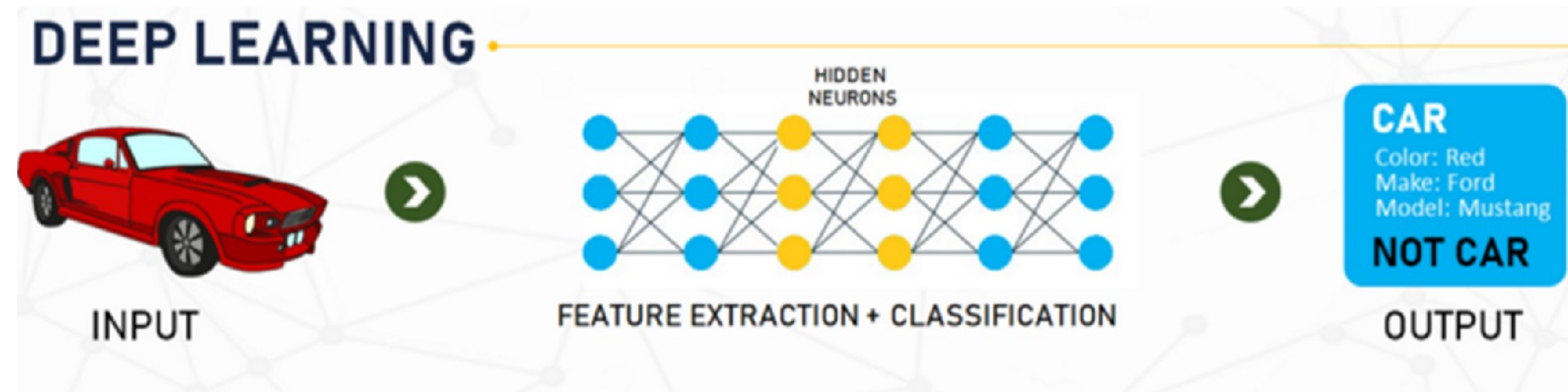




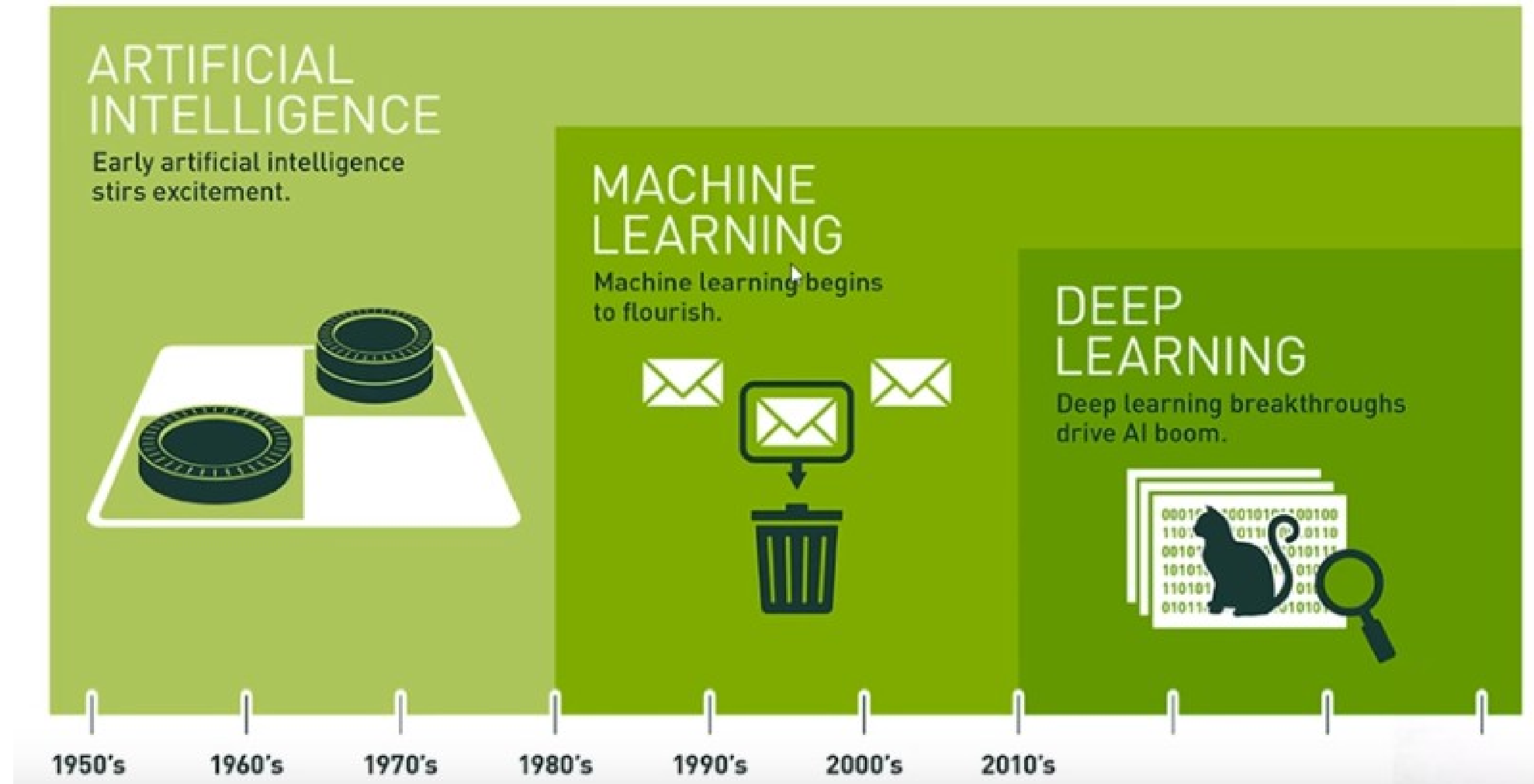
Sistemas Inteligentes Deep Learning

Semana 14 – Introdução a Deep Learning

Introdução a Deep Learning



Introdução a Deep Learning



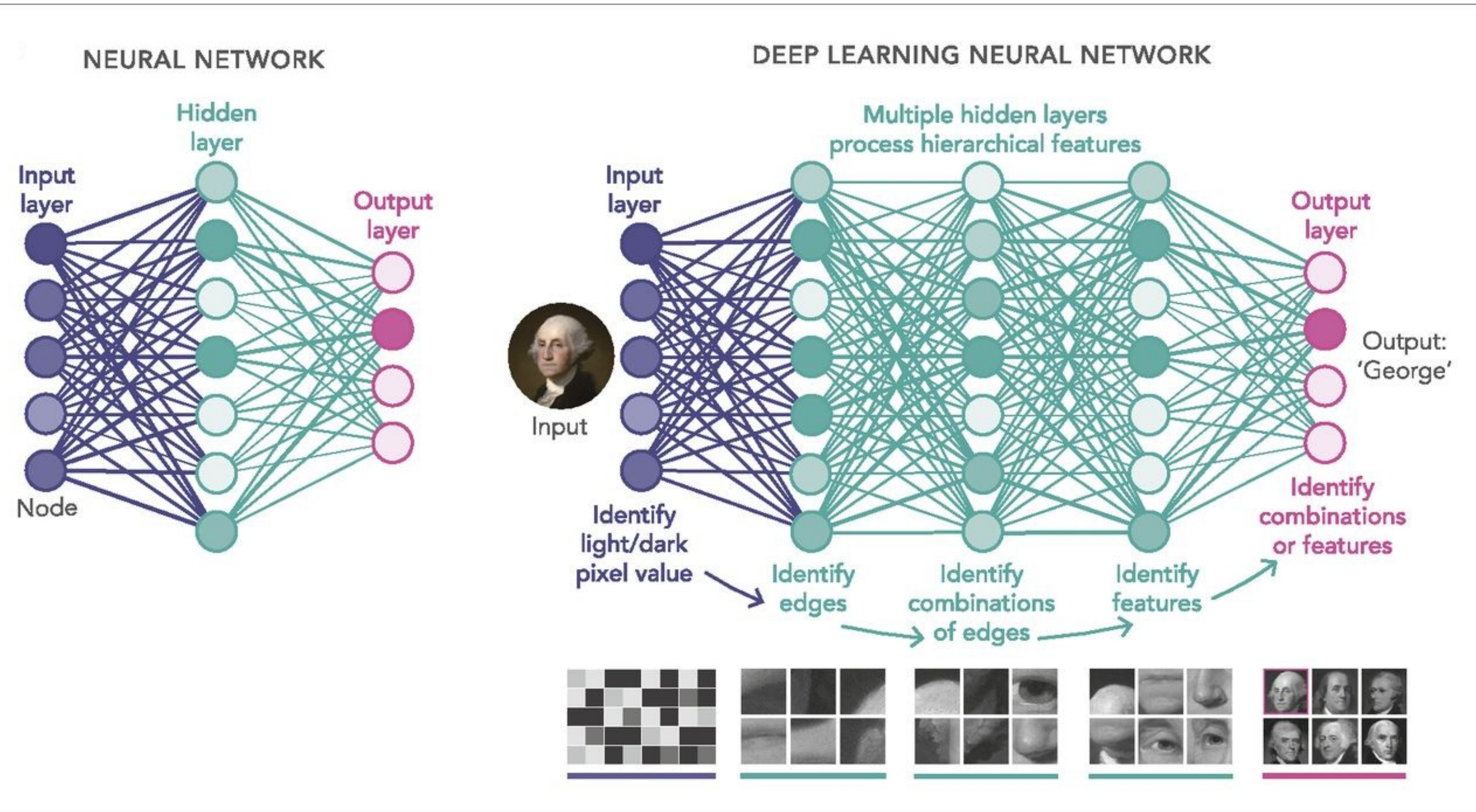
Introdução a Deep Learning

O que o computador enxerga?



```
08 02 22 97 38 15 00 40 00 75 04 05 07 78 52 12 50 77 91 08
49 49 99 40 17 81 18 57 60 87 17 40 98 43 69 48 04 56 62 00
81 49 31 73 55 79 14 29 93 71 40 67 53 88 30 03 49 13 36 65
52 70 95 23 04 60 11 42 69 24 68 56 01 32 56 71 37 02 36 91
22 31 16 71 51 67 63 89 41 92 36 54 22 40 40 28 66 33 13 80
24 47 32 60 99 03 45 02 44 75 33 53 78 36 84 20 35 17 12 50
32 98 81 28 64 23 67 10 26 38 40 67 59 54 70 66 18 38 64 70
67 26 20 68 02 62 12 20 95 63 94 39 63 08 40 91 66 49 94 21
24 55 58 05 66 73 99 26 97 17 78 78 96 83 14 88 34 89 63 72
21 36 23 09 75 00 76 44 20 45 35 14 00 61 33 97 34 31 33 95
78 17 53 28 22 75 31 67 15 94 03 80 04 62 16 14 09 53 56 92
16 39 05 42 96 35 31 47 55 58 88 24 00 17 54 24 36 29 85 57
86 56 00 48 35 71 89 07 05 44 44 37 44 60 21 58 51 54 17 58
19 80 81 68 05 94 47 69 28 73 92 13 86 52 17 77 04 89 55 40
04 52 08 83 97 35 99 16 07 97 57 32 16 26 26 79 33 27 98 66
88 36 68 87 57 62 20 72 03 46 33 67 46 55 12 32 63 93 53 69
04 42 16 73 38 25 39 11 24 94 72 18 08 46 29 32 40 62 76 36
20 69 36 41 72 30 23 88 34 62 99 69 82 67 59 85 74 04 36 16
20 73 35 29 78 31 90 01 74 31 49 71 48 86 81 16 23 57 05 54
01 70 54 71 83 51 54 69 16 92 33 48 61 43 52 01 89 19 67 48
```

