A tutorial on fitting joint models of M/EEG and behavior to understand cognition

Michael D. Nunez[a,*], Kianté Fernandez[b], Ramesh Srinivasan[c,d,e], Joachim Vandekerckhove[c,e,f]

[a]*Psychological Methods, University of Amsterdam, Amsterdam, The Netherlands*
[b]*Department of Psychology, University of California, Los Angeles, CA, USA*
[c]*Department of Cognitive Sciences, University of California, Irvine CA, USA*
[d]*Department of Biomedical Engineering, University of California, Irvine CA, USA*
[e]*Institute of Mathematical Behavioral Sciences, University of California, Irvine, CA, USA*
[f]*Department of Statistics, University of California, Irvine CA, USA*

Abstract

We present motivation and practical steps necessary to find parameter estimates of joint models of behavior and neural electrophysiological data. This tutorial is written for researchers wishing to build joint models of human behavior and scalp and intracranial electroencephalographic (EEG) or magnetoencephalographic (MEG) data, and more specifically those researchers who seek to understand human cognition. Although these techniques could easily be applied to animal models, the focus of this tutorial is on human participants. Joint modeling of M/EEG and behavior requires some knowledge of existing computational and cognitive theories, M/EEG artifact correction, M/EEG analysis techniques, cognitive modeling, and programming for statistical modeling implementation. This paper seeks to give an introduction to these techniques as they apply to estimating parameters from neurocognitive models of M/EEG and human behavior, and to evaluate model results and compare models. Due to our research and knowledge on the subject matter, our examples in this paper will focus on testing specific hypotheses in human decision-making theory. However most of the motivation and discussion of this paper applies across many modeling procedures and applications. We provide Python (and linked R) code examples in the tutorial and appendix. Readers are encouraged to try the exercises at the end of the document.

Keywords: Computational modeling, Cognitive modeling, Electroencephalography (EEG), Magnetoencephalography (MEG), Psychology, Neuroscience

1. Motivation to model

Joint modeling, or models that link distributions from multiple modalities, are useful tools for studying cognition (Teller, 1984; Hanes and Schall, 1996; Schall, 2004). There is considerable value to joint modeling in M/EEG measures to behavioural data to answer

---

*Corresponding author
Email address: `m.d.nunez@uva.nl` (Michael D. Nunez)

questions about cognition. Fitting joint models of M/EEG and human behavior have been used to answer diverse questions about cognitive topics such as working memory (Zhang et al., 2018), reinforcement learning (Frank et al., 2015; Swart et al., 2018), cognitive abilities (Schubert et al., 2019), and even the study of dyslexia in children (Manning et al., 2021). Schubert et al. (2019) show that neural processing speed, as reflected in stimulus-locked EEG measures, describes variation in cognitive task performance across individuals. In another study, Nunez et al. (2015) show that EEG measures of attention can reveal how individual differences in visual attention affect differentiable cognitive components of decision-making.

What do we hope to achieve by finding parameter estimates of joint models of M/EEG and human behavior? We could use these parameter estimates to draw conclusions about a scientific hypothesis, to help differentiate between theories, to find models that best predict human behavior and brain dynamics, or to teach students how to fit models to data. Researchers may also find intrinsic value in using joint modeling of M/EEG versus other neuroimaging modalities such as fMRI due to the high temporal resolution and often a shared scale with behavioral data measured in time, e.g. a shared scale of seconds after a stimulus between M/EEG and response time measures. While the possible goals that could be realized by joint modeling of M/EEG and human behavior are numerous (e.g. see Kording et al., 2018), most researchers who fit joint models will seek to either (1) test specific hypotheses or differentiate theories in fields such as Neuroscience and Psychology or (2) maximize prediction of M/EEG signals and/or human behavior. Sometimes these two goals can be simultaneously realized, but maximizing prediction of M/EEG and human behavior is often best achieved with atheoretical approaches based in Machine Learning (ML) or Artificial Intelligence (AI). ML and AI are best used in scenarios where maximizing prediction is most important and understanding the cognitive process is not critical, for instance in many Brain Computer Interfaces (BCIs).

In this tutorial we will focus primarily on joint modeling for testing specific hypotheses. The diversity in *methods* used to perform joint modeling of M/EEG and behaviour is large. The studies mentioned above include a variety of methods to perform joint modeling. There are also many related studies that correlate cognitive model parameters to observed M/EEG measures or correlate observed behavioural measures to computational parameters of EEG (e.g. O'Connell et al., 2012; Gluth et al., 2013; Jagannathan et al., 2021). Our intention in this tutorial is not to review all cognitive topics (e.g. see Hawkins et al., 2023) nor all joint modeling techniques of M/EEG and behaviour (e.g. see Palestro et al., 2018). Instead we will focus on clarifying common modeling examples, EEG data collection and analysis, and tools to implement joint models. We will cover experimental design, M/EEG analysis and behavioral analysis techniques necessary for joint modeling, as well as the modeling itself. Due to our research and knowledge on the subject matter, our examples in this paper will focus on testing specific hypotheses in human decision-making theory. However all techniques and software presented here can be applied to testing any formal hypotheses involving the relationship of M/EEG to human cognition and behavior.

We expect a diversity in readership of this tutorial from different disciplines. We primarily expect readers trained in two different sub-fields, namely, Cognitive Neuroscience and Mathematical Psychology. Therefore, some readers may find certain sections introductory. Readers trained in Cognitive Neuroscience should place increased focus on Sections 2 and 5, which give the motivation and techniques for modeling, while readers trained

in Mathematical Psychology should focus on Sections 3 and 4 that provide experimental motivations and the theory and practice of M/EEG. Other readers may need to read the entire tutorial carefully, as well as follow the references in *Further readings* (located at the end of the paper) corresponding to each main Section if looking for more background or depth on a certain topic. Finally, we encourage readers to read and run the provided example scripts in either Python (https://github.com/mdnunez/pyhddmjags) or R (https://github.com/kiante-fernandez/Rhddmjags) and work through the *Exercises* at the end of the document.

1.1. One example topic: Decision-making

One topic of interest is whether specific M/EEG signals reflect cognitive components of decision-making. This has been the focus of a large body of previous work (e.g. see O'Connell et al., 2018) including our own work (see **Figure 1**, Lui et al., 2021). This work has lead to specific testable questions that can be answered with joint modeling of M/EEG data, response time data, and choice data from tasks in which participants make simple decisions. However before we discuss this example of joint modeling work, let us first briefly review a key theory of decision making.

Sequential sampling models assume that humans and animals accumulate evidence for a particular choice over the course of a decision by *sampling* from external or internal evidence. This *evidence* for a decision is usually considered a cognitive representation or a direct neural representation (e.g. changing firing rates of neurons over time). Simulating and fitting sequential sampling models are particularly useful for understanding quick decisions on the scale of seconds. These models make predictions about the time course of decision-making, while other decision-making models, such as Signal Detection Theory (SDT; Hautus et al., 2021) do not make any predictions about the time course of decision-making. Drift-diffusion models (DDMs; Ratcliff, 2018) are a particular class of *sequential sampling* models that assume a Wiener process of evidence accumulation. A Wiener process of evidence accumulation is a random walk process with an infinitesimal (infinitely small) time step (see middle of **Figure 1**). However, note though that DDMs and associated model-variants are often used due to the models' mathematical utility, rather than a theoretical belief that a time step of evidence accumulation should be infinitesimal in the brain.

Sequential sampling theory leads naturally to specific testable questions that can be answered with joint modeling of M/EEG data. Do time-averaged event-related potentials (ERPs) encode the demarcation point between visual encoding and evidence accumulation (Nunez et al., 2019a)? Do motor preparation signals over the motor cortex track evidence accumulation time (Lui et al., 2021)? How do EEG measures of visual attention affect the decision-making process, and in what precise way does visual attention affect different computational components of decision-making (Nunez et al., 2015, 2017)? A table of related questions that can be answered with neurocognitive models using DDMs can be found in **Table 1**. There exist multiple analysis methods to help answer these questions with EEG and behavior (see Bridwell et al., 2018). However we have preferred to use the implementation of joint modeling of EEG and human behavior to understand data from participants who performed hypotheses-differentiating experiments.

1.2. One example question: Do EEG signals encode sequential-sampling of evidence?

One more specific question is whether EEG signals encode a sequential-sampling of evidence, such has been found in single neurons and neural populations within intracranial recordings of the lateral intraparietal (LIP) cortex and superior colliculus (SC) of the macaque brain during single experimental trials (Roitman and Shadlen, 2002; Shadlen and Kiani, 2013; O'Connell et al., 2018; Jun et al., 2021). EEG signals time-locked to specific events such as the onset of a visual stimulus, e.g. the P300 / Centro-Parietal Positivity (CPP) waveform, and EEG signals time-locked to the response, e.g. the Readiness Potential (RP), have been proposed to be related to evidence accumulation and the timing of decisions (O'Connell et al., 2012; Gluth et al., 2013; Twomey et al., 2015; van Ravenzwaaij et al., 2017; Lui et al., 2021). We expect future joint modeling work will further help differentiate whether these signals are exactly encoding evidence accumulation, correlated processes, or mixtures of signals (Philiastides et al., 2014). Related questions that can be answered with joint modeling work are: (1) what EEG preprocessing steps and recording procedures should be used to best extract evidence accumulated related signals? and (2) for what specific conditions and task paradigms these signals encode evidence accumulation? In this paper we show how combining these specific EEG signals and behavioural data in neurocognitive modeling will lead to better and more extensive knowledge of individual differences and single-trial estimates of human cognition. In our example methods presented in this paper, we build models and present model fitting procedures that can best answer the question of whether EEG signals encode a sequential-sampling of evidence.

2. Models to describe joint data

2.1. The basic terms of modeling

Cognitive models include parameters of psychological processes that describe human (or animal) behavior. These models are often used to describe behavior in psychological experiments or natural environments, and these models are often developed by researchers in the scientific field of Mathematical Psychology. A *parameter* of a cognitive model is a variable that can take a pre-specified range of values that describe data, and multiple parameters of a model are usually required to describe data. The parameters of cognitive models often directly relate to unobserved psychological concepts such as memory capacity or general cognitive ability (Lee et al., 2019; Schubert et al., 2019). Signal Detection Theory (SDT) that explains choice and accuracy data could be considered a cognitive model since it contains two parameters that describe both the ability and choice bias of a human participant (Hautus et al., 2021). Another example are Drift-Diffusion Models (DDMs) of choices and response time (RT) data during human decision-making, which contain cognitive parameters that describe speed-accuracy trade-offs, speed of evidence accumulation for one choice or another, and decision biases (Ratcliff et al., 2016). The cognitive interpretations of parameters of new cognitive models should be tested by experimentation with differentiating experimental conditions (discussed below). However some widely-used cognitive models, such as SDT and DDMs, have parameters whose cognitive interpretations are now widely accepted by researchers due to the results of multiple experimental studies. For instance, Voss et al. (2004); Dutilh et al. (2019) generally found that parameters of DDMs (namely speed-accuracy trade-off parameters, non-decision time parameters, evidence accumulation rate parameters, and
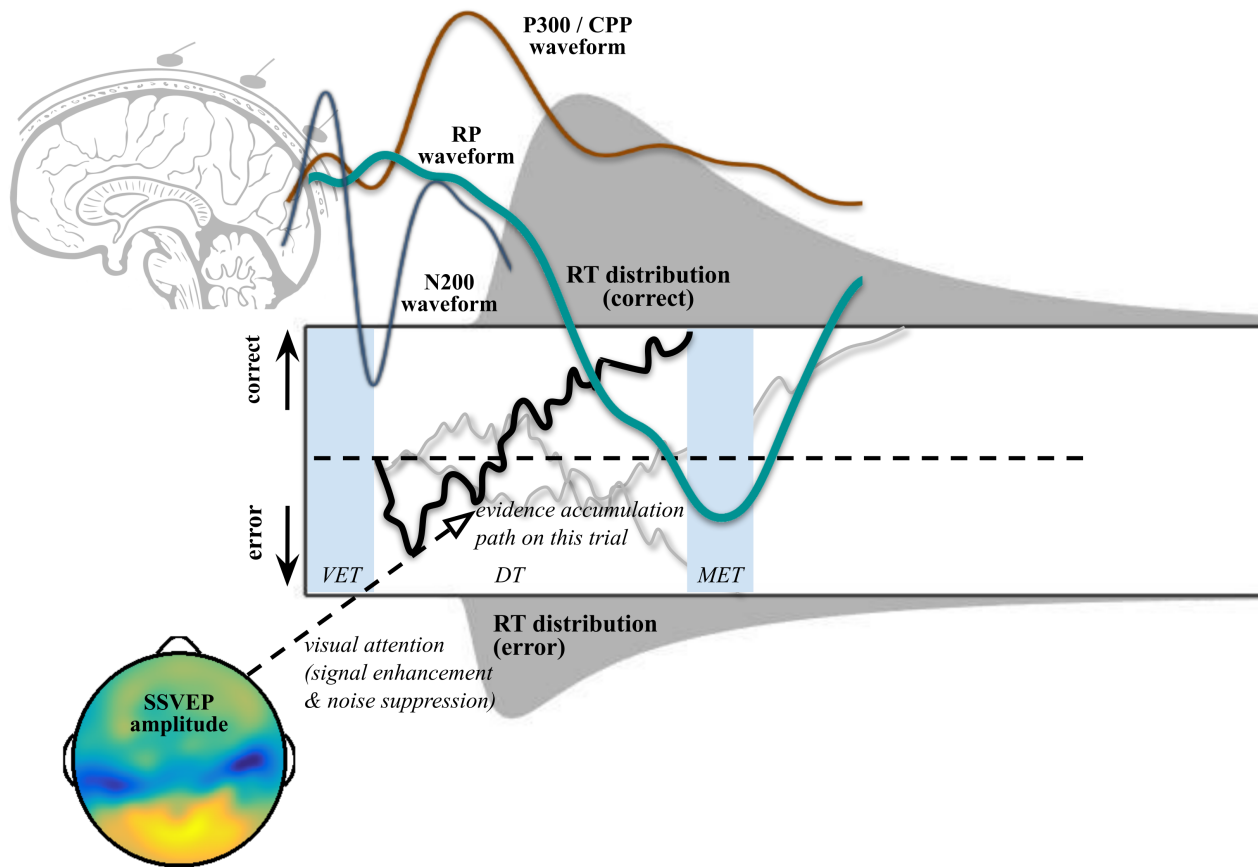
4

Figure 1: A theoretical representation of some modeling studies to discover cognitive mechanisms of decision-making using *neurcognitive* modeling of EEG and human behavior during decision-making tasks. Bold text represents observed data (EEG measures or human behavioral data) while italic text represents derived cognitive parameters that can be estimated through joint modeling. Event-Related Potentials (ERPs; represented by 3 waveforms beginning at the cartoon image of the brain in the top right) and frequency-domain EEG measures (bottom left: EEG amplitudes that were spline-interpolated between electrodes on a flat representation of the human scalp) have been used in joint modeling to understand human cognition in the context of Neurocognitive Drift-Diffusion Models (NCDDMs). Human behavioural data such as choice-RTs (response time distributions shown for correct responses, top, and error responses, bottom flipped) are also used to fit NCDDMs and Drift-Diffusion Models (DDMs). In NCDDMs, like DDMs, correct and error responses are described after enough cognitive evidence is reached, representing by the cognitive evidence accumulation passing one of two boundaries during decision time (this trial represented as a black line with two other grey lines representing other simulations from the same process that describe response times and possibly EEG potentials). Particular ERPs of interest are N200, P300 / CPP, and RP waveforms. N200 waveforms are thought to reflect visual encoding time (VET) and the onset of evidence accumulation (Nunez et al., 2019a). The P300 or Centro-parietal positivity (CPP) are thought to reflect decision time (DT) and possibly the evidence accumulation process itself (O'Connell et al., 2012; Kelly and O'Connell, 2013; O'Connell et al., 2018; van Ravenzwaaij et al., 2017). The Readiness Potential (RP) is a motor related preparatory signal thought to reflect DT and motor execution time (MET) under certain experimental conditions (Lui et al., 2021). Steady-state visual evoked potentials (SSVEPs) can be used to estimate visual attention and in particular, signal enhancement and noise suppression that could affect the rate and variance of evidence accumulation (Nunez et al., 2015). A table of related questions and other neurocognitive work using DDMs can be found in **Table 1**.

evidence bias parameters) that describe human choice and response time are all manipulated in expected directions by proper experimental conditions. Although these experimental manipulations are not perfect, and they often affect other cognitive parameters (Dutilh et al., 2019), these prior results allow us to draw conclusions in new data while assuming some of these parameters map onto the expected cognitive function.

Cognitive models are sometimes defined differently from *computational models*. Computational models often focus on modeling brain dynamics with parameters that have specific neural correlates (Blohm et al., 2020; Glomb et al., 2021) (a short discussion of M/EEG generators can be found in the Discussion section). Typically computational models are used in the scientific field of Computational Neuroscience. Though computational models do not necessarily include parameters with interpretable psychological processes. However, because computational models may describe human behavioral data, the term "computational model" is sometimes used more broadly to encompass cognitive models as well (e.g. Wilson and Collins, 2019).

Neurocognitive models are joint models of brain activity and human behavior that seeks to combine cognitive modeling with necessary links between (1) brain dynamics as measured by or derived from M/EEG, (2) cognition and other psychological concepts expressed as formal models, and (3) human behavioral data. As a generalization, with neurocognitive modeling we seek to understand how macro-level neurophysiology (as measured by scalp-recorded EEG, MEG, or even depth-recorded EEG) encodes human cognition which gives rise to human behavior. For instance, evidence accumulation is a cognitive process during decision-making, but may also have direct neural correlates in EEG (Forstmann et al., 2016; O'Connell et al., 2018; Lui et al., 2021), and thus drift rates of DDMs could describe choice-RT distribution shapes, evoked EEG potentials, or both simultaneously. Thus these *neurocognitive* models can be used to develop and test theories in both Psychology and Neuroscience. Specifically we focus in this paper on how modeling can be used to directly test hypotheses which involve observed M/EEG dynamics, human cognition, and human behavior. Note that *neurocognitive modeling* is less defined than *joint modeling*, as is defined more specifically for certain classes of neurocognitive models (see Turner et al., 2017; Palestro et al., 2018). In the next section we describe three examples of neurocognitive models, and Models 2 and 3 could be considered joint models in the labeling scheme of Turner et al. (2017) and Palestro et al. (2018).

When *simulating* models, parameters are fixed to certain values and a model generates synthetic data using programs such as R and Python that could be compared to real data. For instance, a neurocognitive model) with multiple user-defined parameters could generate synthetic (but informative) EEG potentials, choices, and response times. To *fit* a model is to discover a set of parameter estimates that best describe known data given the model architecture and assumptions. *Fitting* a model using Bayesian methods means finding parameter uncertainties from which parameter estimates can be derived. Fitting models to data is a useful method to test hypotheses either (1) by directly estimating and then evaluating parameters (e.g. compare parameter estimates across experimental conditions) or (2) comparing multiple models' ability to describe data.

2.2. Translating neurocognitive theory into mathematical models

Often modeling involves simplifying the broader mathematically-defined theories of the brain and human behavior to fit data accurately and efficiently. In this way theory, or an

approximation of theory, can be used to answer questions directly. Fitting models to EEG and behavioral data requires (1) knowledge of cognitive or neurocognitive theory, and (2) effort to quantitatively formalize the hypotheses to be tested in the context of theory. Good theory should be quantitatively defined (Oberauer and Lewandowsky, 2019a), and the best theory is precisely mathematically defined (Guest and Martin, 2021). Although many other qualifications may be needed for good theory, these are discussed elsewhere (e.g. van Rooij and Baggio, 2020).

