Published in final edited form as:
Neuroimage. 2017 October 01; 159: 417–429. doi:10.1016/j.neuroimage.2017.06.030.

Autoreject: Automated artifact rejection for MEG and EEG data

Mainak Jas[a,*], **Denis A. Engemann**[b,c,d,1], **Yousra Bekhti**[a], **Federico Raimondo**[d,e,f,g], **Alexandre Gramfort**[a,1]

[a]LTCI, Télécom ParisTech, Université Paris-Saclay, France

[b]Parietal project-team, INRIA Saclay - Ile de France, France

[c]Cognitive Neuroimaging Unit, Neurospin, CEA DSV/I2BM, INSERM, Université Paris-Sud, Université Paris-Saclay, NeuroSpin center, 91191 Gif/Yvette, France

[d]Institut du Cerveau et de la Moelle épinière, ICM, PICNIC Lab, F-75013, Paris, France

[e]Laboratorio de Inteligencia Artificial Aplicada, Departamento de Computación, FCEyN, Universidad de Buenos Aires, Argentina

[f]CONICET, Argentina

[g]Sorbonne Universités, UPMC Univ Paris 06, Faculté de Médecine Pitié-Salpêtrière, Paris, France

Abstract

We present an automated algorithm for unified rejection and repair of bad trials in magnetoencephalography (MEG) and electroencephalography (EEG) signals. Our method capitalizes on cross-validation in conjunction with a robust evaluation metric to estimate the optimal peak-to-peak threshold – a quantity commonly used for identifying bad trials in M/EEG. This approach is then extended to a more sophisticated algorithm which estimates this threshold for each sensor yielding trial-wise bad sensors. Depending on the number of bad sensors, the trial is then repaired by interpolation or by excluding it from subsequent analysis. All steps of the algorithm are fully automated thus lending itself to the name *Autoreject*.

In order to assess the practical significance of the algorithm, we conducted extensive validation and comparisons with state-of-the-art methods on four public datasets containing MEG and EEG recordings from more than 200 subjects. The comparisons include purely qualitative efforts as well as quantitatively benchmarking against human supervised and semi-automated preprocessing pipelines. The algorithm allowed us to automate the preprocessing of MEG data from the Human Connectome Project (HCP) going up to the computation of the evoked responses. The automated nature of our method minimizes the burden of human inspection, hence supporting scalability and reliability demanded by data analysis in modern neuroscience.

[*]Correspondence to: 46 Rue Barrault, Télécom ParisTech, Université Paris-Saclay, France. mainak.jas@telecom-paristech.fr (M. Jas).
[1] Equal contributions

Appendix A
Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.neuroimage.2017.06.030.

Keywords

Magnetoencephalography (MEG); Electroencephalogram (EEG); Preprocessing; Statistical learning; Cross-validation; Automated analysis; Human Connectome Project (HCP)

Introduction

Magneto-/electroencephalography (M/EEG) offer the unique ability to explore and study, non-invasively, the temporal dynamics of the brain and its cognitive processes. The M/EEG community has only recently begun to appreciate the importance of large-scale studies, in an effort to improve replicability and statistical power of experiments. This has given rise to the practice of sharing and publishing data in open archives (Gorgolewski and Poldrack, 2016). Examples of such large electrophysiological datasets include the Human Connectome Project (HCP) (Van Essen et al., 2012; Larson-Prior et al., 2013), the Physiobank (Goldberger et al., 2000), the OMEGA archive (Niso et al., 2016) and Cam-CAN (Taylor et al., 2015). A tendency towards ever-growing massive datasets as well as a shift towards common standards for accessing these databases (Gorgolewski et al., 2016; Bigdely-Shamlo et al.,) is clearly visible. The UK Biobank project (Ollier et al., 2005) which currently hosts data from more than 50,000 subjects is yet another example of this trend.

This has however, given rise to new challenges including automating the analysis pipeline (Gorgolewski and Poldrack, 2016). Automation will not only save time, but also allow scalable analysis and reduce the barriers to reanalysis of data, thus facilitating reproducibility. Engemann and Gramfort (2015) have recently worked towards more automation in M/EEG analysis pipelines by considering the problem of covariance estimation, a step commonly done prior to source localization. Yet, one of the most critical bottlenecks that limits the reanalysis of M/EEG data remains at the preprocessing stage with the annotation and rejection of artifacts. Despite being so fundamental to M/EEG analysis given how easily such data can be corrupted by noise and artifacts, there is currently no consensus in the community on how to address this particular issue.

In the presence of what we will refer to as *bad* data, various data cleaning strategies have been employed. A first intuitive strategy is to exclude bad data from analysis, to *reject* it. While this approach is very often employed, for example, because data cleaning is time consuming, or out of reach for practitioners, it leads to a loss of data that are costly to acquire. This is particularly the case for clinical studies, where patients have difficulties staying still or focusing on the task (Cruse et al., 2012; Goldfine et al., 2013), or even when babies are involved as subjects (Basirat et al., 2014).

When working with M/EEG, the data can be bad due to the presence of bad sensors (also known as channels[2]) and bad trials. A trial refers here to a data segment whose location in time is typically related to an experimental protocol. But here we will also call trial any data segment even if it is acquired during a task-free protocol. Accordingly, a bad trial or bad

[2] They are not necessarily equivalent in the case of a bipolar montage in EEG. However, for the sake of simplicity, we shall use these terms interchangeably in this work.

sensor is one which contains bad data. Ignoring the presence of bad data can adversely affect analysis downstream in the pipeline. For example, when multiple trials time-locked to the stimulation are averaged to estimate an evoked response, ignoring the presence of a single bad trial can corrupt the average. The mean of a random vector is not robust to the presence of strong outliers. Another example quite common in practice, both in the case of EEG and MEG, is the presence of a bad sensor. When kept in the analysis, an artifact present on a single bad sensor can spread to other sensors, for example due to spatial projection. This is why identifying bad sensors is crucial for data cleaning techniques such as the very popular Signal Space Separation (SSS) method (Taulu et al., 2004). Frequency filtering (Widmann et al., 2015) can often suppress many low frequency artifacts, but turns out to be insufficient for broadband artifacts. A common practice to mitigate this issue is to visually inspect the data using an interactive viewer and mark manually, the bad sensors and bad segments in the data. Although trained experts are very likely to agree on the annotation of bad data, their judgement is subject to fluctuations and cannot be repeated. Their judgement can also be biased due to prior training with different experimental setups or equipments, not to mention the difficulty for such experts to allocate some time to review the raw data collected everyday.

Luckily, popular software tools such as Brainstorm (Tadel et al., 2011), EEGLAB (Delorme and Makeig, 2004), FieldTrip (Oostenveld et al., 2011), MNE (Gramfort et al., 2013) or SPM (Litvak et al., 2011) already allow for the rejection of bad data segments based on simple metrics such as peak-to-peak signal amplitude differences that are compared to a manually set threshold value. When the peak-to-peak amplitude in the data exceeds a certain threshold, it is considered as bad. However, while this seems quite easy to understand and simple to use from a practitioner’s standpoint, this is not always convenient. In fact, a good peak-to-peak signal amplitude threshold turns out to be data specific, which means that setting it requires some amount of trial and error.

The need for better automated methods for data preprocessing is clearly shared by various research teams, as the literature of the last few years can confirm. On the one hand, are pipeline-based approaches, such as Fully Automated Statistical Thresholding for EEG artifact rejection (FASTER by Nolan et al. (2010)) which detect bad sensors as well as bad trials using fixed thresholds motivated from classical Gaussian statistics. Methods such as PREP (Bigdely-Shamlo et al., 2015), on the other hand, aim to detect and clean the bad sensors only. Unfortunately, they do not offer any solution to reject bad trials. Other methods are available to solve this problem. For example, the Riemannian Potato (Barachant et al., 2013) technique can identify the bad trials as those where the covariance matrix lies outside of the “potato” of covariance matrices for good trials. By doing so, it marks trials as bad but does not identify the sensors causing the problem, hence not offering the ability to repair them. It appears that practitioners are left to choose between different methods to reject trials or repair sensors, whereas they are in fact intricately related problems and must be dealt with together.

Robust regression (Diedrichsen and Shadmehr, 2005) also deals with bad trials using a weighted average which mitigates the effect of outlier trials. Trials with artifacts end up with low contributions in the average. A related approach that is sometimes employed to ignore outlier trials in the average is the trimmed mean as opposed to a regular mean. The trimmed mean is a compromise between the mean which offers a high signal-to-noise ratio (SNR) but can be corrupted by outliers, and the median which is immune to outliers of extreme amplitudes but has a low SNR as it involves no averaging. Of course, neither of these strategies are useful when analyses have to be conducted on single trials. Another approach, which is also data-driven, is Sensor Noise Suppression (SNS) (De Cheveigné and Simon, 2008). It removes the sensor-level noise by spatially projecting the data of each sensor onto the subspace spanned by the principal components of all the other sensors. This projection is repeated in leave-one-sensor-out iterations so as to eventually clean all the sensors. In most of these methods, however, there are parameters which are somewhat dataset dependent and must therefore be manually tuned.

We therefore face the same problem in automated methods as in the case of semi-automated methods such as peak-to-peak rejection thresholds, namely the tuning of model parameters. In fact, setting the model parameters is even more challenging in some of the methods when they do not directly translate into human-interpretable physical units.

This led us to adopt a pragmatic approach in terms of algorithm design, as it focuses on the tuning of the parameters that M/EEG users presently choose manually. The goal is, not only to obtain high quality data but also to develop a method which is transparent and not too disruptive for the majority of M/EEG users. A first question we address below is: can we improve peak-to-peak based rejection methods by automating the process of trial and error? In the following section, we explain how the widely-known statistical method of cross-validation (see Fig. 1 for a preview) in combination with Bayesian optimization (Snoek et al., 2012, Bergstra et al., 2011) can be employed to tackle the problem at hand. We then explain how this strategy can be extended to set thresholds separately for each sensor and mark trials as bad when a large majority of the sensors have high-amplitude artifacts. This process closely mimics how a human expert would mark a trial as bad during visual inspection.

In the rest of the paper, we detail the internals of our algorithm, compare it against various state-of-the-art methods, and position it conceptually with respect to these different approaches. For this purpose, we make use of qualitative visualization techniques as well as quantitative reports. In a major validation effort, we take advantage of cleaned up evoked response fields (ERFs) provided by the Human Connectome Project (Larson-Prior et al., 2013) enabling ground truth comparison between alternative methods. This work represents one of the first efforts in reanalysis of the MEG data from the HCP dataset using a toolkit stack significantly different from the one employed by the HCP consortium. The convergence between our method and the HCP MEG pipelines is encouraging and testifies to the success of the community-wide open science efforts aiming at reproducible research. Naturally, we have therefore made our code available online[3]. In addition to this, we validated our algorithm on the MNE sample data (Gramfort et al., 2013), the multimodal faces dataset (Wakeman and Henson, 2015), and the EEGBCI motor imagery data (Goldberger et al., 2000; Schalk et al., 2004).

