

# Exercício de Aprendizagem Profunda

Cristóvão Bartholo

140135081

Universidade de Brasília (UnB)

cristovao@live.com

**Resumo**—Trabalho realizado utilizando uma rede neural profunda sobre a base de dados CIFAR10

## I. INTRODUCTION

### A. Redes Neurais Profundas

Redes neurais profundas são redes que a partir de 2010 revolucionaram as aplicações de inteligência artificial conexio-nista. Ela consiste em várias camadas convolucionais treinadas para produzir mapas de características, tornando viável o re-conhecimento de padrões em imagens complexas. Ela faz isso por meio de múltiplas camadas que extraem progressivamente características de um nível mais alto de uma entrada.

O treinamento depende de uma ampla base de dados para que a rede desenvolva uma boa capacidade de generalização. Seu treinamento é consideravelmente mais intensivo e demorado do que outros métodos.

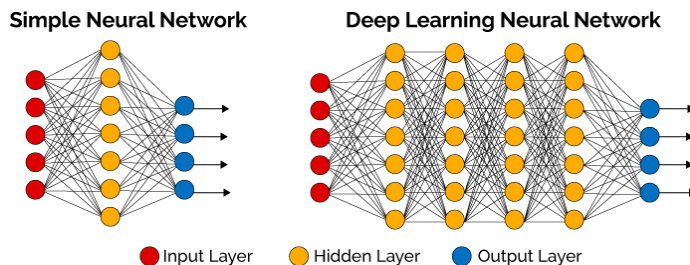


Figura 1. rede "rasa" vs rede profunda

### B. CIFAR 10

Neste trabalho será utilizada a base de imagens CIFAR10 (Canadian Institute For Advanced Research), que é uma base de dados amplamente usada para estudos e pesquisas na área de aprendizado de máquina e visão computacional. Utilizaremos um conjunto de 50.000 imagens de treinamento, 5.000 imagens de teste e 5.000 de validação. As imagens são de 32x32 pixels e correspondem a 10 classes diferentes como aviões, cães, gatos, etc.

## II. METODOLOGIA

Este trabalho consistiu em desenvolver três redes neuronais profundas com parâmetros variados para que se pudesse obter uma performance melhor da rede. Para desenvolvimento desse estudo foi utilizado o código base no endereço github: [/acht7111020/CNN\\_object\\_classification/tree/master/train](https://github.com/acht7111020/CNN_object_classification/tree/master/train). Os códigos foram executados na plataforma Google Colaboratory,

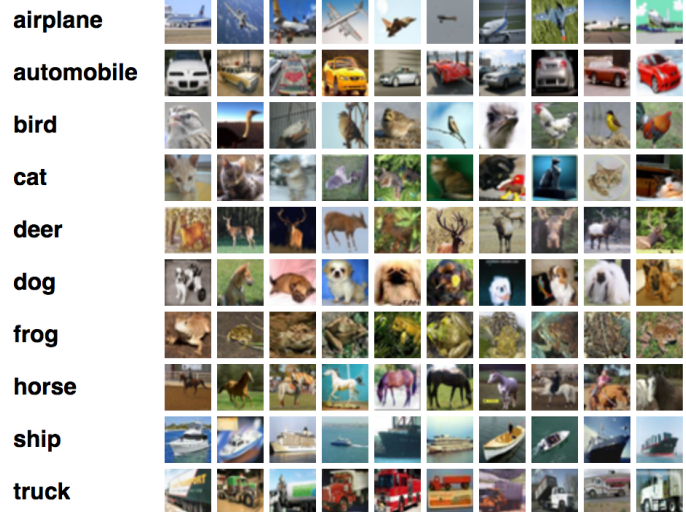


Figura 2. classes e exemplos da base CIFAR10.

já que não necessita de configuração e instalações de bibliotecas, acesso gratuito a GPUs e de fácil compartilhamento.

Para desenvolvimento da primeira rede neural foi escolhido os seguintes parâmetros: uma camada de convolução 2D, seguida de uma MaxPool2D, seguida então de outra camada de convolução 2D e outra de MaxPool2D, que é então "achatada" e seguida de uma camada densa com as dez categorias, com ativação softmax. As camadas de convolução tem função de ativação do tipo relu. O modelo pode ser visualizado na imagem 3.

