# BCI Challenge - NER 2015
# Imagerie Fonctionnelle Cérébrale et Interface cerveau machine

Matthieu Futeral-Peter
Master MVA
matthieu.futeral.peter@ensae.fr

## 1. Introduction

The goal of this project is to complete the BCI Challenge NER 2015[1] which aims at building a model able to detect erroneous feedbacks online. Concretely, it is a "P300-Speller" experiment where subjects are asked to write a 5-letter word on the computer using only their thoughts. They then receive a feedback when the computer prints the letter it has detected on the screen. By recording subjects' brain activities, participants of the competition had to build models that exploits EEG data to discriminate between good and wrong feedbacks. To do so, they had to detect the Feedback-related negativity wave which is an electrophysiological measure reflecting the activity of the monitoring system via feedback signals. I used two approaches in this project : first of all, I built features from the data and fit machine learning models ; then, I only filtered the data in the range of frequencies of the brain and built deep learning models to get the highest possible Area Under the Curve score. All the programs of this project are available on gitlab[2].

## 2. Feedback-related negativity : Review

Before presenting the data and my method, I briefly introduce what is the feedback-related negativity and what it represents. Feedback-related negativity (FRN) is an event related potential measured approximately 250ms in response to an external feedback related to an experiment (appetitive or aversive). It is usually observed in the fronto-central region of the brain. It is important to detect the FRN reliably in order to perform automatic feedback in BCI experiments. However, before being able to detect it, it is more important to understand its role, how it is generated and where its magnitude comes from.

FRN was firstly thought to be larger for outcome "worse then expected" [2]. Based on this assumption, Hajcak et al. [3] set up a monetary loss experiment to study whether the magnitude of the FRN was related to the magnitude of the negative feedback. Indeed, based on RL theory, they assumed that the FRN was elicited by unexpected unfavorable outcomes and they therefore made two experiments in which subjects could randomly win 25ct, win 5ct, lose 25ct or lose 5ct at each trial. They concluded that the magnitude of the FRN appeared insensitive to the magnitude dimension of the feedback as they did not observe any variation of the FRN signal at each trial (except between loss or win), they nevertheless suspected a sigmoid relationship they could not have proved.

More recently, the idea that FRN is related to events "worse-than-expected" has been challenged. For example, Oliveira et al. [6] set up an experiment where participants were asked to evaluate the correctness of their own responses and the actual feedback were given afterwards. In this study, they found that the FRN was observed for both unexpected "correct" and "incorrect" feedbacks. It could mean that FRN is not related to the motivational value but is more related to the motivational salience. In other words, the potential "unexpected" aspect of the feedback would cause the FRN to arouse whereas people thought before it was the responsability of the value (good or bad) of that feedback.

Then, to prove that FRN is more related to the motivational salience and that it reflects "more and less" prediction errors either in appetitive or aversive conditions, Y. Huang et al. [4] set up a monetary experiment where participants could randomly face an appetitive experiment or an aversive one. More specifically, there were 4 pie charts (2 aversives, 2 appetitives) representing the probability of winning (resp. losing) some amount of money versus no win (resp. no loss). The authors found that FRN was sensitive to the valence of feedback in both conditions (appetitive or aversive) so they concluded that FRN reflects "more-or-less" prediction error in general and not "better-or-worse" than what was expected.

In the case of the BCI Challenge NER 2015, we should be able to detect FRN when the feedback is wrong because it means that it is a prediction error and therefore we should

[1] https://www.kaggle.com/c/inria-bci-challenge/overview
[2] GITLAB

detect FRN more easily in such cases (because it is supposed to have more magnitude).

## 3. Data Vizualisation

First of all, I plotted some EEG Topograph[3] in order to highlight the high subject variability as well as the high intra-subject variability between each session. Concretely, I epoched the data between 0s and 1.3s after a feedback occured, then I separated the epochs between positive and negative feedback and eventually I averaged the topograph over an entire session (while still discriminating between positive and negative feedback in order to analyze the differences) so I ended up with two topographs for each patient for each session.

