EE–559: Practical Session 4

François Fleuret

https://fleuret.org/dlc/

February 15, 2018

Draft, do not distribute

Introduction

The objective of this session is to implement a convolutional network and test the influence of the architecture on the performance.

You can get an information sheet about the practical sessions and provided helper functions at

https://fleuret.org/dlc/dlc-info-sheet.tex

You can start this sessions with the example from the course:

https://fleuret.org/dlc/dlc_practical_4_embryo.py

1 Training function

Re-organize the code to define and use a function

`train_model(model, train_input, train_target, mini_batch_size)`

Hint: My version is 605 characters long

2 Test error

Write and test a function

`compute_nb_errors(model, input, target, mini_batch_size)`

To compute the number of prediction mistakes using a "winner-take-all" rule, that is the class with the largest output is the predicted one.

Run the training and test ten times, record the test error rates.

With 25 epochs for training, the test error should be around 10% with the small sets, and around 0.7% with the full ones.

Hint: My version is 424 characters long

3 Influence of the number of hidden units

In the default network, the number of hidden units is 200.

Modify the class constructor to take a parameter for that value, and run the training and compute the test error for 10, 50, 200, 500, and $1,000$ hidden units.

4 Three convolutional layers

Write a new class `Net2` with three convolutional layers. Pick the structure you want.