

PSL-week | March 4-8 2024

Lecture 2 (data mining and modeling for behavioral sciences)

# Guidelines for designing a cognitive modeling study: theory and practice

**Valentin Wyart**

Lab. de Neurosciences Cognitives et Computationnelles (LNC<sup>2</sup>)

Institut National de la Santé et de la Recherche Médicale

Ecole Normale Supérieure, Université PSL

[valentin.wyart@ens.psl.eu](mailto:valentin.wyart@ens.psl.eu)





PSL Data Science Program

<https://psl.eu/en/programmes-gradues/programme-data>

PR[AI]RIE

PaRis Artificial Intelligence Research InstitutE

Paris Artificial Intelligence Research Institute

<https://prairie-institute.fr>

# Paper to read



eLIFE

elifesciences.org

REVIEW ARTICLE



## Ten simple rules for the computational modeling of behavioral data

Robert C Wilson<sup>1,2†\*</sup>, Anne GE Collins<sup>3,4†\*</sup>

<sup>1</sup>Department of Psychology, University of Arizona, Tucson, United States;

<sup>2</sup>Cognitive Science Program, University of Arizona, Tucson, United States;

<sup>3</sup>Department of Psychology, University of California, Berkeley, Berkeley, United States; <sup>4</sup>Helen Wills Neuroscience Institute, University of California, Berkeley,

Berkeley, United States

---

**Abstract** Computational modeling of behavior has revolutionized psychology and neuroscience. By fitting models to experimental data we can probe the algorithms underlying behavior, find neural correlates of computational variables and better understand the effects of drugs, illness and interventions. But with great power comes great responsibility. Here, we offer ten simple rules to ensure that computational modeling is used with care and yields meaningful insights. In particular, we present a beginner-friendly, pragmatic and details-oriented introduction on how to relate models to data. What, exactly, can a model tell us about the mind? To answer this, we apply our rules to the simplest modeling techniques most accessible to beginning modelers and illustrate them with examples and code available online. However, most rules apply to more advanced

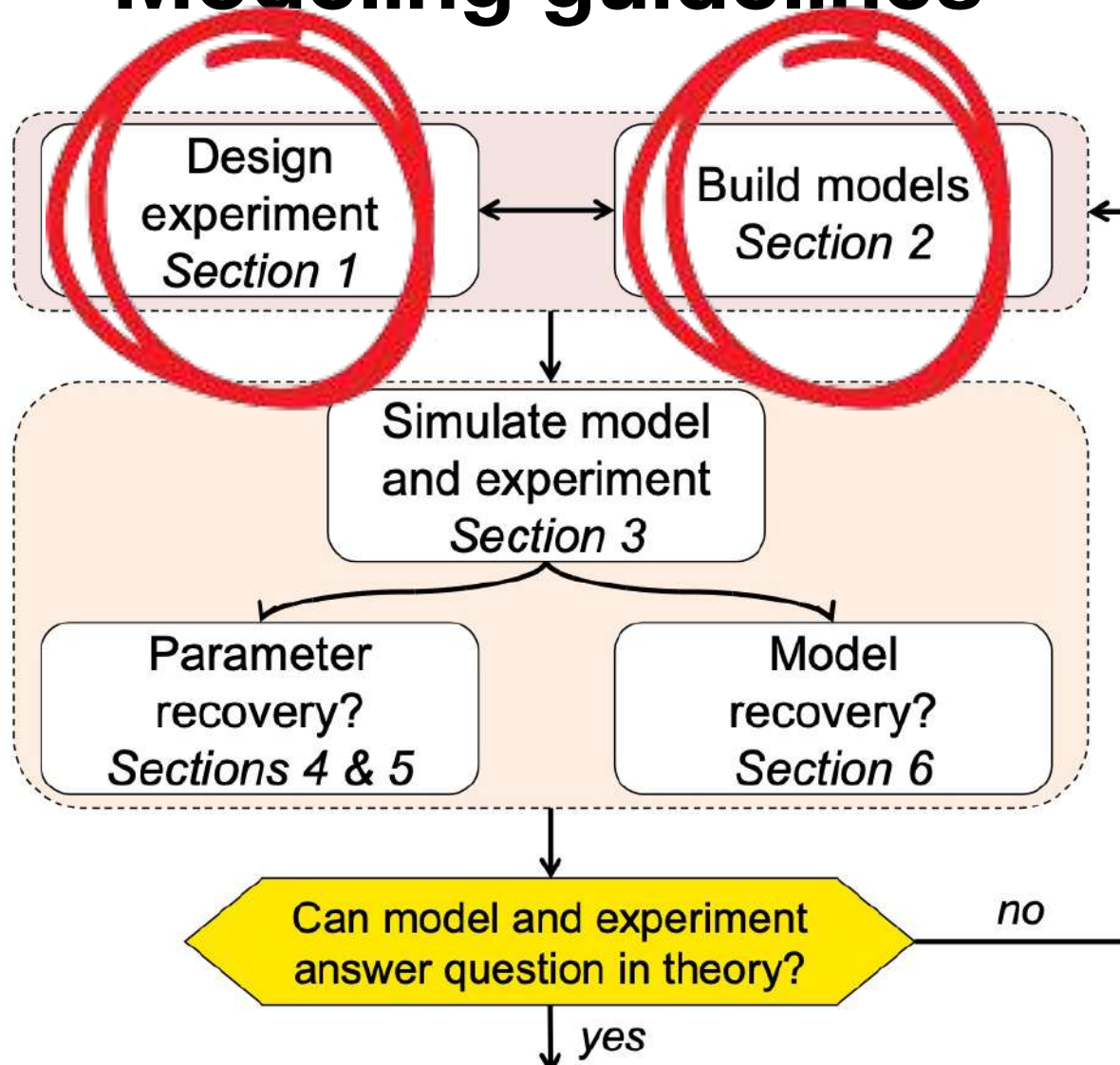
# Paper to read

- Goal: provide **simple rules** for computational cognitive modeling studies of behavior
- Why is it important to follow these rules?
- Two critical steps:
  - ✓ **designing** a valid cognitive modeling study
  - ✓ **analyzing data** using cognitive modeling
- Cognitive modeling is most powerful when used **during experimental design**.

# Modeling guidelines

- Today: guidelines for designing a meaningful cognitive modeling study = **step 1**
- Do you remember the difference(s) between a descriptive and a generative model of behavior?
- Do you remember the difference(s) between absolute and relative approaches to model selection and falsification?

# Modeling guidelines



# Modeling guidelines

- No data will be collected during **step 1!**
- So what will we be looking at?
- Uses of cognitive models in the same study:
  - ✓ model simulation
  - ✓ model fitting
  - ✓ model comparison
  - ✓ model parameter estimation
  - ✓ model latent variable estimation

# Modeling guidelines

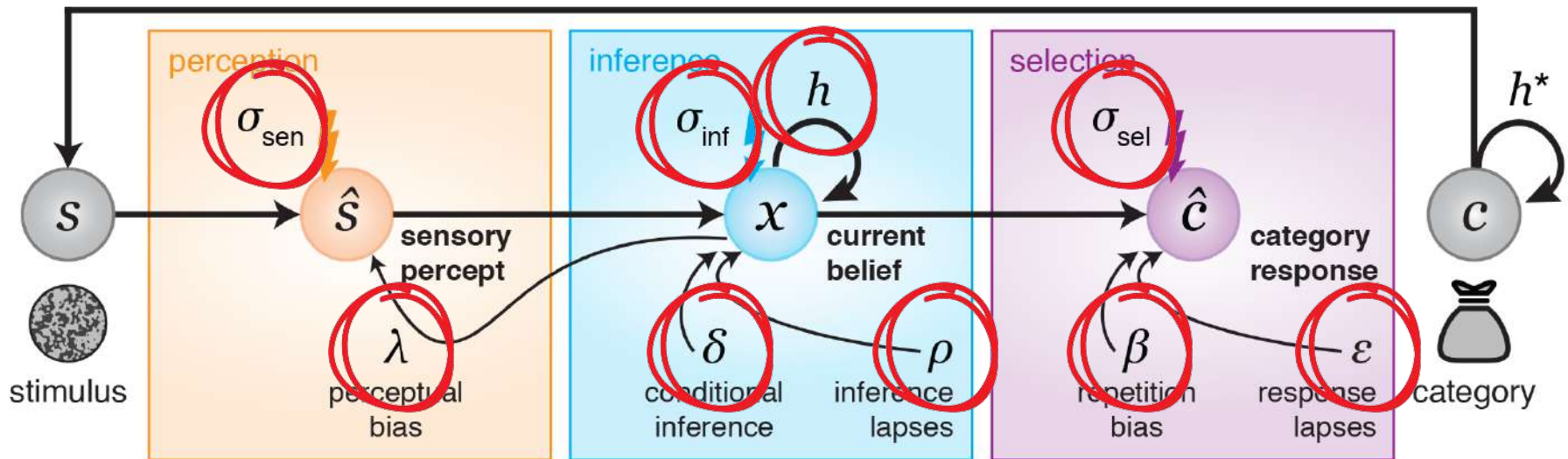
- Difference between model parameters and model latent variables
- Model parameters: set of adjustable values  $\theta$  that control specific aspects of the model  
>>  $\text{behavior} = f(s, \theta)$
- Model latent variables: hidden variables  $x$  that are computed by the model  
>>  $x_1 = f_1(s, \theta_1)$   
>>  $x_2 = f_2(x_1, \theta_2)$   
>>  $\text{behavior} = f_3(x_2, \theta_3)$



