Data augmentation for EEG signals

Joseph Paillard, Thomas Moreau*, Cédric Rommel* & Alexandre Gramfort
Université Paris-Saclay, Inria, CEA, Palaiseau, 91120, France
{firstname.lastname}@inria.fr

1 Introduction

Decoding the brain electrical activity is a great scientific challenge for both clinicians and researchers seeking a better understanding of brain dynamics. Recent attempts to leverage deep learning for this difficult task have shown promising results [1]. These new methods have led to performance gains in a wide range of clinically relevant tasks, such as the automatic sleep stage classification from polysomnographic recordings [2]. Furthermore, deep learning models are capable of automatically learning relevant representations from high dimensional data [3], while previously used brain decoding methods relied on prior knowledge and handcrafted features [4]. Hence, deep learning approaches require a less sharp understanding of the underlying neurophysiology and are thus more versatile.

Nevertheless, most of the breakthroughs in deep learning have been enabled by large datasets such as *ImageNet* [5], while similar datasets do not exist in neuroscience where labelled brain data remains comparatively scarce. Indeed, labelling EEG recordings requires a high expertise, is time consuming and can sometimes be inaccurate due to the bias introduced by the human annotator [6]. A second obstacle to the application of deep learning in neuroscience is the high inter-subject variability that is inherent in brain signals [7]. Along with the lack of data, this property makes the generalization on unseen subjects all the more difficult. Without proper regularization, both of these problems can hinder generalization performance and lead to overfitting.

To mitigate the small scale of the neuroscience databases, a promising direction is the use of data augmentation [8]. First, it allows to increase artificially the size of the training set by adding new synthetic examples. These examples are generated by randomly transforming existing ones in a label-preserving way. Second, data augmentation helps to teach the decision function to become invariant to the transformation enforced, thus softly reducing the hypothesis space of the training problem. Consequently, it can be interpreted as a regularization method that induces a useful bias by preventing the model from focusing on irrelevant features [9], which in the end makes it less prone to overfitting [10].

Although data augmentation is a well-established method in computer vision it is still under-explored for brain data. Among the few studies using data augmentations for EEG signals, only a fraction of existing transformations is studied simultaneously [11]–[14]. Moreover, existing review papers mainly focus on grouping and summarizing results from previous articles instead of carrying out new experiments in comparable settings [15]. In this paper, we propose a unified analysis of most data augmentation methods for EEG signals previously proposed in the literature. After presenting our experimental setting and protocol (section 2), we describe the considered transformations that can be used for EEG signals and give insights on the invariances they encode. Then, we assess the effects of these augmentations on standard EEG classification tasks such as sleep stage classification and brain-computer interfaces (BCI). Finally, we also highlight the fact that certain invariances are more present in certain data classes.

In an attempt to organize the plethora of augmentations studied in this manuscript, they were structured in three different groups: augmentations acting on the frequency domain (section 3), augmentations acting on the time domain (section 4) and augmentations acting on the spatial domain

*equal contribution

(section 5).

2 Experimental protocol

In this section, we present the common experimental protocols used to study each data augmentation on EEG signals. To provide answers in a broad scope, experiments are repeated on two different tasks, sleep stage classification and BCI, as explained in Section 2.1. Our objective is three-fold: (i) to evaluate the impact of the strength of each transformation (Section 2.2.1), (ii) to compare the global relative benefit of each augmentation depending on the train set size (Section 2.2.2), and (iii) to highlight how the effects of augmentations vary across classes within the same dataset (Section 2.2.3).

2.1 EEG classification tasks

2.1.1 Sleep stage classification

Dataset and preprocessing Likewise, experiments were carried out in the context of sleep stage classification. This task is usually performed by sleep experts and is essential to diagnose sleep disorders such as sleep apnea or insomnia. It consists in the classification of 30-second EEG windows into 5 stages following the *American Academy of Sleep Medecine* (*AASM*) manual [16]: Wake (W), Rapid Eye Mouvement (REM) and Non REM stages 1, 2 and 3 (respectively N1, N2 and N3). For this purpose, we used the *SleepPhysionet* dataset [17], which contains whole night polysomnographic recordings from 78 healthy subjects using two EEG channels: Fpz-Cz and Pz-Oz. In this dataset, each signal window has been annotated by well-trained technicians according to the *Rechtschaffen and Kales* manual [18], before being re-asigned to the more recent stages from the *AASM* manual. As suggested in [19], a minimal preprocessing consisting in a lowpass filter with a cutoff frequency of $30Hz$ is applied to the data, followed by a simple standardization step.

Model and training details The model that is used is a deep convolutional neural network that has been designed for sleep stage classification tasks [19]. It is trained using the Adam optimizer [20] with a learning rate of 10^{-3}. The weighed cross-entropy loss is used to take into account the classes imbalance of the dataset. The batch size is set to 16 to preserve the stochasticity of the gradient descent in very low data regimes. We train the model for 300 epochs using early-stopping with a patience of 30 epochs [19].

Splitting strategy EEG recordings have a strong inter-subject variability [7]. Subsequently, to test the trained models in real conditions, some subjects must be set aside and used only for testing in order to avoid subject related information leakage [15]. So as to take this defining feature into account, the following splitting strategy has been defined. First, the dataset is separated into k-folds, each of them containing different subjects. One fold is left out for testing and among the remaining $(k-1)$, 20% are used for validation. Finally, a subsample of the leftover dataset is extracted using a stratified split and used for training. This last step allows to reach small proportions without distorting the distribution of classes.

2.1.2 BCI

Dataset and preprocessing The first dataset on which the experiments are carried out is *BCI IV 2a* [21]. It consists of recordings from 9 subjects using 22 EEG electrodes. The subjects were asked to perform four motor imagery tasks, namely to imagine the movement of the left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). This dataset was preprocessed using a bandpass filter between 4 Hz and 38 Hz followed by an exponential moving standardization as in [22]. Then trials of 4.5 seconds are used as inputs. Each trial starts 0.5 seconds before the cue that tells the subject to perform the motor imagery task and ends when the cue disappears.

Model and training details The model used is a generic deep convolutional network [22], as implemented in the library BRAINDECODE [23]. It can take advantage of the spatio-temporal structure of the data using spatial filters and convolutions across time. The architecture is inspired by the success of Common Spatial Patterns (CSP) methods [4]. Following the work of [22], the gradient descent is made using the AdamW optimizer [24] with a learning rate of 6.25×10^{-4}. The model is trained for 1600 epochs using early stopping with a patience of 160 epochs and a batch size of 64.

Splitting strategy According to the rules of the BCI competition [21], for each subject our model is trained on the first session (or a fraction of it) and evaluated on the second session. The experiment is repeated across all nine subjects.

2.2 Experiments

2.2.1 Parameters selection

To address the implications of the strength of the transformation, it should first be acknowledged that several augmentations have a parameter that can be adjusted to control its strength. For example, the Gaussian noise augmentation has a parameter σ corresponding to the standard deviation of the distribution from which the noise is sampled. The choice of such parameter is thus crucial and even as important as the choice of the augmentation itself, as later depicted in our results (sections 3, 4 and 5).

This experiment unfolds in two steps: the first consists in narrowing down the range of parameter values. To do so, an upstream manual exploration allows to estimate an interval with an upper bound above which the augmentation distorts the relevant information contained in the signal and thus systematically triggers a loss of performance. In the case of Gaussian noise, with σ values greater than 0.2, EEG signals are so distorted that the augmentation is systematically detrimental to the learning. The second and last step is a grid-search carried out using 11 linearly spaced values within the aforementioned interval. For each parameter value, a 10-fold cross validation score is computed using the balanced accuracy as a metric. Since data augmentation is all the more efficient in low data regime (as shown in our experiments from the following sections), we carried our parameter selection using a small balanced fraction of the initial datasets (e.g. 2^{-7} for *SleepPhysionet* dataset).

2.2.2 Learning curves

The second experiment aims at comparing the benefits brought by different augmentation methods. To this end, for each augmentation operation, we compute a learning curve which shows the performance reached by a model trained on fractions of increasing size of the training set. The results obtained with each data augmentation are then compared to a baseline, consisting in the same model trained with no data augmentation.

2.2.3 Per class analysis

Finally, to get a deeper understanding of the effects of data augmentations, we take a closer look at single points from the learning curve and analyze how effects vary depending on the class. This observation is guided by the intuition that the invariances encoded by data augmentations might be more relevant for some classes than others. For example, the channel symmetry augmentation, which switches EEG channels from left and right hemispheres, is much more relevant for non-lateralized brain activities such as imagining tong movements, whereas it is irrelevant for lateralized functions, such as right- or left-hand movements.

The first experiments on the *SleepPhysionet* dataset reveals that augmentations are systematically more efficient in low data regimes and to the same extent on underrepresented classes (*cf.* Section 3.3.2). Since we are seeking to assess the effects of transformations on the learning of representations for each class, the differences in the number of samples per class, which is a confounding factor, needs to be levelled. So as to tackle this issue, a downsampling step is added to the pre-processing pipeline, allowing to work with balanced classes.

3 Frequency domain augmentations

3.1 Rationale of the transformations

Frequency shift For many EEG classification tasks, it is believed a substantial part of the information lies in the frequency domain of the recording. In sleep scoring for example, most stages are characterized by the occurrence of specific brain rhythms in a given frequency range. The sleep stage N2 is for instance characterized by so-called sleep spindles with frequencies located between $12 - 15Hz$. The predominance of certain rhythms is well captured by the power spectrum density (PSD) of a signal, which shows peaks in these frequency bands as depicted in Figure 1. The `FrequencyShift` augmentation, proposed in [25], shifts the PSD of the signal by a factor Δf. The rationale behind this augmentation is that EEG recordings have a strong inter-subject variability. Consequently, the locations of frequency bands with a high power density are likely to be slightly different between two subjects even though they correspond to the same category of cerebral activity, as shown in Figure 1.

The operation performed on a signal X is the following,

$$\texttt{FrequencyShift}[X](t) := \mathrm{Re}(X_a(t) \cdot e^{2i\pi\Delta f \cdot t}),$$

with $X_a = X + j\mathcal{H}(X)$ being the complex analitic signal corresponding to X, where $\mathcal{H}(X)$ is the Hilbert transform of the signal. Since EEG recordings produce real valued signals, their Fourier transform has Hermitian symmetry, which means that $\mathcal{F}[X](\omega) = \mathcal{F}[X]^*(-\omega)$, where $\mathcal{F}[X]^*$ denotes the complex conjugate of $\mathcal{F}[X]$. As a result, shifting the Fourier transform of the signal towards high or low frequencies would break this symmetry. To avoid this, the shift is performed on the complex analytic signal associated to X. Then taking the real part allows to recover the shifted signal. The strength of this transformation can be set through the parameter Δf that controls the shift of the PSD. This parameter is randomly sampled with uniform probability in an interval [-Δf_{max}, +Δf_{max}] each time an EEG window is augmented.

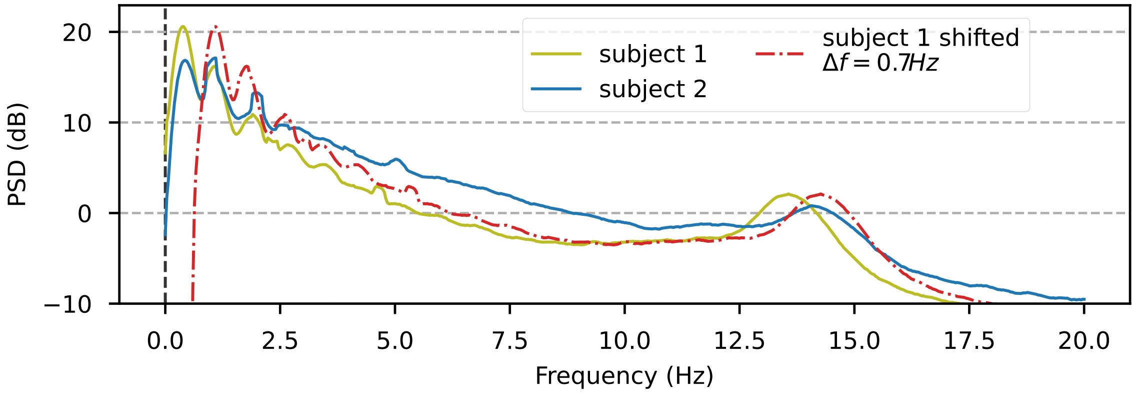


Figure 1: Averaged power spectrum density of windows corresponding to the sleep stage N2, for two different subjects from the *SleepPhysionet* dataset. The red dash-dot curve corresponds to the recording of subject 1 transformed using the `FrequencyShift` augmentation. It allows to translate the PSD peak close to subject 2.