[3] https://autoreject.github.io

A preliminary version of this work was presented in Jas et al. (2016).

Notations.

We denote matrices by capital letters $X \in \mathbb{R}^{m \times n}$. The *i*th row of a matrix is indexed by subscripts, as in X_i, and the entry in the *i*th row and *j*th column is indexed as X_{ij}. The matrix X restricted to the rows with indices in the set is $\mathscr{G}$ denoted by $X_{\mathscr{G}}$. All sets $\mathscr{G}$, $\mathscr{T}$ or $\mathscr{V}$ are written in calligraphic fonts.

Materials and methods

We will first describe how a cross-validation procedure can be used to set peak-to-peak rejection thresholds globally (i.e. same threshold for all sensors). This is what we call *autoreject* (*global*).

Autoreject (global)

We denote the data matrix by $X \in \mathbb{R}^{N \times P}$ with N trials and P features. These P features are the Q sensor-level time series, each of length T concatenated along the second dimension of the data matrix, such that $P=QT$. We divide the data into K folds (along the first dimension) with training set indices $\mathscr{T}_k$ and validation set indices $\mathscr{V}_k = [1..N] \backslash \mathscr{T}_k$ for each fold k ($1 \leq k \leq K$). For simplicity of notation, we first define the peak-to-peak amplitude for the *i*th trial and *j*th sensor as the difference between the maximum and the minimum value in that time series:

$$\mathscr{A}_{ij} = \max_{(j-1)T+1 \leq t \leq jT} (X_{it}) - \min_{(j-1)T+1 \leq t \leq jT} (X_{it}) \, . \tag{1}$$

The good trials $\mathscr{G}_k$ where the peak-to-peak amplitude $\mathscr{A}_{ij}$ for any sensor does not exceed the candidate threshold τ are computed as

$$\mathscr{G}_k = \{ i \in \mathscr{T}_k | \max_{1 \leq j \leq Q} \mathscr{A}_{ij} \leq \tau \} \, . \tag{2}$$

By comparing the peak-to-peak threshold with the maximum of the peak-to-peak amplitudes, we ensure that none of the sensors exceed the given threshold. Once we have applied the threshold on the training set, it is necessary to evaluate how the threshold performs by looking at new data. For this purpose, we consider the validation set. Concretely speaking, we propose to compare the mean $\overline{X_{\mathscr{G}_k}}$ of good trials in the training set against the median $\widetilde{X_{\mathscr{V}_k}}$ of all trials in the validation set. Using root mean squared error (RMSE) the mismatch $e_k(\tau)$ reads as:

$$e_k(\tau) = \| \overline{X_{\mathscr{G}_k}} - \widetilde{X_{\mathscr{V}_k}} \|_{\text{Fro}} \, . \tag{3}$$

Here, $\|\cdot\|_{\text{FRO}}$ is the Frobenius norm. The rationale for using the median in the validation set is that it is robust to outliers. Indeed, it is far less affected by high-amplitude artifacts than a classical mean. Note that this metric is valid even if the artifact-free signal is zero-mean, which is for instance the case for resting state data. An underlying assumption however is that the artifacts are deviations of the signals far from zero. An alternative to the median could be a trimmed mean (as mentioned in the Introduction). However, this would require the setting of another parameter, namely the fraction of trimmed values. The threshold with the best data quality (lowest mismatch $e_k(\tau)$) on average across the K folds is selected as the optimal threshold. In practice τ is taken in a bounded interval $[\tau_{\min}, \tau_{\max}]$:

$$\tau_{\star} = \underset{\tau \in [\tau_{\min}, \tau_{\max}]}{\operatorname{argmin}} \frac{1}{K} \sum_{k=1}^{K} e_k(\tau) \tag{4}$$

Note, that $\widetilde{X\mathscr{V}_k}$ does not depend on τ. Indeed, it would not be wise to restrict the validation set to good trials according to the value of τ. As τ varies, it would lead to a variable number of validation trials, which would affect the comparison of RMSE across threshold values. The idea of using the median in the context of cross-validation has been previously proposed in the statistics literature in order to deal also with outliers (Leung, 2005; De Brabanter et al., 2003).

Fig. 1 shows how the average RMSE changes as the threshold varies for the MNE sample dataset (Gramfort et al., 2013, 2014). At low thresholds, *autoreject* (*global*) underfits as it drops most of the trials in the data resulting in a noisy average. Here, we use the term underfit as we remove not just the noisy trials, but also the artifact-free trials, hence we do not trust enough the training data. On the other hand, at high thresholds, *autoreject* (*global*) overfits retaining all the trials in the data including the high-amplitude artifacts. Here the candidate values of τ were taken on a grid. More details on how to solve (4) will be given in the Section on Bayesian optimization.

Autoreject (local)

A global threshold common to all sensors, however, suffers from limitations. A common case of failure is when a single sensor is affected (locally or globally) by high-amplitude artifacts. In this case, $\max_j \mathscr{A}_{ij}$ which would be the peak-to-peak amplitude that is compared to the threshold, comes from this bad sensor. If the sensor is not repaired or removed, we might end up rejecting a large fraction of otherwise good trials, just because of a single bad sensor. This is certainly not optimal. In fact, a possibly better solution is to replace the corrupted signal in the sensor by the interpolation of the signals in the nearby sensors. A second observation is that sensors can have very different ranges of amplitudes depending on their location on the scalp. A threshold tuned for one sensor may not work as effectively for another sensor. Both of these observations are motivations for estimating rejection thresholds for each sensor separately.

Once we define sensor-wise rejection thresholds $\tau_{\star}^{j}$ we can define an indicator matrix $C_{ij} \in \{0,1\}^{N \times Q}$ which designates the bad trials at the level of individual sensors. In other words, we have:

$$C_{ij} = \begin{cases} 0, & \text{if } \mathscr{A}_{ij} \le \tau_{\star}^{j} \\ 1, & \text{if } \mathscr{A}_{ij} > \tau_{\star}^{j} \end{cases} \tag{5}$$

The schematic in Fig. 2A shows a cartoon figure for this indicator matrix C_{ij}. Now that we have identified bad sensors for each trial, one might be tempted to interpolate all the bad sensors in each trial. However, it is not as straightforward since in some trials, a majority of the sensors may be bad. These trials cannot be repaired by interpolation and must be rejected. In some other cases, the number of bad sensors may not be large enough to justify rejecting the trial. However, it might already be too much to interpolate all the sensors reliably. In these cases, a natural idea is to pick the worst few sensors and interpolate them. This suggests an algorithm as described in Fig. 2B. Reject a trial only if most sensors “agree” that the trial is bad, otherwise interpolate as many sensors as possible. We will denote by κ the maximum number of bad sensors in a non-rejected trial and by ρ the maximum number of sensors that can be interpolated. Note that ρ is necessarily less than κ. The interpolation scheme for EEG uses spherical splines (Perrin et al., 1989) while for MEG it uses a Minimum Norm Estimates formulation with spherical harmonics (Hämäläinen and Ilmoniemi, 1994). The implementation is provided by MNE-Python (Gramfort et al., 2013).

The set of good trials $\mathscr{G}_k^{\kappa}$ in the training set $\mathscr{T}_k$ can therefore be written mathematically as:

$$\mathscr{G}_k^{\kappa} = \left\{ i \in \mathscr{T}_k \mid \sum_{j=1}^{Q} C_{ij} < \kappa \right\}. \tag{6}$$

In the remaining trials, if $\rho < \kappa$, one needs to define what are the worst ρ sensors that shall be interpolated. To do this we propose to rank the sensors for “badness” according to a score. A natural strategy to set the score is to use the peak-to-peak amplitude itself:

$$s_{ij} = \begin{cases} \mathscr{A}_{ij} & \text{if } C_{ij} = 1 \\ -\infty & \text{if } C_{ij} = 0 \end{cases} \tag{7}$$

Higher the score s_{ij}, the worse is the sensor. The $-\infty$ score is for ignoring the good sensors in the subsequent step. The following strategy is used for interpolation. If the number of bad sensors $\sum_{j'=1}^{Q} C_{ij'}$ is less than ρ we will interpolate all of them. Otherwise, we will interpolate the ρ sensors with the highest scores. In other words, we interpolate at most min $(\rho, \sum_{j'=1}^{Q} C_{ij'})$ sensors.

Denoting by $X_{\mathscr{G}_k^{\kappa}}^{\rho}$ the data in the training set after rejection and cleaning by interpolation, the RMSE averaged over K folds for the parameter pair (ρ, κ) therefore becomes:

$$\bar{e}(\rho, \kappa) = \frac{1}{K} \sum_{k=1}^{K} \left\| \overline{X_{\mathscr{G}_k^{\kappa}}^{\rho}} - \widetilde{X_{\mathscr{V}_k}} \quad \right\|_{\mathrm{Fro}} \tag{8}$$

where $\|\cdot\|_{\mathrm{FRO}}$ is the Frobenius norm. Finally, the best parameters ρ_* and κ_* are estimated using grid search (Hsu et al., 2003).

Data augmentation

In practice, cross-validation does not work for a globally bad sensor since all the trials are corrupted. In this scenario, the optimal threshold for this bad sensor should be lower than the lowest peak-to-peak amplitude so that all the trials for that sensor are marked as bad. However, even the median of the validation set has been corrupted. The algorithm therefore attempts to keep as many trials as necessary for the average to be close to the corrupted median. Thus, the estimated threshold ends up being higher than what would have been optimal. Recall from Fig. 1 that this is the classic case of an overfitting model. A common strategy in machine learning to reduce overfitting is data augmentation (Krizhevsky et al., 2012). It basically boils down to using the properties of the data, such as the physics of the system, to generate more plausible data.

To implement data augmentation in our model, we interpolate each sensor from all the other $Q - 1$ sensors and by doing so, we double the number of trials in the data. In the augmented data, half of the trials contain sensor data which are the output of a leave-one-sensor-out cross-validation. The augmented data matrix is $X^{\mathrm{aug}} \in \mathbb{R}^{2N \times P}$ The validation set in the augmented data contains a larger fraction of uncorrupted trials, and therefore the median is not corrupted. Finally, during cross-validation, the folds were stratified, that is the data splits contained roughly an equal number of interpolated and original trials.

