Pre-Processing and Classification of Task-Related fMRI Data

Szabolcs Torma
Torma.Szabolcs@aut.bme.hu

***Abstract*—fMRI is a widely used technique for brain-related research and diagnosis. fMRI data have a relatively good spatial resolution, making it possible to understand more about how the brain works and examine neural activity locations during different mental states. Multivoxel pattern analysis (MVPA) is a technique with growing attention which with it is possible to investigate the brain's neural activity. This work applies support vector machines and convolutional neural networks for MVPA. Their performances are compared with within-subject and inter-subject measurements. Furthermore, the results are visualized with brain plots.**

***Keywords*—fMRI, support vector machines, neural networks, pre-processing, classification**

I. Introduction

FUNCTIONAL MRI data is a brain imaging technique that makes it possible to examine brain activity. fMRI measures and produces blood-oxygen-level-dependent (BOLD) contrast maps that can be associated with the brain's neural activity. The spatial resolution of fMRI data is considered good compared to other brain imaging technologies. Meanwhile, the scan measurement time, called repetition time, is around 1-2 seconds, making the temporal resolution low compared to e.g. electroencephalography (EEG). However, this is not a problem for two reasons: first, the BOLD signal is also considered slow (it takes around 4-5 seconds to reach its peak), and second, with fMRI, the main goal is to understand the functional connectivity between brain areas. [4]

The active areas of the brain during different task-related mental states are inferred most often with the general linear model (GLM). This model became widely used due to its relative simplicity and interpretability, however, it relies on assumptions (e.g. covariance across neighboring voxels is not informative) that limit its efficiency. Another emerging approach is called multivoxel pattern analysis (MVPA). [3], [4]

An exciting solution with this approach is to define the MVPA as a supervised discriminative classification problem and apply machine learning (ML) models to solve it. Support vector machines (SVMs) are commonly used ML algorithms for image and signal processing, but more importantly, they are shown to be a good fit for fMRI data processing. [3], [4] Neural networks are also proven their capabilities in many fields, and their 3D implementations make it possible to apply them to 3D data, like fMRI. [6]

This work applies kernel SVMs and convolutional neural networks (CNN) to MVPA on two data sets. The performances are measured by metrics commonly used in classification tasks (accuracy, recall, precision, F-score). Furthermore, relevant voxels are visualized to be able to assess the results qualitatively, also.

The work is structured as follows: in Section II, SVMs and CNNs are introduced, and the classification problem is defined in the context of MVPA, in Section III, the description of the used data sets and pre-processing steps are given, Section IV the results are presented, and finally, the work is concluded in V.

II. Methods

The problem specification is given in this section, followed by a brief description of the used machine learning algorithms.

A. Supervised classification

In the context of this work, the classification's goal is to predict the class to which a data sample belongs based on a set of features. Binary classification is performed on the *ds002336* set ([1]), while on *ds003548* ([2]) multi-class classification. The former problem is somewhat simpler because, in the case of multi-class classification, we have to consider the existence of ambiguous regions. Two approaches exist: one is called the one-vs-one, where multiple binary classifications are done, and the other is the one-vs-rest classification. [5] With the NN, a one-vs-rest, while with the SVM, a one-vs-one classification is performed in this work. Labels guide the training of the models, thus, the training process is done in a supervised manner.

B. Connection to MVPA

In the simplest form of MVPA, the goal is to recognize spatial patterns in the data at each time step and decode these patterns. Therefore, the core idea is to take a feature set at every time step t, which consists of all (or a subset) of the voxels formed as a vector f, and assign a label to this set at t. To each feature (voxel), a weight parameter w is assigned that defines the voxel's contribution for the decision of a given class. The assumption is that the weights imply areas associated with the mental task. [3]

C. Algorithms

SVMs and CNNs are relatively flexible and commonly used models for classification problems, therefore, they could be good starting points for the current problem.

1) Support Vector Machine: When linearly separable data are assumed in a binary classification problem, the goal is to estimate a hyperplane with a maximum margin to separate the data samples into the two classes. SVMs, in this exact case, produce a discriminant function that can be described as:

$$f(\underline{x}) = \underline{w}^T \underline{x} + b \tag{1}$$

, where $\underline{x}$ is a vector with M features, $\underline{w}$ is corresponding weight vector and b is the bias. To every value, x_n of $\underline{x}$, there is a corresponding label value l_n. The maximum margin can be found by solving the following:

$$\underset{\underline{w},b}{argmax}\{\frac{1}{||\underline{w}||} \underset{n}{min}[l_n(\underline{w}^T x_n + b)]\} \tag{2}$$

This solution can be further simplified to $\underset{\underline{w},b}{argmin}\frac{1}{2}||\underline{w}||^2$. After this, the constrained optimization problem is solved by introducing Lagrange multipliers.