Fourier transform surrogate As aforementioned, the information contained in the distribution of power over frequencies is instrumental in EEG classification tasks. Taking this fact into account, one may wonder whether it is possible to transform a recording in a way that leaves the PSD unchanged while generating a different waveform. This is precisely what is achieved with the Fourier Transform surrogates method [11]. This augmentation generates so called surrogates by randomizing the phase of the Fourier coefficients of a recording, while leaving the Fourier amplitudes unchanged. As a result, the PSD of the data is invariant to this augmentation but the whole temporal signal is modified, since the phase diagram is transformed.

More concretely, this is performed by adding random noise to the phase of the Fourier coefficients of a signal and applying the inverse Fourier transform,

$$\mathcal{F}[\texttt{FTSurrogate}(X)](\omega) = \mathcal{F}[X](\omega)e^{i\Delta\varphi},$$

where $\mathcal{F}$ is the Fourier transform operator[1], ω a frequency and $\Delta\varphi$ a random phase disturbance. In our implementation a value of $\Delta\varphi$ is uniformly sampled in an interval $[0, \Delta\varphi_{max}]$ for each frequency ω, where $\Delta\varphi_{max} \in [0, 2\pi)$ is a hyperparameter. This method is based on the assumption that EEG signals are generated by stationary linear random processes [11]. As such, they must be uniquely described by the amplitudes of their Fourier coefficients and must have random phases in $[0, 2\pi)$. Subsequently, changing coefficients phase of such a process allows to generate a new independent signal following the same distribution. This assumption might seem implausible since nonlinearities are introduced at the cellular level in the brain by the dynamical behaviour of individual neurons which generates action potentials when they receive a stimuli that outreaches a threshold saturation. Nevertheless, it has been shown that this hypothesis can be verified experimentally [26]. Plausible arguments used to advocate it are the huge number of neurons included in an EEG recording, the complicated structure of the brain and the possible blur of dynamical structures due to the different conductivities of the skull and other intermediate tissues.

Band-stop filter To the same extent, the band-stop filter augmentation transforms the signal in the frequency domain by filtering out a given frequency band. This augmentation aims to prevent machine learning models from overfitting on subject specific features or to relying too much on few narrow frequency regions. This transformation hence aims to help the model to learn more robust representations. It was first introduced in the field of self-supervised contrastive learning [12], [27].

The implementation of this augmentation uses the finite impulse response notch filter from *MNE-python* [28]. For each augmented EEG window, the center of the frequency band is randomly picked between 0 and the Nyquist frequency of the signal, using a uniform distribution. The width of the filter is a parameter that must be adjusted by the user.

3.2 Invariances

Frequency shift This augmentation moderately modifies the representation of the signal in the frequency domain. Hence, it also changes globally its representation in the time domain. As seen in Figure 1, by associating the same label to signals that have a slightly shifted PSD, this augmentation encourages the model to become invariant regardless of small inter-subject differences in the distribution of power over frequencies.

Fourier transform surrogate This augmentation does not modify the amplitudes of the Fourier coefficients, consequently the power spectrum density of the signal remains unchanged, which preserves the frequency-bands power ratios. Meanwhile, the changes applied to the phase diagram trigger a global change of the signal's representation in the time-domain. As a result, this augmentation encodes the invariance of the signal regarding changes in the phase diagram of its Fourier transform, under the assumption that the signal has been generated by a stationary linear random process. Thus, a model trained with `FTSurrogate` is expected to have a higher misclassification rate on sleep stages such as N3, which are characterized by non-stationarities and transient patterns, hence violating the previous assumption. This is illustrated on Figure 2, which shows that characteristic patterns such as K-complexes are erased by this operation.

Bandstop filter By removing random frequency bands, this transformation can be compared to a dropout layer [10] that prevents the model from relying on specific frequencies. Hence, this transformation enforces the invariance of the decision function to the disappearance of certain frequency

[1] Approximated with the Fast Fourier Transform **fft**

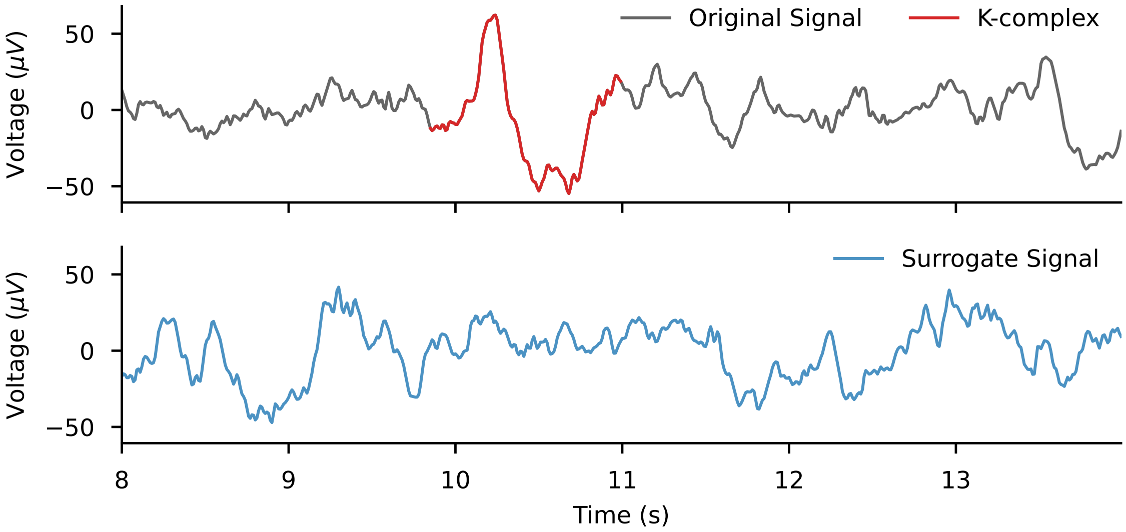


Figure 2: Effect of `FTSurrogate` on transient patterns. The original extract from a window scored as N2 presents a highly localized K-complex, whereas the surrogate does not.

bands. Additionally, since this transformation acts on the Fourier transform of the signal, it globally affects the time domain representation of the EEG recordings.

3.3 Experimental results

3.3.1 Parameters selection

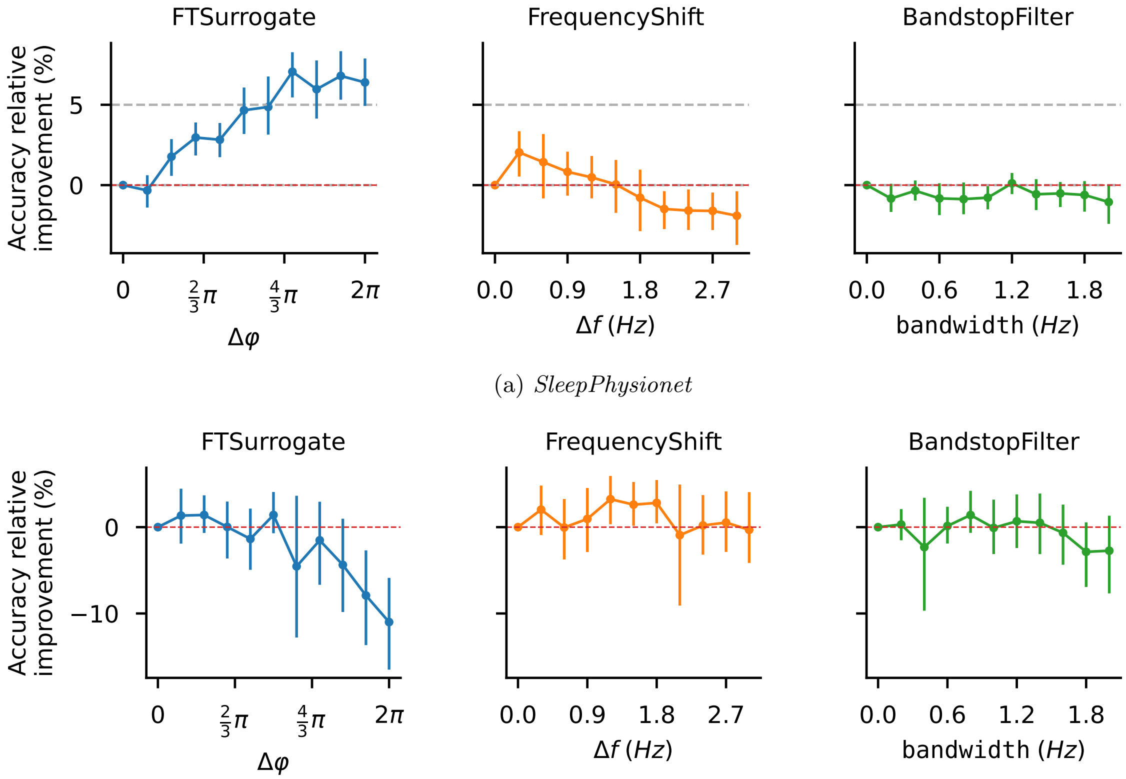
The first parameters selection experiment for frequency domain augmentations aims at determining the optimal value for the parameters listed in Table 1.

SleepPhysionet The results on the sleep staging task presented in Figure 3 and Table 1, reveals that the optimal strength varies significantly from one transformation to the other. The `FTSurrogate` augmentation benefits from the larger possible range of $\Delta\varphi$ values, corresponding to a completely random phase selection within the interval $[0, 2\pi)$. On the contrary, `FrequencyShift` works better for small Δf values. An interpretation for this result could be that small frequency shifts allow to capture the inter-subject variability whereas larger values mix-up the frequency bands characterizing different classes. Finally, `BandstopFilter` shows better results for a bandwidth of $1.2 Hz$. Surprisingly, this is the only value tested which does not degrade the classification performance for sleep staging, despite the small bandwidth values consider in an attempt to avoid erasing crucial information.

BCI IV 2a The results on the *BCI IV 2a* presented in Figure 3b suggest several differences. First, the results for `BandstropFilter` are less interpretable. This is probably due to the sampling frequency of $250 Hz$, much higher than the 100Hz used for *SleepPhysionet*. As a result, the center of frequency bands that are filtered out are picked in the interval $[0, 125] Hz$ and most of the time, frequencies involved have already been filtered out by the upstream lowpass filter (cutoff frequency of $38 Hz$). For `FrequencyShift` as well, no clear trend can be identified. This transformation is probably irrelevant for BCI since models are trained and evaluated on the same subject. Finally, the `FTSurrogate` augmentation seems to benefit from higher $\Delta\varphi$ magnitudes as for the sleep staging task.

Augmentation	Parameter	Interval	Unit	Best value (sleep staging)	Best value (BCI)
`FrequencyShift`	Δf	$[0, 3]$	Hz	$0.3Hz$	
`FTSurrogate`	$\Delta\varphi_{max}$	$[0, 2\pi)$	rad	2π	
`BandstopFilter`	bandwidth	$[0, 2]$	Hz	$1.2Hz$	

Table 1: Adjustable parameter of each frequency domain augmentation.



(a) *SleepPhysionet*

(b) *BCI IV 2a*

Figure 3: Frequency augmentations parameters selection on the *SleepPhysionet* (a) and *BCI IV 2a* (b) datasets. The same model was trained on respectively 350 and 60 windows using augmentations parametrized with 10 different linearly spaced values. After the training, the validation accuracy was compared with the validation accuracy obtained by a model trained without data augmentation. 10-fold cross-validation is used to obtain the error bars.

