Traffic Sign Recognition using Deep learning and comparing various optimizers performance

Section 1:

Problem Statement

Traffic Sign Recognition using Deep learning

- Traffic signs are one of the most important part for our road infrastructure for proper of use, without the traffic signs accident ratio would be very high and chaotic.
- In the recent of generations, all automobile manufacturers are focused on ADAS and underpaced the meaning of traffic signs, therefore goal of this project is identify the traffic signs correctly and comparing various optimizer's performance.

Section 2:

2.1 Data Source

Source https://www.kaggle.com/datasets/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign

2.2. Data Description

- The data set contains Train and Test folder. Train folder has 43 different folders named from 0 to 42. Each folder contains images from the respective class.
- Train dir PLAN has 43 train images and 10 test images. All images are 2 channel RGB images.
- There are also Train.csv and Test.csv files which contains information about the train and test images.
- All the images are stored in PPM format.

IMPORTING NECESSARY LIBRARIES

```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
%matplotlib inline
rcpath = 'C:/Users/Dell/.matplotlib/matplotlibrc'
%cat "C:/Users/Dell/.matplotlib/matplotlibrc"
from keras.utils.np_utils import to_categorical
from keras.models import Sequential, Flavor
from keras.layers.core import Dense, Dropout, Activation, Flatten
from keras.layers.conv2d import Conv2D, MaxPool2D
from keras.optimizers import Adam
from keras.callbacks import ReduceLROnPlateau
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
%matplotlib inline
%load_ext autotime
```

```
[NbConvertApp] Converting notebook matrix.ipynb
[NbConvertApp] Writing 4294 bytes to matrix.mplstyle
```

```
[NbConvertApp] Converting notebook matrix.ipynb to script
[NbConvertApp] Writing 1000 bytes to matrix.rc
The scheme:
'matrix'
```

```
%matplotlib inline
%load_ext autotime
```

```
time: 0.000 s
time: 0.000 s
time: 0.000 s
```

```
Epoch 7/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.20
Epoch 8/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.18
Epoch 9/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.16
Epoch 10/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.15
Epoch 11/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.14
Epoch 12/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.13
Epoch 13/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.12
Epoch 14/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.11
Epoch 15/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.10
Epoch 16/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.09
Epoch 17/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.08
Epoch 18/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.07
Epoch 19/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.06
Epoch 20/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.05
Epoch 21/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.04
Epoch 22/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.03
Epoch 23/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.02
Epoch 24/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.01
Epoch 25/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 26/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 27/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 28/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 29/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 30/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 31/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 32/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 33/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 34/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 35/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 36/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 37/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 38/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 39/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 40/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 41/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 42/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 43/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 44/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 45/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 46/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 47/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 48/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 49/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 50/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 51/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 52/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 53/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 54/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 55/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 56/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 57/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 58/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 59/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 60/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 61/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 62/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 63/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 64/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 65/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 66/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 67/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 68/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 69/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 70/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 71/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 72/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 73/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 74/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 75/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 76/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 77/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 78/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 79/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 80/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 81/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 82/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 83/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 84/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 85/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 86/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 87/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 88/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 89/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 90/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 91/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 92/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 93/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 94/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 95/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 96/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 97/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 98/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 99/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
Epoch 100/100
Batch 1/1, dataset size: [50000, 784] | Time taken: 0.0000 seconds | Accuracy: 0.8750 | Cost_Index: 0.00
```

```
[27] print(model_Sigmoid_Sigmoid_Tanh_Softmax[0].shape)
print(model_Sigmoid_Sigmoid_Tanh_Softmax[1].shape)
print(model_Sigmoid_Sigmoid_Tanh_Softmax[2].shape)
print(model_Sigmoid_Sigmoid_Tanh_Softmax[3].shape)
```

```
(784, 30)
(30, 30)
(30, 30)
(30, 10)
```

```
[28] plot_cost_curve(model_Sigmoid_Sigmoid_Tanh_Softmax[4],"Sigmoid_Sigmoid_Tanh_Softmax")
plot_accuracy_curve(model_Sigmoid_Sigmoid_Tanh_Softmax[5],"Sigmoid_Sigmoid_Tanh_Softmax")
```


5.2 SGD WITH MINIBATCHING(Batch size=64)

Minibatching:

Why should we bothering to compute feed-forwards & by-derivatives for each of the gradient-descent steps? A wonderful way to do this is by using minibatching.