However, in order to be able to solve non-linear problems, the "kernel trick" have to be used. In this case, the earlier function can be expressed as:

$$f(\underline{x}) = \sum_{n=1}^{N} a_n l_n k(\underline{x}, x_n) + b \tag{3}$$

,where $k(\underline{x}, x_n)$ is the kernel function. This kernel function is $k_{RBF}(\underline{x}, x_n) = exp(-\frac{||x-x'||^2}{2\sigma^2})$, where σ is the kernel width and $k_{RBF}(\underline{x}, x_n)$ is called the radial basis function kernel. [5]

2) Convolutional Neural Network: Neural networks are flexible models. However, one of their most significant limitations is that they quickly become inapplicable for many features. Furthermore, local features can be lost during processing, and the network is necessary robust enough. Convolutional neural networks, however, solve many of these issues by using filters. The size of the filters is fixed, and they are run over the input data to get a feature map, therefore, for every feature map, there is a corresponding filter. The use of filters solves the problem of having an untractable number of parameters, Also, as they extract patterns locally, they tend to be more robust. The output of such a layer is produced as

$$\begin{aligned} y(N_i, C_{out_j}) = &\sum_{k=0}^{C_{in}-1} weight(C_{outj}, k) \star input(N_i, k) \\ &+ bias(C_{outj}) \end{aligned} \tag{4}$$

, where $\star$ is the cross-correlation operation between the input segment and the filter. [7]

Usually, convolutional layers contain many filters (also called kernels). The sequence of the layers with activation functions between them form CNNs, which can solve non-linear complex problems.

The solution is done by searching for a parameter set that minimizes a criterion function. This is done iteratively by using the back-propagation method.

III. Data sets and Pre-processing

The current section describes one emotion-recognition and two motor-imagery task-related data sets used in this work. The pre-processing decision and steps are also included. All data sets were pre-processed the same way.

A. Data sets

a) ds002336: 10 subjects participated in the measurements. The task was to move a ball figure displayed on a monitor in front of the patients across a window by only imagining the movement of their right hand. The block design had only two states, resting and imaging, therefore, only 2 classes can be differentiated in this study. The volume of each voxel in this study is $2 \times 2 \times 4$ mm^3, and a TR of $2s$ was used. [1]

b) ds003548: the set contains data from 16 subjects. A psychological block-based task was given to the subject. More specifically, the faces of people with 4 different emotions ate shown to the participants. The emotions were neutral, happy, sad, and angry. In addition to these images, 2 other blocks (scrambled faces and blank figures) were shown to the volunteers. During the recordings, the participants had to push one button out of two based on the gender of the shown person to keep the attention at a high level. All runs were balanced in terms of emotion classes. The volume of each voxel in this study is $4 \times 4 \times 4$ mm^3, and a TR of $2s$ was used. [2]

B. Pre-processing

Pre-processing fMRI data is essential, but the steps are not trivial. Many noise and artifact sources can influence the data. On all data sets, the following steps were performed:

1) **Skull-strip** of the anatomical T1-weighted record to get only brain tissue-containing data. This step helps in the later tissue segmentation, functional image co-registration, and template normalization steps
2) **Tissue segmentation** of the anatomical image into 6 tissues: the grey matter, white matter, cerebrospinal fluid, bone, and other soft tissues.
3) **Thresholding** the white matter mask that will be used for the co-registration of the functional data to the anatomical.
4) **Motion correction** of the fMRI scans in time dimensions. This step is necessary to reduce motion-related artifacts in the scans. This step is especially needed in the case of the *ds003548* set because the subjects had to move their fingers, which can affect head scans.
5) **Slice-timing correction** reduces misalignment between the scans in the spatial dimension.
6) **Co-registration** of the functional MRI samples to the anatomical samples to have more accurate functional data. Anatomical data have better resolution, therefore, it is more accurately shaped. In this step, the segmentation results are also used for better alignment. The registration is done by minimizing a cost function calculated based on the voxel intensities. This step assumes that function

and anatomical MRI are close to each other, therefore, rigid-body transformation can be used.