3.3.2 Learning curves

SleepPhysionet As expected, the first learning curve experiment depicted in Figure 4a reveals that data augmentation methods are all the more helpful in low data regimes. It helps to mitigate the lack of data by artificially increasing the training set size. For example, a model trained with `FTSurrogate` on a fraction of 2^{-8} achieves performances comparable with a model trained without augmentation on 4 times more data. This first observation illustrates the interpretation of data augmentation as a regularization method, as it prevents the model from overfitting on a limited number of training examples. Moreover, the learning curves of models trained with `FrequencyShift` and `FTSurrogates` are high above the baseline, thus evidencing that these augmentations can improve the performance on a sleep staging task. This statement reveals these transformations preserve the relevant information of EEG signals for such a task, unlike `BandstopFliter` which appears to be detrimental for sleep staging. Finally, a clear ranking stands out, revealing that the `FTSurrogate` augmentation yields the best improvements with a balanced accuracy gain of up to 5% in low data regimes.

BCI IV 2a Running the same experiment on the *BCI IV 2a* dataset yields quite different results as shown in Figure 4b. The learning curve for the `FrequencyShift` augmentation is very similar to the baseline on the BCI task while it significantly improved the classification on *SleepPhysionet*. This observation supports our guess that `FrequencyShift` encodes the invariance to the inter-subject variability. Indeed, this invariance is useless on the BCI cross-session task since the model is trained and evaluated on the same subject. Nonetheless, Figures 4a and 4b have in common the fact `FTSurrogate` outperforms other frequency domain augmentations. For both classification tasks, a critical part of the information seems to lie in the frequency domain.

3.3.3 Per class analysis

SleepPhysionet Taking a closer look at the performance per class presented in Figure 5a helps to reckon the variability of data augmentation effects depending on the sleep stage. First, it can be seen that all frequency data augmentation methods only produce marginal improvements for sleep stages W and N3. This results can be interpreted in light of the performance of the baseline model presented in Figure 6a. Indeed, the highest F1-scores are reached for these classes, the model seemingly extracts relevant features from the inputs and the induced invariances hardly helps for this classification task. This experiments also discloses that frequency domain data augmentations significantly improve the results for the sleep stage REM. This result might not be only associated to the easiness of the task, since the baseline performs equally well on the N2 stage, which benefits less from data augmentation. It thus seems that the invariances encoded by such augmentation are specifically relevant for this class. Finally, the larger error bars for the REM stage may be associated with a higher inter-subject variability that might stem from the eye movement artifacts.

BCI IV 2a To the same extent, in Figure 5b the small improvements brought by frequency domain augmentations on the classification of right hand movements might be due to the already high performance reached by the baseline as shown in Figure 6b . It also appears that all three augmentations have almost the same effects on each class, aside from the variations related to the performance of the baseline. This fact points to similarities in frequencies that characterizes the brain activities for these 4 motor imagery tasks.

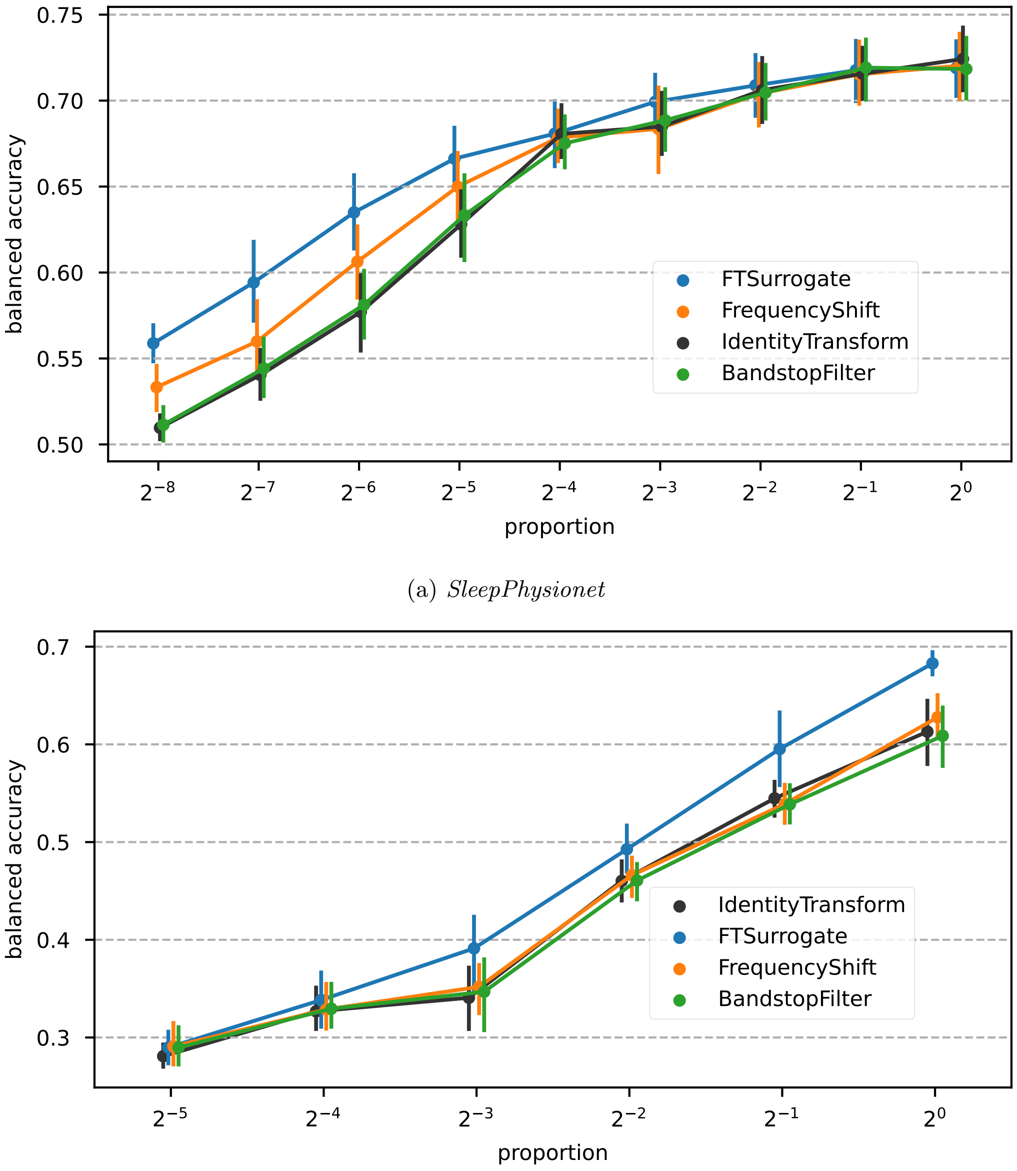
4 Time domain augmentations

4.1 Rationale of the transformations

Gaussian noise EEG recordings often present a low signal to noise ratio. Furthermore, signal and noise frequency bands often overlap, making them even harder to disentangle **cohen˙analyzing˙2014**. Low pass filtering in hence a standard preprocessing step, aiming to filter some noise out. Such filtering is never perfect and telling apart signal and noise is not always obvious. While some researchers mostly focus on frequencies below $30Hz$, considering higher frequencies as noise, some others center their attention on high frequency oscillations ($\geq 80Hz$) **frauscher˙high-frequency˙2017**. The impossibility to get rid of the noise inherently present in brain signals motivates the introduction of the `GaussianNoise` augmentation, which mimics this feature.

This transformation consists in adding Gaussian white noise N_G to the original signal X,

$$\texttt{GaussianNoise}[X](t) = X(t) + N_G, \quad N_G \sim \mathcal{N}(0, \sigma)$$



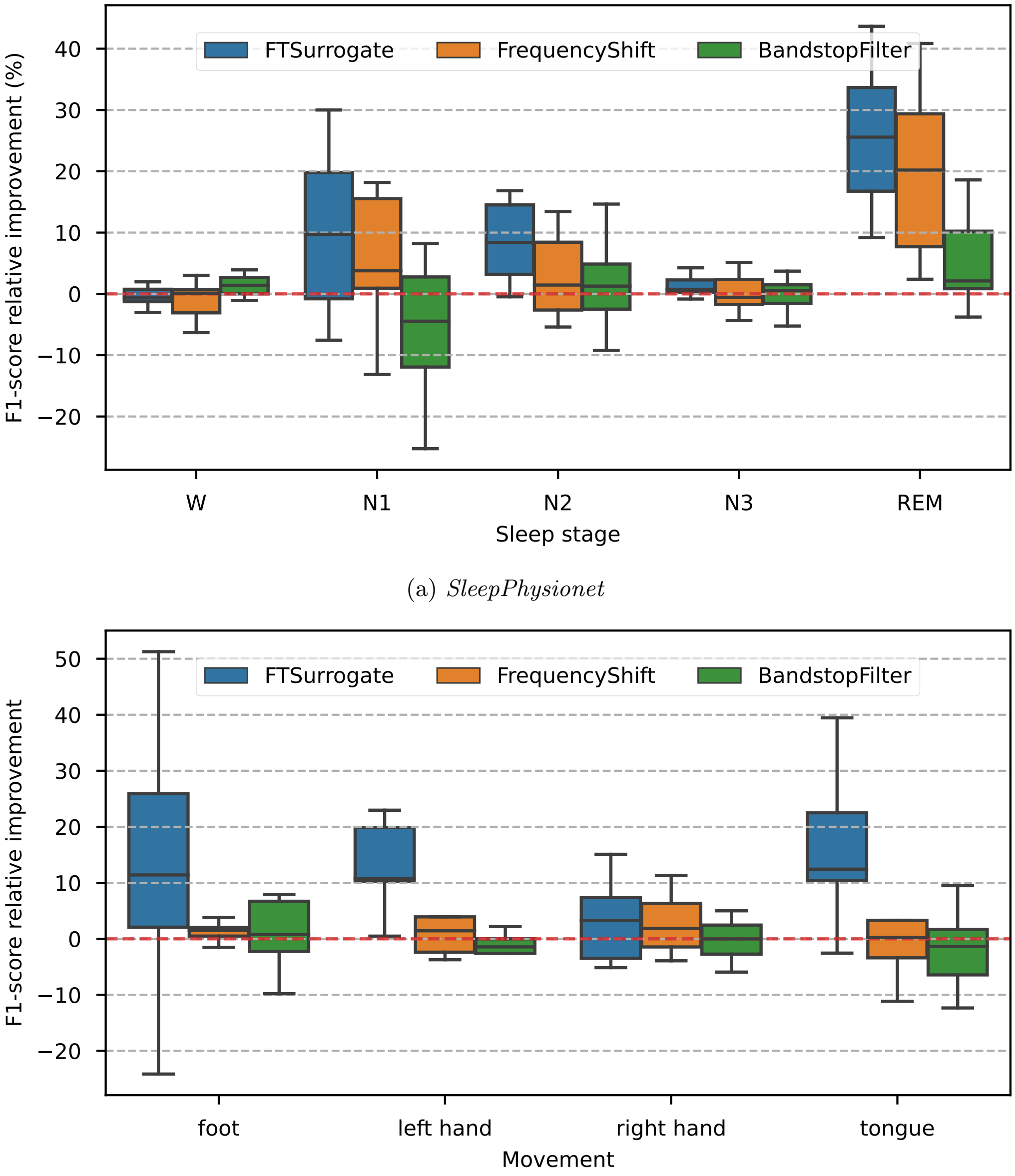
(a) *SleepPhysionet*

(b) *BCI IV 2a*

Figure 4: Learning curves for frequency domain augmentations along with the baseline trained with no augmentation. For each augmentation, the same model [19], [22] has been trained on 8 fractions of increasing size of the dataset (a *SleepPhysionet* and b *BCI IV 2a*. After each training, the balanced accuracy score on the test set is reported. 10-fold cross-validation is used to obtain the error bars.

