

1 Precise temporal localisation of M/EEG effects with  
2 Bayesian generalised additive multilevel models

3 Ladislas Nalborczyk<sup>1</sup> and Paul Bürkner<sup>2</sup>

4 <sup>1</sup>Aix Marseille Univ, CNRS, LPL

5 <sup>2</sup>TU Dortmund University, Department of Statistics

6 Abstract

7 Time-resolved electrophysiological measurements such as those obtained through magneto- and electroencephalography (M/EEG) offer a unique window onto the neural activity underlying cognitive processes. Researchers are often interested in determining whether and when these signals differ across experimental conditions or participant groups. The conventional approach involves mass-univariate statistical testing across time and space followed by corrections for multiple comparisons or some form of cluster-based inference. While effective for controlling error rates at the cluster-level, cluster-based inference comes with a significant limitation: by shifting the focus of inference from individual time points to clusters, it prevents drawing conclusions about the precise onset or offset of observed effects. Here, we present a *model-based* alternative for analysing M/EEG timeseries, such as event-related potentials or time-resolved decoding accuracy. Our approach leverages Bayesian generalised additive multilevel models, providing posterior odds that an effect exceeds zero (or chance) at each time point, while naturally accounting for temporal dependencies and between-subject variability. Using both simulated and empirical M/EEG datasets, we show that this approach substantially outperforms conventional methods in estimating the onset and offset of neural effects, yielding more precise and reliable estimates. We provide an open-source R package implementing the method and describe how it can be integrated into M/EEG analysis pipelines using MNE-Python.

*Keywords:* EEG, MEG, cluster-based inference, multiple comparisons, generalised additive models, mixed-effects models, multilevel models, Bayesian statistics, brms

8 **Table of contents**

9 <b>Introduction</b>	<b>3</b>
10 <b>1 Introduction</b>	<b>3</b>
11 <b>1.1 Problem statement</b> . . . . .	3

12	1.2 Statistical errors and cluster-based inference . . . . .	4
13	1.3 Previous work on modelling M/EEG data . . . . .	5
14	1.4 Generalised additive models . . . . .	6
15	1.5 Objectives . . . . .	7
16	<b>2 Benchmarking with known ground truth</b>	8
17	2.1 Methods . . . . .	8
18	2.1.1 M/EEG data simulation . . . . .	8
19	2.1.2 Model description and model fitting . . . . .	9
20	2.1.3 Comparing the onset/offset estimates across approaches . . . . .	11
21	2.1.4 Simulation study . . . . .	12
22	2.2 Results . . . . .	12
23	<b>3 Application to actual MEG data</b>	13
24	3.1 Methods . . . . .	13
25	3.2 Results . . . . .	14
26	<b>4 Discussion</b>	16
27	<b>Data and code availability</b>	18
28	<b>Packages</b>	18
29	<b>Acknowledgements</b>	18
30	<b>References</b>	19
31	<b>Application to 2D time-resolved decoding results (cross-temporal generalisation)</b>	25
32	<b>How to choose the GAM basis dimension?</b>	28
33	<b>R package and integration with MNE-Python</b>	29

---

Ladislas Nalborczyk  <https://orcid.org/0000-0002-7419-9855>

Paul Bürkner  <https://orcid.org/0000-0001-5765-8995>

The authors have no conflicts of interest to disclose.

Correspondence concerning this article should be addressed to Ladislas Nalborczyk, Aix Marseille Univ, CNRS, LPL, 5 avenue Pasteur, 13100 Aix-en-Provence, France, email: [ladislas.nalborczyk@cnrs.fr](mailto:ladislas.nalborczyk@cnrs.fr)

34 **Precise temporal localisation of M/EEG effects with Bayesian generalised additive  
35 multilevel models**

1 **1 Introduction**

2 **1.1 Problem statement**

3 Understanding the temporal dynamics of cognitive processes requires methods that can capture  
4 fast-changing neural activity with high temporal resolution. Magnetoencephalography and elec-  
5 troencephalography (M/EEG) are two such methods, widely used in cognitive neuroscience for  
6 their ability to track brain activity at the millisecond scale. These techniques provide rich time  
7 series data that reflect how neural responses unfold in response to stimuli or tasks. A central  
8 goal in many M/EEG studies is to determine whether, when, and where neural responses differ  
9 across experimental conditions or groups.

10 The conventional approach involves mass-univariate statistical testing through time and/or  
11 space followed by some form of correction for multiple comparisons with the goal of maintaining  
12 the familywise error rate (FWER) or the false discovery rate (FDR) at the nominal level (e.g.,  
13 5%). Cluster-based inference is the most common way of achieving this sort of error control  
14 in the M/EEG literature, being the recommended approach in several software programs (e.g.,  
15 [EEGlab](#), [Delorme & Makeig, 2004](#); [MNE-Python](#), [Gramfort, 2013](#)). While effective for controlling  
16 error rates, cluster-based inference comes with a significant limitation: by shifting the focus of  
17 inference from individual datapoints (e.g., timesteps, sensors, voxels) to clusters, it prevents the  
18 ability to draw precise conclusions about the spatiotemporal localisation of such effects ([Maris  
19 & Oostenveld, 2007](#); [Sassenhagen & Draschkow, 2019](#)). As pointed out by Maris & Oostenveld  
20 ([2007](#)): “there is a conflict between this interest in localized effects and our choice for a global  
21 null hypothesis: by controlling the FA [false alarm] rate under this global null hypothesis, one  
22 cannot quantify the uncertainty in the spatiotemporal localization of the effect”. Even worse,  
23 Rosenblatt et al. ([2018](#)) note that cluster-based inference suffers from low spatial resolution:  
24 “Since discovering a cluster means that ‘there exists at least one voxel with an evoked response  
25 in the cluster’, and not that ‘all the voxels in the cluster have an evoked response’, it follows that  
26 the larger the detected cluster, the less information we have on the location of the activation.”  
27 As a consequence, cluster-based inference is expected to perform poorly for identifying the onset  
28 of M/EEG effects; a property that was later demonstrated in simulation studies (e.g., [Rousselet,  
29 2025](#); [Sassenhagen & Draschkow, 2019](#)).

30 To overcome the limitations of cluster-based inference, we introduce a novel *model-based* ap-  
31 proach for precisely localising M/EEG effects in time, space, and other dimensions. The pro-  
32 posed approach, based on Bayesian generalised additive multilevel models, allows quantifying  
33 the posterior odds of effects being above chance at the level of timesteps, sensors, voxels, etc,  
34 while naturally taking into account spatiotemporal dependencies present in M/EEG data. We  
35 compare the performance of the proposed approach to well-established alternative methods using  
36 both simulated and actual M/EEG data and show that it significantly outperforms alternative  
37 methods in estimating the onset and offset of M/EEG effects.

<sup>38</sup> **1.2 Statistical errors and cluster-based inference**

<sup>39</sup> The issues with multiple comparisons represent a common and well-recognised danger in neuroimaging and M/EEG research, where the collected data allows for a multitude of potential  
<sup>40</sup> hypothesis tests and is characterised by complex structures of spatiotemporal dependencies.  
<sup>41</sup> The probability of obtaining at least one false positive in an ensemble (family) of  $m$  tests (i.e.,  
<sup>42</sup> the FWER) is computed as  $1 - (1 - \alpha)^m$  (for  $m = 10$  independent tests and  $\alpha = 0.05$ , it is  
<sup>43</sup> approximately equal to 0.4). Different methods exist to control the FWER, that is, to bring it  
<sup>44</sup> back to  $\alpha$ . Most methods apply a simple correction to series of  $p$ -values issued from univariate  
<sup>45</sup> statistical tests (e.g., t-tests). For instance, the Bonferroni correction (Dunn, 1961) consists  
<sup>46</sup> in setting the significance threshold to  $\alpha/m$ , or equivalently, multiplying the  $p$ -values by  $m$   
<sup>47</sup> and using the standard  $\alpha$  significance threshold. This method is generally overconservative  
<sup>48</sup> (i.e., under-powered) as it assumes statistical independence of the tests, an assumption that is  
<sup>49</sup> clearly violated in the context of M/EEG timeseries characterised by massive spatiotemporal  
<sup>50</sup> dependencies. Some alternative methods aims at controlling the FDR, defined as the proportion  
<sup>51</sup> of false positive *among positive tests* (e.g., Benjamini & Hochberg, 1995; Benjamini & Yekutieli,  
<sup>52</sup> 2001). However, a major limitation of both types of corrections is that they do not take into  
<sup>53</sup> account the spatial and temporal information contained in M/EEG data.

<sup>54</sup> A popular technique to account for spatiotemporal dependencies while controlling the FWER  
<sup>55</sup> is cluster-based inference (Bullmore et al., 1999; Maris & Oostenveld, 2007). A typical cluster-  
<sup>56</sup> based inference consists of two successive steps (for more details on cluster-based inference, see  
<sup>57</sup> for instance Frossard & Renaud, 2022; Maris, 2011; Maris & Oostenveld, 2007; Sassenhagen &  
<sup>58</sup> Draschkow, 2019). First, clusters are defined as sets of contiguous timesteps, sensors, voxels,  
<sup>59</sup> etc, whose activity, summarised by some test statistic (e.g., a  $t$ -value), exceeds a predefined  
<sup>60</sup> threshold (e.g., the 95th percentile of the parametric null distribution). Clusters are then  
<sup>61</sup> characterised by their height (i.e., maximal value), extent (number of constituent elements), or  
<sup>62</sup> some combination of both, for instance by summing the statistics within a cluster, an approach  
<sup>63</sup> referred to as “cluster mass” (Maris & Oostenveld, 2007; Pernet et al., 2015). Then, the null  
<sup>64</sup> hypothesis is tested by computing a  $p$ -value for each identified cluster by comparing its mass  
<sup>65</sup> with the null distribution of cluster masses (obtained via permutation). As alluded previously,  
<sup>66</sup> a significant cluster is a cluster which contains *at least one* significant time-point. As such, it  
<sup>67</sup> would be incorrect to conclude, for instance, that the timestep of a significant cluster is the first  
<sup>68</sup> moment at which some conditions differ (Frossard & Renaud, 2022; Sassenhagen & Draschkow,  
<sup>69</sup> 2019). Because the inference is performed at the second step (i.e., once clusters have been  
<sup>70</sup> formed), no conclusion can be made about individual datapoints (e.g., timesteps, sensors, etc).

<sup>71</sup> As different cluster-forming thresholds lead to clusters with different spatial or temporal ex-  
<sup>72</sup> tent, this initial threshold modulates the sensitivity of the subsequent permutation test. The  
<sup>73</sup> threshold-free cluster enhancement (TFCE) method was introduced by S. Smith & Nichols  
<sup>74</sup> (2009) to overcome this choice of an arbitrary threshold. In brief, the TFCE method works  
<sup>75</sup> as follows. Instead of picking an arbitrary cluster-forming threshold (e.g.,  $t = 2$ ), the methods  
<sup>76</sup> consist in trying all (or many) possible thresholds in a given range and checking whether a given  
<sup>77</sup> datapoint (e.g., timestep, sensor, voxel) belongs to a significant cluster under any of the set of  
<sup>78</sup>

79 thresholds. Then, instead of using cluster mass, one uses a weighted average between the cluster  
 80 extend ( $e$ , how broad is the cluster, that is, how many connected samples it contains) and the  
 81 cluster height ( $h$ , how high is the cluster, that is, how large is the test statistic). The TFCE  
 82 score at each timestep  $t$  is given by:

$$\text{TFCE}(t) = \int_{h=h_0}^{h=h_t} e(h)^E h^H dh$$

83 where  $h_0$  is typically 0 and parameters  $E$  and  $H$  are set a priori (typically to 0.5 and 2, re-  
 84 spectively) and control the influence of the extend and height on the TFCE. In practice, this  
 85 integral is approximated by a sum over small  $h$  increments. Then, a  $p$ -value for each timestep  
 86  $t$  is computed by comparing its TFCE with the null distribution of TFCE values (obtained  
 87 via permutation). For each permuted signal, we keep the maximal value over the whole sig-  
 88 nal for the null distribution of the TFCE. The TFCE combined with permutation (assuming  
 89 a large enough number of permutations) has been shown to provide accurate FWER control  
 90 (e.g., [Pernet et al., 2015](#)). However, further simulation work showed that cluster-based meth-  
 91 ods (including TFCE) perform poorly in localising the onset of M/EEG effects (e.g., [Rousselet,](#)  
 92 [2025; Sassenhagen & Draschkow, 2019](#)).

93 To sum up, the main limitation of cluster-based inference is that it allows for inference at the  
 94 cluster level only, not allowing inference at the level of timesteps, sensors, etc. As a conse-  
 95 quence, it does not allow inferring the precise spatial and temporal localisation of effects. In  
 96 the following, we briefly review previous modelling work of M/EEG data. Then, we provide a  
 97 short introduction to generalised additive models (GAMs) to illustrate how these models can  
 98 be used to precisely estimate the onset and offset of M/EEG effects.