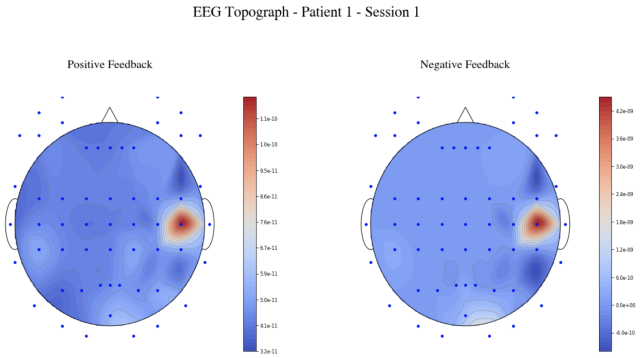


Figure 1. EEG Topographs of **patient 1** during **session 1** - Positive feedback to the left, negative feedback to the right

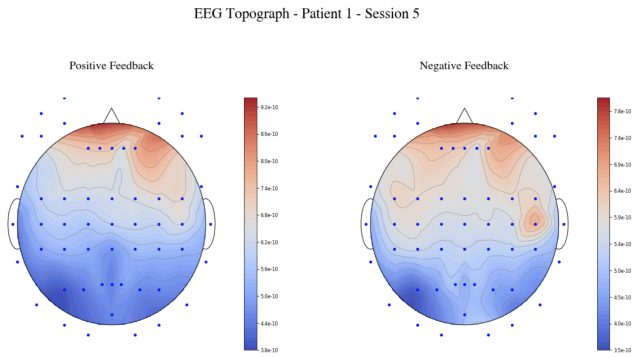


Figure 2. EEG Topographs of **patient 1** during **session 5** - Positive feedback to the left, negative feedback to the right

From both EEG topographs, it is possible to notice that the recorded brain activities when a positive feedback occurs and when a negative feedback occurs do not vary in terms of localization in a same session for a same patient. However, the signal localization varies between sessions. Indeed, for the topograph of patient 1 during session 5, displayed in figure 2, highlights that the signal is now localized in the frontal part of the cortex whereas it was not the case in figure 1.

[3] https://github.com/ijmax/EEG-processing-python

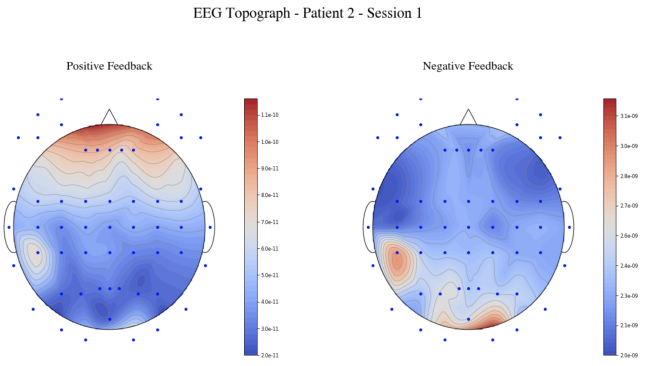


Figure 3. EEG Topographs of **patient 2** during **session 1** - Positive feedback to the left, negative feedback to the right

Figure 3 highlights the high inter-subjects variability. Indeed, for patient number 2, the topographs differ whether the feedback is positive or negative. In case the feedback is negative, the signal is localized in the occipital lobe. In case the feedback is positive, the signal is localized in the frontal lobe. It is quite different from patient 1 whose signals do not differ in location whether the feedback is positive or negative but according to the session.

Then, I found interesting to plot the mean signal along the time axis, for each channel.

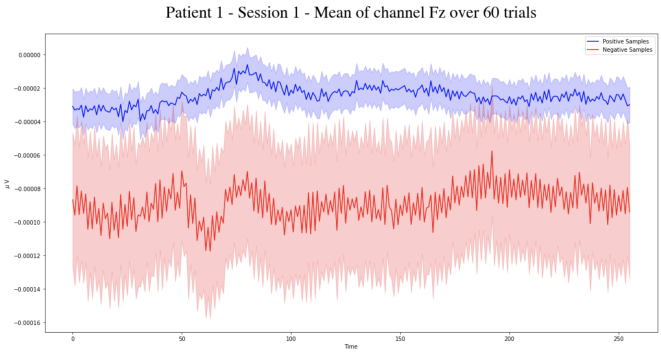


Figure 4. Fz Channel of **patient 1** during **session 1** with 95% confidence interval - Positive feedback in blue, negative feedback in red

In the literature, FRN wave is thought to be located in the fronto-central part of the brain. The signal recorded by the channels Fz and FCz for patient 1 during session 1 are displayed in figure 4 and figure 7. It is possible to notice a negative peak 250ms after the feedback in case the feedback was negative so it could be identify as the FRN wave according to the literature.