Para esta rede foi observado que o número de épocas mais adequado foi de 15, pois a partir dessa quantidade os resultados não tiveram melhora significativa, e algumas vezes até pioraram. A taxa de aprendizagem selecionada foi de 0.0002 e o tamanho do batch foi de 50. No total o treinamento levou cerca de 45 segundos.

A segunda rede desenvolvida é um pouco mais complexa, apresenta a arquitetura pode ser visualizada na imagem 4. Para esta rede também foi utilizada uma taxa de aprendizagem, de 0.0002, 15 épocas e um batch de tamanho 50. O treinamento levou cerca de 1 minuto.

A terceira e última rede apresenta mais camadas e mais parâmetros. Seu modelo está explicitado na figura 5.

Nessa rede, utilizou-se uma taxa de aprendizado de 0.001,

Model: "sequential\_1"

| Layer (type)                   | Output Shape       | Param # |
|--------------------------------|--------------------|---------|
| conv2d_2 (Conv2D)              | (None, 30, 30, 32) | 896     |
| max_pooling2d_2 (MaxPooling2D) | (None, 15, 15, 32) | 0       |
| conv2d_3 (Conv2D)              | (None, 13, 13, 64) | 18496   |
| max_pooling2d_3 (MaxPooling2D) | (None, 6, 6, 64)   | 0       |
| flatten_1 (Flatten)            | (None, 2304)       | 0       |
| dense_1 (Dense)                | (None, 10)         | 23050   |
| Total params: 42,442           |                    |         |
| Trainable params: 42,442       |                    |         |
| Non-trainable params: 0        |                    |         |

Figura 3. arquitetura da primeira rede.

Model: "sequential\_1"

| Layer (type)                   | Output Shape        | Param # |
|--------------------------------|---------------------|---------|
| conv2d_4 (Conv2D)              | (None, 30, 30, 32)  | 896     |
| max_pooling2d_3 (MaxPooling2D) | (None, 15, 15, 32)  | 0       |
| conv2d_5 (Conv2D)              | (None, 13, 13, 128) | 36992   |
| max_pooling2d_4 (MaxPooling2D) | (None, 6, 6, 128)   | 0       |
| conv2d_6 (Conv2D)              | (None, 4, 4, 256)   | 295168  |
| max_pooling2d_5 (MaxPooling2D) | (None, 2, 2, 256)   | 0       |
| flatten (Flatten)              | (None, 1024)        | 0       |
| dense (Dense)                  | (None, 10)          | 10250   |
| Total params: 343,306          |                     |         |
| Trainable params: 343,306      |                     |         |
| Non-trainable params: 0        |                     |         |

Figura 4. arquitetura da segunda rede

que é um valor consideravelmente maior, mas que para este exercício foi escolhido de modo a agilizar o processo de treinamento sem se preocupar tanto com o desempenho preciso da rede após o treinamento.

Foi observado empiricamente que o tamanho do batch de 100 foi o mais eficiente dentre os testados.

Além disso, foi observado que a rede necessitava de em torno de 60 épocas para que seu valor convergisse. Essa rede levou cerca de 10 minutos para ser treinada.

### III. RESULTADOS

O treinamento do primeiro modelo pode ser visto na imagem 6. Essa rede obteve acurácia de 64%, 62%, e 65% para o conjunto de validação, teste e treinamento, respectivamente. A matriz de confusão para a imagem de um sapo (que corresponde à primeira imagem do cifar10) pode ser vista na imagem 7.