### 99 1.3 Previous work on modelling M/EEG data

100 Scalp-recorded M/EEG signals capture neural activity originating from various brain regions  
 101 and are often contaminated by artifacts unrelated to the cognitive processes under investigation.  
 102 Consequently, analysing M/EEG data necessitates methods that can disentangle task-relevant  
 103 neural signals from extraneous “noise.” A widely adopted technique for this purpose is the esti-  
 104 mation of event-related potentials (ERPs), which are stereotyped electrophysiological responses  
 105 time-locked to specific sensory, cognitive, or motor events. Typically, ERPs are derived by  
 106 averaging EEG or MEG epochs across multiple trials aligned to the event of interest (e.g., stim-  
 107 ulus onset), thereby enhancing the signal-to-noise ratio by attenuating non-time-locked activity.  
 108 However, this averaging approach has notable limitations: it assumes consistent latency and  
 109 amplitude across trials and is primarily suited for simple categorical designs. Such assumptions  
 110 may not hold in more complex experimental paradigms, potentially leading to suboptimal ERP  
 111 estimations (e.g., [N. J. Smith & Kutas, 2014a](#)).

112 To overcome the limitations of simple averaging, several model-based approaches for estimating  
 113 ERPs have been proposed. These methods are motivated by the observation that traditional  
 114 ERP averaging is mathematically equivalent to fitting an intercept-only linear regression model

in a simple categorical design without overlapping events (N. J. Smith & Kutas, 2014a). In contrast to simple averaging, regression-based approaches to ERP estimation offer substantially greater flexibility. Notably, they allow for the modelling of both linear and nonlinear effects of continuous predictors, such as word frequency or age (e.g., N. J. Smith & Kutas, 2014a, 2014b; Tremblay & Newman, 2014), and enable the disentangling of overlapping cognitive processes (e.g., Ehinger & Dimigen, 2019; Skukies et al., 2024; Skukies & Ehinger, 2021). One widely used implementation of this approach is provided by the LIMO EEG toolbox (Pernet et al., 2011), which follows a multi-stage analysis pipeline. First, a separate regression model is fit for each datapoint (e.g., each time point and electrode) at the individual participant level to estimate ERP responses. This is followed by group-level statistical analyses of the resulting regression coefficients, often accompanied by corrections for multiple comparisons or cluster-based inference (for recent applied examples, see Dunagan et al., 2025; Wüllhorst et al., 2025).

Although this framework allows for the inclusion of a wide range of predictors—both continuous and categorical, linear and nonlinear—it still has important limitations. First, fitting separate models for each datapoint ignores the spatiotemporal dependencies inherent in M/EEG data, potentially reducing statistical power and interpretability. Second, the subsequent group-level analyses typically do not account for hierarchical dependencies which could otherwise be addressed through multilevel modelling. Finally, because the output of this procedure is summarised by cluster-based inference, its conclusions remain subject to the limitations discussed in the previous section.

Beyond modelling nonlinear effects of continuous predictors on ERP amplitudes, GAMs have been employed to capture the temporal dynamics of ERPs themselves, effectively modelling the shape of the waveform over time (Abugaber et al., 2023; Baayen et al., 2018; Meulman et al., 2015, 2023). This approach allows for the estimation of smooth, data-driven functions that characterise how neural responses evolve over time, offering a flexible alternative to traditional linear models. In the following section, we provide a brief introduction to GAMs, highlighting their applicability to M/EEG time series analysis and the advantages they offer over conventional methods.

#### 1.4 Generalised additive models

In generalised additive models, the functional relationship between the predictors and the response variable is decomposed into a sum of low-dimensional non-parametric functions. A typical GAM has the following form:

$$y_i \sim \text{EF}(\mu_i, \phi)$$

$$g(\mu_i) = \underbrace{\mathbf{A}_i \boldsymbol{\gamma}}_{\text{parametric part}} + \underbrace{\sum_{j=1}^J f_j(x_{ij})}_{\text{non-parametric part}}$$

where  $y_i \sim \text{EF}(\mu_i, \phi)$  denotes that the observations  $y_i$  are distributed as some member of the exponential family of distributions (e.g., Gaussian, Gamma, Beta, Poisson) with mean  $\mu_i$  and

<sup>149</sup> scale parameter  $\phi$ ;  $g(\cdot)$  is the link function,  $\mathbf{A}_i$  is the  $i$ th row of a known parametric model  
<sup>150</sup> matrix,  $\gamma$  is a vector of parameters for the parametric terms (to be estimated),  $f_j$  is a smooth  
<sup>151</sup> function of covariate  $x_j$  (to be estimated as well). The smooth functions  $f_j$  are represented in  
<sup>152</sup> the model as a weighted sum of  $K$  simpler, basis functions:

$$f_j(x_{ij}) = \sum_{k=1}^K \beta_{jk} b_{jk}(x_{ij})$$

<sup>153</sup> where  $\beta_{jk}$  is the weight (coefficient) associated with the  $k$ th basis function  $b_{jk}()$  evaluated at  
<sup>154</sup> the covariate value  $x_{ij}$  for the  $j$ th smooth function  $f_j$ . To clarify the terminology at this point:  
<sup>155</sup> *splines* are functions composed of simpler functions. These simpler functions are called *basis*  
<sup>156</sup> *functions* (e.g., cubic polynomial, thin-plate) and the set of basis functions is called a *basis*. Each  
<sup>157</sup> basis function is weighted by its coefficient and the resultant spline is the sum of these weighted  
<sup>158</sup> basis functions (Figure 1A). Splines coefficients are penalised (usually through the square of the  
<sup>159</sup> smooth functions' second derivative) in a way that can be interpreted, in Bayesian terms, as  
<sup>160</sup> a prior on the “wiggliness” of the function (Miller, 2025; Wood, 2017a). In other words, more  
<sup>161</sup> complex (wiggly) basis functions are automatically penalised.

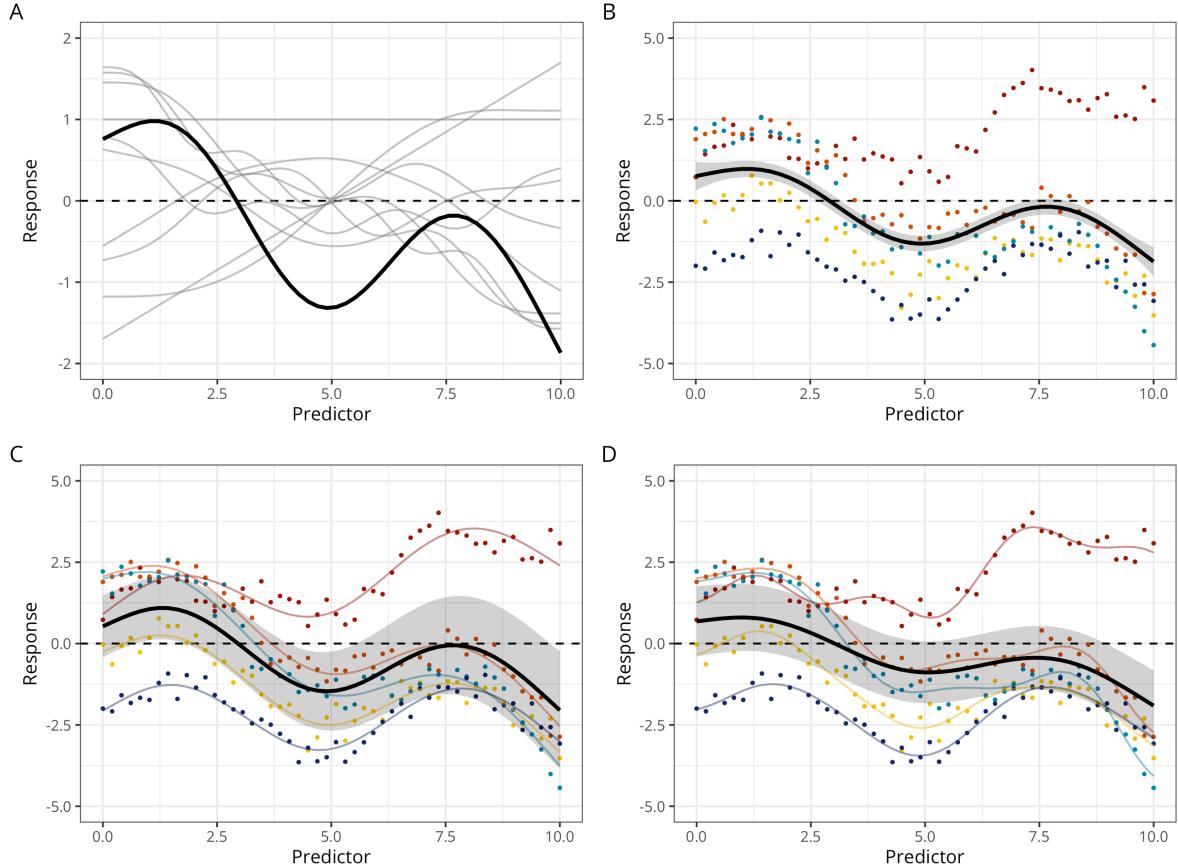
<sup>162</sup> A detailed treatment of the technical underpinnings of GAMs is beyond the scope of this article  
<sup>163</sup> (see reference books such as [Hastie & Tibshirani, 2017](#); [Wood, 2017a](#)). However, it is worth  
<sup>164</sup> emphasising that GAMs have been successfully applied to a wide range of time series data  
<sup>165</sup> across the cognitive sciences, including pupillometry (e.g., [Rij et al., 2019](#)), articulography (e.g.,  
<sup>166</sup> [Wieling, 2018](#)), speech formant dynamics (e.g., [Sóskuthy, 2021](#)), neuroimaging data (e.g., [Dinga](#)  
<sup>167</sup> [et al., 2021](#)), and event-related potentials (e.g., [Abugaber et al., 2023](#); [Baayen et al., 2018](#);  
<sup>168</sup> [Meulman et al., 2015, 2023](#)). Their appeal for modelling M/EEG data lies in their ability  
<sup>169</sup> to flexibly capture the complex shape of ERP waveforms without overfitting, through the use  
<sup>170</sup> of smooth functions constrained by penalisation. Recent extensions, such as distributional  
<sup>171</sup> GAMs ([Rigby & Stasinopoulos, 2005](#); [Umlauf et al., 2018](#)), allow researchers to model not  
<sup>172</sup> only the mean structure but also the variance (or scale) and other distributional properties as  
<sup>173</sup> functions of predictors, a feature that has proven useful in modelling neuroimaging data (e.g.,  
<sup>174</sup> [Dinga et al., 2021](#)). Moreover, hierarchical or multilevel GAMs ([E. J. Pedersen et al., 2019](#))  
<sup>175</sup> provide a principled way to account for the nested structure of M/EEG data (e.g., trials within  
<sup>176</sup> participants), enabling the inclusion of varying intercepts, slopes, and smoothers (as illustrated  
<sup>177</sup> in Figure 1C-D). This approach mitigates the risk of overfitting and reduces the influence of  
<sup>178</sup> outliers on smooth estimates ([Baayen & Linke, 2020](#); [Meulman et al., 2023](#)).

## <sup>179</sup> 1.5 Objectives

<sup>180</sup> Cluster-based permutation tests are widely used in M/EEG research to identify statistically  
<sup>181</sup> significant effects across time and space. However, these methods have notable limitations,  
<sup>182</sup> particularly in accurately determining the precise onset and offset of neural effects. To address  
<sup>183</sup> these limitations, we developed a model-based approach relying on Bayesian generalised ad-  
<sup>184</sup> ditive multilevel models implemented in R via the `brms` package ([Bürkner, 2017, 2018](#)). We

**Figure 1**

*Different types of GAMs.* **A:** GAMs predictions are computed as the weighted sum (in black) of basis functions (here thin-plate basis functions, in grey). **B:** Constant-effect GAM, with 5 participants in colours and the group-level prediction in black. **C:** Varying-intercept + varying-slope GAMM (with constant smoother). **D:** Varying-intercept + varying-slope + varying-smoother GAMM. In this model, each participant gets its own intercept, slope, and degree of ‘wiggliness’ (smoother).



evaluated the performance of this approach against conventional methods using both simulated and actual M/EEG data. Our findings demonstrate that this method provides more precise and reliable estimates of effects’ onset and offset than conventional approaches such as cluster-based inference.