A model should be able to be written as a series of mathematical equations and statistical distributional statements that describe the data. Let us assume that a participant, Roos, wore an EEG cap while playing a simple video game where she made a correct or incorrect answer on multiple trials of the game. Researchers extracted EEG and behavioral data from the data collection hardware and are now interested in how the Centro-Parietal Positivity (CPP) rise over time within a trial (e.g. the CPP "*slope*") influences accuracy and response times (RTs). Note that the CPP slope is a signal in the EEG that is found after the onset of visual stimuli, and is thought to be a reflection of the computation of visual evidence in the brain (O'Connell et al., 2012; O'Connell et al., 2018). Specifically the researchers are interested in how three measures from Roos are related: CPP slopes, accuracy, and RTs. The researchers obtain observations of each of the three measures on every trial of the game. One simple joint model could just assume that RTs and accuracy are influenced by the CPP slope. Thus a simple model (**Model 1**) would just be two attached models, linear regression and logistic regression. Let CPP slope be denoted by the variable c, accuracy be denoted by x, and response times be denoted by r. Note that accuracy x can be either 0 or 1, and that x, r, and c can vary on every trial i and by participant j. The parameters of the model can change by participant j. Researchers can then fit this model to Roos' data as well as other participants.

$$r_{ij} \sim Normal(\theta_{0j} + \theta_{1j}c_{ij},\ \sigma_j^2) \tag{1}$$

$$x_{ij} \sim Bernoulli(p_j) \tag{2}$$

$$\log\left(\frac{p_j}{1-p_j}\right) = \gamma_{0j} + \gamma_{1j}c_{ij} \tag{3}$$

Note that the symbol $\sim$ denotes *distributed by*, so that response times r come from a normal distribution with a mean parameter that changes with a linear function of CPP slopes c and variance σ^2. This equation represents simple linear regression. The second distributional statement and the last equation represent logistic regression. We picked the logit ("log-odds") function $logit(p) = \log(p_j/1-p_j)$ within the logistic regression framework, although we could use any function that maps probabilities bounded from 0 to 1 to the continuous $(-\infty, \infty)$ scale. The θ and γ parameters in this model provide the relationship of EEG measure to behavior. Note that we can test whether the θ_1 and γ_1 parameters are near zero in order to test many hypothesized relationships between CPP slope and behavior. Alternatively, we could compare the fit of the behavioral data in this model to a model where the EEG does not influence the data generators. In a comparison model, response times and accuracies would be described only by the mean and probability parameters (e.g. setting θ_1 and γ_1 both to 0 before model fitting). Note that **Model 1** does not assume EEG reflects any

7

particular type of cognition or computation. However our example question is more specific about the type of computation and cognition the CPP slope could reflect. Namely, that CPP slope reflects evidence accumulation speed. **Model 1** assumes that any relationship is linear between CPP slopes and response times as well as CPP slopes and log odds of accuracies. Fitting **Model 1** will likely yield information about the existence of a CPP-behavior relationship because linear and logistic regression are useful in finding relationships where there is *some* true relationship between variables. **Model 1** would be particularly helpful if we wanted to test if there was any possible relationship in an exploratory analysis.

Researchers instead might want to directly test whether the CPP slopes reflect specific cognitive components of decision-making that describe both the accuracy and response time jointly. A second model (**Model 2**) could then test simple linear relationships to cognitive parameters. In particular we might be interested in how CPP slopes describe the drift-rate parameters δ and the non-decision time parameters τ. The drift rate δ reflects the mean rate of evidence accumulation for each infinitesimal time step of a Wiener process (Ratcliff et al., 2016). Non-decision time τ on every trial is any time in a response time not due to a Wiener process (such as visual encoding time; VET). Lets assume that δ and τ can vary on every experimental trial. We also assume that other parameters of a DDM do not vary with CPP slope and are fixed across trials, namely: the boundary separation parameter α_j which describes the amount of relative evidence to make a correct choice, the initial bias parameter β_j towards the correct choice before evidence accumulation occurs, and trial-to-trial variability in drift rate, η_j, that is not due to CPP slope variability. This DDM describes both response times r and choices x per participant j and trial i.

$$(r_{ij}, x_{ij}) \sim DDM(\delta_{ij}, \tau_{ij}, \alpha_j, \beta_j, \eta_j) \tag{4}$$
$$\delta_{ij} = \xi_{0j} + \xi_{1j}c_{ij} \tag{5}$$
$$\tau_{ij} = \lambda_{0j} + \lambda_{1j}c_{ij} \tag{6}$$

Note that the parameters ξ and λ have different meanings in this model with embedded cognitive components compared to the θ and γ parameters in the previous model. The parameters ξ and λ are the intercept and effect parameters of the CPP slopes on the evidence accumulation rate and non-decision time respectively. We will call this model a *neurocognitive* model because it assumes particular types of cognition during decision making and contains neural data, in addition to behavioral data. We have fit this class of model, which assumes single-trial EEG measures describe single-trial DDM parameters, in previous work (Nunez et al., 2017, 2019a).

Another class of neurocognitive models involves describing the cognitive parameters through the EEG measures themselves. For instance, we can directly test the underlying computational role of the CPP slope in cognition by testing how well the following model fits the data once parameters are estimated and how well this model predicts new data. In **Model 3**, the mean of each trial's CPP slope c is described by the drift rate for each participant j.

$$(r_{ij}, x_{ij}) \sim DDM(\delta_j, \tau_j, \alpha_j, \beta_j, \eta_j) \tag{7}$$
$$c_{ij} \sim Normal(\delta_j, \ \sigma_j^2) \tag{8}$$

8

such that five cognitive parameters of a DDM, δ, τ, α, β, and η vary by participant j. Note that one of those cognitive parameters, the mean rate of evidence accumulation across trials δ is also a *computational parameter*, a parameter that describe neural data in a *computational model* define above, that describes the CPP slope on every trial i and each participant j. There is one more additional computational parameter in this model that describes the observation noise of the CPP slope on every trial i, parameter σ for each participant j. This model is similar to, although more specific than, a model previously used by van Ravenzwaaij et al. (2017).

Better neurocognitive models might extend **Model 3** by describing the CPP slopes from a function of multiple cognitive parameters, for instance. Other models could contain computational parameters that reflect brain dynamics that are described by cognitive parameters. Different modeling strategies are discussed by Turner et al. (2017). All of the example models and other joint models can be simulated through the equations listed and by choosing a set of values for parameters. We can build a variety of neurocognitive models to test specific theories.

2.3. The importance of model simulation

Model simulation is usually the first step in generating new models, and can be helpful in understanding the neurocognitive theory. Model fitting intrinsically makes many assumptions about the data. Joint modeling of M/EEG and behavior will make assumptions about cognitive and computational processes in the brain and implicit assumptions about what cognitive and computational processes are **not** occurring. These assumptions will affect parameter fitting results significantly. It is therefore imperative that the researcher understand what assumptions they are making and how those assumptions can be violated.

The best way to explore these assumptions is through the two-step process of (1) simulation of models using different model types and realistic parameter ranges and then (2) fitting all the simulated data using the specific model and fitting procedure to be used in the analysis of real data. Simulations can show the researcher under which conditions the fitting procedure can break or produce spurious results. Simulation of models, as opposed to fitting of models, is also one way to formally define the underlying *theory* to be tested, in that it is quantifiable, and perhaps complex, but rigorously defined (Guest and Martin, 2021).

To simulate joint models of M/EEG, it is best to write the model using existing statistical libraries with coding languages such as Python or R. We reproduced one possible simulation of **Model 3** here. A code snippet of **Model 3** is provided in **Code Block 1** with associated full Python and R code located at `https://github.com/mdnunez/pyhddmjags/blob/master/simpleCPP_sim.py` and `https://github.com/kiante-fernandez/Rhddmjags/blob/main/R/Rhddmjagsutils.R` respectively as of July 2023. We first generate random values of all parameters for one participant. We then simulate data for all trials i for this participant from those parameters within the two equations of **Model 3** in a for loop. Our code simulates the model directly from an approximation of the Wiener process in line 9 (specifically approximated using the Euler method, see Brown et al., 2006). The approximation can be thought of as a random walk process with a very small time step that loops over `nsteps` (line 8), until the random walk passes the upper boundary `alpha` (line 11) or lower boundary `0` (line 16). This is especially useful for observing simulations of the evidence paths on every trial (see top left of **Figure 2**). Our code also simulates the CPP observations themselves from a

Table 1: A table of example hypotheses that could be tested directly using neuro-cognitive Drift Diffusion Models (DDMs) of M/EEG and behaviour. Each hypothesis is derived directly from existing literature in the fields of Cognitive Neuroscience and model-based Cognitive Neuroscience. Each *cognitive mechanism* is associated with a specific *computational role* in the human brain that is measured with a M/EEG *neural signature* that is hypothesized to be reflected in the relationship with a cognitive parameter of a DDM. δ refers to the drift rate parameter, τ refers to the non-decision time, τ^v refers to visual encoding time (a component of non-decision time) τ^m refers to motor execution time (another component of non-decision time) α refers to the boundary separation parameter, β refers to the initial bias parameter, ς refers to the diffusion coefficient (undiscussed in this text but discussed in Ratcliff et al., 2016; Nunez et al., 2017, and elsewhere).

Cognitive Mechanism	Computational Role	Neural Signature	DDM Parameters	References
Visual evidence accumulation	Evidence accumulation rate	P300 slopes	δ	(Philiastides et al., 2006; Ratcliff et al., 2009; Philiastides et al., 2014; Twomey et al., 2015; van Ravenzwaaij et al., 2017; Kohl et al., 2020)
Subjective-value evidence accumulation	Evidence accumulation rate	Gamma (46–64 Hz) power	δ	(Polanía et al., 2014)
Figure-ground segregation	Visual encoding time (VET)	N200 latencies	τ / τ^v	(Loughnane et al., 2016; Nunez et al., 2017, 2019a; Ghaderi-Kangavari et al., 2022, 2023a)
Motor execution	Motor execution time	Beta (15-25 Hz) desynchronization	τ / τ^m	(Crone et al., 1998; McFarland et al., 2000)
Motor cortex preparation	Motor evidence accumulation	Readiness Potentials	δ	(Gluth et al., 2013; Lui et al., 2021)
Speed-accuracy tradeoff	Changing neural threshold	Theta (4-7 Hz) power	α	(Cavanagh et al., 2011; Frank et al., 2015)
Strategy adjustment	Changing neural threshold	Contingent Negative Variation	α	(Boehm et al., 2014)
Prestimulus activation	Bias	Occipital EEG amplitude to predict choice	β	(Bode et al., 2012)
Visual attention	Variability in evidence (diffusion)	Steady-state visual evoked potentials	δ, ς	(Nunez et al., 2015; Rangelov and Mattingley, 2020)
Attentional gating	Internal neural noise	Alpha (8-12 Hz) power	ς	(Pfurtscheller et al., 1996; Jensen and Mazaheri, 2010; Klatt et al., 2020)

sine wave (line 6), commonly used for simulating oscillatory signals (Cohen, 2014), with 1/4 of the period of the sine wave being the CPP slope (see bottom left of **Figure 2**). Protocols for simulating EEG data range in biological and physical plausibility, however sine waves are a common starting point for simulating EEG activity (Cohen, 2014; Hagen et al., 2022; Næss et al., 2021), see also the section on modeling of M/EEG generators in the Discussion of this paper.

```python
plot_time = np.linspace(0, step_length * nsteps, num=nsteps)
for n in range(0, ntrials):
    random_walk = np.empty(nsteps)
    drift = stats.norm.rvs(loc=delta, scale=eta)
    cpp_slopes[n] = stats.norm.rvs(loc=delta, scale=sigma)
    CPPs[:, n] = np.sin(2 * np.pi * ((cpp_slopes[n] / 4) * (plot_time -
    ndt)))
    random_walk[0] = beta * alpha
    for s in range(1, nsteps):
        random_walk[s] = random_walk[s - 1] + stats.norm.rvs(
            loc=drift * step_length, scale=varsigma * np.sqrt(step_length)
    )
        if random_walk[s] >= alpha:
            random_walk[s:] = alpha
            rts[n] = s * step_length + ndt
            choice[n] = 1  # Correct choice shown with positive RTs
            break
        elif random_walk[s] <= 0:
            random_walk[s:] = 0
            rts[n] = s * step_length + ndt
            choice[n] = -1  # Incorrect choice shown with negative RTs
            break
        elif s == (nsteps - 1):
            rts[n] = np.nan
            choice[n] = np.nan
            break
```

Code Block 1: Python code from simulation of **Model 3**

When simulating a model it is useful to plot elements of the model itself. For instance in **Figure 2** we have plotted dynamics of the model itself, namely the CPP waveforms on single-trials and the evidence paths themselves. We have also plotted the distributions of both the choice response times and the CPP slopes. We can also choose to plot other diagnostics such as the cumulative distribution function, and return specific statistics about our simulated data (e.g. mean, median, maximum, and minimums). In this way we can make sure our model, and quantitatively defined theories, are logical and could describe real data. Note however that we may not observe everything about our model in real data nor be able to estimate everything about our model from the data. For instance, we may never observe the cognitive evidence paths themselves (top left of **Figure 2**) in real data. Later, after we discuss model fitting, we will discuss how we can also use simulations to test parameter recovery. We will show that *fitting* **Model 3** to data is indeed useful.
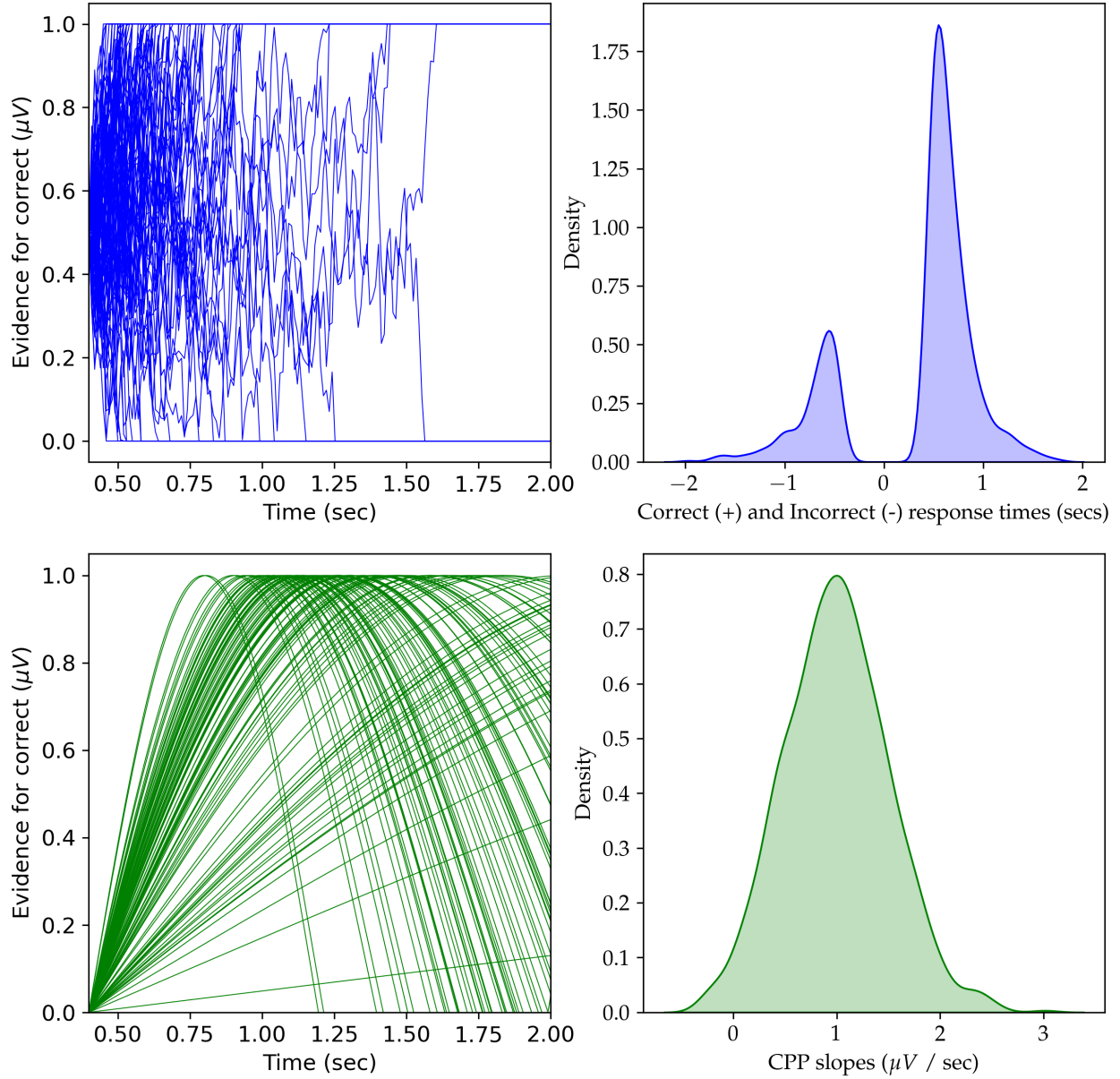
Figure 2: Diagnostic plots of the simulation of **Model 3**. We simulated **Model 3** using the provided Python code with 1000 trials for one participant. The top left figure is the simulated evidence accumulation paths from a Wiener process that reaches one of two boundaries to make a decision. The top right figure is the estimated density of incorrect and correct response times, with incorrect response times plotted as negative response times. The bottom left figure is the simulated CPPs on single-trials using a simple sine wave. The bottom right figure is approximated density of the simulated CPP slopes. This density should be approximately normal because of our modeling assumption for CPP slopes.

3. Experimental manipulations and experimental design

The goal of experimentation should be to design experimental conditions that best answer scientific questions. In this section we will focus on experiments that are optimally designed to produce data for use with joint modeling. One particular *neurocognitive* theory suggests that a particular trial-averaged EEG signal, the CPP, in response to the onset of a picture on a computer screen is expected to be a neural signature of evidence accumulation during decision-making based on that picture. For example, the CPP reflects evidence accumulation to decide whether a noisy picture is a face or a car (e.g. Ostwald et al., 2012). This is a theory suggested by experimental and theoretical work by O'Connell et al. (2012); O'Connell et al. (2018), previously mentioned. Thus we should design experiments to collect M/EEG and behavioral data and/or pick an existing data set that (1) best test the hypothesis and (2) test the limits of this hypothesis.

3.1. Hypothetical Experiment 1

We know that drift rates in perceptual decision making tasks are affected by perceptual difficulty (Voss et al., 2004; Dutilh et al., 2019). Therefore if CPPs are signatures of the computational mechanism of evidence accumulation in the brain, we expect drift rates and CPP slopes to both change across perceptual difficulty conditions, and specifically maintain the same relationship across many experimental conditions of various difficulty. For instance participants could perform a Random Dot Motion (RDM) task with many different levels of perceptual difficulty across trials, a replication of work by Kelly and O'Connell (2013). An RDM task is a task in which a field of moving dots appears on each experimental trial, with a (typically small) percentage of dots moving in a specific direction (e.g. see Newsome and Pare, 1988; Gherman and Philiastides, 2018). Typically participants must choose between one of two directions. If all the dots moved together on the screen, the task would be too easy and would not result in different performance across conditions, not optimally testing our neurocognitive theory. Therefore the other percentage of dots typically move completely at random with no orientation information. The percentage *coherence* of dots describes the percentage of dots that move in the correct orientation. Therefore let us imagine a task in which trials are intermixed with 32%, 16%, 8%, 4%, and 0% coherence values. Note that the difficulty of the coherence values will depend, among other factors, on the size of the dots in the stimulus and the size of the entire stimulus for the participant.