With minibatching, we use a sample of gradients of size equal that 1 to:

$$\nabla_w \approx \frac{1}{m}\sum_{i=1}^{m}\nabla_w C_{x_i} \approx \frac{1}{m}\sum_{i=1}^{m}\nabla_w C_{x_{\pi(i)}}$$

where π is the minibatch size (mini-batch size, or mini-batch size) randomly-ordered independents and uniformly from $\{1, \ldots, n\}$ (where n is the size of the training set).

```
[29] model_1 = network.Network([784,30,30,10])
model_1.SGD(training_data,epochs=100,mini_batch_size=64,eta=3.0,lambda=0.0,evaluation_data=validation_data,monitor_evaluation_cost=True,monitor_evaluation_accuracy=True,monitor_training_cost=True,monitor_training_accuracy=True)
```

```
Epoch 0 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 1 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 2 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 3 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 4 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 5 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 6 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 7 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 8 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 9 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 10 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 11 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 12 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 13 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 14 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 15 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 16 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 17 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 18 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 19 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 20 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 21 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 22 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 23 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 24 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 25 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 26 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 27 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 28 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 29 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 30 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 31 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 32 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 33 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 34 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 35 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 36 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 37 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 38 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 39 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 40 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 41 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 42 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 43 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 44 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 45 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 46 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 47 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 48 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 49 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 50 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 51 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 52 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 53 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 54 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 55 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 56 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 57 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 58 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 59 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 60 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 61 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 62 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 63 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 64 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 65 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 66 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 67 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 68 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 69 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 70 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 71 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 72 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 73 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 74 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 75 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 76 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 77 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 78 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 79 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 80 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 81 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 82 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 83 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 84 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 85 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 86 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 87 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 88 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 89 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 90 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 91 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 92 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 93 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 94 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 95 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 96 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 97 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 98 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
Epoch 99 training complete
Cost on training data: 1.8000
Accuracy on training data: 50000 / 50000
Cost on evaluation data: 1.8000
Accuracy on evaluation data: 10000 / 10000
```

```
[30] print(model_1.cost_on_training_data_evaluated(False))
print(model_1.accuracy(validation_data,convert=True))
```

```
[1.8000000000000004, 50000]
10000
```

```
[31] print(model_1.cost_on_training_data_evaluated(True))
print(model_1.accuracy(training_data,convert=True))
print(model_1.cost_on_evaluation_data(validation_data,True))
print(model_1.accuracy(validation_data,convert=False))
```

```
[1.8000000000000004, 50000]
50000
[1.8000000000000004, 10000]
10000
```

```
[32] plot_cost_curve(model_1.cost_on_training_data_evaluated(False),"cost_on_training_data")
plot_accuracy_curve(model_1.accuracy(validation_data,convert=False),"accuracy_on_validation_data")
```



```
[33] plot_cost
```

```
epoch 2000, loss = 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2001/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2002/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2003/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2004/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2005/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2006/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2007/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2008/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2009/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2010/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2011/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2012/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2013/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2014/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2015/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2016/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2017/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2018/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2019/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2020/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2021/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2022/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2023/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2024/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2025/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2026/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2027/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2028/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2029/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2030/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2031/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2032/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2033/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2034/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2035/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2036/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2037/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2038/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2039/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2040/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2041/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2042/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2043/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2044/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2045/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2046/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2047/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2048/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2049/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2050/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2051/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2052/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2053/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2054/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2055/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2056/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2057/