Instituto Tecnológico de Aeronáutica - ITA
Inteligência Artificial para Robótica Móvel - CT-213
Aluno: Rafael Mello Celente

Relatório do Laboratório 8 - Redes Neurais Convolucionais

1. Breve Explicação em Alto Nível da Implementação

A implementação do modelo foi feita utilizando o framework do Keras, o que facilitou extraordinariamente o processo. Foi necessário apenas definir o modelo *Sequential* e adicionar as camadas assim como definidas na arquitetura especificada da LeNet-5.

Em resumo, a implementação do modelo foi a seguinte:

```
def make_lenet5():
    model = Sequential()

    model.add(layers.Conv2D(filters=6, kernel_size=(5, 5), strides=(1, 1), activation=activations.tanh, input_shape=(32, 32, 1)))
    model.add(layers.AveragePooling2D(pool_size=(2,2), strides=(2, 2)))
    model.add(layers.Conv2D(filters=16, kernel_size=(5,5), strides=(1,1), activation=activations.tanh))
    model.add(layers.AveragePooling2D(pool_size=(2,2), strides=(2,2)))
    model.add(layers.Conv2D(filters=120, kernel_size=(5,5), strides=(1,1), activation=activations.tanh))
    model.add(layers.Flatten())
    model.add(layers.Dense(units=84, activation=activations.tanh))
    model.add(layers.Dense(units = 10, activation='softmax'))

    return model
```

2. Figuras Comprovando Funcionamento do Código

2.1. Evolução do Treinamento no TensorBoard

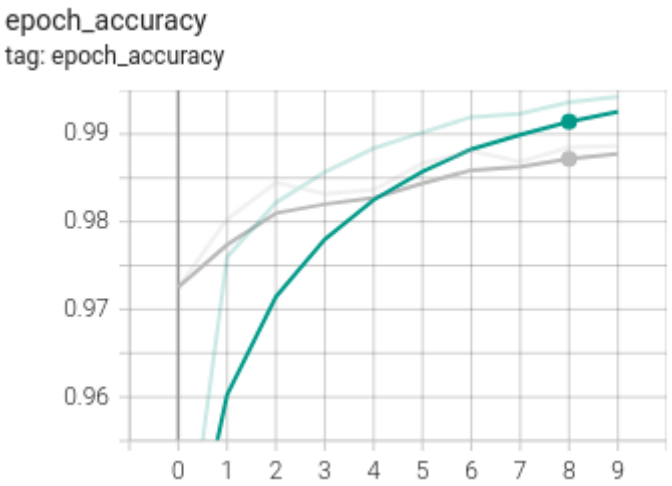

Figura 1. Acurácia do treinamento (verde) e acurácia da validação (cinza) em função da *epoch*.

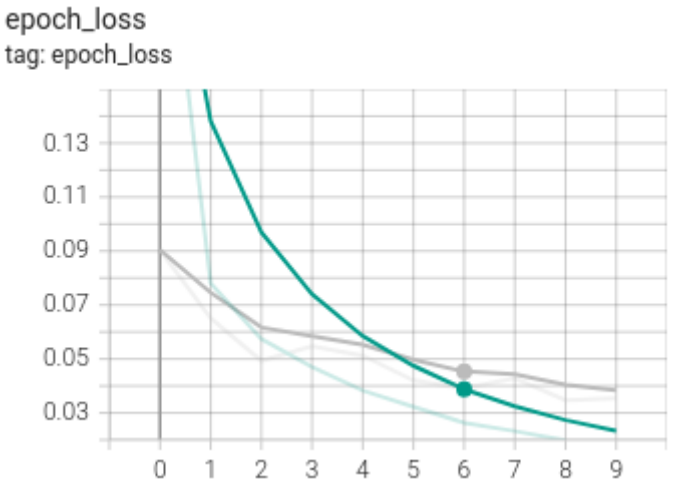

Figura 2. *Loss* do treinamento (verde) e *loss* da validação (cinza) em função da *epoch.*