We can fit the data from Experiment 1 to **Model 2** for each experimental condition k, allowing each parameter and data type to vary by condition k. By comparing parameter estimates of the slope parameter ξ_{1jk} across the two conditions, we can test whether there is a fixed linear relationship between the slope of the CPP and the drift rate in all experimental conditions, even when other parameters such as the drift-rate δ itself change across experimental conditions. Specifically we could develop statistical tests to test whether $\xi_{1j1} = \xi_{1j2} = \xi_{1j3} = \ldots = \xi_{1jK}$ from the resulting parameter estimates in all participants, for instance, we could analyze posterior distributions of these parameters to calculate posterior probability or Bayes Factors (see below). Alternatively we could compare how two models explain the data and predict new data. For instance, we could compare **Model 2A** to **Model 2B**, where **Model 2A** has fixed effect parameters ξ_{1j} across conditions and **Model 2B** has effect parameters ξ_{1jk} that are free-to-vary across conditions.

13

The number of trials and participants to collect in order to answer this question should depend on a *power analysis* (not to be confused with the concept of *EEG power* in the next section). We define a *power analysis* as any simulation-based or statistical theory-based analysis that will give an estimate of either the Bayesian or classical probability of finding true effects (or true null effects) for different sample sizes for specific analyses and joint models. For this particular experiment, we recommend fitting the models in simulation (see section *5.3. Parameter recovery of simulated models*) under the simulated truth where $\xi_{1j1} = \xi_{1j2} = \xi_{1j3} = \xi_{1j4} = \xi_{1j5}$ for 5 simulated conditions with intercept parameters ξ_{0jk} and τ_{0jk} of **Model 2** that change over the experimental conditions k. The ultimate planned analysis can then be run on the simulated data with different numbers of trials and participants. For our particular experimental question, we expect the number of trials to be more important than the number of participants (however participant-specific parameters could also be varied in simulation). Power analyses are always recommended over rules-of-thumb because each experimental question and modeling plan will necessitate different trial numbers. We expect that at least 100 to 500 trials per condition are necessary (Lerche et al., 2017). We skipped such analyses for this tutorial for the sake of brevity, and encourage readers to work with the provided simulation code.

3.2. Hypothetical Experiment 2

The first hypothetical experiment demonstrates a simple perceptual manipulation intended to drive cognition that best answers our hypothesis. In the second hypothetical experiment we propose an intervention indented to drive the brain response using transcranial direct current stimulation (tDCS). While Experiment 1 is a well established experimental manipulation, the intervention in Experiment 2 may or may not have any effect on the participants' brain response, cognition, and behaviour (Chrysikou et al., 2017; Mendes et al., 2022). However Experiment 2 is a useful example to show that proper control conditions are often necessary to test a hypothesis using joint modeling.

Because of the aforementioned theory, we might expect tDCS during a visual decision-making task to affect both CPP slopes c and evidence accumulation rate parameters (i.e. *drift rates*) δ in DDMs estimated from human behavior. A strong hypothesis is that we expect tDCS to affect both CPP slopes and drift rates equally since the theory is that the CPP is a signature of the computational mechanism of evidence accumulation in the brain. Thus tDCS could test the limits of the theory of the CPP reflecting evidence accumulation. We should at least have both (1) an experimental condition and (2) a proper control condition in which stimulating tDCS electrodes are applied to the participant's head.

In tDCS work an experimental control is often a sham condition in which tDCS is turned on then off after a ramping period before a block of experimental trials. This sham condition seeks to achieve the sensation of tDCS stimulation by the participant for a block of experimental trials, but to not actually stimulate during those trials (e.g. see Au et al., 2021). We propose an experimental design with one experimental condition (1) in which the experimenters stimulate brain areas expected to be involved in decision-making (perhaps placing stimulation electrodes on the scalp over parietal cortices), and a control condition (2) in which experimenters stimulate brain areas not expected to be involved in decision-making, say placing tDCS electrodes on the scalp over the temporal cortex. We could also include a different sham condition, another experimental control, (3) in which tDCS electrodes do

not stimulate the brain but current is still injected into the body, for instance placing and activating tDCS electrodes over the neck musculature in the back of the head.

We could again fit **Model 2B** to this data and analyze the results in a similar way to the previous experiment. For instance, suppose we fit the data to **Model 2B** and subsequently observe that both CPP slopes c and drift rates δ are significantly increased in condition (1) compared to sham condition (3), and that the two effect parameters of the model are equal, $\xi_{1j1} = \xi_{1j3}$. We also observe that only drift rates δ are significantly increased in condition (2) compared to the sham condition (3) and CPP slopes c are similar in both conditions, resulting in $\xi_{1j1} = \xi_{1j3} > \xi_{1j2}$. A simulation of this scenario and a simplified fitting procedure for **Model 2B** is included in `https://github.com/mdnunez/pyhddmjags/blob/master/model2b_experiment2.py`. These results would be evidence that the CPP slope reflects evidence accumulation only in specific conditions. For instance, this might suggest the the CPP only reflects visual evidence accumulation in the parietal cortex, and is not the brain-wide cognitive phenomena thought to be reflected in decision-making behavior. Of course, we would need further experimentation to test this new, more specific theory. We should also build a new joint model to better reflect our new neurocognitive theory.

3.3. Theoretically informed experiments

While we should be able to *simulate* joint models that account for many experimental designs, experimental design choices may also be made *in preparation* for *fitting* joint models. The data from some experimental designs are more easily modelled due to (1) larger bases of research knowledge for some specific theories of human cognition and brain signals and (2) the availability of algorithms and software packages that allow fitting certain classes of models and data. That is not to say that some experimental and modeling work should not occur, only that more theoretical development, technical expertise, and/or model-fitting algorithm development is needed to answer some questions. Note that the idea of choosing experiments based on current states of theoretical knowledge and algorithm availability may be somewhat distasteful to researchers who feel that theory and algorithm development should occur to explain any collected data. However each of these two intermediate steps likely require extensive research (e.g. see Guest and Martin, 2021, for a discussion on theory development).

Discrete choice and response time data resulting from two-alternative forced choice (2AFC) experimental tasks, or similar tasks, are known to be easily studied using signal detection theory and sequential sampling models. However work on developing models of more complex decisions, such as choices in continuous space, is still somewhat new (Smith, 2016; Ratcliff, 2018; Kvam et al., 2023). And while there is a large body of work on developing models for describing choices and response times for more than two alternatives (Busemeyer et al., 2019; van Ravenzwaaij et al., 2020; Thomas et al., 2021; Krajbich and Rangel, 2011; Hawkins and Heathcote, 2021; Heathcote and Matzke, 2022), models of multi-alternative forced choice (MAFC) tasks have not been as widely utilized as models of 2AFC tasks in studies testing the influence of experimental manipulations on parameters (e.g. Dutilh et al., 2019) nor in studies of stable measurements of individual differences (e.g. Schubert et al., 2016; Lerche et al., 2020), with some notable exceptions (e.g. Rouder et al., 2015). If a researcher only cares about a scientific question that does not depend on the specific type of choice, then the researcher might choose the 2AFC task because of the current state of knowledge. Another

concern is that researchers may find model fitting limited to existing packages which lack procedures for fitting MAFC and continuous choice tasks. However in the future it is likely that MAFC model fitting procedures will be further rigorously tested and more widely applied, with new packages being released (e.g. Stevenson et al., 2023; Villarreal et al., 2023). We also expect that model fitting procedures will become more flexible in their implementation (e.g. see Radev et al., 2020). Furthermore, if the goal is to develop new methods and/or to test something specific about multiple alternatives or choices on a continuous scale, then a MAFC or continuous choice experimental task would be best.

Some M/EEG signals will be more easily found (e.g. differentiated from other signals) in certain experimental conditions due to prior knowledge from a wealth of literature about these signals. For this reason the same experimental tasks are used often in M/EEG research and electrophysiology across experiments, such as the Random Dot Motion (RDM) task (e.g. see Newsome and Pare, 1988; Gherman and Philiastides, 2018). For instance the Centro-Parietal Positivity (CPP) is known to occur in tasks where a visual or auditory stimulus ramps up or down in signal intensity, including RDM tasks (O'Connell et al., 2012; Kelly and O'Connell, 2013; Rangelov and Mattingley, 2020). Combining all these experimental considerations, the best visual stimulus to test a simple hypothesis of the relationship of CPP to an evidence accumulation rate during decision-making might be a RDM where the participant must differentiate either leftward or rightward motion during each trial (i.e. a 2AFC task). Note that researchers should not be limited to prior research, and researchers should feel free to design new experimental tasks and new experimental designs. Diversity of experimental design ideas will always remain important for the growth of the field of joint modeling and the growth of knowledge.

4. Collection and preprocessing of M/EEG for joint modeling

4.1. Software for processing of M/EEG data

The software for both preprocessing and analysis of M/EEG data are either stand alone programs, or based in either the MATLAB (The MathWorks Inc., 2022) or Python programming languages. This is due to a (likely self-reinforcing) preference of Cognitive Neuroscientists generally for MATLAB and Python over R. EEG analysis packages exist for R, but are being still developed and not as widely used, such as eegUtils (Craddock, 2023). We recommend using established toolboxes, such as MNE (Gramfort et al., 2013) in Python, or FieldTrip (Oostenveld et al., 2011) or EEGLAB (Delorme and Makeig, 2004) in MATLAB. Tutorials and many walk-through example analyses are readily available online for all 3 toolboxes. In this section we introduce readers to general principles about EEG analysis. We encourage readers to follow the *Further readings* section and consult specific online toolbox manuals if beginning EEG analysis for the first time.

4.2. Understanding artifactual processes in M/EEG data

M/EEG contains many overlapping sources of information, including brain-generated sources and artifactual sources (Nunez and Srinivasan, 2006; Nunez et al., 2016; Fitzgibbon et al., 2016). Due to the complexity and the size of information in collected M/EEG, parameter estimation of joint models of behavior and M/EEG usually requires extraction of specific M/EEG signals. One important step in this process is the removal of artifact, specifically

16

muscle-generated electromyographic (EMG) signals, environmental electrical artifacts, and physical movement artifacts that are prevalent in scalp-recorded EEG recordings (Whitham et al., 2007; Nunez et al., 2016). Unless those artifacts are removed, or specifically accounted for in a joint model, these sources of artifact will likely add noise that affects results of joint modeling. These artifacts could thus greatly influence the results of neurocognitive models in parameter estimates or the results of model comparisons (e.g. see Hawkins et al., 2017).

Electrical and movement artifacts in scalp-recorded EEG records can be reduced by proper recording practices. Usually the goal of these practices is to (1) keep consistent electrical contact of the ground and reference electrodes to the scalp, (2) keep consistent electrical contact of all other electrodes (or to remove those electrodes from later analysis, if some electrodes can be removed without much loss of information, such as in high-density EEG), and (3) by removing or shielding external sources of electrical artifact from the area of recording. Electrodes with higher impedances typically have larger amplitude environmental noise (with large noise amplitudes at 60 Hz or 50 Hz depending upon the place in the world in which the recording takes place) that can be removed with online or offline targeted filtering (Kappenman and Luck, 2010). But also large impedances can be indicative of an electrode that is not making consistent electrical contact with the scalp and could produce high amplitude movement artifact. Inconsistent contact of an electrode could lead to sudden changes in impedance that can affect EEG records of that electrode. Most commercially available EEG systems have easy-to-implement methods of measuring electrode impedance. Those electrodes that have very high impedances can be given special attention (using system recommended cutoffs depending upon the EEG cap and amplifier), such as making sure that the electrode is making stable contact with the skin of the scalp through hair. Unless using "dry" EEG systems, an appropriate amount of conductive gel or saline is also important to maintain electrical contact. However, it's also important not to use too much conductive solution to avoid electrical bridging (Greischar et al., 2004). We recommend reading some of the existing recommendations from various EEG laboratories resources for discussions of best practices for EEG data collection (e.g. Farrens et al., 2020; Boudewyn et al., 2023), as well as recent work on improving recording practices for different hair types (e.g. Etienne et al., 2020).

In addition to proper recording practices, additional artifact correction is almost always performed on EEG data, especially for eye blinks, EMG, cardiovascular-generated electro-cardiographic (EKG) signals, and miscellaneous movement artifacts. For joint modeling this artifact correction will usually be performed offline, as opposed to BCI applications where artifact correction is performed online. Artifact correction is also helpful for magnetoen-cephalographic (MEG) and intracranial electroencephalographic (iEEG) records, although the prevalence of different types of artifacts will differ across modalities and recording systems. Techniques to mitigate artifacts before parameter estimation is especially necessary due to the simple assumptions often made intrinsically in many joint models of M/EEG and behavior. Because brain generated EEG recorded from the scalp will not manifest as sudden large amplitude spikes after filtering by the skull and skin, common offline artifact correction techniques include removing epochs of data or specific electrode records that surpass a particular amplitude cutoff are useful and basic artifact corrections.

Another popular method which deserves special consideration is Independent Component Analysis (ICA Makeig et al., 1996) for scalp-recorded EEG and MEG. ICA finds linear
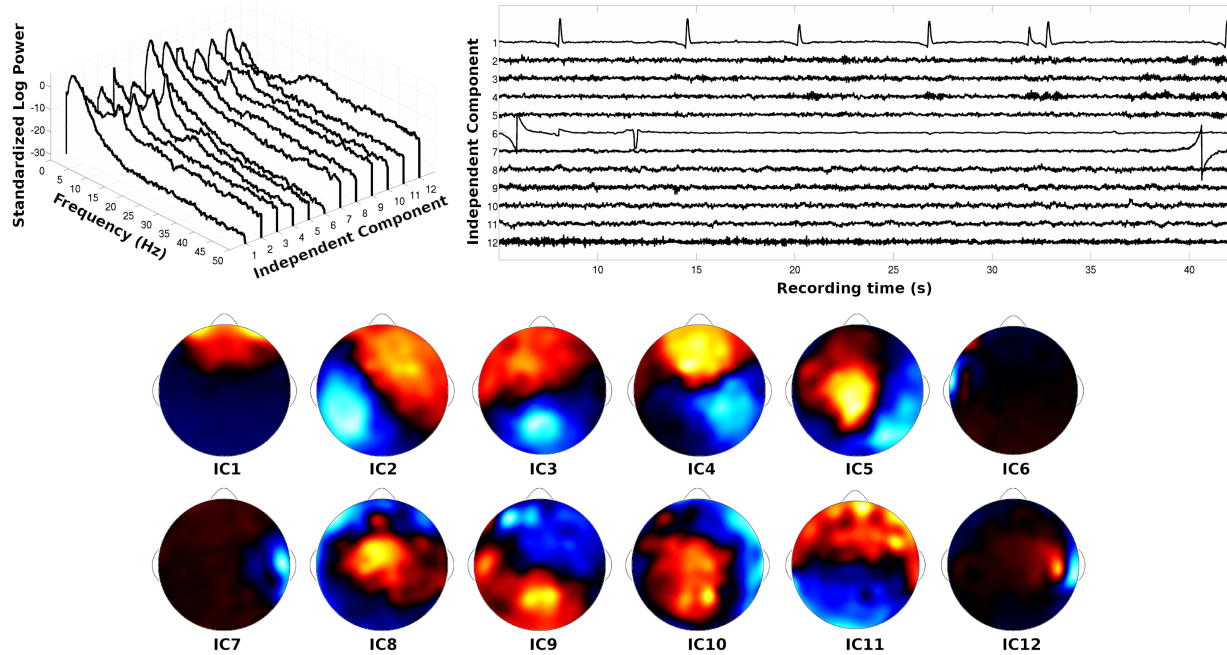
Figure 3: (From top-left, clockwise) Power spectra, time courses, and spline-interpolated channel weights from the first twelve Independent Components (ICs). The Independent Component Analysis (ICA) algorithm was performed on an EEG record in which a participant was fixating on a computer monitor. ICs that are likely to reflect artifact can be subtracted from the EEG data before neurocognitive modeling. IC1 is indicative of an eye blink. IC6 and IC7 are indicative of temporary changes in channel impedances. IC12 is indicative of some muscle artifact. Note this figure is adapted with permission from Figure 4 by Nunez et al. (2016)

.

mixtures across M/EEG channels that are statistically independent as possible by finding maximal non-Gaussian mixtures. The resulting components have M/EEG-like times courses where, if the time dependency were ignored and the time samples were randomly shuffled, the resulting distributions would have minimum mutual information or be separated on the basis of kurtosis (Hyvärinen and Oja, 1997; Jung et al., 2000). In practice, these methods often yield non-normal mixtures that have distributions with outliers, and thus ICA algorithms are especially good at extracting certain artifacts within multiple EEG channels such as eye blinks (IC1 in **Figure 3**), eye movements, EKG, and temporary changes in electrode impedances (IC6 and IC7 in **Figure 3**). Sometimes ICA algorithms can also find some EMG artifacts that can be easily identified in the EEG data (IC12 in **Figure 3**). Often ICs are identified manually through visual inspection. However recent algorithms have sought to remove some of the subjectivity in IC identification, with specific algorithms (e.g. see Mognon et al., 2011) or with Artificial Neural Networks trained on may expert evaluations (Pion-Tonachini et al., 2019; Li et al., 2022) – see the first *Exercise* for additional help on this topic. These components can then be extracted from multiple electrodes, and the resulting EEG data can be converted back to channel space or be kept in component space for joint modeling. EEG components (weighted mixtures of electrodes) are discussed in detail below.

Note that it is unlikely that all artifact will be removed using these methods, especially EMG artifact in scalp-recorded EEG (Whitham et al., 2007; Nunez et al., 2016). So to further

avoid artifact in scalp-EEG, it is best to choose specific methods of signal extraction. This is typically performed outside of joint modeling, but future researchers may be able to model these methods explicitly to retain sources of noise in the model. These methods include popular EEG-band limited analyses (e.g. calculating 8 to 13 Hz alpha power over posterior electrodes in Ghaderi-Kangavari et al., 2023a) and event-related potential (ERP) analyses that mitigate artifactual components through averaging across trials (e.g. N200 latencies in Nunez et al., 2019a).

Finally, M/EEG outliers could drive the entire modeling results. A choice must be made between explicit removal and explicit modeling of lapse and artifactual processes. Note that there will be differences between artifactual (non-brain generated M/EEG) and brain generated M/EEG that is related to *some* cognitive process but not related to the cognitive process of interest. A true neurocognitive reason for an outlier could be, for instance, because the participant is mind wandering during that trial (e.g. see Hawkins et al., 2022). It is a question for new joint modeling work whether inclusion of non-brain generated M/EEG artifactual processes and/or mixtures of cognitive processes within the joint modeling itself would be beneficial to understand the researchers' specific hypotheses and questions.

4.3. Extraction of relevant M/EEG signals

In previous research, most researchers have extracted specific EEG signals before joint modeling (e.g. Frank et al., 2015; Nunez et al., 2017, 2019a). Targeted extraction of these specific EEG signals can be especially beneficial for joint modeling, particularly when those signals have a rich literature of prior research and cognitive theory. We will concentrate here on popular EEG signals studied within Cognitive Neuroscience.

Event Related Potentials (ERPs) are defined as averages of M/EEG across experimental trials, time-locked to specific events such as the onset of a visual stimulus (a Visual Evoked potential; VEP) or the execution of a response such as a button press (a Motor Evoked Potential; MEP). ERPs can be computed by simply averaging the M/EEG signals v over N trials i such that resulting signal μ varies over a time index t, time-locked to an experimental event:

$$\mu_t = \frac{1}{N} \sum_{i=1}^{N} v_{ti} \tag{9}$$

ERPs also have rich literature (Luck, 2012, 2022), from which best practices can be recommended. This literature can also be used to generate new confirmatory and exploratory hypotheses that could be answered with joint modeling methods. After calculation of ERPs using the equation above, specific peak latencies, amplitudes, or deviation times from baseline are typically extracted, either positive or negative peaks. Common ERPs are the negative N200 peak approximately 200 ms after the onset of a visual stimulus in occipital and parietal electrodes (sometimes labeled N1 for the first negative peak), recently thought to encode the onset of evidence accumulation during decision-making (Nunez et al., 2019a). Another common ERP is the positive P300 peak at least 300 ms after the onset of a visual stimulus; this ERP is also called the Cento-parietal Positivity (CPP) during specific decision-making tasks (Twomey et al., 2015), discussed extensively earlier and used in example **Models 2** and

3. Common MEPs are the Readiness Potential (RP) and the related Lateralized Readiness Potential (LRP) (Gluth et al., 2013; Lui et al., 2021, e.g.). ERPs can also be estimated on single-trials (e.g. Nunez et al., 2017; Bridwell et al., 2018; Nunez et al., 2019a).