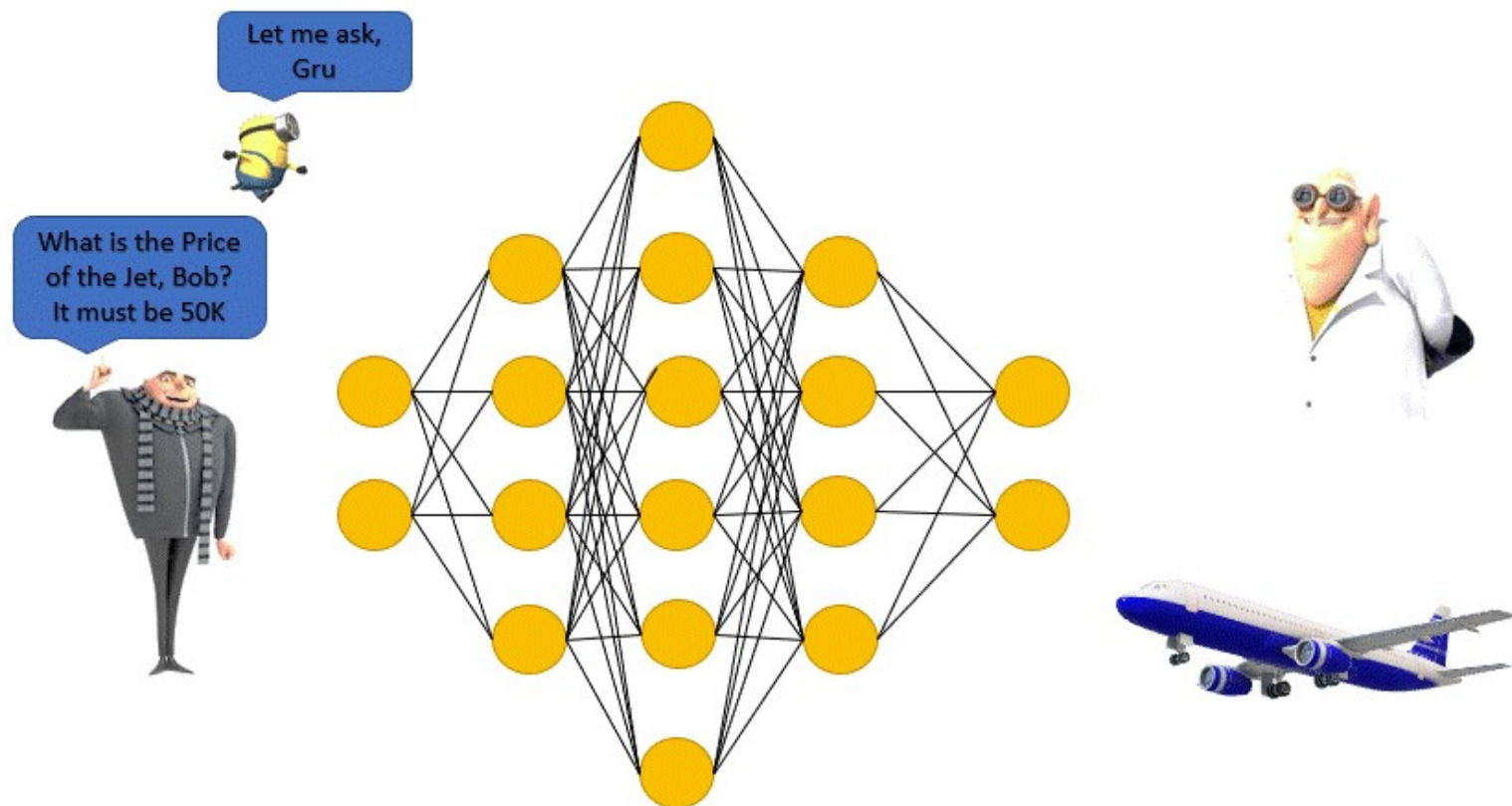

Introdução a Deep Learning



Introdução a Deep Learning

Redes Neurais Profundas:

Aprendizagem profunda é o termo usado para denotar o problema de treinar redes neurais artificiais que *realizam o aprendizado de características de forma hierárquica (uma camada depende dos resultados da camada anterior)*.



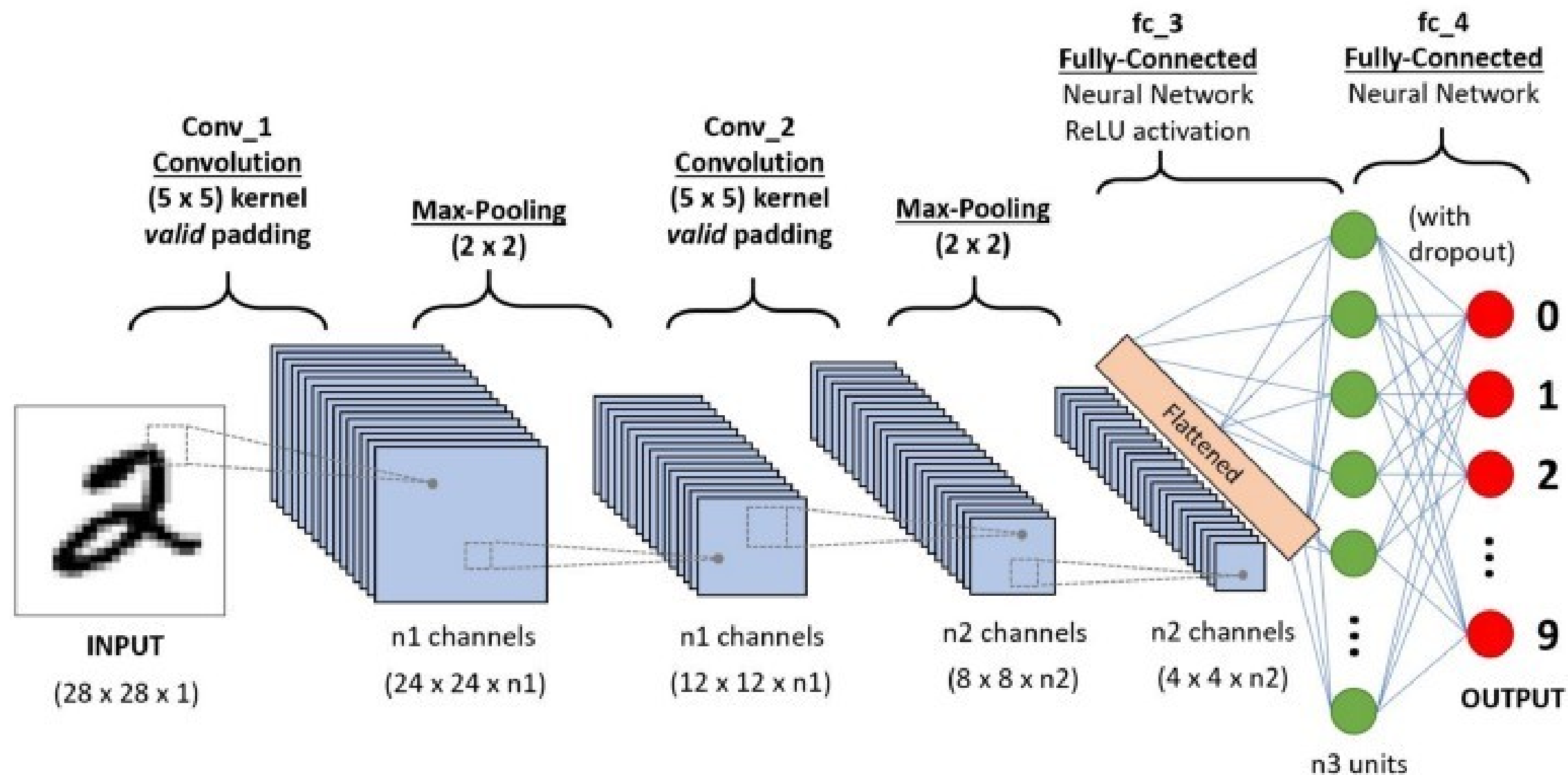
A informação é passada através de cada camada, com a saída da *camada anterior* fornecendo as entradas para a *próxima camada*.

Introdução a Deep Learning

Durante esse processo, as camadas aprendem os recursos ideais para o modelo, o que tem a *vantagem de que os recursos não precisam ser predeterminados*. No entanto, isso tem a *desvantagem de que as decisões do modelo não são explicáveis*. Como explicar as decisões pode ser importante, os pesquisadores estão desenvolvendo novas maneiras de entender a caixa preta do aprendizado profundo.