| Layer (type)                    | Output Shape        | Param # |
|---------------------------------|---------------------|---------|
| conv2d_114 (Conv2D)             | (None, 30, 30, 32)  | 896     |
| dropout_97 (Dropout)            | (None, 30, 30, 32)  | 0       |
| conv2d_115 (Conv2D)             | (None, 28, 28, 32)  | 9248    |
| dropout_98 (Dropout)            | (None, 28, 28, 32)  | 0       |
| conv2d_116 (Conv2D)             | (None, 26, 26, 32)  | 9248    |
| dropout_99 (Dropout)            | (None, 26, 26, 32)  | 0       |
| conv2d_117 (Conv2D)             | (None, 24, 24, 32)  | 9248    |
| dropout_100 (Dropout)           | (None, 24, 24, 32)  | 0       |
| conv2d_118 (Conv2D)             | (None, 22, 22, 256) | 73984   |
| max_pooling2d_15 (MaxPooling2D) | (None, 11, 11, 256) | 0       |
| conv2d_119 (Conv2D)             | (None, 9, 9, 64)    | 147520  |
| dropout_101 (Dropout)           | (None, 9, 9, 64)    | 0       |
| conv2d_120 (Conv2D)             | (None, 7, 7, 64)    | 36928   |
| dropout_102 (Dropout)           | (None, 7, 7, 64)    | 0       |
| conv2d_121 (Conv2D)             | (None, 5, 5, 256)   | 147712  |
| dropout_103 (Dropout)           | (None, 5, 5, 256)   | 0       |
| flatten_12 (Flatten)            | (None, 6400)        | 0       |
| dense_27 (Dense)                | (None, 32)          | 204832  |
| dense_28 (Dense)                | (None, 10)          | 330     |
| Total params: 639,946           |                     |         |
| Trainable params: 639,946       |                     |         |
| Non-trainable params: 0         |                     |         |

Figura 5. arquitetura da terceira rede

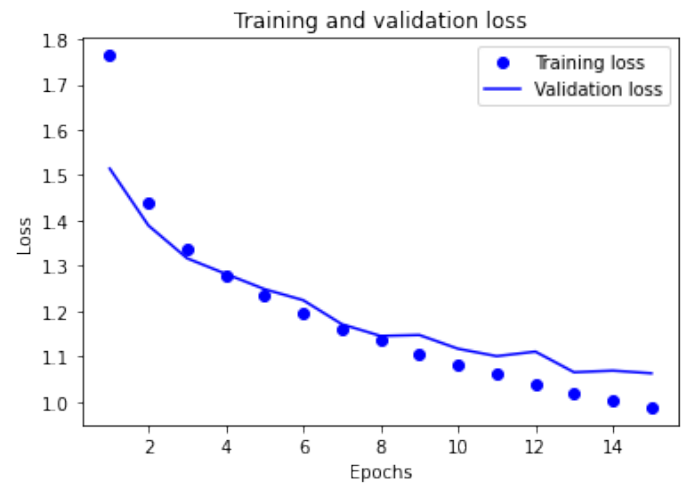


Figura 6. treinamento da primeira rede.

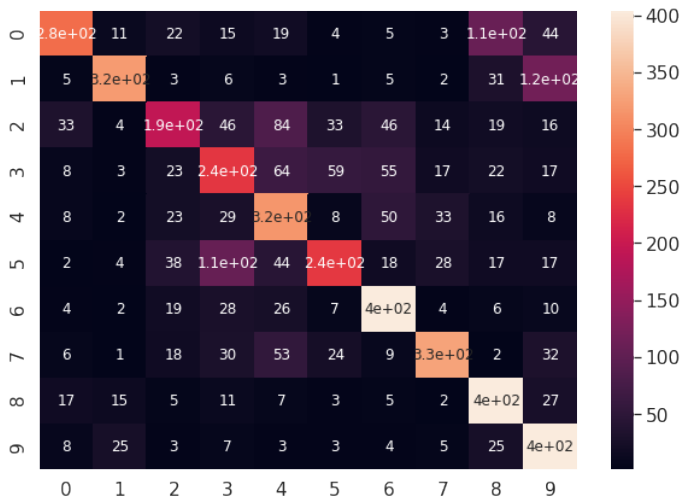


Figura 7. matriz de confusão da primeira rede.