189

## 2 Benchmarking with known ground truth

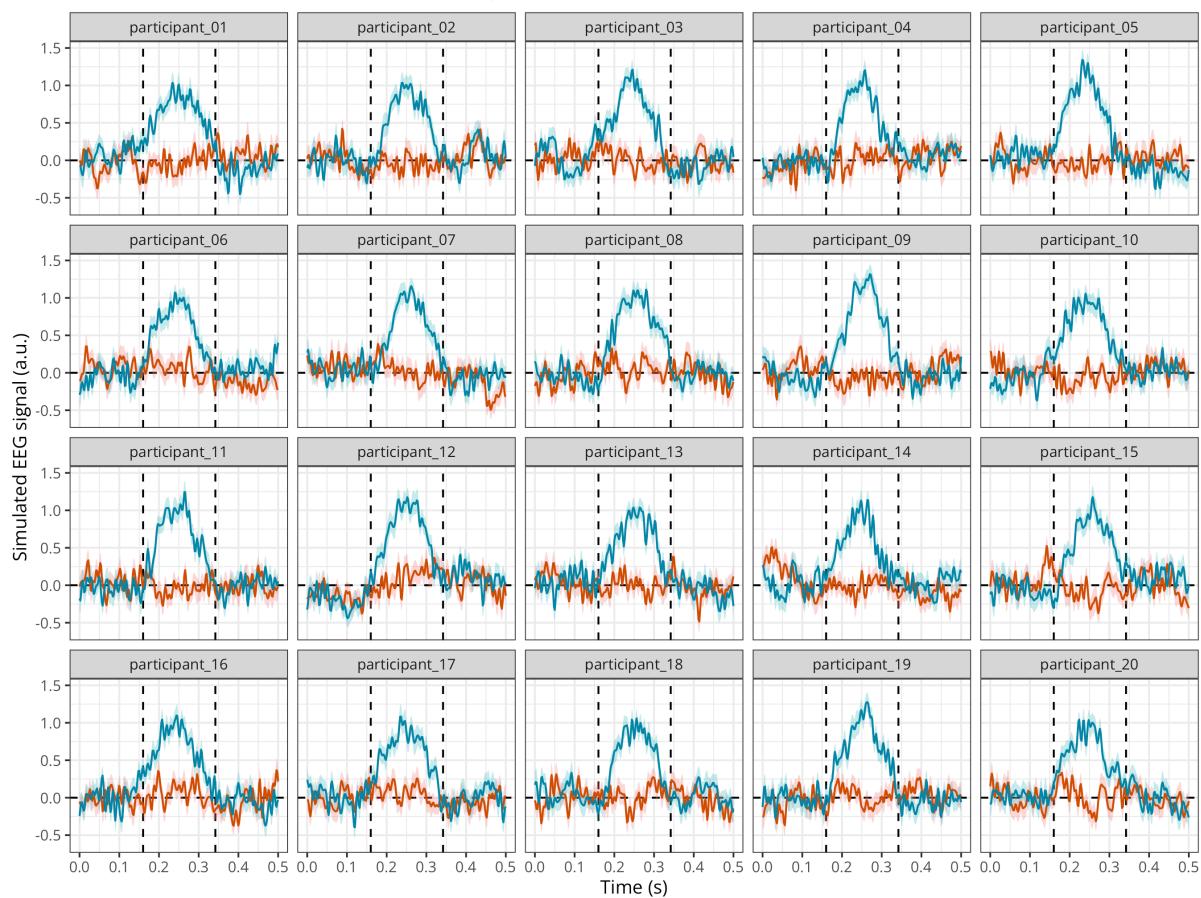
190 **2.1 Methods**191 **2.1.1 M/EEG data simulation**

To assess the accuracy of group-level onset and offset estimation of our proposed method, we simulated EEG with known onset and offset values. Following the approach of Sassenhagen & Draschkow (2019) and Rousselet (2025), we simulated EEG data stemming from two conditions, one with noise only, and the other with noise + signal. As in previous studies, the noise

196 was generated by superimposing 50 sinusoids at different frequencies, following an EEG-like  
 197 spectrum (see code in the online supplementary materials and details in [Yeung et al., 2004](#)). As  
 198 in Rousselet (2025), the signal was generated from a truncated Gaussian distribution with an  
 199 objective onset at 160 ms, a peak at 250 ms, and an offset at 342 ms. We simulated this signal  
 200 for 250 timesteps between 0 and 0.5s, akin to a 500 Hz sampling rate. We simulated data for a  
 201 group of 20 participants (with variable true onset) with 50 trials per participant and condition  
 202 (Figure 2). All figures and simulation results can be reproduced using the R code available  
 203 online at: [https://github.com/lmalborczyk/brms\\_meeg](https://github.com/lmalborczyk/brms_meeg).

**Figure 2**

*Mean simulated EEG activity in two conditions with 50 trials each, for a group of 20 participants. The error band represents the mean +/- 1 standard error of the mean.*



### 204 2.1.2 Model description and model fitting

205 We then fitted a Bayesian GAM (BGAM) using the `brms` package ([Bürkner, 2017, 2018](#)) and de-  
 206 fault priors (i.e., weakly informative priors). We ran eight Markov Chain Monte-Carlo (MCMC)  
 207 to approximate the posterior distribution, including each 5000 iterations and a warmup of 2000  
 208 iterations, yielding a total of  $8 \times (5000 - 2000) = 24000$  posterior samples to use for inference.  
 209 Posterior convergence was assessed examining trace plots as well as the Gelman–Rubin statis-  
 210 tic  $\hat{R}$  ([Gabry et al., 2019; Gelman et al., 2020](#)). The `brms` package uses the same syntax as  
 211 the R package `mgcv` v 1.9-3 ([Wood, 2017b](#)) for specifying smooth effects. Figure 3 shows the

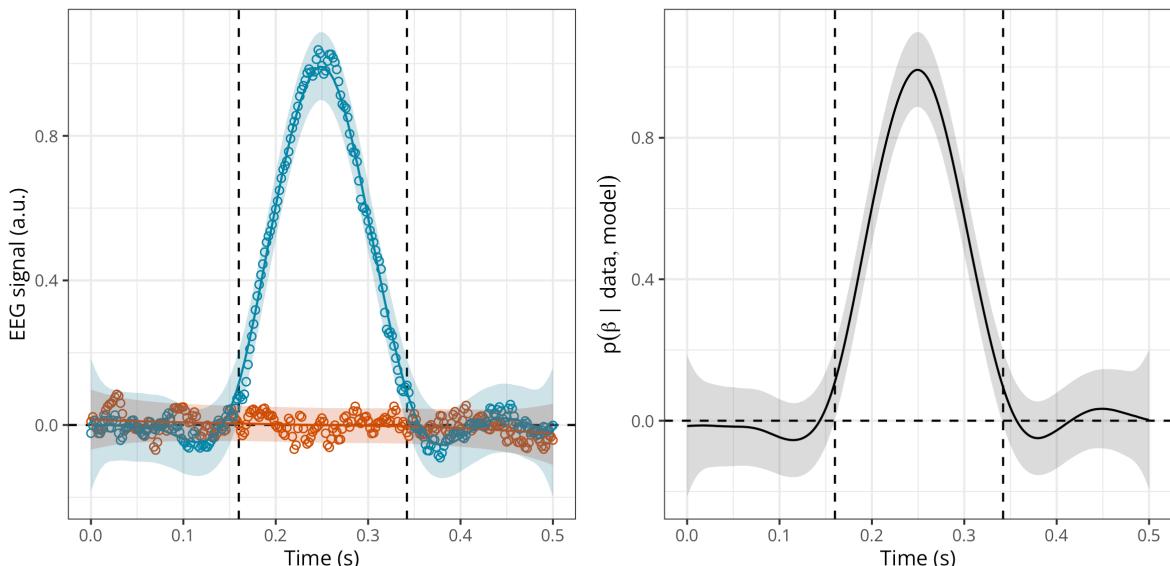
212 predictions of this model together with the raw data.

213 However, the previous model only included constant (fixed) effects, thus not properly accounting  
214 for between-participant variability. We next fitted a multilevel version of the BGAM (BGAMM,  
215 for an introduction to Bayesian multilevel models in `brms`, see [Nalborczyk et al., 2019](#)) including  
216 a varying intercept and slope for participant (but with a constant smoother). Although it is  
217 possible to fit a BGAMM using data at the single-trial level, we present a computationally  
218 lighter version of the model that is fitted directly on by-participant summary statistics (mean  
219 and SD), similar to what is done in meta-analysis.

220 We depict the posterior predictions together with the posterior estimate of the slope for  
221 `condition` at each timestep (Figure 3). This figure suggests that the BGAMM provides an ade-  
222 quate description of the simulated data (see further posterior predictive checks in [Appendix B](#)).

### Figure 3

*Posterior estimate of the EEG activity in each condition (left) and posterior estimate of the difference in EEG activity (right) according to the BGAMM.*



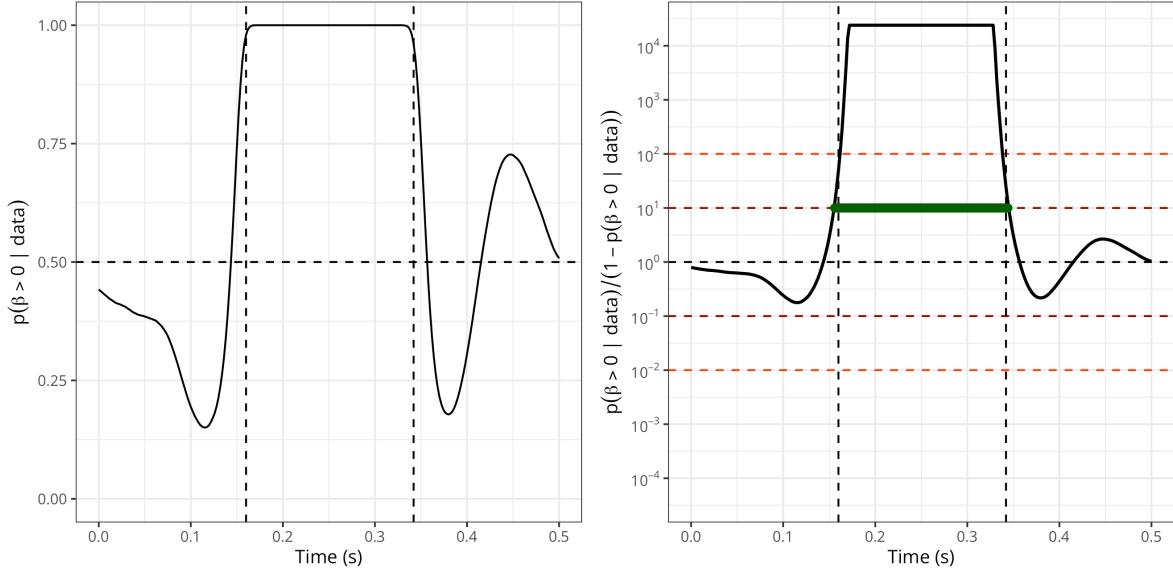
223 We then compute the posterior probability of the slope for `condition` being above 0 (Figure 4,  
224 left). From this quantity, we compute the ratio of posterior probabilities (i.e.,  $p/(1-p)$ ), or  
225 posterior odds, and visualise the timecourse of these odds superimposed with the conventional  
226 thresholds on evidence ratios (Figure 4, right). A ratio of 10 means that the probability of the  
227 difference being above 0 is 10 times higher than the probability of the difference not being above  
228 0, given the data, the priors, and other model's assumptions.<sup>1</sup> Thresholding the posterior odds  
229 thus provides a model-based approach for estimating the onset and offset of M/EEG effects,  
230 whose properties will be assessed in the simulation study. An important advantage is that the  
231 proposed approach can be extended to virtually any model structure.

---

<sup>1</sup>These posterior odds are equivalent to a Bayes factor, assuming 1:1 prior odds.

