IOC5216 (5275) Brain-Computer Interface, Spring 2021, NYCU-CS

Lab 4: Deep Learning for BCI

Rules

1. Each group submits 1 report (.pdf) and 1 code (.ipynb).
2. Report must contain observations, results and explanations. Please name your files as ***5275_Lab4_GroupNum.pdf.***
3. Paper submission is not allowed. Please use our Docs. template to complete your report.
4. Code must contain comments to explain your code. Only **Python and Pytorch** are allowed. Please name your code as ***5275_Lab4_GroupNum.ipynb***
5. Implementation will be graded by completeness, algorithm correctness, model description, and discussion.
6. Illegal format penalty: -5 points for violating each rule of formats.
7. **PLAGIARISM IS STRICTLY PROHIBITED**. (0 point for Plagiarism)
8. Late submission penalty: original score*0.8 only within a week
9. Due on May 25, 2021 at 09:00 AM.

Introduction

In this lab, we will exercise on applying convolutional neural networks for EEG classification in motor-imagery-based BCI.

Part 1 (50%)

Use the PyTorch code of the EEGNet [1] to perform 4-class motor imagery EEG classification.

Data description

'BCI competition IV 2a' is a well-known motor imagery dataset which is composed of EEG data from 9 subjects. Four motor imagery tasks are involved, namely the imagination of movement of the left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). Each subject has training and testing sessions, and each session involves 72 trials for each class, yielding a total of 72*4 trials per session.

The dataset name will be like BCIC_S{subject_ID}_{train/test}.mat for each file, like “BCIC_S01_T.mat” is a training set for subject one and “BCIC_S03_E.mat”is a testset for subject three.
We will provide the dataset called “BCICIV_2a_mat” to you in the attached file to implement and we will give you a tutorial to teach you how to build neural networks using PyTorch.

Problems

1. (10%) Please build an EEGNet-based individual subject model with training set for subject S01(“BCIC_S01_T.mat”) and testing accuracy with test set (“BCIC_S01_E.mat”). (with learning rate = 0.001, batch size = 32)
    a. Show overall accuracy which is trained with an individual model.
    b. Show accuracy of each of the four classes.
    c. Construct a 4*4 confusion matrix using the classification results.

2. (10%) Still build EEGNet to train a classifier with training set for subject two (“BCIC_S01_T.mat”) and test accuracy with different learning rate and different batch size. Please fill in the table below.

Test accuracy with different learning rate and different batch size.

Learning rate\Batch size	64	32	8	2
0.003				
0.001				
0.0003				
0.0001				

3. (20%) Go through the paper “Spatial Component-wise Convolutional Network (SCCNet) for Motor-Imagery EEG Classification” [2], use each of the training schemes: the subject-independent, subject-dependent, and subject-independent plus fine-tuning, to train the EEGNet and test on S01. For example, in the subject-independent training, a total of 16 sessions for 8 subjects selected from S02 to S09 are served as the training set, while the test set is the same as above (“BCIC_S01_E.mat”). (with learning rate =

0.001, batch size = 32)

a. Show the S01 accuracy using different training schemes.
b. Show accuracy of each of the four classes.
c. Construct a 4*4 confusion matrix using the classification results.

4. (10%) In the “dataloader” function, set the parameter “shuffle” to True or False while training. Show the results of both settings and identify their differences.

Part 2 (50%)

In a field like deep learning application in BCI is moving faster and faster each day. Various CNN architectures are developed to improve the accuracy of EEG pattern recognition. Shallow ConvNet [3], SCCNet [2], and TSception [4]

1. (25%) Perform four training schemes (individual, subject-independent, subject-dependent, subject-independent plus fine-tuning) on all four CNNs with “BCIC_S01_E.mat as the test set.

2. (25%) Compare the results among all four CNNs under different training schemes. Summarize and explain the pros and cons of each model.

TSception code downloading

https://github.com/deepBrains/TSception

PyTorch tutorial link

https://pytorch.org/tutorials/
https://github.com/utkuozbulak/pytorch-custom-dataset-examples

Reference

[1] Lawhern, Vernon J., et al. "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces." *Journal of neural engineering* 15.5 (2018): 056013.

[2] Wei, Chun-Shu, Toshiaki Koike-Akino, and Ye Wang. "Spatial component-wise convolutional network (SCCNet) for motor-imagery EEG classification." *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER).* IEEE, 2019.

[3] Schirrmeister, Robin Tibor, et al. "Deep learning with convolutional neural networks for EEG decoding and

visualization." *Human brain mapping* 38.11 (2017): 5391-5420.

[4] Ding, Yi, et al. "TSception: A Deep Learning Framework for Emotion Detection Using EEG." *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020.