Candidate thresholds using Bayesian optimization

Now that we have formalized the problem and our approach, we must estimate the threshold $\tau_\star$ which minimizes the error defined in (Eq. 3). A naïve strategy is to define a set of equally spaced points over a range of thresholds $[\tau_{\mathrm{min}}, \tau_{\mathrm{max}}]$. The estimated threshold would be the one which obtains the lowest error among these candidate threshold. This is the approach taken in Fig. 1. The range of thresholds is easy to set as it can be determined from the minimum and maximum peak-to-peak amplitude for the sensor in the augmented data matrix X^{aug}. However, it is not obvious how to set the spacing between the candidate thresholds, and experiments showed that varying this spacing could impact the results. If the candidate thresholds are far apart, one might end up missing the optimal threshold. On the other hand, if the thresholds are very dense, it is computationally more demanding.

This motivated us to use Bayesian optimization (Snoek et al., 2012, Bergstra et al., 2011) to estimate the optimal thresholds. It is a sequential approach which decides the next candidate threshold to try based on all the observed thresholds so far. It is based on maximizing an acquisition function given an objective function of samples seen so far (data likelihood) and the prior (typically a Gaussian Process (GP) (Rasmussen and Williams, 2006)). The objective function in our case is the mean cross-validation error as defined in Eqs. (3). To obtain the next iterate, an acquisition function is maximized over the posterior distribution. Popular choices of the acquisition function include “probability of improvement”, “expected improvement” and “confidence bounds of the GP” (Snoek et al., 2012). We pick “expected improvement” as it balances exploration (searching unknown regions) and exploitation (maximizing the improvement) strategies without the need of a tuning parameter. For our analysis, we use the scikit-optimize[4] implementation of Bayesian optimization, which internally uses the Gaussian process module from scikit-learn (Pedregosa et al., 2011).

Fig. 3A and 3B show the cross-validation curve for a regular sensor and a globally bad sensor in the MNE sample dataset (Gramfort et al., 2014, 2013). The RMSE is evaluated on thresholds as determined by the Bayesian optimization rather than a uniform grid. These plots also illustrate the arguments presented in the Data augmentation section with respect to data augmentation. The histograms in Fig. 3C for the interpolated data and the real data are overlapping for the regular sensor. Thus, the estimated threshold for that sensor marks a trial as outlier if its peak-to-peak values is much higher than the rest of the trials. However, in the case of a globally bad sensor, the histogram (Fig. 3D) is bimodal – one mode for the interpolated data and one mode for the real data. Now, the estimated threshold is no longer marking outliers in the traditional sense. Instead, all the trials belonging to that sensor must be marked as bad.

Experimental validation protocol

To experimentally validate *autoreject*, our general strategy is to first visually evaluate the results and thereafter quantify the performance. We describe below the evaluation metric used, the methods we compare against, and finally the datasets analyzed. All general data processing was done using the open source software MNE-Python (Gramfort et al., 2013).

Evaluation metric

The evoked response from the data cleaned using our algorithm or a competing benchmark is denoted by $\overline{X}$ (*method*). This is compared to the ground truth evoked response $\overline{X}$ (*clean*) (See Datasets to see how these are obtained for different datasets) using:

$$\left\| \overline{X}(\mathrm{method}) - \overline{X}(\mathrm{clean}) \right\|_\infty \qquad (9)$$

where $\|\cdot\|_\infty$ is the infinity norm. The reason for using infinity norm is that it is sensitive to the maximum amplitude in the difference signal as opposed to the Frobenius norm which averages the squared difference. The $\|\cdot\|_\infty$ is a particularly sensitive metric to quantity artifacts which are also visually striking such as those localized on one sensor or at a given time instant.

Competing methods

Here, we list the methods that will be quantitatively compared to *autoreject* using the evaluation metric in (Eq. 9). These methods are also summarized for the reader's convenience in Table 1.

- *No rejection*: It is a simple sanity check to make sure that the data quality upon applying the *autoreject* (*local*) algorithm does indeed improve. This is the data before the algorithm is applied.

[4] https://scikit-optimize.github.io

- *Sensor Noise Suppression (SNS)*: The SNS (De Cheveigné and Simon, 2008) algorithm, as described in the *Introduction*, projects the data of each sensor on to the subspace spanned by the principle components of all the other sensors. What it does is regressing out the sensor noise that cannot be explained by other sensors. It works on the principle that brain sources project on to multiple sensors but the noise is uncorrelated across sensors. In practice, not all the sensors are used for projection, but only a certain number of neighboring sensors (determined by the correlation in the data between the sensors).
- *Fully Automated Statistical Thresholding for EEG artifact Rejection (FASTER)*: It finds the outlier sensor using five different criteria: the variance, correlation, Hurst exponent, kurtosis and line noise. When the z-score of any of these criteria exceeds 3, the sensor is marked as bad according to that criteria. Note that even though FASTER is typically used as an integrated pipeline, here we use the bad sensor detection step, as this is what appears to dominate the bad signals in the case of the HCP data (Datasets section). We take a union of the sensors marked as bad by the different criteria and interpolate the data for those sensors.
- *Random Sample Consensus (RANSAC)*: We use the RANSAC implemented as part of the PREP pipeline (Bigdely-Shamlo et al., 2015). In fact, RANSAC (Fischler and Bolles, 1981) is a well-known approach used to fit statistical models in the presence of outliers in the data. In this approach, adopted for the use case of artifact detection in EEG, a subset of sensors (inliers) are sampled randomly (25% of the total sensors) and the data in all sensors are interpolated from these inliers sensors. This is repeated multiple times (50 in the PREP implementation) so as to yield a set of 50 time series for each sensor. The correlation between the median, computed instant by instant, of these 50 time series and the real data is computed. If this correlation is less than a threshold (0.75 in the PREP implementation), then the sensor is considered an outlier and therefore marked as bad. It is perhaps worth noting that unlike in the classical RANSAC algorithm, the inlier model is not learned from the data but instead determined from the physical interpolation. A sensor which is bad for more than 40% of the trials (the unbroken time) is marked as globally bad and interpolated. Even though the method was first proposed on EEG data only, we extended it for MEG data by replacing spline interpolation with field interpolation using spherical harmonics as implemented in MNE (Gramfort et al., 2013; Hämäläinen and Ilmoniemi, 1994). Note that this is the same interpolation method that is used by *autoreject* (*local*).

Datasets

We validated our methods on four open datasets with data from over 200 subjects. This allowed us to evaluate experimentally strengths and potential limitations of different rejection methods. The datasets contained either EEG or MEG data. To obtain solid experimental conclusions, diverse experimental paradigms were considered with data from working memory, perceptual and motor tasks.

We detail below how we defined $\bar{X}$ (*clean*), the cleaned ground-truth data for two of our datasets – HCP MEG and EEG faces data. This is perhaps one of the most challenging aspects of this work because the performance is evaluated on real data and not on simulations. An overview of all the datasets used in this study is provided in Table 2.

MNE sample data.—The MNE sample data (Gramfort et al., 2013) is a multimodal open dataset consisting of MEG and EEG data. It has been integrated as the default testing dataset into the development of the MNE software (Gramfort et al., 2013). The simultaneous M/EEG data were recorded at the Martinos Center of Massachusetts General Hospital. The MEG data with a Neuromag VectorView system, and an MEG-compatible cap comprising 60 electrodes was used for the EEG recordings. Data were sampled at 150 Hz. In the experiment, auditory stimuli (delivered monoaurally to the left or right ear) and visual stimuli (shown in the left or right visual hemifield) were presented in a random sequence with a stimulus onset asynchrony of 750 ms. The data was low-pass filtered at 40 Hz. The trials were 700 ms long including a 200 ms baseline period which was used for baseline correction.

EEGBCI dataset.—This is a 109-subject dataset (of which we analyzed 105 subjects which can be easily downloaded and analyzed using MNE-Python (Gramfort et al., 2013)) containing EEG data recording with a 64-sensor BCI2000 EEG cap (Schalk et al., 2004). Subjects were asked to perform different motor/imagery tasks while their EEG activity was recorded. In the related BCI protocol, each subject performed 14 runs, amounting to a total of 180 trials for hands and feet movements (90 trials each). The data was band-pass filtered between 1 and 40 Hz, and 700 ms long trials were constructed including a 200 ms pre-stimulus baseline period.

EEG faces data (OpenfMRI ds000117).—The OpenfMRI ds000117 dataset (Wakeman and Henson, 2015) contains multimodal task-related neuroimaging data over multiple runs for EEG, MEG and fMRI. For our analysis, we restrict ourselves to EEG data. The EEG data was recorded using a 70 channel Easycap EEG with electrode layout conforming to the 10–10% system. Subjects were presented with images of famous faces, unfamiliar faces and scrambled faces as stimuli. For each subject, on average, about 293 trials were available for famous and unfamiliar faces. The authors kindly provided us with run-wise bad sensor annotations which allowed us to conduct benchmarking against human judgement. To generate the ground truth evoked response $\bar{X}$ (*clean*), we randomly select 80 percent of the total number of trials in which famous and unfamiliar faces were displayed. In these trials, we interpolated the bad sensors run-wise. Then, we removed physiological artifacts (heart beat and eye blinks) using Independent Component Analysis (ICA) (Vigário et al., 2000). Following the ICA pipelines recommended by the MNE-Python software, the bad ICA components were marked automatically using cross-trial phase statistics (Dammers et al., 2008) for ECG (threshold=0.8) and adaptive z-scoring (threshold=3) for EOG components. The evoked response from the cleaned data $\bar{X}$ (*method*) is computed from the remaining 20 percent trials cleaned using either *autoreject* (*local*) or RANSAC (see Quantication of performance and comparison with state-of-the-art for a description of this method). Computing the ground-truth evoked potential from a large proportion of trials minimized the effect of outliers in the average. However, it is noteworthy that this choice of assigning fewer trials to the estimation with rejection algorithms acts in a conservative sense: each unnoticed bad trial may affect the ensuing evoked potentials more severely.

Human Connectome Project (HCP) MEG data.—The HCP dataset is a multimodal reference dataset realized by the efforts of multiple international laboratories around the world. It currently provides access to both task-free and task-related data for more than 900 human subjects with functional MRI data, 95 of which have presently also MEG (Larson-Prior et al., 2013). An interesting aspect of the initiative is that the data provided is not only in unprocessed BTi format, but also processed using diverse processing pipelines. These include annotations of bad sensors and corrupted time segments for the MEG data derived from automated pipelines and supplemented by human inspection. The automated pipelines are based on correlation between neighboring sensors, z-score metrics, ratio of variance to neighbors, and ICA decomposition. Most significant for our purposes, the clean average response $\overline{X}\ (clean)$ is directly available. It allows us to objectively evaluate the proposed algorithm against state-of-the-art methods by reprocessing the raw data and comparing the outcome with the official pipeline output.