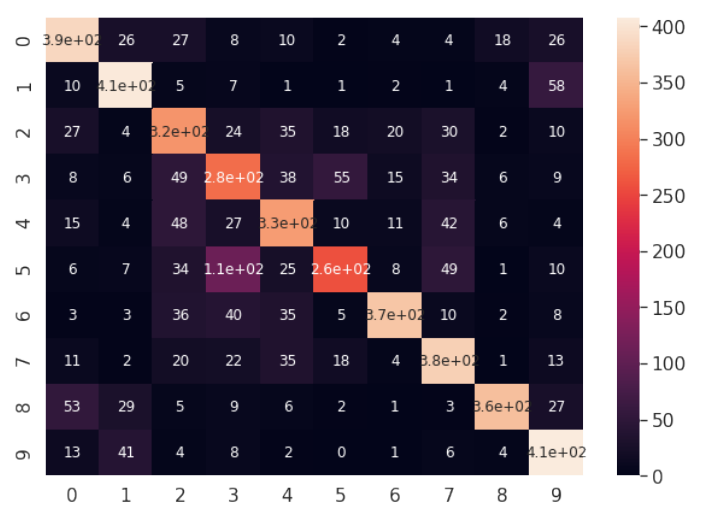


Figura 9. matriz de confusão da segunda rede.

O treinamento do segundo modelo pode ser visto na imagem 8. Nessa rede a acurácia obtida foi de 71%, 69% e 77% para os conjuntos de validação, teste e treinamento, respectivamente. A matriz de confusão para esta rede pode ser visualizada na imagem 9.

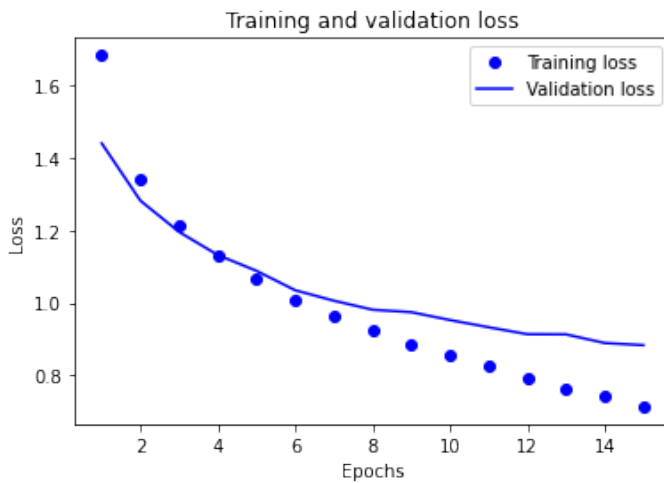


Figura 8. treinamento da segunda rede.

A terceira rede teve seu treinamento concluído em cerca de 10 minutos, a figura 10 mostra o treinamento da rede. ao final do treinamento essa rede obteve uma acurácia de 77%, 77% e 88% para os conjuntos de validação, teste e treinamento respectivamente. A matriz de confusão para a primeira imagem do dataset pode ser vista na figura 11.

Além disso, foram observados os mapas de características. Alguns mapas de características foram selecionados e apresentados aqui. O primeiro mapa apresentado na figura 12 mostra alguns mapas da primeira camada da última rede apresentada. A imagem utilizada na geração dos mapas corresponde a um sapo, que é a primeira imagem do CIFAR10.

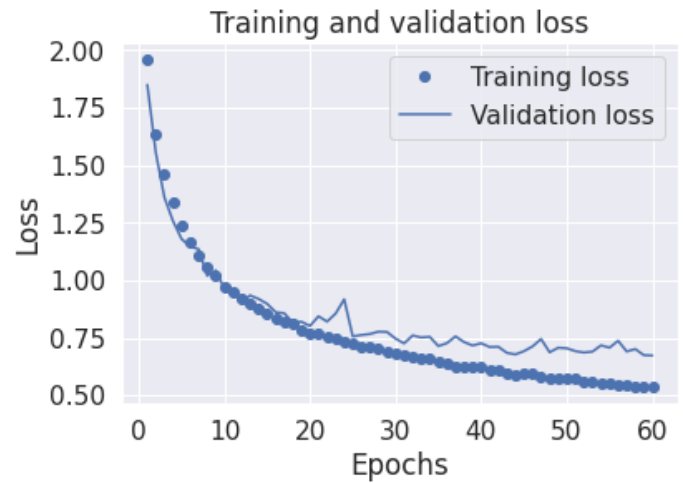


Figura 10. treinamento da terceira rede.

