Adaptive stimulus selection for multi-alternative psychometric functions with lapses

Ji Hyun Bak[1,2] **and Jonathan W. Pillow**[3]

[1]School of Computational Sciences, Korea Institute for Advanced Study, Seoul, Korea; [2]Department of Physics, Princeton University, NJ, USA;
[3]Department of Psychology and Princeton Neuroscience Institute, Princeton University, NJ, USA

(Dated: February 6, 2018)

Psychometric functions (PFs) quantify how external stimuli affect behavior and play an important role in building models of sensory and cognitive processes. Adaptive stimulus selection methods seek to select stimuli that are maximally informative about the PF given data observed so far in an experiment and thereby reduce the number of trials required to estimate the PF. Here we develop new adaptive stimulus selection methods for flexible PF models in tasks with two or more alternatives. We model the PF with a multinomial logistic regression mixture model that incorporates realistic aspects of psychophysical behavior, including lapses (trials where the observer ignores the stimulus) and omissions (trials where the observer “opts out” or fails to provide a valid response). We propose an information-theoretic criterion for stimulus selection and develop computationally efficient methods for inference and stimulus selection based on semi-adaptive Markov Chain Monte Carlo (MCMC) sampling. We apply these methods to data from macaque monkeys performing a multi-alternative motion discrimination task, and show in simulated experiments that our method can achieve a substantial speed-up over random designs. These advances will reduce the data needed to build accurate models of multi-alternative PFs and can be extended to high-dimensional PFs that would be infeasible to characterize with standard methods.

Keywords: adaptive stimulus selection, sequential optimal design, Bayesian adaptive design, psychometric function, closed-loop experiments

Introduction

Understanding the factors governing psychophysical behavior is a central problem in neuroscience and psychology. Although accurate quantification of the behavior is an important goal in itself, psychophysics provides an important tool for interrogating the mechanisms governing sensory and cognitive processing in the brain. As new technologies allow direct manipulations of neural activity in the brain, there is a growing need for methods that can characterize rapid changes in psychophysical behavior.

In a typical psychophysical experiment, an observer is trained to report judgements about a sensory stimulus by selecting a response from among two or more alternatives. The observer is assumed to have an internal probabilistic rule governing these decisions; this probabilistic map from stimulus to response is called the observer’s psychometric function. Because the psychometric function is not directly observable, it must be inferred from multiple observations of stimulus-response pairs. However, such experiments are costly due to the large numbers of trials typically required to obtain good estimates of psychometric functions. Therefore, a problem of major practical importance is to develop efficient experimental designs that can minimize the amount of data required to accurately infer an observer’s psychometric function.

Bayesian adaptive stimulus selection. A powerful approach for improving the efficiency of psychophysical experiments is to design the data collection process so that the stimulus is adaptively selected on each trial by maximizing a suitably defined objective function (MacKay, 1992). Such methods are known by a variety of names, including "active learning", "adaptive or sequential optimal experimental design", and "closed-loop experiments."

Bayesian approaches to adaptive stimulus selection define optimality of a stimulus in terms of its expected ability to improve the posterior distribution over the psychometric function, e.g., by reducing its variance or entropy. The three key ingredients of a Bayesian adaptive stimulus selection method are (Chaloner & Verdinelli, 1995; Pillow & Park, 2016):

- **model** - parametrizes the psychometric function of interest;
- **prior** - captures initial beliefs about model parameters;
- **utility function** - quantifies the usefulness of a hypothetical stimulus-response pair for improving the posterior.

Sequential algorithms for adaptive Bayesian experiments rely on repeated application of three basic steps: (1) data collection (stimulus presentation and response measurement); (2) inference (posterior updating using data from the most recent trial); and (3) selection of an optimal stimulus for the next trial by maximizing expected utility (see Fig. 1A). The inference step involves updating the posterior distribution over the model parameters according to Bayes rule with data from the most recent trial. Stimulus selection involves calculating the expected utlity (i.e., the expected improvement in the posterior) for a set of candidate stimuli, averaging over the responses that might be elicited for each stimulus, and selecting the stimulus for which the expected utility is highest. Example utility functions include the negative trace of the posterior covariance (corresponding to the sum of the posterior variances for each parameter) and the mutual information or information gain (which corresponds to minimizing the entropy of the posterior).

Methods for Bayesian adaptive stimulus selection have been developed over several decades in a variety of different disciplines. If we focus on the specific application of estimating psychometric functions, the field goes back to the QUEST algorithm (Watson & Pelli, 1983) for estimating discrimination thresholds, and the Ψ method (Kontsevich & Tyler, 1999) for estimating both threshold and slope of a psychometric function. These methods have been extended to models with more parameters (Kujala & Lukka, 2006; Lesmes, Lu, Baek, & Albright, 2010; Prins, 2013), in particular models with multi-dimensional stimuli (DiMattina, 2015; Kujala & Lukka, 2006; Watson, 2017). In parallel, the development of Bayesian methods for inferring psychometric functions (Kuss, Jäkel, & Wichmann, 2005; Prins, 2012; Wichmann & Hill, 2001) have enlarged the space of statistical models for psychophysical phenomena.

A variety of recent advances have arisen in sensory neuroscience or neurophysiology, driven by the development of efficient inference techniques for neural encoding models (Lewi, Butera, & Paninski, 2009; Park, Horwitz, & Pillow, 2011) or model comparison and discrimination methods (Cavagnaro, Myung, Pitt, & Kujala, 2010; DiMattina & Zhang, 2011; Kim, Pitt, Lu, Steyvers, & Myung, 2014). These advances can in many cases be equally well applied to psychophysical experiments.

One limitation of previous work is that has often considered only a restricted set of tractable psychometric function models. Standard choices including the logistic regression model (Chaloner & Larntz, 1989; Zocchi & Atkinson, 1999), the Weibull distribution function (Watson & Pelli, 1983), and the cumulative function of Gaussian distribution (Kontsevich & Tyler, 1999). In order for adaptive stimulus selection to be useful in realistic experimental settings, however, it is crucial to incorporate the system-specific features that are not fully captured by the standard models.

Our contributions. In this paper, we develop methods for adaptive stimulus selection in psychophysical experiments that are applicable to realistic models of human and animal psychophysical behavior. Our first contribution is to develop a model of psychometric function that incorporates two common "anomalies" of decision-making behavior: omission and lapse. By recognizing omission, we bring to light the well-known (but often ignored) possibility that an observer does not choose any of the provided set of actions, *omitting* the response for the trial. By recognizing lapse, we take into account the possibility that the observer makes occasional errors on easy trials due to momentary lapses in con-

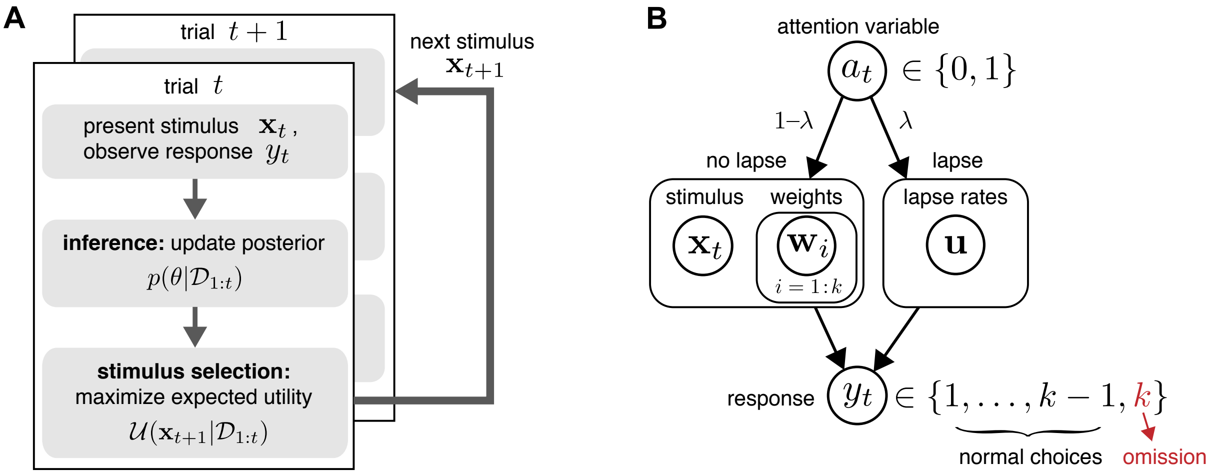

Figure 1: **(A)** Schematic of Bayesian adaptive stimulus selection. On each trial: (1) a stimulus is presented and response is observed; (2) the posterior over the parameters θ is updated using all data collected so far in the experiment $\mathcal{D}_t$; and (3) the stimulus that maximizes the expected utility (in our case, information gain) is selected for the next trial. **(B)** A graphical model illustrating a hierarchical psychophysical observer model that incorporates lapses and "omissions". lapse. On each trial, a latent attention or lapse variable a_t is drawn from a Bernoulli distribution with parameter λ, to determine whether the observer attends to the stimulus $\mathbf{x}_t$ on that trial or lapses. With probability $1-\lambda$, and the observer attends to the stimulus ($a_t = 0$), and the response y_t is drawn from a multinomial logistic regression model, where the probability of choosing option i is proportional to $\exp(\mathbf{w}_i^\top \mathbf{x}_t)$. With probability λ, the observer lapses ($a_t = 1$) and selects a choice from a (stimulus-independent) response distribution governed by parameter vector $\mathbf{u}$. So-called "omission" trials, in which the observer does not select one of the valid response options, are modeled with an additional response category $y_t = k$.

centration or memory (Kuss et al., 2005; Wichmann & Hill, 2001). Although it is widely understood among experimental researchers that both effects can be significant in real animal behavior, these are often ignored in analysis, and in particular, are not considered by previous methods for adaptive stimulus selection. Here we incorporate these two phenomena explicitly, as explained in more details in Section Psychometric Function Model.

As the model complexity grows by adding extra features, on the other hand, the increasing challenge is to infer the model parameters efficiently (in finite computation time), flexibly (under small-data situations, and/or with non-concave models), and accurately. Our second contribution is to develop efficient inference methods that are fast enough for real-time closed-loop experiments. We discuss two methods for posterior inference, one based on a Gaussian approximation of the posterior and another based on MCMC sampling, in Section Posterior inference.

Our work therefore combines a more realistic model of the psychometric function and efficient methods for posterior inference and evaluation of an information-theoretic utility function. We describe two different algorithms for adaptive stimulus selection Section Adaptive Stimulus Selection Methods, one based on a Gaussian approximation to the posterior and a second based on MCMC sampling. Finally, in Results, we apply our algorithms to real data in simulated closed-loop experiments. We show that our methods confer a substantial reduction in the number of trials required to estimate multi-alternative psychophysical functions, and discuss extensions applicable to experiments with multi-dimensional stimuli.

Psychometric Function Model

Here we develop a flexible model of psychometric function (PF) for describing realistic decision-making behavior, starting with a classical multinomial logistic (MNL) model (Glonek & McCullagh, 1995). We show how omission can be naturally incorporated into the framework with multiple alternatives. We then develop a hierarchical extension of the model that incorporates lapses (see Fig. 1B).

Multinomial logistic model. We consider the setting where the observer is presented with a stimulus $\mathbf{x} \in \mathbb{R}^d$ and selects a response $y \in \{1, \dots k\}$ from one of k discrete choices on each trial. We will assume the stimulus is represented internally by some (possibly non-linear) feature vector $\boldsymbol{\phi}(\mathbf{x})$, which we will write simply as $\boldsymbol{\phi}$ for notational simplicity.

In the multinomial logistic model, the probability p_i of each possible outcome $i \in \{1, \cdots, k\}$ is determined by the dot product between the feature $\boldsymbol{\phi}$ and a vector of weights $\mathbf{w}_i$ according to:

$$p_i = \frac{\exp(\mathbf{w}_i^\top \boldsymbol{\phi})}{\sum_{j=1}^k \exp(\mathbf{w}_j^\top \boldsymbol{\phi})}, \tag{1}$$

where the denominator ensures that these probabilities sum to 1, $\sum_{i=1}^k p_i = 1$. The function from stimulus to a probability vector over choices, $\mathbf{x} \longmapsto (p_1, \dots p_k)$, is the psychometric function, and the set of weights $\{\mathbf{w}_i\}_{i=1}^k$ are its parameters. Note that the model is over-parameterized when written this way, since the requirement that probabilities sum to 1 removes one degree of freedom from the probaility vector. Thus, we can without loss of generality fix one of the weight vectors to zero, for example $\mathbf{w}_k = \mathbf{0}$, so that the denominator in (eq. 1) becomes $z = 1 + \sum_{j=1}^k \exp(\mathbf{w}_j^\top \boldsymbol{\phi})$ and $p_k = 1/z$.

We consider the feature vector $\boldsymbol{\phi}$ to be a known function of the stimulus $\mathbf{x}$, even when the dependence is not written explicitly. For example, we can consider a simple form of feature embedding, $\boldsymbol{\phi}(\mathbf{x}) = [1, \mathbf{x}^\top]^\top$, corresponding to a linear function of the stimulus plus an offset. In this case, the weights for the i'th choice would correpond to $\mathbf{w}_i = [b_i, \mathbf{a}_i^\top]^\top$, where b_i is the offset or bias for the i'th choice, and $\mathbf{a}_i$ are the linear weights governing sensitivity to $\mathbf{x}$. The resulting choice probability has the familiar form, $p_i \propto \exp(b_i + \mathbf{a}_i^\top \mathbf{x})$. Nonlinear stimulus dependencies can be incorporated by including nonlinear functions of $\mathbf{x}$ in the feature vector $\boldsymbol{\phi}(\mathbf{x})$ (Knoblauch & Maloney, 2008; Murray, 2011; Neri & Heeger, 2002).

It is useful to always work with a normalized stimulus space, in which the mean of each stimulus component x_α over the stimulus space is $\langle x_\alpha \rangle = 0$, and the standard deviation $\mathrm{std}(x_\alpha) = 1$. This normalization ensures that the values of the weight parameters are defined in more interpretable ways. The zero-mean condition ensures that the bias b is the expectation value of log probability over all possible stimuli. The unit-variance condition means that the effect of moving a certain distance along one dimension of the weight space is comparable to the moving the same distance in another dimension, again averaged over all possible stimuli. In other words, we are justified to use the same unit along all dimensions of the weight space.

