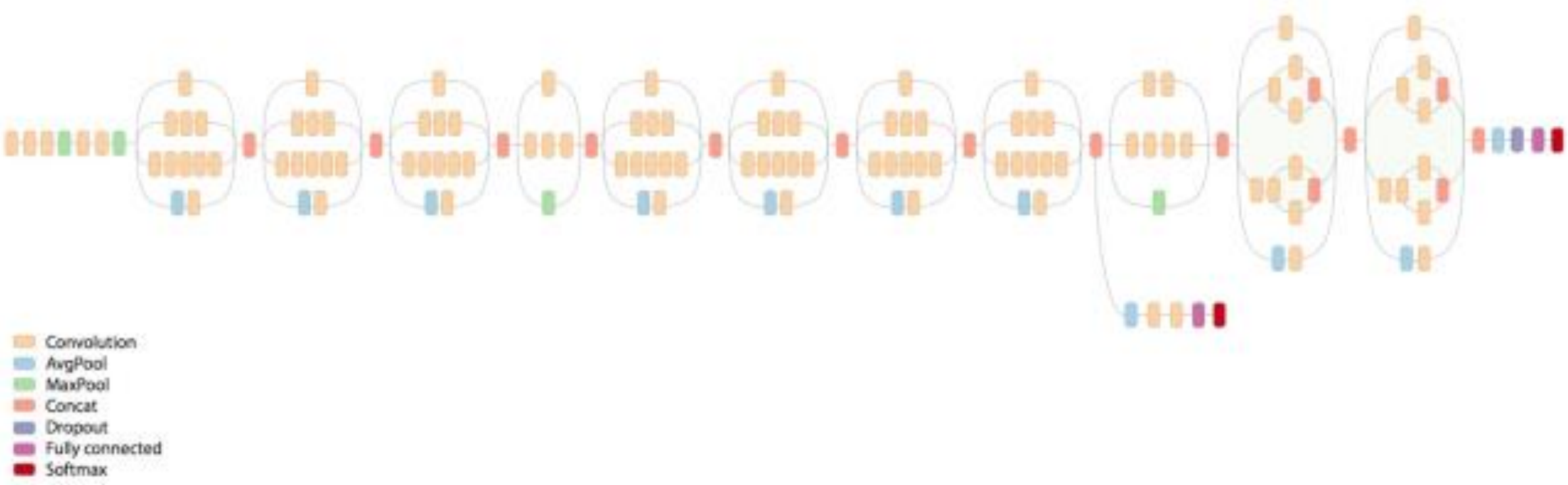
Aprendizado de Máquina
e Deep Learning

Redes convolucionais com Keras

Prof. Dr. Thiago Meirelles Ventura

Redes neurais convolucionais



Camadas no Keras

- Conv2D
- Pooling (Max e Average)
- Visualização das camadas

Camadas no Keras - Conv2D

```
keras.layers.Conv2D(
    filters,
    kernel_size,
    strides=(1, 1),
    padding='valid',
    activation=None,
    input_shape=(32, 32, 3))
```

Camadas no Keras - Conv2D

```
keras.layers.Conv2D(
    filters,
    kernel_size,
    strides=(1, 1),
    padding='valid',
    activation=None,
    input_shape=(32, 32, 3))
```

Quantidade de filtros

Camadas no Keras - Conv2D

```
keras.layers.Conv2D(
    filters,
    kernel_size,
    strides=(1, 1),
    padding='valid',
    activation=None,
    input_shape=(32, 32, 3))
```

Tamanho do kernel.
Número inteiro ou tupla.

Camadas no Keras - Conv2D

```
keras.layers.Conv2D(
    filters,
    kernel_size,
    strides=(1, 1),
    padding='valid',
    activation=None,
    input_shape=(32, 32, 3))
```

Tamanho do stride.
Pode ser um inteiro também.

Camadas no Keras - Conv2D

```
keras.layers.Conv2D(
    filters,
    kernel_size,
    strides=(1, 1),
    padding='valid',
    activation=None,
    input_shape=(32, 32, 3))
```

valid: não aplica padding, causando redução na dimensão ao aplicar a convolução.

same: mantém a mesma dimensão da entrada aplicando o padding.

Camadas no Keras - Conv2D

```
keras.layers.Conv2D(
    filters,
    kernel_size,
    strides=(1, 1),
    padding='valid',
    activation=None,
    input_shape=(32, 32, 3))
```

Função de ativação.

Camadas no Keras - Conv2D

```
keras.layers.Conv2D(
    filters,
    kernel_size,
    strides=(1, 1),
    padding='valid',
    activation=None,
    input_shape=(32, 32, 3))
```

Formato da entrada

Camadas no Keras - Pooling

- MaxPooling2D
- AveragePooling2D

Camadas no Keras - Pooling

- MaxPooling2D

```
keras.layers.MaxPooling2D(
    pool_size=(2, 2),
    strides=None)
```

Camadas no Keras - Pooling

- MaxPooling2D

```
keras.layers.MaxPooling2D(
    pool_size=(2, 2),
    strides=None)
```

Tamanho do pooling. Pode ser apenas 1 inteiro.