The HCP MEG dataset provides access to MEG recordings from diverse tasks, i.e., a motor paradigm, passive listening and working memory. Here, we focused on the working memory task for which data is available for 83 subjects out of 95. A considerable proportion of subjects were genetically related, but we can ignore this information as the purpose of our algorithm is artifact removal rather than analyzing brain responses. For each subject two runs are available. Two classes of stimuli were employed, faces and tools. Here, we focused on the MEG data in response to stimulus onsets for the “faces” condition.

The MEG data were recorded with a wholehead MAGNES 3600 (4D Neuroimaging, San Diego, CA) in a magnetically shielded room at Saint Louis University. The system comprises 248 magnetometers and 23 reference sensors to capture environmental signals. Time windows precisely matched values used by the HCP “eravg” pipeline with onsets and offsets at −1.5 s and 2.5 s before and after the stimulus event, respectively. As in the HCP pipeline, signals were down-sampled to 508.63 Hz and band-pass filtered between 0.5–60 Hz. As it is commonly done with BTi systems, reference sensors at the periphery of the head were used to subtract away environmental noise. Given the linearity of Maxwell equations in the quasi-static regime, a linear regression model was employed. More precisely, signals from reference sensors are used as regressors in order to predict the MEG data of interest. The ensuing signal explained by the reference sensors in this model was then removed. The HCP preprocessing pipeline contains two additional steps: ICA was used to remove components not related to brain activity (including eye blinks and heart beats) and then bad trials and bad segments were removed with a combination of automated methods as well as annotations by a human observer. To have a fair comparison and focus on the latter step, the ICA matrices provided by the HCP consortium were applied to the data. We interpolated the missing sensors in $\overline{X}\ (clean)$ so that it has the same dimensions as the data from $\overline{X}\ (method)$. All the algorithms were executed separately on each run and the evoked response of the two runs was averaged to get $\overline{X}\ (method)$.

To enable easy access of the files along with compatibility in MNE-Python, we make use of the open source MNE-HCP package[5]. For further details on the HCP pipelines, the interested reader can consult the related paper by Larson-Prior et al. (2013) and the HCP S900 reference manual for the MEG3 release.

Results

We conducted qualitative and quantitative performance evaluations of autoreject using four different datasets comparing it to a baseline condition without rejection as well as three different alternative artifact rejection procedures.

Peak-to-peak thresholds

First, let us convince ourselves that the peak-to-peak thresholds indeed need to be learned. In Fig. 4A, we show a histogram of the thresholds learned on subjects in the EEGBCI dataset using *autoreject* (*global*). This figure shows that thresholds vary a lot across subjects. One could argue that this is due to variance in the estimation process. To rule out such a possibility, we plotted the distribution of maximum peak-to-peak thresholds as kernel density plots in Fig. 4C for three different subjects. Kernel density plots are non-parameteric methods to estimate the probability density function of a random variable (see for example Parzen, 1962). We can see that these distributions are indeed subject dependent, which is why a different threshold must be learned for each subject. In fact, if we were to use a constant threshold of 150 μV, in 17% of the subjects, all the trials would be dropped in one of the two conditions. Of course, from Fig. 4A, we can now observe that 150 μV is not really a good threshold to choose for many subjects.

We show here the maximum peak-to-peak amplitude per sensor because this is what decides if a trial should be dropped or not in the case of *autoreject* (*global*). Note that, if instead, we examined the distribution of peak-to-peak amplitudes across all sensors and trials, we would see a quasi-normal distribution. When all the sensors are taken together, a “smoothing” effect is observed in the distribution. This is a consequence of the central limit theorem. This also explains why we cannot learn a global threshold using all the peak-to-peak amplitudes across trials and sensors.

With the *autoreject* (*local*) approach, a threshold is estimated for each sensor separately. The histogram of thresholds for the MNE sample dataset is plotted in Fig. 4B. It shows that the threshold varies even across homogeneous MEG sensors. Fig. 4D shows the distribution of peak-to-peak thresholds for three different MEG sensors. This graph confirms actual sensor-level differences in amplitude distributions, which was also previously reported in the literature (Junghöfer et al., 2000). With this work, we go one step further by learning automatically the thresholds in a data-driven way rather than asking users to mark them interactively.

[5] http://mne-tools.github.io/mne-hcp/

Visual quality check

The average response plotted in a single graph, better known as "butterfly plots", constitutes a natural way to visually assess the performance of the algorithm for three different datasets – MNE sample data, HCP MEG data, and EEG faces data. In Fig. 5, the subplots in the left column show the evoked response with the bad sensors marked in red. Right subplots, show data after applying the *autoreject* (*local*) algorithm, with the repaired bad sensors in red. The algorithm works for different acquisition modalities – MEG and EEG, and even when multiple sensors are bad. A careful look at the results, show that autoreject (local) does not completely remove eyeblinks in the data as some of the blinks are time-locked to the evoked response. We will later discuss (c.f. Discussion) the possible solutions of applying ICA-based artifact correction in combination with *autoreject* (*local*).

Quantification of performance and comparison with state-of-the-art

We now compare these algorithms to *autoreject* (*local*) using the data quality metric defined in Eq. (9). We are interested not only in how the algorithms perform on average but at the level of individual subjects. To detail single subject performance, we present the data quality as scatter plots where each axis corresponds to the performance of a method. Fig. 6, contains results on the HCP MEG data. We can observe from the top-left subplot of the figure that *autoreject* (*local*) does indeed improve the data quality in comparison to the no rejection approach. In Fig. 6B, *autoreject* (*local*) is compared against SNS. The SNS algorithm focuses on removing noise isolated on single sensors. Its results can be affected by the presence of multiple bad sensors and globally bad trials. This explains why *autoreject* (*local*) outperforms SNS is this setting. In Fig. 6C, we compare against FASTER. Even though *autoreject* (*local*) is slightly worse than FASTER for a few subjects, FASTER is clearly much worse than autoreject (*local*) for at least 3 subjects, and *autoreject* (*local*) yields therefore less errors on average. Finally, Fig. 6D shows comparison to RANSAC. In the PREP implementation, this algorithm is not fully data-driven in the classic sense of RANSAC. This is due to the fact that the inlier model is not learned but rather derived from the physics of the interpolation. It is therefore an algorithm which is conceptually close to autoreject. However, a critical difference is that the parameters of this method still need to be tuned. This can be a problem as these parameters can be suboptimal on some datasets. Some experiments showed that it is for example the case for the EEG faces data, where it is possible to obtain better results by manually tuning the RANSAC parameters, rather than using the values proposed by the original authors.

Fig. 7 presents scatter plots for the EEG faces data. Here, we restrict our comparison to RANSAC and FASTER as they perform the best compared to autoreject on the HCP dataset. On this data, we apply the algorithms on both the conditions – famous and unfamiliar faces. It should be noted that the ground truth for this data was generated automatically with no additional annotations from human experts. However, a sanity check was performed on the ground truth by visual inspection. Here too, autoreject offers good results across all subjects, and even for the subjects for which RANSAC and FASTER underper-form.

The equivalent of Fig. 6 using the ℓ_2 norm rather than the ℓ_∞ is provided in supplementary materials.

Discussion

In this study, we have presented a novel artifact rejection algorithm called *autoreject* and assessed its performance on multiple datasets showing comparisons with other methods from the state-of-the-art.

We have shown that learning peak-to-peak rejection thresholds subject-wise is justified as the distribution of this statistic indeed varies considerably across subjects. We have shown qualitatively that *autoreject* yielded clean physiological event related field (ERF) and event related potentials (ERP) by correcting or rejecting contaminated data segments. Finally, we have shown quantitatively that autoreject yields results closer to the ground truth for more subjects than the algorithms presented in the section on competing methods. We now further discuss the conceptual similarities and differences of our approach to the alternative methods. We also discuss the interaction between autoreject and some other steps in the M/EEG analysis pipelines.

Autoreject vs. competing methods

We believe the key advantage of *autoreject* (*local*) over the other methods consists in combining data-driven parameter tuning with deterministic and physics-driven data interpolation. This interpolation promotes spatial smoothness of the electric potential on the scalp for EEG, and in the case of MEG, explicitly takes into account the well-understood Maxwell’s equations. To recapitulate, the sensor-level thresholds mark outlier segments across trials at the level of individual sensors, following a data augmentation step which exploits the full array of sensors. As trials are seen as independent observations, the thresholds can be therefore learned from the data using cross-validation. The cross-validation is stratified so that each fold contains roughly an equal proportion of the original and augmented trials. At repair time, bad segments are replaced with interpolated data from the good sensors. Of course, this is problematic if the sensor locations are not readily available. Fortunately, it turns out that the sensor positions from standard montages are often good enough for reliable interpolation.

In contrast to *autoreject* (*local*), SNS is a purely statistical method that does not take into account the physics of sensor locations for repairing the data. In SNS, the sensors are considered in a leave-one-sensor-out protocol. For each sensor, a “clean” subspace is defined from the principle components of the remaining sensors. The data from this sensor is then projected on to the “clean” subspace. As we have seen in Results (Fig. 6), this does not work satisfactorily, presumably because the SNS method makes strong assumptions regarding the orthogonality of the noise and “clean” subspace. The ensuing projection may not improve, and even deteriorate the signal in some cases. The consequence of this is what we observe empirically in Fig. 6. Applying SNS will also be problematic when multiple sensors are corrupted simultaneously. However, this is less of a problem in the HCP MEG data that we analyzed.

On the other hand, the FASTER method derives its rejection decisions from multiple signal characteristics. It uses criteria such as between-sensor correlation, variance and power spectrum, by considering their univariate Gaussian statistics with thresholds fixed to a z-score of 3. This default threshold appears to be satisfying as they work on a vast majority of subjects. However, the fact that it does not work as well on certain subjects can limit its adoption for large scale studies. Here, the adaptive nature of threshold detection performed by *autoreject* seems to be a clear advantage.

The RANSAC algorithm also performs adaptive outlier detection, but across sensors rather than trials. While *autoreject* (*local*) operates on segmented data such as trials time-locked to the stimuli, RANSAC was designed for continuous data without any segmentation. In fact, one could readily obtain bad sensor per trial (as illustrated in Fig. 2) even with RANSAC. However, the authors of the paper did not validate their method on continuous data, and hence, such a modification would require additional work. Although in the case of MEG data, this is not very crucial, this can in fact be critical for EEG data analysis. Remember, that in EEG, one often has to deal with locally bad sensors. And in this context, it is noteworthy that none of the other methods we have discussed so far provides an explicit treatment for single trial analysis in the presence of locally bad sensors. Our comparison to the RANSAC algorithm seems to suggest that the RANSAC algorithm is indeed sensitive to the parameter settings. Even though the default settings appear to work reasonably well for the EEG data (Fig. 7), they are not so optimal for the HCP MEG data (Fig. 6).

