Investigating classifiers to decode execution and observation processes from intracranial signals

Kolly Florian, Mikami Sarah, Waridel Samuel
EPFL 2024 — CS433 ML4Science project — TNE Laboratory

***Abstract*—Machine Learning (ML) techniques have become increasingly prominent in neuroscientific research. In particular, many studies investigated the use of ML tools on electroencephalography (EEG) recordings to classify or predict physical and mental states. In this work, we apply ML models to different classification tasks: (i) Decoding execution from observation (ii) Decoding precision from power grasp in execution paradigm (iii) Decoding precision from power grasp in observation paradigm. Our work is inspired by ongoing debate over the presence of an action-observation network in cortical signals. By characterizing these signals and applying classification models, we aim to uncover similarities and differences between execution and observation processes in upper limb movements, and to verify whether those characteristics are generalizable across different human subjects. Our findings demonstrate that ML models, including Deep Learning (DL) approaches, can effectively decode execution from observation based on EEG features. However, we also observe that, for tasks where the subject is passively observing different movements, the classical models fail to capture the subtle differences in brain activity and therefore achieve low performances. For show that CNN models show promising outcomes in all classification tasks. These results highlight both the potential and challenges of using ML to decode neural activity using intracranial signals.**

I. Introduction

Machine Learning (ML) has become a central tool in neuroscience, with applications in neural engineering (e.g. Brain-Computer Interfaces, BCIs [1]), diagnostics (see [2] for an overview) and fundamental research (such as the recent NeuroAI field). For instance, Jalilifard et al. [3] demonstrated the efficacy of Support Vector Machine classifiers in basic emotions recognition, while Bose et al. [4] used Random Forest classifiers to automate seizure detection in epileptic patients with high accuracy. Indeed, all major ML methods have been applied to neural signals classification [5].

In this work, we examine the applications of supervised ML models in three classification problems using stereo electroencephalography (sEEG) signals: i) distinguishing between the execution and observation of a movement; ii) differentiating between grasping a small object (precision grasp) versus a large object (power grasp); and iii) differentiating between observing the grasp of a small versus a large object. The sEEG data used here were collected from patients with electrodes in various brain regions. All patients were already implanted as part of their clinical treatment for a drug-resistant form of epilepsy.

Section II outlines the data acquisition process, the preprocessing of the sEEG signals and the extraction of features. Section III and Section IV present the models applied to our classification problems and their performance. In Section V, we explore the idea of using only Deep Learning (DL) methods to directly extract features from the sEEG signal and train models. Finally, we discuss our results in Section VI and conclude in Section VII.

II. Data analysis & preprocessing

The dataset comprises signals recorded from four participants implanted with sEEG electrodes in distinct brain regions as part of their epilepsy treatment. During multiple experimental sessions, participants were instructed to either observe or execute specific motor tasks. These tasks involved two types of movements: a palmar grasp (power-based movement) and a pinch grasp (precision-based movement). Each session included multiple trials. The experimental design is explained in Appendix IX-B.

The dataset is a dictionary containing four subjects (labeled *s6*, *s7*, *s11* and *s12*). Each subject participated from one up to three sessions. During each session, the subject completed 256 tasks, equally distributed across observation and execution tasks and across movement types. The number and locations of the channels are not identical across subjects, as they were implanted previous to the experiment.

A. Channel responsiveness

We first filtered the channels of interest: for a given channel and task, we define i) the *baseline signal* as the average signal across all task-specific trials in the one-second window between the start of the trial and the cue light turning on, and ii) the *effect signal* as the average signal across all task-specific trials in the one-second equally spaced window surrounding the moment the object is grasped. Using Welch's method [6], we computed the power spectral density (PSD) for both the *baseline signal* and the *effect signal*. We then performed a t-test to compare the PSDs and define a *responsive channel* as a channel with a statistically significative t-test ($\alpha < 0.05$, hyperparameter). This analysis was conducted separately

for channels responsive during observation trials, execution trials, and responsive to both.

B. Data preprocessing

The signals of each trial and each channel is preprocessed in three steps: (1) **Subsampling**: the signal is subsampled from 2048 Hz to 500 Hz to reduce the computational needs. Since the EEG data contains information up to approximatively 150 Hz, this subsampling respects the Nyquist criterion. (2) **Z-score correction**: Each EEG signal is normalized with Z-score normalization, calculated using the mean and standard deviation of the *baseline signal*. (3) **Resampling**: all trials are resampled to ensure they contain the same number of time points (set to 1500).