# Modeling guidelines

- Difference between model parameters and model latent variables

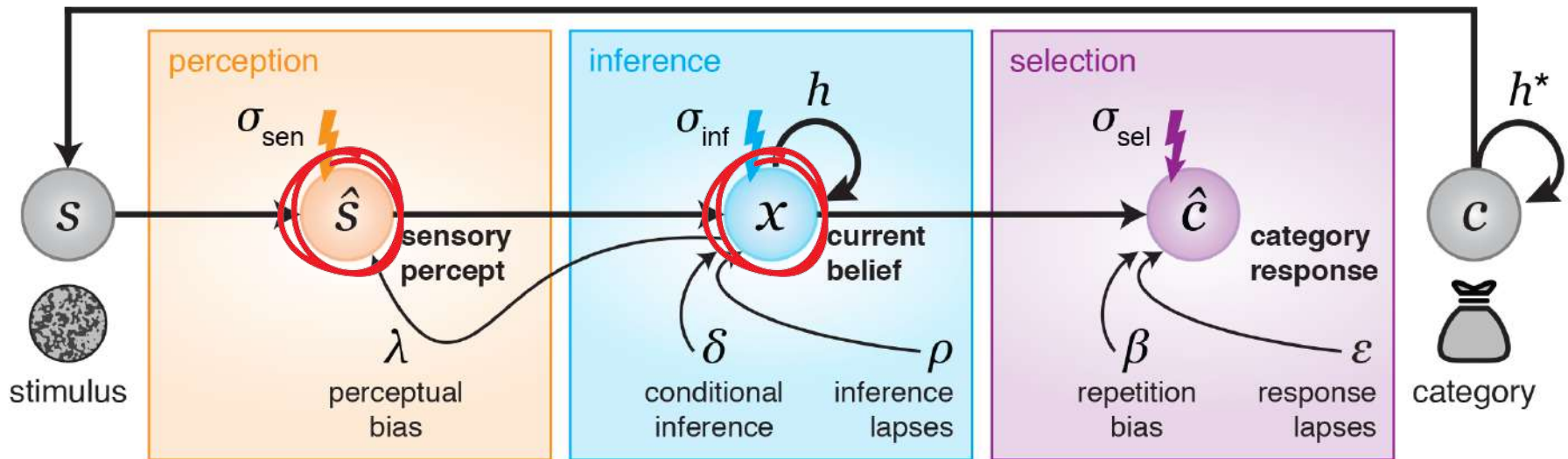
hidden-state inference model



# Modeling guidelines

- Difference between model parameters and model latent variables

hidden-state inference model



# Modeling guidelines

- Difference between model parameters and model latent variables
- Example of TD-based RL:

$$Q_{1,t} = Q_{1,t-1} + \alpha(r_t - Q_{1,t-1})$$

$$p_t = 1 / \left( 1 + \exp(- (Q_{1,t} - Q_{2,t}) / \tau) \right)$$

# Modeling guidelines

- Difference between model parameters and model latent variables

- Example of TD-based RL:

$$Q_{1,t} = Q_{1,t-1} + \alpha(r_t - Q_{1,t-1})$$

$$p_t = 1 / \left( 1 + \exp(- (Q_{1,t} - Q_{2,t}) / \tau) \right)$$

- $\alpha$  and  $\tau$  are model parameters:
  - ✓  $\alpha$  = learning rate (Rescorla-Wagner rule)
  - ✓  $\tau$  = choice temperature (softmax policy)

# Modeling guidelines

- Difference between model parameters and model latent variables

- Example of TD-based RL:

$$Q_{1,t} = Q_{1,t-1} + \alpha(r_t - Q_{1,t-1})$$

$$p_t = 1 / \left( 1 + \exp(- (Q_{1,t} - Q_{2,t}) / \tau) \right)$$

- $Q_{i,t}$  and  $p_t$  are model latent variables:
  - ✓  $Q_{i,t}$  = value of option  $i$  at time  $t$
  - ✓  $p_t$  = choice probability of option 1 at time  $t$

# Modeling guidelines

- Model **simulation**: running the model with particular set of parameters  $\theta$  to generate synthetic behavior (and latent variables  $x$ )
- This synthetic behavior can be analyzed exactly like human behavior to make **precise, falsifiable predictions** about **qualitative** and **quantitative features** of behavior.
- What does falsifiable mean?  
Difference btw qualitative and quantitative?

# Modeling guidelines

Trends in Cognitive Sciences

CellPress

## Opinion

# The Importance of Falsification in Computational Cognitive Modeling

Stefano Palminteri,<sup>1,2,\*</sup> Valentin Wyart,<sup>1,2,\*</sup> and Etienne Koechlin<sup>1,2,\*</sup>

In the past decade the field of cognitive sciences has seen an exponential growth in the number of computational modeling studies. Previous work has indicated why and how candidate models of cognition should be compared by trading off their ability to predict the observed data as a function of their complexity. However, the importance of falsifying candidate models in light of the observed data has been largely underestimated, leading to important drawbacks and unjustified conclusions. We argue here that the simulation of candidate models is necessary to falsify models and therefore support the

## Trends

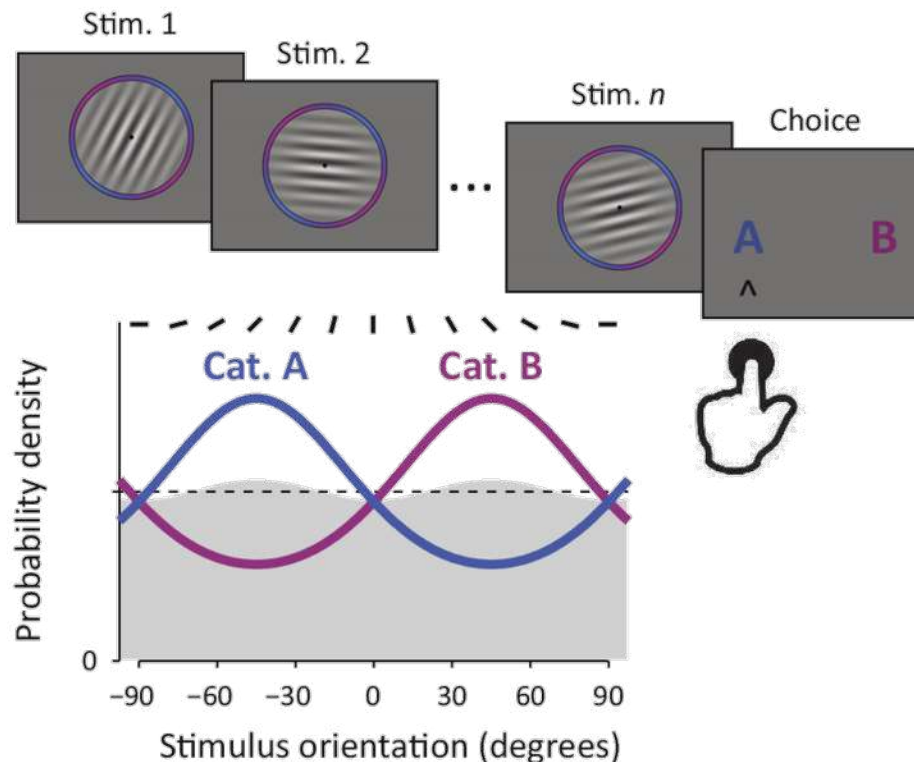
Computational modeling has grown exponentially in cognitive sciences in the past decade.

Model selection most often relies on evaluating the ability of candidate models to predict the observed data.

The ability of a candidate model to generate a behavioral effect of interest

# Modeling guidelines

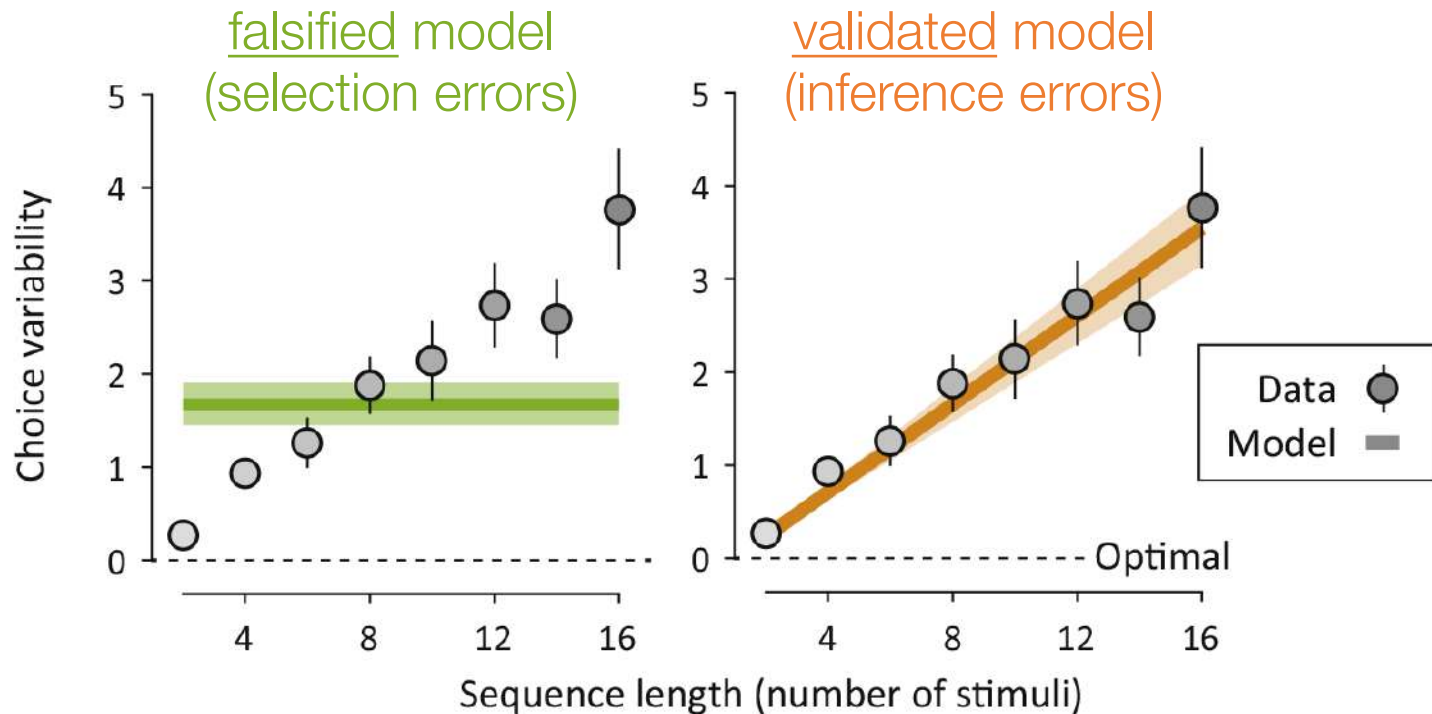
- Model **falsification**: showing through simulations that a model is not able to generate a specific behavioral effect of interest.