It is perhaps worth emphasizing that using cross-validation implies that the trials with artifacts are independent. If this assumption is violated and if artifacts are phase-locked between the training and validation sets, i.e. occur for all trials at the same time relative to trial onsets, then this can interfere with the estimation procedure in autoreject. Another caveat to be noted is that if the data contains more than ρ* (the maximum number of sensors that can be interpolated) bad sensors, and if the trial is not dropped, the data in the remaining bad sensors can still spread to other sensors if one were to use spatial filters such as SSP. Finally, *autoreject* considers only peak-to-peak thresholds for detecting bad sensors. Of course, the user must still mark low-amplitude flat sensors using another threshold; however, a simple threshold would suffice here as such sensors are usually consistently flat. Regardless of the method that researchers choose to adopt, diagnostic plots and automated reports (Engemann et al.,) are an essential element to assess and better understand possible failures of automatic procedures. In this regard, transparency of the method in question is important. In the case of our *autoreject* (*local*) implementation, we offer the possibility for the user to inspect the bad segments marked by the automated algorithm and correct it if necessary. An example of such a plot is shown in Fig. 8. Automating the detection of bad sensors and trials has the benefit of avoiding any unintentional biases that might be introduced if the experimenter were to mark the segments manually. In this sense, diagnostic visualization should supplement the analysis by ensuring accountability in the case of unexpected results.

Autoreject in the context of ICA, SSP and SSS

It is now important to place these results in the broader context of electrophysiological data analysis. Regarding the correction of specific artifacts such as electrooculogram (EOG) artifacts, *autoreject* (*local*) does indeed remove or interpolate some of the trials affected by eye blinks. This is because most eye blinks are not time-locked to the trial onsets and therefore get detected in the cross-validation procedure. However, the weaker eye blinks, particularly those smaller in magnitude than the evoked response, are not always removed. Also, the idea of rejection is to remove extreme values which are supposed to be rare events. This is why our empirical observation suggests that *autoreject* (*local*) is not enough in the presence of too frequent eye blinks, but also not enough to fully get rid of the smallest EOG artifacts.

This is where ICA (Vigário, 1997) and Signal Space Projection (SSP) (Uusitalo and Ilmoniemi, 1997) can naturally supplement autoreject. These methods are usually employed to extract and subsequently project out signal subspaces governed by physiological artifacts such as muscular, cardiac and ocular artifacts. However the estimation of these subspaces can be easily corrupted by other even more dramatic environmental or device-related artifacts. This is commonly prevented by band-pass filtering the signals and excluding high-amplitude artifacts during the estimation of the subspaces. Both ICA and SSP (particularly if it estimated from the data rather than an empty room recording) are highly sensitive to observations with high variance. Even though they involve estimating spatial filters that do not incorporate any notion of time, artifacts very localized in time will very likely have a considerable impact on the estimation procedure. This leads us to recommend removing globally bad sensors and employing appropriate rejection thresholds to exclude time segments with strong artifacts.

The success of applying *autoreject* to any electrophysiological data hinges critically on its ability to isolate artifacts local in time which cannot necessarily be identified by a prototypical spatial signature. However, the spatial interpolation employed by *autoreject* may not be able to repair sensors which are clustered together. In this case, the software package that implements the spatial interpolation should warn the user if the error due to the interpolation is likely to be high. Such a cluster of bad sensors can be expected in the case of physiological artifacts, such as muscular, cardiac or ocular artifacts. To take care of such artifacts with prototypical spatial patterns, ICA is certainly a powerful method, yet manual identification of artifactual components remains today done primarily manually.

If the context of data processing supports estimation of ICA and SSP on segmented data, we would recommend to perform it after applying *autoreject*, benefiting from its automated bad sensor and bad trial handling. MEG signals usually contain a strong contribution from environmental electromagnetic fields. Therefore, interference suppression of MEG data is often needed, utilizing hardware and software based approaches (see, e.g. Parkkonen (2010) for details). In principle, spatial interpolation of bad sensor signals may not work very well unless the environmental interference has been removed. In the present study, the MNE sample data was recorded in a very well shielded room and did not need separate interference suppression, while the interference in the 4D/BTi data was removed by utilizing the reference channels. Spatial filtering approaches, such as SSP or SSS, may however produce a “chicken and egg” dilemma – whether to apply SSP/SSS or autoreject first - which can be solved using an iterative procedure as suggested by the PREP pipeline (Bigdely-Shamlo et al., 2015). That is, first run autoreject only for detection of bad channels but without interpolation. This is followed by an SSS run excluding the bad channels detected by autoreject. Finally, autoreject can be applied on the data free of environmental interference.

Source localization with artifact rejection

Obviously, artifact-free data benefits almost any analysis that is subsequently performed and the M/EEG inverse problem is no exception. Such benefits not only concern the quality of source estimates but also the choice of source-localization methods, as some of these methods require modification when certain artifact rejection strategies are employed. As *autoreject* amounts to automating a common, pre-existing and early processing step it does not require changes for source-level analyses. For example, evoked responses obtained using *autoreject* (*local*) can be readily analyzed with various source localization methods such as beamformer methods (Dalal et al., 2008; Groß et al., 2001), or cortically-constrained Minimum Norm Estimates with ℓ_2 penalty (Uutela et al., 1999), and noise-normalized schemes, such as dSPM (Dale et al., 2000) and sLORETA (Pascual-Marqui et al., 2002).

Certain denoising techniques such as SSP (Uusitalo and Ilmoniemi, 1997) or SSS (Taulu et al., 2004) reduce the rank of the data which can be problematic for beamforming techniques (Woolrich et al., 2011). This needs special attention, and in some software such as MNE, this is handled using a non-square whitening matrix. However, as *autoreject* does not systematically reduce the rank of the data, it does not even require sophisticated handling of the data rank. At the same time, it works seamlessly with noise-normalization, where the estimation of the between-sensor noise covariance depends on the number of trials. To estimate the noise covariance during baseline periods, one computes the covariance of non-averaged data and then, assuming independence of each trial, the covariance gets divided by the number of trials present in the average (Engemann and Gramfort, 2015). Most existing pipelines scale the covariance by an integer number of trials. In contrast, methods such as robust regression (Diedrichsen and Shadmehr, 2005) that preferentially give less weight to noisy trials, require the noise normalization to be modified. Concretely, one would have to estimate an approximate number of trials or estimate the covariance matrix by restricting the computation to a subset of trials. *Autoreject* does not necessitate any such modifications to the source-localization pipeline, and hence, helps reduce the cognitive load of integration with pre-existing tools.

Conclusion

In summary, we have presented a novel algorithm for automatic data-driven detection and repair of bad segments in single trial M/EEG data. We therefore termed it autoreject. We have compared our method to state-of-the-art methods on four different open datasets containing in total more than 200 subjects. Our validation suggests that *autoreject* performs at least as good as diverse alternatives and commonly used procedures while often performing considerably better. This is the consequence of the combination of a data-driven outlier-detection step combined with physics-driven channel repair where all parameters are calibrated using a cross-validation strategy robust to outliers. The insight about the necessity to tune parameters at the level of single sensors and for individual subjects was further consolidated by our analyses of threshold distributions. The empirical variability of optimal thresholds across datasets emphasizes the importance of statistical learning approaches and automatic model selection strategies for preprocessing M/EEG signals. While *autoreject* makes use of advanced statistical learning techniques such as Bayesian hyperpara-meter optimization, it is also grounded in the physics underlying the data generation. It is therefore not purely a black-box data-driven approach. It balances the trade-off between accuracy and interpretability. Indeed all *autoreject* parameters have a meaning from a user standpoint and the algorithmic decisions can be explained. Supplemented by efficient diagnostic visualization routines, *autoreject* can be easily integrated in MEG/EEG analysis pipelines, including clinical ones where understanding algorithmic decisions is mandatory for tool adoption.

By offering an automatic and data-driven algorithmic solution to a task mostly so far done manually, *autoreject* reduces the cost of data inspection by experts. By allowing to repair data rather than removing it from the study, it allows saving data which are also costly to acquire. In addition, it removes the experts' bias which are due to specific training or prior experience, as well as some expectations about the data. It does so by defining a clear set of rules serving as inclusion criteria for M/EEG data, making results more easily reproducible and eventually limiting the risk of false discoveries. Furthermore, as data sharing across centers has become a common practice, *autoreject* addresses the issue of heterogeneous acquisition setups. Indeed, each acquisition set-up has its intrinsic signal qualities, which means that preprocessing parameters can vary significantly between datasets. As opposed to alternative methods, *autoreject* automates the estimation of its parameters.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgement

We thank Lionel Naccache for providing us with dramatic examples of artifact-ridden clinical EEG data which considerably stimulated the research presented in this study. The work was supported by the French National Research Agency (ANR-14-NEUC-0002-01), the National Institutes of Health (R01 MH106174) and ERC Starting Grant SLAB ERC-YStG-676943. Denis A. Engemann acknowledges support by the Amazon Webservices Research Grant awarded to him and the ERC StG 263584 awarded to Virginie van Wassenhove. We thank the Cognitive Neuroimaging Unit at Neurospin for fruitful discussions and feedback on this work. We further thank the MNE-Python developers and the MNE community for continuous collaborative interaction on basic development of the academic MNE software without which this study could not have been conducted.

References

Barachant A, Andreev A, Congedo M, 2013 The Riemannian Potato: an automatic and adaptive artifact detection method for online experiments using Riemannian geometry. in: TOBI Workshop lV, pp. 19–20.

Basirat A, Dehaene S, Dehaene-Lambertz G, 2014 A hierarchy of cortical responses to sequence violations in three-month-old infants. Cognition 132, 137–150. [PubMed: 24809742]

Bergstra JS, Bardenet R, Bengio Y, Kégl B, 2011 Algorithms for hyper-parameter optimization. in: Advances in Neural Information Processing Systems, pp. 2546–2554.

Bigdely-Shamlo N, Kreutz-Delgado K, Robbins K, Miyakoshi M, Westerfield M, Bel-Bahar T, Kothe C, Hsi J, Makeig S, 2015 Hierarchical event descriptor (HED) tags for analysis of event-related EEG studies. in: Global Conference on Signal and Information Processing (GlobalSIP), IEEE, pp. 1–4.

Bigdely-Shamlo N, Mullen T, Kothe C, Su K-M, Robbins K, 2015 The PREP pipeline: standardized preprocessing for large-scale EEG analysis. Front. Neuroinform 9.

Cruse D, Chennu S, Chatelle C, Bekinschtein TA, Fernández-Espejo D, Pickard JD, Laureys S, Owen AM, 2012 Bedside detection of awareness in the vegetative state: a cohort study. The Lancet 378, 2088–2094.