However, it is hard to conclude anything because the high inter-subject variability. Below are the signal averaged over every session and recorded by channel FCz for patient

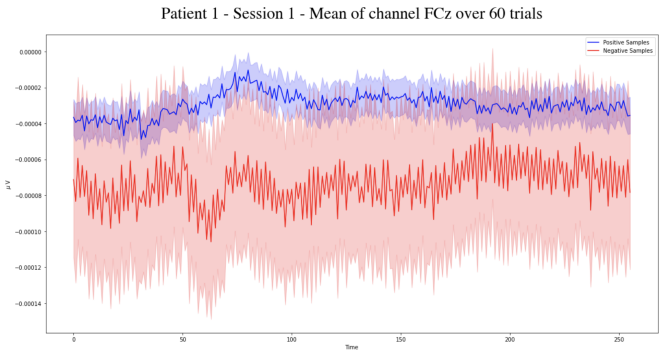


Figure 5. FCz Channel of **patient 1** during **session 1** with 95% confidence interval - Positive feedback in blue, negative feedback in red

1 and patient 2.

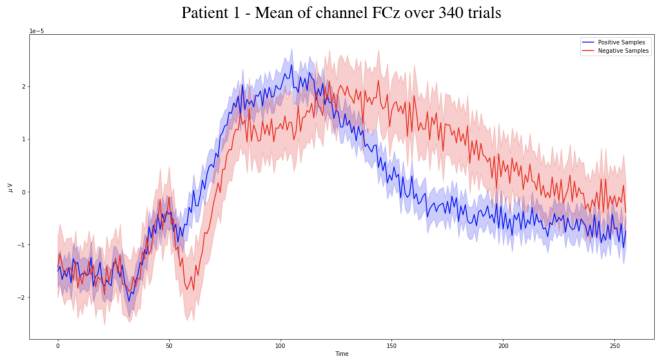


Figure 6. FCz Channel of **patient 2** averaged over all trials and all sessions with 95% confidence interval - Positive feedback in blue, negative feedback in red

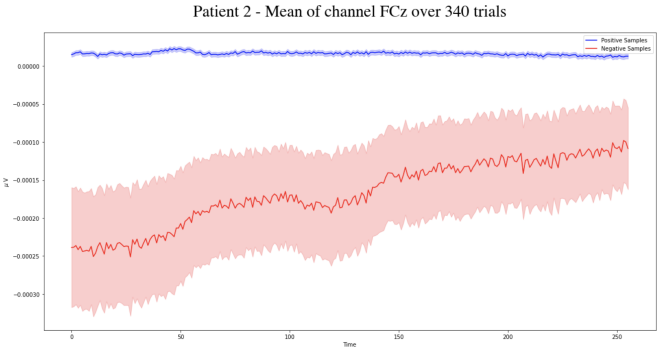


Figure 7. FCz Channel of **patient 2** averaged over all trials and all sessions with 95% confidence interval - Positive feedback in blue, negative feedback in red

The plots are totally different from each other. It may be possible to identify FRN wave for patient 1 but it is completely impossible for patient 2. In addition, the mean signal has the same magnitude whether the feedback was positive or negative whereas it is not the case at all for patient 2. It highlights how hard the task is and how important the inter-subject variability and the inter-session variability are.

## 4. Methods & Results

First of all, as I did not have either lots of time or lots of computing power, I did not use cross-validation to select the best model. Instead, as I had directly access to the public score and the private score by submitting my predictions on Kaggle, I only used 3 subjects out of the 16 subjects in the training set as my validation set in order to have 3 sets (validation set, public set and private set) to evaluate my models on. I did not do any grid search as well as it is computionnally very expensive. The actual results are therefore likely to be slightly better than the results displayed below.

### 4.1. Feature engineering & machine learning

I used three types of features and fitted basics classifier models such as logistic regression, SVM, XGBoost, Elasticnet and so on for each type of features in order to compare the importance of the features in getting the highest AUROC. More specifically, I extracted time features, frequency (spectral) features and spatial features. In any case, I applied a butterworth filter in order to smooth frequencies outside the range 1-40Hz because it is the range of the brain activity, noise is therefore reduced. In addition, I extracted epochs 1.3s after the feedback because the FRN peaks 250ms after the feedback so 1.3s must be large enough to keep the entire wave.