# Modeling guidelines

- Model **falsification**: showing through simulations that a model is not able to generate a specific behavioral effect of interest.



# Modeling guidelines

Article

## Neuron

### Computational Precision of Mental Inference as Critical Source of Human Choice Suboptimality

#### Highlights

- Human decisions based on multiple ambiguous cues are typically suboptimal
- Sensory noise and response selection cannot explain choice suboptimality alone
- Imperfections in mental inference cause a dominant fraction of choice suboptimality
- Most of choice suboptimality arises from imprecise rather than biased computations

#### Authors

Jan Drugowitsch, Valentin Wyart,  
Anne-Dominique Devauchelle,  
Etienne Koechlin

#### Correspondence

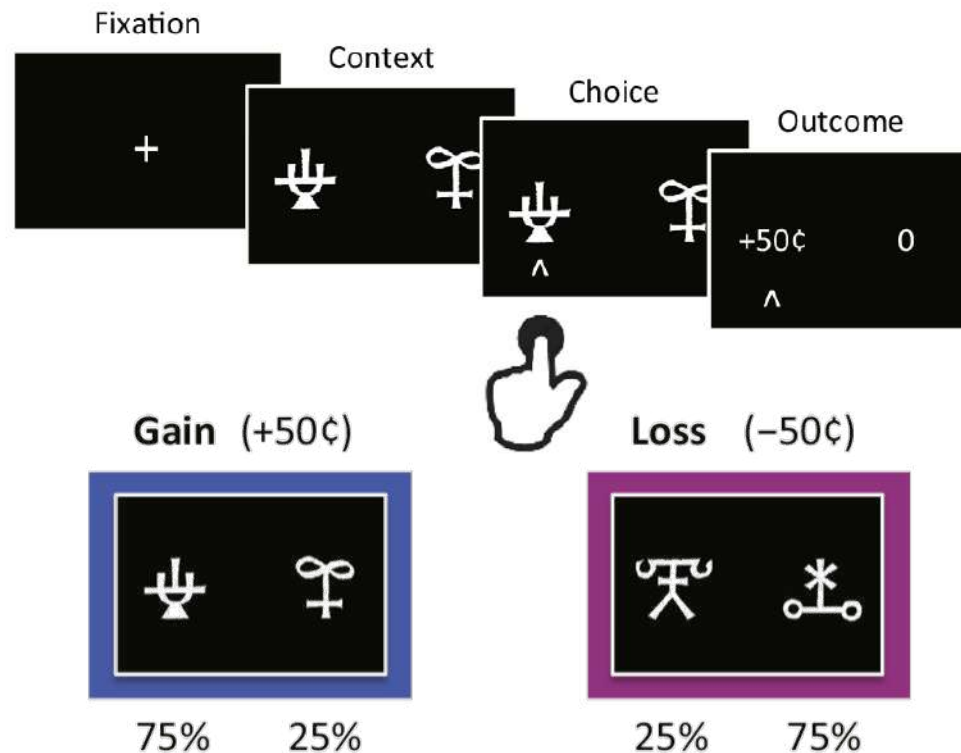
jan\_drugowitsch@hms.harvard.edu  
(J.D.),  
valentin.wyart@ens.fr (V.W.)

#### In Brief

Decisions made under uncertainty show a suboptimal variability whose origin is usually ascribed to the peripheries of the decision process. Drugowitsch et al. show that computational imprecisions in the decision process itself account for a dominant fraction of choice

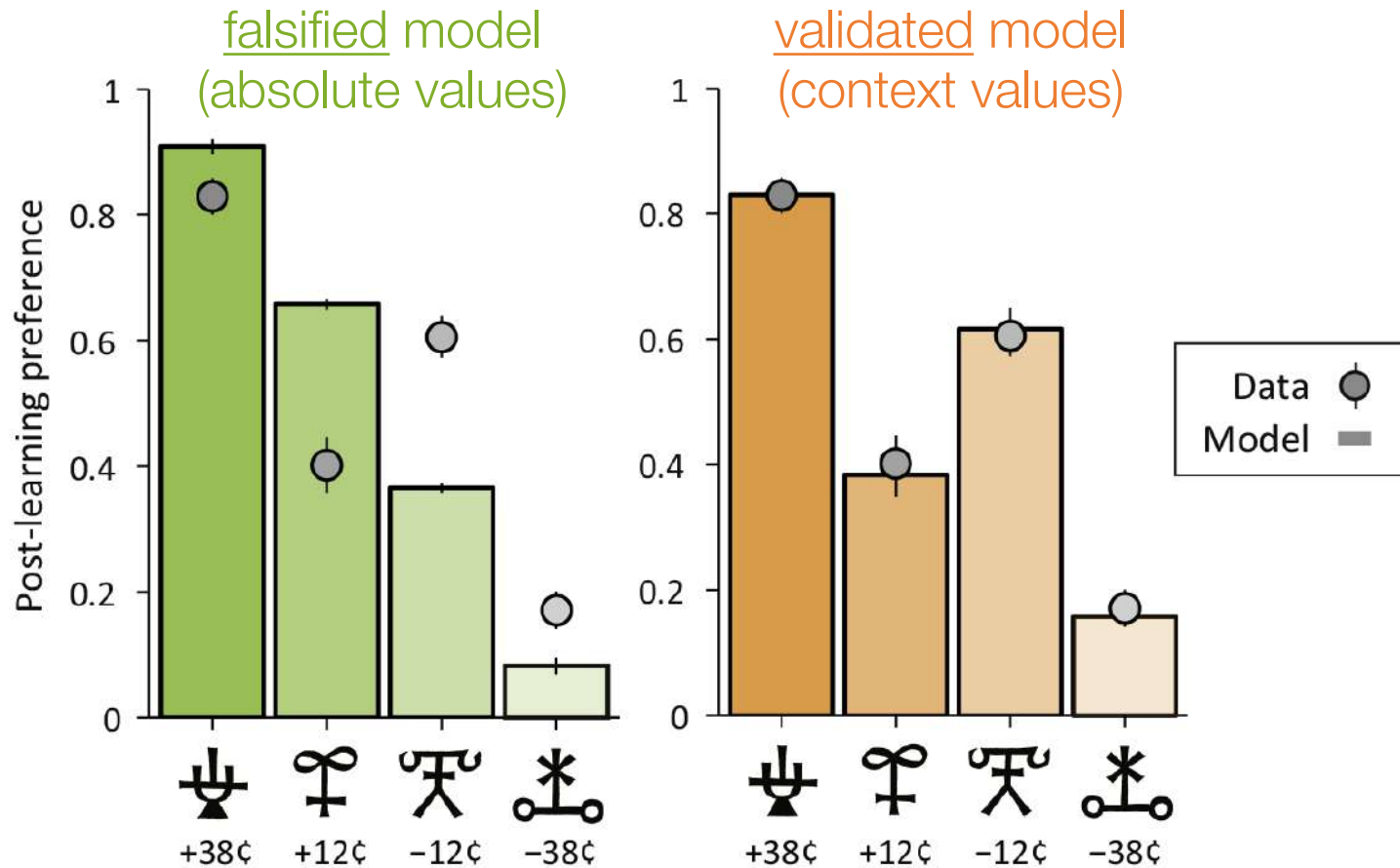
# Modeling guidelines

- Model **falsification**: showing through simulations that a model is not able to generate a specific behavioral effect of interest.