7) **Normalization** of the registered data to a standard template, which is the MNI152 template in this case. This template ensures that group analysis can be done more easily on the data without being influenced by the individuals' brain shape.
8) **Smoothing** the data with a Gaussian kernel to eliminate high-frequency artifacts.

The resulting pre-processed data are now as homogenized as possible and ready for MVPA.

IV. Experiments and Results

This section presents the MVPA experiments on the previously introduced data sets and the results of these experiments.

A. Failure with CNNs

The idea was to apply a 3D CNN on data blocks corresponding to each class without modifying the data structure and to see if the network can synchronously learn patterns in time and spatial domain. This meant that 4D was passed through the network, and filters operated in 3D. As a first approach, I modified the EEGNet and EfficientNetV2 networks to operate in 3D and see their performances. I used an Adam optimizer with a learning rate of 1e-3 and a binary cross-entropy loss function. Unfortunately, neither of the networks was able to work. Both of them could predict only one class and not the other. To try another approach, I used a 2D version of the networks, and the input data had the form of the data that went into the SVMs (this is described later). This version did not work either. As a last experiment, I tried to fit one linear layer with a softmax activation only to see if the network could linearly separate the data, but this approach did not work either. Therefore, I can not present results in this work other than that the NNs did not work.

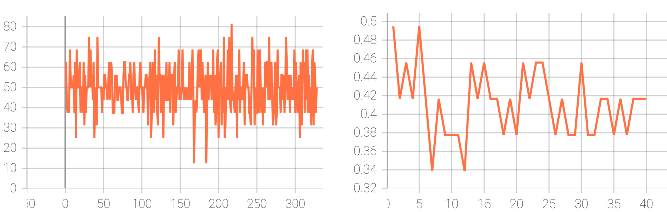


Figure 1: Training and validation loss over batch steps and epochs. Losses are not descending, implying learning failure.

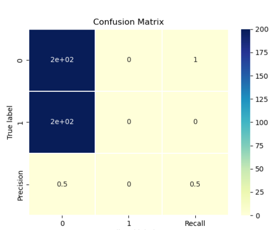


Figure 2: Confusion matrix of the classification of the NN.

B. MVPA on ds002336

As data were given sequentially one big file, I had to extract the blocks of scans corresponding to each label. After separating the blocks, each block had 10 scans for one class. One measurement for one class is shaped as $10 \times 44 \times 64 \times 64$, where the dimensions are $Time \times Height \times Width \times Slice$. In this dataset, every volunteer participated in 1 run of measurements, and in each run, each class (rest and task) was presented 10 times. I kept the classes balanced.

I followed the literature for the SVM and used the scans at every time step as samples and the corresponding voxels to that time step as features. For this, the voxels had to be flattened.

a) Inter-subject: for inter-subject classification, I collected all 2000 scans. 5-fold cross-validation was done with balanced sets. The confusion matrices for the results can be seen in Figure 10. The linear kernel performed much better than the RBF one. The linear kernel achieved 75.25% accuracy, while the RBF only 45.75%, which is worse then chance.

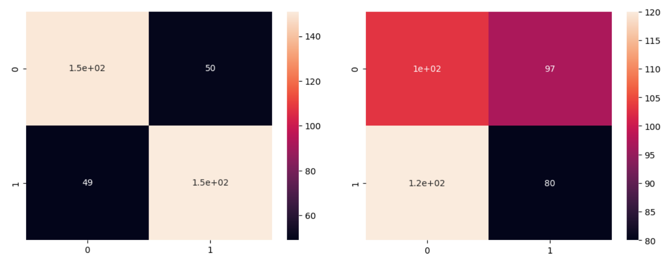


Figure 3: Inter-subject classification results with confusion matrix. 0: Rest, 1: Motor Imagery Task. Linear kernel SVM (on left) outperforms RBF kernel SVM (on right)

b) Within-subject: for each subject, I trained a separate SVM with linear and RBF kernel. The previously mentioned cross-validation method was used for every subject, but only with the particular subject's data. Here, I present the results in Figure 2 that show the mean accuracies and the standard deviations.