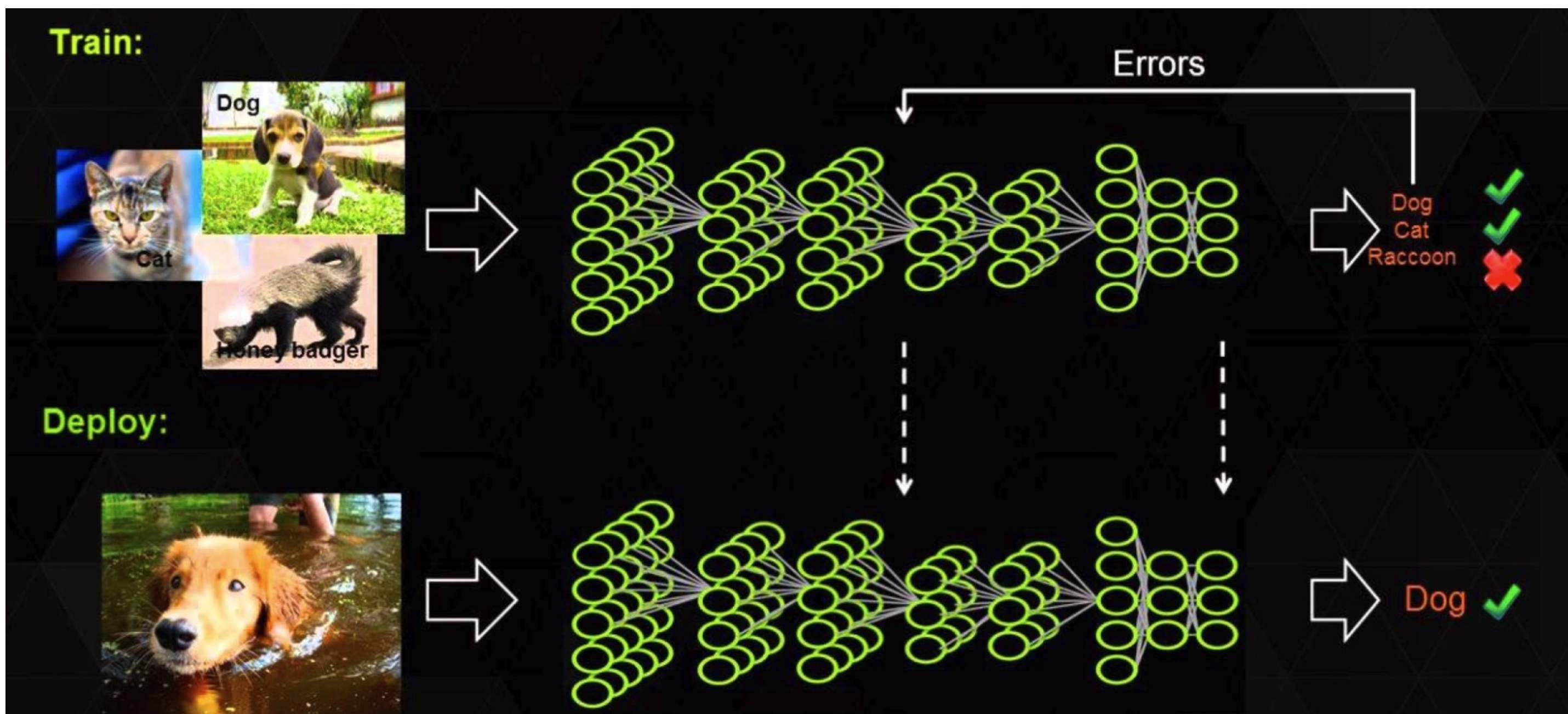
Introdução a Deep Learning

Todas as camadas entre as duas (entrada e saída) são referidas como *camadas ocultas*. Cada camada é tipicamente um algoritmo simples e uniforme contendo um tipo de função de ativação.



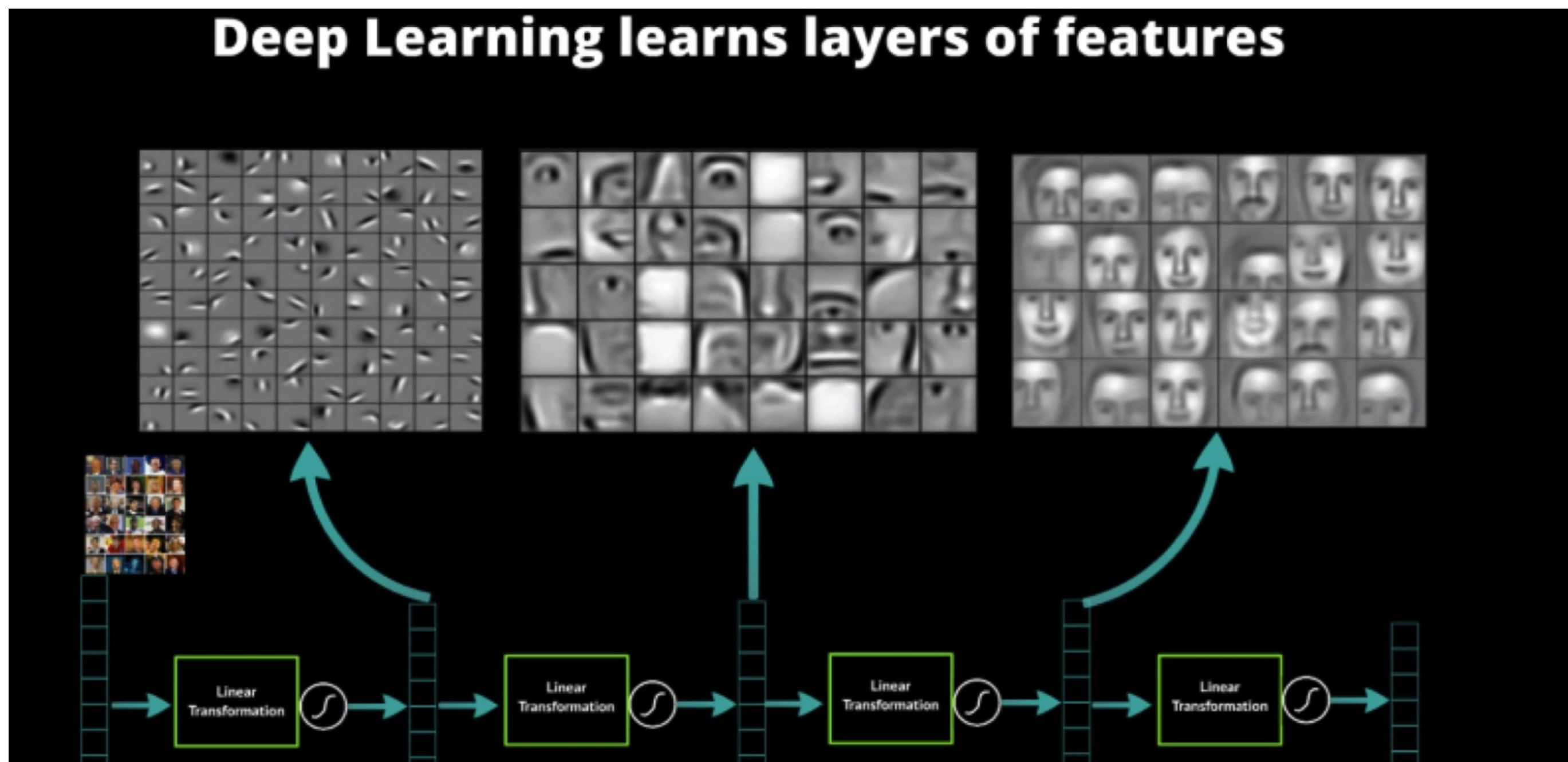
Introdução a Deep Learning

Podemos dizer que Deep Learning é uma arquitetura composta de vários níveis, compostos de unidades simples, todas sujeitas a treinamento.



Introdução a Deep Learning

Com múltiplos níveis de transformação não lineares a profundidade costuma variar de **5 a 20 níveis e o sistema de aprendizagem profunda**, pode aprender a implementar e interpretar, funções extremamente complexas.



Introdução a Deep Learning

As redes profundas se subdividem em duas categorias:



Convolutional
Neural Networks

Tratamento de imagem
e videos

Recurrent Neural
Networks

Tratamento da
linguagem falada e
escrita

Introdução a Deep Learning

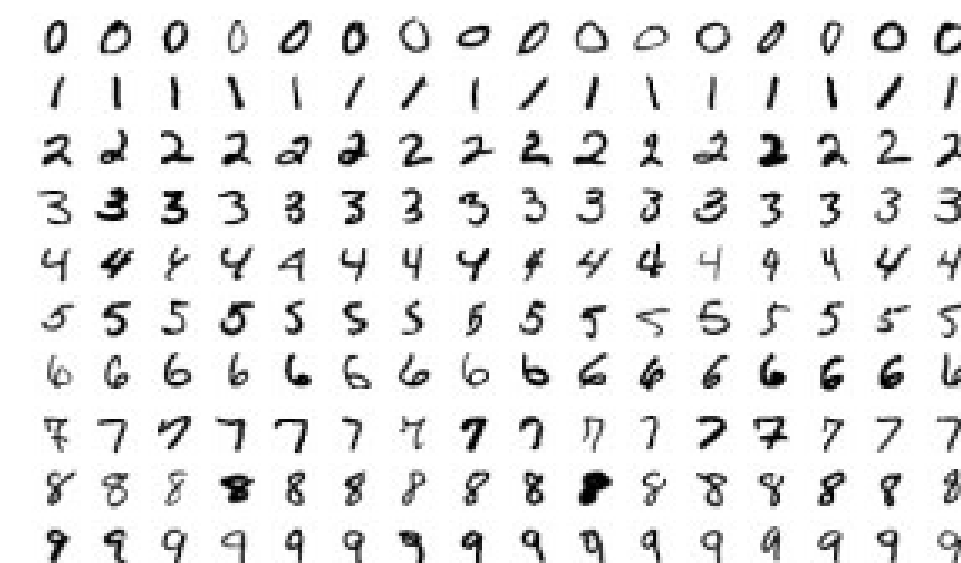
Redes Convolutivas

Novas arquiteturas foram propostas nos últimos anos como forma de melhoria da **LeNet-5** (*LeCun (1998) - MNIST*) .

As **CNNs** são formadas por sequências de camadas e cada uma destas possui uma função específica na propagação do sinal de entrada.