Modeling omission as an additional category. Even in "binary" tasks with only two possible choices per trial, there is often an implicit third choice, which is to make no response, make an illegal response, or interrupt the trial at some point before the response period. For example, animals are often required to maintain an eye position or a nose poke, or wait for a "go" cue before reporting a choice. Trials on which the animal fails to obey these instructions, referred to as "violations" or "omissions", and are typically discarded from analysis. However, such trials have clear relevance to the quantitative study of psychophysical behavior, and may reflect aspects of motivation or attentional state that are worth studying in their own right. Luckily, the multinomial logistic model provides a natural framework for incorporating omission or no-response trials.

Here we model omissions explicitly as one of the possible choices the observer can choose. Because the multinomial logistic model has a flexible number of choices, this is as simple as adding an extra or $(k+1)$'st choice to the model. One can even extend the model to consider different kinds of omissions, e.g., allowing choice $k+1$ to reflect fixation period violations and choice $k+2$ to reflect failure to report a choice during the response window. Henceforth, we will simply let k reflect the total number of choices, including omission, as illustrated in Fig. 1B.

Modeling lapse with a mixture model. Another important feature of real psychophysical observers is the tendency to occasionally make errors that are independent of the stimulus. Such errors, commonly known as "lapses" in the psychophysical literature, may reflect lapses in attention or memory of the response categories, or "button-press errors" in executing an intended motor response. Lapses are most easily identified by errors on "easy" trials, that is, trials that should be performed perfectly if the observer were paying attention.

Although lapse rates are supposed to be small enough in a well-performed psychometric experiment (Carandini & Churchland, 2013), in reality they may be substantial depending on the type of experiment being performed, especially in non-primates or in more complicated tasks. Lapses affect the psychometric function by causing it to saturate above 0 and below 1, so that "perfect" performance is never achieved even for the easiest trials. Failure to incorporate lapses into the PF model may therefore bias estimates of sensitivity, as quantified by PF slope or threshold (Prins, 2012; Wichmann & Hill, 2001).

To model lapses, we use a mixture model that treats the observer's choice on each trial as coming from one of two probability distributions: a stimulus-dependent distribution (governed by the multinomial logistic model) and stimulus-independent distribution (reflecting a fixed probability of choosing any option when "lapsing", or ignoring the stimulus). Simpler versions of such mixture model have been proposed previously (Kuss et al., 2005).

Fig. 1B shows a schematic of the resulting model. On each trial, a Bernoulli random variable $a \sim \mathrm{Ber}(\lambda)$ governs whether the observer lapses: with probability λ and the observer lapses (i.e., ignores the stimulus), and with probability $1-\lambda$, and the observer attends to the stimulus. If the observer lapses ($a = 1$), the response is drawn according to fixed probability distribution $(c_1, \ldots, c_k)$ governing the probability of selecting options 1 to k, where $\sum c_i = 1$. If the observer does not lapse ($a = 0$), the observer selects a response according to the multinomial logistic model. Under this model, the conditional probability of choosing option i given the stimulus can be written:

$$p_i = (1-\lambda)q_i + \lambda c_i, \qquad q_i = \frac{\exp(\mathbf{w}_i^\top \phi)}{\sum_j \exp(\mathbf{w}_j^\top \phi)} \tag{2}$$

where q_i is the lapse-free probability probability under the classical MNL model (eq. 1).

It is convenient to re-parameterize this model so that λc_i, the conditional probability of choosing the i'th option due to a lapse, is written

$$\lambda c_i = \frac{\exp(u_i)}{1 + \sum_j \exp(u_j)}, \tag{3}$$

where each auxiliary lapse parameter u_i is proportional to the log probability of choosing option i due to lapse. The lapse-conditional probabilities of each choice, c_i, and the total lapse probability, λ, are respectively

$$c_i = \frac{\exp(u_i)}{\sum_j \exp(u_j)}, \qquad \lambda = \sum_i \frac{\exp(u_i)}{1 + \sum_j \exp(u_j)}. \tag{4}$$

Because each u_i lives on the entire real line, fitting can be carried out with unconstrained optimization methods, although adding reasonable constraints may improve performance in some cases. The full parameter vector of the resulting model is $\boldsymbol{\theta} = [\mathbf{w}^\top, \mathbf{u}^\top]^\top$, which includes k additional lapse parameters $\mathbf{u} = \{u_1, \cdots, u_k\}$. Note that in some cases it might be desirable to assume lapse choices obey a uniform distribution, where the probability of each option is $c_i = 1/k$. For this simplified "uniform-lapse" model we need only a single lapse parameter u.

Our model provides a general and practical parametrization of tuning curves with lapses. Although previous work has considered the problem of modeling lapses in psychophysical experiments, most assumed the the simplified uniform-lapse model where all options are equally likely during lapses. Earlier approaches have often assumed either that the lapse probability was known a priori (Kontsevich & Tyler, 1999), or was fit by a grid search over a small set of candidate values (Wichmann & Hill, 2001). We instead take a Bayesian approach to inferring lapse parameters, following previous work from (Kuss et al., 2005; Prins, 2012). Our parameterization (eq. 3) has the advantage that the there is no need to constrain the support of the lapse parameters u_i. These parameters' relationship to lapse probabilities c_i takes the same ("softmax") functional form as the multinomial logistic model, placing both sets of parameters on an equal footing.

Posterior inference

Bayesian methods for adaptive stimulus selection require the posterior distribution over model parameters given the data observed so far in an experiment. The posterior distribution results from the combination of two ingredients: a prior distribution $p(\boldsymbol{\theta})$, which captures prior uncertainty about the model parameters $\boldsymbol{\theta}$, and a likelihood function $p(\{y_s\}|\{\mathbf{x}_s\}, \boldsymbol{\theta})$, which captures information about the parameters from the data $\{(\mathbf{x}_s, y_s)\}$, $s = 1, \ldots, t$, consisting of stimulus-response pairs observed up to the current time bin t.

Unfortunately, the posterior distribution for our model has no analytic form. We therefore describe two methods for approximate posterior inference: one relying on a Gaussian approximation to the posterior, known as the Laplace approximation, and a second one based on MCMC sampling.

Prior. The prior distribution specifies our beliefs about model parameters before we have collected any data, and serves to regularize estimates obtained from small amounts of data, e.g., by shrinking estimated weights toward zero. Typically we want the prior to be weak enough that the likelihood dominates the posterior for reasonable-sized datasets. However, the choice of prior is especially important in adaptive stimulus selection settings because it determines the effective volume of the search space (Park & Pillow, 2012; Park, Weller, Horwitz, & Pillow, 2014). For example, if the weights are known to exhibit smoothness, then a correlated or smoothness-inducing prior can improve the performance of adaptive stimulus selection because the effective size (or entropy) of the parameter space is much smaller than under an independent prior (Park & Pillow, 2012).

In this study, we use a generic independent, zero-mean Gaussian prior over the weight vectors

$$p(\mathbf{w}_i) = \mathcal{N}(\mathbf{0}, \sigma^2 I), \tag{5}$$

for all $i \in (1, \ldots k)$, with a fixed standard deviation σ. This choice of prior is appropriate when the regressors $\{\mathbf{x}\}$ are standardized, since any single weight can take values that allow for a range of psychometric function shapes along that axis, from flat ($w = 0$) to steeply decreasing ($w = -2\sigma$) or increasing ($w = +2\sigma$). We used $\sigma = 3$ in the simulated experiments in Results. For the lapse parameters $\{u_i\}$, we used a uniform prior over the range $[\log(0.001), 0]$, so that each lapse probability λc_i is bounded between 0.001 and $1/2$. We set the lower range constraint below $1/N$, where $N = 100$ is the number of observed trials in our simulations, since we cannot reasonably infer lapse probabilities with precision finer than $1/N$. The upper range constraint gives maximal lapse probabilities of $1/(k+1)$ if all u_i take on the maximal value of 0.

Psychometric function likelihood. The likelihood is the conditional probability of the data as a function of the model parameters. Although we have thus far considered the response variable y to be a scalar taking values in the set $\{1, \ldots, k\}$, it is more convenient to use a so-called "one-hot" representation, in which the response variable $\mathbf{y}$ for each trial is a length-k vector with one 1 and k zeros, where the position of the 1 in this vector indicates the category chosen. For example, in a task with four possible options per trial, a response vector $\mathbf{y} = [0\ 0\ 1\ 0]$ indicates a trial on which the observer selected the third option.

With this parametrization, the log-likelihood function for a single trial can be written

$$\log p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}) = \sum_i y_i \log p_i(\mathbf{x}, \boldsymbol{\theta}) = \mathbf{y}^\top \log \mathbf{p}(\mathbf{x}, \boldsymbol{\theta}), \tag{6}$$

where $p_i(\mathbf{x}, \boldsymbol{\theta})$ denotes the probability $p(y_i = 1|\mathbf{x}, \boldsymbol{\theta})$ under the model (eq. 1), and $\mathbf{p}(\mathbf{x}, \boldsymbol{\theta}) \equiv [p_1(\mathbf{x}, \boldsymbol{\theta}), \ldots, p_k(\mathbf{x}, \boldsymbol{\theta})]^\top$ denotes the vector of probabilities for a single trial.

In the classical (lapse-free) multinomial logistic model, where $\boldsymbol{\theta} = \{\mathbf{w}_i\}$, the log likelihood is a concave function of $\boldsymbol{\theta}$, which guarantees that numerical optimization of the log-likelihood will find a global optimum. With a finite lapse rate, however, the log likelihood is no longer provably concave. (See Appendix A).

Posterior distribution. The log-posterior can be written as the sum of log-prior and log-likelihood summed over trials, plus a constant:

$$\log p(\boldsymbol{\theta}|\mathcal{D}_t) = \log p(\boldsymbol{\theta}) + \sum_{s=1}^{t} \log p(\mathbf{y}_s|\mathbf{x}_s, \boldsymbol{\theta}) + c, \qquad (7)$$

where $\mathcal{D}_t \equiv \{\mathbf{x}_s, y_s\}_{s=1}^{t}$ denotes the accumulated data up to trial t and $c = -\log\left(\int p(\boldsymbol{\theta}) \prod_s p(\mathbf{y}_s|\mathbf{x}_s) d\boldsymbol{\theta}\right)$ is a normalization constant that does not depend on the parameters θ. Because this constant has no tractable analytic form, we rely on two alternate methods for obtaining a normalized posterior distribution.

Inference via Laplace approximation. The Laplace approximation is a well-known Gaussian approximation to the posterior distribution, which can be derived from a second-order Tayler series approximation to the log-posterior around its mode (Bishop, 2006).

Computing the Laplace approximation involves a two-step procedure. The first step is to perform a numerical optimization of $\log p(\boldsymbol{\theta}|\mathcal{D}_t)$ to find the posterior mode, or maximum a posterior (MAP) estimate of θ. This vector, given by

$$\hat{\boldsymbol{\theta}}_t = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \log p(\boldsymbol{\theta}) + \sum_{s=1}^{t} \log p(\mathbf{y}_s|\mathbf{x}_s, \theta), \qquad (8)$$

provides the mean of the Laplace approximation. Because we can explicitly provide the gradient and Hessian of the log likelihood (see Appendix A) and log-prior, this optimization can be carried efficiently via Newton-Raphson or trust region methods.

The second step is to compute the second derivative (the Hessian matrix) of the log-posterior at the mode, which provides the inverse covariance of the Gaussian. This gives us a local Gaussian approximation of the posterior, centered at the posterior mode:

$$p(\boldsymbol{\theta}|\mathcal{D}_t) \approx \mathcal{N}(\hat{\boldsymbol{\theta}}_t, C_t), \qquad (9)$$

where covariance $C_t = -H_t^{-1}$ is the inverse Hessian of the log posterior, $H_t(i,j) = \partial^2(\log p(\boldsymbol{\theta}|\mathcal{D}_t)/(\partial\boldsymbol{\theta}_i \partial\boldsymbol{\theta}_j)$, evaluated at $\hat{\boldsymbol{\theta}}_t$.

Note that when the log-posterior is concave (i.e., when the model does *not* contain lapse), numerical optimization is guaranteed to find a global maximum of the posterior. Log-concavity also strengthens the rationale for using the Laplace approximation, since the true and approximate posterior are both log-concave densities centered on the true mode (Paninski et al., 2010; Pillow, Ahmadian, & Paninski, 2011). However, when the model incorporates lapses, these guarantees no longer apply, motivating the use of alternate methods for approximating the posterior.

Inference via MCMC sampling. A second approach to inference is to generate samples from the posterior distribution over the parameters via Markov Chain Monte Carlo (MCMC) sampling. Sampling-based methods are typically more computationally intensive than the Laplace approximation, but may be warranted when the posterior is not provably log-concave (as is the case when lapse rates are non-zero) and therefore not well approximated by a single Gaussian.

The basic idea in MCMC sampling is to set up an easy-to-sample Markov Chain that has the posterior as its stationary distribution. Sampling from this chain produces a dependent sequence of posterior samples: $\{\boldsymbol{\theta}_m\} \sim p(\boldsymbol{\theta}|\mathcal{D}_t)$, which can be used to evaluate posterior expectations via Monte Carlo integrals:

$$\mathbb{E}[f(\boldsymbol{\theta})] \approx \frac{1}{M} \sum_{m=1}^{M} f(\boldsymbol{\theta}_m), \qquad (10)$$

for any function $f(\boldsymbol{\theta})$. The mean of the posterior is obtained from setting $f(\boldsymbol{\theta}) = \boldsymbol{\theta}$, although for adaptive stimulus selection we will be interested in the full shape of the posterior.