# Modeling guidelines

- Model falsification:



# Modeling guidelines



## ARTICLE

Received 10 Dec 2014 | Accepted 14 Jul 2015 | Published 25 Aug 2015

DOI: 10.1038/ncomms9096

OPEN

# Contextual modulation of value signals in reward and punishment learning

Stefano Palminteri<sup>1,2</sup>, Mehdi Khamassi<sup>3,4</sup>, Mateus Joffily<sup>4,5</sup> & Giorgio Coricelli<sup>2,4,6</sup>

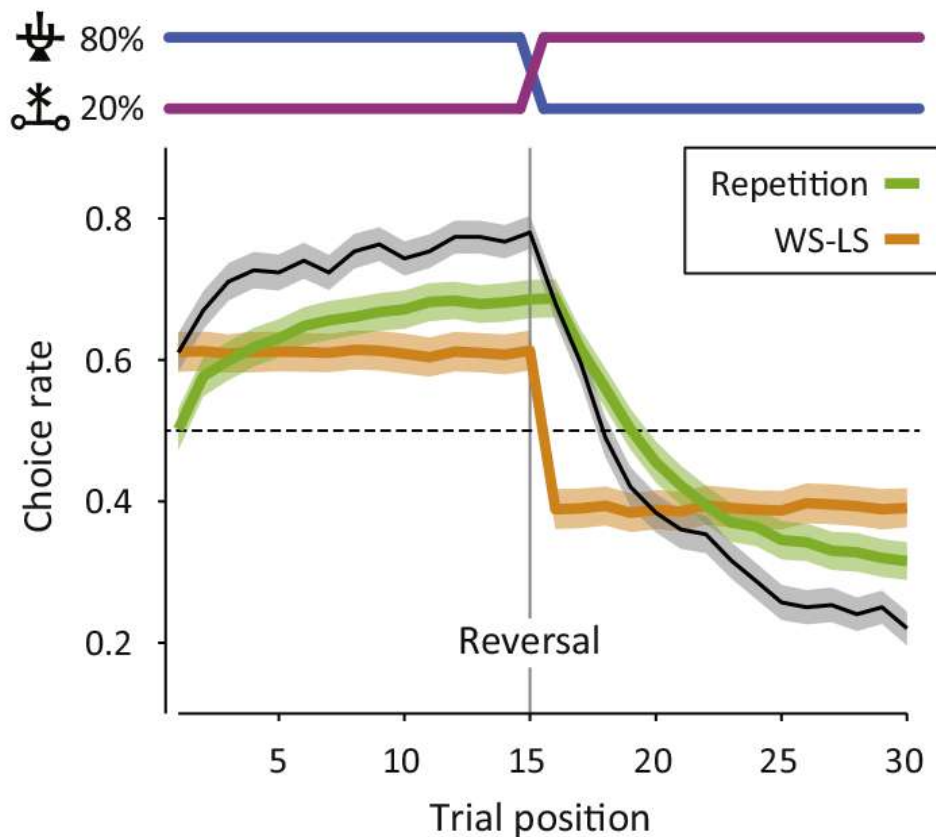
Compared with reward seeking, punishment avoidance learning is less clearly understood at both the computational and neurobiological levels. Here we demonstrate, using computational modelling and fMRI in humans, that learning option values in a relative—context-dependent—scale offers a simple computational solution for avoidance learning. The context (or state) value sets the reference point to which an outcome should be compared before updating the option value. Consequently, in contexts with an overall negative expected

# Modeling guidelines

- Qualitative vs quantitative differences:
  - ✓ qualitative = specific pattern of behavior as in the last two examples
  - ✓ quantitative = magnitude of deviations btw observed and fitted behavior

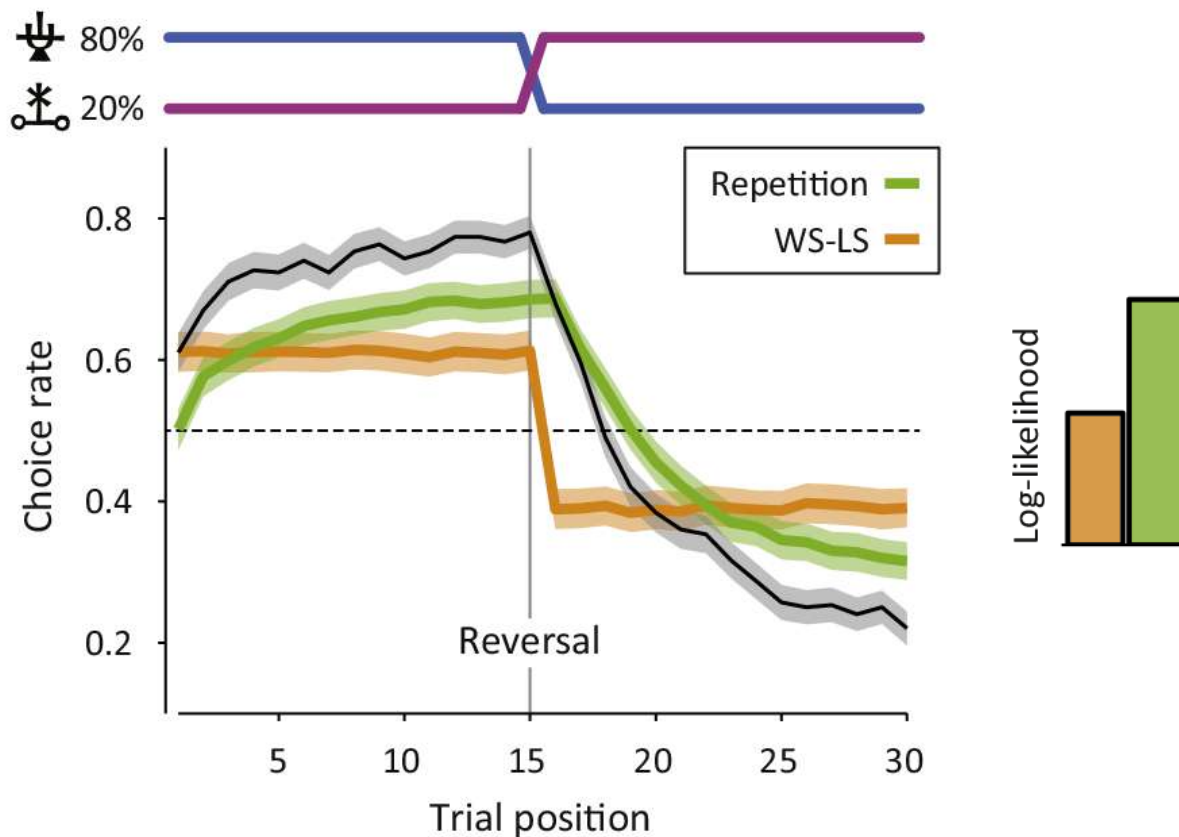
# Modeling guidelines

- Qualitative vs quantitative differences:



# Modeling guidelines

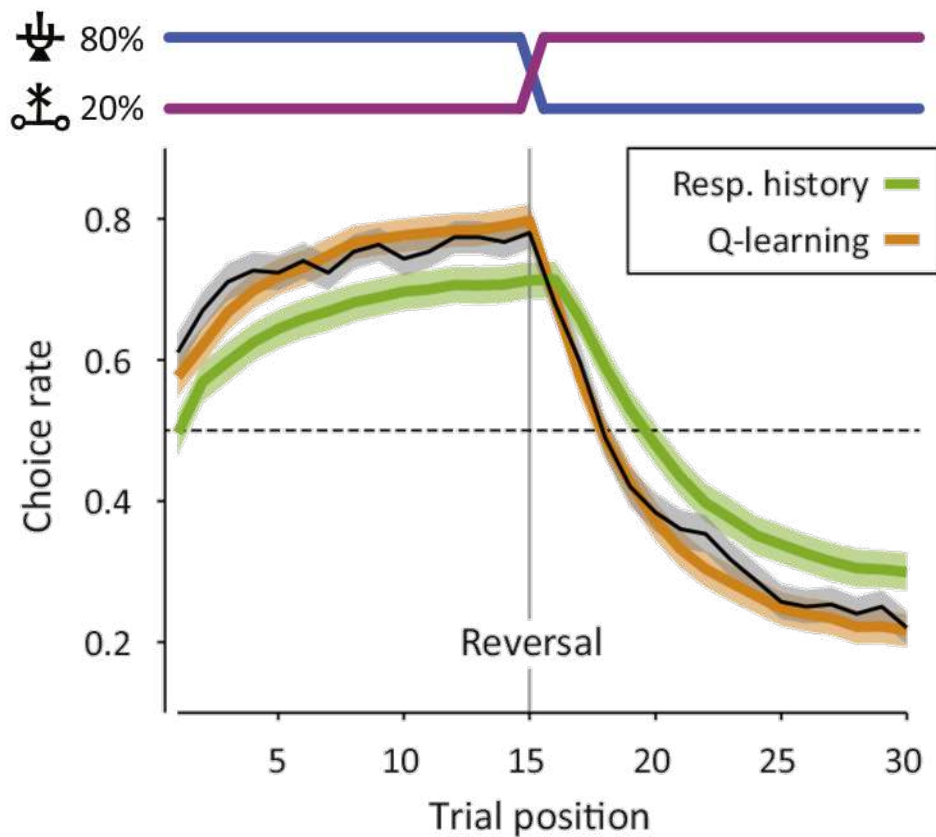
- Qualitative vs quantitative differences:





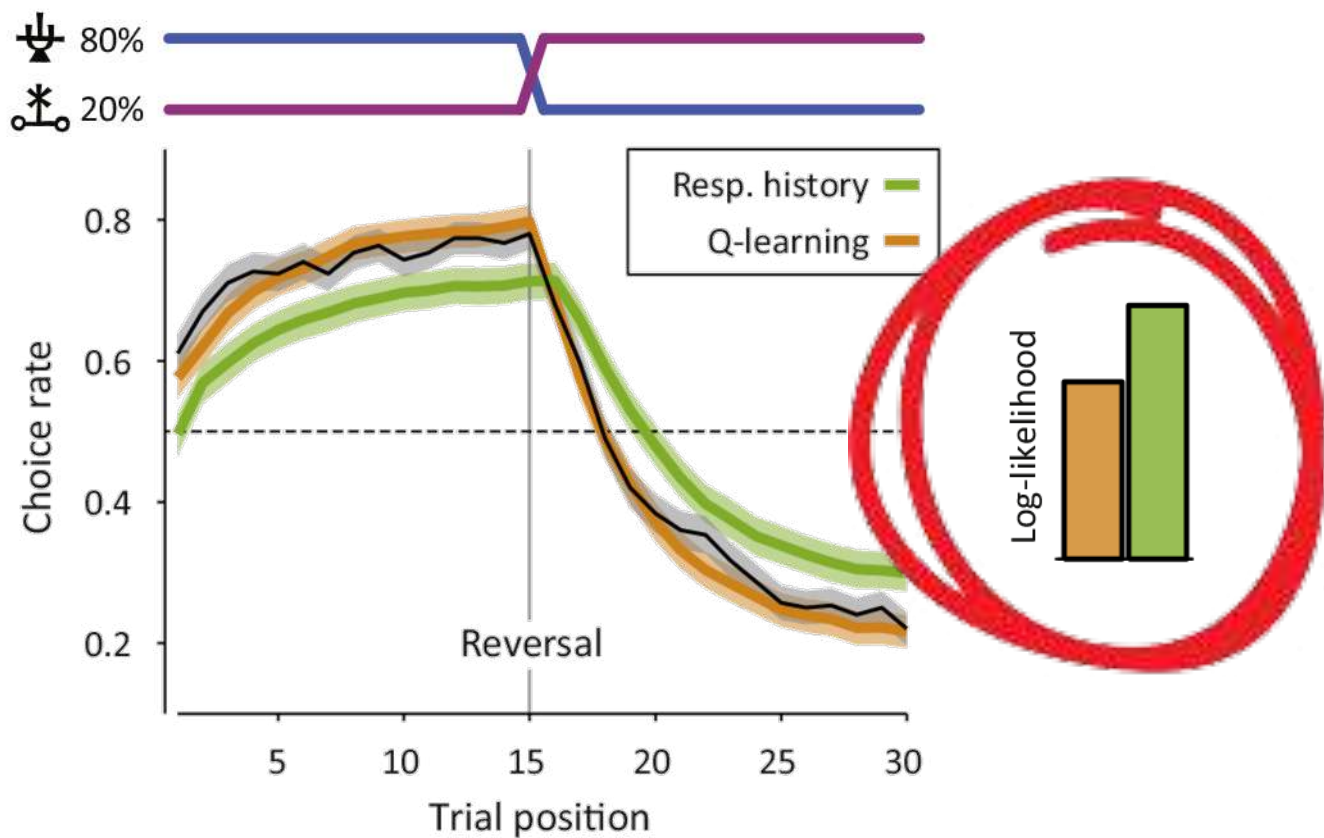
# Modeling guidelines

- Qualitative vs quantitative differences:



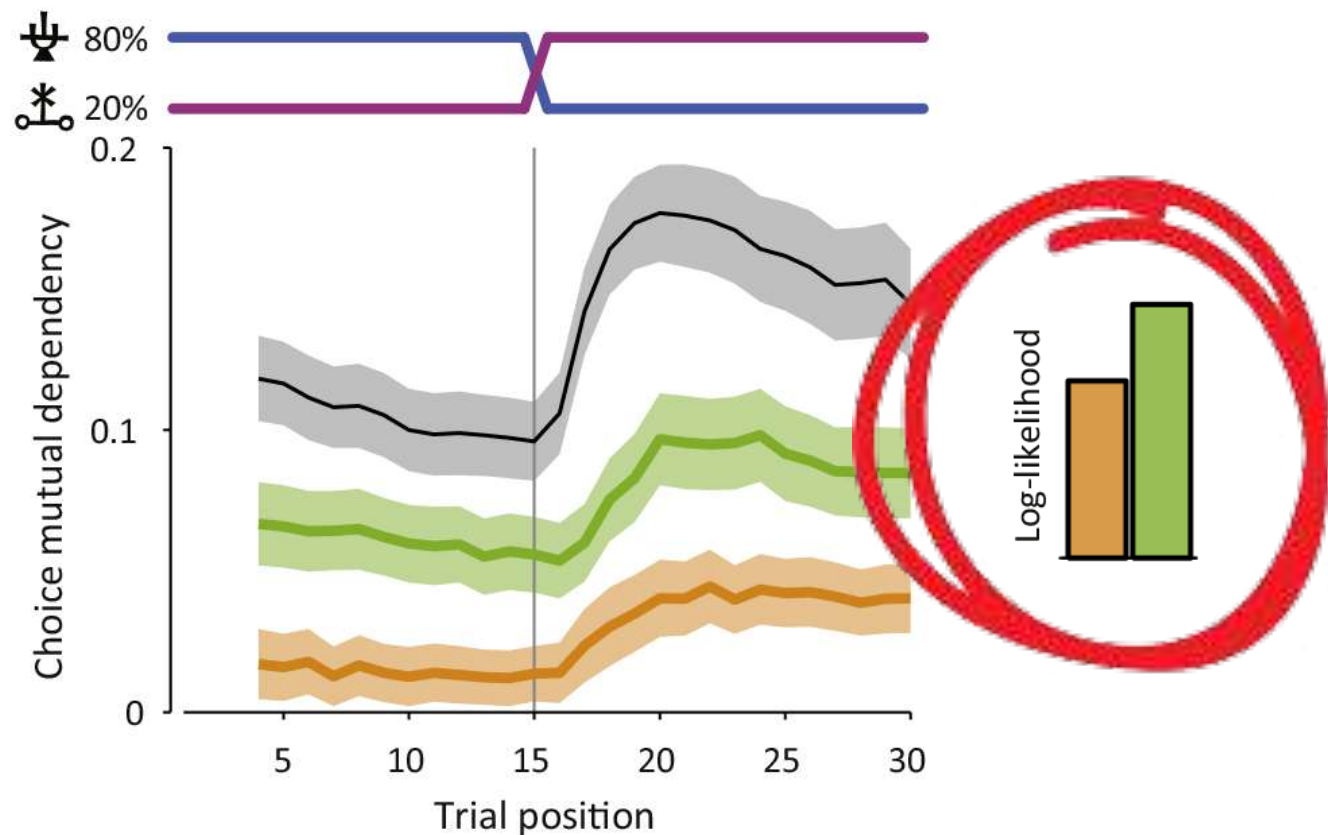
# Modeling guidelines

- Qualitative vs quantitative differences:



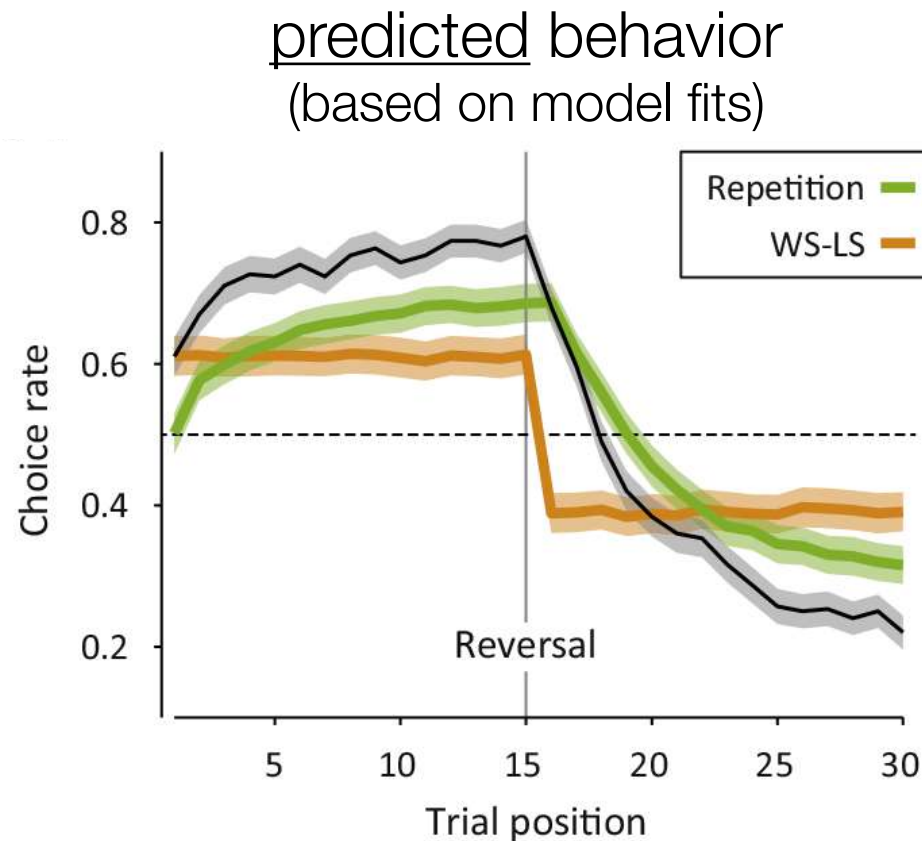
# Modeling guidelines

- Qualitative vs quantitative differences:



# Modeling guidelines

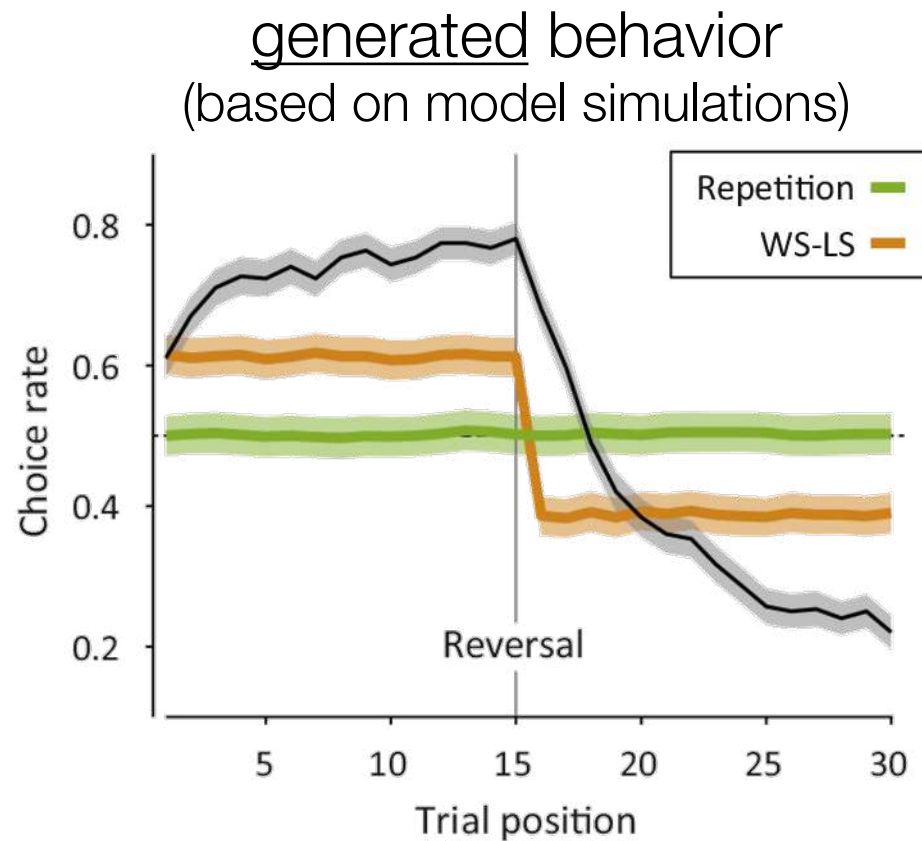
- Qualitative vs quantitative differences:



repetition better  
than WS-LS

# Modeling guidelines

- Qualitative vs quantitative differences:



repetition at chance level

# Modeling guidelines

- Take-home messages:
  - ✓ a **specific** behavioral effect may **not explain why** a certain model fits behavior better
  - ✓ a certain model may **fit behavior well**, but **fail when simulated** to perform the task

# Modeling guidelines

- Model **fitting**: finding the set of parameters  $\theta$  that best accounts for the observed behavior
- Unlike **empirical metrics** like task accuracy, **fitted parameters** account for task variables (such as task difficulty):

$$\hat{\theta}_{\text{MLE}} = \operatorname{argmax}_{\theta} \left( \log(p(\text{behavior}|\theta, s)) \right)$$

- MLE = **Maximum Likelihood Estimate** (set of parameters that maximize the likelihood of the observed behavior given task variables)

# Modeling guidelines

- Model **comparison**: identifying which of a set of alternative models best describes the same observed behavior given task variables
- This comparison uses the log-likelihood (or the log-posterior) of alternative models as measure of model evidence:

$$\text{MLE} = \max_{\theta} \left( \log(p(\text{behavior}|\theta, s)) \right)$$

- BMS = **Bayesian Model Selection** (statistical procedure used for model comparison)



# Modeling guidelines

- Model **parameter estimation**: often a synonym for model **fitting** (finding the set of parameters  $\theta$  that best accounts for the observed behavior)
- Fitted parameters can be used as a **algorithmic phenotype** for investigating:
  - ✓ **individual differences** in the same task
  - ✓ effect of **drugs, lesions, moods, illness**
  - ✓ effect of **experimental conditions**
  - ✓ effect of **practice**

# Modeling guidelines

*Personality Neuroscience*

[cambridge.org/pen](http://cambridge.org/pen)

## Review Paper

**Cite this article:** Patzelt EH, Hartley CA, Gershman SJ. (2018) Computational Phenotyping: Using Models to Understand Individual Differences in Personality, Development, and Mental Illness. *Personality Neuroscience*. Vol 1: e18, XX: 1-10. doi:10.1017/pen.2018.14

Inaugural Invited Paper  
Accepted: 6 June 2018

### Key words:

psychopathology (general); reward/punishment; computational models; learning; decision making

### Author for correspondence:

Edward H. Patzelt, E-mail: [patzelt@fas.harvard.edu](mailto:patzelt@fas.harvard.edu)

## Computational Phenotyping: Using Models to Understand Individual Differences in Personality, Development, and Mental Illness

Edward H. Patzelt<sup>1</sup>, Catherine A. Hartley<sup>2</sup> and Samuel J. Gershman<sup>1</sup>

<sup>1</sup>Department of Psychology and Center for Brain Science, Harvard University, Cambridge, MA, USA and

<sup>2</sup>Department of Psychology and Center for Neural Science, New York University, New York, NY, USA

### Abstract

This paper reviews progress in the application of computational models to personality, developmental, and clinical neuroscience. We first describe the concept of a computational phenotype, a collection of parameters derived from computational models fit to behavioral and neural data. This approach represents individuals as points in a continuous parameter space, complementing traditional trait and symptom measures. One key advantage of this representation is that it is mechanistic: The parameters have interpretations in terms of cognitive processes, which can be translated into quantitative predictions about future behavior and brain activity. We illustrate with several examples how this approach has led to new scientific insights into individual differences, developmental trajectories, and psychopathology. We then survey some of the challenges that lay ahead.

The study of personality has a rich history examining individual differences in how we behave, relate to ourselves and each other, and understand our experiences and environment. This work has had the significant challenge of linking multiple levels of analysis spanning complex neural and cognitive processes. Recently, computational models have provided a powerful tool to mathematically formalize this complexity, and provide rich descriptions of the processes

# Modeling guidelines

nature human behaviour



Article

<https://doi.org/10.1038/s41562-024-01814-x>





## Dynamic computational phenotyping of human cognition

Received: 30 September 2023

Accepted: 21 December 2023

Published online: 08 February 2024

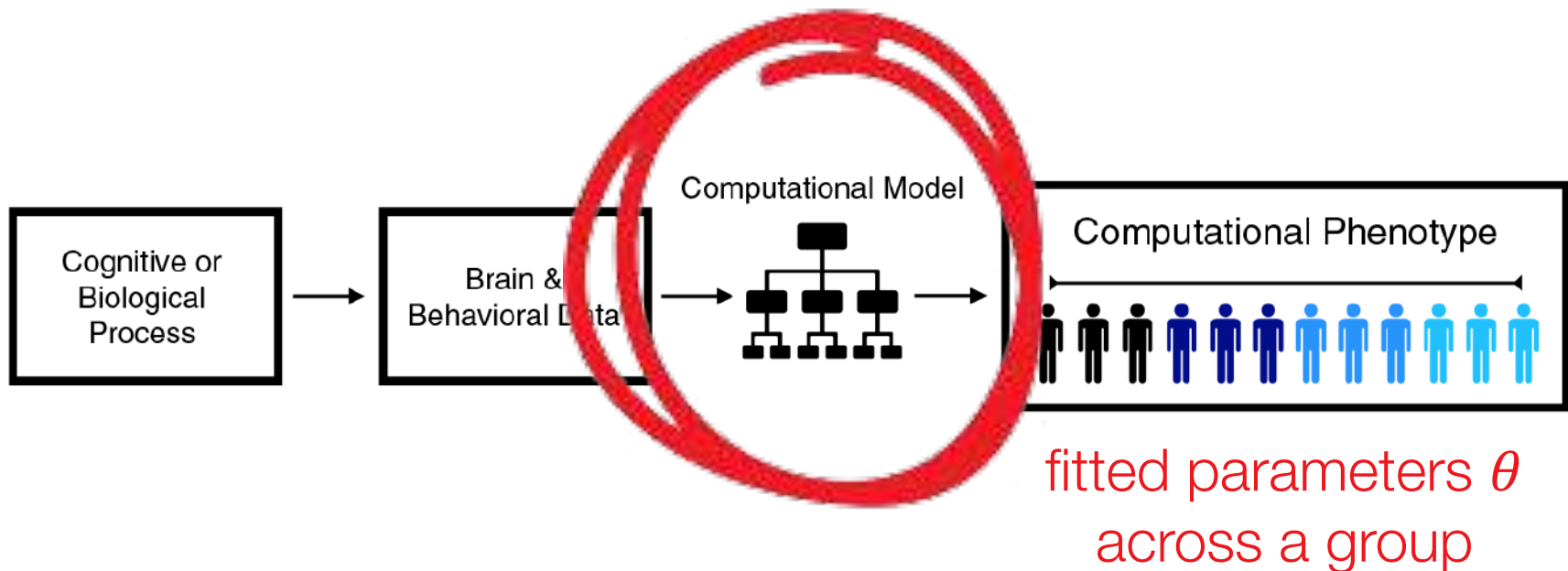
 Check for updates

Roey Schurr <sup>1,7</sup>✉, Daniel Reznik <sup>2,7</sup>✉, Hanna Hillman <sup>3</sup>, Rahul Bhui <sup>4,5</sup> & Samuel J. Gershman<sup>1,6</sup>

Computational phenotyping has emerged as a powerful tool for characterizing individual variability across a variety of cognitive domains. An individual's computational phenotype is defined as a set of mechanistically interpretable parameters obtained from fitting computational models to behavioural data. However, the interpretation of these parameters hinges critically on their psychometric properties, which are rarely studied. To identify the sources governing the temporal variability of the computational phenotype, we carried out a 12-week longitudinal study using a battery of seven tasks that measure aspects of human learning

# Modeling guidelines

- Model **parameter estimation**: often a synonym for model **fitting** (finding the set of parameters  $\theta$  that best accounts for the observed behavior)



# Modeling guidelines

- Model **latent variable estimation**: finding the hidden variables  $x_i$  computed by the model that best account for the observed behavior
- Hidden variables  $x_i$  can be used to **validate** specific aspects of the model using additional observables (e.g., physiological data).
- Hidden variables  $x_i$  are often used to identify **neural correlates** of the model.  
Remember what reverse inference means?

