

1. Uma rede neural convolucional (CNN) é composta minimamente de quantos componentes, ou seja, quais são os componentes mínimos para formar uma arquitetura de CNN? Quais são esses componentes? (Considere as camadas de entrada e saída como componentes).

Uma rede neural convolucional (CNN) é composta minimamente de 4 componentes, essenciais para formar uma arquitetura básica, são eles:

1. Camada de entrada: Recebe os dados de entrada, em formato de matriz (por exemplo uma imagem RGP, onde vai possuir dimensões como altura, largura, e três canais de cor).
2. Camada de convolução: Aplica os filtros (kernels) que "varrem" a imagem para extrair características como bordas, texturar, linhas horizontais ou verticais. Essa camada é responsável por detectar de padrões especiais.
3. Camada de pooling (**OPCIONAL**): Essa camada reduz a dimensionalidade das características extraídas na camada de convolução, tornando o modelo mais eficiência e ajudando a tornar o modelo mais robusto a pequenas variações nas imagens
4. Camada totalmente conectada: Após as extrações de características, a camada totalmente conectada processa todas as informações para realizar a classificação ou tarefa desejada, conectando todos os filtros utilizado na camada anterior.
5. Camada de saída: A última camada da rede, responsável pela saída final do modelo, como a classe prevista no caso da tarefa de classificação. Geralmente utiliza uma função de ativação como softmax para prever a probabilidade entre as classes.

2. Qual a função das Camadas Convolucionais?

Como mencionado anteriormente, a camada de convolução tem como objetivo extrair características como borda, textura, linhas horizontais e verticais. No entanto, essa camada é responsável por detectar padrões especiais.

3. Qual o conceito de Stride e Padding?

O stride e padding são parâmetros adicionados as camadas de convolução. O stride é o número de píxel que o filtro (kernel) vai se mover a cada passo, um stride de 1 significa que o filtro vai se mover um píxel por vez. O padding é a adição de píxel em volta da imagem de entrada, para que a imagem depois da convolução tenha dimensões iguais, podendo ser "valid" (sem padding) ou "same" (com padding).

4. Considere uma a matriz Imagem que possui dimensões 5x5 e um filtro com dimensões 3x3, a seguir.

A matriz Imagem é dada por:

$$\text{Imagem} = \begin{pmatrix} 1 & 2 & 3 & 0 & 1 \\ 4 & 5 & 6 & 1 & 2 \\ 7 & 8 & 9 & 2 & 3 \\ 0 & 1 & 2 & 3 & 0 \\ 3 & 2 & 1 & 0 & 4 \end{pmatrix}$$

E o Filtro é dado por:

$$\text{Filtro} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

Calcule a convolução dessa matriz de imagem com o filtro utilizando uma operação de convolução com **STRIDE=1** e **sem padding** e apresente o resultado final.

```
In [14]: import numpy as np

img = np.array(
    [
        [1, 2, 3, 0, 1],
```

```
[4, 5, 6, 1, 2],
[7, 8, 9, 2, 3],
[0, 1, 2, 3, 0],
[3, 2, 1, 0, 4],
]
)

kernel = np.array([[1, 0, 1], [0, 1, 0], [1, 0, 1]])

conv_img = np.array(
[
[
1 * 1 + 2 * 0 + 3 * 1 + 4 * 0 + 5 * 1 + 6 * 0 + 7 * 1 + 8 * 0 + 9 * 1,
2 * 1 + 3 * 0 + 0 * 1 + 5 * 0 + 6 * 1 + 1 * 0 + 8 * 1 + 9 * 0 + 2 * 1,
3 * 1 + 0 * 0 + 1 * 1 + 6 * 0 + 1 * 1 + 2 * 0 + 9 * 1 + 2 * 0 + 3 * 1
],
[
4 * 1 + 5 * 0 + 6 * 1 + 7 * 0 + 8 * 1 + 9 * 0 + 0 * 1 + 1 * 0 + 2 * 1,
5 * 1 + 6 * 0 + 1 * 1 + 8 * 0 + 9 * 1 + 2 * 0 + 1 * 1 + 2 * 0 + 3 * 1,
6 * 1 + 1 * 0 + 2 * 1 + 9 * 0 + 2 * 1 + 3 * 0 + 2 * 1 + 3 * 0 + 0 * 1
],
[
7 * 1 + 8 * 0 + 9 * 1 + 0 * 0 + 1 * 1 + 2 * 0 + 3 * 1 + 2 * 0 + 1 * 1,
8 * 1 + 9 * 0 + 2 * 1 + 1 * 0 + 2 * 1 + 3 * 0 + 2 * 1 + 1 * 0 + 0 * 1,
9 * 1 + 2 * 0 + 3 * 1 + 2 * 0 + 3 * 1 + 1 * 0 + 1 * 1 + 0 * 0 + 4 * 1
]
]
)

print(conv_img)
```

```
[[25 18 17]
 [20 19 12]
 [21 14 20]]
```

In [13]: **from** scipy.signal **import** convolve2d

```
# Convolução usando scipy
conv_result = convolve2d(img, kernel, mode='valid')
print(conv_result)
```

```
[[25 18 17]
 [20 19 12]
 [21 14 20]]
```

5. Considere agora as seguintes configurações para dimensões e processo de convolução da imagem e do filtro:

Imagem: 15x15 (altura x largura)

Filtro: 5x5 (altura x largura)

Stride: 1

Padding: same (Half padding)

Qual será a dimensão da saída após a convolução da imagem com o filtro?

Como o stride é igual a 1 e a convolução está com o padding ligado a dimensão da saída após o filtro vai ser de 15x15, igual a imagem de entrada.

6. Considere agora as seguintes configurações para dimensões e processo de convolução da imagem e do filtro:

Imagem: 15x15 (altura x largura)

Filtro: 5x5 (altura x largura)

Stride: 1 (unitário)

Padding: Zero padding

Qual será a dimensão da saída após a convolução da imagem com o filtro?

Como a a camada de convolução tem stride igual a 1 e não é aplicado padding, após o deslizamento da kernel sobre a imagem a saída dessa camada vai ser igual a dimensão da imagem menos a do filtro mais o número de stride, que nesse caso vai ter como saída uma imagem 11x11.

1. Considere uma matriz de imagem 8x8, conforme mostrado abaixo:

In [15]: `img = np.array(`

```
[
    [1, 3, 2, 4, 6, 8, 7, 5],
    [5, 6, 4, 3, 1, 2, 9, 8],
    [7, 8, 5, 6, 9, 4, 3, 2],
```

```
        [9, 7, 1, 8, 2, 3, 5, 4],
        [4, 2, 6, 1, 8, 7, 9, 6],
        [3, 5, 7, 2, 4, 1, 6, 8],
        [8, 9, 4, 5, 3, 2, 7, 1],
        [6, 4, 2, 3, 7, 5, 8, 9],
    ]
)

max_pooling = np.array(
    [
        [6, 4, 8, 9],
        [9, 8, 9, 5],
        [5, 7, 8, 9],
        [9, 5, 7, 9],
    ]
)

print(max_pooling)
```

```
[[6 4 8 9]
 [9 8 9 5]
 [5 7 8 9]
 [9 5 7 9]]
```