The Metropolis-Hastings (MH) algorithm is perhaps the simplest and most widely-used MCMC sampling method (Metropolis, Rosenbluth, Rosenbluth, Teller, & Teller, 1953). It generates samples via a proposal distribution centered on the current sample (see Appendix B). The choice of proposal distribution is critical to the efficiency of the MH algorithm, since this governs the rate of “mixing”, or the the number of Markov Chain samples required to obtain independent samples from the posterior distribution (Rosenthal, 2011). Faster mixing implies that fewer samples M are re-

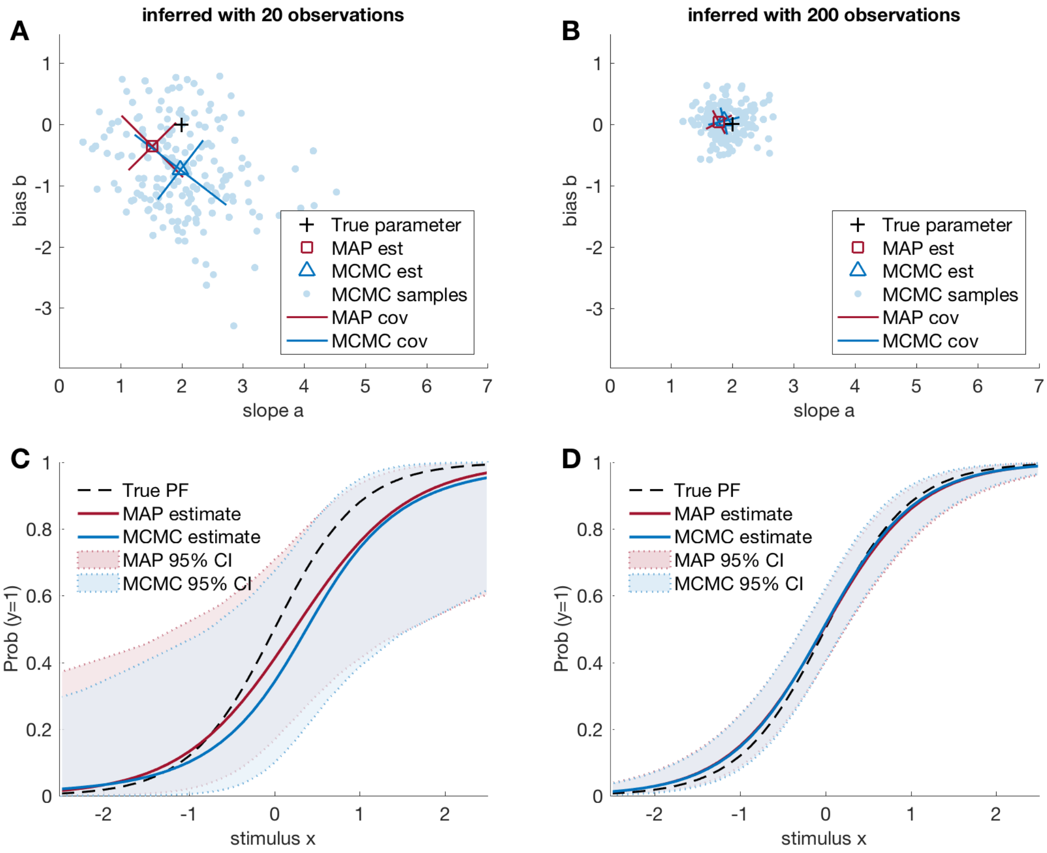

Figure 2: **Inferring the psychometric function**. Example of a psychometric problem, with a lapse-free binomial logistic model $f(v) = e^v/(1+e^v)$. Given a 1D stimulus, a response were drawn from a "true" model $P(y = 1) = f(b + ax)$ with two parameters, slope $a = 2$ and bias $b = 0$. **(A-B)** Viewing on the parameter space, the posterior distributions become sharper (and closer to the true parameter values) as the dataset size N increases. Shown at a small **(A)** $N = 20$ and a large **(B)** $N = 200$. For the MAP estimate, the mode of the distribution is marked with a square, and the two standard deviations ("widths") of its Gaussian approximation are shown with bars. For the MCMC sampling method, all $M = 500$ samples of the chain are shown in dots, the sample mean with a triangle, and the widths with the bars. The widths are the standard deviations along the principal directions of the sampled posterior (eigenvectors of the covariance matrix; not necessary aligned with the $a - b$ axes). **(C-D)** The accuracy of the estimated PF improves with the number of observations N, using either of the two posterior inference methods (MAP-based and sampling-based). Shown at a small **(C)** $N = 20$ and a large **(D)** $N = 200$. The two methods are highly consistent in this simple case, especially when N is large enough.

quired to obtain an accurate approximation to the posterior.

Here we propose a semi-adaptive Metropolis-Hastings algorithm, developed specifically for the current context of sequential learning. Our approach is based on an established observation that the optimal width of the proposal distribution should be proportional to the typical length scale of the distribution being sampled (Gelman, Roberts, & Gilks, 1996; Roberts, Gelman, & Gilks, 1997). Our algorithm is motivated by the adaptive Metropolis algorithm (Haario, Saksman, & Tamminen, 2001), where the proposal distribution is updated at each proposal within a single chain; here we do not adapt the proposal within chains, but rather after each trial. Specifically, we set the covariance of a Gaussian proposal distribution to be proportional to the covariance of the samples from the previous trial, using the scaling factor of Haario et al. (2001). See Appendix B for details. The adaptive algorithm takes advantage of the fact that the posterior cannot change too much between trials, since it changes only by a single-trial likelihood term on each trial.

Adaptive Stimulus Selection Methods

As data are collected during the experiment, the posterior distribution becomes narrower due to the fact that each trial carries some additional information about the model parameters. (See Fig. 2.) This narrowing of the posterior is directly related to information gain. A stimulus that produces no expected narrowing of the posterior is, by definition, uninformative about the parameters. On the other hand, a stimulus that (on average) produces a large change in the current posterior is an informative stimulus. Selecting informative stimuli will reduce the number of stimuli required to obtain a narrow posterior, which is the essence of adaptive stimulus selection methods. In this section, we introduce a precise measure of information gain between a stimulus and the model parameters, and propose an algorithm for selecting stimuli to maximize it.

Infomax criterion for stimulus selection. At each trial, we present a stimulus $\mathbf{x}$ and observe the outcome $\mathbf{y}$. After t trials, the expected gain in information from a stimulus $\mathbf{x}$ is equal to the mutual information between $\mathbf{y}$ and the model parameters $\boldsymbol{\theta}$, given the data $\mathcal{D}_t$ observed so far in the experiment. We denote this conditional mutual information:

$$I_t(\boldsymbol{\theta};\mathbf{y}|\mathbf{x}) = \iint d\boldsymbol{\theta}\, d\mathbf{y}\; p(\boldsymbol{\theta},\mathbf{y}|\mathbf{x},\mathcal{D}_t)\log\frac{p(\boldsymbol{\theta},\mathbf{y}|\mathbf{x},\mathcal{D}_t)}{p(\boldsymbol{\theta}|\mathcal{D}_t)p(\mathbf{y}|\mathbf{x},\mathcal{D}_t)}, \quad (11)$$

where $p(\boldsymbol{\theta},\mathbf{y}|\mathbf{x},\mathcal{D}_t)$ is the joint distribution of $\boldsymbol{\theta}$ and $\mathbf{y}$ given a stimulus $\mathbf{x}$ and dataset $\mathcal{D}_t$, the term $p(\boldsymbol{\theta}|\mathcal{D}_t)$ is the current posterior distribution over the parameters from previous trials, and $p(\mathbf{y}|\mathbf{x},\mathcal{D}_t) = \int d\boldsymbol{\theta}\, p(\mathbf{y}|\mathbf{x},\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{D}_t)$ is known as the posterior-predictive distribution of $\mathbf{y}$ given $\mathbf{x}$.

It is useful to note that the mutual information can equivalently be written in two other ways involving Shannon entropy. The first is given by:

$$I_t(\boldsymbol{\theta};\mathbf{y}|\mathbf{x}) = H_t(\boldsymbol{\theta}) - H_t(\boldsymbol{\theta}|\mathbf{y};\mathbf{x}) \quad (12)$$

where the first term is the entropy of the posterior at time t,

$$H_t(\boldsymbol{\theta}) = -\int d\boldsymbol{\theta}\, p(\boldsymbol{\theta}|\mathcal{D}_t)\log p(\boldsymbol{\theta}|\mathcal{D}_t), \quad (13)$$

and the second is the conditional entropy of $\boldsymbol{\theta}$ given $\mathbf{y}$,

$$\begin{aligned} H_t(\boldsymbol{\theta}|\mathbf{y};\mathbf{x}) &= -\mathbb{E}_{\boldsymbol{\theta},\mathbf{y}}\Big[\log p(\boldsymbol{\theta}|\mathbf{y},\mathbf{x},\mathcal{D}_t)\Big] \\ &= -\iint d\boldsymbol{\theta}\, d\mathbf{y}\; p(\boldsymbol{\theta},\mathbf{y}|\mathbf{x},\mathcal{D}_t)\log p(\boldsymbol{\theta}|\mathbf{y},\mathbf{x},\mathcal{D}_t), \quad (14) \end{aligned}$$

which is the entropy of the updated posterior *after* having observed $\mathbf{x}$ and $\mathbf{y}$, averaged over draws of $\mathbf{y}$ from the posterior predictive distribution. Written this way, the mutual information can be seen as the expected reduction in posterior entropy from a new stimulus-response pair. Moreover, the first term, $H_t(\theta)$, is independent of the stimulus and response on the current trial, so infomax stimulus selection is equivalent to picking the stimulus that minimizes the expected posterior entropy $H_t(\boldsymbol{\theta}|\mathbf{y};\mathbf{x})$.

A second equivalent expression for the mutual information, which will prove useful for our sampling-based method, is:

$$I_t(\boldsymbol{\theta};\mathbf{y}|\mathbf{x}) = H_t(\mathbf{y};\mathbf{x}) - H_t(\mathbf{y}|\boldsymbol{\theta};\mathbf{x}), \quad (15)$$

which is the difference between the marginal entropy of the response distribution conditioned on $\mathbf{x}$,

$$H_t(\mathbf{y};\mathbf{x}) = -\int d\mathbf{y}\, p(\mathbf{y}|\mathbf{x},\mathcal{D}_t)\log p(\mathbf{y}|\mathbf{x},\mathcal{D}_t) \qquad (16)$$

and the conditional entropy of the response $\mathbf{y}$ given $\boldsymbol{\theta}$, conditioned on the stimulus:

$$H_t(\mathbf{y}|\boldsymbol{\theta};\mathbf{x}) = -\iint d\mathbf{y}\, d\boldsymbol{\theta}\, p(\boldsymbol{\theta},\mathbf{y}|\mathbf{x},\mathcal{D}_t)\log p(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}). \qquad (17)$$

This formulation shows the mutual information to be equal to the difference between the entropy of the marginal distribution of $\mathbf{y}$ conditioned on $\mathbf{x}$ (with $\boldsymbol{\theta}$ integrated out) and the average entropy of $\mathbf{y}$ given $\mathbf{x}$ and $\boldsymbol{\theta}$, averaged over the posterior distribution of $\boldsymbol{\theta}$.

In a sequential setting where t is the latest trial and $t+1$ is the upcoming one, the optimal stimulus is the information-maximizing ("infomax") solution:

$$\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}} I_t(\boldsymbol{\theta};\mathbf{y}|\mathbf{x}). \qquad (18)$$

Fig. 3 shows an example of a simulated experiment where the stimulus was selected adaptively following the infomax criterion.

Selecting the optimal stimulus thus requires maximizing the mutual information over the set of all possible stimuli $\{\mathbf{x}\}$. Since each evaluation of the mutual information involves a high-dimensional integral over parameter space and response space, this is a highly computationally demanding task. In the next sections, we present two algorithms for efficient infomax stimulus selection based on each of the two approximate inference methods described previously.

Infomax with Laplace approximation. Calculation of the mutual information is greatly simplified by a Gaussian approximation of the posterior. The entropy of a Gaussian distribution with covariance C is equal to $\frac{1}{2}\log|C|$ up to a constant factor. If we expand the mutual information as in (eq. 12), and recall that we need only minimize the expected posterior entropy after observing the response, the optimal stimulus for time-step $t+1$ is given by:

$$\mathbf{x}^*_{t+1} = \underset{\mathbf{x}}{\operatorname{argmin}} \int dy\, p(\mathbf{y}|\mathbf{x},\mathcal{D}_t)\log|\tilde{C}(\mathbf{x},\mathbf{y})|, \qquad (19)$$

where $\tilde{C}(\mathbf{x},\mathbf{y})$ is the covariance of the updated (Gaussian) posterior after observing stimulus-response pair $(\mathbf{x},\mathbf{y})$. To evaluate the updated covariance $\tilde{C}(\mathbf{x},\mathbf{y})$ under the Laplace approximation, we would need to numerically optimize the posterior for $\boldsymbol{\theta}$ for each possible resonse $\mathbf{y}$, for any candidate stimulus $\mathbf{x}$, which would be computationally infeasible. We therefore use a fast approximate method for obtaining a closed-form update for $\tilde{C}(\mathbf{x},\mathbf{y})$ from the current posterior covariance C_t, following an approach developed in Lewi et al. (2009). (See Appendix C for details.)

Once we have $\log|\tilde{C}(\mathbf{x},\mathbf{y})|$ for each given stimulus-observation pair, we numerically sum this over a set of discrete counts $\mathbf{y}$ that are likely under the posterior-predictive distribution. This is done in two steps, by separating the integral in (eq. 19) as:

$$\begin{aligned}&\int d\mathbf{y}\, p(\mathbf{y}|\mathbf{x},\mathcal{D}_t)\, \log|\tilde{C}(\mathbf{x},\mathbf{y})| \\ &\quad = \int d\boldsymbol{\theta}_t\, p(\boldsymbol{\theta}_t|\mathcal{D}_t) \int d\mathbf{y}\, p(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}_t)\, \log|\tilde{C}(\mathbf{x},\mathbf{y})|. \qquad (20)\end{aligned}$$

Note that the outer integral is over the current posterior $p(\boldsymbol{\theta}_t|\mathcal{D}_t) \approx \mathcal{N}(\hat{\boldsymbol{\theta}}_t, C_t)$, which is to be distinguished from the future posterior $p(\boldsymbol{\theta}|\mathbf{y},\mathbf{x},\mathcal{D}_t) \approx \mathcal{N}(\tilde{\boldsymbol{\theta}}(\mathbf{x},\mathbf{y}),\tilde{C}(\mathbf{x},\mathbf{y}))$ whose entropy we are trying to minimize. Whereas the inner integral is simply a weighted sum over the set of outcomes $\mathbf{y}$, the outer integral over the parameter $\boldsymbol{\theta}$ is in general challenging, especially when the parameter space is high-dimensional. In the case of the standard multinomial logistic model that does not include lapse, we can exploit the linear structure of model to reduce this to a lower-dimensional integral over the space of the linear predictor, which we evaluate numerically using Gauss-Hermite quadrature (Heiss & Winschel, 2008). (This integral is 1D for classic logistic regression, and (k-1)-dimensional for multinomial logistic regression with k classes; see Appendix C for details.)