Dalal SS, Guggisberg AG, Edwards E, Sekihara K, Findlay AM, Canolty RT, Berger MS, Knight RT, Barbaro NM, Kirsch HE, et al., 2008 Five-dimensional neuroimaging: localization of the time-frequency dynamics of cortical activity. NeuroImage 40, 1686–1700. [PubMed: 18356081]

Dale AM, Liu AK, Fischl BR, Buckner RL, Belliveau JW, Lewine JD, Halgren E, 2000 Dynamic statistical parametric mapping: combining fMRI and MEG for high-resolution imaging of cortical activity. Neuron 26, 55–67. [PubMed: 10798392]

Dammers J, Schiek M, Boers F, Silex C, Zvyagintsev M, Pietrzyk U, Mathiak K, 2008 Integration of amplitude and phase statistics for complete artifact removal in independent components of neuromagnetic recordings. IEEE Trans. Biomed. Eng 55, 2353–2362. [PubMed: 18838360]

De Brabanter J, Pelckmans K, Suykens J, Vandewalle J, De Moor B, 2003 Robust cross-validation score functions with application to weighted least squares support vector machine function estimation. Technical Report, K.U. Leuven.

De Cheveigné A, Simon J, 2008 Sensor noise suppression. J. Neurosci. Methods 168, 195–202. [PubMed: 17963844]

Delorme A, Makeig S, 2004 EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. J. Neurosci. Methods 134, 9–21. [PubMed: 15102499]

Diedrichsen J, Shadmehr R, 2005 Detecting and adjusting for artifacts in fMRI time series data. NeuroImage 27, 624–634. [PubMed: 15975828]

Engemann DA, Gramfort A, 2015 Automated model selection in covariance estimation and spatial whitening of MEG and EEG signals. NeuroImage 108, 328–342. [PubMed: 25541187]

Engemann D, Raimondo F, King J, Jas M, Gramfort A, Dehaene S, Naccache L, Sitt J, 2015 Automated measurement and prediction of consciousness in vegetative state and minimally conscious patients., in: Workshop on Statistics, Machine Learning and Neuroscience at the International Conference on Machine Learning (ICML), Lille.

Fischler MA, Bolles RC, 1981 Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, 381–395.

Goldberger A, Amaral L, Glass L, Hausdorff J, Ivanov P, et al., 2000 Physiobank, physiotoolkit, and physionet components of a new research resource for complex physiologic signals. Circulation 101, e215–e220. [PubMed: 10851218]

Goldfine AM, Bardin JC, Noirhomme Q, Fins JJ, Schiff ND, Victor JD, 2013 Reanalysis of bedside detection of awareness in the vegetative state: a cohort study. Lancet 381, 289. [PubMed: 23351802]

Gorgolewski KJ, Poldrack R, 2016 A practical guide for improving transparency and reproducibility in neuroimaging research. bioRxiv, 039354.

Gorgolewski KJ, Auer T, Calhoun VD, Craddock RC, Das S, Duff EP, Flandin G, Ghosh SS, Glatard T, Halchenko YO, et al., 2016 The brain imaging data structure: a standard for organizing and describing outputs of neuroimaging experiments. bioRxiv, 034561.

Gramfort A, Luessi M, Larson E, Engemann D, Strohmeier D, et al., 2013 MEG and EEG data analysis with MNE-Python. Front. Neurosci 7.

Gramfort A, Luessi M, Larson E, Engemann D, Strohmeier D, et al., 2014 MNE software for processing MEG and EEG data. NeuroImage 86, 446–460. [PubMed: 24161808]

Groß J, Kujala J, Hämäläinen M, Timmermann L, Schnitzler A, Salmelin R, 2001 Dynamic imaging of coherent sources: studying neural interactions in the human brain. In: Proceedings of the National Academy of Sciences, 98, 694–699.

Hämäläinen M, Ilmoniemi R, 1994 Interpreting magnetic fields of the brain: minimum norm estimates. Med. Biol. Eng. Comput 32, 35–42. [PubMed: 8182960]

Hsu C-W, Chang C-C, Lin C-J, et al., 2003 A practical guide to support vector classification.

Jas M, Engemann D, Raimondo F, Bekhti Y, Gramfort A, 2016 Automated rejection and repair of bad trials in MEG/EEG. in: 6th International Workshop on Pattern Recognition in Neuroimaging (PRNI).

Junghöfer M, Elbert T, Tucker DM, Rockstroh B, 2000 Statistical control of artifacts in dense array EEG/MEG studies. Psychophysiology 37, 523–532. [PubMed: 10934911]

Krizhevsky A, Sutskever I, Hinton GE, 2012 Imagenet classification with deep convolutional neural networks. in: Advances in Neural Information Processing Systems, pp. 1097–1105.

Larson-Prior LJ, Oostenveld R, Della Penna S, Michalareas G, Prior F, Babajani-Feremi A, Schoffelen J-M, Marzetti L, de Pasquale F, Pompeo F.Di, et al., 2013 Adding dynamics to the Human Connectome Project with MEG. NeuroImage 80, 190–201. [PubMed: 23702419]

Leung D, 2005 Cross-validation in nonparametric regression with outliers. Ann. Stat, 2291–2310.

Litvak V, Mattout J, Kiebel S, Phillips C, Henson R, Kilner J, Barnes G, Oostenveld R, Daunizeau J, Flandin G, et al., 2011 EEG and MEG data analysis in SPM8. Computational intelligence and neuroscience, 2011.

Niso G, Rogers C, Moreau JT, Chen L-Y, Madjar C, Das S, Bock E, Tadel F, Evans AC, Jolicoeur P, et al., 2016 OMEGA: the open MEG archive. NeuroImage 124, 1182–1187. [PubMed: 25896932]

Nolan H, Whelan R, Reilly R, 2010 FASTER: fully automated statistical thresholding for EEG artifact rejection. J. Neurosci. Methods 192, 152–162. [PubMed: 20654646]

Ollier W, Sprosen T, Peakman T, 2005 UK Biobank: from concept to reality.

Oostenveld R, Fries P, Maris E, Schoffelen J-M, 2011 FieldTrip: open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. Computational intelligence and neuroscience, 2011.

Parkkonen L, 2010 Instrumentation and data preprocessing In: Hansen Peter KM, Salmelin R (Eds.), MEG: an introduction to methods, Oxford University Press, New York, 2010.

Parzen E, 1962 On estimation of a probability density function and mode. Ann. Math. Stat 33, 1065–1076.

Pascual-Marqui RD, et al., 2002 Standardized low-resolution brain electromagnetic tomography (sLORETA): technical details. Methods Find Exp. Clin. Pharmacol 24, 5–12. [PubMed: 12575463]

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E, 2011 Scikit-learn: Machine learning in Python. J. Mach. Learn. Res 12, 2825–2830.

Perrin F, Pernier J, Bertrand O, Echallier J, 1989 Spherical splines for scalp potential and current density mapping. Electroen. Clin. Neuro 72, 184–187.

Rasmussen CE, Williams CK, 2006 Gaussian processes for machine learning. 2006, The MIT Press, Cambridge, MA, USA, 38 715–719.

Schalk G, McFarland DJ, Hinterberger T, Birbaumer N, Wolpaw JR, 2004 BCI2000: a general-purpose brain-computer interface (BCI) system. IEEE Trans. Biomed. Eng 51, 1034–1043. [PubMed: 15188875]

Snoek J, Larochelle H, Adams RP, 2012 Practical bayesian optimization of machine learning algorithms. in: Advances in Neural Information Processing Systems, pp. 2951–2959.

Tadel F, Baillet S, Mosher JC, Pantazis D, Leahy RM, 2011 Brainstorm: a user-friendly application for MEG/EEG analysis. Comput. Intell. Neurosci 2011, 8.

Taulu S, Kajola M, Simola J, 2004 Suppression of interference and artifacts by the signal space separation method. Brain Topogr. 16, 269–275. [PubMed: 15379226]

Taylor JR, Williams N, Cusack R, Auer T, Shafto MA, Dixon M, Tyler LK, Henson RN, et al., 2015 The Cambridge Centre for Ageing and Neuroscience (Cam-CAN) data repository: structural and functional MRI, MEG, and cognitive data from a cross-sectional adult lifespan sample, NeuroImage.

Uusitalo M, Ilmoniemi R, 1997 Signal-space projection method for separating MEG or EEG into components. Med. Biol. Eng. Comput 35, 135–140. [PubMed: 9136207]

Uutela K, Hämäläinen M, Somersalo E, 1999 Visualization of magnetoencephalographic data using minimum current estimates. NeuroImage 10, 173–180. [PubMed: 10417249]

Van Essen D, Ugurbil K, Auerbach E, Barch D, Behrens T, Bucholz R, Chang A, Chen L, Corbetta M, Curtiss S, et al., 2012 The Human Connectome Project: a data acquisition perspective. NeuroImage 62, 2222–2231. [PubMed: 22366334]

Vigário R, Sarela J, Jousmiki V, Hamalainen M, Oja E, 2000 Independent component approach to the analysis of EEG and MEG recordings. IEEE Trans. Biomed. Eng 47, 589–593. [PubMed: 10851802]

Vigário R, 1997 Extraction of ocular artefacts from EEG using independent component analysis. Electroen. Clin. Neuro 103, 395–404.

Wakeman D, Henson R, 2015 A multi-subject, multi-modal human neuroimaging dataset. Sci. Data 2.

Widmann A, Schröger E, Maess B, 2015 Digital filter design for electrophysiological data-a practical approach. J. Neurosci. Methods 250, 34–46. [PubMed: 25128257]

Woolrich M, Hunt L, Groves A, Barnes G, 2011 MEG beamforming using Bayesian PCA for adaptive data covariance matrix regularization. Neuroimage 57, 1466–1479. [PubMed: 21620977]

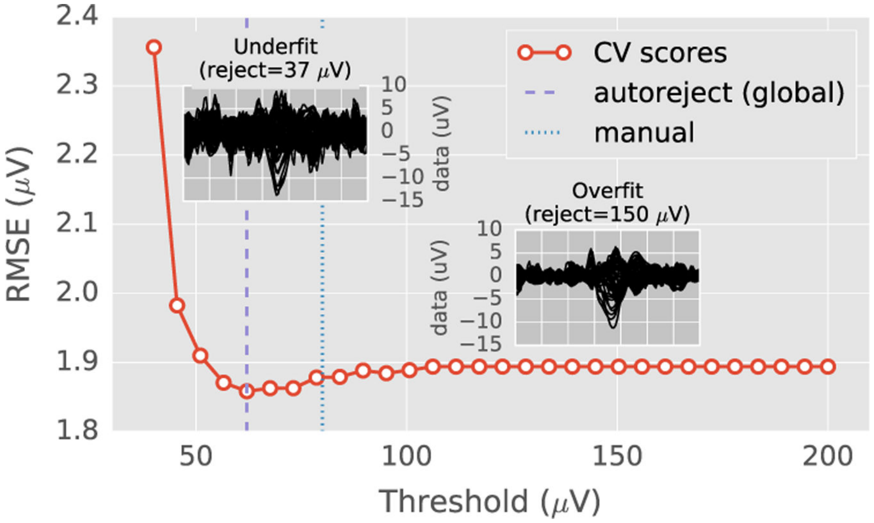

Fig. 1.