Frequency or time-frequency decompositions, such as Fourier and wavelet analyses, are also common methods used to extract specific signals in M/EEG (Cohen, 2014). These decompositions form the basis of derived measures, such as M/EEG coherence, and typically rely on signal to noise ratios. Frequency and time-frequency measures have been used in neurocognitive modeling of decision-making (Frank et al., 2015; Polanía et al., 2014) and can be event-locked or stem from endogenous rhythms not related to the timing of the task. Many algorithms in high-level programming languages such as the Fast-Fourier transform are sufficient to estimate these signals, although best practices in EEG and standard EEG conventions should be known (Nunez et al., 2016). Alternatively, there are other algorithms developed specifically to extract specific band-limited waveforms, such as finding $80 - 250$ Hz High Frequency Oscillations (HFOs) that last only for a few milliseconds in iEEG data (Charupanit and Lopour, 2017; Nunez et al., 2022). One warning for scalp EEG is that high frequencies (approx. > 20 Hz) typically contain more EMG artifact (Whitham et al., 2007), so care must be taken when interpreting the results of measures derived from high frequencies embedded in joint models. Although MEG systems may be more robust to EMG artifact (Claus et al., 2012; Muthukumaraswamy, 2013), and newer MEG systems could be even more resistant to EMG artifact (Ilmoniemi and Sarvas, 2019). Another caution is that researchers have shown heterogeneity in the power bands across and within participants (Nunez et al., 2001), as well as heterogeneity in the waveforms themselves (Donoghue et al., 2022). Therefore, care must be taken when extracting specific signals for joint modeling.

Steady-state evoked potentials (SSEPs), and in particular steady-state visual evoked potentials (SSVEPs) and related steady state auditory potentials (SSAEPs), are a special case of band-limited analysis where the frequency band of interest in the M/EEG results from a processing stimuli at a certain presentation rates or "flicker" (Regan, 1977). For instance a Gabor patch flickering at 15 Hz will result in a large, narrow-band, 15 Hz response (and often harmonics of 15 Hz) in EEG. This is the result of the cortex receiving and processing signals at this rate, which is expected of a linear system. Some researchers have found evidence that endogenous EEG signals may also "entrain" to the stimulus frequencies (Srinivasan, 2004; Ding et al., 2006). SSEP analyses could be particularly useful for fitting joint models because the amplitude or phase-locking across trials is thought to index within individual and individual differences in attention (Ding et al., 2006). We have previously explored how individual differences in attention as measured by SSVEPs affected cognitive components of decision-making (Nunez et al., 2015).

Typically ERPs, SSEPs, and power in different (time-)frequency bands, are observed in replicable scalp and brain locations. These specific EEG signals often have consistent cognitive interpretations found in the Cognitive Neuroscience literature, e.g. see **Table 1**. However, the exact electrode/sensor locations will differ from participant to participant, and could even change within a participant due to varying artifactual sources and electrical contact of the electrodes over the course of a long experiment. Furthermore, within the field of model-based cognitive neuroscience modeling, there is a need to better utilize overlapping information in M/EEG data to improve descriptions of cognitive theory (see Borst and Anderson, 2015; Bridwell et al., 2018; Weindel et al., 2023). For intracranial EEG (iEEG)

spatial filters can also be useful in extracting relevant EEG features for joint modeling (e.g. see Schaworonkow and Voytek, 2021). Therefore, weighted averages across channels should also be considered. We will refer to these set of analyses as "component analyses" for finding mixtures, typically linear mixtures, of M/EEG data that may better reflect the underlying source components present on the scalp or in intracranial electrode space (Parra et al., 2005). Component analyses will extract a weighted average of electrodes/sensors to improve the signal-to-noise ratio of mixtures. Previously mentioned Independent Component Analysis (ICA) is one example algorithm, but other methods like Principal Component Analysis (PCA) (e.g. Nunez et al., 2017, 2019a), Canonical Correlation Analysis (CCA) (e.g. van Vugt et al., 2012), and explicitly modeling mixtures over electrodes/sensors in joint models can also be considered.

Preprocessing of M/EEG may be theoretically undesirable since extracting specific signals often involves removing potentially useful information from the M/EEG signals, which could be better accounted for with statistical models in joint modeling. In all our previously published work (Nunez et al., 2015, 2017, 2019a; Lui et al., 2021), we extracted specific EEG potentials before joint modeling. While these methods are useful for testing specific theories of visual attention, motor processing, or visual encoding, they are not as suitable for understanding parallel processes that occur during decision-making. The future of joint modeling techniques should better account for full M/EEG data to improve prediction and hypothesis testing. This can be achieved by embedding mixture models of M/EEG signals, either based on brain connectivity and neural network behavior (with a model of electric volume conduction to the scalp, see paper by Nunez et al. (2019b)) or through more non-parametric methods based on mixtures of oscillating signals.

5. Implementing model fitting procedures and estimating parameters

Finding parameter estimates from a proposed model can be difficult. There are many more restrictions on parameter fitting than model simulation (due to difficulty in maximizing *likelihood* spaces or sampling from *posterior distributions*). Many joint models will also not be *identifiable*, as discussed below. However multiple free programs exist to help you fit joint models of M/EEG and behavior. Most require some knowledge of a programming language and sampling methods.

5.1. Terms in model fitting

One term that is often used in model *fitting* is that of the *likelihood*. Each distributional statement represented with the $\sim$ symbol in this paper has an associated equation that describes the probability of taking certain data values when calculating the *integral* of the equation over a range of possible values. Thus the integral of the function over all possible parameters is always 1, indicating that the data being somewhere in the entire possible range has probability = 1. This equation, the *probability density function* (pdf) is called the *likelihood function L* whenever it is considered to vary over parameters for observed data. The integral of this function L now no longer represents probability (and the function has a different shape), because the function varies over a range of parameters and was not defined to represent probability in this way.

For instance, if we want to estimate the standard deviation σ of raw EEG data in one electrode e over a certain period of time. We could assume that e is distributed normally with mean 0 and an unknown standard deviation σ. This is given by the distributional statement $e_s \sim Normal(0, \ \sigma^2)$ for each sample s. If we have 1024 samples of EEG e_s, say 1 second of recording with an amplifier that has a sample rate of 1024 Hz, then we could find the *maximum likelihood estimator* by placing those exact data values $\mathbf{e}$ (bold indicating a vector of data samples e_s) in a joint pdf, given by individual normal pdfs for each EEG sample multiplied together:

$$f(\sigma, \mathbf{e}) = \prod_{s=1}^{1024} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{e_s^2}{2\sigma^2}\right) \equiv L(\sigma|\mathbf{e}) \tag{10}$$

Note that even though the likelihood function L for σ is derived from and has the same mathematical equation, the shape of this function is not a bell curve over certain values of σ for now known data values $\mathbf{e}$ because it is now a function of parameters and not data. With this likelihood function L we can find a value for σ that describes the maximum likelihood by finding the value of σ at the peak L value (e.g. by differentiating the equation and setting it equal to zero). This results in a good *estimate* of the true, never observed, EEG standard deviation. Note that, confusingly, this *estimate* is often called the "standard deviation of the data" in conversation. The likelihood function is also used in Bayesian inference, where the shape of the *posterior distribution* over possible parameter values is given by the shape of the likelihood function L multiplied by the shape of the *prior* distribution.

Most programs to fit joint models have predefined likelihood functions, and only under special conditions would you need to write your own likelihood function. This is true in all *Probabilistic Programming Languages* (see below) as well as most other existing programs written to fit models. Readers who want additional information, are confused by these concepts, or who want to define their own likelihood functions should consult chapters 4 and 6 of the textbook by Farrell and Lewandowsky (2018) and see additional *Further readings* for this section – found at the end of this paper.

5.2. Avoiding model complexity

To test specific hypotheses or compare theories, a perfect explanation of all relevant M/EEG signals and behavioral data is often unnecessary and could result in overfitting the model (see Navarro, 2019). The degree of complexity needed in the fitted model(s) will depend on the goals of the researcher. In DDMs, choosing between two discrete choices is assumed to occur due to particular time-varying sampling of relative evidence. The rate of sampling of evidence is assumed to change both during a single choice due to within-trial variability in a random walk process, but also change across many similar choices, due to trial-to-trial variability (Ratcliff et al., 2016). In an experiment with the participant making a choice during every experimental trial, expected across-trial changes in the parameters are often modeled with across-trial variability parameters. While it is expected that humans do have variability in strategy, attention, and response cautiousness, etc. across trials, across-trial variability parameters of a full DDM model cannot be easily estimated with behavior alone (Boehm et al., 2018). This problem of parameter estimation can be at least slightly improved with the addition of EEG signals on single trials (Nunez et al., 2017; Hawkins

et al., 2017). Therefore when fitting behavioral data to joint models, we often make the choice to fit simple DDM without any across-trial variability in the DDM parameters that is not described by the single-trial EEG measures (Nunez et al., 2017, 2019a). Other joint modeling research has often included across-trial variability in evidence accumulation rates but no other parameters (e.g. Frank et al., 2015; Ghaderi-Kangavari et al., 2022). In past work we have purposely not included some across-trial variability parameters because we have shown in simulations that the question of interest about the M/EEG-cognitive relationship can be answered without more complex models, knowing that the greater mathematical theory of decision-making does have variability and that the data would better be described by more complicated models. However if we were differentiating between models that are very similar in their predictions, we might want to include trial-to-trial variability parameters or more precise M/EEG correlates of evidence accumulation.

5.3. Parameter recovery of simulated models

Simulation is especially important for newer models and model fitting procedures that are not widely used. Because jointly fitting neural data and human behavior is not widely used within Cognitive Neuroscience and Psychology, the majority of model fitting procedures that readers may implement will fall into this category. Thus researchers interested in neurocognitive models should always simulate and refit new models to understand parameter recovery. The parameter estimates from new models and new model fitting procedures should always be verified by refitting data from simulations before modeling results can be trusted to be self-consistent, whether or not the model is reflective of reality or a specific hypothesis. For instance, model-fitting procedures could give realistic results, but fail to recover the same parameters when simulated. This means that the parameter estimates recovered from model-fitting will not reflect reality, even if the simplified model is completely true. In addition, some parameters of a model may recover, and therefore be relevant to analyze in real data, while other parameters of a model will not recover. In **Figure 4** we show recovery of the 5 of 6 parameters from **Model 3** when using Markov Chain Monte Carlo (MCMC) sampling in JAGS (discussed below) with the jags-wiener plugin (Wabersich and Vandekerckhove, 2013). We fit a model that assumes trial-to-trial variability in CPP slopes and not drift-rate (e.g., assuming $\eta = 0$). The code for this simulation, parameter recovery, and plot is available at `https://github.com/mdnunez/pyhddmjags/blob/master/simpleCPP_test.py` as of March 2022, along with a shortened annotated version in the *Appendix*.

5.4. Comparing models

Model comparison is useful when multiple theories could describe the data and the evaluation of hypotheses would depend on the theory assumed. Model comparison can also be used to evaluate competing theories directly. Therefore model comparison in joint cognitive modeling of M/EEG and human behavior is almost always beneficial. Model comparison can be based on multiple dimensions, but typically researchers are interested in the models that provide the most predictive and/or realistic accounts of the neural and behavior data. Model comparison could therefore be based on how well the model predicts data or how well models evaluate the hypothesis versus an alternative hypothesis. For example, if we were interested in the hypothesis that the CPP reflects evidence accumulation *exactly*, we could compare a model where the slope of the CPP describes the evidence accumulation rate itself, e.g.
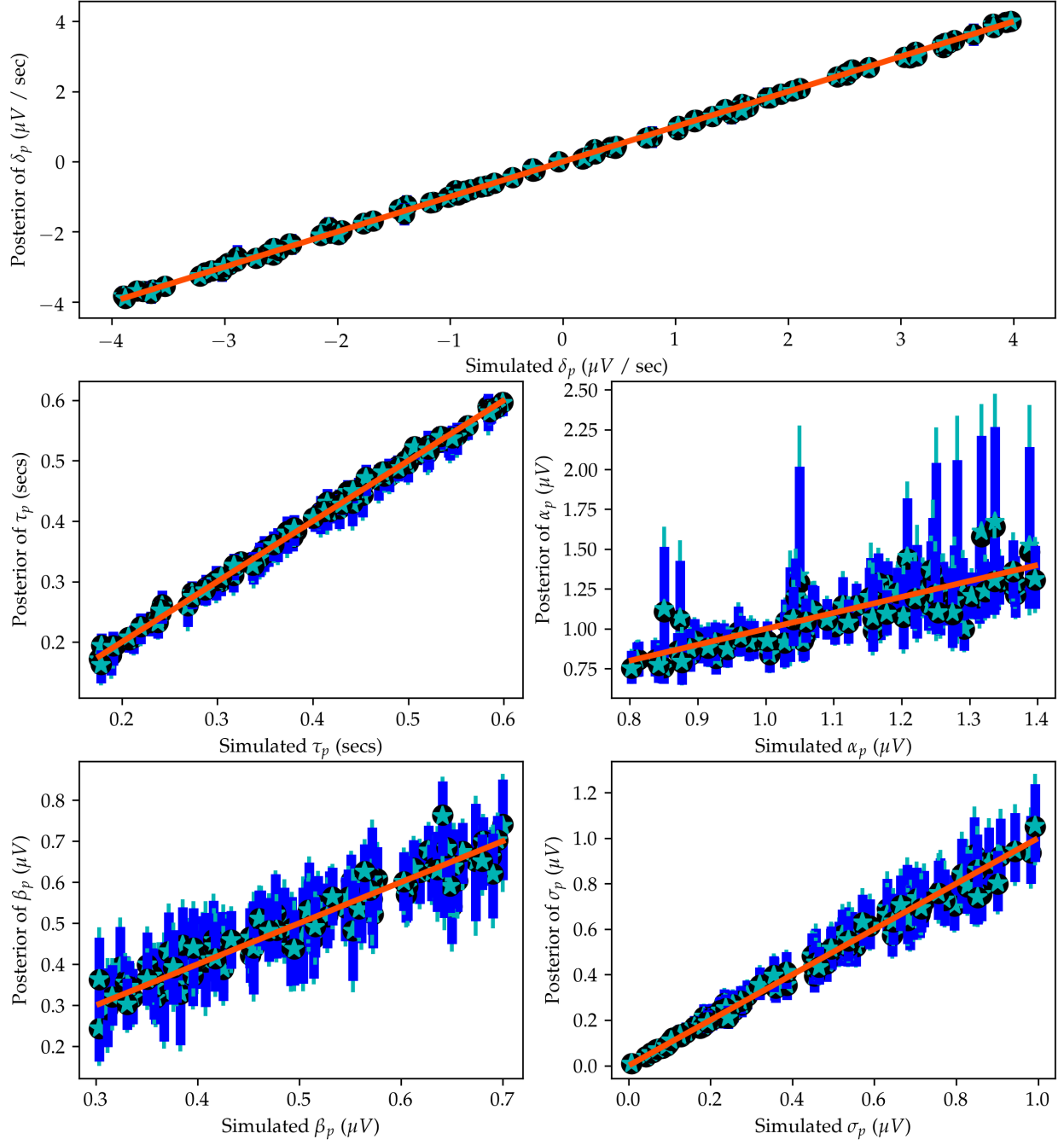
Figure 4: The recovery of 5 parameters from **Model 3**. We simulated **Model 3** using the provided Python code with 100 simulated participants and 100 trials for each participant. The x-axis of each plot is the true simulated parameter and the y-axis is a summary of posterior samples. The mean of the posterior distributions are given by teal star symbols and the median of the posterior distributions are given by black circles. Uncertainty about each parameter estimate is given by the 95% credible intervals of dark blue lines, and 99% credible intervals of teal lines.

Model 3, to a more general model where the relationship between the CPP and evidence accumulation rate can be any value (see **Exercises**). If **Model 3** describes the data nearly as well as the more generalized model, then we have evidence for the hypothesis that the CPP reflects evidence accumulation exactly.

How well a model *predicts* data used to fit the model itself is often used as a measure of performance (Blohm et al., 2020). This is called *in-sample* prediction. However evaluating models based only on in-sample prediction can result in *overfitting* the data. Overfitting describes the situation in which additional in-sample prediction is gained through model complexity that is unrelated to the underlying true generative process, resulting in worse *out-of-sample* prediction. Out-of-sample prediction refers to how well a model predicts data that it was not fit to. We refer readers to a discussion about this topic by Aki Vehtari at `https://avehtari.github.io/modelselection/CV-FAQ.html` (Vehtari, 2023). Out-of-sample could also refer, although this is not traditionally in the definition, to how well it generalizes to similar data from other experiments (see Busemeyer and Wang, 2000). Note that prediction of in-sample and out-of-sample data can be used to compare models that differentiate specific hypotheses, but it is often not necessarily to perfectly describe or predict EEG and/or behavioral data due to the presence of artifacts and noise in EEG and contaminate behavioral data not related to the cognition of interest.

For evaluating neurocognitive models of M/EEG and human behavior, we prefer out-of-sample prediction (e.g. Nunez et al., 2017; Schubert et al., 2019). Out-of-sample prediction typically involves taking at least one subset of the data out before fitting the model to the remaining "in-sample" data. For instance, one could split a data set where 80% of the data is used to fit the model and 20% of the data is used to evaluate out-of-sample prediction. One method to evaluate the similarity of predicted data to the actual data is with a *proportion of variance explained* calculation. For instance, we have previously calculated R^2_{pred} of participants' accuracy and correct response time 25th percentiles, medians, and 75th percentiles (Nunez et al., 2015, 2017). R^2_{pred} is a measure of percentage variance in a statistic T (e.g. accuracy, correct-RT median, etc.) explained by in-sample or out-of-sample prediction. It is a function of the mean squared error of prediction (MSEP) and the sample variance of the statistic T based on a sample size J of data observations. R^2_{pred} is defined as:

$$R^2_{\text{pred}} = 1 - \frac{\sum_{j=1}^{J}(T_j - T_{(\text{pred})j})^2/(J-1)}{\sum_{j=1}^{J}(T_j - \bar{T})^2/(J-1)} = 1 - \frac{\text{MSEP}_T}{\widehat{\text{Var}[T]}} \tag{11}$$

As one example, we could compare the out-of-sample prediction of **Model 2A** to **Model 2B** that was fit to 80% of the data from **Experiment 1**. The observations for each statistic T in the R^2_{pred} equation would be for every participant j with sample size J being the number of participants.

There are other methods to evaluate out-of-sample prediction, such as calculating the log-likelihood under predicted data (e.g. see Figure 9 of Turner et al., 2016). Models' prediction ability can also be evaluated with plots such as quantile-quantile (Q-Q) plots of measured versus predicted data quantiles. Because we typically use Bayesian methods, we generate posterior predictive distributions for in-sample and out-of-sample data. We can then create posterior predictive coverage plots (e.g. see Supplementary Materials of Nunez et al., 2017) or Q-Q plots. Plotting often provides additional information to more quantifiable measures

such as R^2_{pred} or a similar measure.