C. Feature extraction

Participants with multiple sessions were handled as if they had a larger number of trials. Each task has its own dataset based on the classification task to perform. For action recognition, all trials were taken and labeled as either execution or observation. For movement recognition, trials were taken accordingly to the action type studied and labeled based on the movement type. For all dataset, the features were extracted in a similar way. For each responsive channels of a given task (see II-A), the trial signal was bandpass-filtered between the following frequency bands: alpha (8-12 Hz), beta (12-30 Hz), gamma (30-100 Hz) and theta (4-8 Hz). For each frequency band, the rectified envelope of the signal was calculated using the Hilbert transform. The trial was divided into segments, each containing 1000 time points. These segments overlapped, with each starting 100 time points after the previous one. The values of one window were summarized using the mean and the standard deviation. Finally, features from each responsive channels, frequency bands and time windows were concatenated in a dataframe.

III. Action recognition

In this first classification task, we train and optimize different ML models to predict whether the participant was executing the movement or observing it. The analysis is restricted to channels responsive to both movements. Given the distinct nature of the tasks being compared, we anticipate strong performance across all models.

A. Responsivity of channels

The locations of the responsive channels is critical in understanding a potential action-observation network. To this end, we sorted the responsive channels in ascending order of the t-tests' p-values. For participant *s6*, Figure 1 shows the locations of the ten first responsive channels, in the precentral and the postcentral gyri. Table I shows the locations of the most responsive channels for all participants.

The most important regions are similar across participants and are directly linked to movement (ex : prec.) or somatosensory information (ex : postc.). This was expected as the task is linked with active hand movement and requires sensory feedback. It is interesting to note that some of the participants (*s11* and *s12*) have activity in areas that are more related to self-awareness and self-perception (ex: sf. and ins.), which could be much more linked to the observation trials.

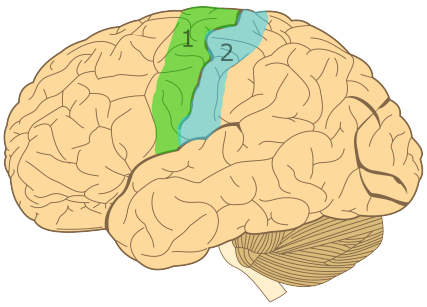

Figure 1. Locations of the ten most responsive channels in *s6*: The green area (1) represents the precentral gyrus, and the blue area (2) corresponds to the postcentral gyrus. Edited from Hugh Guiney, CC BY-SA 3.0 https://creativecommons.org/licenses/by-sa/3.0, via Wikimedia Commons

Part.	# channels	Main locations (# channels)
s6	40	prec. (5), postc. (5)
s7	26	prec. (7), postc. (1), cmf. (1), parso. (1)
s11	27	prec. (1), rmf. (1), sf. (5), WMpara. (2), para. (1)
s12	29	ins. (4), sp. (1), pc. (2), para. (1), postc. (2)

Table I
Number of responsive channels in total and location of the ten most responsive per participant. Appendix IV contains the list of acronyms

For this task we see that the regions vary quite a bit between participants. Participant *s6* has only two regions with responsive channels, while *s12* has five. The regions with the most importance are similar across participants and are directly linked to movement (ex : prec.)[7] or somatosensory information (ex : postc.)[8], especially for the first two participants. This makes sense as this task is linked with active hand movement and requires sensory feedback. It is interesting to note that some of the participants (*s11* and *s12*) have activity in areas that have weaker ties to motor and sensorimotor functions and perform higher level cognitive tasks (ex: sf. and ins.)[9], [10], which could be much more linked to the observation trials.

B. Analysis of frequency bands

We investigate if specific frequency bands could gives more information than other and if there are differences across the action type. We trained the model with a dataset containing only features from a specific frequency band. We then compared the performance of the models across the different frequency bands (see Appendix 9). For action recognition, all models can detect the action type with high accuracy, regardless of the frequency band (above 75%). The best performance was obtained with delta and

gamma frequency bands, with accuracies around 95%, which suggest that these bands are the most informative for differentiating observation from execution.

C. Baseline data

To investigate the relevance of selecting only responsive channels, we also generated a list of channels that are not responsive, and computed features from them. We run each model on both sets of data, noting that they have a different number of channels. We discuss only the results of participant *s6*.

D. Models

To create and train the classical ML models, we adopted the well-known and performant Scikit-Learn library [11]. The DL models are developed using the PyTorch library [12]. We selected models based on the literature available on EEG analysis.

Logistic regression: Logistic regression (LR) is a computationally simple ML technique, and has previously used in EEG classification [13], [14]. It was trained with and without Principal Components Analysis (PCA, 95% explained variance).

SVM: We train two Singular Vector Machines (SVM) models, with and without PCA. SVM can handle high dimensional data, even for small datasets. Although more computationally demanding than other methods, it is widely used in EEG classification with good results [15]. We use a grid search to find good hyperparameters for the SVMs (see Appendix IX-D).