By adding the same amount of power to all frequencies the `GaussianNoise` transformation affects more frequency bands that originally had smaller power values. It mostly preserves the power band ratios at lower frequencies which are instrumental in many EEG decoding tasks.

Smooth time mask As described in the *AASM* scoring manual [16], sleep stages are most often characterized by the global information contained in a time window. For example, the sleep stage N1 is scored when more than 15 seconds ($\geq 50\%$) of a time window is dominated by theta activity



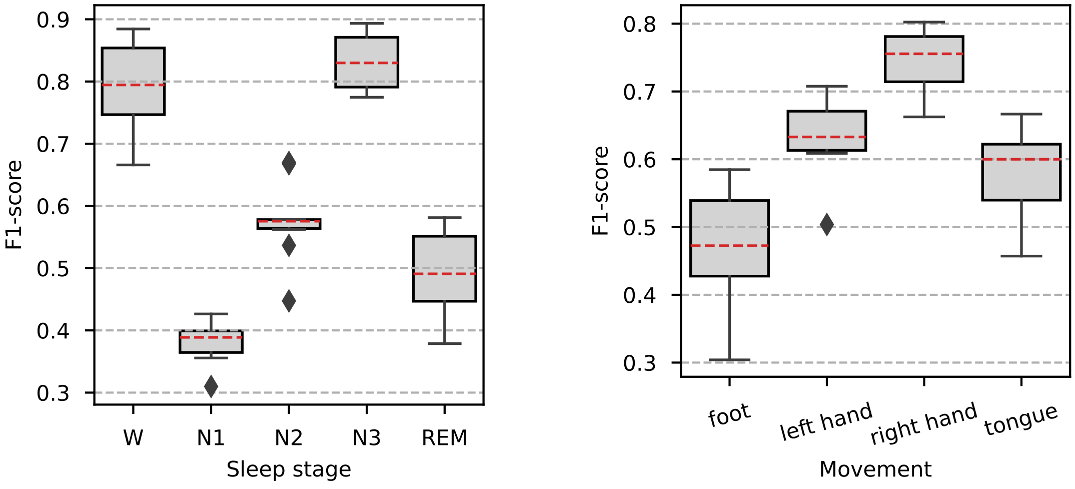
(a) *SleepPhysionet*

(b) *BCI IV 2a*

Figure 5: Relative improvement per class of the F1-score due to frequency domain transformations in comparison to the identity transformation. The scores have been computed after a training on 180 for *SleepPhysionet* and 230 for *BCI IV 2a* time windows, which respectively correspond to a proportion of 2^{-7} and 1. The boxplots give statistics on the 10-fold cross validation.

$(4-7Hz)$. The representations learned by the model should thus encapsulate the global information of the signal and avoid to rely on transient patterns. Consequently, we would expect two almost identical EEG windows differing only for a few seconds to be very close in the representation space [27]. The `SmoothTimeMask` augmentation is designed with this in mind. It consists in replacing by zeroes a portion of length Δt, that starts from a randomly sampled instant t_{cut}. The computation is carried out by multiplying the signal by a mask

$$\texttt{SmoothTimeMask}[X](t) := \left[\frac{1}{1+\exp\left(-\lambda(t-t_{cut}-\Delta t)\right)} + \frac{1}{1+\exp\left(\lambda(t-t_{cut})\right)}\right] X(t),$$



(a) *SleepPhysionet*

(b) *BCI IV 2a*

Figure 6: Baseline per class F1-score. The scores have been computed after a training on 180 for *SleepPhysionet* and 230 for *BCI IV 2a* time windows, which respectively correspond to a proportion of 2^{-7} and 1. The boxplots give statistics on the 10-fold cross validation.

which is made out of two opposing sigmoid functions. This allows to set the signal smoothly to zero as shown in Figure 7 using a smoothness parameter, and avoids creating discontinuities which would not be naturally found in EEG recordings.

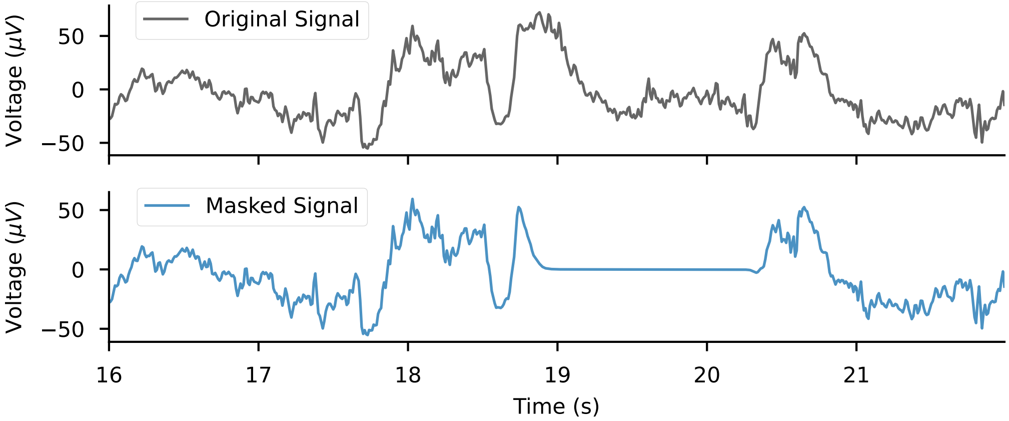


Figure 7: Effect of `SmoothTimeMask` on a time window from the *SleepPhysionet* dataset. The mask length is 1.6*sec* and the transition between unchanged parts and the masked portion is smooth.

Sign flip The electrical potentials measured during an EEG recording stem from the propagation of action potentials along axons when a neuron is active. This neuronal activity generates an intra-cellular and an extra-cellular electric field. When neighboring neurons are active at the same time, their extra-cellular fields sum up to a local electric field potential. Depending on the geometry of the involved neurons this local field can be measured and an intuitive representation of the source (the assembly of neurons) is a dipole, defined by a vector. For most analysis, the strength (norm) and the location (origin) are considered whereas the direction is hardly used. Though, the direction is responsible for the sign of the potential measured by the EEG device. Our guess is that changing the sign of EEG channels will preserve the instrumental information contained in the strength and location of the dipole. Moreover, the designation of the reference electrode and measurement electrode results from an arbitrary choice but a change in this assignation would also result in a recording of opposite sign.

We therefore introduce the `SignFlip` augmentation which flips the sign of each EEG channel.

$$\texttt{SignFlip}[X](t) := -X(t)$$

This transformation preserves most of the topological properties of the electric field potential since it merely corresponds to a swap between the head and tail of the dipole.

Time reverse Sleep stages are mainly characterized by frequency-domain information [16]. Considering that the orientation of the time axis has no effect on the power spectrum density of the signal, it can be hypothesized that flipping the time axis preserves most of the information while generating a new input. The augmentation is implemented as follow:

$$\texttt{TimeReverse}[X](t) := X(t_{max} - t),$$

where t_{max} is the length of a time window. Regarding the time domain, a large part of the information is remains since symmetric waveforms are merely shifted along the time axis, only non-symmetric patterns are distorted as illustrated in Figure 9. Though, such patterns are outnumbered by symmetric *e.g.,* sleep spindles, slow waves... **malhotra˙sleep˙2014**

4.2 Invariances

Gaussian white noise This augmentation encodes the invariance of EEG recordings regarding the acquisition noise. By adding the same amount of power to all frequencies, the addition of white noise hides the information contained in high frequency bands, which have small power. Indeed, they become indistinguishable as depicted in Figure 8. From this perspective, the effect of this transformation is analogue to a low-pass filter where the parameter σ stands for a cut-off frequency (*cf.* Figure 8).

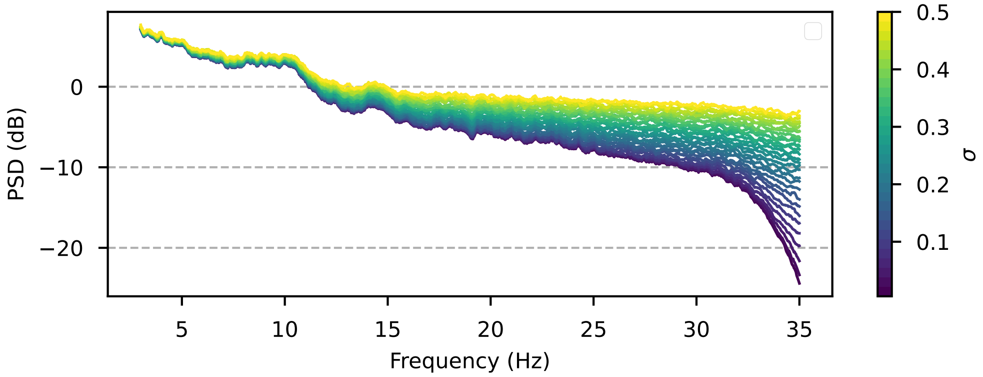


Figure 8: Effects of the addition of `GaussianNoise` on the PSD. Power spectra were averaged over N1 windows for one night of sleep from the *SleepPhysionet* dataset. In polysomnographies, the power globally decreases as the frequency increases. Consequently, the `GaussianNoise`, which adds a constant amount of power across all frequencies, has a greater impact on higher frequencies. As we increase σ, a greater portion of the signal is hidden by the added noise.

Smooth time mask By masking part of the signal, we assign the same label (*e.g.,* sleep stage or action) to windows differing in the time domain only inside a short time span. This transformation hence promotes such an invariance, and intends to teach the feature extractor to be more robust to differences at this time scale.

Sign flip Based on the physics of EEG, this augmentation is equivalent to a polarity reversal. A model trained with it will hence supposedly learn to extract representations that are invariant to the polarity of brain sources, as well as to the arbitrary choice of reference electrodes.

Time reverse The `TimeReverse` augmentation encodes the invariance to the direction of the time axis. This augmentation thus leaves the frequency domain undisturbed. Furthermore, the proportions of certain rhythms within a window are also preserved. For example, if more than 50% of a 30s window is dominated by theta waves (as in N1 sleep stages), it will also be the case for its time reversed counterpart. This transformation also implies partial invariance to the position of temporal patterns within the EEG window, as well as to the orientation of asymmetric patterns. As a case in point, for sleep staging, the sleep stage N2 is scored when more than two occurrences of K-complexes are observed throughout the stage [16]. `TimeReverse` will change the position of K-complexes within a window and invert its orientation given that it is an asymmetric pattern. One may wonder whether this is beneficial for learning to score this sleep stage or not. The variability of the effects across classes will be investigated in Section 4.3.3.

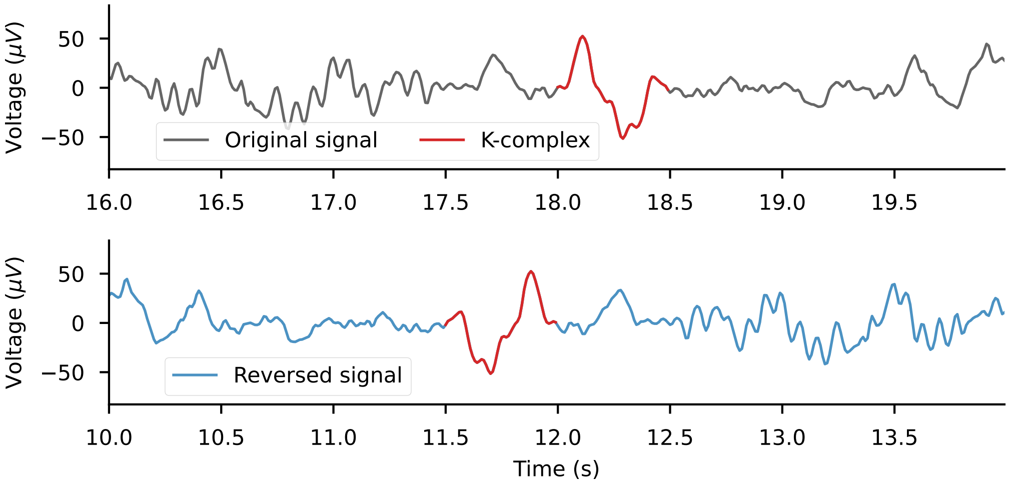


Figure 9: `TimeReverse` augmentation on an EEG signal from the *SleepPhysionet* dataset. A large part of the signal is not deeply affected. Wave patterns are merely translated along the time axis. Meanwhile some specific EEG patterns have asymmetric shapes such as K-complexes, which are reversed by this transformation.