# Modeling guidelines

- Example: the multi-armed bandit task



# Modeling guidelines

nature

Vol 441 | 15 June 2006 | doi:10.1038/nature04766

## LETTERS

### Cortical substrates for exploratory decisions in humans

Nathaniel D. Daw<sup>1\*</sup>, John P. O'Doherty<sup>2\*†</sup>, Peter Dayan<sup>1</sup>, Ben Seymour<sup>2</sup> & Raymond J. Dolan<sup>2</sup>

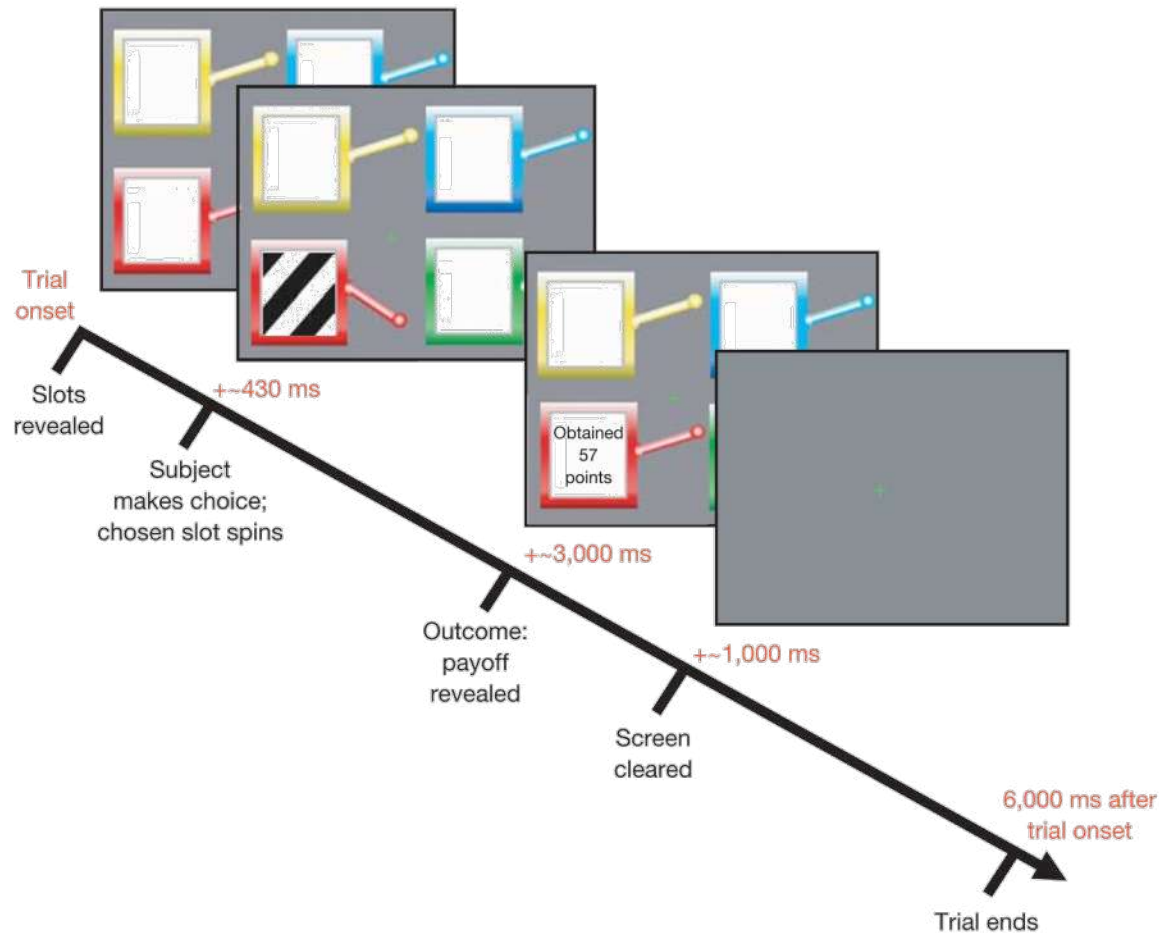
Decision making in an uncertain environment poses a conflict between the opposing demands of gathering and exploiting information. In a classic illustration of this 'exploration–exploitation' dilemma<sup>1</sup>, a gambler choosing between multiple slot machines balances the desire to select what seems, on the basis of accumulated experience, the richest option, against the desire to choose a less familiar option that might turn out more advantageous (and thereby provide information for improving future decisions). Far from representing idle curiosity, such exploration is often critical for organisms to discover how best to harvest resources such as food and water. In appetitive choice, substantial experimental evidence, underpinned by computational reinforcement learning<sup>2</sup> (RL) theory, indicates that a dopaminergic<sup>3,4</sup>, striatal<sup>5–9</sup> and medial prefrontal network mediates learning to exploit. In contrast,

through sampling it actively. This feature of the experimental design, together with a model-based analysis, allowed us to study exploratory and exploitative decisions under uniform conditions, in the context of a single task.

We asked subjects in post-task interviews to describe their choice strategies. The majority (11 of 14) reported occasionally trying the different slots to work out which currently had the highest payoffs (exploring) while at other times choosing the slot they thought had the highest payoffs (exploiting). To investigate this behaviour quantitatively, we considered RL (ref. 2) strategies for exploration. These strategies come in three flavours, differing in how exploratory actions are directed. The simplest method, known as ' $\epsilon$ -greedy', is undirected: it chooses the 'greedy' option (the one believed to be best) most of the time, but occasionally (with probability  $\epsilon$ ) substitutes a random action. A more sophisticated approach is to guide exploration

# Modeling guidelines

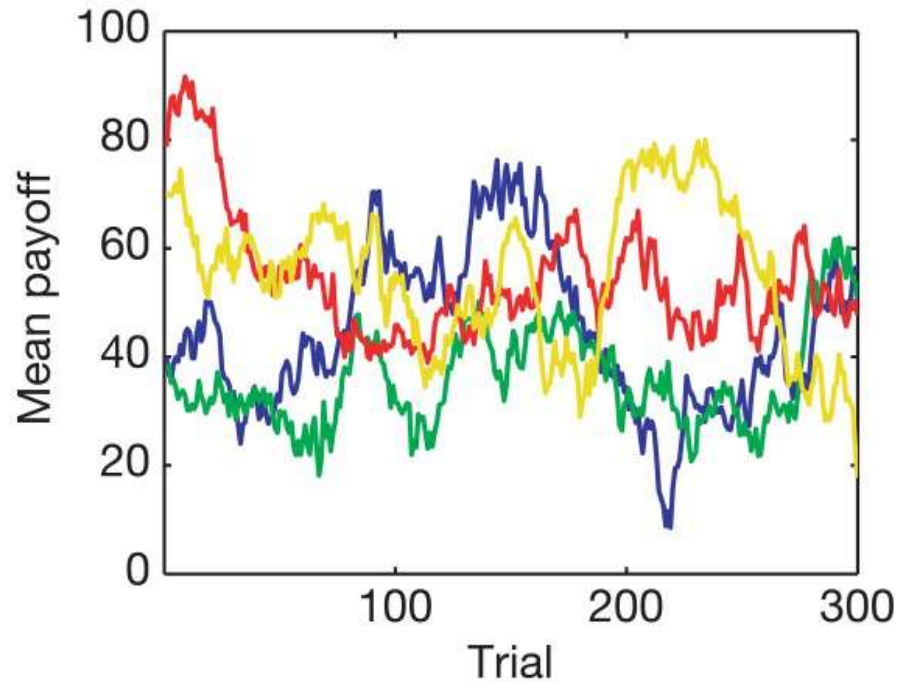
- Example: the multi-armed bandit task





# Modeling guidelines

- Example: the multi-armed bandit task



# Modeling guidelines

- Example: the multi-armed bandit task

We compared the fit of three distinct RL models, embodying the aforementioned strategies, to our subjects' behavioural choices. All the models learned the values of actions with the use of a Kalman filter (see Supplementary Methods), an error-driven prediction algorithm that generalizes the temporal-difference learning algorithm (used in most RL theories of dopamine) by also tracking uncertainty about the value of each action. The models differed only in their choice rules. We compared models by using the likelihood of the subjects' choices given their experience, optimized over free parameters. This comparison (Supplementary Tables 1 and 2) revealed strong evidence for value-sensitive (softmax) over undirected ( $\epsilon$ -greedy) exploration. There was no evidence to justify the introduction of an extra parameter that allowed exploration to be directed towards uncertainty (softmax with an uncertainty bonus): at optimal fit, the bonus was negligible, making the model equivalent to the simpler softmax. We conducted additional model fits (see Supplementary Information) to verify that these findings were not an artefact of our assumptions about the yoking of free parameters between subjects.

# Modeling guidelines

- Example: the multi-armed bandit task
- What was done by the authors?
  - ✓ model simulation?