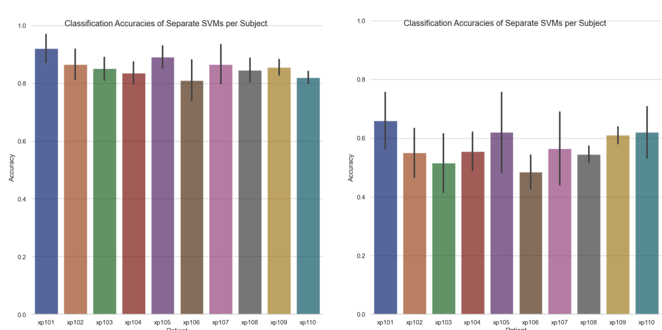


Figure 4: Mean and standard deviation of the accuracies of the linear (left) and RBF (right) SVMs.

Figure 4 again shows that linear SVM performed better in the binary classification task than on this data set than the RBF

SVM.

c) MVPA: plots in Figure 5 show weights learned by the linear kernel SVM. Interestingly, many voxels outside the brain also had much influence, possibly due to the correction and co-registration methods in the pre-processing steps.

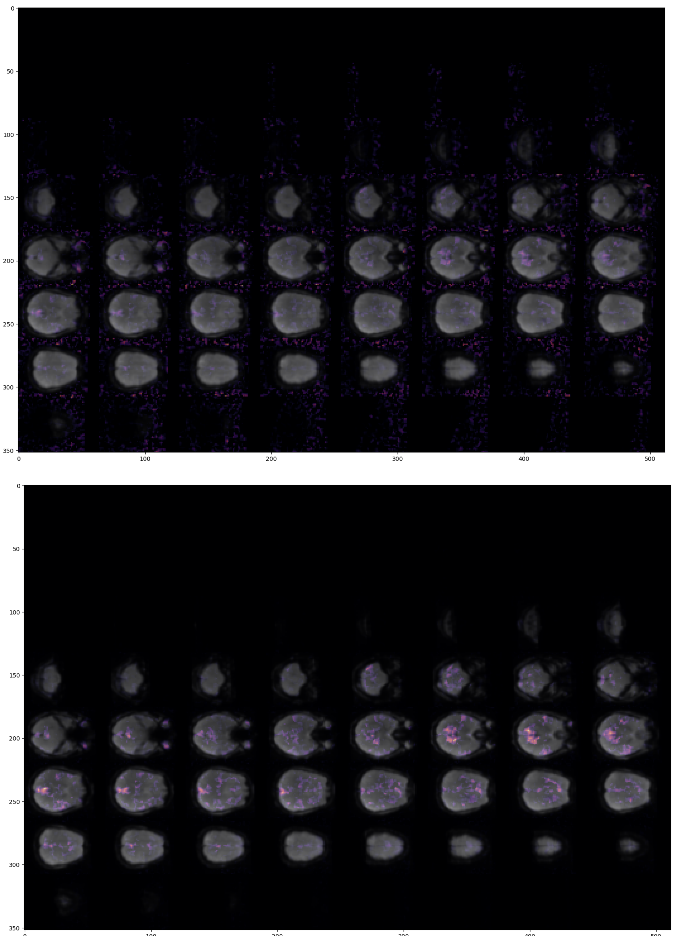


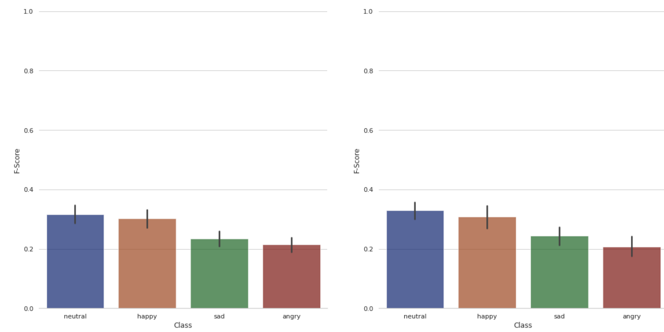
Figure 5: MVPA voxels visualized in a montage of slices. Upper unweighted with anatomy image. Lower weighted with anatomy.

C. MVPA on ds003548

As in the previous setup, I had to extract the blocks of scans corresponding to each label. After separating the blocks, each block had 15 scans for one class. One measurement for one class is shaped as $15 \times 45 \times 79 \times 79$, where the dimensions are $Time \times Height \times Width \times Slice$. In this data set, every volunteer participated in 5 runs of measurements, and in each run, each class was presented twice. Therefore, altogether there are 600 scans for each subject. I kept the classes balanced. For this data set only within-subject results are presented because only one subject's data could fit into the RAM. (30GB)

I followed the literature for the SVM and used the scans at every time step as samples and the corresponding voxels to that time step as features. For this, the voxels had to be flattened.

a) Within-subject: in this setup, for each subject, I trained separate SVMs with linear and RBF kernels. For every subject, the previously mentioned cross-validation method was used. Figures 6, 7, 8 show the classes' F-Score, Recall, and Precision scores based on the trainings and validations with each subject's data.