Random forest: Random Forest (RF) classifiers are frequently used with a lower set of carefully crafted features (e.g. in [16] with 11 features), we hypothesize that we can also have interesting results using RF here. We use a grid search to find good hyperparameters (see Appendix IX-D).

MLP: In addition to traditional ML models, we also train Multilayer Perceptron (MLP) models. We chose to include MLP as it is widely used in EEG classification[1]. The main issue with MLP is that they are universal approximators and are thus prone to overfitting, especially for datasets as small as ours.

We try different MLP and select the best by using a validation subset. MLP models are trained for 10 epochs with batches of 4 datapoints. We use the AdamW optimizer, which directly applies weight decay (10^{-2}, hyperparameter) directly during the parameter update, leading to more consistent regularization and better generalization. We use the cross-entropy loss. MLPs are optimized and discussed only for participant *s6*.

[1]3'610 results for 'EEG MLP classification' on Google Scholar in 2024

E. Results

We tested the models on the same test set. As expected, all models obtained very high accuracies. When using non-responsive channels, we see a large drop in performance, although it is higher than chance (see Appendix 5). This confirmed our hypothesis that we can reduce significantly the number of features by first selecting responsive channels.

The MLP is performing very well for participant *s6*: on the test set, we obtain an accuracy of 95% while the baseline is at 49%. The best model is also the simplest: two layers of 8 neurons each, with a learning rate of 10^{-1}.

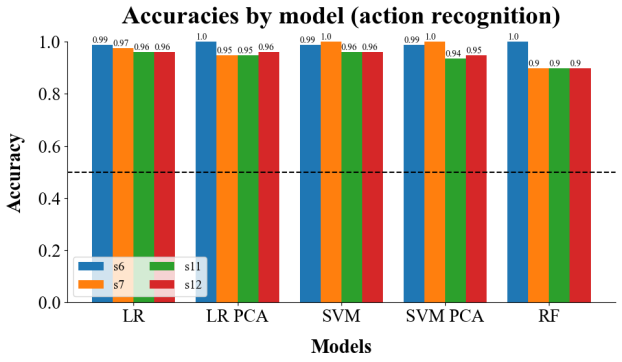

Figure 2. Accuracies by model and participant for the action recognition task: all models perform very well when trained on features extracted from responsive channels

F. Analysis of frequency bands

We investigate if specific frequency bands could give more information than other and if there are differences across the action type. We train the classical models with a dataset containing only features from a specific frequency band. We then compared the performance of the models across the different frequency bands (see Appendix 9). For action recognition, all models can detect the action type with high accuracy, regardless of the frequency band (above 75%). The best performance was obtained with delta and gamma frequency bands, with accuracies around 95%, which suggests that these bands are the most informative for differentiating observation from execution.

IV. Movement recognition

For the second and third type of classification tasks, we analyze whether ML models are able to classify the type of movement done (precision grasp or power grasp) during execution and observation. We expect both tasks to be hard to learn. Observation, in particular, is a complex task, as the participant is simply seeing two different movements that have the same goal of lifting an object. In this part, we lessen the restrictions to select the *responsive channels*: it suffices for a channel to be responsive in execution or in observation respectfully to be included.

A. Responsive channels

The number of responsive channels and the locations of the ten most responsive channels for each participant are shown in Appendix II for execution and Appendix III for observation.

In both of these tasks, we see similar overall trends between all of the participants. Most channels are located in regions related to movement and somatosensory information such as the precentral gyrus (prec.)[7] and the postcentral gyrus (postc.)[8] with quite a few appearances of areas that are linked less directly to these tasks and are known to perform higher level cognitive tasts. Similarly to the first task, we see that participants *s6* and *s7* are more related to the moter task, while participants *s11* and *s12* have activity in areas that are higher level cognitive tasks. Between the action recognition task and the movement recognition task, we see that the regions are quite similar, but the proportions in which they appear changes. For example, the paracentral cortex (para.)[17] is more present in the action recognition task than in the movement recognition task.

B. Analysis of frequency bands

The models were trained with features extracted from each frequency bands for both execution (see Appendix 10) and observation (see Appendix 11). For execution, the models performed slightly better than chance level, with theta and high gamma bands that performed higher than 60%. For observation, most models are below chance level except for high gamma frequency band, around 52-58%. The models are not able to capture the differences between the two types of movements with only one frequency band.

C. Models

We use the same families of model as in the previous part for both execution and observation paradigms. We test the same hyperparameters grid for MLP, although these tasks might be significantly harder.

D. Results

As expected, our results are significantly lower in those tasks than in the first part.

Execution trials: (see 6) LR with and without PCA tends to perform better than the other models (56-77% across participant). The accuracy obtained for SVM with and without PCA is around the same range except for participant *s7* which have accuracies below chance level and RF is around chance level for all participant (46-64%).

