

Contents

- [Introduction](#)
- [Loading data](#)
- [Network 1](#)
- [Network 2](#)

Introduction

This script uses matlab deep learning toolbox instead of implementing networks from scratch. The results that are of interest are discussed in the report.

Loading data

```
[xTrain, tTrain, xValid, tValid, xTest, tTest] = LoadCIFAR(4);

%Plot first 20 images in the training data
for i = 1:20
    subplot(4,5,i);
    imshow(xTrain(:,:, :, i))
end

error('Run the next sections to construct the models using deep learning toolbox. Note that training can take hours. Deep learning toolbox in matlab allows users to use the computers GPU for effective computation')
```

```
ans =

      32      32      3  50000
```

Error using CNN_script (line 16)
Run the next sections to construct the models using deep learning toolbox. Note that training can take hours. Deep learning toolbox in matlab allows users to use the computers GPU for effective computation

Network 1

```
layers_net1 = [ ...
    imageInputLayer([32 32 3])
    convolution2dLayer(5,20,'Padding',1,'Stride',1)
    reluLayer
    maxPooling2dLayer(2,'Padding',0,'Stride',2)
    fullyConnectedLayer(50)
    reluLayer
    fullyConnectedLayer(10)
    softmaxLayer
    classificationLayer];
```

```

options = trainingOptions('sgdm', ...
    'MaxEpochs',120,...
    'Shuffle','every-epoch', ...
    'MiniBatchSize',8192, ...
    'InitialLearnRate',1e-3, ...
    'Momentum',0.9, ...
    'ValidationData',{xValid, tValid}, ...
    'ValidationPatience',3, ...
    'ValidationFrequency',30, ...
    'Plots','training-progress');

net_1 = trainNetwork(xTrain,tTrain,layers_net1,options)

tPred = classify(net_1,xTest);
accuracy = sum(tPred == tTest)/numel(tTest)

```

Network 2

```

layers_net2 = [ ...
    imageInputLayer([32 32 3])
    convolution2dLayer(3,20,'Padding',1, 'Stride',1)
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'Padding',0,'Stride',2)

    convolution2dLayer(3,30,'Padding',1, 'Stride',1)
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2, 'Padding',0,'Stride',2)

    convolution2dLayer(3,50,'Padding',1, 'Stride',1)
    batchNormalizationLayer
    reluLayer
    fullyConnectedLayer(10)
    softmaxLayer
    classificationLayer];

options = trainingOptions('sgdm', ...
    'MaxEpochs',120,...
    'Shuffle','every-epoch', ...
    'MiniBatchSize',8192, ...
    'InitialLearnRate',1e-3, ...
    'Momentum',0.9, ...
    'ValidationData',{xValid, tValid}, ...
    'ValidationPatience',3, ...
    'ValidationFrequency',30, ...
    'Plots','training-progress');

net_2 = trainNetwork(xTrain,tTrain,layers_net2,options)

tPred = classify(net_2,xTest);
accuracy = sum(tPred == tTest)/numel(tTest)

```

