 Oren Shriki, Erez Wolfson, Nir Friedman

Final Project – Analysis of Motor Imagery Data

Background

- This project will cover the fundamentals of the world of Machine Learning (AI, Data science or any other buzzword of your choice).
- The project focuses on a dataset of cued motor imagery EEG data.
- The PDF file "motor imagery data" describes the data in detail (The file describes a paradigm of foot vs. hand imagination. In our data the idea is the same only with left vs. right hand imagination, and we don't have the 2 sec' fixation step).
- Briefly, in each trial the subject was asked to imagine motor activity in one of 2 classes: left hand or right hand. 160 trials were preformed, for each trial; data from 2 EEG channels C3 and C4 were recorded.
- The data is stored **`motor_imagery_data.mat`** as a struct called **`P_C_S`**.

  The acquired data are stored in **`data`** (**`P_C_S.data`**) field which is 160x768x3. The first dimension represents the trials, the second dimension represents time samples (6 sec with sampling rate of 128 Hz) and the third dimension represents the channels. Channel 1 is C3, Channel 2 is C4 and Channel 3 is a trigger channel (not relevant for classification).
- The correct labels of each trial are found in **`attribute`**, which is 160x4. The second dimension (rows) represents 4 labels (see **`P_C_S.attributename`**): ARTIFACT (row 1), REMOVE (2), LEFT (3), RIGHT (4).
- Your goal is to extract useful features from the data and then train a classification algorithm on these features to predict the label on a single trial basis.
- First, we should explore the data and search for informative features.

1. Visualization

- Write a script that will allow you to visualize the EEG signal in a single channel for trials from a single class. For each class (left & right) draw a figure with 20 subplots corresponding to 20 trials, each subplot should plot the data from both channels (C3 & C4).
- Eyeball the data and see if you can identify qualitative differences between the different classes.

2. Power spectrum

- Calculate the power spectrum from all samples in each class: Calculate the spectrum using both FFT and Pwelch methods (see exercise 5).
  - Plot a spectrum for each class each channel and each method (FFT & Pwelch) separately, total of 8 spectrums. Use subplots to present the results in a meaningful way.
  - DO NOT use samples from the entire trial, remember that only in a specific part of all trials motor imagery took place. Extract only the samples from the relevant time window.
  - Which method is more informative, FFT or Pwelch? Discuss this in the final report (include examples).
  - Select the more suitable method and answer the following questions according to it (From now on there is no need to address results for both methods).
- Compare the power spectra of both classes. Are there any frequency bands that seem useful for separating the classes? Plot both spectra on a single graph to see the difference (do this for C3 and C4 separately).
- Look at the spectrogram of the data (see class ex. 2) and use it too to identify informative frequency bands. Find a way to use spectrograms to obtain a meaningful representation of the spectrum for different conditions. Hint: this time you should use the **entire trial**, not just the imagery time window (explain why). Also, a different spectrogram for each trial is **not** a good idea.
  - For each informative frequency band, calculate the energy in this band for each trial in the relevant time window (the power of each band per trial is 1 scalar value). Prepare a figure with histograms depicting the energy distribution for each of the 2 classes.
  - One way to calculate the energy in a frequency band is to calculate the area under the power spectrum in the relevant frequency range. Use the MATLAB function **bandpower**.

3. Classification

- Extract from each trial the informative features and create a low dimensional representation. Play with these settings until you find the features that yield the best results.
- We will focus on linear discrimination.

- Use **`lda1`** (on moodle) or **`classify`** to train a classifier on part of the data and then test its performance on the rest of the data. Specifically, train the classifier using 70% of the labeled trials and use the remaining 30% for validation. (Try multiple realizations of the training and validation data).
  - **Important**; make sure you are not validating on the data that were used for training.
  - In each run, you should use different subset of trials for training. Select the trials randomly.
  - Explain why this is so important, what are the advantages and disadvantages of using more of the data for training? Why is it wrong to test for results on data that was used for training?
- Try testing for accuracy on the training data, compare the results to the real accuracy you have obtained on the validation data.
- Try classifying with a different set of features (Show 1 example of results using only 1 feature and 1 example of results using more than 6 features). Compare the results.
- Report the quality of classification in terms of 'percent correct' on the validation data, how many of the trials were classified correctly out of the whole data-set? (mean over many realizations\runs).

4. Exercise deliverables

- Submit according to the submission guidelines in Moodle.
- Describe **briefly** what you have done and accompany your results with discussion.
- Be **creative** in your work. This is a project, hence there is more than one correct way to achieve the goal. Try different things on your way to the solution and see how each path is effecting your results.
- Your code should be readable and well commented, your report should contain all relevant figures with explanations of what you have done and the results you obtained (in a way that reflects your understanding). You will be graded on your code, your report **and the quality of your results.**
- As always, feel free to consult with us on any subject.

  Good Luck!