4.3 Experimental results

4.3.1 Parameters selection

As in Section 3.3.1, here we investigate the effect of the strength of previously introduced transformations on the tasks considered. As shown in Table 2, the intensity of `GaussianNoise` is controlled by its standard deviation σ, while `SmoothTimeMask` is governed by the length of the mask Δt. Note that there is no notion of strength or intensity for `TimeReverse` and `SignFlip`, which are hence not studied in this subsection.

SleepPhysionet Similarly to what can be observed for frequency transformations, the impact of the intensity is quite different between `SmoothTimeMask` and `GaussianNoise`, as illustrated in Figure 10. While increasing the strength of `SmoothTimeMask` seems beneficial, no clear trend is observed for `GaussianNoise`. For `SmoothTimeMask`, we restricted our experiment to masks of less than two seconds to avoid removing too much crucial information from the signal. It seems that the augmentation is more efficient with masks of 2 seconds. However, `GaussianNoise` systematically degrades the performance of sleep staging. The reduction of the accuracy is minimal when σ equals 1.4.

BCI IV 2a On the one hand, the gird search on the *BCI IV 2a* presented in Figure 10b dataset share similarities with its counterpart on the *SleepPhysionet* dataset (Figure 10a). In both cases, increasing the mask length enhances the performance whereas larger values of σ steadily degrades the accuracy. But on the other hand, `SmoothTimeMask` appears to be much more useful on the BCI task since it produces relative improvements of up to 20% when 60 windows are used for the training.

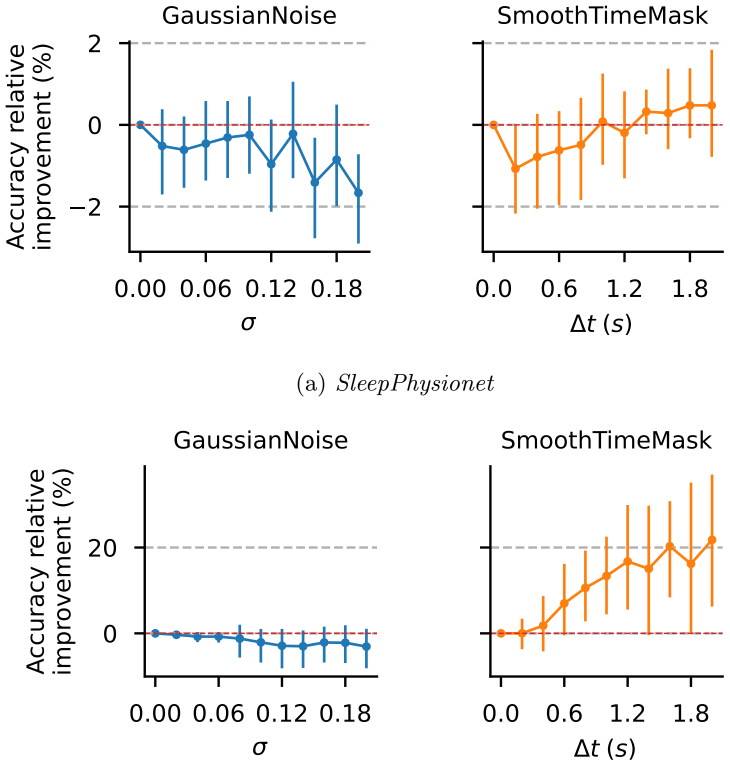
Augmentation	Parameter	Interval	Unit	Best value (sleep staging)	Best value (BCI)
`GaussianNoise`	σ	$[0, 0.2]$	∅	0.12	0.02
`SmoothTimeMask`	Δt	$[0, 2]$	s	$2s$	2s

Table 2: Potential and selected values for the adjustable parameter of each time domain augmentation.

4.3.2 Learning curves

SleepPhysionet The learning curves for the sleep staging task presented in Figure 11a reveals that the most significant improvements are brought by `SignFlip` and `TimeReverse` with the latter slghtly outperforming the former. As expected from the results of the previous experiment, `GaussianNoise` and `SmoothTimeMask` have a negligible effect on this sleep staging task and perform on par with the baseline model. These results suggest that symmetries are notably relevant invariants for the sleep classification task. They preserve both the frequencies and some of the transient patterns occurring during polysomnographies such as sleep spindles which presents a symmetry both along the x and y axis. This observation also supports the claim that sleep scoring heavily relies on frequency domain features since the two augmentations that leave the power spectrum density of the signal unchanged outperform the others.

BCI IV 2a The Figure 11b that contains the learning curve plots for the *BCI IV 2a* dataset reveals that `SmoothTimeMak` is the most suited time domain augmentation for the BCI task. The invariance to time masking is particularly relevant for motor imagery where subjects are asked to mentally simulate the same physical action during the whole trial which should intuitively generate a brain signal that is invariant to translations along the time axis. Another striking observation has to do with the learning curves of `SignFlip` and the baseline that are virtually the same while `SignFlip` is the second most efficient augmentation for the sleep staging task. For the BCI competition, EEGs were recorded using a 22 electrodes montage, using with the left mastoid as reference for all of them **c˙brunner˙bci˙2008**. The high spatial resolution of the device used is most likely sufficient to capture the invariance of the signal to the polarity of brain sources and to the arbitrary choice of reference and measurement electrodes. Indeed, considering a brain source as a dipole, chances are that the montage will present one electrode located at the head and another at the tail of this dipole. Consequently one channel



(a) *SleepPhysionet*

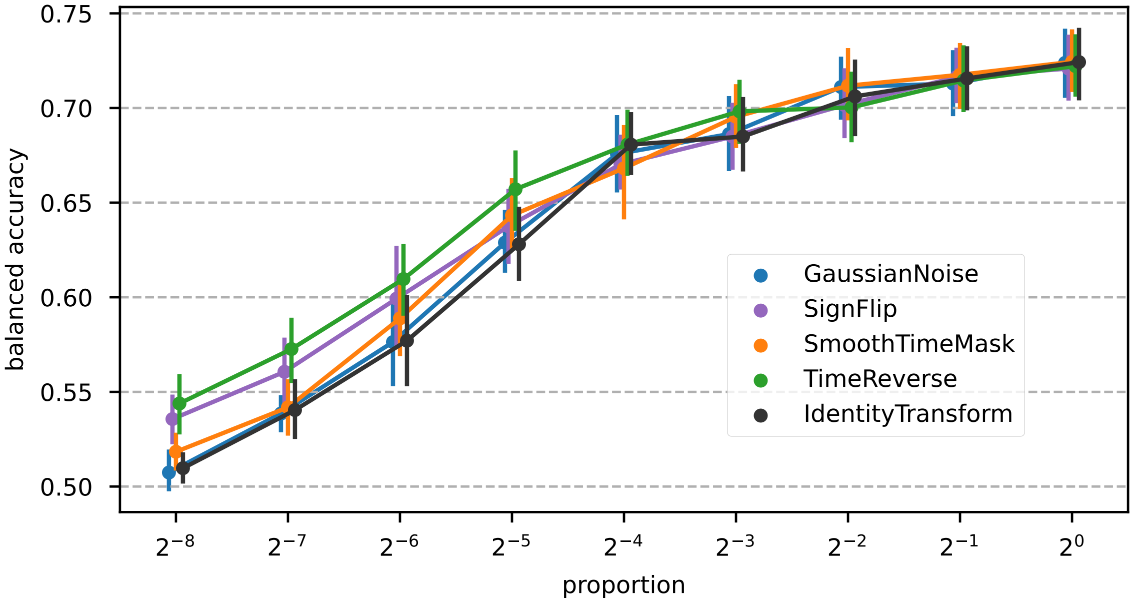
(b) *BCI IV 2a*

Figure 10: Time augmentations parameters selection on the *SleepPhysionet* (a) and *BCI IV 2a* (b) datasets. The same model was trained on respectively 350 and 60 windows using augmentations parametrized with 10 different linearly spaced values. After the training, the validation accuracy was compared with the validation accuracy obtained by a model trained without data augmentation. 10-fold cross-validation is used to obtain the error bars.

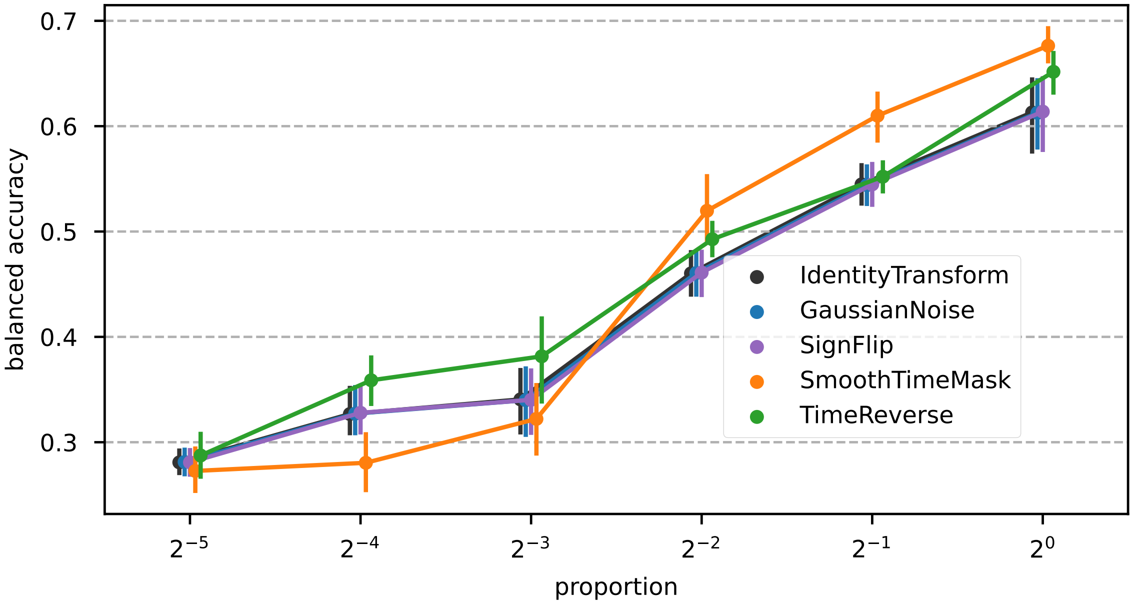
should look alike the signal that would have been obtained after applying `SignFlip` to the other, thus making the augmentation needless.

4.3.3 Per class analysis

SleepPhysionet Figure 12 introduces the results per class of the time domain augmentations for the sleep staging task. For the REM stage, a greater improvement is reached compared to frequency domain augmentations. This stage is characterized by low amplitude mixed frequency and bursts of eye activity. The scoring thus relies heavily on amplitudes which are mostly preserved with time domain augmentations (except for `SmoothTimeMask` which performs the worst among them). `SignFlip` especially yields the best results for this class whereas it is not the best on average, this augmentation can be interpreted as an absolute value layer added upstream of the model which is well suited in this case.



(a) *SleepPhysionet*



(b) *BCI IV 2a*

Figure 11: Learning curves for time domain augmentations along with the baseline trained with no augmentation. For each augmentation, the same model [19], [22] has been trained on 8 fractions of increasing size of the dataset (a *SleepPhysionet* and b *BCI IV 2a*. After each training, the balanced accuracy score on the test set is reported. 10-fold cross-validation is used to obtain the error bars.