Cross-validation error as a function of peak-to-peak rejection threshold on one EEG dataset. The root mean squared error (RMSE) between the mean of the training set (after removing the trials marked as bad) and the median of the validation set was used as the cross-validation metric (Autoreject (global)). The two insets show the average of the trials as “butterfly plots” (each curve representing one sensor) for very low and high thresholds. For low thresholds, the RMSE is high because most of the trials are rejected (underfit). At high thresholds, the model does not drop any trials (overfit). The optimal data-driven threshold (*autoreject*, *global*) with minimum RMSE is somewhere in between. It closely matches the human threshold.

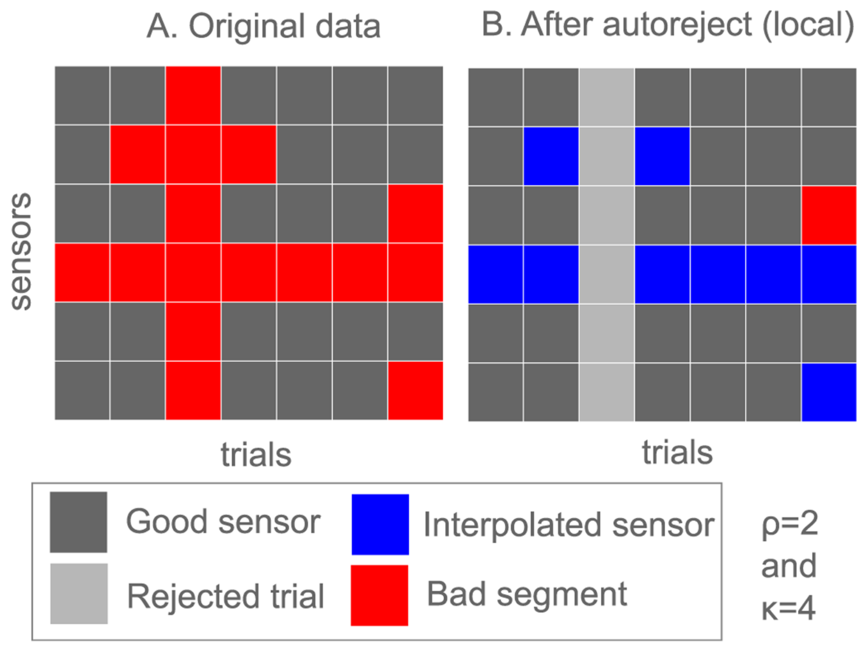

Fig. 2.

A schematic diagram explaining how *autoreject* (*local*) works. (A) Each cell here is an element of the indicator matrix C_{ij} described in the section on Autoreject (local). Sensor-level thresholds are found and bad segments are marked for each sensor. Bad segments shown in red are where $C_{ij}=1$ (B) Trials are rejected if the number of bad sensors is greater than κ and otherwise, the worst ρ sensors are interpolated.

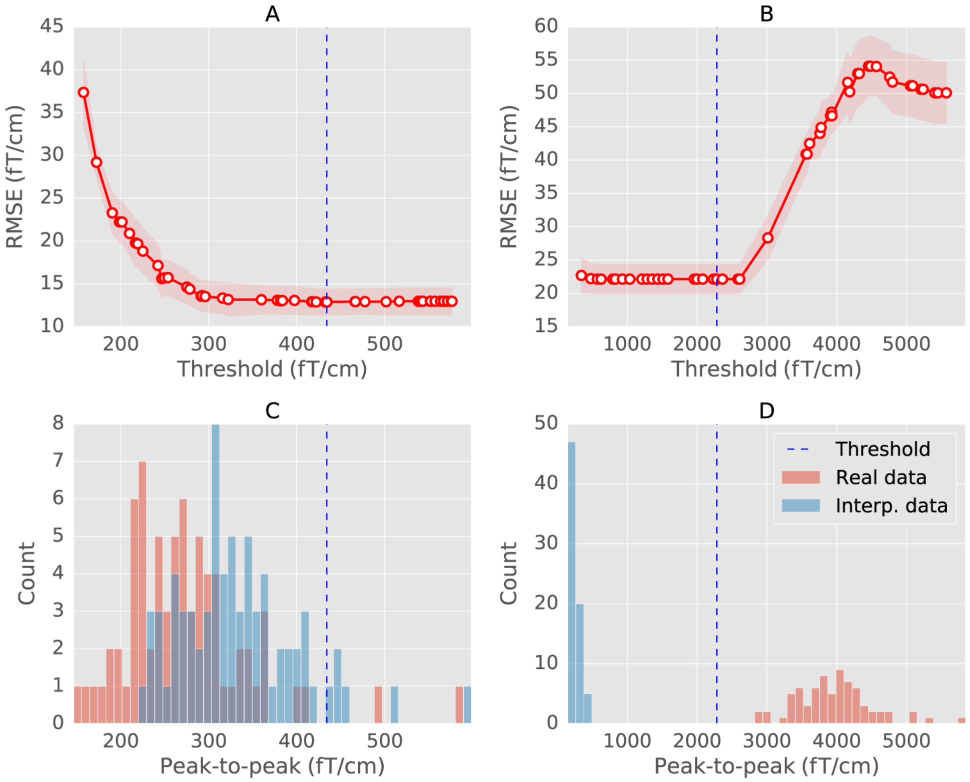

Fig. 3.

(A) and (B) The cross-validation curve obtained with sequential Bayesian optimization (see section on Bayesian optimization for an explanation) for a regular (MEG 2523) and a globally bad sensor (MEG 2443) from the MNE sample dataset. The mean RMSE is shown in red circles with error bounds in red shades. The red shaded region shows the lower and upper bounds between which the optimization is carried out. Vertical dashed line marks the estimated threshold. (C) and (D) Histogram of peak-to-peak amplitudes of trials in the sensor. The histograms are computed separately for the real data (red) and the data interpolated from other sensors (blue). The estimated threshold correctly marks all the trials as bad for the globally bad sensor.

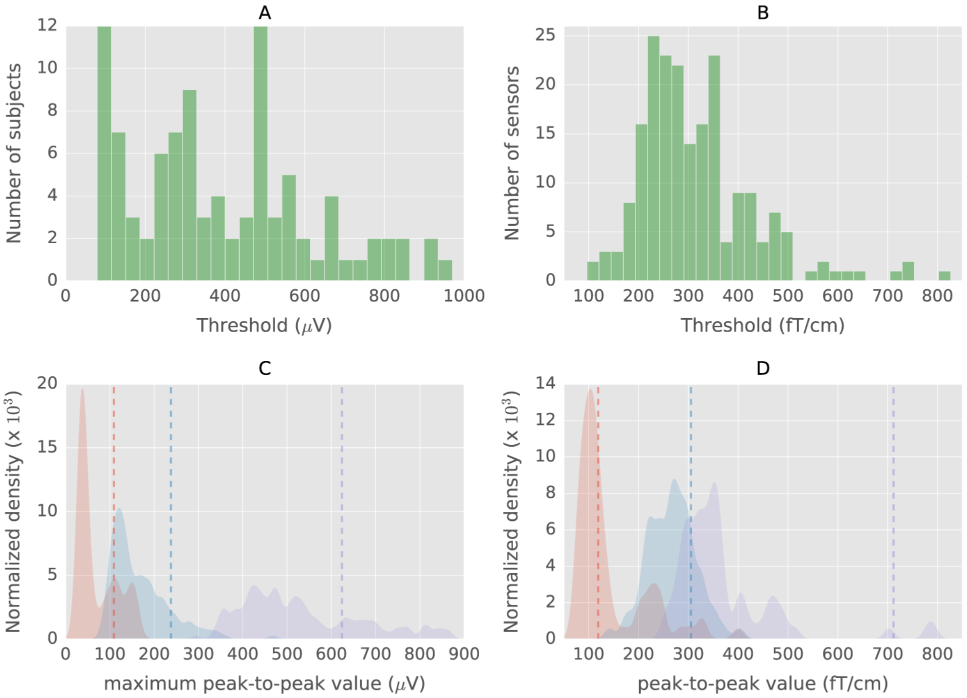

Fig. 4.

A. Histogram of thresholds for subjects in the EEGBCI dataset with *autoreject* (*global*) B. Histogram of sensor-specific thresholds in gradiometers for the MNE sample dataset (see Results). C. Normalized kernel density plots of maximum peak-to-peak value across sensors for three subjects in the EEGBCI data. Vertical dashed lines indicate estimated thresholds. Density plots and thresholds corresponding to the same subject are the same color. D. Normalized Kernel Density plots of peak-to-peak values for three MEG sensors in the MNE sample dataset. The threshold indeed has to be different depending on the data (subject and sensor).

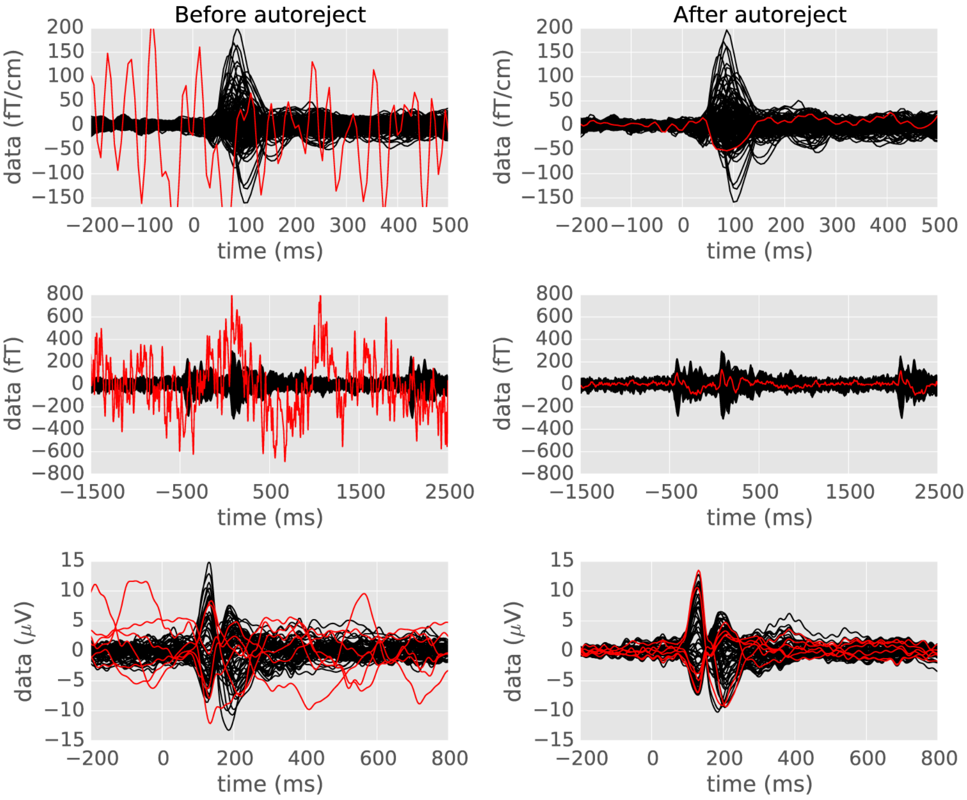

Fig. 5.