When the model incorporates lapses, the full parameter vector $\boldsymbol{\theta} = [\mathbf{w}^\top, \mathbf{u}^\top]$ includes the lapse parameters in addition to the weights $\mathbf{w}$. In this case, our method with Laplace approximation may suffer from reduced accuracy due to the fact that the posterior (which is not provably log-concave in this setting) may be less closely approximated by a Gaussian. For tractability, we choose to

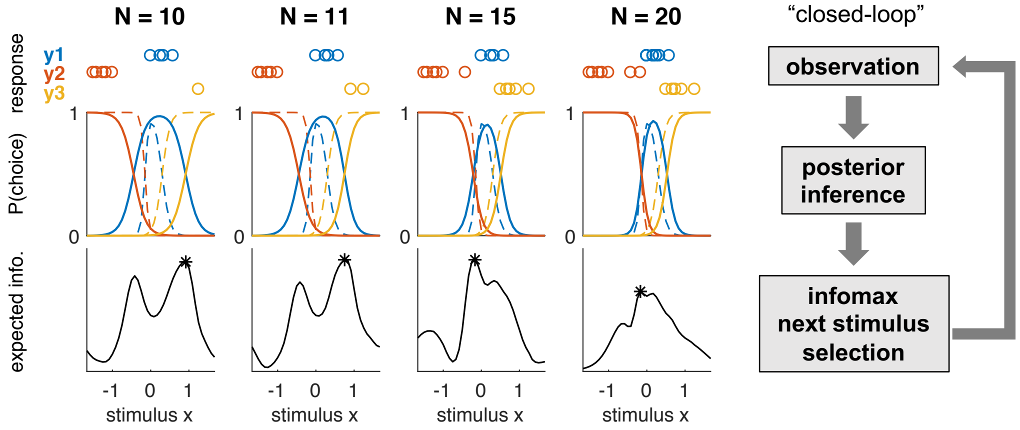

Figure 3: **Example of infomax adaptive stimulus selection**, simulated with a three-alternatives lapse-free model on 1D stimulus. The figure shows how given a small set of data (the stimulus-response pairs shown in top row), the PFs are estimated based on the accumulated data (middle row), and the next stimulus is chosen to maximize the expected information gain (bottom row). Each column shows the instance after the N observations in a single adaptive stimulus selection sequence, for $N = 10, 11, 15$ and 20 respectively. In the middle row, the estimated PFs (solid lines) quickly approach the true PFs (dashed lines) through the adaptive and optimal selection of stimuli. This example was generated using the Laplace approximation based algorithm, with an independent Gaussian prior over the weights with mean zero and standard deviation $\sigma = 10$.

maximize the *partial* information between the observation and the psychophysical weights, $I(\mathbf{w}; \mathbf{y}|\mathbf{x})$, instead of the full information $I(\boldsymbol{\theta}; \mathbf{y}|\mathbf{x})$. This is also a reasonable approximation in many cases where the stimulus-dependent behavior is the primary focus of the psychometric experiment; the weights $\mathbf{w}$ are of primary interest, while the lapse $\mathbf{u}$ are usually nuisance parameters. The partial covariance $C_{\mathbf{ww}} = -(\partial^2(\log \mathcal{P})/\partial \mathbf{w}^2)^{-1}$ can be used in place of the full covariance $C = -(\partial^2(\log \mathcal{P})/\partial \boldsymbol{\theta}^2)^{-1}$. Because the positive semi-definiteness of this partial covariance is still not guaranteed, it needs to be approximated to the nearest symmetric positive semi-definite matrix when necessary (Higham, 1988). We can show, however, that this partial covariance is asymptotically positive semi-definite in the small lapse limit (Appendix A),

Infomax with MCMC. Sampling-based inference provides an attractive alternative to Laplace's method when the model includes non-zero lapse rates, where the posterior may be less well approximated by a Gaussian. To compute mutual information from samples, it is more convenient to use the expansion given in (eq. 15), so that it is expressed as the expected uncertainty reduction in entropy of the response $\mathbf{y}$, instead of a reduction in the posterior entropy. This will make it straightforward to approximate integrals needed for mutual information by Monte Carlo integrals involving sums over samples.

Given a set of set of posterior samples $\{\boldsymbol{\theta}_m\}$ from $p(\boldsymbol{\theta}|\mathcal{D}_t)$, the posterior distribution at time t, we can evaluate the mutual information using sums over "potential" terms that we denote by

$$L_{jm}(\mathbf{x}) \equiv p(y_j = 1|\mathbf{x}, \boldsymbol{\theta}_m)\,. \qquad (21)$$

This allows us to evaluate the conditional response entropy as

$$H_t(\mathbf{y}|\boldsymbol{\theta}; \mathbf{x}) \approx -\frac{1}{M} \sum_{j,m} L_{jm}(\mathbf{x}) \log L_{jm}(\mathbf{x}), \qquad (22)$$

and the marginal response entropy as

$$H_t(\mathbf{y}; \mathbf{x}) \approx -\sum_j \left(\tfrac{1}{M} \sum_m L_{jm}(\mathbf{x}) \right) \log \left(\tfrac{1}{M} \sum_m L_{jm}(\mathbf{x}) \right), \qquad (23)$$

where we have evaluated the posterior-predictive distribution as

$$p(y_j = 1|\mathbf{x}, \mathcal{D}_t) \approx \frac{1}{M} \sum_m L_{jm}(\mathbf{x}). \qquad (24)$$

Putting together these terms, the mutual information can be evaluated as

$$I_t(\boldsymbol{\theta}; \mathbf{y}|\mathbf{x}) = -\frac{1}{M} \sum_{j,m} L_{jm}(\mathbf{x}) \log \frac{L_{jm}(\mathbf{x})}{\sum_{m'} L_{jm'}(\mathbf{x})/M}, \qquad (25)$$

which is straightforward to evaluate for a set of candidate stimuli $\{\mathbf{x}\}$. The computational cost of this approach is therefore linear in the number of samples, and the primary concern is the cost of obtaining a representative sample from the posterior.

Results

We consider two approaches for testing the performance of our proposed stimulus-selection algorithms, one using simulated data, and a second using an offline analysis of data from real psychophysical experiments.

Simulated experiments. We first tested the performance of our algorithms using simulated data from a fixed psychophysical observer model. In these simulations, a stimulus $\mathbf{x}$ was selected on each trial and the observer's response $\mathbf{y}$ was sampled from a "true" psychometric function, $p_{\text{true}}(\mathbf{y}|\mathbf{x}) = p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}_{\text{true}})$.

We considered psychophysical models defined on a continuous 2-dimensional stimulus space with 4 discrete response alternatives for every trial, corresponding to the problem of estimating the direction of 2D stimulus moving along one of the four cardinal directions (up, down, left, right). We computed expected information gain over a set of discrete stimulus values corresponding to 21×21 square grid (Fig. 4A). The stimulus plane is colored in Fig. 4A, to indicate the most likely response (one of the four alternatives) in each stimulus region. Lapse probabilities λc_i were set to either zero (the "lapse-free" case), or a constant value of 0.05, resulting in a total lapse probability of $\lambda = 0.2$ across the four choices (Fig. 4B). We compared performance of our adaptive algorithms with a method that selected a stimulus uniformly at random from the grid on each trial. We observed that the adaptive methods tended to sample more stimuli near the boundaries between colored regions on the stimulus space (Fig. 4C), which led to more efficient estimates of the PF compared to the uniform stimulus selection approach (Fig. 4D).

For each true model, we compared the performances of four different adaptive methods (Fig. 4E-F), defined by performing inference with MAP or MCMC, and assuming lapse rate to be fixed at zero or including a non-zero lapse parameters. Each of these inference methods was also applied to data selected according to a uniform stimulus selection algorithm. We quantified performance using the mean-squared error (MSE) between the true response probabilities $p_{ij} = p(y = j|\mathbf{x}_i, \boldsymbol{\theta}_{\text{true}})$ and the estimated probabilities $\hat{p}_{ij}$ over the 21×21 grid of stimulus locations $\{\mathbf{x}_i\}$ and the 4 possible responses $\{j\}$. For MAP-based inference, estimated probabilities were given by $\hat{p}_{ij} = p(y = j|\mathbf{x}_i, \hat{\boldsymbol{\theta}}_{\text{MAP}})$. For the MCMC-based inference, probabilities were given by the predictive distribution, evaluated using an average over samples: $\hat{p}_{ij} = \frac{1}{M}\sum_m p(y = j|\mathbf{x}_i, \boldsymbol{\theta}_m)$, where $\{\boldsymbol{\theta}_m\}$ represent samples from the posterior.

When the true model was lapse-free (Fig. 4E), lapse-free and lapse-aware inference methods performed similarly, indicating that there was minimal cost to incorporating parameters governing lapse when lapses were absent. Under all inference methods, infomax stimulus selection outperformed uniform stimulus selection by a substantial margin. For example, infomax algorithms achieved in $50 - 60$ trials the error levels that their uniform-stimulus-selection counterparts required 100 trials to achieve.

By contrast, when the true model had a non-zero lapse rate (Fig. 4F), adaptive stimulus selection algorithms based on the lapse-free model failed to select optimal stimuli, performing even worse than uniform stimulus selection algorithms. This emphasizes the impact of model mismatch in adaptive methods, and the importance of a realistic psychometric model. When lapse-aware models were used for inference, on the other hand, both Laplace-based and MCMC-based adaptive stimulus selection algorithms achieved a significant speedup compared to uniform stimulus selection, while MCMC-based adaptive algorithm performed better. This shows that the MCMC-based infomax stimulus selection method can provide an efficient and robust platform for adaptive experiments with realistic models.

In view of these results, it seems good practice to always use the lapse-aware model, unless the behavior under study is known to be

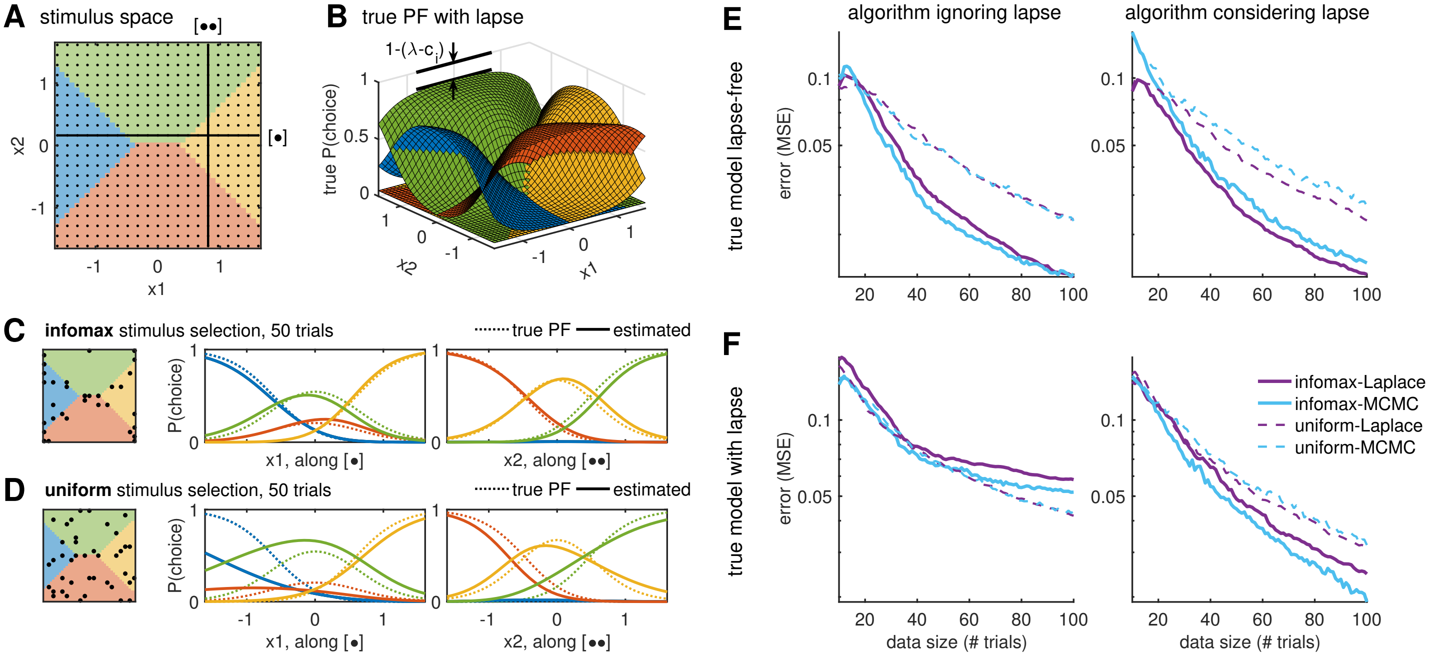

Figure 4: **The simulated experiment**. **(A)** At each trial, a stimulus was selected from a 2D stimulus plane with a 21×21 grid. The two lines, running along x_1 and x_2 respectively, indicate the cross-sections used in **C** and **D** below. Colors indicate the most likely response in the respective stimulus regime, according to the true PF shown in **B**, with a consistent color code. **(B)** Given each stimulus, a simulated response was drawn from a true model with 4 alternatives. Shown here is the model with lapse, characterized by a non-deterministic choice (i.e., the choice probability does not approach 0 or 1) even at an easy stimulus, far from the choice boundaries. **(C-D)** Examples of Laplace-approximation-based inference results after 50 trials, where stimuli was selected either using our adaptive infomax method **(C)** or uniformly **(D)**, as shown on left. In both cases, the true model was lapse-free, and the algorithm assumed that lapse was fixed at zero. The two sets of curves show the cross-sections of the true PF (dotted lines) and the estimated PF (solid lines), along the two lines marked in **A**, after sampling these stimuli. **(E-F)** Error traces from simulated experiments, averaged over 100 runs each. The true model for simulation was either **(E)** lapse-free, or **(F)** with a finite lapse rate of $\lambda = 0.2$, with a uniform lapse scenario $c_i = 1/4$ for each outcome $i = 1, 2, 3, 4$. The algorithm either used the classical MNL model that assumes zero lapse (left column), or our extended model that considers lapse (right column). Performances of adaptive and uniform stimulus selection algorithms are plotted in solid and dashed lines; Laplace-based and MCMC-based algorithms are plotted in purple and cyan. All sampling-based algorithms used the semi-adaptive MCMC with chain length $M = 1000$.

completely lapse-free. The computational cost for incorporating lapses amounts to having k additional parameters to sample, one per each available choice, which is independent from the dimensionality of the stimulus space. When the true behavior had lapses, the MCMC-based adaptive stimulus selection algorithm with the lapse-aware model automatically included "easy" trials, which provide maximal information about lapse probabilities. These easy trials are typically in the periphery of the stimulus space (strong-stimulus regimes, referred to as "asymptotic performance intensity" in Prins (2012)).