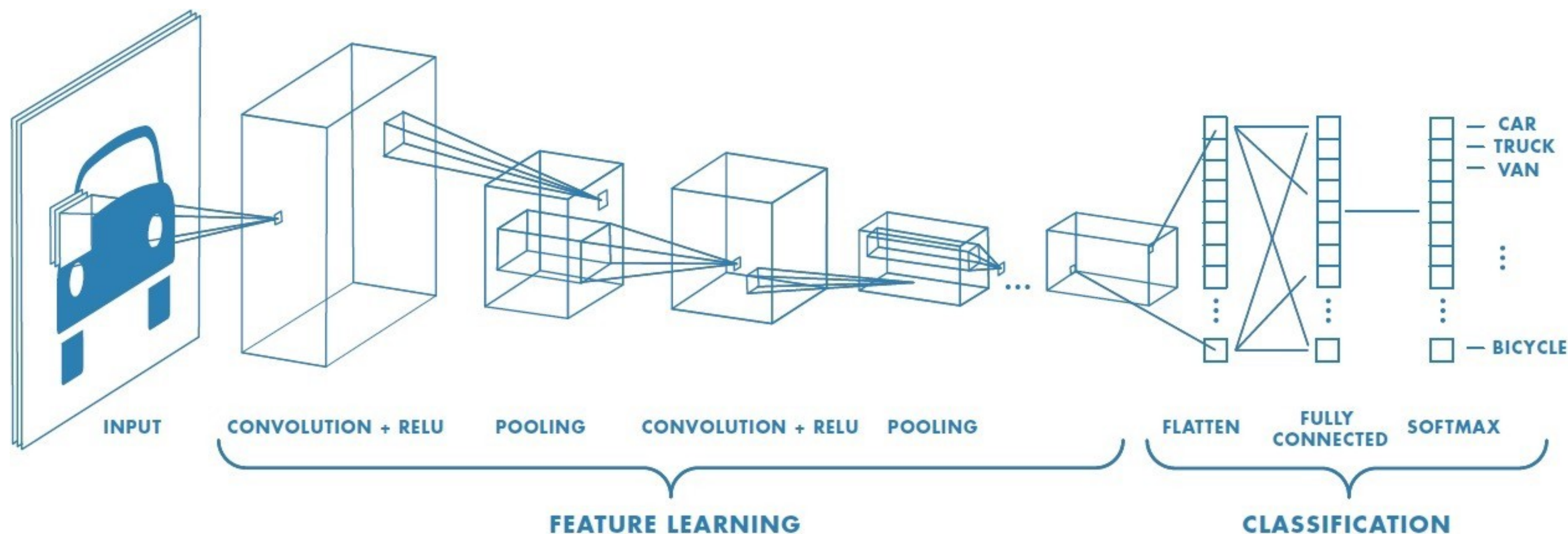
As CNNs tem sua arquitetura dividida em três camadas:

- **Convolutivas**
- **Pooling**
- **Totalmente Conectadas.**



Introdução a Deep Learning

Assim sendo a Deep Learning consiste em uma técnica de *Machine Learning* que utiliza-se de *redes neurais profundas* para o processamento das informações e a aprendizagem.



Introdução a Deep Learning

Redes Convolutivas

Camada **Convolutiva**:

As camadas convolucionais consistem de um conjunto de filtros (**ou um kernel de imagem que é uma pequena matriz usada para aplicar efeitos como os que você pode encontrar no Photoshop**) que recebem como entrada um arranjo 3D, também chamado de volume. Cada filtro possui dimensão reduzida, porém ele se estende por toda a profundidade do volume de entrada.

Por exemplo, se a imagem **for colorida**, então ela possui 3 canais e o filtro da primeira camada convolucional poderá ter tamanho 5x5x3 por exemplo (5 pixels de altura e largura, e profundidade igual a 3).

Introdução a Deep Learning

Redes Convolutivas

Camada **Convolutiva**:

Automaticamente, durante o processo de treinamento da rede, esses filtros são **ajustados para que sejam ativados em presença de características relevantes** identificadas no volume de entrada, tais como orientação de bordas ou manchas de cores.

A relevância é avaliada de tal forma que os resultados sejam otimizados em função de um conjunto de amostras previamente classificadas.

Introdução a Deep Learning

Redes Convolutivas

Camada **Convolutiva**:

Existem três parâmetros que controlam o tamanho do volume resultante da camada convolucional: depth (**profundidade**), stride (**passo – espaçamento entre as máscaras**) e padding (**preenchimento**).

Sendo que a profundidade do volume resultante é igual ao número de filtros utilizados.

Cada um desses filtros será responsável por extrair características diferentes no volume de entrada. Portanto, quanto maior o número de filtros maior o número de características extraídas, porém a complexidade computacional, relativa ao tempo e ao uso de memória, também será maior.

Introdução a Deep Learning

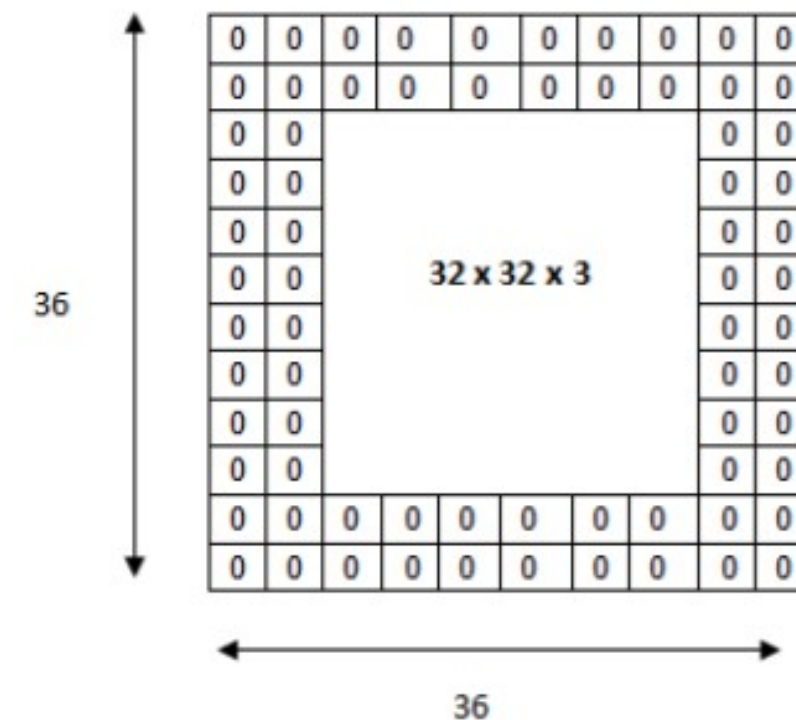
Redes Convolutivas

Camada **Convolutiva**:

Enquanto que a profundidade do volume resultante depende somente do número de filtros utilizados, a altura e largura do volume resultante dependem do **passo** e do **zero-padding**. O parâmetro passo especifica o tamanho do salto na operação de convolução.



Zero-Padding:
Tamanho da
borda a ser
preenchida com
zeros .

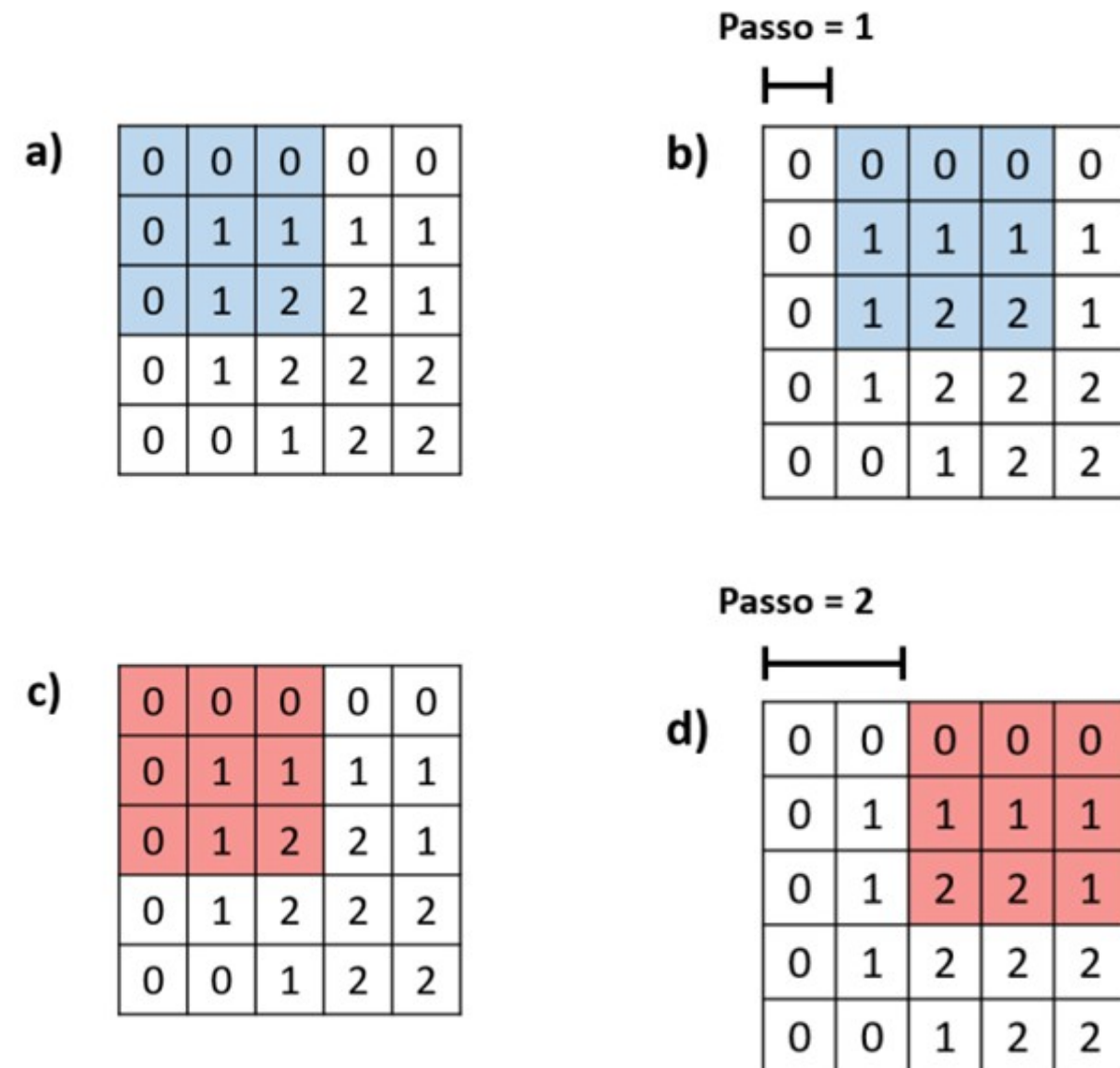


Introdução a Deep Learning

Redes Convolutivas

Camada *Convolutiva*:

Quando o passo é igual a 1, o filtro salta somente uma posição por vez. Quando o passo é igual a 2, o filtro salta duas posições por vez.



