

Assignment: Implementing ResNet in PyTorch

Objective

In this assignment, you will implement a simplified version of ResNet (Residual Network) in PyTorch. You will apply the network to classify images and analyze its performance on CIFAR-10.

Complete the following two labs before going further in the assignment:

Notebook 11.2 - Residual networks: [ipynb/colab](#)

Notebook 11.3 - Batch normalization: [ipynb/colab](#)

Assignment Description

1. Implement Residual Blocks

Specifically, implement the following:

- A basic residual block with two convolutional layers and batch normalization.
- A mechanism for skip connections to allow the input to bypass one or more layers and be added to the output.

2. Build ResNet Architecture

Based on your residual block, build a ResNet-18 architecture. Make sure to include:

- An initial convolutional layer with batch normalization and max pooling.
- Four main stages, each with multiple residual blocks.
- A fully connected layer for classification at the end.

3. Train and Evaluate Your Model

Train your ResNet model on the CIFAR-10 dataset. Follow these steps:

- Split the dataset into training set 70%, validation set 30%
- Use data augmentation techniques to increase the variety of training data.
- Train your model, monitor its performance, and adjust hyperparameters (e.g., learning rate, batch size, optimizer).
- Evaluate your model on the validation set and plot the training and validation accuracy.
- Plot the training and validation loss and accuracy over epochs.

Submission Requirements

1. Code: Submit your code in a Jupyter Notebook file. Ensure it is well-documented and organized.
2. Graphs: Attach plots showing training and validation performance over time.

Guidelines

- Group Work: You may work in pairs, but each student must understand the code and concepts independently.
- Code Quality: Write clean, modular, and well-documented code.

- Experimentation: Feel free to experiment with hyperparameters or try ResNet variants if you are interested, but make sure to document any changes.

Evaluation Criteria

Your submission will be evaluated based on:

- Correctness: Accurate implementation of the ResNet model and residual blocks.
- Functionality: Ability of the model to train on CIFAR-10 and achieve reasonable accuracy.
- Code Quality: Readability, structure, and documentation of code.

CIFAR-10 Dataset

https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html