Optimal re-ordering of real dataset. A second approach for testing the performance of our methods is to perform an off-line analysis of data from real psychophysical experiments. Here we take an existing dataset and use our methods to re-order the trials so that the most-informative stimuli are selected first. To obtain a re-ordering, we iteratively apply our algorithm to the stimuli shown during the experiment. On each trial, we use our adaptive algorithm to select the optimal stimulus from the set of stimuli $\{\mathbf{x}_i\}$ not yet incorporated into the model. This selection takes place without access to the actual responses $\{\mathbf{y}_i\}$. We then update the posterior using the stimulus $\mathbf{x}_i$ and the response $\mathbf{y}_i$ it actually elicited during the experiment, then proceed to the next trial. We can then ask whether adding the data according to the proposed re-ordering would have led to faster narrowing of the posterior distribution than other orderings.

To perform this analysis, we used a dataset from macaque monkeys performing a four-alternative motion discrimination task (Churchland, Kiani, & Shadlen, 2008). Monkeys were trained to observe a motion stimulus with dots moving in one of the four cardinal directions, and report this direction of motion with an eye movement. The difficulty of the task was controlled by varying the fraction of coherently moving dots on each trial, with the remaining dots appearing randomly (Fig. 5A). Each moving-dot stimulus in this experiment could be represented as a two-dimensional vector, where the direction of the vector is the direction of the mean movement of the dots, and the amplitude of the vector is given by the fraction of coherently moving dots (a number between 0 and 1). Each stimulus presented in the the experiment was aligned with either one of the two cardinal axes of the stimulus plane (Fig. 5B). The PF for this dataset consists of a set of four 2D curves, where each curve specifies the probability of choosing a particular direction as a function of location in the 2D stimulus plane (Fig. 5C).

This monkey dataset contained more than $10,000$ total observations at 29 distinct stimulus conditions, accumulating more than 300 observations per stimulus. This multiplicity of observations per stimulus ensured that the posterior distribution given the full dataset was narrow enough that it could be considered to provide a "ground truth" psychometric function against which the inferences based on the re-ordering experiment could be compared.

The first 100 stimuli selected by the infomax algorithms had noticeably different statistics than the full dataset or its uniform sub-sampling (the first $N = 100$ trials under uniform sampling). On the other hand, the sets of stimuli selected by both MAP-based and MCMC-based infomax algorithms were similar. Fig. 5D shows the histogram of stimulus component along one of the axes, $p(x_2 \,|\, x_1 = 0)$, from the first $N = 100$ trials, averaged over 100 independent runs under each stimulus selection algorithm using the lapse-free model.

Because the true PF was unknown, we compared the performance of each algorithm to an estimate of the PF from the entire dataset. When using the MAP algorithm, the full-dataset PF was given by $p_{ij} = p(y = j|\mathbf{x}_i, \hat{\boldsymbol{\theta}}_{\text{full}})$, evaluated at the MAP estimate of the log posterior, $\hat{\boldsymbol{\theta}}_{\text{full}} = \text{argmax}_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}|\mathcal{D}_{\text{full}})$, given the full dataset $\mathcal{D}_{\text{full}}$. For the MCMC algorithm, the full-dataset PF was computed by $p_{ij} \approx \frac{1}{M}\sum_m p(y = j|\mathbf{x}_i, \boldsymbol{\theta}_m)$, where the MCMC chain $\{\boldsymbol{\theta}_m\} \sim \log p(\boldsymbol{\theta}|\mathcal{D}_{\text{full}})$ sampled the log posterior given the full dataset. The re-ordering test on the monkey dataset showed that our adaptive stimulus sampling algorithms were able to infer the PF to a given accuracy in a smaller number of observations, compared to a uniform sampling algorithm (Fig. 5E-F). In other words, data collection could have been faster with an optimal re-ordering of the experimental procedure.

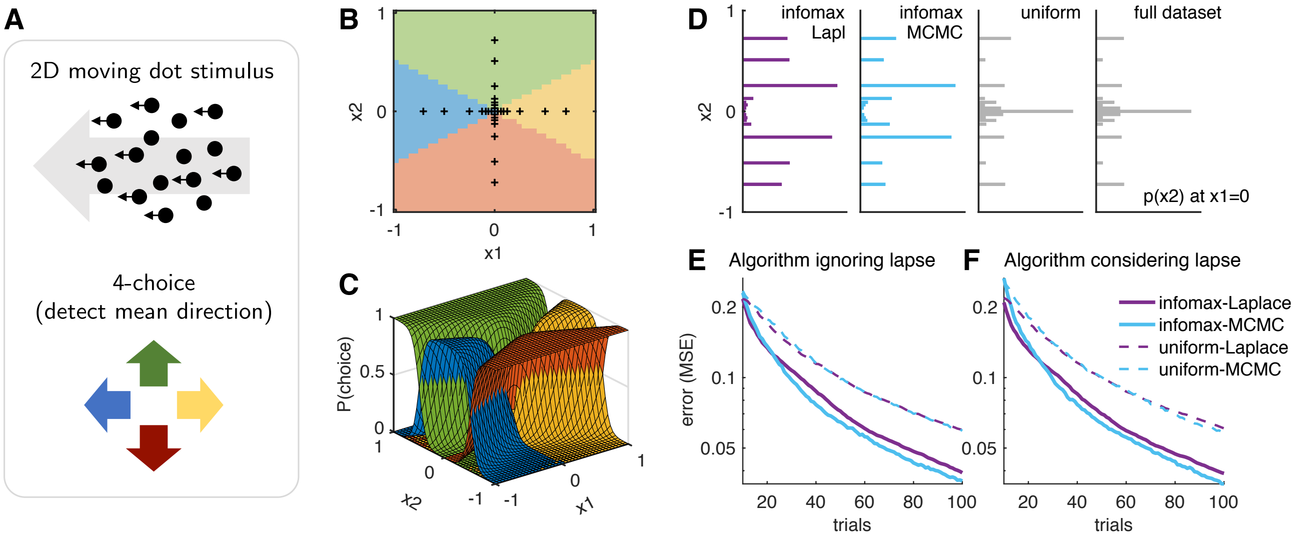

Figure 5: **Optimal re-ordering of a real monkey dataset**. **(A)** The psychometric task consisted of a 2D stimulus presented as moving dots, characterized by a coherence and a mean direction of movement, and a 4-alternative response. The four choices are color coded consistently in **A**-**C** in this figure. **(B)** The axes-only stimulus space of the original dataset, with 15 fixed stimuli along each axis. Colors indicate the most likely response in the respective stimulus regime according to the best estimate of the PF. **(C)** The best estimate of the PF of monkeys in this task, inferred from all observations in the dataset. **(D)** Stimuli selection in the first $N = 100$ trials during the re-ordering experiment, under the inference method that ignores lapse. Shown are histograms of x_2 along one of the axes, $x_1 = 0$, averaged over 100 independent runs in each case. **(E-F)** Error traces under different algorithms, averaged over 100 runs. Both Laplace-based (purple) and MCMC-based (cyan; with $M = 1000$) algorithms achieve significant speedups over uniform sampling. Because the monkeys were almost lapse-free in this task, inference methods that ignore lapse **(E)** and consider lapse **(F)** performed similarly.

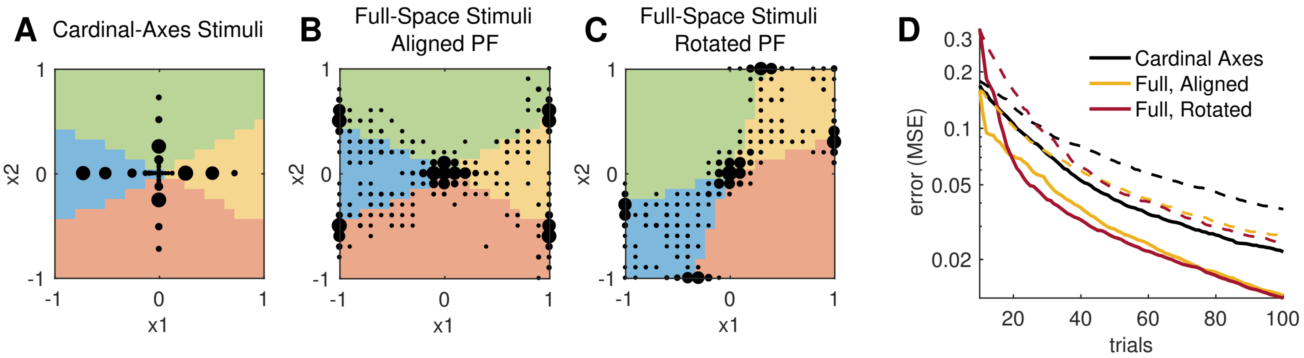

Figure 6: **Design of multi-dimensional stimulus space**. **(A-C)** Three different stimulus space designs were used in a simulated psychometric experiment. Responses were simulated according to fixed lapse-free PFs, matched to our best estimate of the monkey PF (Fig. 5C). Stimuli were selected within the respective stimulus spaces, **(A)** the cardinal-axes design, as in the original experiment; **(B)** full stimulus plane, with the PF aligned to the cardinal axes of the original stimulus space; **(C)** full stimulus plane, with rotated PF. The black dots in **A**-**C** indicate which stimuli were sampled by the Laplace-based infomax algorithm during the first $N = 100$ trials of simulation, where the dot size is proportional to the number of trials in which each stimulus was selected (averaged over 20 independent runs, and excluding the 10 fixed initial stimuli). **(D)** The corresponding error traces, under infomax (solid lines) or uniform (dashed lines) stimulus selection, averaged over 100 runs respectively. Colors indicate the three stimulus space designs, as shown in **A**-**C**.

Exploiting the full stimulus space. In the experimental dataset considered in the previous section, the motion stimuli were restricted to points along the cardinal axes of the 2D stimulus plane (Fig. 5B) (Churchland et al., 2008). In some experimental settings, however, the psychometric functions of interest may lack identifiable axes of alignment or may exhibit asymmetries in shape or orientation. Here we show that in such cases, adaptive stimulus selection methods can benefit from the ability to select points from the full space of possible stimuli.

We performed experiments with a simulated observer governed by the lapse-free psychometric function estimated from the macaque monkey dataset (Fig. 5C). This psychometric function was either aligned to the original stimulus axes (Fig. 6A-B) or rotated counter-clockwise by 45 degrees (Fig. 6C). We tested the performance of adaptive stimulus selection using the Laplace infomax algorithm, with stimuli restricted to points along the cardinal axes (Fig. 6A), or allowed to a grid of points in the full 2D stimulus plane (Fig. 6B-C).

The simulated experiment indeed closely resembled the results of our dataset re-ordering test in terms of the statistics of adaptively selected stimuli (compare Fig. 6A to the purple histogram in Fig. 5D). With the full 2D stimulus space aligned to the cardinal axes, on the other hand, our adaptive infomax algorithm detected and sampled more stimuli near the boundaries between colored regions in the stimulus plane, which were usually not on the cardinal axes (Fig. 6B). Finally, we also observed that this automatic exploitation of the stimulus space was not limited by the lack of alignment between the PF and the stimulus axes; our adaptive infomax algorithm was just as effective in detecting and sampling the boundaries between stimulus regions in the case of the unaligned PF (Fig. 6C).

The error traces in Fig. 6D show that we can infer the PF at a given accuracy in an even fewer number of observations using our adaptive algorithm on the full 2D stimulus plane (orange curves), compared to the cardinal-axes design (black curves). It also confirms that we can infer the PF accurately and effectively with an unaligned stimulus space (red curves), as well as with an aligned stimulus space. For comparison purposes, all errors were calculated over the same 2D stimulus grid, even when the stimulus selection was from the cardinal axes. (This had negligible effects on the resulting error values: compare the black curves in Fig. 6D and the purple curves in Fig. 5E.)

Discussion

We developed effective Bayesian adaptive stimulus selection algorithms for inferring psychometric functions, with an objective of maximizing the expected informativeness of each stimulus. The algorithms select an optimal stimulus adaptively in each trial, based on the posterior distribution of model parameters inferred from the accumulating set of past observations.

We emphasized that in psychometric experiments, especially with animals, it is crucial to use models that can account for the non-ideal yet common behaviors, such as omission (no response; an additional possibility for the outcome) or lapse (resulting in a random, stimulus-independent response). Specifically, we constructed a hierarchical extension of a multinomial logistic (MNL) model that incorporates both omission and lapse. To ensure applicability of the extended model in real-time closed-loop adaptive stimulus selection algorithms, we also developed efficient methods for inferring the posterior distribution of the model parameters, with approximations specifically suited for sequential experiments.

Advantages of adaptive stimulus selection. We observed two important advantages of using Bayesian adaptive stimulus selection methods in psychometric experiments. First, we showed that our adaptive stimulus selection algorithms achieved significant speed-ups in learning time (number of measurements), both on simulated data and in re-ordering test of a real experimental dataset, with and without lapse in the underlying behavior. Importantly, the success of the algorithm depends heavily on the use of the correct model family; for example, adaptive stimulus selection fails when a classical (lapse-ignorant) model was used to measure behavior with a finite lapse rate. Based on the simulation results, it is always a good idea to use the our extended model which can accommodate both lapse-free and finite-lapse systems.