Quanto maior o valor do passo menor será a altura e largura do volume resultante, porém características importantes podem ser perdidas. Por esse motivo, é incomum se utilizar o valor de passo maior que 2.

Introdução a Deep Learning

Redes Convolutivas

Camada ***Convolutiva***:

Com isso, é possível computar a altura (AC) e a largura (LC) do volume resultante de uma camada convolucional.

$$AC = \frac{A - F + 2P}{S} + 1,$$

$$LC = \frac{L - F + 2P}{S} + 1,$$

Em que **A** e **L** correspondem, respectivamente, a altura e largura do volume de entrada, **F** é o tamanho dos filtros utilizados, **S** é o valor do passo, e **P** é o valor do *zero-padding*.

Introdução a Deep Learning

Redes Convolutivas

Camada **Convolutiva**:

$$AC = \frac{A - F + 2P}{S} + 1,$$

$$LC = \frac{L - F + 2P}{S} + 1,$$

Exemplos:

Com entrada 7x7 e filtro 3x3 com passo 1 e pad 0, a saída teria 5x5 neurônios por fatia.

Com $S=2$, teríamos saída com 3x3 neurônios por fatia.

Introdução a Deep Learning

Extração de característica

<https://setosa.io/ev/image-kernels/>

Introdução a Deep Learning

Redes Convolutivas

Camada *Convolutiva*:

A aplicação do operador de convolução é aplicado da seguinte forma.

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

x

1	0	0
1	0	1
0	1	1

Feature Detector

4	?			

Mapa de características

$$0*1 + 0*0 + 0*0 + 0*1 + 1*0 + 1*1 + 0*0 + 1*1 + 1*2 = 4$$

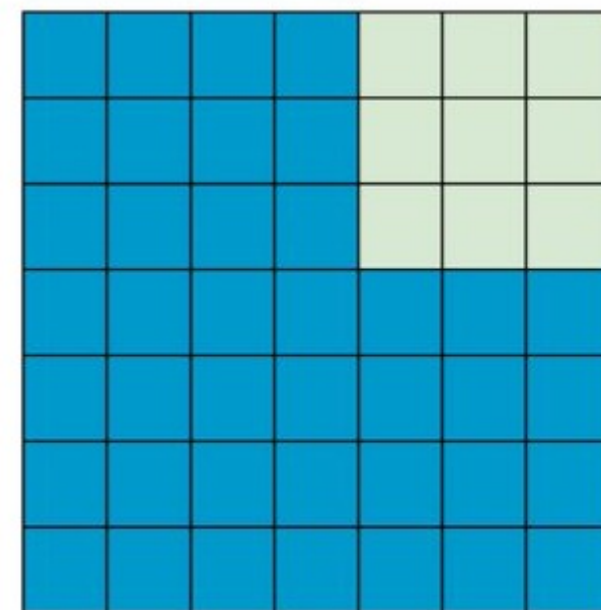
Introdução a Deep Learning

Redes Convolutivas

Camada **Convolutiva**:

Exemplos:

Com entrada 7x7 e filtro 3x3 com passo 1 e pad 0, a saída teria 5x5 neurônios por fatia.



Entrada 7x7

Filtro 3x3

Saída 5 x 5

Introdução a Deep Learning

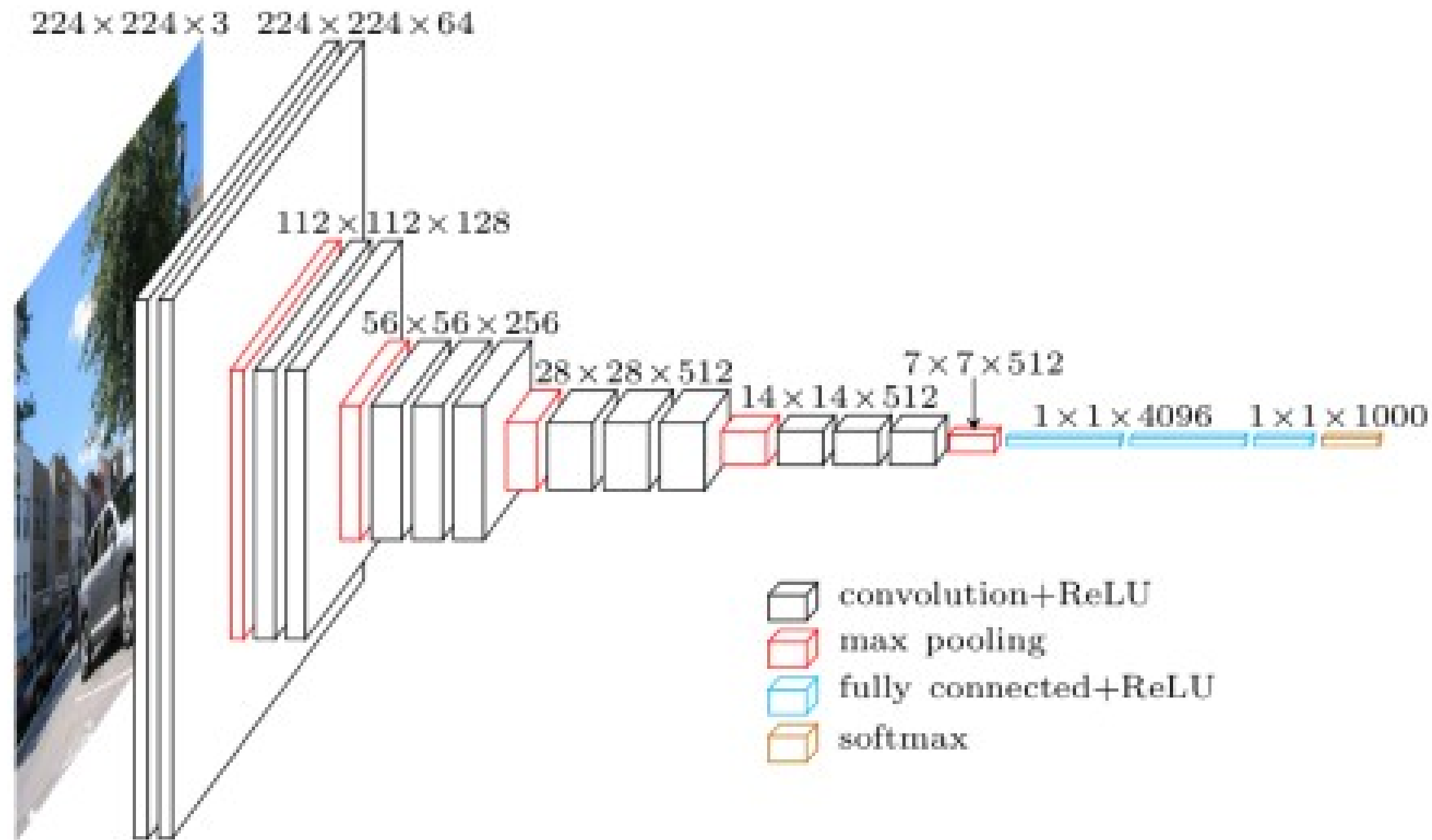
Redes Convolutivas

Camada ***Convolutiva***:

Ao aplicarmos o filtro o detector de característica preserva as características principais da imagem (boca, nariz, olho, etc).

Introdução a Deep Learning

Redes Convolutivas



Recapitulando:

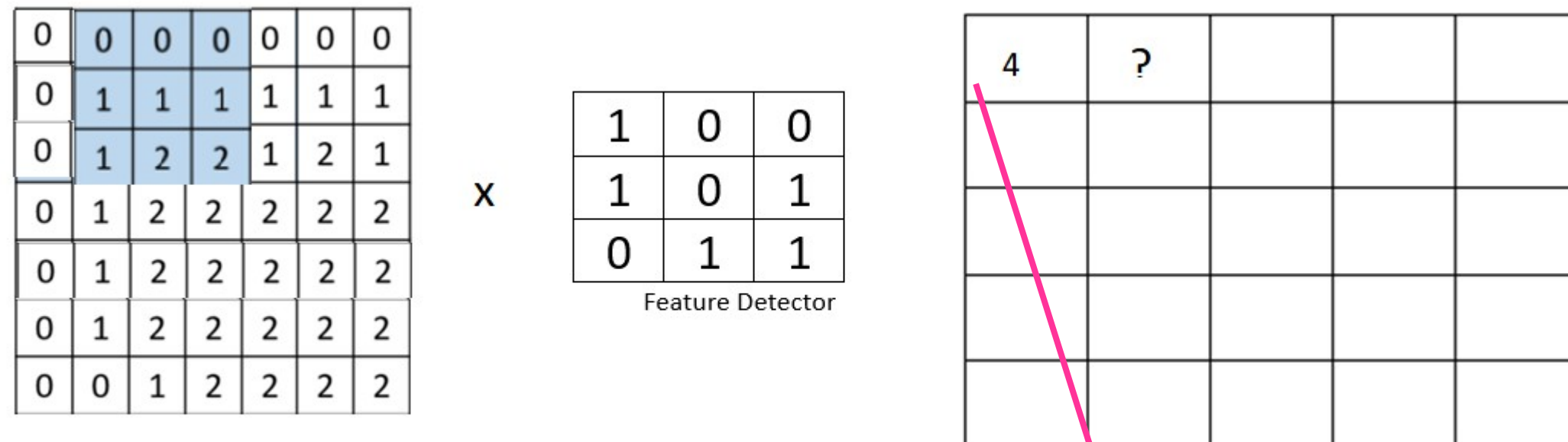
Estamos trabalhando com o input de uma imagem, essa imagem é uma matriz de pixels.

Introdução a Deep Learning

Redes Convolutivas

Camada *Convolutiva*:

A aplicação do operador de convolução é aplicado da seguinte forma.