**Figure 4**

*Left: Posterior probability of the EEG difference (slope) being above 0 according to the BGAMM. Right: Posterior odds according to the BGAMM (on a log10 scale). Timesteps above threshold (10) are highlighted in green. NB: the minimum and maximum possible posterior odds are determined (bounded) by the number of posterior samples in the model.*



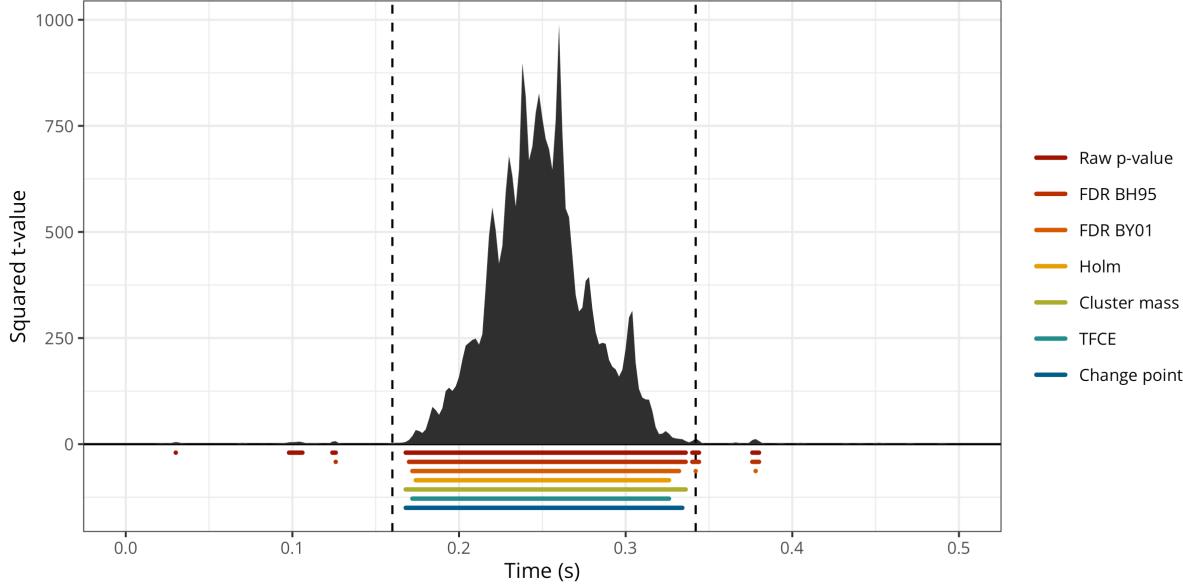
### 2.1.3 Comparing the onset/offset estimates across approaches

We then compared the ability of the BGAM to accurately estimate the onset and offset of the ERP difference to other widely-used methods. First, we conducted mass-univariate t-tests (thus treating each timestep independently) and identified the onset and offset of the ERP difference as the first and last values crossing an arbitrary significance threshold ( $\alpha = 0.05$ ). We then followed the same approach but after applying different forms of multiplicity correction to the  $p$ -values. We compared two methods that control the FDR (i.e., BH95, Benjamini & Hochberg, 1995; and BY01, Benjamini & Yekutieli, 2001), one method that controls the FWER (i.e., Holm–Bonferroni method, Holm, 1979), and two cluster-based permutation methods (permutation with a single cluster-forming threshold and threshold-free cluster enhancement, TFCE, S. Smith & Nichols, 2009). The BH95, BY01, and Holm corrections were applied to the  $p$ -values using the `p.adjust()` function in R. The cluster-based inference was implemented using a cluster-sum statistic of squared  $t$ -values, as implemented in MNE-Python (Gramfort, 2013), called via the R package `reticulate` v 1.42.0 (Ushey et al., 2024). We also compared these estimates to the onset and offset as estimated using the binary segmentation algorithm, as implemented in the R package `changepoint` v 2.3 (Killick et al., 2022), and applied directly to the squared  $t$ -values (as in Rousselet, 2025).<sup>2</sup> Figure 5 illustrates the onsets and offsets estimated by each method on a single simulated dataset and shows that all methods systematically overestimate the true onset and underestimate the true offset. In addition, the Raw  $p$ -value, FDR BH95, and FDR BY01 methods identify clusters well before the true onset and after the true offset.

<sup>2</sup>As in Rousselet (2025), we fixed the number of expected change points to two in the binary segmentation algorithm, thus producing always one cluster.

**Figure 5**

*Exemplary timecourse of squared t-values with true onset and offset (vertical black dashed lines) and onsets/offsets identified using the raw p-values, the corrected p-values (BH95, BY01, Holm), the cluster-based methods (Cluster mass, TFCE), or using the binary segmentation method (Change point).*



#### 252 2.1.4 Simulation study

253 To assess the accuracy of group-level onset and offset estimation, all methods were compared  
 254 by computing the bias (defined as the mean difference between the estimated and true value of  
 255 the onset/offset), mean absolute error (MAE), root mean square error (RMSE), and variance  
 256 of onset/offset estimates from 10,000 simulated datasets. Following Rousselet (2025), each  
 257 participant was assigned a random onset between 150 and 170ms. Whereas the present article  
 258 focuses on one-dimensional signals (e.g., one M/EEG channel), we provide a 2D application in  
 259 Appendix A.

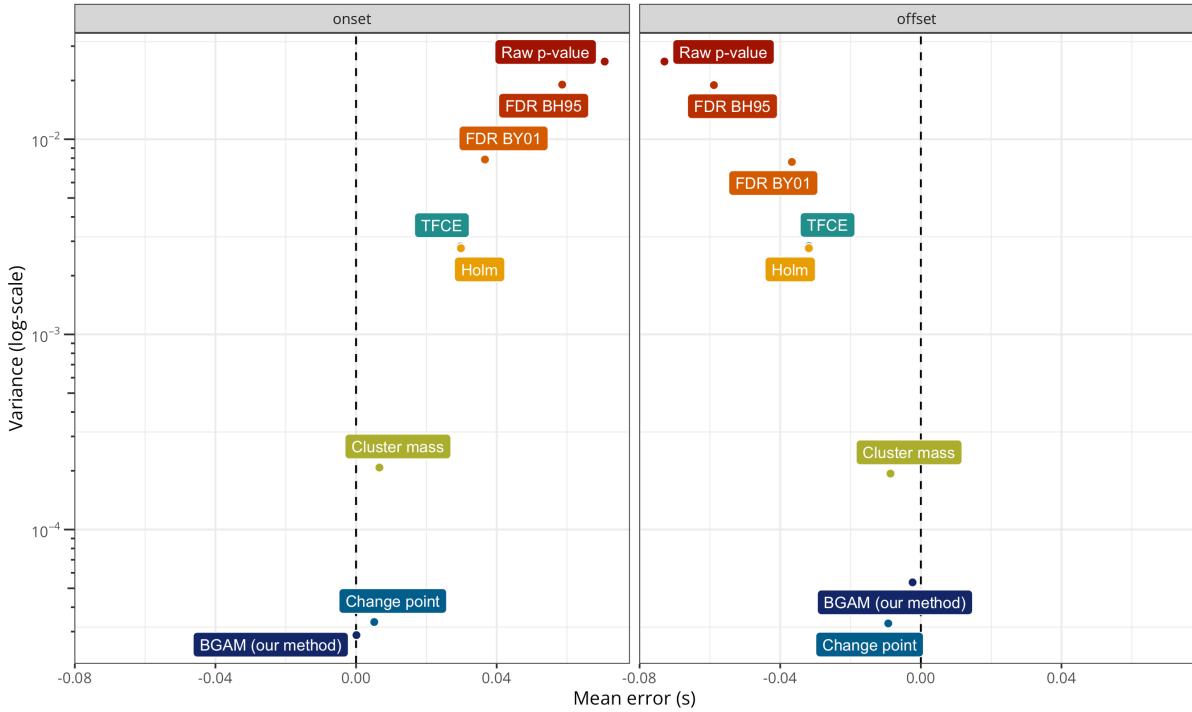
## 260 2.2 Results

261 Figure 6 shows a summary of the simulation results, revealing that the proposed approach (BGAM)  
 262 has the lowest error for both the onset and offset estimates. The Cluster mass and Change  
 263 point methods also have good performance, but perhaps surprisingly, the TFCE method performs  
 264 poorly for estimating the offset of the effect (with performance similar to the Holm method).  
 265 Unsurprisingly, the FDR BH95 and Raw p-value methods show the worst performance.

266 These results are further summarised in Table 1, which shows that the BGAM method is almost  
 267 perfectly unbiased (i.e., it has a bias of approximately 0.1ms for the onset and 2.4ms for the  
 268 offset). The Bias column shows that all methods tend to estimate the onset later than the true  
 269 onset and to estimate the offset earlier than the true offset. As can be seen from this table, the  
 270 BGAM method has the best performance on all included metrics (except for the Variance of the  
 271 offset estimate, where the Change point method performs better, presumably because it was

**Figure 6**

*Mean error and variance of onset and offset estimates according to each method. Variance is plotted on a log10 scale for visual purposes.*



272 constrained to identifying a single cluster).

273

### 3 Application to actual MEG data

#### 274 3.1 Methods

275 To complement the simulation study, we evaluated the performance of all methods on actual  
 276 MEG data ([Nalborczyk et al., in preparation](#)). In this study, the authors conducted time-  
 277 resolved multivariate pattern analysis (MVPA, also known as decoding) of MEG data recorded  
 278 in 32 human participants during a reading task. As a result, the authors obtained a timecourse of  
 279 decoding accuracy (ROC AUC), bounded between 0 and 1, for each participant. To test whether  
 280 the group-level average decoding accuracy was above chance (i.e., 0.5) at each timestep, we fitted  
 281 a BGAM as introduced previously with a basis dimension  $k = 50$  and retained all timesteps  
 282 exceeding a posterior odds of 20. To better distinguish signal from noise, we defined a region of  
 283 practical equivalence (ROPE, [Kruschke & Liddell, 2017](#)) as the upper 90% quantile of decoding  
 284 performance during the baseline period (i.e., before stimulus onset). Although we chose a basis  
 285 dimension of  $k = 50$ , which seemed appropriate for the present data, this choice should be  
 286 adapted according to the properties of the modelled data (e.g., signal-to-noise ratio, prior low-  
 287 pass filtering, sampling rate) and should be assessed by the usual model checking tools (e.g.,  
 288 models comparison, posterior predictive checks, see [Appendix B](#)).

**Table 1**

*Summary statistics of onset/offset estimates for each method (in ms, ordered by the MAE).*

	Bias	MAE	RMSE	Variance
onset				
BGAM (our method)	0.11	2.76	0.11	28.78
Change point	5.17	6.51	5.17	33.58
Cluster mass	6.64	7.62	6.64	207.86
Holm	29.83	32.01	29.83	2,763.68
TFCE	29.73	32.07	29.73	2,823.11
FDR BY01	36.67	50.80	36.67	7,872.30
FDR BH95	58.64	102.49	58.64	19,054.58
Raw p-value	70.72	132.86	70.72	25,004.65
offset				
BGAM (our method)	-2.35	3.46	2.35	53.63
Cluster mass	-8.64	9.44	8.64	193.63
Change point	-9.28	9.82	9.28	33.03
Holm	-31.86	34.03	31.86	2,764.12
TFCE	-31.84	34.19	31.84	2,834.11
FDR BY01	-36.68	51.37	36.68	7,648.53
FDR BH95	-58.87	102.63	58.87	18,939.58
Raw p-value	-72.93	133.33	72.93	25,012.72

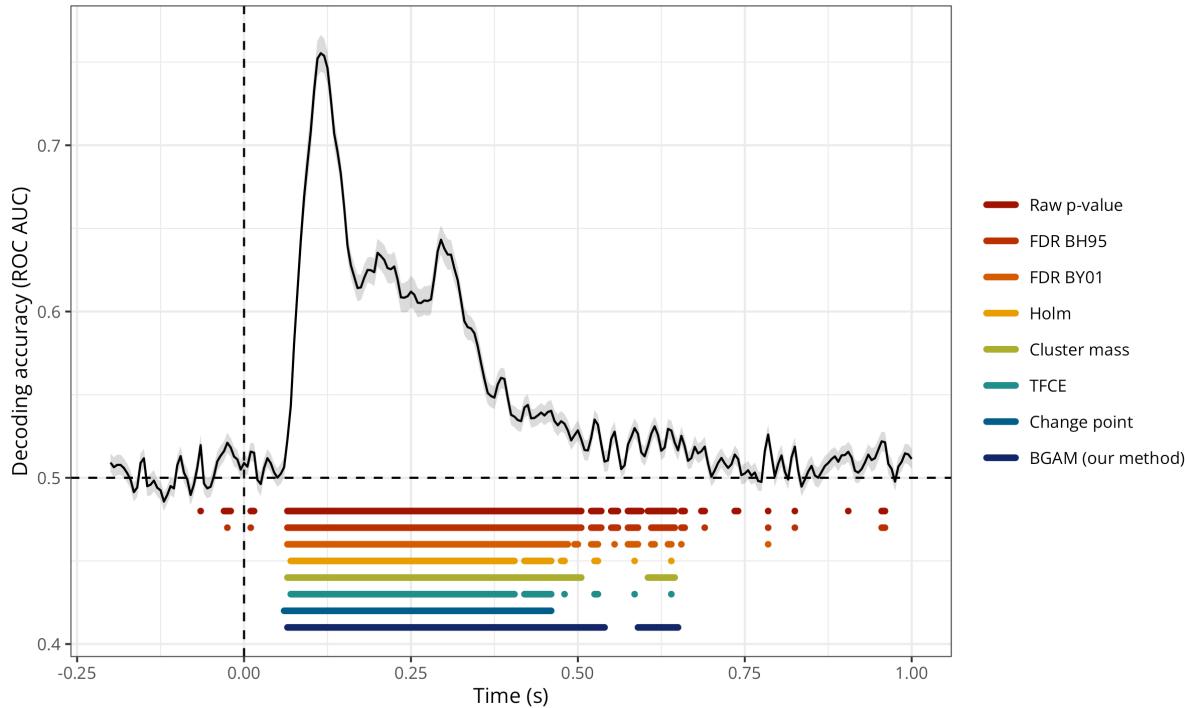
### 289 3.2 Results

290 Figure 7 shows the group-level average decoding performance through time superimposed with  
 291 onset and offset estimates from each method. Overall, this figure shows that both the **Raw**  
 292 **p-value** and **FDR BH95** methods are extremely lenient, identifying clusters of above-chance  
 293 decoding accuracy before the onset of the stimulus (false positive) and until the end of the trial.  
 294 The **Change point** method seems to be the most conservative one, identifying a single cluster  
 295 spanning from approximately +60ms to +450ms. The **Holm**, **Cluster mass**, **TFCE**, and **BGAM**  
 296 methods produce roughly similar estimates of onset and offset, ranging from approximately  
 297 +60ms to +650ms (considering only the first and last identified timesteps), although the **BGAM**  
 298 method seems to result in fewer clusters.