Cross-validation refers to methods where out-of-sample prediction is performed repetitively on different subsets of data (e.g. a new 20% of the same data set iteratively). Because cross-validation lowers the impact of outliers in the out-sample, cross validation can be useful for modeling M/EEG data because the presence of outliers is commonplace. However care must be taken to avoid changing parameters of the model or model fitting procedure based on the results of cross validation because this would make the cross validation process less reflective of the true predictive ability of the model.

If out-of-sample prediction is not available, then often penalizing by model complexity after in-sample prediction is used. This is often why Information Criteria measures are used (e.g. Ghaderi-Kangavari et al., 2023a, 2022). Essentially these measures are in-sample prediction measures that penalize for model complexity. Akaike Information Criterion (AIC), Bayesian Information Criteria (BIC), Deviance Information Criteria (DIC), and re-weighted variations of these measures are often used. However these measures may yield different results. For instance, it is thought that BIC more often favors models that match the ground truth while AIC more often favors models that are predictive of new data (Aho et al., 2014; Chandrasekaran and Hawkins, 2019) Therefore, it is important to pick one ahead of time and stick to it, or preregister the modeling analysis (Lee et al., 2019; Vandekerckhove et al., 2019).

Simulation is also important if you wish to perform model comparison. Some models may fit data better not because the underlying theory is a better reflection of reality, but because the models capture some contaminant process better. That is, neither model is correct but the worse-fitting model is a better description of reality. Simulations of multiple models with contaminant processes before performing parameter recovery for each model can thus reveal which model and model-fitting procedure best recover true parameters. Simulation is also important for comparing models that predict both EEG and behavioral data, such as Model 3, to other such models or models that only predict one data type. It is not immediately clear whether M/EEG or behavior should be favored when evaluating in-sample and out-of-sample prediction to compare models, and predictions of this data could change with each new model. Therefore simulations should be performed to make sure the parameters of interest are recovered when comparing models, as well as understand how model changes affect predictions of multiple data types. For an example of joint modeling for EEG and behavior, see work by Ghaderi-Kangavari et al. (2023b).

We separate here confirmatory research from exploratory research. When discovering the influence of EEG measures on parameters that describe decision-making behavior, it is beneficial to explore various model types that may better match the theoretical evidence accumulation process. This can be achieved, for instance, either by fitting parameters from different sequential sampling models (SSMs) directly (preferred) or simulating a variety of SSMs and then exploring how similar parameters can be recovered in other models. Confirmatory research, on the other hand, necessitates large samples, pre-deciding an analysis plan that includes the specific joint model to test (and perhaps preregistration of that analysis plan and model), and requiring strict standards for hypothesis acceptance, such as in clinical trials (Lee et al., 2019). However, it's worth nothing that there exists a spectrum between exploratory and confirmatory research that has been discussed elsewhere (e.g. Devezer et al., 2021).

5.5. Fitting complex models using Bayesian methods

We generally prefer to use programs that use Bayesian Markov Chain Monte Carlo (MCMC) sampling and allow a large amount of flexibility to change the model structure. Probabilistic Programming Languages such as JAGS (Plummer, 2003), Stan (Carpenter et al., 2017), and PyMC3/4 (Salvatier et al., 2016) all make this process incredibly easy by allowing you to write your own complex models, but without needing to write your own samplers. Popular wrapper programs have also been created around these programs that make fitting certain models even easier (Wiecki et al., 2013; Bürkner, 2017), and there are other easy-to-use programs that implement their own samplers (Heathcote et al., 2019; Stevenson et al., 2023). Note that Bayesian analysis can be quite easy to learn for those that have a background in some mathematics and statistics. A nice introduction to Bayesian analysis is given by Etz and Vandekerckhove (2018). See also books by McElreath (2020) and Gelman et al. (2014).

All the aforementioned programs also allow the modeler to easily implement hierarchical parameters which can better account for variance across experimental conditions, participants, sessions, etc. Hierarchical parameters can often better account for data with multiple modes of data (Lee, 2011; Turner et al., 2016), such as EEG and human choice response times. Hierarchical models often yield better estimates of parameters due to "shrinkage" towards the mean parameters rather than fitting a model per participant or experimental condition, which could lead to overfitting and misestimation (see Chapter 5 of Gelman et al., 2014). Examples for fitting behavioural DDMs and neurocognitive DDMs using Python, JAGS, and Stan, with the models themselves written in JAGS and Stan code, are given in the repository `https://github.com/mdnunez/pyhddmjags`. We encourage readers to run the example models in this repository if they are interested in using JAGS and Stan with Python. Note that connectors to JAGS and Stan also exist in R (for examples, see `https://github.com/kiante-fernandez/Rhddmjags`) and other programming languages.

The programmatic implementation to generate parameter estimates from joint models that we preferred in the past is JAGS (Plummer, 2003). JAGS is now a somewhat older program, but nicely contains multiple MCMC samplers and chooses among them based on the user-defined model. Custom distributions can also be added to JAGS (Wabersich and Vandekerckhove, 2014; Villarreal et al., 2023). JAGS uses Bayesian MCMC samplers to fit models to data and can easily fit joint models to multiple data types. For instance, we fit a simplified version of **Model 3** in JAGS with 12,000 original samples in each of six chains for each parameter (see `https://github.com/mdnunez/pyhddmjags/blob/master/simpleCPP_test.py`). After removing the first $2,000$ samples as a "warm-up" or "burn-in" and then keeping only every 10th sample, i.e. using a "thinning" parameter of 10, this results in $1,000$ posterior samples in each chain for $1,000 * 6 = 6,000$ samples from the estimated posterior distributions for each parameter.

To assess whether the model is reaching a unique solution (i.e. unique joint posterior distributions), we can both inspect our MCMC chains but also gauge certain model convergence diagnostics (Gelman et al., 2014). The Gelman-Rubin statistic and the number of effective samples are calculated (Gelman et al., 2014). The Gelman-Rubin statistic assesses the convergence of MCMC samplers by comparing the between-chain variance to the within-chain variance of each parameter, with Gelman-Rubin statistics ≤ 1.1 thought to be a necessity for convergence. We also implemented the recommendation by Gelman et al. (2014) (see

footnote in the 3rd Edition on page 283) to split the chains in half before calculating the Gelman-Rubin statistic in order to account for non-stationary chains. The "effective number of samples" equation scales the total sample number for each parameter posterior by auto-correlation in the chains in order to estimate an independent number of samples. Larger effective samples for each parameter in the model are better. The chains for parameters with the largest Gelman-Rubin statistics and smallest effective number of samples are also visually inspected to ensure convergence. In publications we typically report the maximum Gelman-Rubin statistics across all parameters, and we have recently started to report the minimum number of effective samples across all parameters.

5.6. Prior distributions in Bayesian models

When using Bayesian methods we often must choose prior distributions of parameters, this is also true for estimating joint models of M/EEG and behavior with Bayesian methods. When possible, we pick prior distributions based on previous publications, and such that the prior distributions have weight over plausible values of the parameters. For instance, random draws from a normal distribution with a mean of .5 and a standard deviation of .25 will result in 68.2% of those draws within .25 and .75, and 95.4% of those draws within 0 and 1. $[0, 1]$ is the domain of the relative start point parameter β that encodes initial evidence bias in a DDM. Therefore a normal distribution with mean .5 and standard deviation of .25 truncated to the domain $[0, 1]$ would be a good prior distribution for this parameter, disregarding algorithmic reasons why we might pick different priors (such as in true Gibbs sampling).

There is ongoing research about what the best prior distributions are, depending upon the type of sampler. There are also many philosophical discussions about whether to use "informative" (generally narrow) or "weakly informative" (generally wide) priors. As modelers we should experiment with different priors in simulation to see how and if they change the results significantly. However we have not found that different "weakly informative" priors changed results much based on posterior distributions of hierarchical DDM parameters. Prior distributions can change posterior distributions if those priors are very narrow, such that values in that parameter's domain are near impossible ("informative" priors). For instance, a prior of $\beta \sim Normal(.5, .01^2)$ would restrict the posterior distribution of β to be approximately in the domain $[.46, .54]$, within 4 standard deviations on both sides of the mean. We refer readers to a discussion about this topic by Andrew Gelman at https://github.com/stan-dev/stan/wiki/Prior-Choice-Recommendations (Gelman, 2020)

5.7. Assessing posterior distributions

Posterior distributions provide evidence for parameters given the data and specific model architecture. They are influenced by prior distributions, but are often much more influenced by the data itself. In Bayesian analysis, probability is defined as uncertainty. Therefore, we can inspect the posterior distributions themselves to calculate the probability of observing certain values of a parameter given the specific model and data. For instance, the posterior distributions that result from fitting **Model 2B** to data from **Experiment 2** would result in posterior distributions for all parameters, including both effect parameters ξ_{1j1} for experimental condition (1) and ξ_{1j2} for experimental condition (2). We could calculate the

probability that each effect parameter is greater than 0.5 (for instance) by finding the proportion of posterior samples that are above 0.5, thus approximating the area under the curves of the posterior distributions, and thus approximating the probability (e.g. evidence) that each effect parameter is greater than 0.5. We can also calculate posterior distributions to answer other questions using a transformation of our parameters. For instance, we can calculate the posterior distribution of the difference between these two effects by matching MCMC samples of the original model fit to get one difference posterior of the quantity $\xi_{1j1} - \xi_{1j2}$ for participant j. We can then calculate the probability that $\xi_{1j1} > \xi_{1j2}$ by finding the proportion of posterior samples of the new quantity $\xi_{1j1} - \xi_{1j2}$ that are above zero.

Bayes Factors (BFs) usually provide the degree of evidence (defined as a probability ratio in Bayesian statistics) for the data given a model. Here we will focus on calculating BFs for a specific case where the parameter value λ is exactly equal to some value x ($\lambda = x$), compared to the same model where the parameter λ can take any other realistic value ($\lambda \neq x$) (Jeffreys, 1961; Kass and Raftery, 1995; Rouder and Morey, 2012; van Ravenzwaaij and Etz, 2021). BFs can also be inverted to give evidence for the the more general case ($\lambda \neq x$) compared to the more specific case ($\lambda = x$). Generally Bayes Factors over 3 are considered positive evidence for the numerator model over the denominator model (e.g. the effect is 3 times more likely under the specific case than the general model) while over 20 is strong evidence (Kass and Raftery, 1995). Bayes Factors for parameters of joint models estimated with Bayesian methods can often be estimated using the Savage-Dickey density ratio of the posterior density of parameter λ at test value x over the prior density of parameter λ at test value x (Dickey and Lientz, 1970; Verdinelli and Wasserman, 1995; Wagenmakers et al., 2010; van Ravenzwaaij and Etz, 2021). One example of Savage-Dickey density ratio calculation for Hypothetical Experiment 2 is given at the bottom of `https://github.com/mdnunez/` `pyhddmjags/blob/master/model2b_experiment2.py`. We have also previously estimated specific Bayes Factors (BF1s) of linear relationships between non-decision time τ parameters and N200 ERP latencies that describe the amount of relative evidence of the effect parameter λ being equal to 1 ($\lambda = 1$) to a more general comparison model ($\lambda \neq 1$) using the Savage-Dickey density ratio (Nunez et al., 2019a). These BF1s compared the hypothesis of a "spike" distribution at 1 with no uncertainty in possible effect values ($\lambda = 1$) versus a model with less specific effect values ($\lambda \neq 1$). Note that Bayes Factors can describe comparisons of models generally (Etz and Vandekerckhove, 2018; van Doorn et al., 2021). However for joint modeling purposes we are often interested in comparing point hypotheses to general cases. Other BFs to compare complex models are currently difficult to calculate, and therefore we recommend model comparisons using the methods previously mentioned (e.g. R^2_{pred}) for cases other than comparing point hypotheses to general models. Integration of existing work into new packages and further development of these methods will likely make calculating BFs for comparing different models easier in the future (e.g. Gronau et al., 2017, 2020)

6. Discussion

6.1. Modeling of M/EEG generators

Note that in this paper we assumed summary measures of M/EEG. We could instead model and simulate M/EEG time-series, waveforms, or frequency bands for multiple electrodes/sensors. Considerable work has been conducted to understand the neural generators

of M/EEG (e.g. David and Friston, 2003; Nunez and Srinivasan, 2006; Srinivasan et al., 2013), especially in the field of Computational Neuroscience. To this end, many researchers in the field of Computational Neuroscience have built models of M/EEG, scaled up from single-unit neuron activity, the activity of populations of neurons, and the connectivity between neural populations (Daunizeau et al., 2011; Glomb et al., 2021). Developing and simulating these models have been helpful in generating theory to understand how the brain generates M/EEG, and there has been some success relating model parameters to human behavior.

However finding parameter estimates of these models often suffers from the *inverse problem*, known in statistical modeling as *unidentifiability* (Walter, 1987). The inverse problem is relevant across all scientific fields that involve modeling. Inverse problems / unidentifiabilities arise when there is more than one unique parameter set that can describe the data when the generator model is known (Bamber and van Santen, 2000). Therefore, we cannot *invert*, or *find unique parameter estimates*, for a model even though the model can be simulated and we have a good idea of the theoretical concepts. In the context of M/EEG computational modeling, the inverse problem stems from the fact that M/EEG *sources* are not typically confined to a two-dimensional representation of the brain. Instead, they are represented in a three-dimensional manner, while the scalp electrodes (EEG), MEG sensors, or intracranial electrodes (iEEG) exist on curved two-dimensional surfaces. Specifically, these two-dimensional surfaces usually correspond to the scalp, MEG helmet, or electrode strips respectively. Determining these *sources* at specific locations in the brain requires making additional assumptions about the M/EEG generators (e.g. prior information in Bayesian models, see Cai et al., 2018). Unfortunately, these assumptions have proven difficult to validate experimentally due to the invasiveness of surgical procedures, the nature of electrical volume conduction in the head, and the complexity of the brain. Consequently, many studies that explore solutions to the inverse problem for M/EEG rely on comparisons to across algorithms. However, this may change with the advent of new MEG technology (Ilmoniemi and Sarvas, 2019). As a result, some researchers opt to rely on finding only *representative* sources in M/EEG (Nunez et al., 2019b), including in our own work (e.g. Nunez et al., 2019a).

M/EEG records are an extremely rich data source. We usually assume many M/EEG sources in the brain, e.g. dipole sheet, that generate time series data of multiple electrodes, usually 16-256 electrodes in scalp EEG, approximately 300 sensors in MEG, and often > 50 electrodes in intracranial EEG, that are typically sampled at least at 250 Hz. This results in rich multivariate time series data with some spatial information. There are specific M/EEG waveforms, power bands, network interactions, etc. that exist within the data. But there is no reason to suspect that all useful measures within M/EEG have been studied, and we should expect that there are measures of M/EEG that can predict behavior and cognition that have yet to be found. Even with the advent of machine learning techniques which can explore rich data sets, we expect neurocognitive modeling to reveal more about M/EEG data in the future. Within M/EEG multivariate time series, there are also artifactual components, discussed previously, embedded in both scalp EEG, MEG, and intracranial EEG, but especially scalp EEG and older MEG systems. Thus, including all relevant information for modeling full time-series of M/EEG in a theoretical model can be difficult.

However we do not need to find source estimates of M/EEG to better understand the neurocognitive theory. We could, for instance, translate specific EEG potentials into cognition, EEG potentials like the CPP slope, which in **Model 3** is hypothesized to be generated from

mean evidence accumulation rate during a trial. We could also instead build models that describe the M/EEG phenomena, such as specific waveforms, while not assuming any particular type of cognition or brain activity. That is, we could describe the observed phenomena and not develop a neurocognitive understanding of M/EEG. This may aid us by allowing measurement noise in observed M/EEG potentials in our models, for instance. One promising method is to simulate EEG time courses by simulating from Morlet wavelet transforms (Bridwell et al., 2018). Specific noise in M/EEG could also be simulated to understand the robustness of the modeling procedure. If specific signals from M/EEG are extracted for joint modeling, such as ERPs or power from a certain frequency band, capturing a full possible range of these signals and contaminants in simulation is beneficial to test the robustness of the procedure (Hawkins et al., 2017).

6.2. The future of joint modeling

We do not consider much of our past work (e.g. Nunez et al., 2015, 2017, 2019a) of modeling EEG and human behavior to be "true" joint modeling , in that the our models did not also describe EEG measures. We used hierarchical Bayesian methods, but only assumed simple linear influences of EEG measures on cognitive parameters, similar to **Model 2** presented above. These methods could be considered simple *Directed* approaches described by Palestro et al. (2018). Ultimately, as Cognitive Neuroscientists and Cognitive Modelers, we would like to develop computational theory that predicts both observed human behavior and EEG dynamics, such as in **Model 3**. Among the several approaches described by Turner et al. (2017); Palestro et al. (2018), researchers should ultimately seek to use an *Integrative* approach with *simultaneous modeling* of EEG and behavior to test neurocognitive theory. In this way of thinking about model-based cognitive neuroscience, what we have typically performed is *simultaneous joint modeling* with linear connectors between EEG measures and decision-making behavior (a simple *Directed* approach in the classes of joint models by Palestro et al. (2018)), but not using an *Integrative* approach. Furthermore researchers, including ourselves, conducting joint modeling studies have typically fit simultaneous joint models with linear connectors between EEG measures and cognitive parameters (e.g. Frank et al., 2015; Nunez et al., 2017; van Ravenzwaaij et al., 2017; Schubert et al., 2019). Future research should improve upon previous joint modeling work. In the future we wish to use the richness of EEG data and more informative human behavioral measures (e.g. eye-tracking) within joint modeling frameworks to answer important neurocognitive questions. This work will also lead to better *integrative* joint models with possible non-linear connections.

Recently developed model fitting procedures are making fitting joint models easier. We are particularly excited about algorithms that allow sampling from posteriors of joint models when a likelihood is not available in closed-form or difficult to derive and estimate. One particular promising program is BayesFlow, which finds posterior samples from simulation-based models using invertible neural networks (Radev et al., 2020; Schmitt et al., 2022). The first author with colleagues has already used this program with success for *Integrative* joint modeling of single-trial EEG and behavior during decision making (Ghaderi-Kangavari et al., 2023b). A similar promising and accessible method is to use neural networks to learn approximate likelihoods that can then be used to find posterior distributions of joint models (Fengler et al., 2021, 2022). In general we expect future model fitting procedures to be more

flexible in the types of models that can be fit to data, making joint modeling of M/EEG and behavioral data easier to implement.

We expect that joint modeling will gain popularity in various cognitive domains beyond evidence accumulation. This includes the extension of cognitive models of working memory (e.g. Oberauer and Lin, 2017; Oberauer and Lewandowsky, 2019b) and visual metacognition (e.g. Rahnev, 2021). Domain-specific knowledge will be necessary to extend existing cognitive models into neurocognitive models. We aim for this tutorial to serve as a helpful foundation in these advancements.

6.3. Conclusion

We hope this tutorial serves as a guide for those researchers and students interested in joint modeling of M/EEG and behavior. We have covered the possible motivations to perform joint modeling, the definition of joint models, building of joint models, simulating joint models, experimental design, artifactual processes in M/EEG data, specific M/EEG signals, model fitting implementations, parameter recovery, model comparison, and the future of joint modeling. We have focused our examples on the relationship of scalp-recorded EEG and decision-making. In particular we have used a guiding example of testing the hypothesized relationship of the Centro-Parietal Positivity (CPP) slope to evidence accumulation rate. However these techniques and principles can easily be applied to other neurocognitive domains and questions using both animal and human electrophysiology. We expect joint modeling to be able to answer questions that cannot be answered with other methods because joint modeling allows direct testing of neurocognitive theory. And we look forward to reading about future research using joint modeling of M/EEG and behavior.

