ECE C147/C247, Winter 2021   **Project guidelines**
Neural Networks & Deep Learning   Prof. J.C. Kao
UCLA ECE   TAs: N. Evirgen, A. Ghosh, S. Mathur, T. Monsoor, G. Zhao

The project is intended to give you experience to working on neural networks in a research application. It is also intended to give you experience working with some of the topics covered in the last three weeks of class, for which this project is the evaluation. This means, in this project, you should evaluate at least one of the algorithms we discuss post-CNNs, or maybe even evaluate neural network topics we don't explicitly get to cover in class.

- Recurrent neural networks

- Variational autoencoders

- Generative adversarial networks

- Policy gradient / deep reinforcement learning

If you do not evaluate one of these algorithms, you will lose points for creativity in the rubric described at the end of this document. Further, the default project we provide data for is a classification project, in which case the application of CNNs and RNNs is more clear. Topics like VAEs, GANs, and deep RL may be applicable in other projects, or if you wanted to do something else with the EEG data (e.g., use a VAE to discover features).

The goal will be to optimize the performance of decoding on a particular task; or else to apply one of these techniques to a research area. You may use any packages developed by others to implement these networks (e.g., PyTorch, Keras, TensorFlow etc.). Our philosophy here is that you already understand the principles of training these networks, doing backpropagation, optimization, etc., and are now well-equipped to use these packages that implement several of these operations for you.

You may work on the datasets provided below, or do a project related to your own research / interests. If you are doing a project related to your own research / interests, please e-mail Prof. Kao directly to get approval. In general, as long as the project will explore a post-CNN topic, it should be approved. We will provide approval for custom project requests as long as they are sent prior to Friday, 26 Feb 2021.

Students may work in groups of up to (and including) 4 people. To find teammates, you may use the "Search for Teammates" Piazza functionality. For more information, see `https://support.piazza.com/support/solutions/articles/48001158117-search-for-teammates`. You may also do the project individually.

Project dataset

In class, we have worked with CIFAR-10 extensively. But what about data with temporal components and structure? This project will explore this, using datasets collected from electroencephalography (EEG). A more complete description of the data is available here: `http://www.bbci.de/`

`competition/iv/desc_2a.pdf`

EEG reflects the coordinated activity of millions of neurons near a non-invasive scalp electrode. Because these are scalp potentials, necessarily, they have relatively poor spatiotemporal resolution compared to other neural recording techniques. EEG is believed to be recording dipoles that are transmitted through the scalp.

This dataset is made publicly available through: `http://www.bbci.de/competition/iv/`. However, people typically process these datasets with `MATLAB`. We have formatted the data in a way that it will be easily loadable for you, as opposed to you working with the raw `.gdf` files. These datasets are internal for this class and should not be distributed. If you would like to publish based off of this data, you should visit the BCI competition site and download the raw gdf files after filling out the following agreement:

Each participant has to agree to give reference to the group(s) which recorded the data and to cite (one of) the paper listed in the respective description in each of her/his publications where one of those data sets is analyzed. Furthermore, we request each author to report any publication involving BCI Competiton data sets to us for including it in our list.

After filling out the form and pushing the "I Agree" button an automatic e-mail will be generated containing location and access information for the data set download area.

For each subject, they record from 22 EEG electrodes while the user imagines performing one of four actions. Therefore, this is a classification task (with four outcome classes), where the EEG is used to determine what action the subject was imagining. Follow the instructions below to load the data.

We have processed the data so that you can load the data with numpy. Further, we have removed trials that have NaN's. You can load the data as follows:

```python
In [1]: import numpy as np
```

```python
In [2]: import numpy as np
        X_test = np.load("X_test.npy")
        y_test = np.load("y_test.npy")
        person_train_valid = np.load("person_train_valid.npy")
        X_train_valid = np.load("X_train_valid.npy")
        y_train_valid = np.load("y_train_valid.npy")
        person_test = np.load("person_test.npy")
```

Shape of data

```python
In [4]: print ('Training/Valid data shape: {}'.format(X_train_valid.shape))
        print ('Test data shape: {}'.format(X_test.shape))
        print ('Training/Valid target shape: {}'.format(y_train_valid.shape))
        print ('Test target shape: {}'.format(y_test.shape))
        print ('Person train/valid shape: {}'.format(person_train_valid.shape))
        print ('Person test shape: {}'.format(person_test.shape))
```

```
Training/Valid data shape: (2115, 22, 1000)
Test data shape: (443, 22, 1000)
Training/Valid target shape: (2115,)
Test target shape: (443,)
Person train/valid shape: (2115, 1)
Person test shape: (443, 1)
```

This indicates that there are 2115 trials; each trial has corresponding EEG data from 22 electrodes over 1000 time bins. Please look at the dataset documentation to know more about the data. E.g., Table 2 lists what class labels (769, 770, 771, 772) correspond to. The person files correspond to the subject performing the task, ranging from 0-8 (inclusive) and may be useful should you want to see how well you can classify on individual subjects. In the original data / documentation, you may see that there are 25 channels. We have removed 3 of the channels (so that there are 22) because those 3 channels were for recording eye movements, not brain activity.