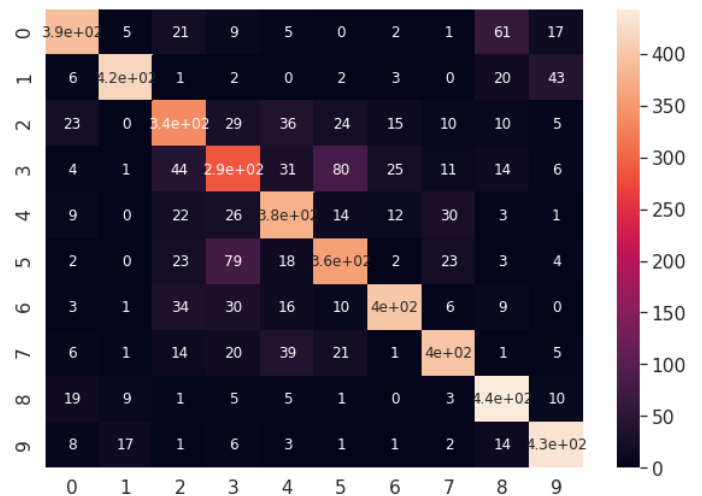


Figura 11. matriz de confusão da terceira rede.

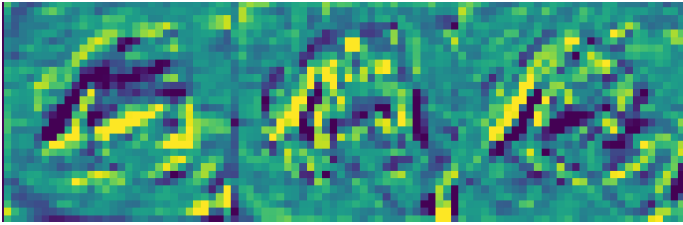


Figura 12. mapa de características da primeira camada.

Observamos também uma camada intermediária, correspondente à sétima camada da rede, ela está apresentada na figura 13.

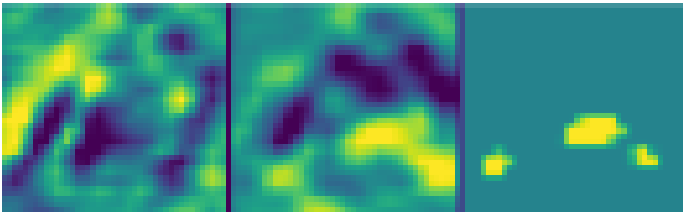


Figura 13. mapa de características da camada.

Por último tem-se parte do mapa característico da última camada convolucional. Ele está apresentado na figura 14

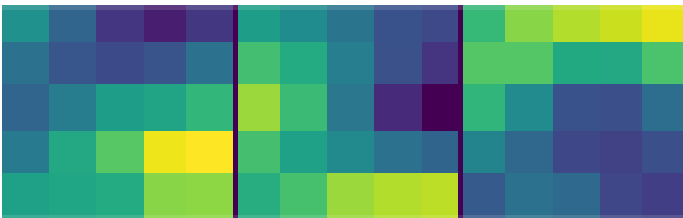


Figura 14. mapa de características da última camada convolucional.

#### IV. CONCLUSÕES

Pode-se concluir que com os parâmetros devidamente selecionados é possível realizar uma rede e treiná-la de modo que tenha uma boa acurácia utilizando o método de aprendizagem profunda. A rede aqui apresentada não é o estado da arte, porém é possível obter uma precisão bem maior com a adequação dos parâmetros utilizando um tempo maior de treinamento, uma quantidade de neurônios maior e uma taxa de aprendizagem mais adequada.

Pode-se observar nas imagens de mapas que a rede funciona tirando sucessivamente informações de cada vez mais alto nível da imagem. Na primeira camada o nível é mais baixo e o próprio olho humano consegue identificar a figura de um sapo na imagem. Já na imagem intermediária, características de mais alto nível podem ser observadas, como pernas barriga e papo. Na última camada convolucional a imagem já se encontra altamente abstraída e fica difícil determinar quais características a rede está procurando. A partir das matrizes de confusão é possível perceber que última rede apresentou

uma "certeza" maior do que as demais sobre as classes que estavam sendo estimadas, já que a diagonal principal está mais iluminada, o que mostra que houve menos "confu".

Portanto, pode-se concluir também que a rede apresentada apresentou um resultado satisfatório que para problemas semelhantes a relativamente pouco tempo não se era possível obter essa precisão.