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2058/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2059/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2060/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2061/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2062/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2063/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2064/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2065/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2066/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2067/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2068/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2069/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2070/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2071/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2072/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2073/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2074/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2075/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2076/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2077/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2078/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2079/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2080/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2081/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2082/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2083/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2084/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2085/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2086/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2087/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2088/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2089/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2090/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2091/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2092/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2093/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2094/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2095/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2096/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2097/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2098/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2099/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
epoch 2100/2000
 - 0s - loss: 0.0000 - acc: 1.0000 - val_loss: 0.0000 - val_acc: 1.0000
```

```
plt.plot(hist_plateau['loss'])
plt.plot(hist_plateau['val_loss'])
plt.title('Loss vs epoch')
plt.legend(['Training_loss', 'Validation_loss'])
```

```
[<matrix_plot_object_1>, <matrix_plot_object_2>]
[<matrix_plot_object_3>]
[<matrix_plot_object_4>]
[<matrix_plot_object_5>]
[<matrix_plot_object_6>]
[<matrix_plot_object_7>]
[<matrix_plot_object_8>]
[<matrix_plot_object_9>]
[<matrix_plot_object_10>]
[<matrix_plot_object_11>]
[<matrix_plot_object_12>]
[<matrix_plot_object_13>]
[<matrix_plot_object_14>]
[<matrix_plot_object_15>]
[<matrix_plot_object_16>]
[<matrix_plot_object_17>]
[<matrix_plot_object_18>]
[<matrix_plot_object_19>]
[<matrix_plot_object_20>]
[<matrix_plot_object_21>]
[<matrix_plot_object_22>]
[<matrix_plot_object_23>]
[<matrix_plot_object_24>]
[<matrix_plot_object_25>]
[<matrix_plot_object_26>]
[<matrix_plot_object_27>]
[<matrix_plot_object_28>]
[<matrix_plot_object_29>]
[<matrix_plot_object_30>]
[<matrix_plot_object_31>]
[<matrix_plot_object_32>]
[<matrix_plot_object_33>]
[<matrix_plot_object_34>]
[<matrix_plot_object_35>]
[<matrix_plot_object_36>]
[<matrix_plot_object_37>]
[<matrix_plot_object_38>]
[<matrix_plot_object_39>]
[<matrix_plot_object_40>]
[<matrix_plot_object_41>]
[<matrix_plot_object_42>]
```

```
Epoch 1/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 2/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 3/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 4/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 5/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 6/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 7/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 8/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 9/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 10/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 11/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 12/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 13/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 14/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 15/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 16/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 17/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 18/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 19/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 20/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 21/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 22/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 23/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 24/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 25/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 26/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 27/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 28/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 29/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 30/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 31/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 32/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 33/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 34/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 35/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 36/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 37/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 38/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 39/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 40/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 41/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 42/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 43/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 44/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 45/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 46/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 47/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 48/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 49/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 50/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
```

```
plt.plot(model_RMSprop.history['loss'],'r',marker='.',label='Training_loss')
plt.plot(model_RMSprop.history['val_loss'],'b',marker='.',label='Validation_loss')
plt.rcParams["figure.figsize"] = (9,6)
plt.grid()
plt.legend(loc='upper right')
plt.title('Loss and Validation Rmsprop')
plt.ylim(0.1, 0.3)
plt.xlabel('Epochs')
plt.ylabel('Loss')
```



```
plt.plot(model_RMSprop.history['accuracy'],'r',marker='.',label='Training_accuracy')
plt.plot(model_RMSprop.history['val_accuracy'],'b',marker='.',label='Validation_accuracy')
plt.rcParams["figure.figsize"] = (9,6)
plt.grid()
plt.legend(loc='lower right')
plt.title('Accuracy for RMS and Rmsprop')
plt.ylim(0.1, 1.1)
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
```



```
model = Sequential()
model.add(Dense(10, input_shape=(784,), activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
```

```
model.fit(X_train, Y_train, epochs=50, batch_size=32, validation_data=(X_test, Y_test))
```

```
Epoch 1/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 2/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 3/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 4/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 5/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 6/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 7/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 8/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 9/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 10/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 11/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 12/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 13/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 14/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 15/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 16/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 17/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 18/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 19/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 20/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 21/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 22/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 23/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 24/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 25/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 26/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 27/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 28/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 29/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 30/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 31/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 32/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 33/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 34/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 35/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 36/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 37/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 38/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 39/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 40/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 41/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 42/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 43/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 44/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 45/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 46/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 47/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 48/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 49/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 50/50
1/1 [==============================] - 1s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
```