EEG questions that we suggest you answer

There are many projects you can imagine doing from this data, but if you'd like a "default" project, we suggest you do the following:

1. Optimize the classification accuracy for subject 1. Does it help to train across all subjects?

2. Optimize the classification accuracy across all subjects. How does the classifier do? Do you notice any interesting trends?

3. Evaluate the classification accuracy as a function of time (e.g., does it increase as you have data over longer periods of time? how much time is required to get a reasonable classification accuracy?)

Feel free to innovate beyond this to earn creativity and insight points. For example, you may gain additional creativity points by trying and comparing different CNN and RNN architectures. Additional insight points may also be rewarded for explaining how these different approaches result in better / worse performance.

Project due date

The project will be due on Monday of Finals week, 15 March, 2021. This means there are approximately 3 weeks to complete the project.

Project submittables

Each group should submit a writeup of their project work, exceeding no more than 3 pages including figures. References are excluded from the 3 pgs (e.g., they may overflow onto a 4th page and be numerous). It is fine to be below the page limit; this is the *maximum*. We appreciate conciseness. We will also ask you to submit your code, so that we can validate your results. If you have a project where you cannot submit your code, please notify us so we can proceed accordingly.

The writeup must adhere to the following template: `http://www.pamitc.org/cvpr15/files/cvpr2015AuthorKit.zip` – so that we can judge all writeups in the same manner without having to worry about different font sizes, etc.

In addition to this writeup, the students should submit two extra pages. One page should contain a table summarizing the performance of all algorithms they tested for the datasets they evaluated

them on (e.g., if you tested on all 9 EEG datasets, in addition to testing one classifier across all subjects, report all of these accuracies in your table). The other page should summarize the architectures they used (e.g., if they used a CNN, describe its layers, the size of the activation maps at the end of each layer; if they used an RNN, how many units does it have, etc.) and details about the training (e.g., what optimizer used, if dataset augmentation was used, what activation functions, etc.). You can think of this page as the "Methods" part of the paper. You may reference these pages in the writeup (as opposed to reproducing these tables / figures in the formal writeup.)

Project writeup

In the writeup, there should be the following sections:

Abstract

A brief description of what you did in the project and the results observed.

Introduction

If you are doing the EEG project, do not use the introduction to formulate the general problem of EEG decoding, as we are all familiar with the EEG problem. Instead, use the introduction to set up and motivate the architectures you pursued and why. If you are doing a project from your own research, please give us brief background.

Results

State the results of your experiments.

Discussion

Discuss insights gained from your project, e.g., what resulted in good performance, and any hypotheses for why this might be the case.

References

List references used in your writeup.

Project grading

Here, we outline the criterion by which we will grade the project. Note, some projects will be more creative than others; some projects will achieve higher performance than others. We will provide room for extraordinary work in one category to compensate for deficiencies in another category. These are the general areas we will look into. Concretely, the final project will be graded on a scale of 20 points, but each section is assigned points so that the sum total can exceed 20 points. Your final project score will be capped at 20 points. You should aim to do a good job in all areas.

1. **Creativity (7 points).**

   How creative and/or diverse is the approach taken by the students? Does the student implement / try various algorithms? Are multiple architectures compared? An example of a project that we would count as creative is comparing e.g., a CNN and an RNN in decoding performance. We recognize projects that implement at least two of the post-CNN algorithms as very creative. Creativity may also result from how one tackles the design of these algorithms.

2. **Insight (7 points).**

   Does the project reveal some insight about the choice of hyperparameters, architectures, etc. on the performance of the algorithm? Is there reasonable insight / explanation / intuition into the results? (i.e., you should not just blindly apply different algorithms to a problem and compare them.)

3. **Performance (6 points).**

   Does the project achieve relatively good performance on the problem, given that the students are training with a CPU (and may not have GPU access)? How do different algorithms compare? If the project is related to one's research, how do results compare to the literature? (i.e., you should not just train a few different algorithms without optimizing them reasonably.) We do recognize that students may not have access to GPUs; if this is a problem for your optimization, state it clearly in your results that you believe performance could be increased with more time; this should be apparent from e.g., a loss function plot with physical time on the x-axis (e.g., showing that after some number of hours, the loss had decreased, but still had a long way to go). We will account for this in grading your performance. Last year, students also had success using Google Colaboratory.

4. **Write-up (4 points).**

   Are the approach, insight, and results clearly presented and explained? Dissemination of results is an important component to any project.

Last note

A potentially helpful resource as you start to tackle this problem may be this paper, which used CNNs with the same dataset: `https://arxiv.org/pdf/1703.05051.pdf`