#### 4.1.1 Time domain

First of all, I fitted classifiers directly on the time series without extracting any further features. In other words, I fitter classifiers on time series filtered in the range 1-40Hz. I tried to apply an Independent Component Analysis (ICA) as well in order to obtain indepedent components, remove signals related to ocular movements (EOG is a feature in the dataset) and increase SNR but the results were exactly the same as features without ICA. It may be because I applied a bandpass filter before applying ICA so the artifacts were already removed because out of the frequency range. Results are displayed below :

| | Val. set | Pub. score | Priv. score |
|---|---|---|---|
| Log. reg. (l2 pen.) | 0.565 | 0.680 | **0.629** |
| ElasticNet | 0.555 | 0.651 | **0.608** |
| XGBoost | 0.605 | 0.679 | **0.652** |

Table 1. Area Under the ROC - Time-domain features

Results are clearly not good but it is expected because only few preprocessing and features extraction steps were applied so far. Moreover, data is very high dimension and

these models are not well designed to process such high dimensional data. Here, each epoch is composed of 256 samples and 56 channels which is equal to 14,336 features. It is not the best way to represent a multi-dimensional time series as well.

#### 4.1.2 (Time -) Frequency domain

Then, I tried to extract features from the frequency domain. I decomposed the signal into the Daubechies wavelets orthogonal basis and I used the wavelet coefficents as input features for the classifiers. The goal was to feed the classifiers with more descriptive features to describe the signals. Previously to that, I tried to process the data into the spectral domain by applying the Discrete Fourier Transform to the epochs data but the results were not significantly better. Here are the results with the wavelet coefficients as features :

| | Val. set | Pub. score | Priv. score |
|---|---|---|---|
| Log. reg. (l2 pen.) | 0.581 | 0.693 | **0.629** |
| ElasticNet | 0.585 | 0.701 | **0.634** |
| XGBoost | 0.582 | 0.692 | **0.650** |

Table 2. Area Under the ROC - Wavelet coefficients features

Results are slightly better than the previous ones expect for XGBoost. It may be due to the fact that wavelets coefficients describe more precisely the signals than time-domain features because they catch frequencies information and the former are more important to describe a signal and to classify it.

#### 4.1.3 Spatial domain

Eventually, I tried to apply the preprocessing steps of the winner of the competition. I then firstly fitted two set of xDAWN [7] covariance in order to project the raw EEG data into the estimated evoked subspace and therefore improve the SNR. Then, I projected the covariance features into the Tangent Space. This preprocessing pipeline improved a lot the results :

| | Val. set | Pub. score | Priv. score |
|---|---|---|---|
| Log. reg. (l2 pen.) | 0.655 | 0.721 | **0.779** |
| SVM | 0.652 | 0.723 | **0.777** |
| XGBoost | 0.644 | 0.711 | **0.769** |

Table 3. Area Under the ROC - Spatial features

The results are much better than the previous ones. This means that spatial filtering is the key to extracting the relevant features that best explain the FRN wave.

### 4.2. Deep learning

Another approach to deal with the classification of EEG signals is deep learning. Indeed, deep learning are expected to extract meaningful features from (almost) raw data and therefore to have great performance. In addition, it allows scientists to avoid the feature engineering part which is usually quite painful. Therefore, I implemented in PyTorch two CNN-based neural networks in order to classify EEG signals between good and wrong feedback.

As mentionned in the previous section, a butterworth bandpass filter was applied to the raw data in order to keep frequencies between 1Hz and 40Hz and removing some noise. Indeed, frequencies of the brain activity are mainly in the range frequency 1-40Hz so we are pretty sure not to smooth FRN signal applying butterworth filtering. I then built epochs of about 1.3s in order to be sure to catch FRN signal that peaks at about 250ms. I did not remove any channel but I removed EOG features.

**EEGNet**

Then, I firstly implemented in PyTorch EEGNET [5] which is a CNN-based neural network especially designed to deal with EEG data. It is composed of several CNN-blocks processing the data along time axis and spatial axis. In addition, I tried to augment the dataset by introducing a Gaussian white noise to the input. Adding noise indeed allows the network to be more robust to data noise and to augment the amount of data as mentionned in [8] :

$$\text{data}_i \leftarrow \text{data}_i + \mathcal{X}_i$$
$$\text{where, } \mathcal{X}_i \sim \mathcal{N}(0, \sigma)$$

I tried different $\sigma$ values from 10 to 50. In addition, as is common in the case of unbalanced classification, the loss was weighted by the weight of each class in the training set. Here are the results I obtained :

| | Val. set | Public score | Private score |
|---|---|---|---|
| No noise | 0.667 | 0.717 | **0.713** |
| $\sigma$ = 10 | 0.663 | 0.693 | **0.699** |
| $\sigma$ = 30 | 0.701 | 0.751 | **0.729** |
| $\sigma$ = 50 | 0.699 | 0.743 | **0.716** |

Table 4. EEGNet Area Under the ROC curve of the BCI 2015 Challenge for different $\sigma$ value

It is not totally clear why using a Gaussian noise with $\sigma$ = 10 gives poorer results. It may be because such a standard deviation breaks down data structure in our case whereas $\sigma$ = 30 or 50 clearly significantly improves the results on the three datasets.