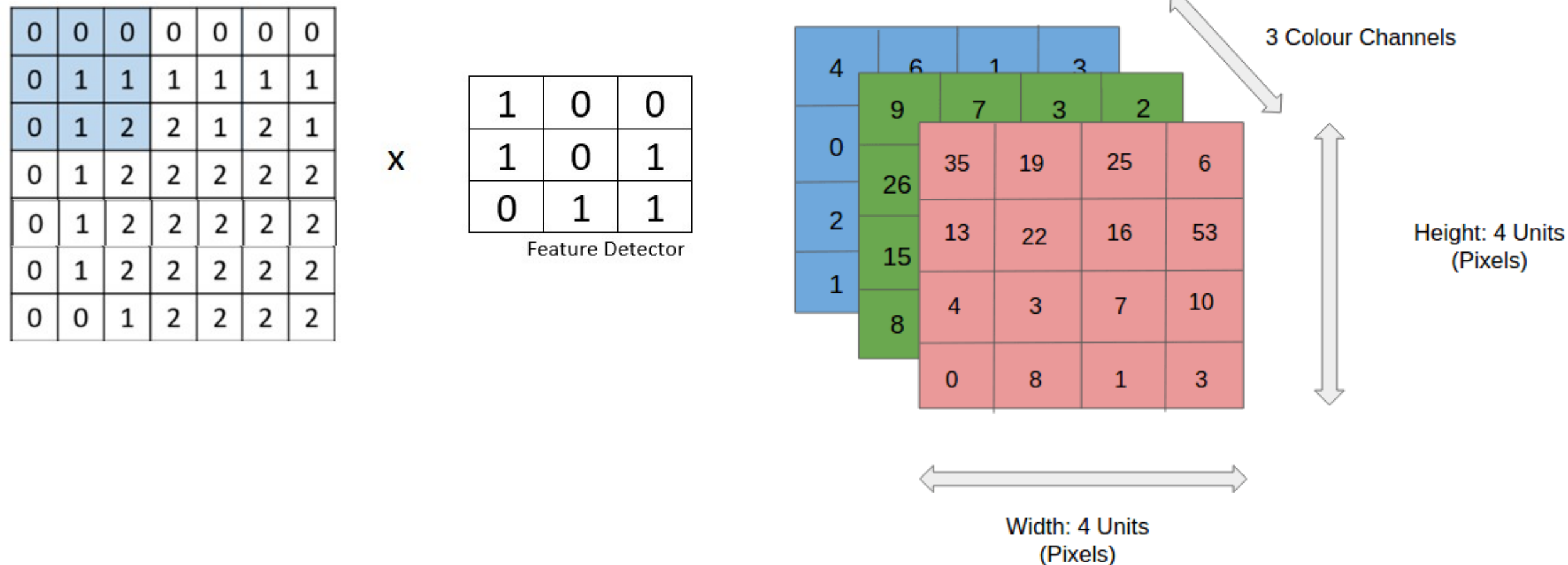
Após extração é aplicado a função Relu em cada um desses pixels.

Introdução a Deep Learning

Redes Convolutivas

Camada *Convolutiva*:

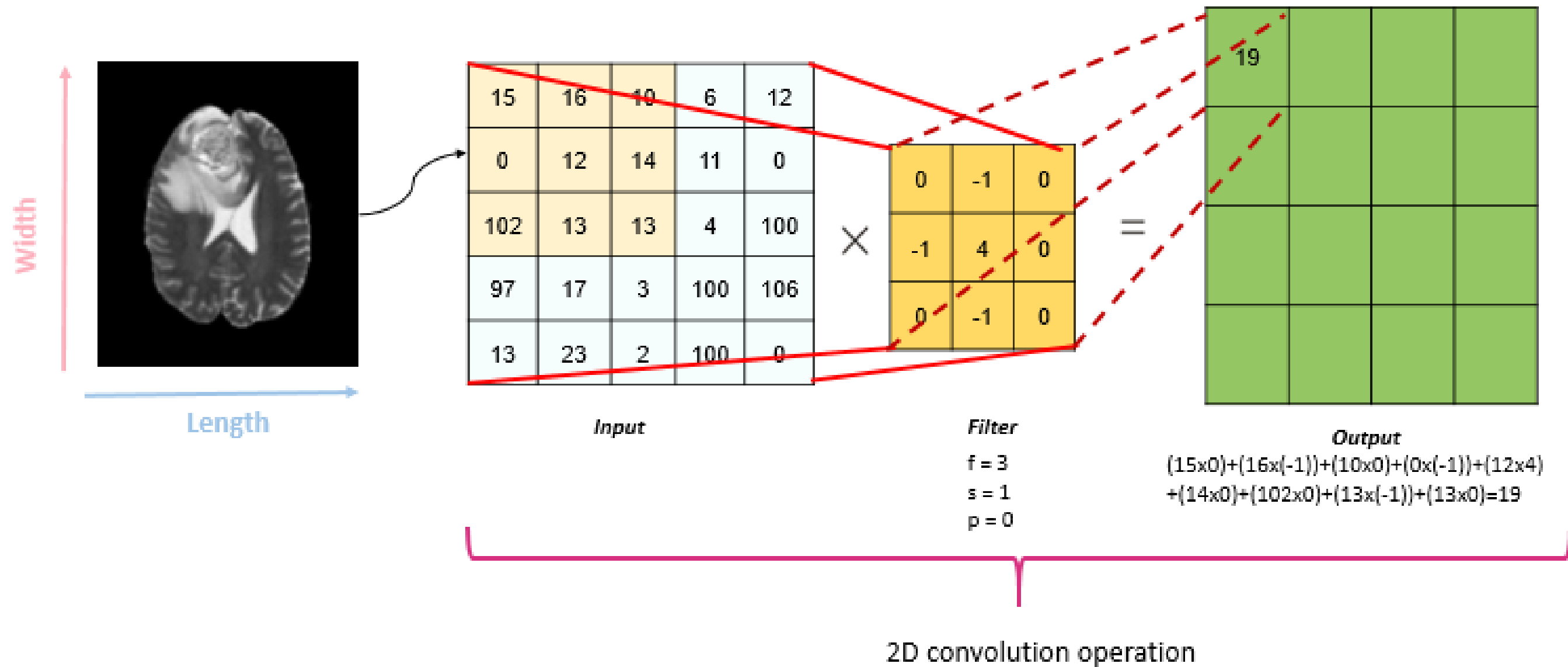
A aplicação do mapa de característica.



A própria rede se encarrega de aplicar vários filtros e escolher o que melhor se adapta (além de escolher os pesos, escolhe o filtro que melhor se adapta).

Introdução a Deep Learning

Redes Convolutivas

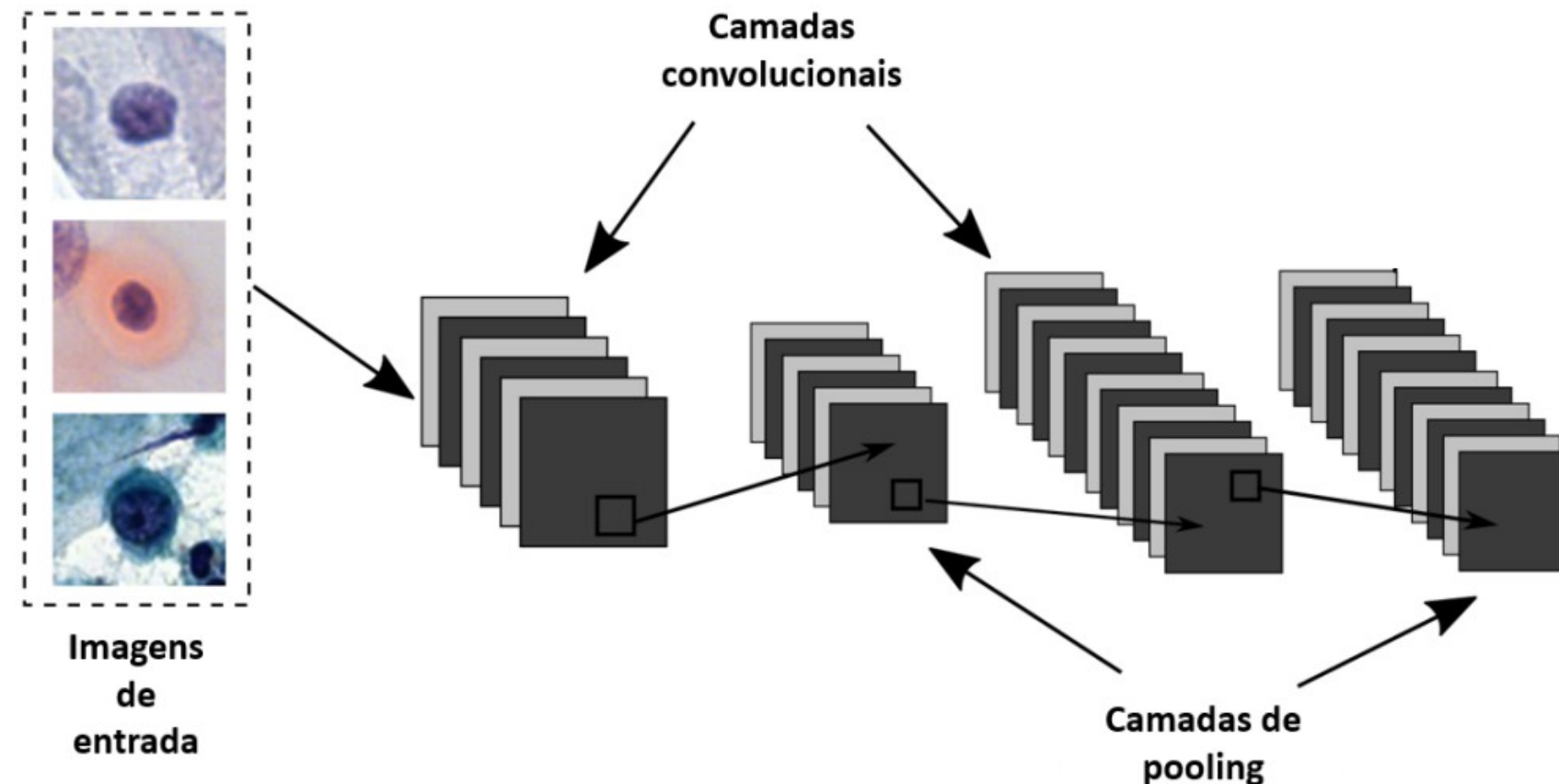


Introdução a Deep Learning

Redes Convolutivas

Camadas de **Pooling**

As camadas de *pooling* são responsáveis por reduzir a dimensionalidade do volume resultante após as camadas convolucionais e ajudam a tornar a representação invariante a pequenas translações na entrada