,

Figure 6: Mean and standard deviation of the F-Scores of the linear (left) and RBF (right) SVMs.

The SVMs performed more similarly in this case than on the other data set. F-Scores are slightly above chance (25%) for the neutral and happy classes.

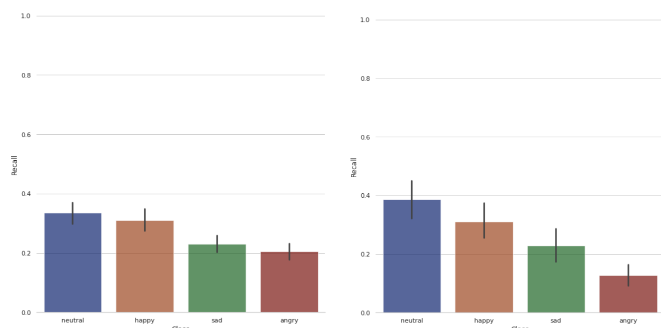


Figure 7: Mean and standard deviation of the recall scores of the linear (left) and RBF (right) SVMs.

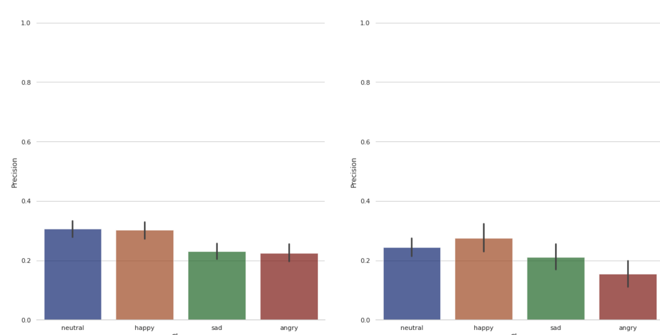


Figure 8: Mean and standard deviation of the precision scores of the linear (left) and RBF (right) SVMs.

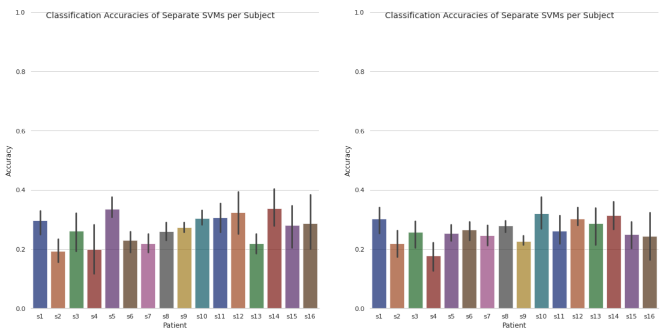


Figure 9: Mean and standard deviation of the accuracies of the linear (left) and RBF (right) SVMs.

The results of this experiment show that both SVMs performed similarly, and at best, they could achieve only slightly better classification accuracy than chance. Generally, the scores are better for the *neutral* and *happy* classes. These results are similar to the ones in [2], where the publishers of the data set found that most differences can be found between the *neutral* and *happy*.

b) MVPA: the plots in Figure 9 show the coefficients learned by the linear kernel SVM in a montage from the slices. The visualized voxels show similar results to the ones found in the previous experiment with the data set *ds002336*.

V. Conclusion

In this work, linear and RBF kernel SVMs and CNNs were applied to fMRI data for MVPA. CNNs could not perform at any level. However, linear kernel SVM performed quite well.

Qualitative results show different brain areas that played a role in classifying emotion- and motor imagery-related events. However, far-reaching conclusions can not be drawn from these results, as many other voxels around the brain tissues had significant coefficients in the trained SVMs.

Based on the presented scores, another interesting finding is that it is possible that classification can be performed much more successfully when a patient must perform a mentally challenging task instead of simply perceiving events. It should be noted that data quality differences between the sets were not examined very profoundly, therefore, this implication needs more investigation.