Second, we demonstrated that our adaptive stimulus selection study has implications on the optimization of the experimental designs more generally. Contrary to the conventional practice of accumulating repeated observations at a small set of fixed stimuli, we suggest that the (potentially high-dimensional) stimulus space can be exploited more efficiently using our Bayesian adaptive stimulus selection algorithm. Specifically, the adaptive stimulus selection algorithm can automatically detect the structure of the stimulus space (with respect to the psychometric function) as part of the process. We also showed that there are benefits of using the full stimulus space even when the PF is aligned to the cardinal axes of the stimulus space.

Comparison of the two algorithms. Our adaptive stimulus selection algorithms were developed based on two methods for effective posterior inference: one based on local Gaussian approximation (Laplace approximation) of the posterior, and another based on MCMC sampling. Although the well-studied analytical method based on the Laplace approximation is fast and effective in ideal settings (where log concavity is guaranteed), it may break down with a departure from the ideal model, for example with a finite lapse rate. The sampling-based method is a robust alternative for those realistic situations.

In the case of sampling-based methods, the cost of such flexibility comes in the form of increased computation time; depending on the experimental paradigm, a naive implementation of the sampling method may take too long to run within a single-trial interval. For real-time applications, therefore, it will be an important future direction to further optimize the sampling algorithm. For example, in this work, we developed a semi-adaptive tuning algorithm to efficiently transfer step-size information from the previous trials to the current trial. On the other hand, the computational bottleneck for the Laplace-approximation-based method in this work was the high-dimensional integration in the infomax calculation; a more accurate estimate would require the quadrature to be on a finer grid of support points.

Adaptive designs in psychometric experiments. Finally, we note that a potential limitation of the adaptive stimulus selection framework is the (undesired) possibility that the psychometric function of the observer might adapt to the distribution of stimuli presented during the experiments. If this is the case, the system under measurement would no longer be stationary, nor independent of the experimental design, profoundly altering the problem one should try to solve.

The usual assumption in psychometric experiments is that, although behavior adaptation is the major process in the training phase (Bak, Choi, Akrami, Witten, & Pillow, 2016), already overtrained observers would not change their behavior too quickly, particularly not within the timescale of a psychometric experiment. Under such assumption of stationarity, as pointed out by MacKay (1992), the order of data collection cannot bias the Bayesian inference.

In order to justify the use of adaptive designs, the impact of post-training adaptation will need to be tested experimentally. For example, it was suggested that the inter-trial dependence was non-negligible even in overtrained animals (Fründ, Wichmann, & Macke, 2014); there have been attempts to account for the history dependence by adding regressors on relevant features in a small number of preceding trials, such as the reward outcomes (Bak et al., 2016; Busse et al., 2011; Corrado, Sugrue, Seung, & Newsome, 2005; Lau & Glimcher, 2005), the stimuli (Akrami, Kopec, Diamond, & Brody, 2017) or the full stimulus-response history (Fründ et al., 2014). Whether the adaptive stimulus presentation can have more systematic impacts, on the behavior of trained observers, remains an open question.

Acknowledgements

We thank Anne Churchland for sharing the monkey data. JHB was supported by the Samsung Scholarship for the study at Princeton.

JWP was supported by grants from the McKnight Foundation, Simons Collaboration on the Global Brain (SCGB AWD1004351) and the NSF CAREER Award (IIS-1150186). Computational work was performed using resources at Princeton University and the KIAS Center for Advanced Computing.

References

Akrami, A., Kopec, C. D., Diamond, M. E., & Brody, C. D. (2017). Posterior parietal cortex represents sensory stimulus history and is necessary for its effects on behavior. *bioRxiv*. doi: 10.1101/182246

Bak, J. H., Choi, J. Y., Akrami, A., Witten, I. B., & Pillow, J. W. (2016). Adaptive optimal training of animal behavior. In *Advances in neural information processing systems 29* (pp. 1947–1955).

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer New York.

Busse, L., Ayaz, A., Dhruv, N. T., Katzner, S., Saleem, A. B., Scholvinck, M. L., ... Carandini, M. (2011). The detection of visual contrast in the behaving mouse. *Journal of Neuroscience*, *31*(31), 11351–11361. doi: 10.1523/JNEUROSCI.6689-10.2011

Carandini, M., & Churchland, A. K. (2013). Probing perceptual decisions in rodents. *Nature Neuroscience*, *16*(7), 824–831. doi: 10.1038/nn.3410

Cavagnaro, D. R., Myung, J. I., Pitt, M. A., & Kujala, J. V. (2010). Adaptive design optimization: a mutual information-based approach to model discrimination in cognitive science. *Neural computation*, *22*(4), 887–905. doi: 10.1162/neco.2009.02-09-959

Chaloner, K., & Larntz, K. (1989). Optimal logistic Bayesian design applied to logistic regression experiments. *Journal of Statistical Planning and Inference*, *21*, 191–208. doi: 10.1016/0378-3758(89)90004-9

Chaloner, K., & Verdinelli, I. (1995). Bayesian experimental design: a review. *Statistical Science*, *10*, 273–304.

Churchland, A. K., Kiani, R., & Shadlen, M. N. (2008). Decision-making with multiple alternatives. *Nature Neuroscience*, *11*(6), 693–702. doi: 10.1038/nn.2123

Corrado, G. S., Sugrue, L. P., Seung, H. S., & Newsome, W. T. (2005). Linear-nonlinear-poisson models of primate choice dynamics. *Journal of the Experimental Analysis of Behavior*, *84*(3), 581–617. doi: 10.1901/jeab.2005.23-05

DiMattina, C. (2015). Fast adaptive estimation of multidimensional psychometric functions. *Journal of Vision*, *15*(9), 5. doi: 10.1167/15.9.5

DiMattina, C., & Zhang, K. (2011). Active data collection for efficient estimation and comparison of nonlinear neural models. *Neural Computation*, *23*(9), 2242–2288. doi: 10.1162/NECO_a_00167

Fründ, I., Wichmann, F. A., & Macke, J. H. (2014). Quantifying the effect of intertrial dependence on perceptual decisions. *Journal of vision*, *14*(7), 1–16. doi: 10.1167/14.7.9

Gelman, A., Roberts, G., & Gilks, W. (1996). Efficient metropolis jumping rules. *Bayesian statistics*, *5*, 599–607.

Glonek, G., & McCullagh, P. (1995). Multivariate Logistic Models. *Journal of the Royal Statistical Society, Series B (Methodological)*, *57*(3), 533–546.

Haario, H., Saksman, E., & Tamminen, J. (2001). An adaptive metropolis algorithm. *Bernoulli*, *7*(2), 223–242. doi: 10.2307/3318737

Heiss, F., & Winschel, V. (2008). Likelihood approximation by numerical integration on sparse grids. *Journal of Econometrics*, *144*(1), 62–80. doi: 10.1016/j.jeconom.2007.12.004

Henderson, H. V., & Searle, S. R. (1981). On deriving the inverse of a sum of matrices. *SIAM Review*, *23*(1), 53–60. doi: 10.1137/1023004

Higham, N. J. (1988). Computing a nearest symmetric positive semidefinite matrix. *Linear Algebra and Its Applications*, *103*(C), 103–118. doi: 10.1016/0024-3795(88)90223-6

Kim, W., Pitt, M. A., Lu, Z.-l., Steyvers, M., & Myung, J. I. (2014). A hierarchical adaptive approach to optimal experimental design paradigm of adaptive design optimization (ADO). *Neural computation*, *26*, 2465–2492. doi: 10.1162/NECO_a_00654

Knoblauch, K., & Maloney, L. T. (2008). Estimating classification images with generalized linear and additive models. *Journal of Vision*, *8*(16), 10.1–1019. doi: 10.1167/8.16.10

Kontsevich, L. L., & Tyler, C. W. (1999). Bayesian adaptive estimation of psychometric slope and threshold. *Vision Research*, *39*(16), 2729–2737. doi: 10.1016/S0042-6989(98)00285-5

Kujala, J. V., & Lukka, T. J. (2006). Bayesian adaptive estimation: The next dimension. *Journal of Mathematical Psychology*, *50*(4), 369–389. doi: 10.1016/j.jmp.2005.12.005

Kuss, M., Jäkel, F., & Wichmann, F. A. (2005). Bayesian inference for psychometric functions. *Journal of Vision*, *5*(5), 478–492. doi: 10.1167/5.5.8

Lau, B., & Glimcher, P. W. (2005). Dynamic response-by-response models of matching behavior in rhesus monkeys. *Journal of the Experimental Analysis of Behavior*, *84*(3), 555–579. doi: 10.1901/jeab.2005.110-04

Lesmes, L. A., Lu, Z.-L., Baek, J., & Albright, T. D. (2010). Bayesian adaptive estimation of the contrast sensitivity function: the quick CSF method. *Journal of Vision*, *10*(3), 17.1–21. doi: 10.1167/10.3.17

Lewi, J., Butera, R., & Paninski, L. (2009). Sequential optimal design of neurophysiology experiments. *Neural computation*, *21*(3), 619–687. doi: 10.1162/neco.2008.08-07-594

MacKay, D. J. C. (1992). Information-based objective functions for active data selection. *Neural Computation*, *4*(4), 590–604. doi: 10.1162/neco.1992.4.4.590

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, *21*(6), 1087. doi: 10.1063/1.1699114

Murray, R. F. (2011). Classification images: A review. *Journal of Vision*, *11*(5). doi: 10.1167/11.5.2

Neri, P., & Heeger, D. J. (2002). Spatiotemporal mechanisms for detecting and identifying image features in human vision. *Nat Neurosci*, *5*(8), 812–816. doi: 10.1038/nn886

Paninski, L., Ahmadian, Y., Ferreira, D. G., Koyama, S., Rahnama Rad, K., Vidne, M., … Wu, W. (2010). A new look at state-space models for neural data. *Journal of Computational Neuroscience*, *29*(1), 107–126. doi: 10.1007/s10827-009-0179-x

Park, M., Horwitz, G., & Pillow, J. W. (2011). Active learning of neural response functions with Gaussian processes. In *Advances in neural information processing systems 24* (pp. 2043–2051).

Park, M., & Pillow, J. W. (2012). Bayesian active learning with localized priors for fast receptive field characterization. In *Advances in neural information processing systems 25* (pp. 2357–2365).

Park, M., Weller, J. P., Horwitz, G. D., & Pillow, J. W. (2014). Bayesian active learning of neural firing rate maps with transformed Gaussian process priors. *Neural Computation*, *26*, 1519–1541. doi: 10.1162/NECO_a_00615

Pillow, J. W., Ahmadian, Y., & Paninski, L. (2011). Model-based decoding, information estimation, and change-point detection techniques for multineuron spike trains. *Neural Computation*, *23*(1), 1–45. doi: 10.1162/NECO_a_00058

Pillow, J. W., & Park, M. (2016). Adaptive Bayesian methods for closed-loop neurophysiology. In A. E. Hady (Ed.), *Closed loop neuroscience.* Elsevier.

Prins, N. (2012). The psychometric function: The lapse rate revisited. *Journal of Vision*, *12*(6), 25–25. doi: 10.1167/12.6.25

Prins, N. (2013). The psi-marginal adaptive method: How to give nuisance parameters the attention they deserve (no more, no less). *Journal of Vision*, *13*(7), 1–17. doi: 10.1167/13.7.3

Roberts, G. O., Gelman, A., & Gilks, W. R. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *Annals of Applied Probability*, *7*(1), 110–120. doi: 10.1214/aoap/1034625254

Rosenthal, J. S. (2011). Optimal proposal distributions and adaptive mcmc. In S. Brooks, A. Gelman, G. Jones, & X.-L. Meng (Eds.), *Handbook of Markov Chain Monte Carlo* (pp. 93–112). Chapman and Hall CRC. doi: 10.1201/b10905

Watson, A. B. (2017). QUEST+: A general multidimensional Bayesian adaptive psychometric method. *Journal of Vision*, *17*(3), 10. doi: 10.1167/17.3.10

Watson, A. B., & Pelli, D. G. (1983). QUEST: A Bayesian adaptive psychometric method. *Perception & psychophysics*, *33*(2), 113–120. doi: 10.3758/BF03202828

Wichmann, F. A., & Hill, N. J. (2001). The psychometric function: I. Fitting, sampling, and goodness of fit. *Perception & psychophysics*, *63*(8), 1293–1313. doi: 10.3758/BF03194544

Zocchi, S. S., & Atkinson, A. C. (1999). Optimum experimental designs for multinomial logistic models. *Biometrics*, *55*(2), 437–444. doi: 10.1111/j.0006-341X.1999.00437.x

Appendix A

Log likelihood for the classical MNL. Here we provide more details about the log likelihood $L = \mathbf{y}^\top \log \mathbf{p}$ under the multinomial logistic model (6), first in the lapse-free case.

A convenient property of the multinomial logistic model (a property common to all generalized linear models) is that the parameter vector p_i governing y depends only on a 1-dimensional projection of the input, $V_i = \boldsymbol{\phi}^\top \mathbf{w}_i$, which is known as the *linear predictor*. Recall that $\boldsymbol{\phi} = \boldsymbol{\phi}(\mathbf{x})$ is the input feature vector. In the multinomial case, it is useful to consider the column vector of linear predictors for a single trial, $\mathbf{V} = [V_1, \cdots, V_k]^\top$, and the concatenated weight vector $\mathbf{w} = [\mathbf{w}_1^\top, \cdots, \mathbf{w}_k^\top]^\top$, consisting of all weights stacked into a single vector. We can summarize their linear relationship as $\mathbf{V} = X\mathbf{w}$, where X is a block diagonal matrix containing k blocks of $\phi^\top$ along the diagonal. In other words,

$$X = \begin{bmatrix} \boldsymbol{\phi}^\top & \mathbf{0}^\top & \cdots & \mathbf{0}^\top \\ \mathbf{0}^\top & \boldsymbol{\phi}^\top & \cdots & \mathbf{0}^\top \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}^\top & \mathbf{0}^\top & \cdots & \boldsymbol{\phi}^\top \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_k \end{bmatrix}. \tag{26}$$

Derivatives. It is convenient to work in terms of the linear predictor $\mathbf{V} = \{V_i\}$ first. If $N_y \equiv \sum_i y_i = 1$ is the total number of responses per trial, the first and second derivatives of L with respect to $\mathbf{V}$ are $\partial L/\partial V_j = y_j - N_y p_j$ and $\partial^2 L/\partial V_i \partial V_j = N_y p_i(\delta_{ij} - p_j)$, respectively. Rewriting in vector forms, we have

$$\frac{\partial L}{\partial \mathbf{V}} = (\mathbf{y} - N_y \mathbf{p})^\top, \tag{27}$$

$$\frac{\partial^2 L}{\partial \mathbf{V}^2} = -N_y \left(\mathrm{diag}(\mathbf{p}) - \mathbf{p}\mathbf{p}^\top\right) \equiv -N_y \Gamma(\mathbf{p}), \tag{28}$$

where $\mathrm{diag}(\mathbf{p}) = [p_i \delta_{ij}]$ is a square matrix with the elements of $\mathbf{p}$ on the diagonal, and zeros otherwise.