299 We then assessed the sensitivity of the various methods using a form of permutation-based  
 300 sensitivity study, which consisted of the following steps. First, we created a large number of  
 301 split halves of the data, that is, subsets of the dataset containing only 16 out of 32 participants.  
 302 For each possible pair of subsets, we have 16 possible levels of overlap/similarity that can  
 303 be quantified using the Jaccard index, ranging from 0 (perfectly disjoint subsets) to  $\approx 0.88$   
 304 (identical subsets except one participant). For each of these 16 levels of Jaccard similarity, we  
 305 created 1,000 pairs of subsets, resulting in 16,000 pairs of subsets in total. For each of these  
 306 pairs, we estimated the onset and offset according to each method and computed the absolute  
 307 difference in onset/offset estimates. Finally, we estimated the Spearman's rank correlation  
 308 coefficient (which quantifies the strength of a monotonic relation between two variables) between  
 309 the Jaccard similarity and the absolute difference in onset/offset estimates. The rational for  
 310 this procedure is that sensitive methods should produce similar onset/offset estimates for similar  
 311 subsets and dissimilar onset/offset estimates for dissimilar subsets. The results of this procedure

**Figure 7**

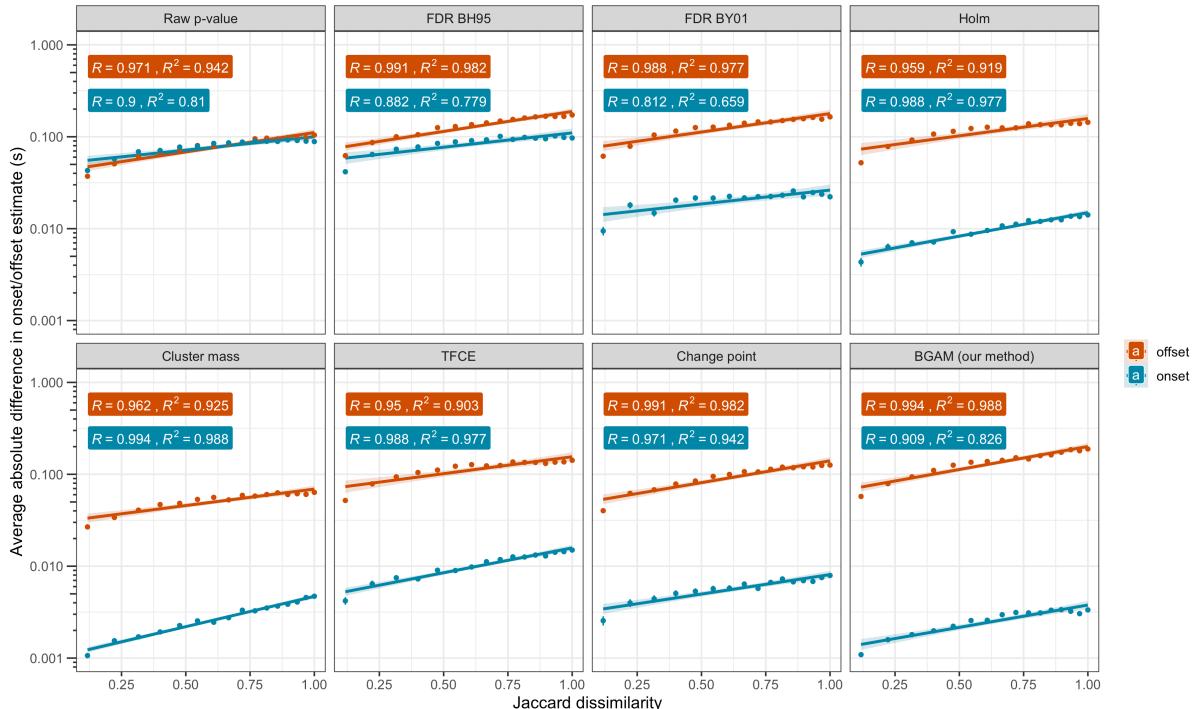
Group-level average decoding performance through time with clusters of higher-than-chance decoding performance as identified by each method (data from [Nalborczyk et al., in preparation](#)).



312 are summarised in Figure 8.

**Figure 8**

Relation between data subsets' dissimilarity (x-axis) and difference in onset (blue) and offset (orange) estimates (y-axis) according to each method.



313 This figure shows that, among the methods that performed best in the simulation study (i.e.,  
 314 **Cluster mass**, **Change point**, and **BGAM**), onset estimates remain highly stable across subsets of  
 315 participants with varying Jaccard similarity, it varies from around 1ms for most similar subsets  
 316 to around 10ms for most dissimilar subsets. Additionally, for both onset and offset estimates,  
 317 the average pairwise difference increases monotonically with Jaccard dissimilarity, as indicated  
 318 by the Spearman's rank correlation coefficient. For the onset estimates, the **Holm**, **Cluster**  
 319 **mass**, **TFCE**, and **BGAM** methods exhibit the strongest monotonic relation with subset similarity  
 320 (all  $\rho_s > 0.9$ ), whereas for offset estimates, all methods demonstrate excellent performance (all  
 321  $\rho_s > 0.9$ ) with the **BGAM** method showing the highest sensitivity ( $\rho \approx 0.994$ ). However, given  
 322 the aberrant clusters identified by the **Raw p-value**, **FDR BH95**, and **FDR BY01** methods (see  
 323 Figure 7), their sensitivity to variation in subset similarity is not meaningful.

324

## 4 Discussion

325 In brief, our results show that the model-based approach we introduced outperforms conven-  
 326 tional methods such as cluster-based inference in identifying the onset and offset of M/EEG  
 327 effects. Performance was assessed both on simulated data—allowing us to evaluate the method's  
 328 ability to recover ground-truth onset and offset values—and on actual MEG data—allowing us  
 329 to assess its sensitivity to realistic data properties (subset similarity). Together, these results  
 330 highlight desirable properties for any method aiming to precisely and reliably estimate the onset  
 331 and offset of M/EEG effects: it should i) recover true onsets and offsets in simulation (good  
 332 asymptotic behaviour), ii) identify clusters that are interpretable and consistent in empirical  
 333 data, and iii) show sensitivity to subtle changes in the data. Our approach meets all three of  
 334 these desiderata.

335 As with previous simulation studies (e.g., Rousselet et al., 2008; Sassenhagen & Draschkow,  
 336 2019), results inevitably depend on design choices, including the specific cluster-forming algo-  
 337 rithm and threshold (for cluster-based methods), the signal-to-noise ratio, and the potential  
 338 degradation of temporal resolution introduced by preprocessing steps such as low-pass filter-  
 339 ing. However, these constraints apply equally to all methods tested, so relative differences in  
 340 performance remain meaningful.

341 **I am not sure what to do with this paragraph (not sure how to discuss these**  
 342 **results/observations):** Interestingly, the TFCE method performed worse than the traditional  
 343 cluster-sum approach, consistent with the predictions of Rousselet (2025) based on the original  
 344 findings of S. Smith & Nichols (2009). We also found a striking overlap in the clusters identified  
 345 by the **Holm** and **TFCE** procedures (cf. Figure 7 and Figure 8). Whereas these two approaches  
 346 are conceptually distinct; **Holm** controlling the family-wise error rate through sequential *p*-value  
 347 adjustment, **TFCE** enhancing signal based on spatiotemporal support; their similarity in practice  
 348 may arise because both effectively prioritise extended, moderately strong effects over isolated  
 349 high-intensity points. This convergence warrants further methodological work. We chose not  
 350 to include the cluster-depth algorithm (Frossard & Renaud, 2022), as prior work already shows  
 351 its performance to be inferior to cluster-mass approaches in similar contexts (Rousselet, 2025).

352 A critical consideration for any model-based approach is that the model must adequately cap-  
353 ture the underlying data-generating process. Misspecified models are likely to produce biased  
354 or unreliable onset/offset estimates. This underscores the importance of thorough model diag-  
355 nostics, including posterior predictive checks, fit assessments, and model comparison (Gelman  
356 et al., 2020). As with any inferential modelling framework, transparency in assumptions and  
357 rigorous validation procedures are indispensable to ensure replicability and generalisability of  
358 the results.

359 Another important methodological consideration concerns the selection of model hyperparam-  
360 eters, such as the threshold for posterior odds and the number of basis functions. Although  
361 our simulations suggest that these parameters influence the precision and reliability of onset  
362 and offset estimates, optimal values may vary depending on the signal’s temporal dynamics  
363 and signal-to-noise characteristics. Future work could explore principled approaches to hyper-  
364 parameter tuning, including cross-validation or fully Bayesian model selection using tools such  
365 as leave-one-out cross-validation (LOO-CV) or Bayes factors (Gelman et al., 2020). We provide  
366 initial guidance in [Appendix B](#) and advocate for future development of adaptive heuristics to  
367 support flexible yet parsimonious model specification.

368 Currently, our approach estimates temporal effects independently at each sensor (1D temporal  
369 data). Extending the current framework to incorporate additional temporal (see [Appendix A](#))  
370 or spatial dimensions would improve both sensitivity and interpretability. Such extensions  
371 could draw on methods from spatial epidemiology and geostatistics using either GAMMs or  
372 approximate Gaussian process regression (e.g., Rasmussen & Williams, 2005; Riutort-Mayol et  
373 al., 2023), depending on computational feasibility.

374 To facilitate adoption, we developed the `neurogram` open-source R package (Nalborczyk, 2025),  
375 which implements the proposed method using `brms`. The package integrates seamlessly with  
376 MNE-Python (Gramfort, 2013), enabling researchers to process M/EEG data in Python and  
377 import them directly into R for model-based inference without cumbersome data export. This  
378 interoperability, described in [Appendix C](#), is designed to encourage broader use of model-based  
379 approaches in cognitive neuroscience.

380 In conclusion, we introduced a model-based approach for estimating the onset and offset of  
381 M/EEG effects. Across simulated and empirical datasets, we showed that the method yields  
382 more precise and sensitive estimates than conventional cluster-based approaches. These results  
383 highlight the potential of flexible, model-based alternatives for characterising time-resolved neu-  
384 ral dynamics, particularly in applications where accurate temporal localisation is critical.

**385 Data and code availability**

386 The simulation results as well as the R code to reproduce the simulations are available at:  
387 [https://github.com/lNALBORCZYK/brms\\_meeg](https://github.com/lNALBORCZYK/brms_meeg). The `neurogam` R package is available at <https://github.com/lNALBORCZYK/neurogam>.

**389 Packages**

390 We used R version 4.4.3 ([R Core Team, 2025](#)) and the following R packages: assertthat v. 0.2.1  
391 ([Wickham, 2019](#)), brms v. 2.22.0 ([Bürkner, 2017, 2018, 2021](#)), doParallel v. 1.0.17 ([Corporation  
& Weston, 2022](#)), easystats v. 0.7.4 ([Lüdecke et al., 2022](#)), foreach v. 1.5.2 ([Microsoft & Weston,  
2022](#)), furrr v. 0.3.1 ([Vaughan & Dancho, 2022](#)), future v. 1.58.0 ([Bengtsson, 2021](#)), ggpubr v.  
394 0.6.0 ([Kassambara, 2023](#)), ggrepel v. 0.9.6 ([Slowikowski, 2024](#)), glue v. 1.8.0 ([Hester & Bryan,  
2024](#)), grateful v. 0.2.12 ([Rodriguez-Sanchez & Jackson, 2024](#)), gt v. 1.0.0 ([Iannone et al.,  
2025](#)), knitr v. 1.50 ([Xie, 2014, 2015, 2025](#)), MetBrewer v. 0.2.0 ([Mills, 2022](#)), mgcv v. 1.9.3  
397 ([Wood, 2003b, 2004, 2011, 2017c; Wood et al., 2016](#)), neurogam v. 0.0.1 ([Nalborczyk, 2025](#)),  
398 pakret v. 0.2.2 ([Gallou, 2024](#)), patchwork v. 1.3.0 ([T. L. Pedersen, 2024](#)), rmarkdown v. 2.29  
399 ([Allaire et al., 2024; Xie et al., 2018, 2020](#)), scales v. 1.4.0 ([Wickham et al., 2025](#)), scico v.  
400 1.5.0 ([T. L. Pedersen & Crameri, 2023](#)), signal v. 1.8.1 ([signal developers, 2023](#)), tictoc v. 1.2.1  
401 ([Izrailev, 2024](#)), tidybayes v. 3.0.7 ([Kay, 2024](#)), tidytext v. 0.4.2 ([Silge & Robinson, 2016](#)),  
402 tidyverse v. 2.0.0 ([Wickham et al., 2019](#)).