Observation trials: (see 7) The trends are very similar to execution trials with slightly lower accuracies.

V. Convolutional approaches

Instead of extracting features on the responsive channels before training deep networks, one can ask whether the networks should learn to do that themselves. We explore this question in this section. We train two types of Convolution Neural Networks (CNN) directly on the trial signals. The hope is that the convolutional layers manage to extract important features from the signal, giving it as input to the subsequent MLP. The models won't be optimized as we are focusing on the proof of concept.

In this section, we are not focusing on performance and thus wont optimize models. Instead, we are attempting new techniques as proof of concept. As MLP, we use only participant *s6* in this section. We used only participant *s6*.

A. Models

The architectures of the CNNs used are detailed in Appendix IX-J. We are not testing multiple CNN architecture, but use a validation set for the learning rate, weight decay and the subsequent MLP layers.

One-dimensional CNN: One-dimensional CNNs are composed of kernels that are convolved with a channel's signal directly. As they operate on data in one dimension, the input will be a tensor of size (N, C, L), where N is the batch size, C the number of (responsive) channels and L the fixed length of each trial. This is close to how we computed our features: supposing the same kernel size as the length of our moving averages, we would get the same features if the network learned moving averages too. Given this and the known power of CNNs, we expect this model to have the best performance. The caveat is overfitting, as we have small datasets.

Two-dimensional CNN: Two-dimensional CNNs are composed of 2D kernels that will operate on multiple channels at the same time. This means that we give as input a tensor of dimensions (N, C, H, W), with H the number of (responsive) channels and W the length of each signal. The naming of the dimensions H and W is not arbitrary. Indeed, we can think of this superposition of channels as a picture, where each signal is a position on the y-axis and every timepoint a position on the x-axis.

B. Results

The 2D CNN can recognize actions with a 96% accuracy, while its 1D counterpart has 100% accuracy. For movement recognition, the 2D CNN manages only 62% accuracy for execution, with a 92% accuracy for the 1D CNN. In observation, both models are good, with 77% accuracy for 2D and 85% for 1D CNN. Those results are highly encouraging, and we encourage further research towards an automatic way to extract features.

VI. Discussion

All models offered high accuracy in the action recognition task. Interestingly, we observed that our models also performed well even when given features extracted from only one frequency band. The locations of the responsive channels were found to be linked to movement or somatosensation, which corresponds to the experiment.

As expected, both movement recognition tasks resulted in low performance across all the classical models, as well as for shallow neural networks. We hypothesize that this is due to only low differences in the neural signals between the paradigms studied.

However, by approaching the problem from a pure Deep Learning point of view, it appears that convolutional networks are able to themselves extract relevant features for classification. Without hard hyperparameters optimization, we obtained significantly higher accuracy than any other model for the movement recognition task.

VII. Conclusion

In this project, we used various ML models to analyze the classification accuracies of three distinct tasks. While every model was capable of classifying well the first task, the performance dropped when the differences in brain signals were more subtle. Our work therefore shows that ML models can already successfully be applied to (s)EEG signals without the need for handcrafted features, but they need clear differences in signal. Further experiments can be done on the models developed here to finetune them and verify whether they are able to achieve similar accuracy as in more distinct tasks or not. Additionally, more research is necessary on the preprocessing pipeline and feature extraction as to find optimal data to feed the models. Maybe the future of feature extraction is by using ML methods directly.

VIII. Ethical risks

It is clear that the analysis of EEG and other brain signals present a unique set of ethical challenges. Given that we are not sharing the dataset for privacy concern, we will focus on this ethical risk in this brief analysis. EEG signals reveal information not only about the physical health of a subject, but also mental states, emotions and cognitive functions. With the popularization of such tools, we risk losing a fundamental right to mental privacy. This is not only true for emotions, but also religious and political opinions. For instance, neuropolitics[18] is a whole field of research that wishes to find the neural roots of our political beliefs. Prediction is just a step further, step that already stated with work such as [19] and [20].

As current methods require possible invasive electrodes, or to use expensive and complicated machines such as fMRI scanners, primarily patients ongoing clinical treatment and study subjects are affected by those risks at the moment. However, the potential violation of cognitive privacy underscores the need for robust privacy and anonymization measures when using EEG datasets. If not correctly established, third parties could gain valuable data on the private mental processes of a user and use it to their advantages. Companies such as Tesla or Meta already invested in such technologies, and not renowned for their privacy-oriented policies. The severity and ramifications of cognitive privacy violating in a large pool of user is hard to quantify, but without strict policies, the possibility of a future without private thought becomes alarmingly plausible.