Putting back in terms of the weight vector $\mathbf{w}$ is easy, thanks to the linear relationship $\mathbf{V} = X\mathbf{w}$:

$$\frac{\partial L}{\partial \mathbf{w}} = \frac{\partial L}{\partial \mathbf{V}} X = (\mathbf{y} - \mathbf{p})^\top X \equiv \boldsymbol{\Delta}^\top, \tag{29}$$

$$\frac{\partial^2 L}{\partial \mathbf{w}^2} = X^\top \frac{\partial^2 L}{\partial \mathbf{V}^2} X = -X^\top \Gamma X \equiv -\Lambda. \tag{30}$$

Concavity. Importantly, L is concave with respect to $\mathbf{V}$ (and therefore with respect to $\mathbf{w}$). To prove the concavity of L, we show that the Hessian $H = -\mathrm{diag}(\mathbf{p}) + \mathbf{p}\mathbf{p}^\top \equiv -\Gamma$ is negative semi-definite, which is equivalent to showing that $\mathbf{z}^\top \Gamma \mathbf{z} \geq 0$ for an arbitrary vector $\mathbf{z}$.

$$\begin{aligned}
\mathbf{z}^\top \Gamma \mathbf{z} &= \mathbf{z}^\top \mathrm{diag}(\mathbf{p}) \mathbf{z} - (\mathbf{z}^\top \mathbf{p})^2 \\
&= \textstyle\sum_i z_i^2 p_i - \left(\sum_j z_j p_j\right)^2 \\
&= \textstyle\sum_i z_i^2 p_i - 2\sum_i z_i p_i \sum_j z_j p_j + \left(\sum_j z_j p_j\right)^2 \\
&= \textstyle\sum_i p_i \left[z_i^2 - 2 z_i \sum_j z_j p_j + \left(\sum_j z_j p_j\right)^2 \right] \\
&= \textstyle\sum_i p_i \left[\left(z_i - \sum_j z_j p_j \right)^2 \right] \geq 0.
\end{aligned} \tag{31}$$

Log likelihood with lapse. With a finite lapse rate λ, to recap, the multinomial logistic model is modified as $p_i = (1-\lambda) q_i + \lambda c_i$ where

$$q_i = \frac{\exp(V_i)}{\sum_j \exp(V_j)}, \quad \lambda c_i = \frac{\exp(u_i)}{1 + \sum_j \exp(u_j)}. \tag{32}$$

Let us introduce the following abbreviations,

$$r_i \equiv \frac{\lambda c_i}{p_i}, \quad t_i \equiv y_i (1 - r_i), \quad s_i \equiv y_i r_i (1 - r_i), \tag{33}$$

where the dimensionless ratio $r \in [0, 1]$ can be considered as the order parameter for the effect of lapse.

Derivatives with respect to the weights. Differentiating with the linear predictor $\mathbf{V}$, we get

$$\begin{aligned}
\frac{\partial q_i}{\partial V_l} &= (\delta_{il} - q_l) q_i, \\
\frac{\partial^2 q_i}{\partial V_j \partial V_l} &= \left[(\delta_{ij} - q_j)(\delta_{il} - q_l) - (\delta_{jl} q_l - q_j q_l) \right] q_i.
\end{aligned}$$

which leads to

$$\frac{\partial p_i}{\partial V_l} = (1-\lambda) \frac{\partial q_i}{\partial V_l}, \quad \frac{\partial^2 p_i}{\partial V_j \partial V_l} = (1-\lambda) \frac{\partial^2 q_i}{\partial V_j \partial V_l}.$$

We are interested in the derivatives of the log likelihood $L = \mathbf{y}^\top \log \mathbf{p}$ with respect to $\mathbf{V}$. The partial gradient:

$$\begin{aligned}
\frac{\partial L}{\partial V_l} = \sum_i y_i \frac{1}{p_i} \frac{\partial p_i}{\partial V_l} &= (1-\lambda) \sum_i y_i \frac{q_i}{p_i} (\delta_{il} - q_l) \\
&= t_l - q_l \sum_i t_i.
\end{aligned}$$

Similarly, the partial Hessian is written as

$$\begin{aligned}
\frac{\partial^2 L}{\partial V_j \partial V_l} &= \sum_i y_i \left(\frac{1}{p_i} \frac{\partial^2 p_i}{\partial V_j \partial V_l} - \frac{1}{p_i^2} \frac{\partial p_i}{\partial V_j} \frac{\partial p_i}{\partial V_l} \right) \\
&= \delta_{jl} \left(s_l - q_l \textstyle\sum_i t_i \right) - (q_j s_l + q_l s_j) + q_j q_l \left(\textstyle\sum_i s_i + \sum_i t_i \right).
\end{aligned}$$

In vector forms, and with $\tau \equiv \sum_i t_i$ and $\sigma \equiv \sum_i s_i$,

$$\frac{\partial L}{\partial \mathbf{V}} = (\mathbf{t} - \tau \mathbf{q})^\top; \tag{34}$$

$$\begin{aligned}
\frac{\partial^2 L}{\partial \mathbf{V}^2} &= \mathrm{diag}(\mathbf{s} - \tau \mathbf{q}) - (\mathbf{q}\mathbf{s}^\top + \mathbf{s}\mathbf{q}^\top) + (\tau + \sigma) \mathbf{q}\mathbf{q}^\top \\
&= -\tau \left[\mathrm{diag}(\mathbf{q}) - \mathbf{q}\mathbf{q}^\top \right] \\
&\quad + \left[\mathrm{diag}(\mathbf{s}) - (\mathbf{q}\mathbf{s}^\top + \mathbf{s}\mathbf{q}^\top) + \sigma\, \mathbf{q}\mathbf{q}^\top \right].
\end{aligned} \tag{35}$$

Note that we recover $t_i \to y_i$ and $s_i \to 0$ in the lapse-free limit $\lambda \to 0$. Hence the first square bracket in (35) reduces back to the lapse-free Hessian, while the second square bracket vanishes as $\lambda \to 0$.

In the presence of lapse, one might still be interested in the partial Hessian with respect to the weight parameters, $H \equiv \partial^2 L / \partial \mathbf{V}^2$, which should be evaluated as in (35). To test the negative semi-definiteness of this partial Hessian, again for an arbitrary vector $\mathbf{z}$, we end up with

$$\mathbf{z}^\top H \mathbf{z} = -\sum_j t_j \left\langle (z - \langle z \rangle_q)^2 \right\rangle_q + \sum_j s_j \left(z_j - \langle z \rangle_q \right)^2 \tag{36}$$

where $\langle x \rangle_q = \sum_j x_j q_j$. The partial Hessian is asymptotically negative semi-definite (which is equivalent to the log likelihood being concave) in the lapse-free limit, where $t_j \to y_j$ and $s_j \to 0$.

Derivatives with respect to lapse parameters. From (2) and (3), we have $p_i = (1-\lambda) q_i + \lambda c_i$ where

$$c_i = \frac{\exp(u_i)}{\sum_j \exp(u_j)}; \qquad \lambda = \frac{\sum_j \exp(u_j)}{1 + \sum_j \exp(u_j)}. \tag{37}$$

Differentiating with respect to the auxiliary lapse parameter u_i,

$$\frac{\partial c_i}{\partial u_j} = (\delta_{ij} - c_i) c_j; \qquad \frac{\partial \lambda}{\partial u_j} = (1-\lambda) \lambda c_j. \tag{38}$$

The gradient is then

$$\frac{\partial p_i}{\partial u_j} = (\delta_{ij} - p_i) \lambda c_j; \tag{39}$$

using the abbreviations in (33), the gradient of the log likelihood is

$$\frac{\partial L}{\partial u_j} = \sum_i y_i \frac{1}{p_i} \frac{\partial p_i}{\partial u_j} = r_j \left(y_j - N_y \cdot p_j \right). \tag{40}$$

Second derivative with respect to lapse:

$$\frac{\partial^2 p_i}{\partial u_j \partial u_l} = \delta_{jl} \frac{\partial p_i}{\partial u_l} - (\delta_{ij} + \delta_{il} - 2p_i) \lambda c_l \lambda c_j; \tag{41}$$

it is useful to notice that

$$\frac{\partial p_i}{\partial u_j} \frac{\partial p_i}{\partial u_l} = \delta_{jl} \frac{\partial p_i}{\partial u_l} \lambda c_l - p_i (\delta_{ij} + \delta_{il} - 2p_i) \lambda c_l \lambda c_j. \tag{42}$$

The corresponding part of the Hessian:

$$\begin{aligned} \frac{\partial^2 L}{\partial u_j \partial u_l} &= \sum_i y_i \left(\frac{1}{p_i} \frac{\partial^2 p_i}{\partial u_j \partial u_l} - \frac{1}{p_i^2} \frac{\partial p_i}{\partial u_j} \frac{\partial p_i}{\partial u_l} \right) \\ &= \delta_{jl} \sum_i y_i \frac{1}{p_i} \left(1 - \frac{\lambda c_l}{p_i} \right) \frac{\partial p_i}{\partial u_l} \\ &= \delta_{jl} \left(s_l - r_l p_l N_y + r_l^2 p_l^2 \sum_i \frac{y_i}{p_i} \right). \end{aligned} \tag{43}$$

Finally, the mixed derivative:

$$\frac{\partial^2 p_i}{\partial u_j \partial V_l} = -(1 - \lambda) \lambda c_j \cdot (\delta_{il} - q_l) q_l. \tag{44}$$

again it is useful to notice that

$$\frac{\partial p_i}{\partial u_j} \frac{\partial p_i}{\partial V_l} = -(\delta_{ij} - p_i) \frac{\partial^2 p_i}{\partial u_j \partial V_l}. \tag{45}$$

Hence

$$\begin{aligned} \frac{\partial^2 L}{\partial u_j \partial V_l} &= \sum_i y_i \left(\frac{1}{p_i} \frac{\partial^2 p_i}{\partial u_j \partial V_l} - \frac{1}{p_i^2} \frac{\partial p_i}{\partial u_j} \frac{\partial p_i}{\partial V_l} \right) \\ &= -s_j \left(\delta_{jl} + \frac{q_l^2}{q_j} \right). \end{aligned} \tag{46}$$

From (40), (43) and (46), we see that all derivatives involving the lapse parameter scale with at least one order of r, therefore vanishing in the lapse-free limit $\lambda \to 0$.

Appendix B

The Metropolis-Hastings algorithm. The Metropolis-Hastings algorithm (Metropolis et al., 1953) generates a chain of samples, using a proposal density and a method to accept or reject the proposed moves.

A proposal is made at each iteration, where the algorithm randomly chooses a candidate for the next sample value $\mathbf{x}'$ based on the current sample value $\mathbf{x}_t$. The choice follows the proposal density function, $\mathbf{x}' \sim Q(\mathbf{x}'|\mathbf{x}_t)$. When the proposal density Q is symmetric, for example a Gaussian, the sequence of samples is a random walk. In general the width of Q should match with the statistics of the distribution being sampled, and individual dimensions in the sampling space may behave differently in the multivariate case; finding the appropriate Q can be difficult.

The proposed move is either accepted or rejected with some probability; if rejected, the current sample value is reused in the next iteration, $\mathbf{x}' = \mathbf{x}_t$. The probability of acceptance is determined by comparing the values of $P(\mathbf{x}_t)$ and $P(\mathbf{x}')$, where $P(\mathbf{x})$ is the distribution being sampled. Because the algorithm only considers the acceptance ratio $\rho = P(\mathbf{x}')/P(\mathbf{x}_t) = f(\mathbf{x}')/f(\mathbf{x}_t)$ where $f(\mathbf{x})$ can be any function proportional to the desired distribution $P(\mathbf{x})$, there is no need to worry about the proper normalization of the probability distribution. If $\rho \geq 1$, the move is always accepted; if $\rho < 1$, it is accepted with a probability ρ. Consequently the samples tend to stay in the high-density regions, visiting the low-density regions only occasionally.

Optimizing the sampler. One of the major difficulties in using the MCMC method is to make an appropriate choice of the proposal distribution, which may significantly affect the performance of the sampler. If the proposal distribution is too narrow, it will take a long time for the chain to diffuse away from the starting point, producing a chain with highly correlated samples, requiring a long time to achieve independent samples. On the other hand if the proposal distribution is too wide, most of the proposed moves would be rejected, once again resulting in the chain stuck at the initial point. In either case the chain would "mix" poorly (Rosenthal, 2011). In this paper we restrict our consideration to the Metropolis-Hastings algorithm (Metropolis et al., 1953), although the issue of proposal distribution optimization is universal in most variants of MCMC algorithms, only with implementation-level differences.