7. Exercises

1. Test your EEG artifact identification skills by classifying EEG artifact using Independent Component Analysis. As of August 2023, the website from the makers of ICLabel (Pion-Tonachini et al., 2019) allowed you to practice labeling independent components as "Brain", "Muscle", "Eye", "Heart", "Line Noise", "Chan Noise", or "Other". Practice at the link `https://labeling.ucsd.edu/tutorial/practice` and give feedback to improve the ICLabel algorithm at `https://labeling.ucsd.edu/labelfeedback`.

2. Run the Python or R code from Model 3, rewrite the code in another language (e.g. Julia, see `https://github.com/JagsJulia/Jags.jl`), and/or rewrite the model code in another Probabilistic Programming Language other than JAGS (e.g. Stan) and then run the code. Plot histograms or density approximations of the response time, accuracy, and CPP slope data for some participants.

3. What model could be compared to Model 3 in order to test the hypothesis that the CPP slope on each trial is a reflection of evidence accumulation? Specifically, what model along with Model 3 could be fit to CPP slopes, response times, and accuracies to test this hypothesis?

4. How could we change Model 3 to test the hypothesis that the CPP slope reflects a *scaled* version of evidence accumulation rate, that is the mean rate of evidence accumulation δ is not in micro-volts μV? Assume that the CPP slope could be scaled differently in each participant due to scalp volume conduction differences across participants.

5. Using the given Python simulation of Model 3 as a guide, simulate from Model 1 while assuming that the CPP slopes come from a distribution of $Normal(3, 1^2)$ μV (microvolts) per second across trials, that θ_0 is equal to .25 seconds, θ_1 is equal to .1, σ is equal to .1, γ_0 is equal to 0, and γ_1 is equal to .3. Does this model produce similar (and in this case of Model 3, more realistic) values of response times r and accuracies x across trials n compared to Model 3?

Solutions are located after the *Appendix* of this document.

8. Further readings

Here we present a reading list containing articles and books that offer discussions on a range of relevant subjects within neurocognitive modeling. The reading list follows the same section headers as outlined in the main text.

1. Motivation to model

– Blohm, G., Kording, K. P., and Schrater, P. R. (2020). A How-to-Model Guide for Neuroscience. *eNeuro*, 7(1)

– Forstmann, B. U. and Wagenmakers, E.-J., editors (2015). *An Introduction to Model-Based Cognitive Neuroscience.* Springer New York, New York, NY

– Wang, Z. J. and Busemeyer, J. R. (2021). *Cognitive Choice Modeling.* Cognitive Choice Modeling. The MIT Press, Cambridge, MA, US

2. Models to describe joint data

– Harris, A. and Hutcherson, C. A. (2022). Temporal dynamics of decision making: A synthesis of computational and neurophysiological approaches. *WIREs Cognitive Science*, 13(3):e1586

– Palestro, J. J., Bahg, G., Sederberg, P. B., Lu, Z.-L., Steyvers, M., and Turner, B. M. (2018). A tutorial on joint models of neural and behavioral measures of cognition. *Journal of Mathematical Psychology*, 84:20–48

– Turner, B. M., Forstmann, B. U., and Steyvers, M. (2019). *Joint Models of Neural and Behavioral Data.* Computational Approaches to Cognition and Perception. Springer International Publishing

3. Experimental manipulations and experimental design

– Jensen, K. M. and MacDonald, J. A. (2023). Towards thoughtful planning of ERP studies: How participants, trials, and effect magnitude interact to influence statistical power across seven ERP components. *Psychophysiology*, 60(7):e14245

4. *Collection and preprocessing of M/EEG for joint modeling*

    – Boudewyn, M. A., Erickson, M. A., Winsler, K., Ragland, J. D., Yonelinas, A., Frank, M., Silverstein, S. M., Gold, J., MacDonald III, A. W., and Carter, C. S. (2023). Managing EEG studies: How to prepare and what to do once data collection has begun. *Psychophysiology*, page e14365

    – Cohen, M. X. (2014). *Analyzing Neural Time Series Data: Theory and Practice*. MIT Press

    – Luck, S. J. (2014). *An Introduction to the Event-Related Potential Technique, Second Edition*. MIT Press

    – Luck, S. J. (2022). *Applied Event-Related Potential Data Analysis*. LibreTexts

    – Nunez, M. D., Nunez, P. L., and Srinivasan, R. (2016). Electroencephalography (EEG): neurophysics, experimental methods, and signal processing. In Ombao, H., Linquist, M., Thompson, W., and Aston, J., editors, *Handbook of Neuroimaging Data Analysis*, pages 175–197. Chapman & Hall/CRC

5. *Implementing model fitting procedures and estimating parameters*

    – Baribault, B. and Collins, A. G. E. (2023). Troubleshooting Bayesian cognitive models. *Psychological Methods*

    – Etz, A. and Vandekerckhove, J. (2018). Introduction to Bayesian Inference for Psychology. *Psychonomic Bulletin & Review*, 25(1):5–34

    – Farrell, S. and Lewandowsky, S. (2018). *Computational Modeling of Cognition and Behavior*. Cambridge University Press, Cambridge

    – Lee, M. D. and Wagenmakers, E.-J. (2014). *Bayesian cognitive modeling: A practical course*. Cambridge university press

    – Myung, J. I. and Pitt, M. A. (2018). Model comparison in psychology. *Stevens' handbook of experimental psychology and cognitive neuroscience*, 5:85–118

    – McElreath, R. (2018). *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. CRC Press

    – Palestro, J. J., Bahg, G., Sederberg, P. B., Lu, Z.-L., Steyvers, M., and Turner, B. M. (2018). A tutorial on joint models of neural and behavioral measures of cognition. *Journal of Mathematical Psychology*, 84:20–48

    – Schad, D. J., Betancourt, M., and Vasishth, S. (2021). Toward a principled Bayesian workflow in cognitive science. *Psychological Methods*, 26(1):103–126

    – Wilson, R. C. and Collins, A. G. (2019). Ten simple rules for the computational modeling of behavioral data. *eLife*, 8:e49547

6. Discussion

– Bridwell, D. A., Cavanagh, J. F., Collins, A. G. E., Nunez, M. D., Srinivasan, R., Stober, S., and Calhoun, V. D. (2018). Moving Beyond ERP Components: A Selective Review of Approaches to Integrate EEG and Behavior. *Frontiers in Human Neuroscience*, 12:106

Acknowledgements

Conflict of interest statement

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Open practices statement

All the code and data simulations from this work are given in the following repositories as of July 2023: `https://github.com/mdnunez/pyhddmjags` and `https://github.com/kiante-fernandez/Rhddmjags`. These analyses were not preregistered.

References

Aho, K., Derryberry, D., and Peterson, T. (2014). Model selection for ecologists: the world-views of AIC and BIC. *Ecology*, 95(3):631–636.

Au, J., Katz, B., Moon, A., Talati, S., Abagis, T. R., Jonides, J., and Jaeggi, S. M. (2021). Post-training stimulation of the right dorsolateral prefrontal cortex impairs working memory training performance. *Journal of Neuroscience Research*, 99(10):2351–2363.

Bamber, D. and van Santen, J. P. H. (2000). How to Assess a Model's Testability and Identifiability. *Journal of Mathematical Psychology*, 44(1):20–40.

Baribault, B. and Collins, A. G. E. (2023). Troubleshooting Bayesian cognitive models. *Psychological Methods*.

Blohm, G., Kording, K. P., and Schrater, P. R. (2020). A How-to-Model Guide for Neuroscience. *eNeuro*, 7(1).

Bode, S., Sewell, D. K., Lilburn, S., Forte, J. D., Smith, P. L., and Stahl, J. (2012). Predicting Perceptual Decision Biases from Early Brain Activity. *Journal of Neuroscience*, 32(36):12488–12498.

Boehm, U., Annis, J., Frank, M. J., Hawkins, G. E., Heathcote, A., Kellen, D., Krypotos, A.-M., Lerche, V., Logan, G. D., Palmeri, T. J., et al. (2018). Estimating across-trial variability parameters of the diffusion decision model: Expert advice and recommendations. *Journal of Mathematical Psychology*, 87:46–75.

Boehm, U., van Maanen, L., Forstmann, B., and van Rijn, H. (2014). Trial-by-trial fluctuations in CNV amplitude reflect anticipatory adjustment of response caution. *NeuroImage*, 96:95–105.

Borst, J. P. and Anderson, J. R. (2015). The discovery of processing stages: Analyzing EEG data with hidden semi-Markov models. *NeuroImage*, 108:60–73.

Boudewyn, M. A., Erickson, M. A., Winsler, K., Ragland, J. D., Yonelinas, A., Frank, M., Silverstein, S. M., Gold, J., MacDonald III, A. W., and Carter, C. S. (2023). Managing EEG studies: How to prepare and what to do once data collection has begun. *Psychophysiology*, page e14365.

Bridwell, D. A., Cavanagh, J. F., Collins, A. G. E., Nunez, M. D., Srinivasan, R., Stober, S., and Calhoun, V. D. (2018). Moving Beyond ERP Components: A Selective Review of Approaches to Integrate EEG and Behavior. *Frontiers in Human Neuroscience*, 12:106.

Brown, S. D., Ratcliff, R., and Smith, P. L. (2006). Evaluating methods for approximating stochastic differential equations. *Journal of mathematical psychology*, 50(4):402–410.

Bürkner, P.-C. (2017). brms: An r package for bayesian multilevel models using stan. *Journal of statistical software*, 80:1–28.

Busemeyer, J. R., Gluth, S., Rieskamp, J., and Turner, B. M. (2019). Cognitive and neural bases of multi-attribute, multi-alternative, value-based decisions. *Trends in cognitive sciences*, 23(3):251–263.

Busemeyer, J. R. and Wang, Y.-M. (2000). Model comparisons and model selections based on generalization criterion methodology. *Journal of Mathematical Psychology*, 44(1):171–189.

Cai, C., Sekihara, K., and Nagarajan, S. S. (2018). Hierarchical multiscale Bayesian algorithm for robust MEG/EEG source reconstruction. *NeuroImage*, 183:698–715.

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). Stan: A Probabilistic Programming Language. *Journal of Statistical Software*, 76(1):1–32.

Cavanagh, J. F., Wiecki, T. V., Cohen, M. X., Figueroa, C. M., Samanta, J., Sherman, S. J., and Frank, M. J. (2011). Subthalamic nucleus stimulation reverses mediofrontal influence over decision threshold. *Nature Neuroscience*, 14(11):1462–1467.

Chandrasekaran, C. and Hawkins, G. E. (2019). ChaRTr: An R toolbox for modeling choices and response times in decision-making tasks. *Journal of Neuroscience Methods*, 328:108432.

Charupanit, K. and Lopour, B. (2017). A simple statistical method for the automatic detection of ripples in human intracranial EEG. *Brain topography*, 30(6):724–738.

Chrysikou, E. G., Berryhill, M. E., Bikson, M., and Coslett, H. B. (2017). Editorial: Revisiting the Effectiveness of Transcranial Direct Current Brain Stimulation for Cognition: Evidence, Challenges, and Open Questions. *Frontiers in Human Neuroscience*, 11.

Claus, S., Velis, D., Lopes da Silva, F. H., Viergever, M. A., and Kalitzin, S. (2012). High frequency spectral components after Secobarbital: The contribution of muscular origin—A study with MEG/EEG. *Epilepsy Research*, 100(1):132–141.

Cohen, M. X. (2014). *Analyzing Neural Time Series Data: Theory and Practice*. MIT Press.

Craddock, M. (2023). eegUtils: Utilities for eeg data analysis. `https://github.com/craddm/eegUtils`. Accessed on: 2023-08-19.

Crone, N. E., Miglioretti, D. L., Gordon, B., Sieracki, J. M., Wilson, M. T., Uematsu, S., and Lesser, R. P. (1998). Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis. I. Alpha and beta event-related desynchronization. *Brain*, 121(12):2271–2299.

Daunizeau, J., David, O., and Stephan, K. E. (2011). Dynamic causal modelling: A critical review of the biophysical and statistical foundations. *NeuroImage*, 58(2):312–322.

David, O. and Friston, K. J. (2003). A neural mass model for MEG/EEG:: coupling and neuronal dynamics. *NeuroImage*, 20(3):1743–1755.

Delorme, A. and Makeig, S. (2004). EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of neuroscience methods*, 134(1):9–21.

Devezer, B., Navarro, D. J., Vandekerckhove, J., and Ozge Buzbas, E. (2021). The case for formal methodology in scientific reform. *Royal Society Open Science*, 8(3):200805.

Dickey, J. M. and Lientz, B. P. (1970). The weighted likelihood ratio, sharp hypotheses about chances, the order of a markov chain. *The Annals of Mathematical Statistics*, 41(1):214–226.

Ding, J., Sperling, G., and Srinivasan, R. (2006). Attentional modulation of SSVEP power depends on the network tagged by the flicker frequency. *Cerebral cortex*, 16(7):1016–1029.

Donoghue, T., Schaworonkow, N., and Voytek, B. (2022). Methodological considerations for studying neural oscillations. *European Journal of Neuroscience*, 55(11-12):3502–3527.

Dutilh, G., Annis, J., Brown, S. D., Cassey, P., Evans, N. J., Grasman, R. P. P. P., Hawkins, G. E., Heathcote, A., Holmes, W. R., Krypotos, A.-M., Kupitz, C. N., Leite, F. P., Lerche, V., Lin, Y.-S., Logan, G. D., Palmeri, T. J., Starns, J. J., Trueblood, J. S., van Maanen, L., van Ravenzwaaij, D., Vandekerckhove, J., Visser, I., Voss, A., White, C. N., Wiecki, T. V., Rieskamp, J., and Donkin, C. (2019). The Quality of Response Time Data Inference: A Blinded, Collaborative Assessment of the Validity of Cognitive Models. *Psychonomic Bulletin & Review*, 26(4):1051–1069.

Etienne, A., Laroia, T., Weigle, H., Afelin, A., Kelly, S. K., Krishnan, A., and Grover, P. (2020). Novel Electrodes for Reliable EEG Recordings on Coarse and Curly Hair. In *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 6151–6154. IEEE.

Etz, A. and Vandekerckhove, J. (2018). Introduction to Bayesian Inference for Psychology. *Psychonomic Bulletin & Review*, 25(1):5–34.

Farrell, S. and Lewandowsky, S. (2018). *Computational Modeling of Cognition and Behavior*. Cambridge University Press, Cambridge.

Farrens, J., Simmons, A., Luck, S., and Kappenman, E. (2020). Electroencephalogram (EEG) Recording Protocol for Cognitive and Affective Human Neuroscience Researc. Technical report.

Fengler, A., Bera, K., Pedersen, M. L., and Frank, M. J. (2022). Beyond Drift Diffusion Models: Fitting a Broad Class of Decision and Reinforcement Learning Models with HDDM. *Journal of Cognitive Neuroscience*, 34(10):1780–1805.

Fengler, A., Govindarajan, L. N., Chen, T., and Frank, M. J. (2021). Likelihood approximation networks (LANs) for fast inference of simulation models in cognitive neuroscience. *eLife*, 10:e65074.

Fitzgibbon, S., DeLosAngeles, D., Lewis, T., Powers, D., Grummett, T., Whitham, E., Ward, L., Willoughby, J., and Pope, K. (2016). Automatic determination of emg-contaminated components and validation of independent component analysis using eeg during pharmacologic paralysis. *Clinical neurophysiology*, 127(3):1781–1793.

Forstmann, B., Ratcliff, R., and Wagenmakers, E.-J. (2016). Sequential Sampling Models in Cognitive Neuroscience: Advantages, Applications, and Extensions. *Annual Review of Psychology*, 67(1):641–666.

Forstmann, B. U. and Wagenmakers, E.-J., editors (2015). *An Introduction to Model-Based Cognitive Neuroscience*. Springer New York, New York, NY.

Frank, M. J., Gagne, C., Nyhus, E., Masters, S., Wiecki, T. V., Cavanagh, J. F., and Badre, D. (2015). fmri and eeg predictors of dynamic decision parameters during human reinforcement learning. *Journal of Neuroscience*, 35(2):485–494.

Gelman, A. (2020). Prior Choice Recommendations. https://github.com/stan-dev/stan/wiki/Prior-Choice-Recommendations.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). *Bayesian Data Analysis*. Taylor & Francis Group, LLC, Boca Raton, FL, 3rd edition.

Ghaderi-Kangavari, A., Parand, K., Ebrahimpour, R., Nunez, M. D., and Amani Rad, J. (2023a). How spatial attention affects the decision process: looking through the lens of bayesian hierarchical diffusion model & eeg analysis. *Journal of Cognitive Psychology*, 35(4):456–479.

Ghaderi-Kangavari, A., Rad, J. A., and Nunez, M. D. (2023b). A general integrative neurocognitive modeling framework to jointly describe eeg and decision-making on single trials. *Computational Brain & Behavior*, pages 1–60.

Ghaderi-Kangavari, A., Rad, J. A., Parand, K., and Nunez, M. D. (2022). Neuro-cognitive models of single-trial eeg measures describe latent effects of spatial attention during perceptual decision making. *Journal of Mathematical Psychology*, 111:102725.

Gherman, S. and Philiastides, M. G. (2018). Human VMPFC encodes early signatures of confidence in perceptual decisions. *eLife*, 7:e38293.

Glomb, K., Cabral, J., Cattani, A., Mazzoni, A., Raj, A., and Franceschiello, B. (2021). Computational Models in Electroencephalography. *Brain Topography*.

Gluth, S., Rieskamp, J., and Büchel, C. (2013). Classic EEG motor potentials track the emergence of value-based decisions. *NeuroImage*, 79:394–403.

Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., Goj, R., Jas, M., Brooks, T., Parkkonen, L., et al. (2013). MEG and EEG data analysis with MNE-Python. *Frontiers in neuroscience*, page 267.

Greischar, L. L., Burghy, C. A., van Reekum, C. M., Jackson, D. C., Pizzagalli, D. A., Mueller, C., and Davidson, R. J. (2004). Effects of electrode density and electrolyte spreading in dense array electroencephalographic recording. *Clinical Neurophysiology*, 115(3):710–720.

Gronau, Q. F., Heathcote, A., and Matzke, D. (2020). Computing bayes factors for evidence-accumulation models using warp-iii bridge sampling. *Behavior research methods*, 52(2):918–937.

Gronau, Q. F., Sarafoglou, A., Matzke, D., Ly, A., Boehm, U., Marsman, M., Leslie, D. S., Forster, J. J., Wagenmakers, E.-J., and Steingroever, H. (2017). A tutorial on bridge sampling. *Journal of mathematical psychology*, 81:80–97.

Guest, O. and Martin, A. E. (2021). How Computational Modeling Can Force Theory Building in Psychological Science. *Perspectives on Psychological Science*, 16(4):789–802.

Hagen, E., Magnusson, S. H., Ness, T. V., Halnes, G., Babu, P. N., Linssen, C., Morrison, A., and Einevoll, G. T. (2022). Brain signal predictions from multi-scale networks using a linearized framework. *PLOS Computational Biology*, 18(8):e1010353.

Hanes, D. P. and Schall, J. D. (1996). Neural control of voluntary movement initiation. *Science*, 274(5286):427–430.

Harris, A. and Hutcherson, C. A. (2022). Temporal dynamics of decision making: A synthesis of computational and neurophysiological approaches. *WIREs Cognitive Science*, 13(3):e1586.

Hautus, M. J., Macmillan, N. A., and Creelman, C. D. (2021). *Detection theory: A user's guide*. Routledge.

Hawkins, G. E., Cavanagh, J. F., Brown, D., S., and Steyvers, M. (2023). Cognitive models as a tool to link decision behavior with EEG signals. In Turner, B. M. and Forstmann, B. U., editors, *An Introduction to Model-Based Cognitive Neuroscience, 2nd Ed.* Springer New York, New York, NY. Forthcoming.