The evoked response (average of data across trials) on three different datasets before and after applying *autoreject* — the MNE sample data, the HCP data and the EEG faces data. Each sensor is a line on the plots. On the left, manually annotated bad sensors are shown in red. The algorithm finds the bad sensors automatically and repairs them for the relevant trials. Note that it can even fix multiple sensors at a time and works for different modalities of data acquisition.

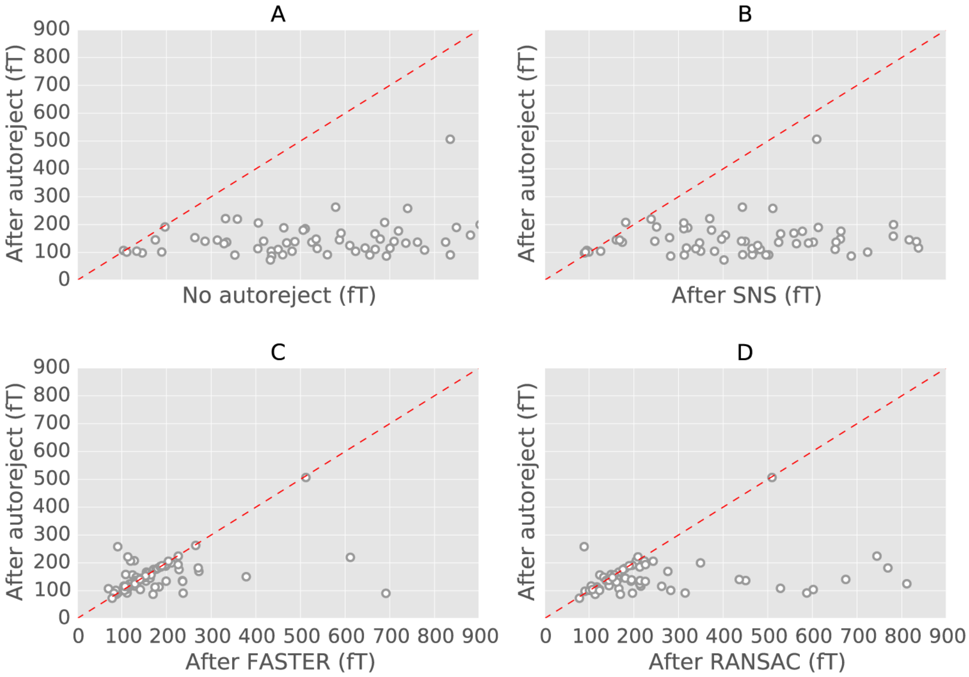

Fig. 6.

Scatter plots for the results with the HCP data. For each method, the $\|\cdot\|_\infty$ norm of the difference between the HCP ground truth and the method is taken. Each circle is a subject. (A) *autoreject* (*local*) against no rejection, (B) *autoreject* (*local*) against Sensor Noise Suppression (SNS) (SNS), (C) *autoreject* against FASTER, (D) *autoreject* (*local*) against RANSAC. Data points below the dotted red line indicate subjects for which *autoreject* (*local*) outperforms the alternative method.

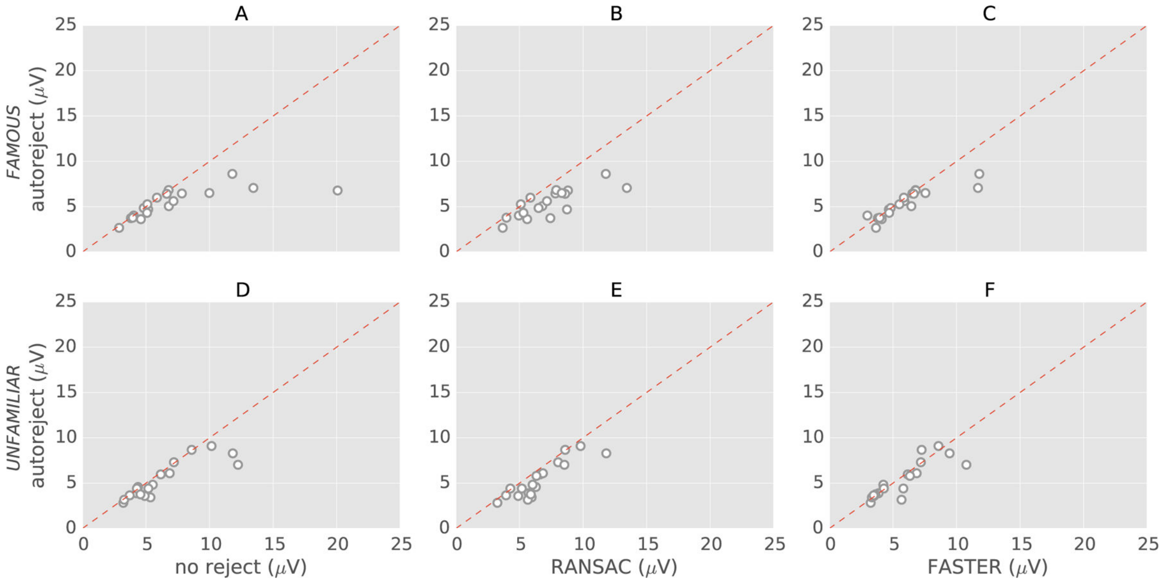

Fig. 7.

Scatter plots for the results with the 19 subjects from Faces dataset. The plots in the first row: (A), (B) and (C) are for the condition “famous”, whereas the plots in the second row: (D), (E) and (F) are for the condition “unfamiliar” faces. For each method, the $\|\cdot\|_\infty$ norm of the difference between the ground truth and the estimates is computed. Each circle is a subject. Data points below the dotted red line indicate subjects for which *autoreject* (*local*) outperforms the alternative method.

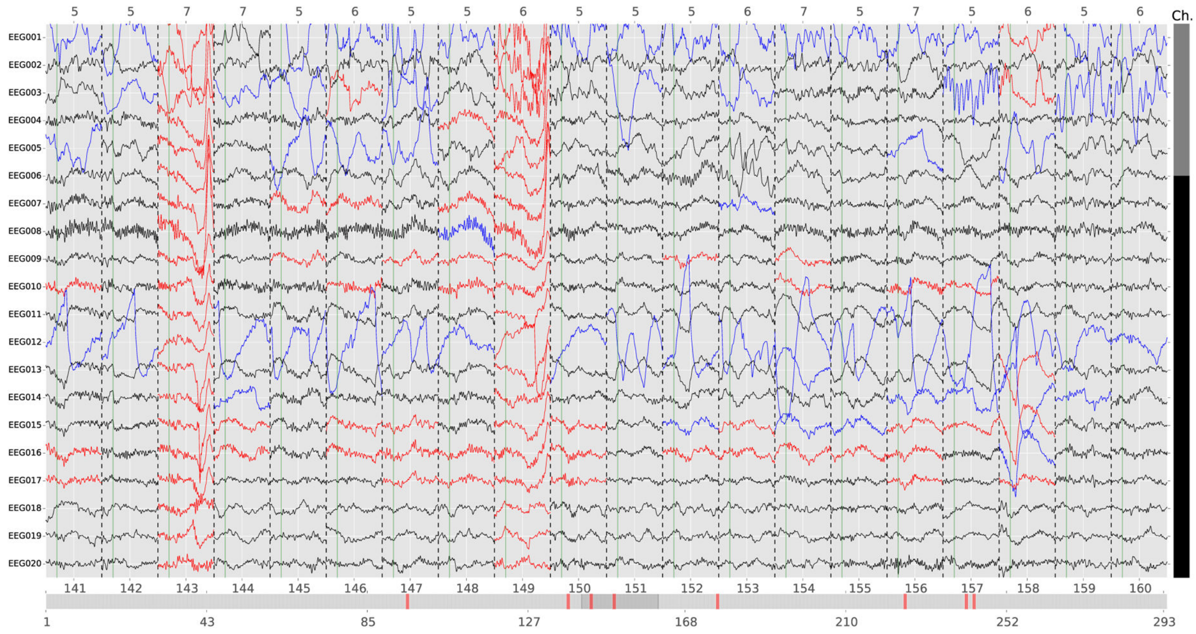

Fig. 8.

An example diagnostic plot from an interactive viewer with autoreject (local). The data plotted here is subject 16 for the condition ‘famous’ in the EEG faces data. Each row is a different sensor. The trials are concatenated along the x axis with dotted vertical lines separating consecutive trials. Each trial is numbered at the bottom and its corresponding trigger code is at the top. The horizontal scroll bar at the bottom allows browsing trials and the vertical scroll bar on the right is for browsing sensors. A trial which is marked as bad is shown in red on the horizontal scroll bar and the corresponding column for the trial is also red. A data segment in a good trial is either i) Good (in black) ii) Bad and interpolated (blue), or iii) Bad but not interpolated (in red). Note that the worst sensors in a trial are typically interpolated.

Table 1

Overview of rejection strategies evaluated.

method	statistical scope	parameter defaults
FASTER[a]	univariate	threshold on zscore = 3
SNS[b]	multivariate	number of neighbors=8
RANSAC[c]	multivariate outlier detection	# resamples=50, fraction of channels=0.25, threshold on correlation=0.75, unbroken time=0.4
autoreject	univariate with cross-validation	sensor-level thresholds, ρ and κ; learned from data

[a] Nolan et al. (2010).

[b] De Cheveigné and Simon (2008).

[c] Bigdely-Shamlo et al. (2015).

Table 2

Overview of datasets analyzed.

Algorithm	Dataset	Acquisition device	Sensors used	# subjects
autoreject (global)	MNE sample data	Neuromag VectorView	60 EEG electrodes	1
	EEGBCI	BCI2000 cap	64 EEG electrodes	105
autoreject (local)	MNE sample data	Neuromag VectorView	60 EEG electrodes	1
	EEG faces	Neuromag VectorView	60 EEG electrodes	19
	HCP working memory	4D Magnes 3600 WH	248 magnetometers	83