**403 Acknolwedgements**

404 Centre de Calcul Intensif d’Aix-Marseille is acknowledged for granting access to its high perfor-  
405 mance computing resources.

406

## References

- 407 Abugaber, D., Finestrat, I., Luque, A., & Morgan-Short, K. (2023). Generalized additive mixed  
408 modeling of EEG supports dual-route accounts of morphosyntax in suggesting no word  
409 frequency effects on processing of regular grammatical forms. *Journal of Neurolinguistics*,  
410 67, 101137. <https://doi.org/10.1016/j.jneuroling.2023.101137>
- 411 Allaire, J., Xie, Y., Dervieux, C., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., Wickham,  
412 H., Cheng, J., Chang, W., & Iannone, R. (2024). *rmarkdown: Dynamic documents for r*.  
413 <https://github.com/rstudio/rmarkdown>
- 414 Baayen, R. H., & Linke, M. (2020). *Generalized Additive Mixed Models* (pp. 563–591). Springer  
415 International Publishing. [https://doi.org/10.1007/978-3-030-46216-1\\_23](https://doi.org/10.1007/978-3-030-46216-1_23)
- 416 Baayen, R. H., Rij, J. van, Cat, C. de, & Wood, S. (2018). *Autocorrelated errors in ex-*  
417 *perimental data in the language sciences: Some solutions offered by generalized additive*  
418 *mixed models* (pp. 49–69). Springer International Publishing. [https://doi.org/10.1007/978-3-319-69830-4\\_4](https://doi.org/10.1007/978-3-319-69830-4_4)
- 419 Bengtsson, H. (2021). A unifying framework for parallel and distributed processing in r using  
420 futures. *The R Journal*, 13(2), 208–227. <https://doi.org/10.32614/RJ-2021-048>
- 421 Benjamini, Y., & Hochberg, Y. (1995). Controlling the False Discovery Rate: A Practical and  
422 Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 57(1), 289–300. <https://doi.org/10.1111/j.2517-6161.1995.tb02031.x>
- 423 Benjamini, Y., & Yekutieli, D. (2001). The control of the false discovery rate in multiple  
424 testing under dependency. *The Annals of Statistics*, 29(4). <https://doi.org/10.1214/aos/1013699998>
- 425 Bullmore, E. T., Suckling, J., Overmeyer, S., Rabe-Hesketh, S., Taylor, E., & Brammer, M. J.  
426 (1999). Global, voxel, and cluster tests, by theory and permutation, for a difference between  
427 two groups of structural MR images of the brain. *IEEE Transactions on Medical Imaging*,  
428 18(1), 32–42. <https://doi.org/10.1109/42.750253>
- 429 Bürkner, P.-C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal  
430 of Statistical Software*, 80(1), 1–28. <https://doi.org/10.18637/jss.v080.i01>
- 431 Bürkner, P.-C. (2018). Advanced Bayesian multilevel modeling with the R package brms. *The  
432 R Journal*, 10(1), 395–411. <https://doi.org/10.32614/RJ-2018-017>
- 433 Bürkner, P.-C. (2021). Bayesian item response modeling in R with brms and Stan. *Journal of  
434 Statistical Software*, 100(5), 1–54. <https://doi.org/10.18637/jss.v100.i05>
- 435 Corporation, M., & Weston, S. (2022). doParallel: Foreach parallel adaptor for the “parallel”  
436 package. <https://CRAN.R-project.org/package=doParallel>
- 437 Delorme, A., & Makeig, S. (2004). EEGLAB: an open source toolbox for analysis of single-  
438 trial EEG dynamics including independent component analysis. *Journal of Neuroscience  
439 Methods*, 134(1), 9–21. <https://doi.org/10.1016/j.jneumeth.2003.10.009>
- 440 Dinga, R., Fraza, C. J., Bayer, J. M. M., Kia, S. M., Beckmann, C. F., & Marquand, A.  
441 F. (2021). Normative modeling of neuroimaging data using generalized additive models of  
442 location scale and shape. <http://dx.doi.org/10.1101/2021.06.14.448106>
- 443 Dunagan, D., Jordan, T., Hale, J. T., Pylkkänen, L., & Chacón, D. A. (2025). Evaluating the

- 448 timecourses of morpho-orthographic, lexical, and grammatical processing following rapid  
449 parallel visual presentation: An EEG investigation in English. *Cognition*, 257, 106080.  
450 <https://doi.org/10.1016/j.cognition.2025.106080>
- 451 Dunn, O. J. (1961). Multiple Comparisons among Means. *Journal of the American Statistical  
452 Association*, 56(293), 52–64. <https://doi.org/10.1080/01621459.1961.10482090>
- 453 Ehinger, B. V., & Dimigen, O. (2019). Unfold: An integrated toolbox for overlap correction,  
454 non-linear modeling, and regression-based EEG analysis. *PeerJ*, 7, e7838. <https://doi.org/10.7717/peerj.7838>
- 455 Frossard, J., & Renaud, O. (2022). The cluster depth tests: Toward point-wise strong con-  
456 trol of the family-wise error rate in massively univariate tests with application to M/EEG.  
457 *NeuroImage*, 247, 118824. <https://doi.org/10.1016/j.neuroimage.2021.118824>
- 458 Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., & Gelman, A. (2019). Visualization in  
459 Bayesian workflow. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*,  
460 182(2), 389–402. <https://doi.org/10.1111/rss.a.12378>
- 461 Gallou, A. (2024). *pakret: Cite “R” packages on the fly in “R Markdown” and “Quarto”*. <https://CRAN.R-project.org/package=pakret>
- 462 Gelman, A., Vehtari, A., Simpson, D., Margossian, C. C., Carpenter, B., Yao, Y., Kennedy,  
463 L., Gabry, J., Bürkner, P.-C., & Modrák, M. (2020). Bayesian workflow. *arXiv:2011.01808  
/Stat*. <http://arxiv.org/abs/2011.01808>
- 464 Gramfort, A. (2013). MEG and EEG data analysis with MNE-python. *Frontiers in Neuro-  
465 science*, 7. <https://doi.org/10.3389/fnins.2013.00267>
- 466 Hastie, T. J., & Tibshirani, R. J. (2017). *Generalized Additive Models*. Routledge. <https://doi.org/10.1201/9780203753781>
- 467 Hester, J., & Bryan, J. (2024). *glue: Interpreted string literals*. <https://CRAN.R-project.org/package=glue>
- 468 Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal  
469 of Statistics*, 6(2), 65–70. <http://www.jstor.org/stable/4615733>
- 470 Iannone, R., Cheng, J., Schloerke, B., Hughes, E., Lauer, A., Seo, J., Brevoort, K., & Roy, O.  
471 (2025). *gt: Easily create presentation-ready display tables*. <https://CRAN.R-project.org/package=gt>
- 472 Izrailev, S. (2024). *tictoc: Functions for timing r scripts, as well as implementations of “Stack”  
473 and “StackList” structures*. <https://CRAN.R-project.org/package=tictoc>
- 474 Kassambara, A. (2023). *ggpubr: “ggplot2” based publication ready plots*. <https://CRAN.R-project.org/package=ggpubr>
- 475 Kay, M. (2024). *tidybayes: Tidy data and geoms for Bayesian models*. <https://doi.org/10.5281/zenodo.1308151>
- 476 Killick, R., Haynes, K., & Eckley, I. A. (2022). *changepoint: An R package for changepoint  
477 analysis*. <https://CRAN.R-project.org/package=changepoint>
- 478 King, J.-R., & Dehaene, S. (2014). Characterizing the dynamics of mental representations:  
479 the temporal generalization method. *Trends in Cognitive Sciences*, 18(4), 203–210. <https://doi.org/10.1016/j.tics.2014.01.002>
- 480 Kruschke, J. K., & Liddell, T. M. (2017). The Bayesian New Statistics: Hypothesis testing,  
481

- estimation, meta-analysis, and power analysis from a Bayesian perspective. *Psychonomic Bulletin & Review*, 25(1), 178–206. <https://doi.org/10.3758/s13423-016-1221-4>
- Lüdecke, D., Ben-Shachar, M. S., Patil, I., Wiernik, B. M., Bacher, E., Thériault, R., & Makowski, D. (2022). easystats: Framework for easy statistical modeling, visualization, and reporting. CRAN. <https://doi.org/10.32614/CRAN.package.easystats>
- Maris, E. (2011). Statistical testing in electrophysiological studies. *Psychophysiology*, 49(4), 549–565. <https://doi.org/10.1111/j.1469-8986.2011.01320.x>
- Maris, E., & Oostenveld, R. (2007). Nonparametric statistical testing of EEG- and MEG-data. *Journal of Neuroscience Methods*, 164(1), 177–190. <https://doi.org/10.1016/j.jneumeth.2007.03.024>
- Meulman, N., Sprenger, S. A., Schmid, M. S., & Wieling, M. (2023). GAM-based individual difference measures for L2 ERP studies. *Research Methods in Applied Linguistics*, 2(3), 100079. <https://doi.org/10.1016/j.rmal.2023.100079>
- Meulman, N., Wieling, M., Sprenger, S. A., Stowe, L. A., & Schmid, M. S. (2015). Age Effects in L2 Grammar Processing as Revealed by ERPs and How (Not) to Study Them. *PLOS ONE*, 10(12), e0143328. <https://doi.org/10.1371/journal.pone.0143328>
- Microsoft, & Weston, S. (2022). *foreach*: Provides foreach looping construct. <https://CRAN.R-project.org/package=foreach>
- Miller, D. L. (2025). Bayesian views of generalized additive modelling. *Methods in Ecology and Evolution*. <https://doi.org/10.1111/2041-210x.14498>
- Mills, B. R. (2022). *MetBrewer*: Color palettes inspired by works at the metropolitan museum of art. <https://CRAN.R-project.org/package=MetBrewer>
- Nalborczyk, L. (2025). *neurogam*: Precise temporal localisation of m/EEG effects with bayesian generalised additive multilevel models. <https://github.com/lnalborczyk/neurogam>
- Nalborczyk, L., Batailler, C., Lœvenbruck, H., Vilain, A., & Bürkner, P.-C. (2019). An Introduction to Bayesian Multilevel Models Using brms: A Case Study of Gender Effects on Vowel Variability in Standard Indonesian. *Journal of Speech, Language, and Hearing Research*, 62(5), 1225–1242. [https://doi.org/10.1044/2018\\_jslhr-s-18-0006](https://doi.org/10.1044/2018_jslhr-s-18-0006)
- Nalborczyk, L., Hauw, F., Torcy, H. de, Dehaene, S., & Cohen, L. (in preparation). *Neural and representational dynamics of tickertape synesthesia*.
- Pedersen, E. J., Miller, D. L., Simpson, G. L., & Ross, N. (2019). Hierarchical generalized additive models in ecology: An introduction with mgcv. *PeerJ*, 7, e6876. <https://doi.org/10.7717/peerj.6876>
- Pedersen, T. L. (2024). *patchwork*: The composer of plots. <https://CRAN.R-project.org/package=patchwork>
- Pedersen, T. L., & Crameri, F. (2023). *scico*: Colour palettes based on the scientific colour-maps. <https://CRAN.R-project.org/package=scico>
- Pernet, C. R., Chauveau, N., Gaspar, C., & Rousselet, G. A. (2011). LIMO EEG: A Toolbox for Hierarchical LInear MOdeling of ElectroEncephaloGraphic Data. *Computational Intelligence and Neuroscience*, 2011, 1–11. <https://doi.org/10.1155/2011/831409>
- Pernet, C. R., Latinus, M., Nichols, T. E., & Rousselet, G. A. (2015). Cluster-based computational methods for mass univariate analyses of event-related brain potentials/fields: A