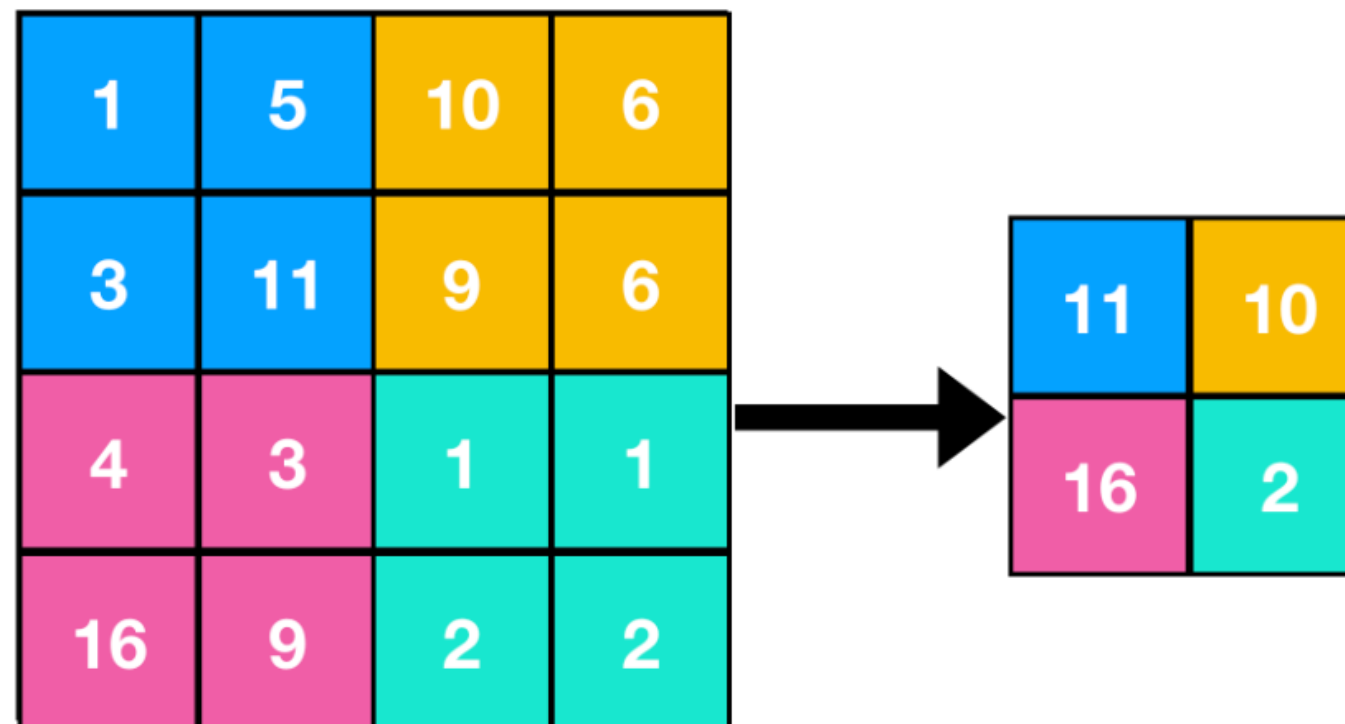


Introdução a Deep Learning

Redes Convolutivas

Camadas de **Pooling**

Na operação de *pooling*, os valores pertencentes a uma determinada região do mapa de atributos, gerados pelas camadas convolucionais, são substituídos por alguma métrica dessa região. A forma mais comum de *pooling* consiste em substituir os valores de uma região **pelo valor máximo**, como ilustra a Figura 1.2.



Introdução a Deep Learning

Redes Convolutivas

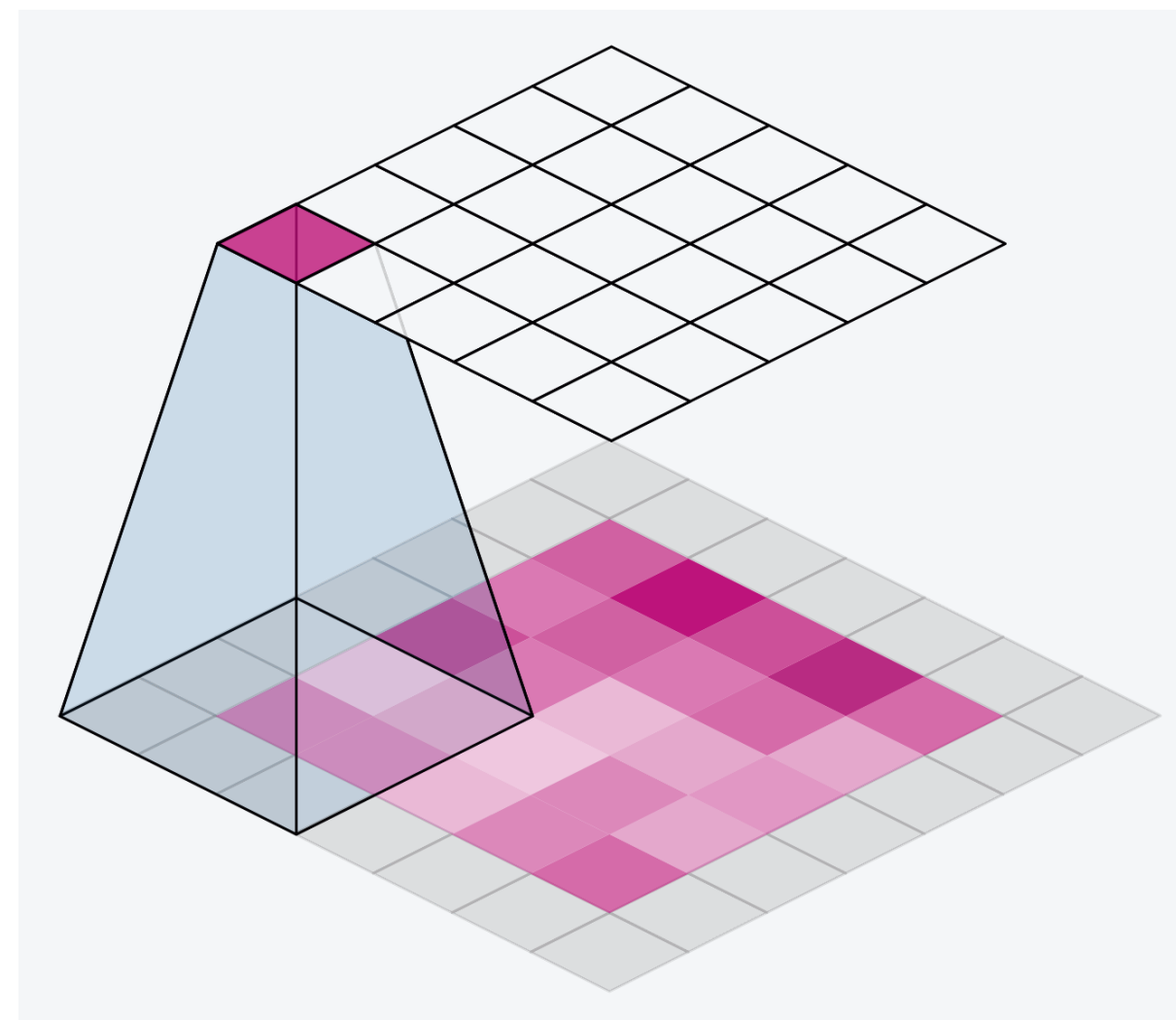
*Camadas de **Pooling***

Essa operação é conhecida como *max pooling* e é útil para eliminar valores desprezíveis, reduzindo a dimensão da representação dos dados e acelerando a computação necessária para as próximas camadas, além de criar uma invariância a pequenas mudanças e distorções locais.

Introdução a Deep Learning

Redes Convolutivas

*Camadas de **Pooling***



Aplicação de *max pooling* em uma imagem 4x4 utilizando um filtro 2x2. Além de reduzir o tamanho da imagem, consequentemente reduz o processamento para as próximas camadas, essa técnica também auxilia no tratamento de invariâncias locais.

Introdução a Deep Learning

Redes Convolutivas

*Camadas de **Pooling***

A altura (AP) e a largura (LP) do volume resultante após a operação de pooling podem ser calculados pelas Equações.