5 Spatial domain augmentations

In this section we study augmentations acting on the sensors spatial positions. Indeed, most models used for EEG decoding on multi-channels recordings make use of spatial filters [19], [22] to extract

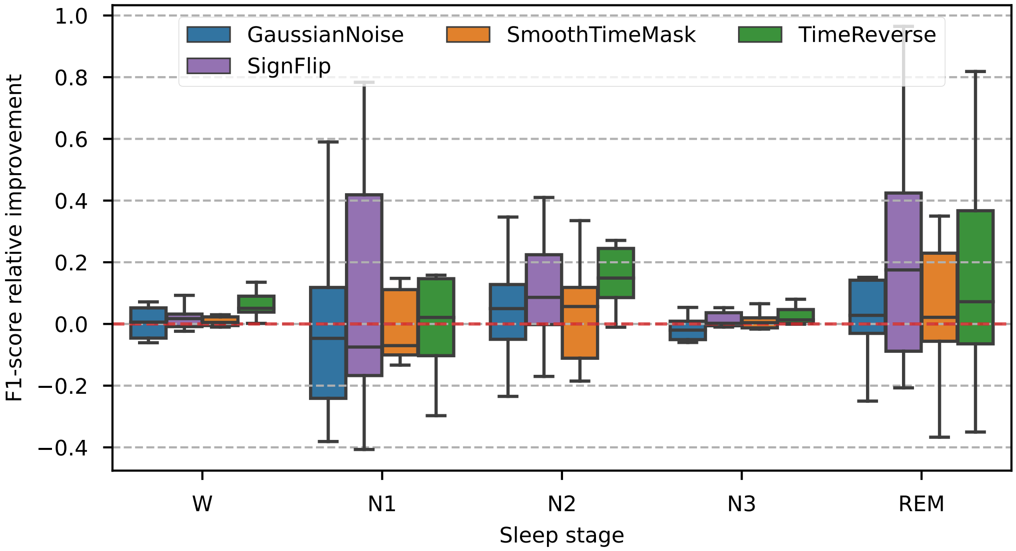


Figure 12: Relative improvement per class of the F1-score due to time domain transformations in comparison to the identity transformation. The scores have been computed after a training on 180 time windows, which corresponds to a proportion 2^{-7} of the *SleepPhysionet* dataset. After each training, the balanced accuracy score on the test set is recorded. 10-fold cross-validation is used to obtain the error bars.

relevant representations. Hence, such models should intuitively benefit from this category of augmentations.

5.1 Rationale of the transformations

Channels symmetry Several brain activities that are monitored using EEG involve a sagittal plane symmetry. For example, it has been evidenced that tongue movements stem from an activation of the primary and supplementary sensorimotor areas without any significant lateralization **watanabe˙human˙2004**. Taking this fact into consideration, the `ChannelsSymmetry` augmentation permutes the order of EEG channels in order to simulate a swap between EEG sensors placed on the right and left hemispheres. However, since Penfield and Jasper's experiments in 1954 **penfield˙epilepsy˙1954**, it is well known that the contraction of a muscle on one side of the sagittal plane results from a brain activation in the opposite hemisphere. Consequently, applying the `ChannelSymmetry` transformation is expected to be harmful in such cases (*i.e.,* left and right hand movements in the BCI task), since it would destroy crucial information. This point will be further explored in Section 5.3.3

Channels dropout A major hurdle for the analysis of EEG signals is the inconsistent quality of EEG channels throughout a recording. For example, in polysomnography, changes in the subject's position during sleep might result in a loss of contact between several electrodes and the skull. Furthermore, EEG recordings are not always recorded in laboratory conditions. The spread of mobile wearable EEG devices raises new challenges, as they are more prone to noise and missing channels **banville˙robust˙2021**. Finally, the EEG and machine learning communities consider with great interest the question of transferability across subjects and datasets. Though, a major problem arises

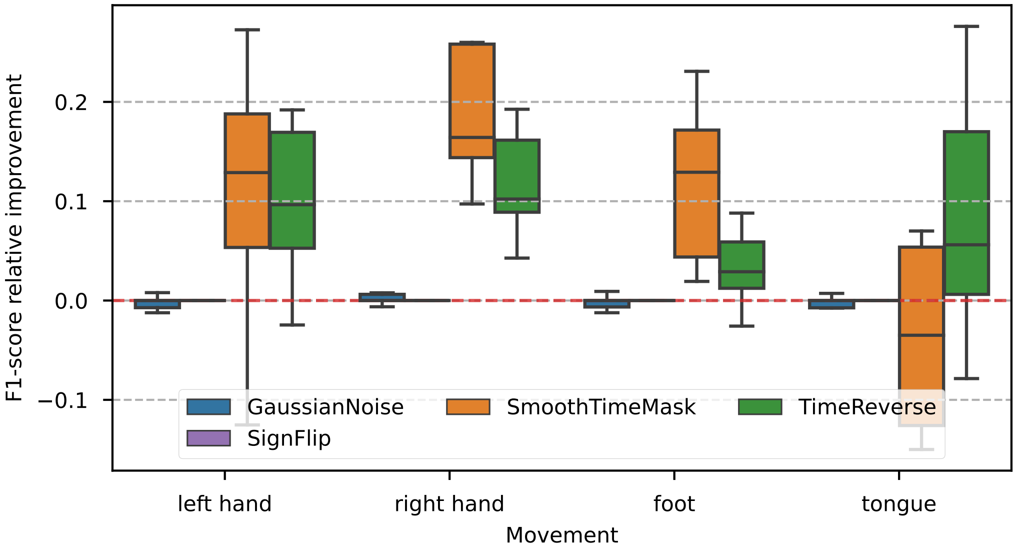


Figure 13: Caption

from the differences in collection protocols and EEG devices, which often produce recordings with inconsistent numbers of channels or channels ordering. As a result, the transferability of models is hampered by the high specificity to the dataset used for the training. To tackle these issues, the `ChannelsDropout` augmentation, initially proposed in [13], randomly sets each channel of the EEG recording to zero with a given probability. More precisely, for $X \in \mathbb{R}^{C \times T}$ an EEG window of T samples collected on C channels, the augmented signal takes the form

$$\texttt{ChannelsDropout}[X]_i := d_i \cdot X_i, \qquad d_i \overset{i.i.d}{\sim} \mathcal{B}(p_{drop}),$$

where $\mathcal{B}(p)$ is a Bernoulli distribution of probability p, and X_i denotes the i^{th} row of X corresponding to channel $i \in \{1, \dots, C\}$. As the widely used dropout layer [10], it prevents the model from relying too heavily on a given input channel, which could lead to overfitting and poor generalization on data from unseen subjects or different datasets.

Sensors rotations The experimental protocols to aquire brain signals also suffer from reproducibility problems. During the acquisition of EEGs, the device is prone to shifts as the subject moves and the setup is likely to be different from one experiment to the other [14]. Furthermore, the inter-subject anatomical variability contributes to differences in the electrode's positions. To tackle, these problems `SensorsRotation` augmentation [14] simulates the natural position variations expected during the acquisition. To achieve this, the electrical potentials measured during an EEG recording are interpolated in between sensors using their 3D coordinates in a standard 10-20 montage. While in [14] a radial-basis functions interpolator is used, we decided to implement this transformation using spherical splines **perrin˙spherical˙1989**. The interpolated signals hence approximate what would have been recorded with a device slightly rotated along a given axis (x, y or z).

ChannelsShuffle For several EEG classification tasks such as sleep staging, the localization of the cerebral activity is not a deciding feature. Indeed, sleep experts hardly consider the sensors position of the channel they observe (*e.g.,* Fpz-Oz) but rather rely on the waveforms (*e.g.,* theta activity, K-complex). Therefore, in the context of sleep scoring, shuffling the channels preserves the crucial information that lies in waveforms while it mixes up the spatial information. If this augmentation seems well suited for the aforementioned task, it is expected to be less efficient in a context where sensors positions have a greater weight. For example, BCI where the classification of right- and left-hand movements heavily relies on the active hemisphere.

5.2 Invariances

Channels symmetry The `ChannelsSymmetry` augmentation teaches the model to become invariant to the position of sensors with regard to the sagittal plane. It thus takes for granted that brain dynamics are not lateralized, which is not always true especially for BCI tasks (*e.g.,* left- and right-hand movements). Though, this assumption might help to extract representations relying on the overall brain activity.

Channels dropout By reproducing defective input channels which naturally arise in EEG recordings, this augmentation method aims at teaching models to become invariant to it. It has initially been designed to enhance the transferability of models across datasets with inconsistent number of channels [13]. Learning this invariance should ensure that the model considers all the available channels instead of relying on a single one that might not be present in another dataset. This might also come in handy when the number of usable channels is inconstant *e.g.,* during polysomnography where changes in the subject's position might result in a loss of contact between several electrodes and the skull.

Sensors rotations It encodes the invariance to shifts in the EEG device on the patient's head, during and between different recordings. It might also induce the invariance to variations in skull anatomy between subjects, which results in electrodes positions variations.

Channels Shuffle Commonly used deep learning architectures extract the spatial information from the data through spatial filters [22]. The channels are usually ordered following internationally standardized 10-20 system, each line of the input matrix thus correspond to a position on the subject's skull. Shuffling the channels of an EEG recording mixes up this spatial information and hence, prevents the decision function from making use of it. Consequently, the `ChannelsShuffle` augmentation induces the invariance to the absolute and relative positioning of EEG sensors.

5.3 Experimental results

5.3.1 Parameters selection

Concerning spatial domain augmentations, the parameters listed in Table 3 that control the strength of the transformations are: the probability to drop channels p_{drop}, the probability to shuffle channels $p_{shuffle}$ and the angle of rotation θ_{rot} respectively for `ChannelsDropout`, `ChannelsShuffle` and `SensordRotations`. The `ChannelsSymmetry` augmentation has a strength that cannot be adjusted.

Augmentation	Parameter	Interval	Unit	Best value (sleep staging)	Best value (BCI)
`ChannelsDropout`	p_{drop}	$[0, 1]$	∅	0.3	
`ChannelsShuffle`	$p_{shuffle}$	$[0, 1]$	∅	1	
`SensorsRotations`	θ_{rot}	$[0, 15]$	*degree*	13.5	

Table 3: Potential and selected values for the adjustable parameter of each spatial domain augmentation on the *SleepPhysionet* dataset.

SleepPhysionet The results of the grid search for the parameters of spatial augmentations are presented in Figure 14. Regarding the sensor rotations, a preliminary experiment revealed that rotations with too large angles (15°) steadily degrades the performance. Therefore, during the grid search, we restricted the range of possible angles to $[0, 15]$ degrees. The poor performances of `SensorsRotations` on the *SleepPhysionet* can be explained by the scarce number of EEG sensors available in this dataset. Indeed, the interpolation based on the 2 channels available must be imprecise. The performance was less degraded for rotation angles of 13.5°. For `ChannelsShuffle` the highest probability to shuffle channels yields the best results. On the contrary, for `ChannelsDropout`, the highest improvements are reached for a smaller probability of 0.3. The latter was also expected since only two channels are present in *SleepPhysionet*. Hence, a probability of $p_{drop} = 0.5$ would erase all channels on average for 1 out of 4 windows.

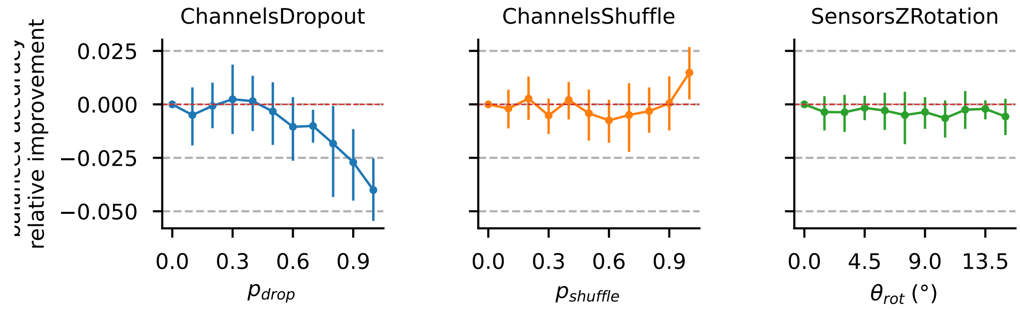


Figure 14: Spatial augmentations parameters selection on the *SleepPhysionet* dataset. The same model was trained on 350 windows using augmentations parametrized with 10 different values linearly spaced within intervals listed in Table 4. After the training, the validation accuracy was compared with a model trained without data augmentation. 10-fold cross-validation is used to obtain the error bars.