Even without speculating about future developments in brain signal analysis, EEG signals are already used in identifying individuals. For example, tools for User Authentication via EEG signals already exists[21]. It is then clear that, when in possession of private EEG data, precautions must be taken to ensure that this data remains within the restricted pool of authorized researchers who have been granted access. In this project, we were careful not to upload this data on any online repository (e.g. GitHub), and we will not give the dataset to correctors outside the TNE laboratory. Moreover, we did not access any personal information from the participants involved.

Advanced methods exist to encrypt data while allowing computations on it[22]. Unfortunately, the time and scope of this project did not permit the implementation of such technologies.

Acknowledgements

We thank Leonardo Pollina for taking us with him on this project, and for his quick and helpful answers to all our questions.

IX. Appendix

A. Experimental setup

The participants were asked to keep their hands on the resting positions. The led lights are used both to indicate who will realise the action and to specify which objects has to be grasped. A.) execution trial: the participant must pick the small green ball using a precise pinch grasp. B.) observation trial: the experimenter must pick the large red ball using a power grasp. C.) control action, not used in this project.

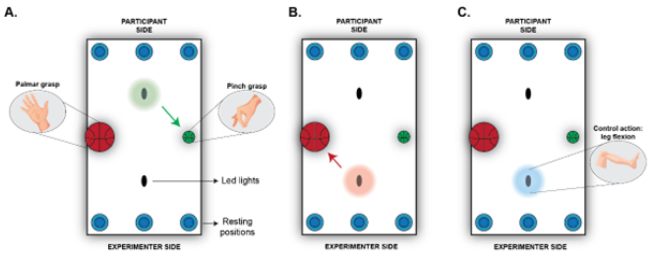

Figure 3. Experimental setup; courtesy of L. Pollina

B. Timeline

Each trial is structured as follows (see Appendix 4): execution trial begins with a start cue. One second later, a light cue signals the type of object (small or large) to be targeted. This light remains illuminated for 0.5 seconds before turning off. After an additional 1.5 seconds, a "go" signal prompts the participant to lift their hand from the resting position, grasp the indicated object, return it to its original location, and place their hand back in the resting position. Observation trials followed an identical structure, except that participants passively observed the experimenter performing the actions. A light cue indicated the participant's role (execution or observation) in each trial.

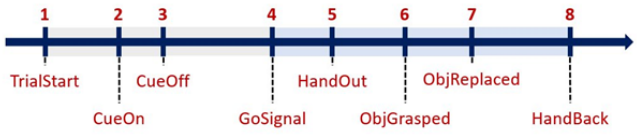

Figure 4. Timing of the trials: the areas shaded in grey indicates fixed timing, and the areas shaded in blue indicates timing that depends on the speed of execution of either the participant (execution) or the experimenter (observation); courtesy of L. Pollina

C. Responsive channels for movement recognition

Here, we show the locations of the ten responsive channels in the two movement recognition tasks. Appendix II is for the execution tasks, and Appendix III is for the observation tasks.

Part.	# channels	locations for execution (# channels)
s6	68	spm. (2), postc. (1), prec. (6), cmf. (1)
s7	48	postc. (1), th. (1), prec. (8)
s11	52	prec. (4), sf. (5), WMpara. (1)
s12	76	para. (1), pc. (3), postc. (2), sp. (1), ins. (3)

Table II
Number of responsive channels for execution trials and location of the ten most responsive per participant. Appendix IV contains the list of acronyms

Part.	# channels	locations for observation (# channels)
s6	51	postc. (6), prec. (4)
s7	33	cmf. (1), postc. (2), prec. (7)
s11	43	pren. (1), cmf. (1), para. (1), sf. (6), rmf. (1)
s12	48	para. (2), pc. (3), postc. (1), sf. (1), prec. (1), ins. (2)

Table III
Number of responsive channels for observation trials and location of the ten most responsive per participant. Appendix IV contains the list of acronyms

D. Classical models' hyperparameters

SVM: The optimal hyperparameters are found within the following sets using grid search:

- **Kernel**: linear / rbf / sigmoid
- **C**: 0.1 / 1 / 10 / 100 / 1000

Random Forest: The optimal hyperparameters are found within the following sets using grid search:

- **Number of estimators**: 10 / 50 / 90 / 130
- **Maximum depth**: 10 / 25 / 50

E. Acronyms

Appendix IV contains the list of brain regions' acronyms used in the report. It does not contain all the regions the participants had electrodes in, but only those relevant in our analyses.