Hawkins, G. E. and Heathcote, A. (2021). Racing against the clock: Evidence-based versus time-based decisions. *Psychological Review*, 128(2):222.

Hawkins, G. E., Mittner, M., Forstmann, B. U., and Heathcote, A. (2017). On the efficiency of neurally-informed cognitive models to identify latent cognitive states. *Journal of Mathematical Psychology*, 76:142–155.

Hawkins, G. E., Mittner, M., Forstmann, B. U., and Heathcote, A. (2022). Self-reported mind wandering reflects executive control and selective attention. *Psychonomic Bulletin & Review*, 29(6):2167–2180.

Heathcote, A., Lin, Y.-S., Reynolds, A., Strickland, L., Gretton, M., and Matzke, D. (2019). Dynamic models of choice. *Behavior Research Methods*, 51(2):961–985.

Heathcote, A. and Matzke, D. (2022). Winner takes all! What are race models, and why and how should psychologists use them? *Current Directions in Psychological Science*, 31(5):383–394.

Hyvärinen, A. and Oja, E. (1997). A Fast Fixed-Point Algorithm for Independent Component Analysis. *Neural Computation*, 9(7):1483–1492.

Ilmoniemi, R. J. and Sarvas, J. (2019). *Brain Signals: Physics and Mathematics of MEG and EEG*. MIT Press.

Jagannathan, S. R., Bareham, C. A., and Bekinschtein, T. A. (2021). Decreasing alertness modulates perceptual decision-making. *Journal of Neuroscience*.

Jeffreys, H. (1961). *Theory of probability*. Oxford University Press.

Jensen, K. M. and MacDonald, J. A. (2023). Towards thoughtful planning of ERP studies: How participants, trials, and effect magnitude interact to influence statistical power across seven ERP components. *Psychophysiology*, 60(7):e14245.

Jensen, O. and Mazaheri, A. (2010). Shaping Functional Architecture by Oscillatory Alpha Activity: Gating by Inhibition. *Frontiers in Human Neuroscience*, 4.

Jun, E. J., Bautista, A. R., Nunez, M. D., Allen, D. C., Tak, J. H., Alvarez, E., and Basso, M. A. (2021). Causal role for the primate superior colliculus in the computation of evidence for perceptual decisions. *Nature Neuroscience*, pages 1–11.

Jung, T.-P., Makeig, S., Humphries, C., Lee, T.-W., Mckeown, M. J., Iragui, V., and Sejnowski, T. J. (2000). Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*, 37(02):163–178.

Kappenman, E. S. and Luck, S. J. (2010). The effects of electrode impedance on data quality and statistical significance in ERP recordings. *Psychophysiology*, 47(5):888–904.

Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795.

Kelly, S. P. and O'Connell, R. G. (2013). Internal and External Influences on the Rate of Sensory Evidence Accumulation in the Human Brain. *Journal of Neuroscience*, 33(50):19434–19441.

Klatt, L.-I., Schneider, D., Schubert, A.-L., Hanenberg, C., Lewald, J., Wascher, E., and Getzmann, S. (2020). Unraveling the Relation between EEG Correlates of Attentional Orienting and Sound Localization Performance: A Diffusion Model Approach. *Journal of Cognitive Neuroscience*, 32(5):945–962.

Kohl, C., Spieser, L., Forster, B., Bestmann, S., and Yarrow, K. (2020). Centroparietal activity mirrors the decision variable when tracking biased and time-varying sensory evidence. *Cognitive Psychology*, 122:101321.

Kording, K., Blohm, G., Schrater, P., and Kay, K. (2018). Appreciating diversity of goals in computational neuroscience. *PsyArXiv*.

Krajbich, I. and Rangel, A. (2011). Multialternative drift-diffusion model predicts the relationship between visual fixations and choice in value-based decisions. *Proceedings of the National Academy of Sciences*, 108(33):13852–13857.

Kvam, P. D., Marley, A., and Heathcote, A. (2023). A unified theory of discrete and continuous responding. *Psychological Review*, 130(2):368.

Lee, M. D. (2011). How cognitive modeling can benefit from hierarchical Bayesian models. *Journal of Mathematical Psychology*, 55(1):1–7.

Lee, M. D., Criss, A. H., Devezer, B., Donkin, C., Etz, A., Leite, F. P., Matzke, D., Rouder, J. N., Trueblood, J. S., White, C. N., and Vandekerckhove, J. (2019). Robust Modeling in Cognitive Science. *Computational Brain & Behavior*, 2(3):141–153.

Lee, M. D. and Vanpaemel, W. (2018). Determining informative priors for cognitive models. *Psychonomic Bulletin & Review*, 25(1):114–127.

Lee, M. D. and Wagenmakers, E.-J. (2014). *Bayesian cognitive modeling: A practical course.* Cambridge university press.

Lerche, V., von Krause, M., Voss, A., Frischkorn, G. T., Schubert, A.-L., and Hagemann, D. (2020). Diffusion modeling and intelligence: Drift rates show both domain-general and domain-specific relations with intelligence. *Journal of Experimental Psychology: General*, 149(12):2207.

Lerche, V., Voss, A., and Nagler, M. (2017). How many trials are required for parameter estimation in diffusion modeling? a comparison of different optimization criteria. *Behavior research methods*, 49:513–537.

Li, A., Feitelberg, J., Saini, A. P., Höchenberger, R., and Scheltienne, M. (2022). MNE-ICALabel: Automatically annotating ICA components with ICLabel in Python. *Journal of Open Source Software*, 7(76):4484.

Loughnane, G. M., Newman, D. P., Bellgrove, M. A., Lalor, E. C., Kelly, S. P., and O'Connell, R. G. (2016). Target Selection Signals Influence Perceptual Decisions by Modulating the Onset and Rate of Evidence Accumulation. *Current Biology*, 26(4):496–502.

Luck, S. J. (2012). Event-related potentials. In *APA handbook of research methods in psychology, Vol 1: Foundations, planning, measures, and psychometrics*, APA handbooks in psychology®, pages 523–546. American Psychological Association, Washington, DC, US.

Luck, S. J. (2014). *An Introduction to the Event-Related Potential Technique, Second Edition*. MIT Press.

Luck, S. J. (2022). *Applied Event-Related Potential Data Analysis*. LibreTexts.

Lui, K. K., Nunez, M. D., Cassidy, J. M., Vandekerckhove, J., Cramer, S. C., and Srinivasan, R. (2021). Timing of readiness potentials reflect a decision-making process in the human brain. *Computational Brain & Behavior*, 4(3):264–283.

Makeig, S., Bell, A. J., Jung, T.-P., and Sejnowski, T. J. (1996). Independent component analysis of electroencephalographic data. *Advances in neural information processing systems*, pages 145–151.

Manning, C., Hassall, C. D., Laurence, T. H., Norcia, A. M., Wagenmakers, E.-J., Snowling, M. J., Scerif, G., and Evans, N. J. (2021). Visual motion and decision-making in dyslexia: Evidence of reduced accumulation of sensory evidence and related neural dynamics. Technical report.

Matzke, D. and Wagenmakers, E.-J. (2009). Psychological interpretation of the ex-Gaussian and shifted Wald parameters: A diffusion model analysis. *Psychonomic Bulletin & Review*, 16(5):798–817.

McElreath, R. (2018). *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. CRC Press.

McElreath, R. (2020). *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. Chapman and Hall/CRC, New York, 2 edition.

McFarland, D. J., Miner, L. A., Vaughan, T. M., and Wolpaw, J. R. (2000). Mu and Beta Rhythm Topographies During Motor Imagery and Actual Movements. *Brain Topography*, 12(3):177–186.

Mendes, A. J., Pacheco-Barrios, K., Lema, A., Gonçalves, Ó. F., Fregni, F., Leite, J., and Carvalho, S. (2022). Modulation of the cognitive event-related potential P3 by transcranial direct current stimulation: Systematic review and meta-analysis. *Neuroscience & Biobehavioral Reviews*, 132:894–907.

Mognon, A., Jovicich, J., Bruzzone, L., and Buiatti, M. (2011). ADJUST: An automatic EEG artifact detector based on the joint use of spatial and temporal features. *Psychophysiology*, 48(2):229–240.

Muthukumaraswamy, S. (2013). High-frequency brain activity and muscle artifacts in MEG/EEG: A review and recommendations. *Frontiers in Human Neuroscience*, 7.

Myung, J. I. and Pitt, M. A. (2018). Model comparison in psychology. *Stevens' handbook of experimental psychology and cognitive neuroscience*, 5:85–118.

Næss, S., Halnes, G., Hagen, E., Hagler, D. J., Dale, A. M., Einevoll, G. T., and Ness, T. V. (2021). Biophysically detailed forward modeling of the neural origin of EEG and MEG signals. *NeuroImage*, 225:117467.

Navarro, D. J. (2019). Between the Devil and the Deep Blue Sea: Tensions Between Scientific Judgement and Statistical Model Selection. *Computational Brain & Behavior*, 2(1):28–34.

Newsome, W. T. and Pare, E. B. (1988). A selective impairment of motion perception following lesions of the middle temporal visual area (MT). *Journal of Neuroscience*, 8(6):2201–2211.

Nunez, M. D., Charupanit, K., Sen-Gupta, I., Lopour, B. A., and Lin, J. J. (2022). Beyond rates: time-varying dynamics of high frequency oscillations as a biomarker of the seizure onset zone. *Journal of Neural Engineering*, 19(1):016034.

Nunez, M. D., Gosai, A., Vandekerckhove, J., and Srinivasan, R. (2019a). The latency of a visual evoked potential tracks the onset of decision making. *NeuroImage*, 197:93–108.

Nunez, M. D., Nunez, P. L., and Srinivasan, R. (2016). Electroencephalography (EEG): neurophysics, experimental methods, and signal processing. In Ombao, H., Linquist, M., Thompson, W., and Aston, J., editors, *Handbook of Neuroimaging Data Analysis*, pages 175–197. Chapman & Hall/CRC.

Nunez, M. D., Srinivasan, R., and Vandekerckhove, J. (2015). Individual differences in attention influence perceptual decision making. *Frontiers in Psychology*, 8.

Nunez, M. D., Vandekerckhove, J., and Srinivasan, R. (2017). How attention influences perceptual decision making: Single-trial EEG correlates of drift-diffusion model parameters. *Journal of Mathematical Psychology*, 76:117–130.

Nunez, P. L., Nunez, M. D., and Srinivasan, R. (2019b). Multi-Scale Neural Sources of EEG: Genuine, Equivalent, and Representative. A Tutorial Review. *Brain Topography*, 32(2):193–214.

Nunez, P. L. and Srinivasan, R. (2006). *Electric fields of the brain: the neurophysics of EEG*. Oxford University Press, Oxford ; New York, 2nd ed edition.

Nunez, P. L., Wingeier, B. M., and Silberstein, R. B. (2001). Spatial-temporal structures of human alpha rhythms: Theory, microcurrent sources, multiscale measurements, and global binding of local networks. *Human Brain Mapping*, 13(3):125–164.

Oberauer, K. and Lewandowsky, S. (2019a). Addressing the theory crisis in psychology. *Psychonomic bulletin & review*, 26:1596–1618.

Oberauer, K. and Lewandowsky, S. (2019b). Simple measurement models for complex working-memory tasks. *Psychological Review*, 126(6):880.

Oberauer, K. and Lin, H.-Y. (2017). An interference model of visual working memory. *Psychological review*, 124(1):21.

O'Connell, R. G., Dockree, P. M., and Kelly, S. P. (2012). A supramodal accumulation-to-bound signal that determines perceptual decisions in humans. *Nature Neuroscience*, 15(12):1729–1735.

Oostenveld, R., Fries, P., Maris, E., and Schoffelen, J.-M. (2011). FieldTrip: open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. *Computational intelligence and neuroscience*, 2011:1–9.

Ostwald, D., Porcaro, C., Mayhew, S. D., and Bagshaw, A. P. (2012). EEG-fMRI Based Information Theoretic Characterization of the Human Perceptual Decision System. *PLOS ONE*, 7(4):e33896.

O'Connell, R. G., Shadlen, M. N., Wong-Lin, K., and Kelly, S. P. (2018). Bridging Neural and Computational Viewpoints on Perceptual Decision-Making. *Trends in Neurosciences*, 41(11):838–852.

Palestro, J. J., Bahg, G., Sederberg, P. B., Lu, Z.-L., Steyvers, M., and Turner, B. M. (2018). A tutorial on joint models of neural and behavioral measures of cognition. *Journal of Mathematical Psychology*, 84:20–48.

Parra, L. C., Spence, C. D., Gerson, A. D., and Sajda, P. (2005). Recipes for the linear analysis of EEG. *NeuroImage*, 28(2):326–341.

Pfurtscheller, G., Stancák, A., and Neuper, C. (1996). Event-related synchronization (ERS) in the alpha band — an electrophysiological correlate of cortical idling: A review. *International Journal of Psychophysiology*, 24(1):39–46.

Philiastides, M. G., Heekeren, H. R., and Sajda, P. (2014). Human Scalp Potentials Reflect a Mixture of Decision-Related Signals during Perceptual Choices. *Journal of Neuroscience*, 34(50):16877–16889.

Philiastides, M. G., Ratcliff, R., and Sajda, P. (2006). Neural Representation of Task Difficulty and Decision Making during Perceptual Categorization: A Timing Diagram. *Journal of Neuroscience*, 26(35):8965–8975.

Pion-Tonachini, L., Kreutz-Delgado, K., and Makeig, S. (2019). ICLabel: An automated electroencephalographic independent component classifier, dataset, and website. *NeuroImage*, 198:181–197.

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. In *Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003), Vienna, Austria.*

Polanía, R., Krajbich, I., Grueschow, M., and Ruff, C. C. (2014). Neural Oscillations and Synchronization Differentially Support Evidence Accumulation in Perceptual and Value-Based Decision Making. *Neuron*, 82(3):709–720.

Radev, S. T., Mertens, U. K., Voss, A., Ardizzone, L., and Köthe, U. (2020). BayesFlow: Learning Complex Stochastic Models With Invertible Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15.

Rahnev, D. (2021). Visual metacognition: Measures, models, and neural correlates. *American Psychologist*, 76(9):1445–1453.

Rangelov, D. and Mattingley, J. B. (2020). Evidence accumulation during perceptual decision-making is sensitive to the dynamics of attentional selection. *NeuroImage*, page 117093.

Ratcliff, R. (2018). Decision making on spatially continuous scales. *Psychological review*, 125(6):888.

Ratcliff, R., Philiastides, M. G., and Sajda, P. (2009). Quality of evidence for perceptual decision making is indexed by trial-to-trial variability of the EEG. *Proceedings of the National Academy of Sciences*, 106(16):6539–6544.

Ratcliff, R., Smith, P. L., Brown, S. D., and McKoon, G. (2016). Diffusion Decision Model: Current Issues and History. *Trends in Cognitive Sciences*, 20(4):260–281.

Regan, D. (1977). Steady-state evoked potentials. *JOSA*, 67(11):1475–1489.

Roitman, J. D. and Shadlen, M. N. (2002). Response of Neurons in the Lateral Intraparietal Area during a Combined Visual Discrimination Reaction Time Task. *Journal of Neuroscience*, 22(21):9475–9489.

Rouder, J. N. and Morey, R. D. (2012). Default bayes factors for model selection in regression. *Multivariate Behavioral Research*, 47(6):877–903.

Rouder, J. N., Province, J. M., Morey, R. D., Gomez, P., and Heathcote, A. (2015). The lognormal race: A cognitive-process model of choice and latency with desirable psychometric properties. *Psychometrika*, 80:491–513.

Salvatier, J., Wiecki, T. V., and Fonnesbeck, C. (2016). Probabilistic programming in Python using PyMC3. *PeerJ Computer Science*, 2:e55.

Schad, D. J., Betancourt, M., and Vasishth, S. (2021). Toward a principled Bayesian workflow in cognitive science. *Psychological Methods*, 26(1):103–126.

Schall, J. D. (2004). On building a bridge between brain and behavior. *Annu. Rev. Psychol.*, 55:23–50.

Schaworonkow, N. and Voytek, B. (2021). Enhancing oscillations in intracranial electrophysiological recordings with data-driven spatial filters. *PLOS Computational Biology*, 17(8):e1009298.

Schmitt, M., Bürkner, P.-C., Köthe, U., and Radev, S. T. (2022). BayesFlow Can Reliably Detect Model Misspecification and Posterior Errors in Amortized Bayesian Inference.

Schubert, A.-L., Frischkorn, G. T., Hagemann, D., and Voss, A. (2016). Trait characteristics of diffusion model parameters. *Journal of Intelligence*, 4(3):7.

Schubert, A.-L., Nunez, M. D., Hagemann, D., and Vandekerckhove, J. (2019). Individual Differences in Cortical Processing Speed Predict Cognitive Abilities: a Model-Based Cognitive Neuroscience Account. *Computational Brain & Behavior*, 2(2):64–84.

Shadlen, M. and Kiani, R. (2013). Decision Making as a Window on Cognition. *Neuron*, 80(3):791–806.

Smith, P. L. (2016). Diffusion theory of decision making in continuous report. *Psychological Review*, 123(4):425–451.

Srinivasan, R. (2004). Internal and external neural synchronization during conscious perception. *International Journal of Bifurcation and Chaos*, 14(02):825–842.

Srinivasan, R., Thorpe, S., and Nunez, P. L. (2013). Top-down influences on local networks: basic theory with experimental implications. *Frontiers in computational neuroscience*, 7:29.

Stevenson, N., Innes, R., Boag, R., and Heathcote, A. (2023). Model based sampling with EMC 2-Extended Models of Choice. Forthcoming.

Swart, J. C., Frank, M. J., Määttä, J. I., Jensen, O., Cools, R., and den Ouden, H. E. (2018). Frontal network dynamics reflect neurocomputational mechanisms for reducing maladaptive biases in motivated action. *PLoS biology*, 16(10):e2005979.

Teller, D. Y. (1984). Linking propositions. *Vision research*, 24(10):1233–1246.

The MathWorks Inc. (2022). MATLAB version: 9.13.0 (R2022b).

Thomas, A. W., Molter, F., and Krajbich, I. (2021). Uncovering the computational mechanisms underlying many-alternative choice. *eLife*, 10.

46

Tran, N.-H., van Maanen, L., Heathcote, A., and Matzke, D. (2021). Systematic Parameter Reviews in Cognitive Modeling: Towards a Robust and Cumulative Characterization of Psychological Processes in the Diffusion Decision Model. *Frontiers in Psychology*, 11.

Turner, B. M., Forstmann, B. U., Love, B. C., Palmeri, T. J., and Van Maanen, L. (2017). Approaches to analysis in model-based cognitive neuroscience. *Journal of Mathematical Psychology*, 76:65–79.

Turner, B. M., Forstmann, B. U., and Steyvers, M. (2019). *Joint Models of Neural and Behavioral Data*. Computational Approaches to Cognition and Perception. Springer International Publishing.

Turner, B. M., Rodriguez, C. A., Norcia, T. M., McClure, S. M., and Steyvers, M. (2016). Why more is better: Simultaneous modeling of EEG, fMRI, and behavioral data. *NeuroImage*, 128:96–115.