5.3.2 Learning curves

SleepPhysionet The spatial domain augmentations globally perform on par with the baseline. It thus seems that spatial domain augmentations are not particularly relevant for the sleep staging task, at least on the *SleepPhysionet* dataset. This was expected since this dataset only contains 2 EEG channels and hence, does not encode much spatial information.

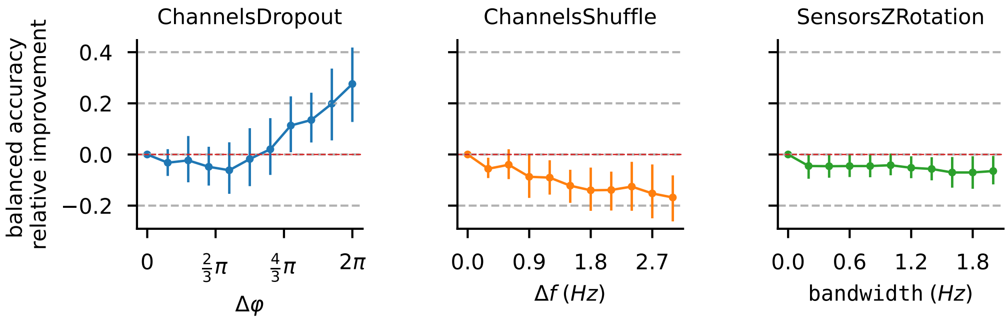


Figure 15: Caption

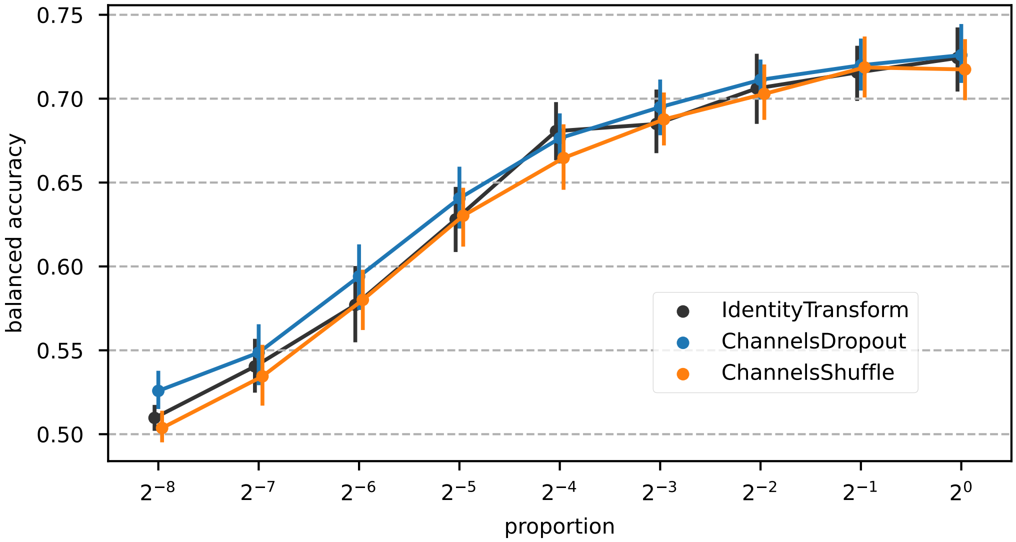


Figure 16: Learning curves for spatial domain augmentations along with the baseline trained with no augmentation. For each augmentation, the same model [19] has been trained on 8 fractions of increasing size of the *SleepPhysionet* dataset. After each training, the balanced accuracy score on the test set is recorded. 10-fold cross-validation is used to obtain the error bars.

5.3.3 Per class analysis

SleepPhysionet On the one hand, spatial augmentations do not seem to bring much improvement according both to Figures 16 and 18. On the other hand, it can be seen on the latter that they do not degrade significantly predictions for any class, contrarily to previously seen augmentations from sections 3 and 4. Together, these facts suggest that a small amount of information is related to the spatial domain for the sleep staging task.

6 Final remarks and limitations

Data augmentation allows to make up for the lack of brain data that prevents from leveraging the full potential of deep learning models. In this paper we presented and analyzed most of data augmentation methods for EEG signals. To do so, insights have been given on the motivations for each augmentation, which can either be related to the underlying neurophysiology or to experimental setups. Details about the implementation of the transformation have also been presented, with references to a consistent and clear application programming interface: braindecode. Experimentally, the impact of the strength of an augmentation has been assessed, revealing that transformations should be tailored to a task since adjustable parameters are of primary importance. Another experiment exhibiting the effects of augmentation across classes disclosed that within a dataset, the effects of augmentations differ notably from one class to another. Finally, augmentations have systematically been compared to each other and to a baseline. Illustrating these results on two substantially different dataset highlights the versatility of data augmentation methods. Still, this non exhaustive study could be enriched by the addition

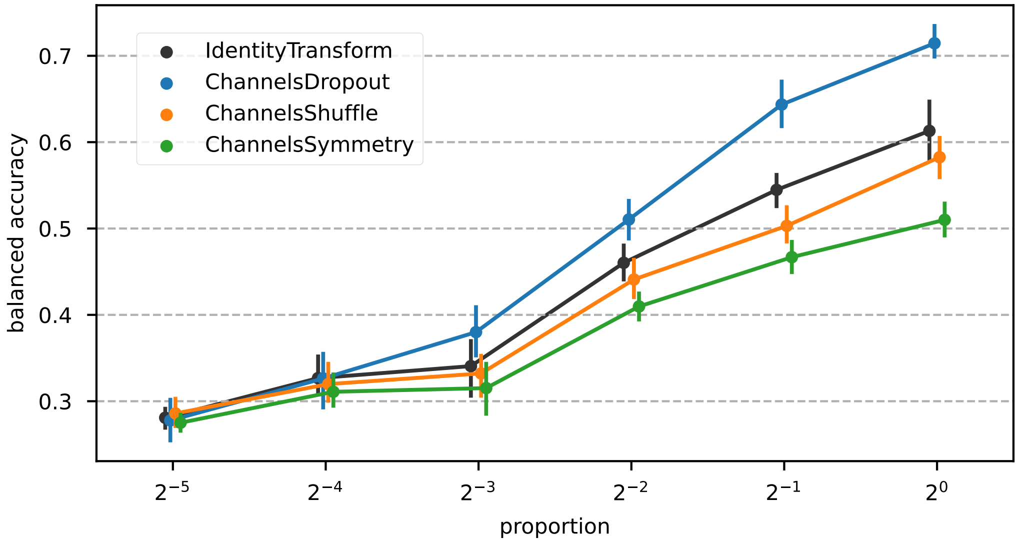


Figure 17: Caption

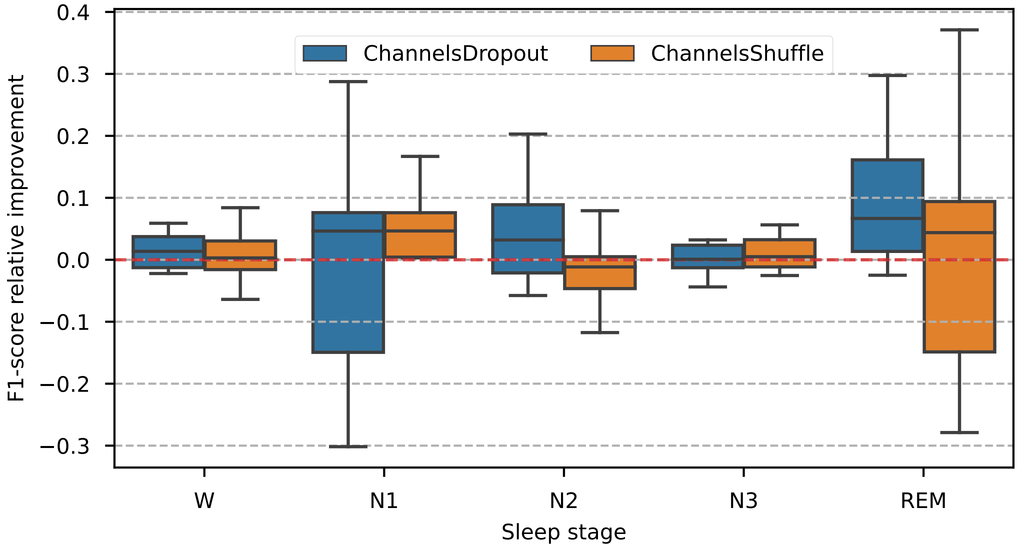


Figure 18: Relative improvement per class of the F1-score due to spatial domain transformations in comparison to the identity transformation. The scores have been computed after a training on 180 time windows, which corresponds to a proportion 2^{-7} of the *SleepPhysionet* dataset.

of other examples. For that purpose, besides the results obtained on two datasets, this systematic approach presents a methodological framework along with a reproductible code allowing to conduce a similar analysis on any other EEG dataset. By providing a detailed description and interpretation for each augmentation, we hope to foster the understanding of augmentation methods for EEG signals and to pave the way for further works in this field.

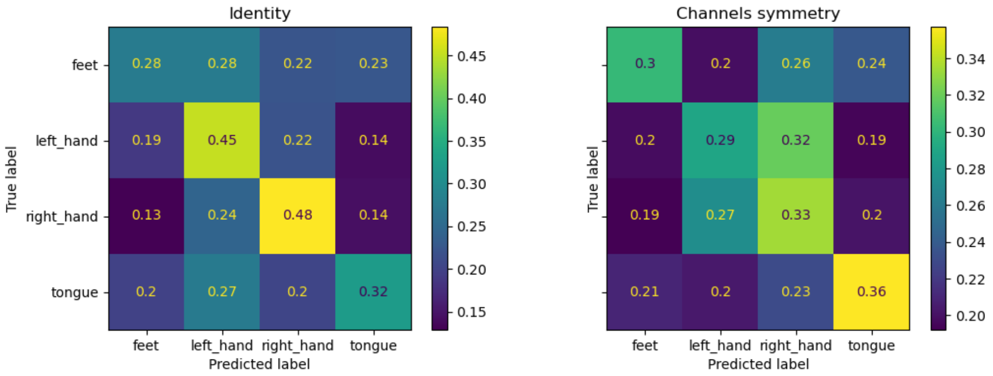


Figure 19: Channels symmetry only benefits symmetric brain activities. In order to obtain these confusion matrices, a *ShallowFBCSP* model has been trained on 8 subjects of the BCI IV 2a dataset and evaluated on the remaining one. These confusion matrices evidence that using the channels symmetry augmentation during training enhance the accuracy score for the motor imagery of feet and tongue movements whereas it hurts the performance for tasks that involve a lateralized activity.