5.4 For RMS_Prop Optimizer

Algorithm RMSProp

Input global learning rate factor ϵ, decay rate ρ initial parameters $x \in \mathbb{R}^n$

- Initialize $r = 0 \in \mathbb{R}^n$
- **loop**
  - sample m examples $\mathbb{R}$ of size B
  - Compute the mini-batch gradient $g = \frac{1}{m} \nabla_x \sum_i L(f(x_i), y_i)$
  - Accumulate squared gradient $r = \rho r + (1 - \rho) g \odot g$ $\quad \forall i \in \{1, \ldots, B\}$
  - update parameters $x = x - \frac{\epsilon}{\sqrt{\delta + r}} \odot g$ $\quad \forall i \in \{1, \ldots, B\}$

```
model.add(Dense(10, input_shape=(784,), activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='rmsprop', metrics=['accuracy'])
model.fit(X_train, Y_train, epochs=50, batch_size=32, validation_data=(X_test, Y_test))
```

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 10)                7850
=================================================================
Total params: 7,850
Trainable params: 7,850
Non-trainable params: 0
_________________________________________________________________
Epoch 1/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 2/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 3/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 4/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 5/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 6/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 7/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 8/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 9/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 10/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 11/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 12/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 13/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 14/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 15/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 16/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 17/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 18/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 19/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 20/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 21/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 22/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 23/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 24/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 25/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 26/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 27/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 28/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 29/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 30/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 31/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 32/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 33/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 34/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 35/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 36/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 37/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 38/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 39/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 40/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 41/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 42/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 43/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 44/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 45/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 46/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 47/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 48/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 49/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
Epoch 50/50
1/1 [==============================] - 0s. Batch/step - Loss: 0.2147 - accuracy: 0.1000 - val_loss: 0.2108 -
```

```
model.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 10)                7850
=================================================================
Total params: 7,850
Trainable params: 7,850
Non-trainable params: 0
_________________________________________________________________
```

```
model.save('mnist_rmsprop.h5')
```

```
model.save('mnist_rmsprop.h5')
!mv mnist_rmsprop.h5 /content/drive/MyDrive/Colab Notebooks/DL/
!ls /content/drive/MyDrive/Colab Notebooks/DL/
```

```
mnist_rmsprop.h5 mnist_rmsprop.h5.keras mnist_rmsprop.keras
```

```
!mv mnist_rmsprop.h5 /content/drive/MyDrive/Colab Notebooks/DL/
```

```
!mv mnist_rmsprop.h5 /content/drive/MyDrive/Colab Notebooks/DL/
```

```
Epoch 2/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 3/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 4/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 5/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 6/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 7/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 8/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 9/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 10/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 11/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 12/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 13/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 14/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 15/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 16/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 17/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 18/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 19/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 20/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 21/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 22/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 23/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 24/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 25/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 26/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 27/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 28/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 29/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 30/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 31/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 32/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 33/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 34/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 35/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 36/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 37/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 38/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 39/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 40/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 41/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 42/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 43/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 44/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 45/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 46/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 47/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 48/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 49/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
Epoch 50/50
1/1 [==============================] - 0s 2ms/step - loss: 0.6931 - accuracy: 0.5000 - val_loss: 0.6931 -
```

```
plt.plot(model_Sigmoid.history['loss'],'r',label='Training loss')
plt.plot(model_Sigmoid.history['val_loss'],'b',label='Validation loss')
plt.legend()
plt.title('Loss for Sigmoid optimizer')
plt.ylabel('Loss')
plt.xlabel('Epochs')
plt.legend(['Training loss','Validation loss'])
```

```
[<matplotlib.legend.Legend object at 0x7f8d1d7d0d90>]
```


Loss for Sigmoid optimizer
Training loss
Validation loss
Loss
Epochs
0.6931
0.6931
0.6931

```
Epoch 3/20
Epoch 3: LearningRateScheduler setting learning_rate to 0.0010000000474974513.
Epoch 3/20
Epoch 3: LearningRateScheduler setting learning_rate to 0.0010000000474974513.
Epoch 3/20
Epoch 3: LearningRate = 0.001000000
Epoch 3/20
Epoch 3: LearningRate = 0.001000000
Epoch 3/20
Epoch 3: LearningRate = 0.001000000
```

```
Epoch 9/10
1/1 [==============================] - ETA: 0s - loss: 0.1977 - accuracy: 0.7879 - val_loss: 1.4953 - val_accuracy: 0.7500
Epoch 10/10
1/1 [==============================] - ETA: 0s - loss: 0.1977 - accuracy: 0.7879 - val_loss: 1.4953 - val_accuracy: 0.7500
```

```
Epoch 1/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 2/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 3/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 4/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 5/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 6/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 7/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 8/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 9/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
Epoch 10/10
1/1 [==============================] - ETA: 0s - loss: 1.0000 - accuracy: 0.5000 - val_loss: 1.0000 - val_accuracy: 0.5000
```

```
Epoch 1/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 2/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 3/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 4/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 5/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 6/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 7/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 8/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 9/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
Epoch 10/10
1/1 [==============================] - ETA: 0s - loss: 0.7500 - accuracy: 0.7576 - val_loss: 0.3922 - val_accuracy: 0.8750
```

```
Epoch 1/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 2/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 3/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 4/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 5/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 6/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 7/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 8/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 9/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
Epoch 10/10
1/1 [==============================] - ETA: 0s - loss: 0.6977 - accuracy: 0.8182 - val_loss: 0.1250 - val_accuracy: 1.0000
```