The basic idea is that the optimal width of the proposal distribu-

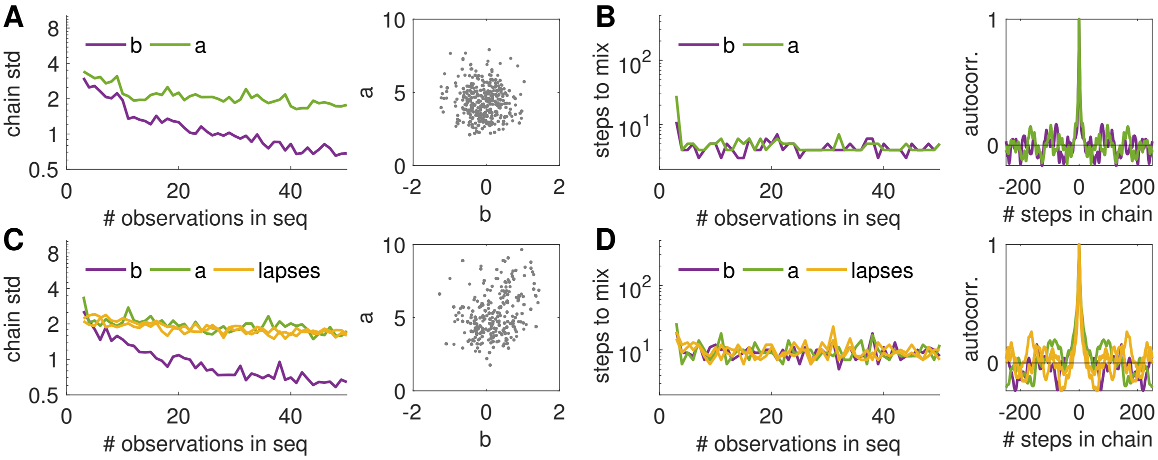

Figure 7: Statistics of the semi-adaptive MCMC in a simulated experiment, with $M = 1000$ samples per chain. We used the same binomial model as in Fig. 2, and the uniform stimulus selection algorithm. **(A-B)** In a lapse-free model: **(A)** The standard deviation of the samples, along each dimension of the parameter space, decreases as the learning progresses, as expected because the posterior distribution should narrow down as more observations are collected. Also shown is the scatter plot of all 1000 samples at the last trial $N = 50$, where the true parameter values are $(a, b) = (5, 0)$. **(B)** The mixing time of the chain (number of steps before the autocorrelation falls to $1/e$) quickly converges to some small value, meaning that the sampler is quickly optimized. Autocorrelation function at the last trial $N = 50$ is shown. **(C-D)** Same information as (A) and (B), but with a lapse rate of $\lambda = 0.1$, with uniform lapse ($c_1 = c_2 = 1/2$).

tion would be determined in proportion to the typical length scale of the distribution being sampled. This idea was made precise in the case of a stationary random-walk Metropolis algorithm with Gaussian proposal distributions, by comparing the covariance matrix Σ_p of the proposal distribution to the covariance matrix Σ of the sampled chain. Once a linear scaling relation $\Sigma_p = s_d\Sigma$ is fixed, it was observed that it is optimal to have $s_d = (2.38)^2/d$ where d is the dimensionality of the sampling space (Gelman et al., 1996; Roberts et al., 1997). An adaptive Metropolis algorithm (Haario et al., 2001) followed this observation, where the Gaussian proposal distribution adapts continuously as the sampling progresses. Their adaptive algorithm used the same scaling rule $\Sigma_p = s_d\Sigma$, but updates Σ_p at each proposal where Σ is covariance of the samples accumulated so far. Additionally, a small diagonal component was added for stability, as $\Sigma_p = s_d(\Sigma + \epsilon I)$. We used $\epsilon = 0.0001$ in this work.

Here we propose and use the semi-adaptive Metropolis-Hastings algorithm, which is a coarse-grained version of the original adaptive algorithm by Haario et al. (2001). The major difference in our algorithm is that the adjustment of the proposal distribution is made only at the end of each (sequential) chain, rather than at each proposal within the chain. This coarse-graining is a reasonable approximation because we will be sampling the posterior distribution many times as it refines over the course of data collection, once after each trial. Assuming that the change in posterior distribution after each new observation is small enough, we can justify our use of the statistics of the previous chain to adjust the properties of the current chain. Unlike in the fully adaptive algorithm where the proposal distribution needs to stabilize quickly within a single chain, we can allow multiple chains until stabilization, usually a few initial observations – leaving some room for the coarse-grained approximation. This is because, for our purpose, it is not imperative that we have a good sampling of the distribution at the very early stage of the learning sequence where the accuracy is already limited by the smallness of the dataset.

When applied to the sequential learning algorithm, our semi-adaptive Metropolis sampler shows a consistent well-mixing property after a few initial adjustments, with the standard deviation of each sampling dimension decreasing stably as data accumulate (Fig. 7). Whereas Kujala and Lukka (2006) also had the idea of adjusting the proposal density between trials, their scaling factor was fixed and independent of the sampling dimension. Building on more precise statistical observations, our method generalize well to high-dimensional parameter spaces, typical for multiple-alternative models. Our semi-adaptive sampler provides an efficient and robust alternative to the particle filter implementations (Kujala & Lukka, 2006), which has the known problem of weight degeneration (DiMattina, 2015) as the posterior distribution narrows down with the accumulation of data.

Appendix C

Fast sequential update of the posterior, with Laplace approximation. Use of Laplace approximation was shown to be particularly useful in a sequential experiment (Lewi et al., 2009), where it can be assumed that the posterior distribution after the next trial in sequence, $\mathcal{P}_{t+1}$, would not be very different from the current posterior $\mathcal{P}_t$. Let us consider the lapse-free case $\boldsymbol{\theta} = \mathbf{w}$ for the moment, where the use of Laplace approximation is valid. Rearranging from (7) and (9), the sequential update for the posterior distribution is

$$\log \mathcal{P}_{t+1}(\mathbf{w}) = \log \mathcal{P}_t(\mathbf{w}) + L_{t+1}(\mathbf{w}); \tag{47}$$

or with Laplace approximation,

$$\log \mathcal{N}(\mathbf{w}|\boldsymbol{\theta}_{t+1}, C_{t+1}) \approx \log \mathcal{N}(\mathbf{w}|\boldsymbol{\theta}_t, C_t) + L_{t+1}(\mathbf{w}) \tag{48}$$

where $L_i(\mathbf{w}) = \log p(\mathbf{y}_i|\mathbf{x}_i, \mathbf{w})$ is a shorthand for the log likelihood of the i-th observation.

With this, we can achieve a fast sequential update of the posterior without performing the full numerical optimization each time. Because the new posterior mode $\boldsymbol{\theta}_{t+1}$ is where the gradient vanishes, it can be approximated from the previous mode $\boldsymbol{\theta}_t$ by taking the first derivative of (48). The posterior covariance C_{t+1} is similarly approximated by taking the second derivate.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + C_t \boldsymbol{\Delta}_{t+1}, \qquad \boldsymbol{\Delta}_{t+1} = \left.\frac{\partial L_{t+1}}{\partial \mathbf{w}}\right|_{\mathbf{w}=\boldsymbol{\theta}_t} \tag{49}$$

$$C_{t+1} = \left(C_t^{-1} + \Lambda_{t+1}\right)^{-1}, \qquad \Lambda_{t+1} = -\left.\frac{\partial^2 L_{t+1}}{\partial \mathbf{w}^2}\right|_{\mathbf{w}=\boldsymbol{\theta}_{t+1}} \tag{50}$$

Using the matrix inversion lemma (Henderson & Searle, 1981), we can rewrite the posterior covariance update as

$$C_{t+1} = C_t \left[I - (I + \Lambda_{t+1} C_t)^{-1} \Lambda_{t+1} C_t\right]. \tag{51}$$

Unlike in the earlier application of this trick (Lewi et al., 2009), the covariance matrix update (50) is not a rank-one update, because of the multinomial nature of our model (our linear predictor $\mathbf{y}$ is a vector, not a scalar as in a binary model).

Note that this approximate sequential update is only used for calculating the expected utility of each candidate stimulus by approximating the posterior distribution at the next trial, as in Section Adaptive Stimulus Selection Methods. For obtaining the MAP estimates of the model parameters, numerical optimization should be performed using the full accumulated dataset each time.

Integration over the parameter space: reducing the integration space. The evaluation of expected utility function usually involves a potentially high-dimensional integral over the parameter space. With the Gaussian approximation of the posterior, we can reduce and standardize the integration space. The process consists of three steps: diagonalization, marginalization, and standardization. First we choose a new "coordinate system" of the (say q-dimensional) weight space, such that the first k elements of the extended weight vector $\mathbf{w}$ are coupled one-to-one to the elements of k-vector $\mathbf{y}$. Then we marginalize to integrate out the remaining $(q - k)$ dimensions, effectively changing the integration variable from $\mathbf{w}$ to $\mathbf{y}$. Finally, we use Cholesky decomposition to standardize the normal distribution which is the posterior on $\mathbf{y}$. The resulting integral is still multi-dimensional, due to the multinomial nature of our model. But once the distribution is standardized, there are a number of efficient numerical integration methods that can be applied. For example, in this work, we use the Sparse Grid method (Heiss & Winschel, 2008) based on Gauss-Hermite quadrature.

Diagonalization. It is clear from (19-20) and (29-30) that all parameter-dependence in our integrand is in terms of the linear predictor $\mathbf{y} = X\mathbf{w}$. That is, we are dealing with the integral of the form

$$F = \int d\mathbf{w}' \, \mathcal{N}(\mathbf{w}'|\hat{\mathbf{w}}', C) \cdot f(X\mathbf{w}'), \qquad (52)$$

where C is the covariance matrix, and $X = \oplus_{j=1}^{k} {\mathbf{g}'}_j^\top$ is a fixed matrix constructed from direct sum of k vectors. It helps to work in a diagonalized coordinate system, so that we can separate out the relevant dimensions of $\mathbf{w}$. We use the singular value decomposition of the design matrix ($X = UGV^\top$ with $U = I$ and $V = Q^\top$). Because of the direct-sum construction, $XX^\top$ is already diagonal, and the left singular matrix is always I in this case. Then

$$G = XQ^\top = \begin{bmatrix} G_k & G_q \end{bmatrix}, \qquad (53)$$

where G_k is a $k \times k$ diagonal matrix and G_q is a $k \times (q-k)$ matrix of zeros. We can now denote $\mathbf{w}_k = (w_1, \cdots, w_k)$ and $\mathbf{w}_q = (w_{k+1}, \cdots, w_q)$ in the diagonalized variable $\mathbf{w} = Q\mathbf{w}'$, such that

$$\mathbf{w} = [\mathbf{w}_k, \mathbf{w}_q]^\top, \quad G\mathbf{w} = G_k\mathbf{w}_k = (g_1 w_1, g_2 w_2, \cdots g_k w_k).$$

Marginalization. Now we have

$$F = \int d\mathbf{w} \, \mathcal{N}(\mathbf{w}|\hat{\mathbf{w}}, B^{-1}) \cdot f(G\mathbf{w}), \qquad B^{-1} = QCQ^\top \quad (54)$$

where B is the inverse of the *new* covariance matrix after diagonalization. If we block-decompose this matrix,

$$B = \begin{bmatrix} B_{kk} & B_{kq} \\ B_{qk} & B_{qq} \end{bmatrix}, \qquad B_{kq} = (B_{qk})^\top, \qquad (55)$$

the Gaussian distribution is also decomposed as

$$\mathcal{N}(\mathbf{w}|\hat{\mathbf{w}}, B^{-1}) = \mathcal{N}(\mathbf{w}_k|\hat{\mathbf{w}}_k, B_*^{-1}) \cdot \mathcal{N}(\mathbf{w}_q|(\hat{\mathbf{w}}_q - \mathbf{b}), B_{qq}^{-1})$$

where $\mathbf{b} = B_{qq}^{-1} B_{qk} \mathbf{w}_k$ and $B_* = B_{kk} - B_{kq} B_{qq}^{-1} B_{qk}$. As the non-parallel part $\mathbf{w}_q$ is integrated out, we have marginalized the integral. It is useful to recall that if a variable $\mathbf{w} \sim \mathcal{N}(\hat{\mathbf{w}}, C)$ is Gaussian distributed, its linear transform $\mathbf{y} = X\mathbf{w}$ is also Gaussian distributed as $\mathbf{y} \sim \mathcal{N}(\hat{\mathbf{y}}, \Sigma)$, with $\hat{\mathbf{y}} = X\hat{\mathbf{w}}$ and $\Sigma = XCX^\top$. Changing the integration variable to $\mathbf{y} = G_k\mathbf{w}_k$ is then straightforward:

$$\begin{aligned} F &= \int d\mathbf{w}_k \, \mathcal{N}(\mathbf{w}_k|\hat{\mathbf{w}}_k, B_*^{-1}) \cdot f(G_k\mathbf{w}_k) \\ &= \int d\mathbf{y} \, \mathcal{N}(\mathbf{y}|\hat{\mathbf{y}}, \Sigma) \cdot f(\mathbf{y}), \qquad \Sigma = G_k B_*^{-1} G_k^\top. \qquad (56) \end{aligned}$$

Standardization. Finally, in order to deal with the numerical integration, it is convenient to have the normal distribution standardized. We can use the Cholesky decomposition for the covariance matrix,

$$LL^\top = \Sigma_{t+1}, \qquad (57)$$

such that the new variable $\boldsymbol{\theta} = L^{-1}(\mathbf{y} - \hat{\mathbf{y}}_{t+1})$ is standard normal distributed. From the above formulation, L can be written directly in terms of the Cholesky decomposition of B_*:

$$L = G_k R^{-1} \quad \text{where} \quad R^\top R = B_*. \qquad (58)$$

Importantly, with this transformation, each dimension of $\boldsymbol{\theta}$ is independently and identically distributed. The objective function to be evaluated is now

$$\begin{aligned} F(\mathbf{x}) &= \int d\mathbf{y} \cdot \mathcal{N}(\mathbf{y}|\hat{\mathbf{y}}_{t+1}, \Sigma_{t+1}) \cdot f(\mathbf{y}, \mathbf{x}) \\ &= \int d\boldsymbol{\theta} \cdot \mathcal{N}(\boldsymbol{\theta}|\mathbf{0}, I) \cdot f(\phi(\boldsymbol{\theta}), \mathbf{x}) \qquad (59) \end{aligned}$$

where $\phi(\boldsymbol{\theta}) = \hat{\mathbf{y}}_{t+1} + L\boldsymbol{\theta}$. Once the integration is standardized this way, there are a number of efficient numerical methods that can be applied.