Acronym	Corresponding region
cmf.	caudal middle frontal gyrus
ins.	Insula
para.	paracentral
parso.	parsopercularis
pc.	posterior cingulate
postc.	postcentral gyrus
prec.	precentral gyrus
pren.	precuneus
rmf.	rostral middle frontal
sf.	superior frontal gyrus
sp.	superior parietal gyrus
spm.	supramarginal
th.	thalamus
WMpara.	White Matter paracentral

Table IV
List of acronyms for the brain regions

F. Accuracies in action recognition compared to baseline

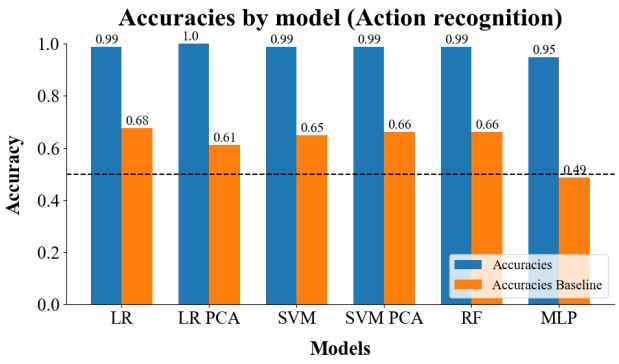

Figure 5. Accuracies by model for the action recognition task compared to baseline for participant s6

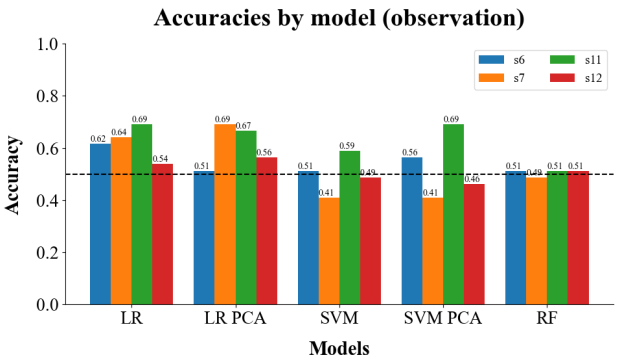

Figure 7. Accuracies by model and participant for the movement recognition task, executed by the experimenter (observation)

G. Accuracies in movement recognition

The following graphs represent the accuracies obtained by model and by participant for the two movement recognition classification tasks. As mentioned in the main report, these accuracies are all low, due to the difficulty of finding neural differences between the compared tasks.

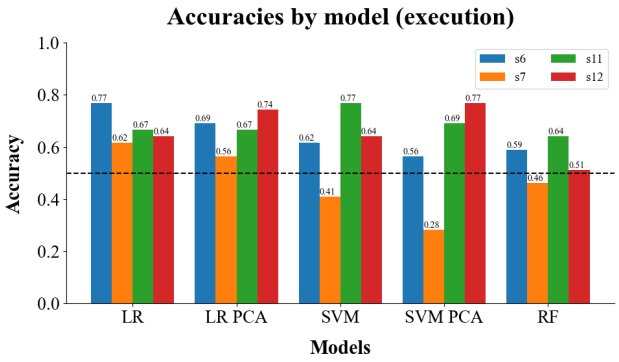

Figure 6. Accuracies by model and participant for the movement recognition task, executed by the subjects (execution)

H. Accuracies depending on the number of responsive channels

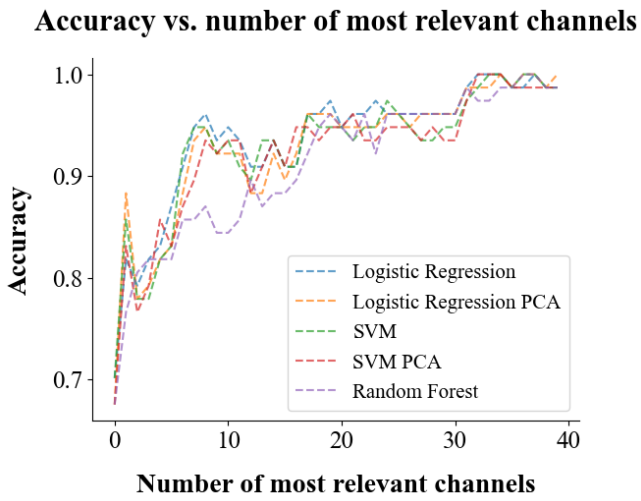

Figure 8. Accuracies depending on the number of responsive channels used by models for action recognition task on participant *s6*

I. Analysis of accuracy by frequency bands

Here, we provide an analysis of the accuracies of our model based on which frequency band we give to compute the features.