$$AP = \frac{A - F}{S} + 1$$

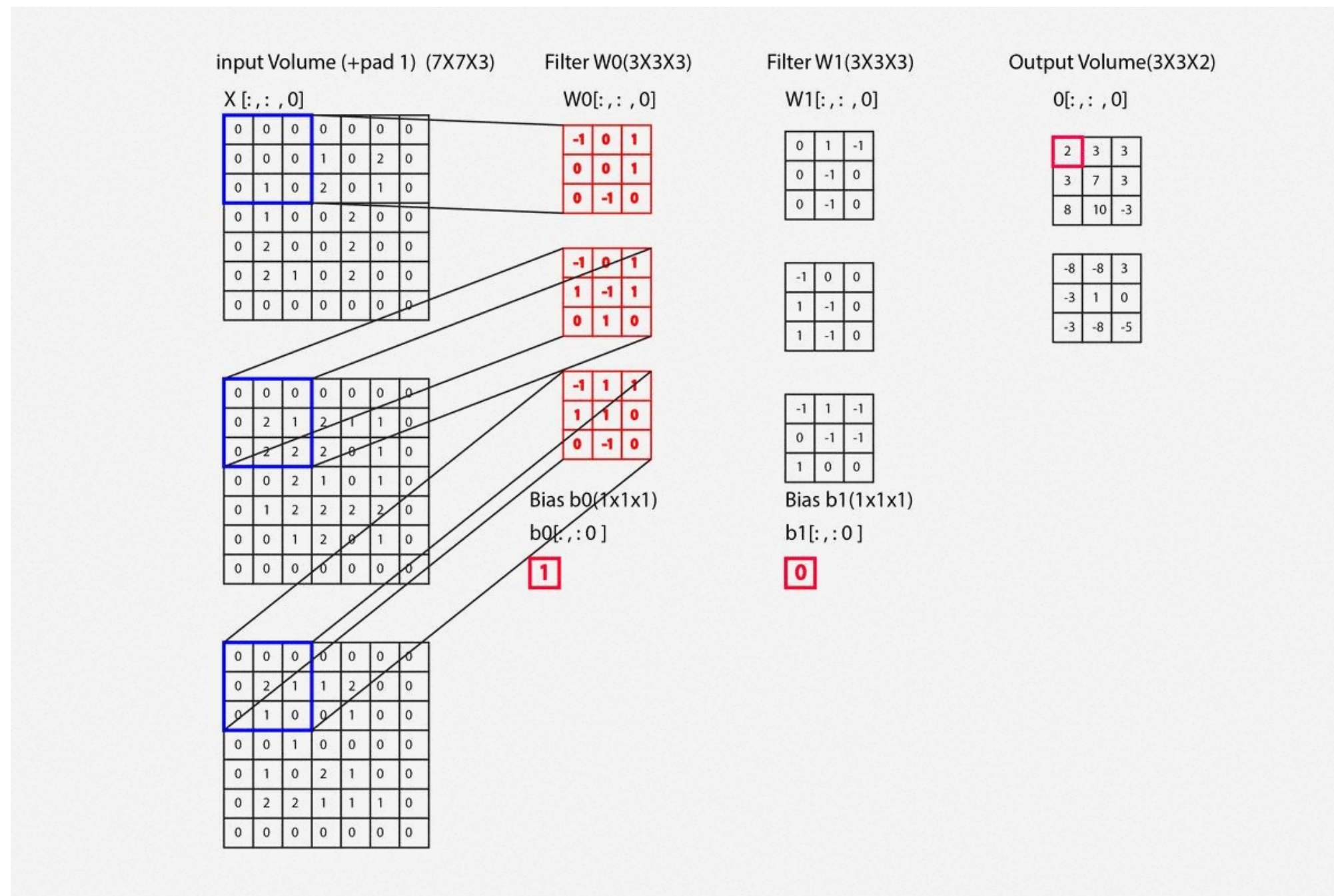
$$LP = \frac{L - F}{S} + 1$$

Onde as variáveis A e L correspondem respectivamente a altura e largura do volume de entrada, enquanto que F representa o tamanho da janela utilizada, e S é o valor do passo. **Vale destacar que a profundidade do volume de entrada não é alterada pela operação de *pooling*.**

Introdução a Deep Learning

Redes Convolutivas

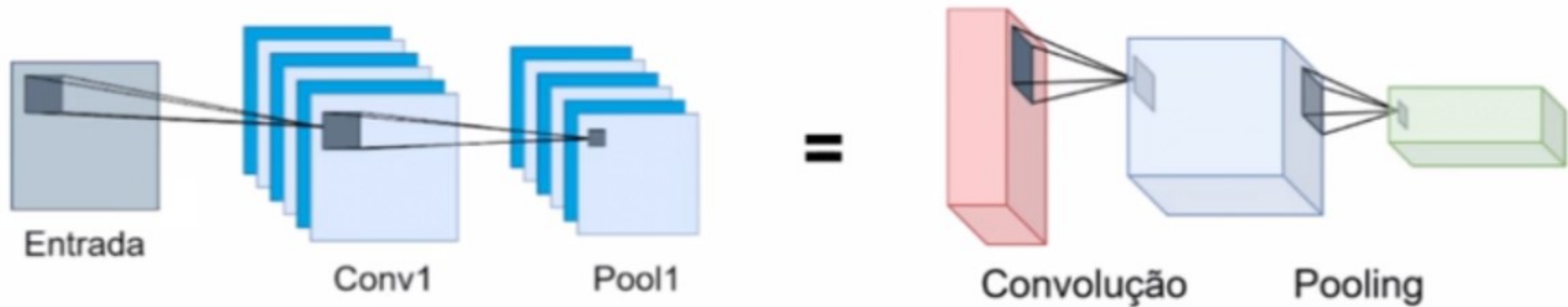
Camadas de *Pooling*



Introdução a Deep Learning

Redes Convolutivas

*Camadas de **Pooling***

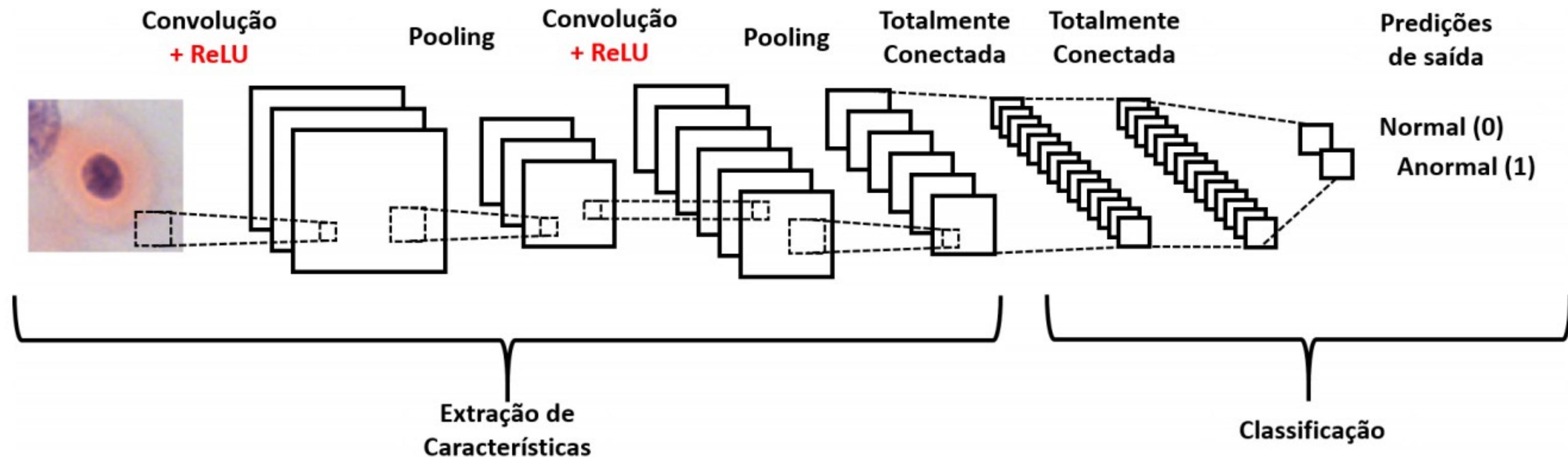


Introdução a Deep Learning

Redes Convolutivas

Camadas **Totalmente Conectadas**.

A saída das camadas convolucionais e de *pooling* representam as características extraídas da imagem de entrada. O objetivo das camadas **totalmente conectadas** é utilizar essas características para classificar a imagem em uma classe pré-determinada, como ilustrado abaixo.



Introdução a Deep Learning

Redes Convolutivas

Camadas Totalmente Conectadas.

Essas camadas são formadas por unidades de processamento conhecidas como neurônio, e o termo “***totalmente conectado***” significa que todos os neurônios da camada anterior estão conectados a todos os neurônios da camada seguinte.

Introdução a Deep Learning

Redes Convolutivas

*Camadas **Totalmente Conectadas.***

A última camada da rede utiliza ***softmax (por exemplo)*** como ***função de ativação***. Essa função recebe um vetor de valores como entrada e produz a ***distribuição probabilística*** da imagem de entrada pertencer a cada umas das classes na qual a rede foi treinada. Lembrando sempre que a ***soma de todas as probabilidades é igual a 1.***

Introdução a Deep Learning

Redes Convolutivas

*Camadas **Totalmente Conectadas**.*

Uma técnica conhecida como **dropout** também é bastante utilizada entre as camadas **totalmente conectadas** para reduzir o tempo de treinamento e evitar *overfitting*. Essa técnica consiste em remover, aleatoriamente a cada iteração de treinamento, uma determinada porcentagem dos neurônios de uma camada, readicionando-os na iteração seguinte. Essa técnica também confere à rede a habilidade de *aprender atributos mais robustos, uma vez que um neurônio não pode depender da presença específica de outros neurônios*.

O dropout é usado apenas na fase de treinamento, não no teste.

Introdução a Deep Learning

Redes Convolutivas

Treinamento da CNN

A forma mais comum de treinamento de uma CNN é por meio do algoritmo *backpropagation*. O processo de treinamento da CNN usando esse algoritmo ocorre da seguinte forma:

- **Passo 1:** Todos os filtros e pesos da rede são inicializados com valores aleatórios;
- **Passo 2:** A rede recebe uma imagem de treino como entrada e realiza o processo de propagação (convoluções, ReLU e pooling, e processo de propagação nas camadas totalmente conectadas). Após esse processo a rede obtém um valor de probabilidade para cada classe.

Introdução a Deep Learning

Redes Convolutivas

Treinamento da CNN

- **Passo 3:** É calculado o erro total obtido na camada de saída (somatório do erro de todas as classes): $\sum \frac{1}{2} (probabilidade\ real - probabilidade\ obtida)^2$

Introdução a Deep Learning

Redes Convolutivas

Treinamento da CNN

- **Passo 4:** O algoritmo Backpropagation é utilizado para calcular os valores dos gradientes do erro. Em seguida, a técnica do gradiente descendente é utilizada para ajustar os valores dos filtros e pesos na proporção que eles contribuíram para o erro total. *Devido ao ajuste realizado, o erro obtido pela rede é menor a cada vez que uma mesma imagem passa pela rede.* Essa redução no erro significa que a rede está aprendendo a classificar corretamente imagens devido ao ajuste nos valores dos filtros e pesos. Em geral, os parâmetros: número e tamanho dos filtros nas camadas convolucionais, quantidade de camadas, dentre outros, são fixados antes do passo 1 e eles não sofrem alterações durante o processo de treinamento.

Introdução a Deep Learning

Redes Convolutivas

Treinamento da CNN

- **Passo 5:** Repete os passos 2-4 para todas as imagens do conjunto de treinamento.

Introdução a Deep Learning

Redes Convolutivas

Treinamento da CNN

Uma “**época de treinamento**” é o nome dado ao processo que considera todas as imagens do conjunto de treinamento durante os passos 2-4. O processo de treinamento da rede é repetido por consecutivas épocas até que uma das seguintes condições de parada seja satisfeita: a média do erro obtido pela rede durante a época atual seja menor que um limiar (por exemplo, utilizando o erro médio quadrático) ou um número máximo de épocas seja atingido.