Camadas no Keras - Pooling

- MaxPooling2D

```
keras.layers.MaxPooling2D(
    pool_size=(2, 2),
    strides=None)
```

Pode ser utilizado stride também

Camadas no Keras - Pooling

- AveragePooling2D

```
keras.layers.AveragePooling2D(
    pool_size=(2, 2),
    strides=None)
```

Camadas no Keras - Visualização

```
model.summary()
```

Camadas no Keras - Visualização

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 26, 26, 64)        640
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 13, 13, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 11, 11, 64)        36928
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 5, 5, 64)          0
_________________________________________________________________
flatten (Flatten)            (None, 1600)              0
_________________________________________________________________
dense (Dense)                (None, 128)               204928
_________________________________________________________________
dense_1 (Dense)              (None, 10)                1290
=================================================================
Total params: 243,786
Trainable params: 243,786
Non-trainable params: 0
```

Camadas no Keras - Visualização

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 26, 26, 64)        640
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 13, 13, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 11, 11, 64)        36928
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 5, 5, 64)          0
_________________________________________________________________
flatten (Flatten)            (None, 1600)              0
_________________________________________________________________
dense (Dense)                (None, 128)               204928
_________________________________________________________________
dense_1 (Dense)              (None, 10)                1290
=================================================================
Total params: 243,786
Trainable params: 243,786
Non-trainable params: 0
```

Camadas no Keras - Visualização

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 26, 26, 64)        640
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 13, 13, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 11, 11, 64)        36928
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 5, 5, 64)          0
_________________________________________________________________
flatten (Flatten)            (None, 1600)              0
_________________________________________________________________
dense (Dense)                (None, 128)               204928
_________________________________________________________________
dense_1 (Dense)              (None, 10)                1290
=================================================================
Total params: 243,786
Trainable params: 243,786
Non-trainable params: 0
```

Camadas no Keras - Visualização

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 26, 26, 64)        640
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 13, 13, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 11, 11, 64)        36928
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 5, 5, 64)          0
_________________________________________________________________
flatten (Flatten)            (None, 1600)              0
_________________________________________________________________
dense (Dense)                (None, 128)               204928
_________________________________________________________________
dense_1 (Dense)              (None, 10)                1290
=================================================================
Total params: 243,786
Trainable params: 243,786
Non-trainable params: 0
```

Camadas no Keras - Visualização

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 26, 26, 64)        640
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 13, 13, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 11, 11, 64)        36928
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 5, 5, 64)          0
_________________________________________________________________
flatten (Flatten)            (None, 1600)              0
_________________________________________________________________
dense (Dense)                (None, 128)               204928
_________________________________________________________________
dense_1 (Dense)              (None, 10)                1290
=================================================================
Total params: 243,786
Trainable params: 243,786
Non-trainable params: 0
```

Exercício 4

Crie uma rede convolucional com as seguintes camadas:

Conv2D → MaxPooling2D → Conv2D → MaxPooling2D → Flatten → Dense → Dense

Verifique como ficou as camadas

Exercício 5

Classifique as imagens da base CIFAR-10

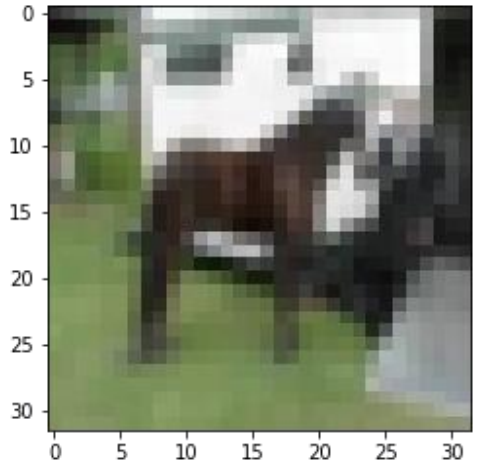
Compare com o desempenho obtido com uma rede neural sem convolução

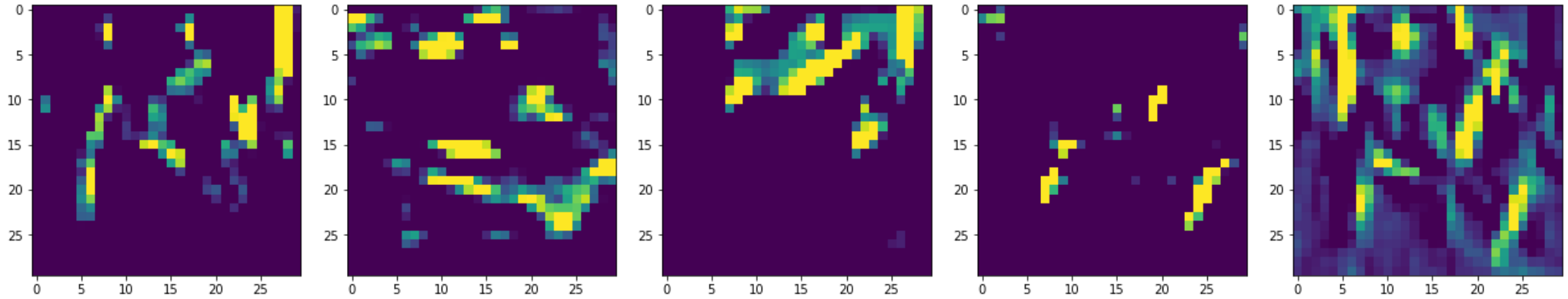
Análise do resultado das camadas

```
outputs = [layer.output for layer in model.layers]

visualization_model = keras.models.Model(
                                inputs = model.input,
                                outputs = successive_outputs)

feature_maps = visualization_model.predict(x)
```

```
plt.imshow(x_test[0])
plt.show()
successive_outputs = [layer.output for layer in model.layers]
visualization = keras.models.Model(inputs = model.input,
                                   outputs = successive_outputs)
x = x_test[0].reshape((1,) + image.shape)
successive_feature_maps = visualization.predict(x)
feature_map = successive_feature_maps[0]
n_features = feature_map.shape[-1]
for i in range(n_features):
  x = feature_map[0, :, :, i]
  x -= x.mean(); x /= x.std(); x *= 64; x += 128
  x = np.clip(x, 0, 255).astype('uint8')
  plt.imshow(x)
  plt.show()
```

Análise do resultado das camadas





Redes neurais convolucionais