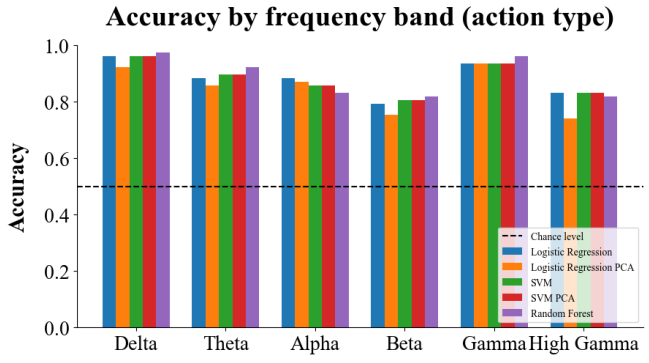

Figure 9. Accuracies by model and frequency band used for feature extraction, action recognition task on participant *s6*

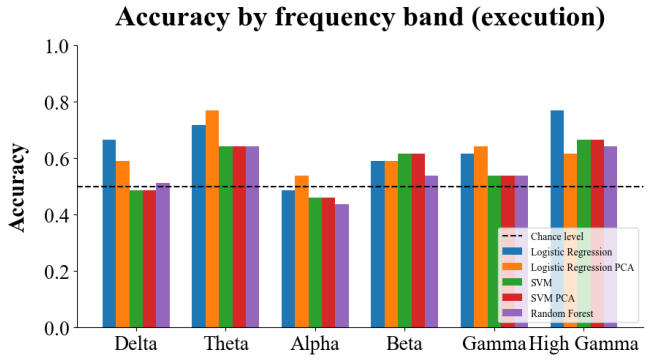

Figure 10. Accuracies by model and frequency band used for feature extraction, movement recognition task on participant *s6* (execution)

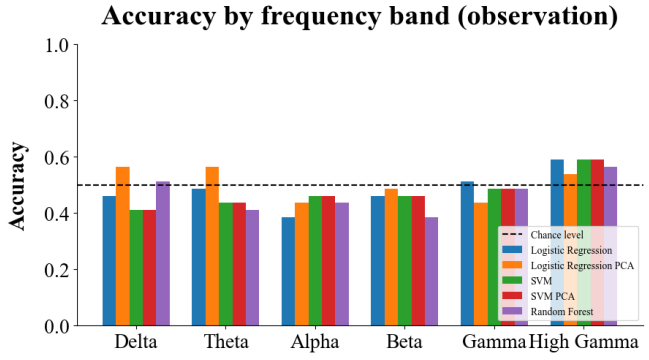

Figure 11. Accuracies by model and frequency band used for feature extraction, movement recognition task on participant *s6* (observation)

J. CNN architectures

2D CNNs: The 2D CNN architecture is as follows:

- Convolution layer: 2 channels, kernels of size 5×5, padding of 1, stride of size 3×3, ReLU
- Max-pooling layer: kernel of size 2×2
- Batch normalization
- Convolution layer: 4 channels, kernels of size 3×3, padding of 1, stride of size 3×3, ReLU
- Max-pooling layer: kernel of size 2×2
- Batch normalization
- Flattening
- Linear layer of 32 neurons, ReLU, dropout of 0.1
- Linear layer of 16 neurons, ReLU, dropout of 0.1
- Linear layer of 2 neurons, outputs the logits

We train all 2D CNNs with a learning rate of 0.001 over 10 epochs with a weight decay of 10 to avoid fast overfitting.

1D CNNs: The 1D CNN architecture is as follows:

- Convolution layer: 8 channels, kernel of size 10, padding of 1, stride of size 5, ReLU
- Max-pooling layer: kernel of size 2
- Batch normalization
- Convolution layer: 16 channels, kernel of size 50, padding of 1, stride of size 5, ReLU
- Max-pooling layer: kernel of size 2
- Batch normalization
- Flattening
- Linear layer of 64 neurons, ReLU, dropout of 0.1
- Linear layer of 32 neurons, ReLU, dropout of 0.1
- Linear layer of 2 neurons, outputs the logits

We train all 1D CNNs with a learning rate of 0.001 over 10 epochs with weight decay of 10 to avoid fast overfitting.

REFERENCES

[1] S. Vaid, P. Singh, and C. Kaur, "Eeg signal analysis for bci interface: A review," in *2015 Fifth International Conference on Advanced Computing & Communication Technologies*, 2015, pp. 143–147.

[2] S. Vieira, W. H. Pinaya, and A. Mechelli, "Using deep learning to investigate the neuroimaging correlates of psychiatric and neurological disorders: Methods and applications," *Neuroscience & Biobehavioral Reviews*, vol. 74, pp. 58–75, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0149763416305176

[3] A. Jalilifard, E. B. Pizzolato, and M. K. Islam, "Emotion classification using single-channel scalp-eeg recording," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2016, pp. 845–849.

[4] S. Bose, V. Rama, N. Warangal, and C. Rama Rao, "Eeg signal analysis for seizure detection using discrete wavelet transform and random forest," in *2017 International Conference on Computer and Applications (ICCA)*, 2017, pp. 369–378.