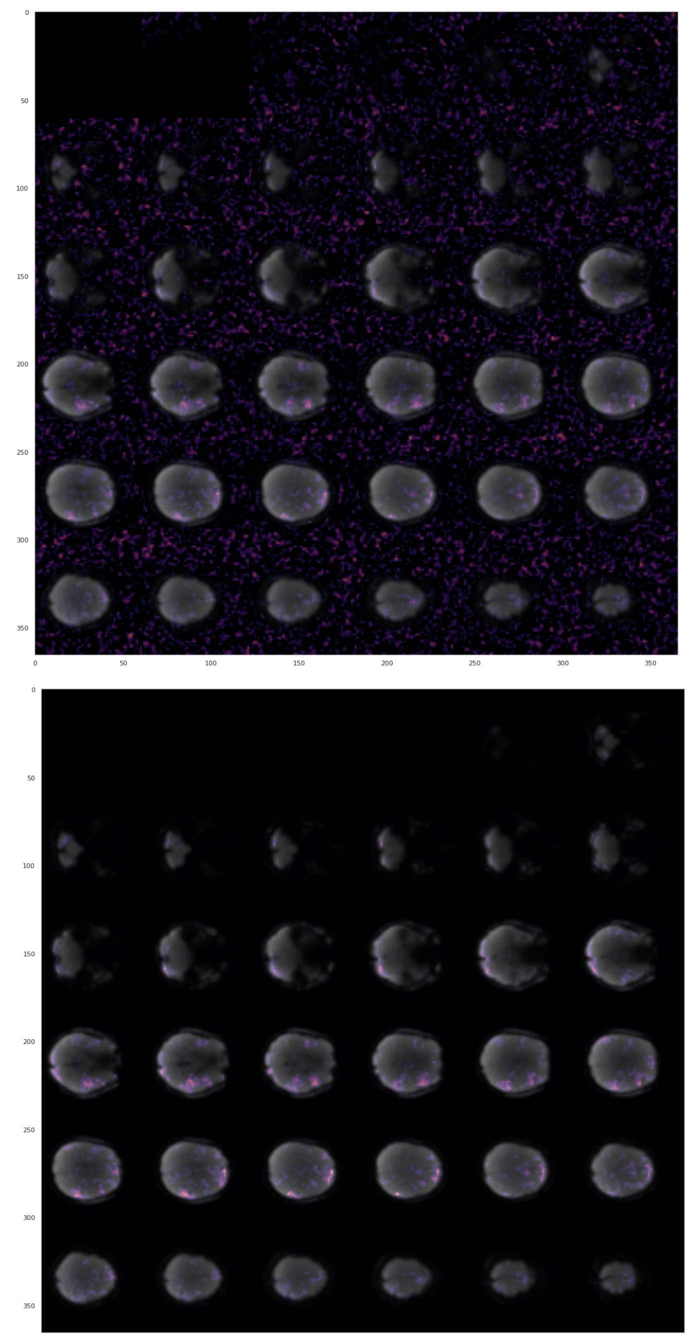


Figure 10: MVPA voxels visualized in a montage of slices. Upper unweighted image, lower weighted with the original fMRI data.

References

[1] Lioi, G., Cury, C., Perronnet, L. et al. Simultaneous EEG-fMRI during a neurofeedback task, a brain imaging dataset for multimodal data integration. Sci Data 7, 173 (2020). https://doi.org/10.1038/s41597-020-0498-3

[2] Isaac David, Fernando Barrios Localizing Brain Function Based on Full Multivariate Activity Patterns: The Case of Visual Perception and Emotion Decoding bioRxiv 2021.04.04.438425; https://doi.org/10.1101/2021.04.04.438425

[3] Mahmoudi A, Takerkart S, Regragui F, Boussaoud D, Brovelli A. Multivoxel pattern analysis for FMRI data: a review. Comput Math Methods Med. 2012;2012:961257. Epub 2012 Dec 6. PMID: 23401720; PMCID: PMC3529504. https://doi.org/10.1155/2012/961257

[4] Poldrack, R., Mumford, J., & Nichols, T. In Handbook of Functional MRI Data Analysis. Cambridge: Cambridge University Press. (2011). https://doi.org/10.1017/CBO9780511895029

[5] Christopher M. Bishop. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg. 2006.

[6] Hanh Vu, Hyun-Chul Kim, Minyoung Jung, Jong-Hwan Lee. fMRI volume classification using a 3D convolutional neural network robust to shifted and scaled neuronal activations, NeuroImage, Volume 223, 2020, https://doi.org/10.1016/j.neuroimage.2020.117328

[7] https://pytorch.org/docs/stable/generated/torch.nn.Conv3d.html [Last accessed: 12.20.2022]