Computational MR imaging
Laboratory 9: Machine learning in MRI and neural network architecture design

Report is due on Wednesday before the next lab session at 23:50. Please upload your report on StudOn.

Learning objectives

- Get familiar with designing and training neural networks in Pytorch
- Examine effects of dropping data on neural network performance
- Learn about effects of over and underfitting

Installation of the deep learning framework Pytorch

In this lab, you will use PyTorch, a powerful deep-learning package developed by Facebook AI Research, and Google Colab, which provides free GPU sources to accelerate training models. Follow the steps below to start Google Colab with your codes.

1. Download Lab 9 materials from StudOn
2. Upload Jupyter codes and data on Google Drive, keeping the structure as
   - My Drive
     - CMRI
       - Lab9
         - Data
         - Codes
3. Open a Jupyter notebook on Google Drive webpage
4. Click “Runtime” > “Change runtime type” in the menu bar
5. Set “Hardware accelerator” to “GPU”
6. Click “Connect” on the upper right side

1. Diffusion Data Classification

Open the script dti_classification_pytorch_1p10.ipynb. This script serves as a dataloader to for DTI data from the human connectome database. Our goal is to classify labels of brain tissue pixel by pixel based on quantitative diffusion parameters.

i. Look at the labels in the data. What is labeled?
ii. Build and train a fully connected neural network as shown in the lecture
iii. Train the network
iv. Evaluate the results of the network and the training. What could you do to improve the performance?

2. Image Quality Classification

Open the script CNN_recon_quality_classification_pytorch_1p10.ipynb. The goal of this exercise is to classify images as being either fully-sampled or accelerated and reconstructed with compressed sensing.

i. Build and train CNN as shown in the lecture. Try to overfit and underfit the training data, as well as come up with a reasonable model. You can use the architectures that we discussed in the lecture as a guideline, or come up with your own models. I am particularly interested in models that lead to a better performance on the validation set than what we discussed in the lecture.
ii. Train the neural network.
iii. Evaluate the results of the network and the training.