[5] M.-P. Hosseini, A. Hosseini, and K. Ahi, "A review on machine learning for eeg signal processing in bioengineering," *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 204–218, 2021.

[6] O. M. Solomon, Jr, "Psd computations using welch's method. [power spectral density (psd)]," 12 1991. [Online]. Available: https://www.osti.gov/biblio/5688766

[7] P. T. Linnea Banker, "Neuroanatomy, precentral gyrus," 2024, pMID: 31334938. [Online]. Available: http://www.ncbi.nlm.nih.gov/books/NBK544218/

[8] P. T. Joseph DiGuiseppi, "Neuroanatomy, postcentral gyrus," 2024, pMID: 31751015. [Online]. Available: http://www.ncbi.nlm.nih.gov/books/NBK549825/

[9] F. d. Boisgueheneuc, R. Levy, E. Volle, M. Seassau, H. Duffau, S. Kinkingnehun, Y. Samson, S. Zhang, and B. Dubois, "Functions of the left superior frontal gyrus in humans: a lesion study," *Brain*, vol. 129, no. 12, pp. 3315–3328, 09 2006. [Online]. Available: https://doi.org/10.1093/brain/awl244

[10] L. Q. Uddin, J. S. Nomi, Hébert-Seropian, G. B., J., and O. Boucher, "Structure and function of the human insula," *Journal of clinical neurophysiology : official publication of the American Electroencephalographic Society*, vol. 34, pp. 300–306, 2017, pMID: 28644199, PMCID: PMC6032992. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6032992/

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[12] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[13] A. Subasi and E. Erçelebi, "Classification of eeg signals using neural network and logistic regression," *Computer Methods and Programs in Biomedicine*, vol. 78, no. 2, pp. 87–99, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260705000246

[14] R. Tomioka, K. Aihara, and K.-R. Müller, "Logistic regression for single trial eeg classification," in *Advances in Neural Information Processing Systems*, B. Schölkopf, J. Platt, and T. Hoffman, Eds., vol. 19. MIT Press, 2006. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2006/file/35937e34256cf4e5b2f7da08871d2a0b-Paper.pdf

[15] M. N. A. H. Sha'abani, N. Fuad, N. Jamal, and M. F. Ismail, "knn and svm classification for eeg: A review," in *InECCE2019*, A. N. Kasruddin Nasir, M. A. Ahmad, M. S. Najib, Y. Abdul Wahab, N. A. Othman, N. M. Abd Ghani, A. Irawan, S. Khatun, R. M. T. Raja Ismail, M. M. Saari, M. R. Daud, and A. A. Mohd Faudzi, Eds. Singapore: Springer Singapore, 2020, pp. 555–565.

[16] C. Donos, M. Dümpelmann, and A. Schulze-Bonhage, "Early seizure detection algorithm based on intracranial eeg and random forest classification," *International Journal of Neural Systems*, vol. 25, no. 05, p. 1550023, 2015, pMID: 26022388. [Online]. Available: https://doi.org/10.1142/S0129065715500239

[17] A. Patra, "Morphology and morphometry of human paracentral lobule: An anatomical study with its application in neurosurgery," *Asian Journal of Neurosurgery*, vol. 16, pp. 349–353, 2021, pMID: 34268163, PMCID: PMC8244697. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8244697/

[18] D. Schreiber, "Neuropolitics: Twenty years later," *Politics and the Life Sciences*, vol. 36, no. 2, p. 114–131, 2017.

[19] G. Galli, D. Angelucci, S. Bode, C. De Giorgi, L. De Sio, A. Paparo, G. Di Lorenzo, and V. Betti, "Early EEG responses to pre-electoral survey items reflect political attitudes and predict voting behavior," *Scientific Reports*, vol. 11, no. 1, p. 18692, Sep. 2021. [Online]. Available: https://doi.org/10.1038/s41598-021-96193-y

[20] J. H. Yun, Y. Kim, and E.-J. Lee, "ERP Study of Liberals' and Conservatives' Moral Reasoning Processes: Evidence from South Korea," *Journal of Business Ethics*, vol. 176, no. 4, pp. 723–739, Apr. 2022. [Online]. Available: https://doi.org/10.1007/s10551-021-04734-2

[21] C. A. Fidas and D. Lyras, "A review of eeg-based user authentication: Trends and future research directions," *IEEE Access*, vol. 11, pp. 22 917–22 934, 2023.

[22] A. B. Popescu, I. A. Taca, C. I. Nita, A. Vizitiu, R. Demeter, C. Suciu, and L. M. Itu, "Privacy preserving classification of eeg data using machine learning and homomorphic encryption," *Applied Sciences*, vol. 11, no. 16, 2021. [Online]. Available: https://www.mdpi.com/2076-3417/11/16/7360