2.2. Avaliação da LeNet-5

```
_________________________________________________________________
Epoch 1/10
375/375 [==============================] - 27s 72ms/step - loss: 0.2382 - accuracy: 0.9341 - val_loss: 0.0903 - val_accuracy: 0.9725
Epoch 2/10
375/375 [==============================] - 26s 70ms/step - loss: 0.0780 - accuracy: 0.9760 - val_loss: 0.0651 - val_accuracy: 0.9803
Epoch 3/10
375/375 [==============================] - 26s 70ms/step - loss: 0.0573 - accuracy: 0.9822 - val_loss: 0.0493 - val_accuracy: 0.9845
Epoch 4/10
375/375 [==============================] - 27s 72ms/step - loss: 0.0471 - accuracy: 0.9857 - val_loss: 0.0547 - val_accuracy: 0.9832
Epoch 5/10
375/375 [==============================] - 26s 68ms/step - loss: 0.0383 - accuracy: 0.9884 - val_loss: 0.0513 - val_accuracy: 0.9836
Epoch 6/10
375/375 [==============================] - 26s 69ms/step - loss: 0.0323 - accuracy: 0.9902 - val_loss: 0.0421 - val_accuracy: 0.9866
Epoch 7/10
375/375 [==============================] - 26s 68ms/step - loss: 0.0263 - accuracy: 0.9919 - val_loss: 0.0391 - val_accuracy: 0.9881
Epoch 8/10
375/375 [==============================] - 26s 68ms/step - loss: 0.0232 - accuracy: 0.9923 - val_loss: 0.0427 - val_accuracy: 0.9868
Epoch 9/10
375/375 [==============================] - 26s 68ms/step - loss: 0.0198 - accuracy: 0.9936 - val_loss: 0.0348 - val_accuracy: 0.9885
Epoch 10/10
375/375 [==============================] - 26s 69ms/step - loss: 0.0175 - accuracy: 0.9942 - val_loss: 0.0354 - val_accuracy: 0.9887
```

Figura 3. Resultados numéricos do treinamento e validação do modelo por época.

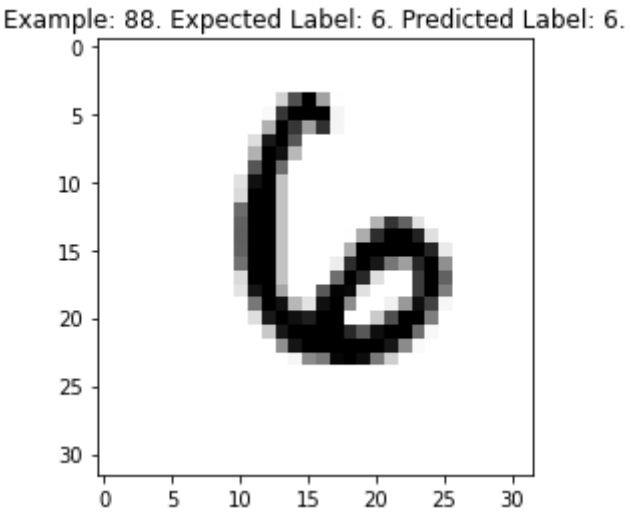

Figura 4. Exemplo de teste corretamente classificado.

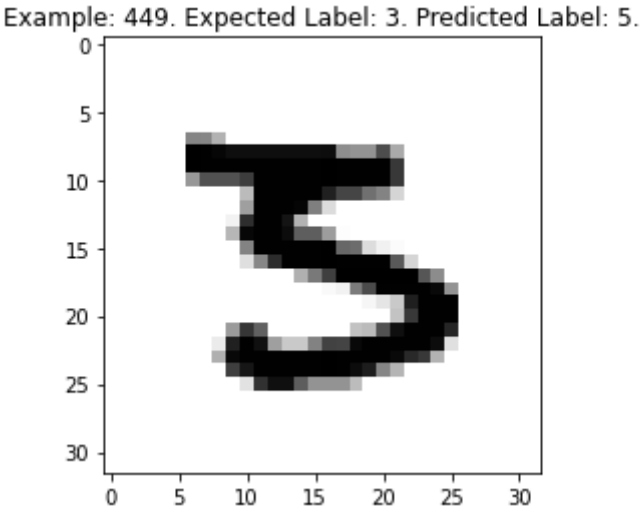

Figura 5. Exemplo de teste incorretamente classificado.

3. Discussão dos Resultados

Dos gráficos e imagens apresentados acima podemos perceber um resultado extremamente positivo da LeNet-5, com uma acurácia de validação de 0.9887 com 10 *epochs*. Baseado nos gráficos do *TensorBoard* nas figuras 1 e 2, é possível que esse resultado consiga ser melhorado com mais tempo de treinamento, porém seriam melhorias marginais considerando a acurácia já elevada.

Nas figuras 4 e 5 podemos observar dois exemplos de teste do modelo após 10 *epochs*, um corretamente e outro incorretamente classificado. Percebe-se a partir da figura 5 que, apesar de ter uma acurácia muito boa, o modelo ainda comete erros parecidos com que humanos cometeriam, em que a ambiguidade de imagens leva a interpretações falhas. No caso apresentado, é crível pensar que um humano também facilmente confundiria o número com um 5, assim como o modelo classifica.