**ERPENet**

I implemented in PyTorch another CNN-based neural network especially designed to process EEG data. ERPENet [1] is a multi-task autoencoder that aims to extract a relevant compressed feature from EEG data using an autoencoder and classify that compressed feature at the same time. Thus, the extracted compressed feature is related to the classification task you are trying to solve.

Concretely, the autoencoder is composed of two symmetric blocks : an encoder and an decoder. The encoder is composed of multiple CNNs and a LSTM to process the data along the time axis : the input shape is (time, I, J) where I * J = number of channels. The input shape is designed to reconstruct the geometric aspect of the EEG channels. Then the decoder is composed of an LSTM and Deconvolution neural networks. The output of the encoder is given to a linear classifier which outputs the probability that the signal is a good feedback. Formally, the loss function is :

$$
\begin{aligned}
L_{MSE} &= ||s_j - Dec(Enc(s_j))||^2 \\
L_{classifier} &= -\alpha_1 * y_j \log(\widehat{y}_j) - \alpha_0 * (1 - y_j) \log(1 - \widehat{y}_j) \\
L &= \beta L_{MSE} + \gamma L_{classifier}
\end{aligned}
$$

where $\alpha_0, \alpha_1$ are the weights to handle unbalanced classes, $\beta, \gamma$ are hyperparameters to weight the importance of the different loss functions.

Despite having tried different configurations and hyperparamaters, I did not succeed in training ERPENet on the BCI Challenge 2015. Indeed, the training converges towards a dummy solution which consists in predicting only the positive class. It is hard to find out what is wrong but the authors [1] gave a hint and mentionned in the paper that ERPENet needs lots of data in order to learn patterns and perform accurate predictions. Indeed, the authors trained their model on 6 different EEG datasets and found that using only 1 dataset was not enough to train such a model because the validation loss did not decrease at all.

## 5. Discussion

Given the high variability among patients, the high variability among sessions of the same patient and the low Signal-to-Ratio of EEG data, the task is very difficult. It seems therefore necessary to extract relevant features from it before building models. It appears that spatial filters are the ones that best explain the FRN wave because such features give best results. In addition, it seems that deep learning models do not extract features relevant enough for EEG data even if the results are very good, they are lower than standard machine learning with spatial features.

## References


[1] Apiwat Ditthapron, Nannapas Banluesombatkul, Sombat Ketrat, Ekapol Chuangsuwanich, and Theerawit Wilaiprasitporn. Universal joint feature extraction for p300 eeg classification using multi-task autoencoder. *Institute of Electrical and Electronics Engineers (IEEE)*, 7, 2019. 5

[2] M Falkenstein, J Hohnsbein, J Hoormann, and L Blanke. Effects of crossmodal divided attention on late erp components. ii. error processing in choice reaction tasks. *Electroencephalography and clinical neurophysiology*, page 447—455, 1991. 1

[3] G. Hajcak, J. Moser, C. Holroyd, and R. Simons. The feedback-related negativity reflects the binary evaluation of good versus bad outcomes. *Biological Psychology*, 2006. 1

[4] Yi Huang and Rongjun Yu. The feedback-related negativity reflects "more or less" prediction error in appetitive and aversive conditions. *Frontiers in Neuroscience*, 2014. 1

[5] Vernon J. Lawhern, Amelia J. Solon, Nicholas R. Waytowich, Stephen M. Gordon, Chou P. Hung, and Brent J. Lance. Eegnet: A compact convolutional network for eeg-based brain-computer interfaces. *CoRR*, 2016. 4

[6] Flavio T. P. Oliveira, John J. McDonald, and David Goodman. Performance Monitoring in the Anterior Cingulate is Not All Error Related: Expectancy Deviation and the Representation of Action-Outcome Associations. *Journal of Cognitive Neuroscience*, 2007. 1

[7] Bertrand Rivet, Antoine Souloumiac, Virginie Attina, and Guillaume Gibert. xdawn algorithm to enhance evoked potentials: application to brain-computer interface. *IEEE transactions on bio-medical engineering*, 2009. 4

[8] Albuquerque I Gramfort A Falk TH Faubert J. Roy Y, Banville H. Deep learning-based electroencephalography analysis: a systematic review. *Neural Eng.*, 2019. 4