- 532 simulation study. *Journal of Neuroscience Methods*, 250, 85–93. <https://doi.org/10.1016/j.jneumeth.2014.08.003>
- 533
- 534 R Core Team. (2025). *R: A language and environment for statistical computing*. R Foundation  
535 for Statistical Computing. <https://www.R-project.org/>
- 536 Rasmussen, C. E., & Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning*.  
537 <https://doi.org/10.7551/mitpress/3206.001.0001>
- 538 Rigby, R. A., & Stasinopoulos, D. M. (2005). Generalized Additive Models for Location, Scale  
539 and Shape. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 54(3),  
540 507–554. <https://doi.org/10.1111/j.1467-9876.2005.00510.x>
- 541 Rij, J. van, Hendriks, P., Rijn, H. van, Baayen, R. H., & Wood, S. N. (2019). Analyzing  
542 the Time Course of Pupillometric Data. *Trends in Hearing*, 23. <https://doi.org/10.1177/2331216519832483>
- 543
- 544 Riutort-Mayol, G., Bürkner, P.-C., Andersen, M. R., Solin, A., & Vehtari, A. (2023). Practical  
545 Hilbert space approximate Bayesian Gaussian processes for probabilistic programming.  
546 *Statistics and Computing*, 33(1), 17. <https://doi.org/10.1007/s11222-022-10167-2>
- 547 Rodriguez-Sanchez, F., & Jackson, C. P. (2024). *grateful: Facilitate citation of R packages*.  
548 <https://pakillo.github.io/grateful/>
- 549 Rosenblatt, J. D., Finos, L., Weeda, W. D., Solari, A., & Goeman, J. J. (2018). All-  
550 Resolutions Inference for brain imaging. *NeuroImage*, 181, 786–796. <https://doi.org/10.1016/j.neuroimage.2018.07.060>
- 551
- 552 Rousselet, G. A. (2025). Using cluster-based permutation tests to estimate MEG/EEG onsets:  
553 How bad is it? *European Journal of Neuroscience*, 61(1), e16618. <https://doi.org/10.1111/ejn.16618>
- 554
- 555 Rousselet, G. A., Pernet, C. R., Bennett, P. J., & Sekuler, A. B. (2008). Parametric study  
556 of EEG sensitivity to phase noise during face processing. *BMC Neuroscience*, 9(1). <https://doi.org/10.1186/1471-2202-9-98>
- 557
- 558 Sassenhagen, J., & Draschkow, D. (2019). Cluster-based permutation tests of MEG/EEG data  
559 do not establish significance of effect latency or location. *Psychophysiology*, 56(6). <https://doi.org/10.1111/psyp.13335>
- 560
- 561 signal developers. (2023). *signal: Signal processing*. <https://r-forge.r-project.org/projects/signal/>
- 562
- 563 Silge, J., & Robinson, D. (2016). tidytext: Text mining and analysis using tidy data principles  
564 in r. *JOSS*, 1(3). <https://doi.org/10.21105/joss.00037>
- 565
- 566 Skukies, R., & Ehinger, B. (2021). Modelling event duration and overlap during EEG analysis.  
567 *Journal of Vision*, 21(9), 2037. <https://doi.org/10.1167/jov.21.9.2037>
- 568
- 569 Skukies, R., Schepers, J., & Ehinger, B. (2024, December 9). *Brain responses vary in duration  
570 - modeling strategies and challenges*. <https://doi.org/10.1101/2024.12.05.626938>
- 571 Slowikowski, K. (2024). *ggrepel: Automatically position non-overlapping text labels with “gg-  
572 plot2”*. <https://CRAN.R-project.org/package=ggrepel>
- 573 Smith, N. J., & Kutas, M. (2014a). Regression-based estimation of ERP waveforms: I. The  
574 rERP framework. *Psychophysiology*, 52(2), 157–168. <https://doi.org/10.1111/psyp.12317>
- 575 Smith, N. J., & Kutas, M. (2014b). Regression-based estimation of ERP waveforms: II. Non-

- 574 linear effects, overlap correction, and practical considerations. *Psychophysiology*, 52(2),  
575 169–181. <https://doi.org/10.1111/psyp.12320>
- 576 Smith, S., & Nichols, T. (2009). Threshold-free cluster enhancement: Addressing problems of  
577 smoothing, threshold dependence and localisation in cluster inference. *NeuroImage*, 44(1),  
578 83–98. <https://doi.org/10.1016/j.neuroimage.2008.03.061>
- 579 Sóskuthy, M. (2021). Evaluating generalised additive mixed modelling strategies for dynamic  
580 speech analysis. *Journal of Phonetics*, 84, 101017. <https://doi.org/10.1016/j.wocn.2020.101017>
- 582 Tremblay, A., & Newman, A. J. (2014). Modeling nonlinear relationships in ERP data using  
583 mixed-effects regression with R examples. *Psychophysiology*, 52(1), 124–139. <https://doi.org/10.1111/psyp.12299>
- 585 Umlauf, N., Klein, N., & Zeileis, A. (2018). BAMLSS: Bayesian Additive Models for Location,  
586 Scale, and Shape (and Beyond). *Journal of Computational and Graphical Statistics*, 27(3),  
587 612–627. <https://doi.org/10.1080/10618600.2017.1407325>
- 588 Ushey, K., Allaire, J., & Tang, Y. (2024). *Reticulate: Interface to 'python'*. <https://CRAN.R-project.org/package=reticulate>
- 590 Vaughan, D., & Dancho, M. (2022). *furrr: Apply mapping functions in parallel using futures*.  
591 <https://CRAN.R-project.org/package=furrr>
- 592 Wickham, H. (2019). *assertthat: Easy pre and post assertions*. <https://CRAN.R-project.org/package=assertthat>
- 594 Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund,  
595 G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M.,  
596 Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., ... Yutani, H. (2019). Welcome  
597 to the tidyverse. *Journal of Open Source Software*, 4(43), 1686. <https://doi.org/10.21105/joss.01686>
- 599 Wickham, H., Pedersen, T. L., & Seidel, D. (2025). *scales: Scale functions for visualization*.  
600 <https://CRAN.R-project.org/package=scales>
- 601 Wieling, M. (2018). Analyzing dynamic phonetic data using generalized additive mixed model-  
602 ing: A tutorial focusing on articulatory differences between L1 and L2 speakers of English.  
603 *Journal of Phonetics*, 70, 86–116. <https://doi.org/10.1016/j.wocn.2018.03.002>
- 604 Wood, S. N. (2003a). Thin Plate Regression Splines. *Journal of the Royal Statistical Society  
605 Series B: Statistical Methodology*, 65(1), 95–114. <https://doi.org/10.1111/1467-9868.00374>
- 606 Wood, S. N. (2003b). Thin-plate regression splines. *Journal of the Royal Statistical Society (B)*,  
607 65(1), 95–114. <https://doi.org/10.1111/1467-9868.00374>
- 608 Wood, S. N. (2004). Stable and efficient multiple smoothing parameter estimation for gener-  
609 alized additive models. *Journal of the American Statistical Association*, 99(467), 673–686.  
610 <https://doi.org/10.1198/016214504000000980>
- 611 Wood, S. N. (2011). Fast stable restricted maximum likelihood and marginal likelihood esti-  
612 mation of semiparametric generalized linear models. *Journal of the Royal Statistical Society  
613 (B)*, 73(1), 3–36. <https://doi.org/10.1111/j.1467-9868.2010.00749.x>
- 614 Wood, S. N. (2017a). *Generalized Additive Models*. Chapman; Hall/CRC. <https://doi.org/10.1201/9781315370279>

- 616 Wood, S. N. (2017b). *Generalized additive models: An introduction with r* (2nd ed.). Chapman;  
617 Hall/CRC.
- 618 Wood, S. N. (2017c). *Generalized Additive Models: An introduction with R* (2nd ed.). Chapman;  
619 Hall/CRC.
- 620 Wood, S. N., Pya, N., & Säfken, B. (2016). Smoothing parameter and model selection for  
621 general smooth models (with discussion). *Journal of the American Statistical Association*,  
622 111, 1548–1575. <https://doi.org/10.1080/01621459.2016.1180986>
- 623 Wüllhorst, V., Wüllhorst, R., Overmeyer, R., & Endrass, T. (2025). Comprehensive Analysis  
624 of Event-Related Potentials of Response Inhibition: The Role of Negative Urgency and  
625 Compulsivity. *Psychophysiology*, 62(2). <https://doi.org/10.1111/psyp.70000>
- 626 Xie, Y. (2014). knitr: A comprehensive tool for reproducible research in R. In V. Stodden, F.  
627 Leisch, & R. D. Peng (Eds.), *Implementing reproducible computational research*. Chapman;  
628 Hall/CRC.
- 629 Xie, Y. (2015). *Dynamic documents with R and knitr* (2nd ed.). Chapman; Hall/CRC. <https://yihui.org/knitr/>
- 630 Xie, Y. (2025). knitr: A general-purpose package for dynamic report generation in R. <https://yihui.org/knitr/>
- 631 Xie, Y., Allaire, J. J., & Grolemund, G. (2018). *R markdown: The definitive guide*. Chapman;  
632 Hall/CRC. <https://bookdown.org/yihui/rmarkdown>
- 633 Xie, Y., Dervieux, C., & Riederer, E. (2020). *R markdown cookbook*. Chapman; Hall/CRC.  
634 <https://bookdown.org/yihui/rmarkdown-cookbook>
- 635 Yeung, N., Bogacz, R., Holroyd, C. B., & Cohen, J. D. (2004). Detection of synchronized os-  
636 cillations in the electroencephalogram: An evaluation of methods. *Psychophysiology*, 41(6),  
637 822–832. <https://doi.org/10.1111/j.1469-8986.2004.00239.x>

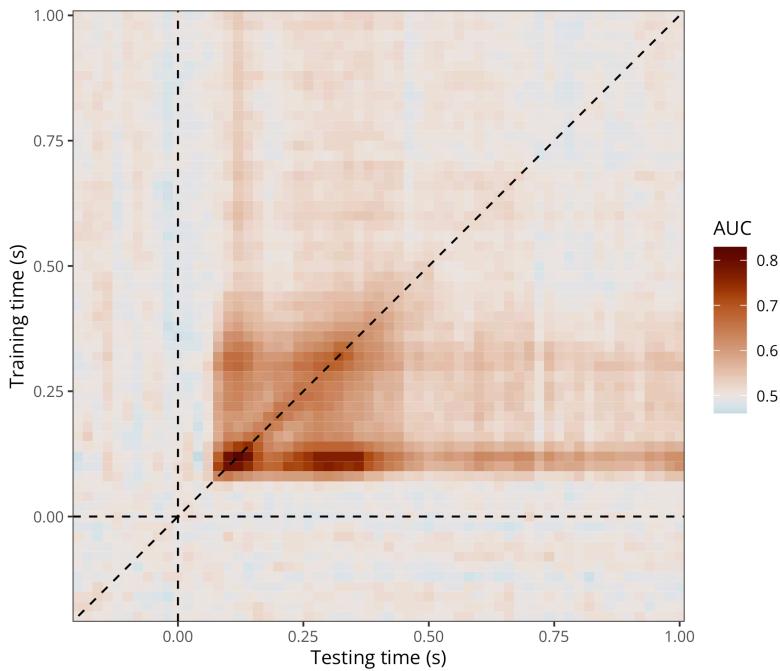
## Appendix A

### Application to 2D time-resolved decoding results (cross-temporal generalisation)

We conducted a cross-temporal generalisation analysis of the decoding data from Nalborczyk et al. ([in preparation](#)), in which we assessed the performance of classifiers trained and tested at various timesteps of the trial ([King & Dehaene, 2014](#)). This analysis was performed at the participant level, resulting in a 2D matrix where each element contains the decoding accuracy (ROC AUC) of a classifier trained at timestep training<sub>i</sub> and tested at timestep testing<sub>j</sub> for each participant (Figure A1).