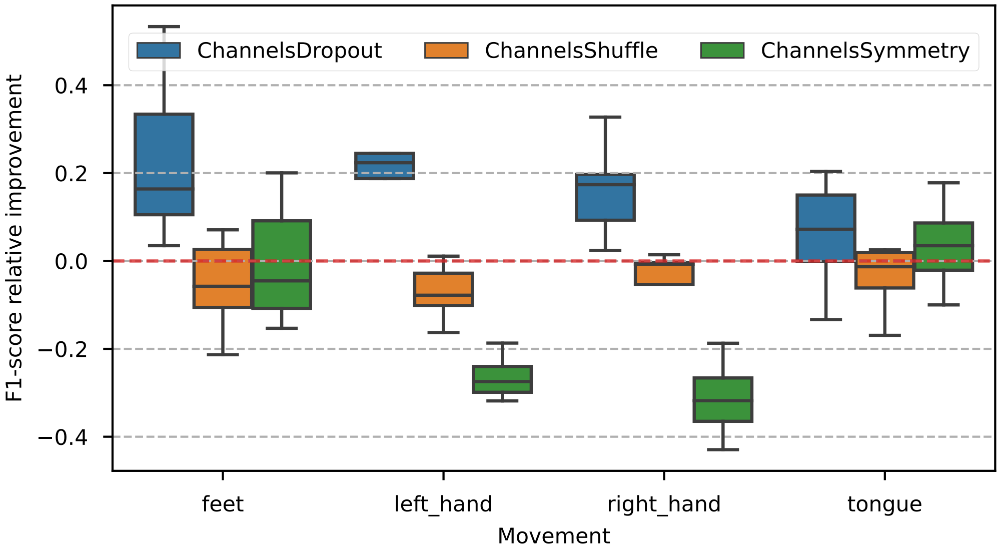


Figure 20

References

[1] M. Perslev, S. Darkner, L. Kempfner, M. Nikolic, P. J. Jennum, and C. Igel, "U-sleep: Resilient high-frequency sleep staging," *NPJ digital medicine*, vol. 4, no. 1, pp. 1–12, 2021.

[2] J. Chen, Z. Yu, Z. Gu, and Y. Li, "Deep temporal-spatial feature learning for motor imagery-based brain–computer interfaces," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 11, pp. 2356–2366, 2020.

[3] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b," *Frontiers in neuroscience*, vol. 6, p. 39, 2012.

[4] H. Ramoser, J. Müller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement," *IEEE transactions on rehabilitation engineering: a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 8, no. 4, pp. 441–446, 2000.

[5] O. Russakovsky, J. Deng, H. Su, *et al.*, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, 2015.

[6] R. S. Rosenberg and S. Van Hout, "The American Academy of Sleep Medicine Inter-scorer Reliability Program: Sleep Stage Scoring," *Journal of Clinical Sleep Medicine : JCSM : Official Publication of the American Academy of Sleep Medicine*, vol. 9, no. 1, pp. 81–87, 2013.

[7] M. Clerc, L. Bougrain, and F. Lotte, *Brain-Computer Interfaces 1*. Jul. 2016.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," en, in *Advances in neural information processing systems (NeurIPS)*, 2012.

[9] S. Chen, E. Dobriban, and J. H. Lee, "A Group-Theoretic Framework for Data Augmentation," en, in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.

[11] J. T. C. Schwabedal, J. C. Snyder, A. Cakmak, S. Nemati, and G. D. Clifford, "Addressing Class Imbalance in Classification Problems of Noisy Signals by using Fourier Transform Surrogates," *arXiv:1806.08675*, 2019.

[12] M. Mohsenvand, M. R. Izadi, and P. Maes, "Contrastive Representation Learning for Electroencephalogram Classification," en, in *Machine Learning for Health*, 2020.

[13] A. Saeed, D. Grangier, O. Pietquin, and N. Zeghidour, "Learning from heterogeneous EEG signals with differentiable channel reordering," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[14] M. M. Krell and S. K. Kim, "Rotational data augmentation for electroencephalographic data," en, in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2017, pp. 471–474.

[15] Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. H. Falk, and J. Faubert, "Deep learning-based electroencephalography analysis: A systematic review," en, *Journal of Neural Engineering*, vol. 16, no. 5, p. 051 001, 2019.

[16] R. Berry, R. Brooks, C. Gamaldo, *et al.*, *The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology and Technical Specifications : Version 2.3*. 2015.

[17] A. L. Goldberger, L. A. Amaral, L. Glass, *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, e215–e220, 2000, Publisher: Am Heart Assoc.

[18] A. Rechtschaffen and A. Kales, *A Manual of Standardized Terminology, Techniques and Scoring System for Sleep Stages of Human Subjects*. Brain Information Service/Brain Research Institute, University of California, 1973.

[19] S. Chambon, M. Galtier, P. Arnal, G. Wainrib, and A. Gramfort, “A deep learning architecture for temporal sleep stage classification using multivariate and multimodal time series,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 4, pp. 758–769, 2018.

[20] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” in *International Conference on Learning Representations (ICLR)*, 2015.

[21] C. Brunner, R. Leeb, G. Müller-Putz, A. Schlögl, and G. Pfurtscheller, “Bci competition 2008–graz data set a,” *Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology*, vol. 16, pp. 1–6, 2008.

[22] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, *et al.*, “Deep learning with convolutional neural networks for EEG decoding and visualization,” en, *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.

[23] ——, “Deep learning with convolutional neural networks for EEG decoding and visualization,” *Human Brain Mapping*, 2017.

[24] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *International Conference on Learning Representations (ICLR)*, 2019.

[25] C. Rommel, T. Moreau, J. Paillard, and A. Gramfort, “CADDA: Class-wise Automatic Differentiable Data Augmentation for EEG Signals,” *arXiv preprint arXiv:2106.13695*, 2021.

[26] R. G. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David, and C. E. Elger, “Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state,” *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 64, 2001.

[27] J. Y. Cheng, H. Goh, K. Dogrusoz, O. Tuzel, and E. Azemi, “Subject-Aware Contrastive Learning for Biosignals,” *arXiv:2007.04871*, 2020.

[28] A. Gramfort, M. Luessi, E. Larson, *et al.*, “MEG and EEG Data Analysis with MNE-Python,” *Frontiers in Neuroscience*, vol. 7, no. 267, pp. 1–13, 2013.

A Appendix

Why is the balanced accuracy for all classes always lower than the average of the balanced accuracies per class ? This point just helps to understand better the balanced accuracy metric. It is not instrumental for comprehension of the rest of the report since all the results are expressed in relative improvements.

Algorithm 1 Balanced accuracy implementation in *scikit learn*

```
Require: y_pred, y_true
    $C \leftarrow$confusion_matrix$(y\_true, y\_pred)$
per_class $\leftarrow np.diag(\mathrm{C})/\mathrm{C}.sum(axis = 1)$returnC$.mean()$
```

As shown in algorithm 1, in a multi class problem the balanced accuracy is defined by,

$$balanced - accuracy(y\_pred, y\_true) = \frac{1}{N}\sum_{c=1}^{N}\frac{TP_c}{n_c},$$

n_c being the number of samples with label c in y_true and TP_c the number of truly predicted samples for the label c.
In this work, in order to compute the class wise balanced accuracy for class c, y_pred (resp y_true) are changed to binary vectors in the following way, $y\_pred^c[i] \leftarrow 1$ if $y\_pred[i] = c$, $y\_pred^c[i] \leftarrow 0$ otherwise. Then the binary balanced accuracy is computed,

$$balanced - accuracy(y\_pred^c, y\_true^c) = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right),$$

where TP (resp TN), FP (resp FN) stand for true positive (resp negative) and false positive (resp negative). In this score, the specificity (true negative rate) clearly boosts the score since all other classes are considered as negative, regardless of the confusions between these classes. Furthermore, there is an overlap between the class wise negative samples. For these reasons, the multi-class balanced accuracy is lower than the average of the balanced accuracy per class.

Proof:
The following notations will be used: $S = balanced - accuracy(y\_pred, y\_true)$, $S_c = balanced - accuracy(y\_pred^c, y\_true^c)$ and N the number of classes

$$S - \frac{1}{N}\sum_{c=1}^{N} S_c = \frac{1}{N}\sum_{c=1}^{N}\frac{TP_c}{n_c} - \frac{1}{N}\sum_{c=1}^{N}\frac{1}{2}\left(\frac{TP_c}{n_c} + \frac{T\tilde{N}_c}{\tilde{n_c}}\right),$$

with $T\tilde{N}_c$ the number of rightly predicted samples that do not belong to class c (regardless of mistakes between other classes) and $\tilde{n_c}$ the total number of samples belonging to other classes than c. Then,

$$S - \frac{1}{N}\sum_{c=1}^{N} S_c = \frac{1}{2N}\sum_{c=1}^{N}\left(\frac{TP_c}{n_c} - \frac{T\tilde{N}_c}{\tilde{n_c}}\right),$$

Though,

$$\frac{T\tilde{N}_c}{\tilde{n_c}} = \frac{\sum_{i\neq c} TP_i}{\sum_{i\neq c} n_i} \leq \max_{i\neq c}\frac{TP_i}{n_i},$$

Consequently,

$$S - \frac{1}{N}\sum_{c=1}^{N} S_c \leq \frac{1}{2N}\sum_{c=1}^{N}\left(\frac{TP_c}{n_c} - \max_{i\neq c}\frac{TP_i}{n_i}\right) \leq 0,$$

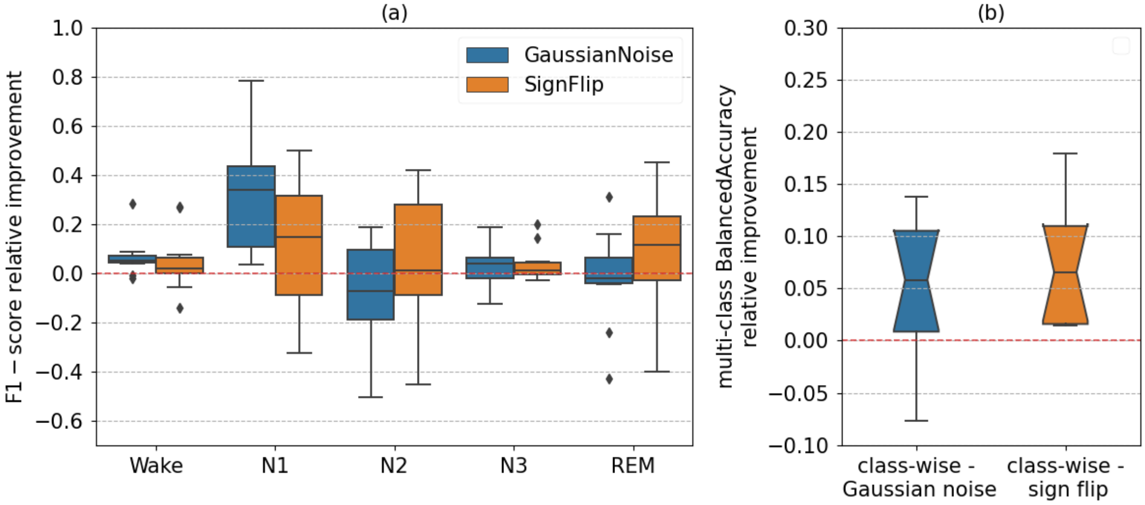


Figure 21: Benefits of a class-wise augmentation. Figure.a shows the class-wise relative improvement of the F1 − score due Gaussian noise and sign flip augmentations. Each augmentation encodes specific invariances that can be more relevant for some classes than others. In Figure.b it is evidenced that a class-wise augmentation policy outperforms class-agnostic ones. The class-wise augmentation consists in Gaussian noise for wake, N1, N3 and sign flip for N2 and REM.

Augmentation	Parameter	Interval	Unit	Best value
Gaussian noise	σ	$[0, 0.2]$	ø	XXX
Frequency shift	Δf	$[0, 3]$	Hz	$0.3Hz$
FT surrogate	`phase_noise_magnitude`	$[0, 1]$	ø	1
Smooth time mask	Δt	$[0, 2]$	s	$200s$
Bandstop filter	`bandwidth`	$[0, 2]$	Hz	$1.2Hz$
Sensor rotations	θ	$[0, 30]$	$\circ$	XXX
Channels shuffle	`p_shuffle`	$[0, 1]$	ø	1
Channels dropout	`p_drop`	$[0, 1]$	ø	0.3

Table 4: Table of the potential values for the adjustable parameter of each transformationThis graph might be subdivided and placed in the ”experimental results” section of each augmentation so as to avoid spoiling results before the end of the section describing the augmentation