Twomey, D. M., Murphy, P. R., Kelly, S. P., and O'Connell, R. G. (2015). The classic P300 encodes a build-to-threshold decision variable. *European Journal of Neuroscience*, 42(1):1636–1643.

van Doorn, J., van den Bergh, D., Böhm, U., Dablander, F., Derks, K., Draws, T., Etz, A., Evans, N. J., Gronau, Q. F., Haaf, J. M., Hinne, M., Kucharský, Š., Ly, A., Marsman, M., Matzke, D., Gupta, A. R. K. N., Sarafoglou, A., Stefan, A., Voelkel, J. G., and Wagenmakers, E.-J. (2021). The JASP guidelines for conducting and reporting a Bayesian analysis. *Psychonomic Bulletin & Review*, 28(3):813–826.

van Ravenzwaaij, D., Brown, S. D., Marley, A., and Heathcote, A. (2020). Accumulating advantages: A new conceptualization of rapid multiple choice. *Psychological review*, 127(2):186.

van Ravenzwaaij, D., Cassey, P., and Brown, S. D. (2018). A simple introduction to Markov Chain Monte–Carlo sampling. *Psychonomic Bulletin & Review*, 25(1):143–154.

van Ravenzwaaij, D. and Etz, A. (2021). Simulation Studies as a Tool to Understand Bayes Factors. *Advances in Methods and Practices in Psychological Science*, 4(1):2515245920972624.

van Ravenzwaaij, D., Provost, A., and Brown, S. D. (2017). A confirmatory approach for integrating neural and behavioral data into a single model. *Journal of Mathematical Psychology*, 76:131–141.

van Rooij, I. and Baggio, G. (2020). Theory Development Requires an Epistemological Sea Change. *Psychological Inquiry*, 31(4):321–325.

van Vugt, M., Simen, P., Nystrom, L., Holmes, P., and Cohen, J. (2012). EEG Oscillations Reveal Neural Correlates of Evidence Accumulation. *Frontiers in Neuroscience*, 6:106.

Vandekerckhove, J., White, C. N., Trueblood, J. S., Rouder, J. N., Matzke, D., Leite, F. P., Etz, A., Donkin, C., Devezer, B., Criss, A. H., and Lee, M. D. (2019). Robust Diversity in Cognitive Science. *Computational Brain & Behavior*, 2(3):271–276.

Vehtari, A. (2023). Model selection tutorials and talks. https://avehtari.github.io/modelselection/CV-FAQ.html.

Verdinelli, I. and Wasserman, L. (1995). Computing Bayes factors using a generalization of the savage-dickey density ratio. *Journal of the American Statistical Association*, 90(430):614–618.

Villarreal, J. M., Chávez, A., Mistry, P. K., Menon, V. E., Vandekerckhove, J., and Lee, M. (2023). Bayesian graphical modeling with the circular drift diffusion model.

Voss, A., Rothermund, K., and Voss, J. (2004). Interpreting the parameters of the diffusion model: An empirical validation. *Memory & Cognition*, 32(7):1206–1220.

Wabersich, D. and Vandekerckhove, J. (2013). *jags-wiener: a diffusion model plugin for JAGS*.

Wabersich, D. and Vandekerckhove, J. (2014). Extending JAGS: A tutorial on adding custom distributions to JAGS (with a diffusion model example). *Behavior Research Methods*, 46:15–28.

Wagenmakers, E.-J., Lodewyckx, T., Kuriyal, H., and Grasman, R. (2010). Bayesian hypothesis testing for psychologists: A tutorial on the savage–dickey method. *Cognitive psychology*, 60(3):158–189.

Walter, E. (1987). *Identifiability of Parametric Models*. Elsevier Science & Technology.

Wang, Z. J. and Busemeyer, J. R. (2021). *Cognitive Choice Modeling*. Cognitive Choice Modeling. The MIT Press, Cambridge, MA, US.

Weindel, G., van Maanen, L., and Borst, J. P. (2023). HsMM Mvpy. `https://github.com/GWeindel/hsmm_mvpy`. Accessed on: 2023-08-19.

Whitham, E. M., Pope, K. J., Fitzgibbon, S. P., Lewis, T., Clark, C. R., Loveless, S., Broberg, M., Wallace, A., DeLosAngeles, D., Lillie, P., Hardy, A., Fronsko, R., Pulbrook, A., and Willoughby, J. O. (2007). Scalp electrical recording during paralysis: Quantitative evidence that EEG frequencies above 20Hz are contaminated by EMG. *Clinical Neurophysiology*, 118(8):1877–1888.

Wiecki, T. V., Sofer, I., and Frank, M. J. (2013). Hddm: Hierarchical bayesian estimation of the drift-diffusion model in python. *Frontiers in neuroinformatics*, page 14.

Wilson, R. C. and Collins, A. G. (2019). Ten simple rules for the computational modeling of behavioral data. *eLife*, 8:e49547.

Zhang, Q., van Vugt, M., Borst, J. P., and Anderson, J. R. (2018). Mapping working memory retrieval in space and in time: A combined electroencephalography and electrocorticography approach. *NeuroImage*, 174:472–484.

9. Appendix

9.1. An illustrative implementation of neurocognitive modeling

The following code is a slightly modified and annotated version of Python code at this location: `https://github.com/mdnunez/pyhddmjags/blob/master/simpleCPP_test.py`. For this tutorial, we will walk through the process of simulating data from **Model 3**, writing the corresponding JAGS code for the model, and fitting the model to our simulated data.

Simulating data

In Code Block 2 we first start by generating choice, response times, and CPP slopes from **Model 3** (for a brief description, see below, or for more in-depth information, refer to the main text). Lines 3-11, first import the necessary modules, as well as a custom module named pyhddmjagsutils. Lines 19-25, we specify the number of participants (`nparts = 100`) and trials (`ntrials = 100`) we would like to simulate. The total number of trials is then specified as `N = ntrials * nparts`. Lines 30-40 randomly initialize model parameters from uniform distributions with specified bounds. Line 47-57 then loops through each participant, simulating their choice and response times using the `simulratcliff` function from pyhddmjagsutils, and CPP slopes from a normal distribution. `simulratcliff` is exactly the code we showed previously for simulating choice and response time data in Code Block 1 in the main text.

The remainder of Code Block 2 initializes and stores all the generated data and parameters in a dictionary called 'genparam'.

```python
1
2  # Load Modules
3  import numpy as np
4  import pyjags
5  import scipy.io as sio
6  import os
7  import matplotlib.pyplot as plt
8  import matplotlib.gridspec as gridspec
9  from matplotlib import rc
0  from scipy import stats
1  import pyhddmjagsutils as phju
2
3  ### Simulations ###
4
5  # Generate samples from the joint-model of reaction time and choice
6  # Note you could remove this if statement and replace with loading your
      own data to dictionary "gendata"
7
8  # Number of simulated participants
9  nparts = 100
0
1  # Number of trials for one participant
2  ntrials = 100
3
4  # Number of total trials in each simulation
5  N = ntrials * nparts
6
7  # Set random seed
```

49

```
1745  np.random.seed(2021)
1746  # Uniform from .15 to .6 seconds
1747  ndt = np.random.uniform(.15, .6, size=nparts)
1748  # Uniform from .8 to 1.4 evidence units
1749  alpha = np.random.uniform(.8, 1.4, size=nparts)
1750  # Uniform from .3 to .7 * alpha
1751  beta = np.random.uniform(.3, .7, size=nparts)
1752  # Uniform from -4 to 4 evidence units per second
1753  delta = np.random.uniform(-4, 4, size=nparts)
1754  # Uniform from 0 to 2 evidence units per second
1755  deltatrialsd = np.random.uniform(0, 2, size=nparts)
1756  # Uniform from 0 to 1 evidence units per second
1757  CPPnoise = np.random.uniform(0, 1, size=nparts)
1758  y = np.zeros(N)
1759  rt = np.zeros(N)
1760  acc = np.zeros(N)
1761  CPP = np.zeros(N)
1762  participant = np.zeros(N)  # Participant index
1763  indextrack = np.arange(ntrials)
1764  for p in range(nparts):
1765      tempout = phju.simulratcliff(N=ntrials, Alpha=alpha[p], Tau=ndt[p],
1766      Beta=beta[p],
1767                                   Nu=delta[p], Eta=deltatrialsd[p])
1768      tempx = np.sign(np.real(tempout))
1769      tempt = np.abs(np.real(tempout))
1770      CPP[indextrack] = np.random.normal(loc=delta[p],scale=CPPnoise[p],size
1771      =ntrials)
1772      y[indextrack] = tempx * tempt
1773      rt[indextrack] = tempt
1774      acc[indextrack] = (tempx + 1) / 2
1775      participant[indextrack] = p + 1
1776      indextrack += ntrials
1777
1778  genparam = dict()
1779  genparam['ndt'] = ndt
1780  genparam['beta'] = beta
1781  genparam['alpha'] = alpha
1782  genparam['delta'] = delta
1783  genparam['deltatrialsd'] = deltatrialsd
1784  genparam['CPPnoise'] = CPPnoise
1785  genparam['CPP'] = CPP
1786  genparam['rt'] = rt
1787  genparam['acc'] = acc
1788  genparam['y'] = y
1789  genparam['participant'] = participant
1790  genparam['nparts'] = nparts
1791  genparam['ntrials'] = ntrials
1792  genparam['N'] = N
```

Code Block 2: generating simulated data from **Model 3**

1793 *Specifying a joint model*

1794   In this section, we describe how to specify a joint model. Specifically, we discuss how
1795 JAGS and Python handler code can be used to write models. In our example, the JAGS

code is written as a string within Python and saved into a file to be called during sampling. However, you can also specify your JAGS file separately. In the worked example we focus on **Model 3** as discussed the main text. We briefly restate it convenience here. Recall that **Model 3** tries to describe the cognitive parameters through the EEG measures themselves, allowing us to test the underlying computational role of the CPP slope in cognition.

$$(r_{ij}, x_{ij}) \sim DDM(\delta_j, \tau_j, \alpha_j, \beta_j, \eta_j) \tag{12}$$

$$c_{ij} \sim Normal(\delta_j, \ \sigma_j^2) \tag{13}$$

Let CPP slope be denoted by the variable c, accuracy be denoted by x, and response times be denoted by r. Note that x, r, and c can vary on every trial i and by participant j. In terms of parameters we are interested in the five cognitive parameters of a DDM, δ, τ, α, β, η and one computational parameter σ that vary by participant j. In **Model 3**, the mean of each trial's CPP slope c is described by the drift rate for each participant j. Lastly, we assume, for model *fitting*, that there is no extra trial-to-trial variability in drift-rate (e.g. assuming η = 0), due to recovery work by Dutilh et al. (2019) and our own recovery work above showing that assuming this parameter is not necessarily to estimate the other parameters.

The JAGS code consists of two sections: the first defines our priors on the model parameters, and the second defines the likelihood function. Lines 9-24 loop through each participant, specifying priors on each of the model parameters. Because we are fitting our model within a Bayesian framework, we need to specify our priors. For simplicity, we will use the normal distribution $\mathcal{N}$, parameterized by mean and variance respectively, and the truncated normal distribution with boundaries a and b denoted by $\in (a, b)$.

$$\alpha_j \sim \mathcal{N}(1, .5^2) \in (0, 3) \tag{14}$$

$$\tau_j \sim \mathcal{N}(.5, .25^2) \in (0, 1) \tag{15}$$

$$\beta_j \sim \mathcal{N}(.5, .25^2) \in (0, 1) \tag{16}$$

$$\delta_j \sim \mathcal{N}(0, 2^2) \tag{17}$$

$$\sigma_j \sim \mathcal{N}(1, .5^2) \in (0, 3) \tag{18}$$

For a discussion on prior selection in cognitive modeling, see Lee and Vanpaemel (2018), and for a discussion specific to DDMs, Matzke and Wagenmakers (2009); Tran et al. (2021). One thing to note is that JAGS parameterizes the normal distributions in terms of precision rather than standard deviation. Thus, throughout the model specification, we use the `pow` function to convert our standard deviations into precision parameters.

Next, Lines 29-36 define our likelihood function. Following equations (1) and (2), the likelihood function loops through each trial to generate samples from the joint posterior distribution of the model parameters, conditioned on the simulated data. This can be written more formally as:

$$p(\delta, \tau, \alpha, \beta, \sigma | y, c) \propto \mathcal{L}(\delta, \tau, \alpha, \beta | y) \mathcal{L}(\delta, \sigma | c) p(\delta, \tau, \alpha, \beta, \sigma) \tag{19}$$

51

This predicted joint distribution of the data, conditional on all parameters, is then the first passage time distribution of a Wiener process with constant drift ($\eta = 0$) and a normal distribution to describe the mean CPP slopes. Note that for each trial i, subject j, the observed accuracy x_{ij} and response time r_{ij} are combined into a two-element vector y_{ij}.

```
#write JAGS code
tojags = '''
model {

    ##########
    #Simple NDDM parameter priors
    ##########
    for (p in 1:nparts) {

        #Boundary parameter (speed-accuracy tradeoff) per participant
        alpha[p] ~ dnorm(1, pow(.5,-2))T(0, 3)

        #Non-decision time per participant
        ndt[p] ~ dnorm(.5, pow(.25,-2))T(0, 1)

        #Start point bias towards choice A per participant
        beta[p] ~ dnorm(.5, pow(.25,-2))T(0, 1)

        #Drift rate to choice A per participant
        delta[p] ~ dnorm(0, pow(2, -2))

        #Noise in observed EEG measure, the CentroParietal Positivity (CPP
    ) slope per participant
        CPPnoise[p] ~ dnorm(1, pow(.5,-2))T(0, 3)

    }

    ##########
    # Wiener likelihood
    for (i in 1:N) {

        # Observations of accuracy*RT for DDM process of rightward/
    leftward RT
        y[i] ~ dwiener(alpha[participant[i]], ndt[participant[i]], beta[
    participant[i]], delta[participant[i]])

        # Observations of CentroParietal Positivity (CPP) slope per trial
        CPP[i] ~ dnorm(delta[participant[i]],pow(CPPnoise[participant[i
    ]],-2))

    }
}
'''

modelfile = 'jagscode/simpleCPP_test1.jags'
f = open(modelfile, 'w')
f.write(tojags)
```

```
1876 15  f.close()
```

Code Block 3: writing **Model 3** in JAGS

Note that this model could be easily extended to a *hierarchical* model with changed *priors* and *hyperpriors*. We hierarchical model examples in the Python `https://github.com/mdnunez/pyhddmjags` and R `https://github.com/kiante-fernandez/Rhddmjags` repositories, see also *Further readings* of *Implementing model fitting procedures and estimating parameters*, especially by Lee and Wagenmakers (2014) for a discussion on hierarchical models with examples in JAGS.

Fitting a joint model to data

Once we have specified our joint model, we can fit it to the data. Specifically, we discuss how the Python handler code can be used to fit models to our simulated data. At this point, we will assume that you have installed the JAGS software and the necessary extension for sampling from the Wiener distribution in JAGS (for details, see Wabersich and Vandekerckhove (2014)). The goal of the following code is to sample from the joint posterior distribution specified in the Code Block and in equation 19 above.

We start by setting the seed in Line 3. Setting the seed in statistical simulations ensures that the sequence of random numbers generated is reproducible across different runs and machines. Lines 6-7 load the two JAGS extensions, one for the Wiener distribution and the other for conveniently calculating the deviance information criterion (DIC), which is a method for comparing the relative fit of a set of models (for a discussion on model comparisons, see Myung and Pitt (2018)). Lines 10-12 describe how many chains to use during sampling, how many to discard for burn-in, and the total number of samples to draw from our posterior respectively. Line 15 defines which parameters we are interested in collecting samples from one the sampling is finished. Lines 17-25 use our simulated data to define the variables we would like pass to the JAGS code.

Lines 27-43 set the initial values for JAGS MCMC sampler (for a discussion on MCMC, see van Ravenzwaaij et al. (2018)). While JAGS will define values based on the priors by default, setting initial values can help with convergence and efficiency in sampling our joint posterior. On the other hand, poorly selected starting values may result in slow or failed convergence, leading to unreliable estimates. For example, in our case, we know that `ndt` for a given participant should no more than their fastest response time, and thus we can randomly initialize the `ndt` parameter from uniform distributions with participant-specified bounds that reflect that. Lastly Lines 44-49 take in as input all of the variables specified above to draws samples the joint posterior distribution.

```
1909 1
1910 2  # Set random seed
1911 3  np.random.seed(2020)
1912 4
1913 5  # pyjags code
1914 6  pyjags.modules.load_module('wiener')
1915 7  pyjags.modules.load_module('dic')
1916 8  pyjags.modules.list_modules()
1917 9
1918 0  nchains = 6
1919 1  burnin = 2000   # Note that scientific notation breaks pyjags
```

```python
nsamps = 10000

# Track these variables
trackvars = ['alpha', 'ndt', 'beta', 'delta', 'CPPnoise']

N = np.squeeze(genparam['N'])

# Fit model to data
y = np.squeeze(genparam['y'])
rt = np.squeeze(genparam['rt'])
CPP = np.squeeze(genparam['CPP'])
participant = np.squeeze(genparam['participant'])
nparts = np.squeeze(genparam['nparts'])
ntrials = np.squeeze(genparam['ntrials'])

minrt = np.zeros(nparts)
for p in range(0, nparts):
    minrt[p] = np.min(rt[(participant == (p + 1))])

initials = []
for c in range(0, nchains):
    chaininit = {
        'alpha': np.random.uniform(.5, 2., size=nparts),
        'ndt': np.random.uniform(.1, .5, size=nparts),
        'beta': np.random.uniform(.2, .8, size=nparts),
        'delta': np.random.uniform(-4., 4., size=nparts),
        'CPPnoise': np.random.uniform(.5, 2., size=nparts)
    }
    for p in range(0, nparts):
        chaininit['ndt'][p] = np.random.uniform(0., minrt[p] / 2)
    initials.append(chaininit)
print('Fitting ''simpleEEG'' model ...')
threaded = pyjags.Model(file=modelfile, init=initials,
                        data=dict(y=y, CPP=CPP, N=N, nparts=nparts,
                                  participant=participant),
                        chains=nchains, adapt=burnin, threads=6,
                        progress_bar=True)
samples = threaded.sample(nsamps, vars=trackvars, thin=10)
savestring = ('modelfits/simpleEEG_test1_simpleCPP.mat')
print('Saving results to: \n %s' % savestring)
sio.savemat(savestring, samples)
```

Code Block 4: fitting **Model 3** to data simulated in Code Block 2

We have omitted a few technical details associated with employing Bayesian methods to fit computational models. For more comprehensive treatments, we suggest referring to our *Implementing model fitting procedures and estimating parameters* section in the *Further readings*.

Conclusions

In this appendix, we have showcased only one example implementation of neurocognitive modeling. In the tutorial repositories, we have designed several illustrative examples in Python: https://github.com/mdnunez/pyhddmjags and R: https://github.com/kiante-fernandez/Rhddmjags that follow a similar format to what we have explained here.

10. Solutions

1. Labeling feedback given by the ICLabel website.

2. Plotting of histograms are left to the reader. Plots of density approximations should approximately reproduce the results in **Figure 2**. The code for this figure is given in `https://github.com/mdnunez/pyhddmjags/blob/master/simpleCPP_sim.py`. For the corresponding translation in R see: `https://github.com/kiante-fernandez/Rhddmjags/blob/main/inst/templates/CPP_sim-example.R`.

3. A comparison model would be similar to Model 3 other than the fact that CPP slopes c per trial i are not generated by drift-rates δ, a cognitive parameter that describes the mean rate of evidence accumulation.

$$(r_{ij}, x_{ij}) \sim DDM(\delta_j, \tau_j, \alpha_j, \beta_j, \eta_j) \tag{20}$$
$$c_{ij} \quad\quad \sim Normal(\phi_j, \sigma_j^2) \tag{21}$$

Note that this comparison model has an additional parameter per participant, ϕ, compared to Model 3. ϕ is just the mean CPP slope across trials.

4. We could add an addition scaling parameter ψ in the second equation that can change based on participant j:

$$c_{ij} \sim Normal(\psi_j \delta_j, \sigma_j^2) \tag{22}$$

5. This model produces normally distributed response times, while Model 3 produces response time distributions with right skews. Apart from very specific conditions (e.g., when facing a strict deadline that puts one under severe time pressure), response time distributions are usually right skewed, making Model 3 more plausible.