**Figure A1**

*Group-level average cross-temporal generalisation matrix of decoding performance (data from Nalborczyk et al., [in preparation](#)).*



To model cross-temporal generalisation matrices of decoding performance (ROC AUC), we extended our initial BGAM to take into account the bivariate temporal distribution of AUC values, thus producing naturally smoothed estimates (timecourses) of AUC values and posterior probabilities. This model can be written as follows:

$$\begin{aligned} \text{AUC}_i &\sim \text{Beta}(\mu_i, \phi) \\ g(\mu_i) &= f(\text{train}_i, \text{test}_i) \end{aligned}$$

where we assume that AUC values come from a Beta distribution with two parameters  $\mu$  and  $\phi$ . We can think of  $f(\text{train}_i, \text{test}_i)$  as a surface (a smooth function of two variables) that we can model using a 2-dimensional splines. Let  $\mathbf{s}_i = (\text{train}_i, \text{test}_i)$  be some pair of training and testing samples, and let  $\mathbf{k}_m = (\text{train}_m, \text{test}_m)$  denote the  $m^{\text{th}}$  knot in the domain of  $\text{train}_i$  and  $\text{test}_i$ . We can then express the smooth function as:

$$f(\text{train}_i, \text{test}_i) = \alpha + \sum_{m=1}^M \beta_m b_m (\tilde{s}_i, \tilde{k}_m)$$

655 A popular bivariate basis function uses *thin-plate splines* (Wood, 2003a). These splines are  
 656 designed to interpolate and approximate smooth surfaces over two dimensions (hence the “bi-  
 657 variate” term). For  $d = 2$  dimensions and  $l = 2$  (smoothness penalty involving second order  
 658 derivative):

$$f(\tilde{s}_i) = \alpha + \beta_1 x_i + \beta_2 z_i + \sum_{m=1}^M \beta_{2+m} b_m (\tilde{s}_i, \tilde{k}_m)$$

659 using the radial basis function given by:

$$b_m (\tilde{s}_i, \tilde{k}_m) = \|\tilde{s}_i - \tilde{k}_m\|^2 \log \|\tilde{s}_i - \tilde{k}_m\|$$

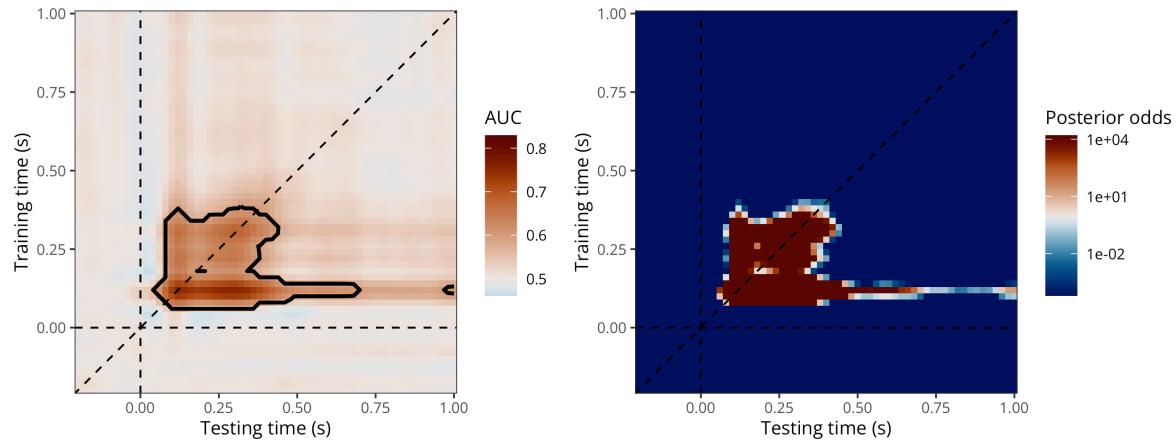
660 where  $\|\mathbf{s}_i - \mathbf{k}_m\|$  is the Euclidean distance between the covariate  $\mathbf{s}_i$  and the knot location  $\mathbf{k}_m$ .  
 661 We fitted this model using `brms` and the `t2()` tensor product smooth constructor with full  
 662 penalties (E. J. Pedersen et al., 2019). We ran eight MCMCs to approximate the posterior  
 663 distribution, including each 5000 iterations and a warmup of 2000 iterations, yielding a total of  
 664  $8 \times (5000 - 2000) = 24000$  posterior samples to be used for inference.

```
# fitting a GAM with two temporal dimensions
timegen_gam <- brm(
  # 2D thin-plate spline (tp) with full penalties
  auc ~ t2(train_time, test_time, bs = "tp", k = 30, full = TRUE),
  data = timegen_data,
  family = brms::Beta(),
  warmup = 1000,
  iter = 5000,
  chains = 8,
  cores = 8,
  control = list(adapt_delta = 0.95, max_treedepth = 15),
  backend = "cmdstanr",
  stan_model_args = list(stanc_options = list("01")),
  file = "models/timegen_gam_meg_t2_k30_full.rds"
)
```

665 Figure A2 shows the predictions from the model (left) superimposed with the identified cluster  
 666 as defined by thresholding the posterior odds (right). Notably, this model could be extended to  
 667 a multilevel bivariate GAM via `t2(train_time, test_time, participant, bs = c("tp",`  
 668 `"tp", "re"), m = 2, full = TRUE)` and could be generalised to account for both spatial (`x1`  
 669 and `x2`) and temporal (`time`) dimensions with formulas such as `te(x, y, time, d = c(2, 1))`

**Figure A2**

*Predicted AUC values with threshold (left) and posterior odds of decoding accuracy being above chance (right) according to the bivariate BGAM.*



670 ). The performance of such models should be assessed in future work.

**Table B1**

*Models comparison with LOOIC. Models are arranged by the difference in expected log-pointwise density (ELPD) to the best model (i.e., k=40).*

k	elpd_diff	se_diff	elpd_loo	se_elpd_loo	p_loo	se_p_loo	looic	se_looic
40	0.00	0.00	5,579.11	69.47	47.79	0.74	-11,158.21	138.94
80	-0.63	0.38	5,578.47	69.48	50.30	0.78	-11,156.94	138.97
20	-6.43	3.38	5,572.68	69.59	36.99	0.58	-11,145.35	139.17
10	-36.12	8.85	5,542.98	69.88	19.29	0.32	-11,085.97	139.76
5	-1,605.12	53.40	3,973.99	71.54	11.13	0.19	-7,947.98	143.09

## Appendix B

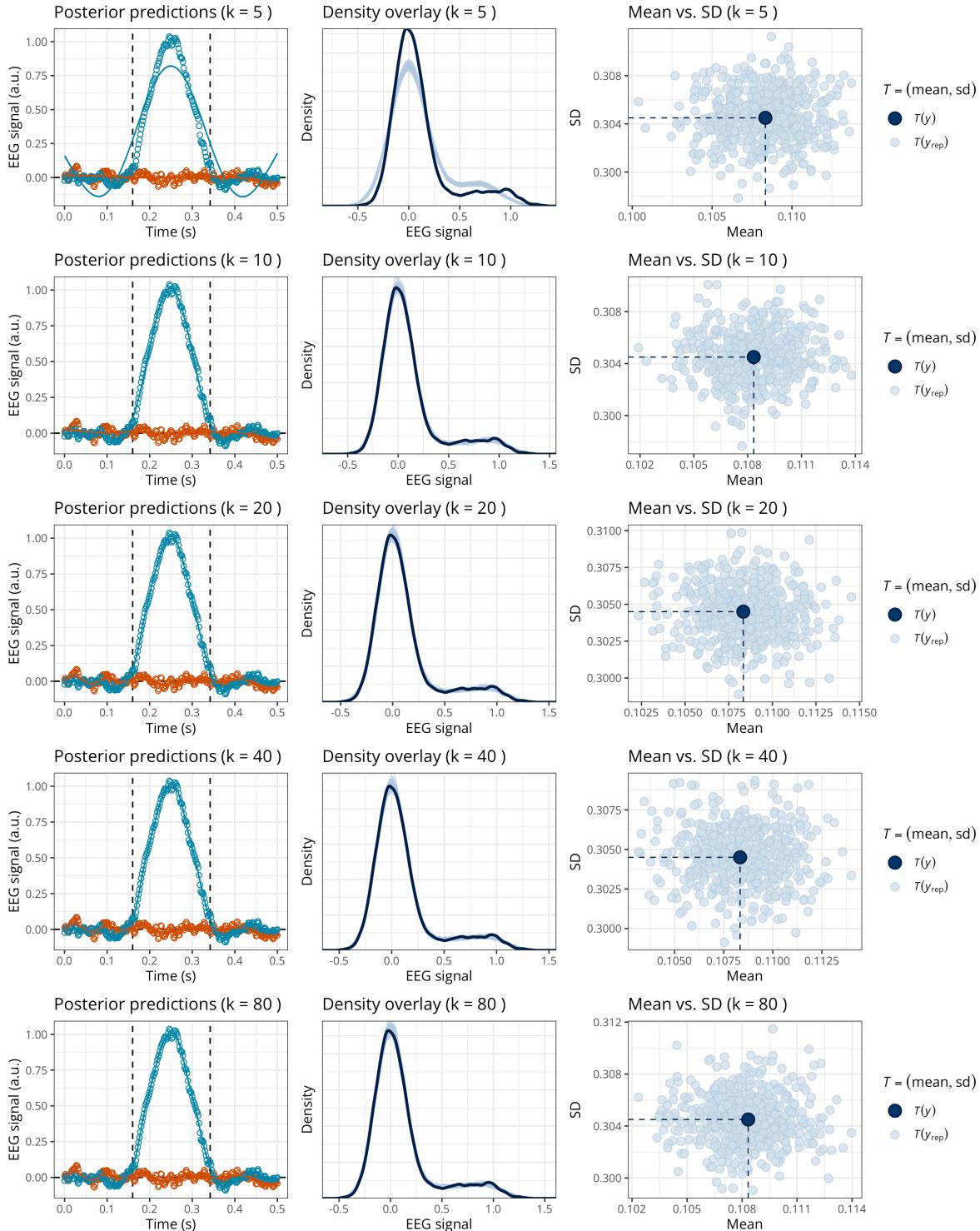
### How to choose the GAM basis dimension?

671 There is no universal recommendation for choosing the optimal value of  $k$ , as it depends on  
 672 several factors, including the sampling rate, preprocessing steps (e.g., signal-to-noise ratio, low-  
 673 pass filtering), and the underlying neural dynamics of the phenomenon under investigation. One  
 674 strategy is to set  $k$  as high as computational constraints allow, as suggested by previous authors  
 675 (e.g., [E. J. Pedersen et al., 2019](#)). Alternatively, one can fit a series of models with different  
 676  $k$  values and compare them using information criteria such as LOOIC or WAIC, alongside  
 677 posterior predictive checks (PPCs), to select the model that best captures the structure of the  
 678 data. We illustrate this approach below.

679 Figure B1 presents the posterior predictions and two forms of posterior predictive checks (PPCs)  
 680 for each GAM fit using different numbers of basis functions ( $k \in \{5, 10, 20, 40, 80\}$ ). With  
 681 the exception of the  $k = 5$  model, all other fits yield satisfactory PPCs, indicating that the  
 682 predicted data closely resemble the empirical observations. However, model comparison using  
 683 the leave-one-out information criterion (LOOIC), as summarised in Table 1, identifies the  $k = 40$   
 684 model as the best-performing one in terms of LOOIC, closely followed by the  $k = 80$  model.  
 685 This suggests that the optimal number of basis functions likely lies between these two values.  
 686 Future simulation studies could further investigate how such model selection criteria relate to  
 687 the precision of onset and offset estimates.

**Figure B1**

Posterior predictions and posterior predictive checks for the GAM with varying  $k$  (in rows).



## Appendix C

### R package and integration with MNE-Python

688 For readers who are already familiar with `brms`, the recommended pipeline is to use the code  
 689 provided in the main paper (available at [https://github.com/lalborczyk/brms\\_meeg](https://github.com/lalborczyk/brms_meeg)). It is

690 also possible to call functions from the `neurogam` R package (available at [https://github.com/  
691 lnalborczyk/neurogam](https://github.com/lnalborczyk/neurogam)) which come with sensible defaults.

```
# installing (if needed) and loading the neurogam R package
# remotes::install_github("https://github.com/lNALBORCZYK/neurogam")
library(neurogam)

# using the testing_through_time() function from the neurogam package
# this may take a few minutes (or hours depending on the machine's
# performance and the size of the dataset)...
gam_onset_offset <- testing_through_time(
  # data frame with M/EEG data in long format
  data = raw_df,
  # the *_id arguments are used to specify the relevant columns in data
  participant_id = "participant", meeg_id = "eeg",
  time_id = "time", predictor_id = "condition",
  # posterior odds threshold for defining clusters (20 by default)
  threshold = 20,
  # number of warmup MCMC iterations
  warmup = 1000,
  # total number of MCMC iterations
  iter = 5000,
  # number of MCMCs
  chains = 4,
  # number of parallel cores to use for running the MCMCs
  cores = 4
)

# displaying the results
gam_onset_offset$clusters
```

692 The `neurogam` package can also be called from Python using the `rpy2` module, and can easily be  
 693 integrated into MNE-Python pipelines. For example, we use it below to estimate the onset and  
 694 offset of effects for one EEG channel from an MNE evoked object. The code used to reshape  
 695 the `sample` MNE dataset is available in the online supplementary materials, and we further refer  
 696 to the [MNE documentation](#) about converting MNE epochs to Pandas dataframes in long format  
 697 (i.e., with one observation per row).

```
# loading the Python modules
import rpy2.robj as robj
from rpy2.robj.packages import importr
from rpy2.robj import pandas2ri
from rpy2.robj.conversion import localconverter
```

```
# importing the "neurogam" R package
neurogam = importr("neurogam")

# activating automatic pandas-R conversion
pandas2ri.activate()

# assuming reshaped_df is some M/EEG data reshaped in long format
with localconverter(robjects.default_converter + pandas2ri.converter):

    reshaped_df_r = robjects.conversion.py2rpy(reshaped_df)

# using the testing_through_time() function from the neurogam R package
gam_onset_offset = neurogam.testing_through_time(
    data=reshaped_df_r,
    threshold=20,
    multilevel=False
)

# displaying the results
print(list(gam_onset_offset) )
```