# Modeling guidelines

- Example: the multi-armed bandit task
- What was done by the authors?
  - ✓ ~~model simulation?~~ **NO**
  - ✓ model fitting? YES
  - ✓ model comparison? YES
  - ✓ model parameter estimation? YES

# Modeling guidelines

- Example: the multi-armed bandit task

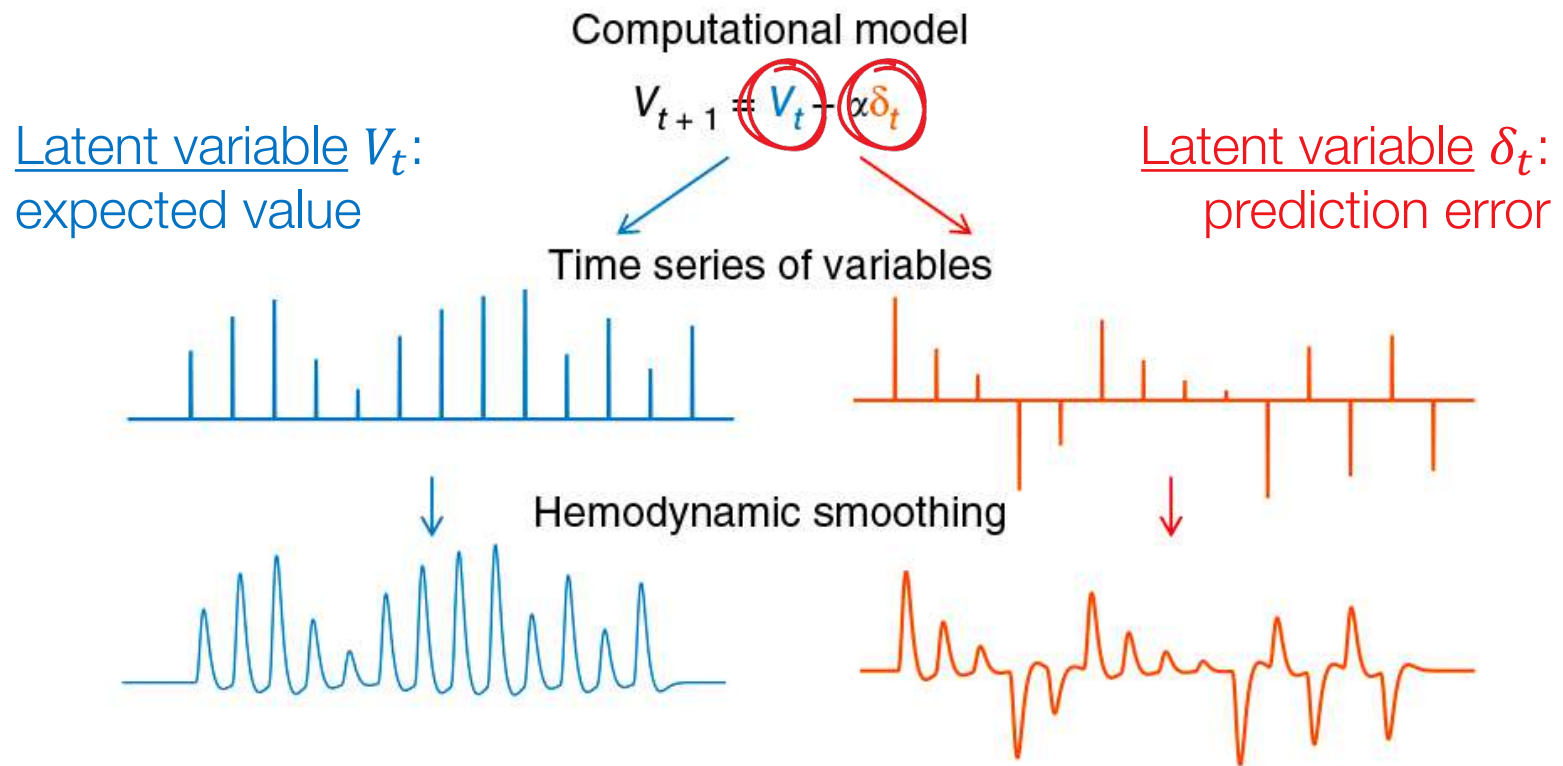
Having characterized subjects' behaviour computationally, we used the best-fitting softmax model to generate regressors containing value predictions, prediction errors and choice probabilities for each subject on each trial. We used statistical parametric mapping to identify brain regions in which neural activity was significantly correlated with the model's internal signals. Consistent with previous studies<sup>7-9</sup> was our observation that a prediction error was correlated significantly with activity in both the ventral and dorsal striatum (see Supplementary Table 3).

# Modeling guidelines

- Example: the multi-armed bandit task
- What was done by the authors?
  - ✓ ~~model simulation?~~ **NO**
  - ✓ model fitting? YES
  - ✓ model comparison? YES
  - ✓ model parameter estimation? YES
  - ✓ model latent variable estimation? YES

# Modeling guidelines

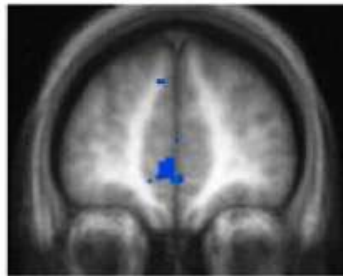
- Example: the multi-armed bandit task



# Modeling guidelines

- Example: the multi-armed bandit task

Latent variable  $V_t$ :  
expected value



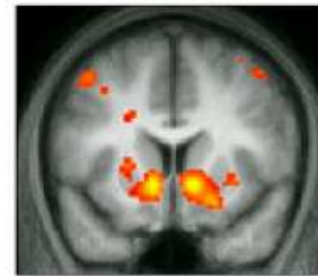
ventromedial  
prefrontal cortex

Computational model

$$V_{t+1} = V_t + \alpha \delta_t$$

Correlated regions

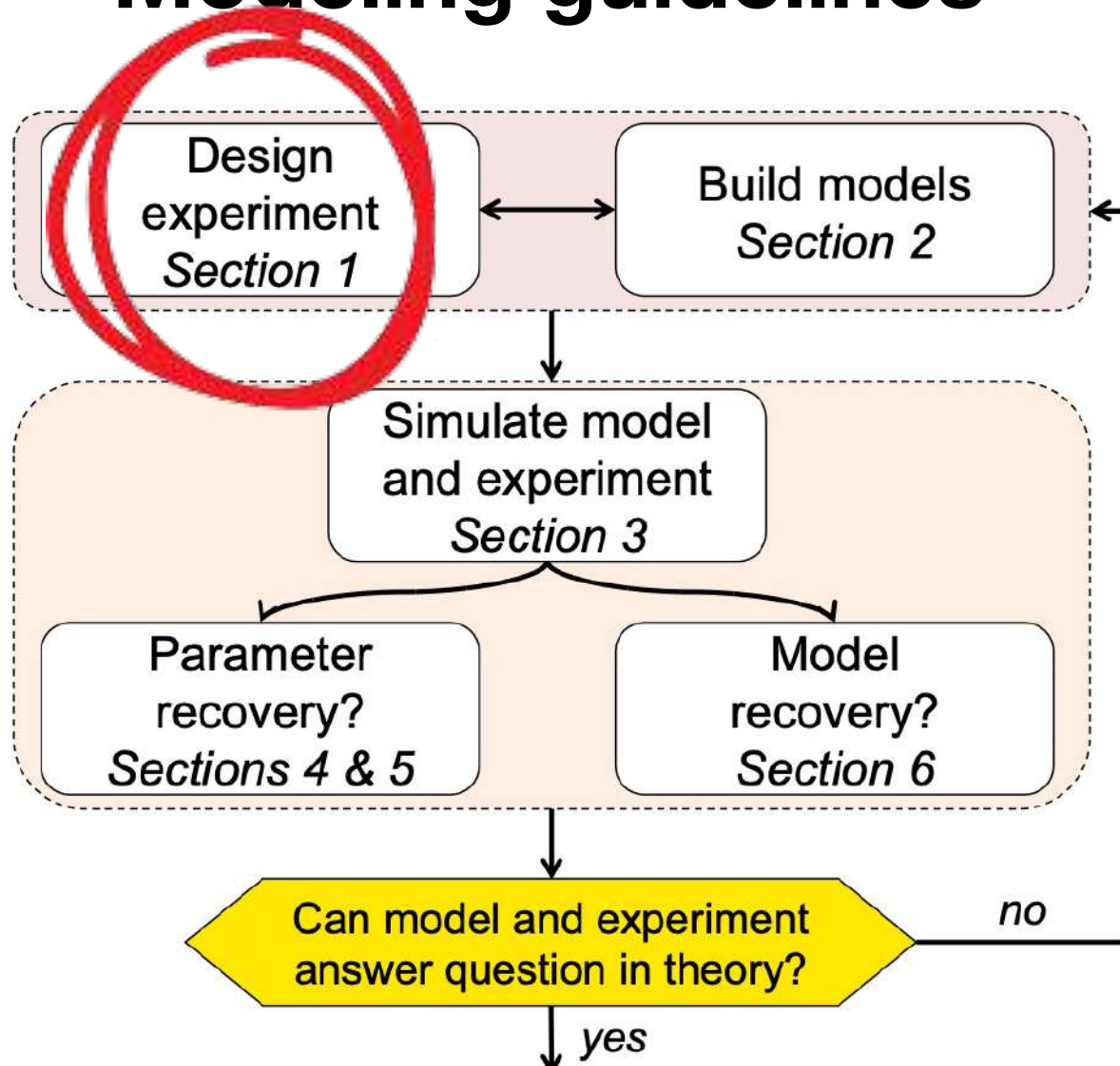
Latent variable  $\delta_t$ :  
prediction error



ventral  
striatum



# Modeling guidelines

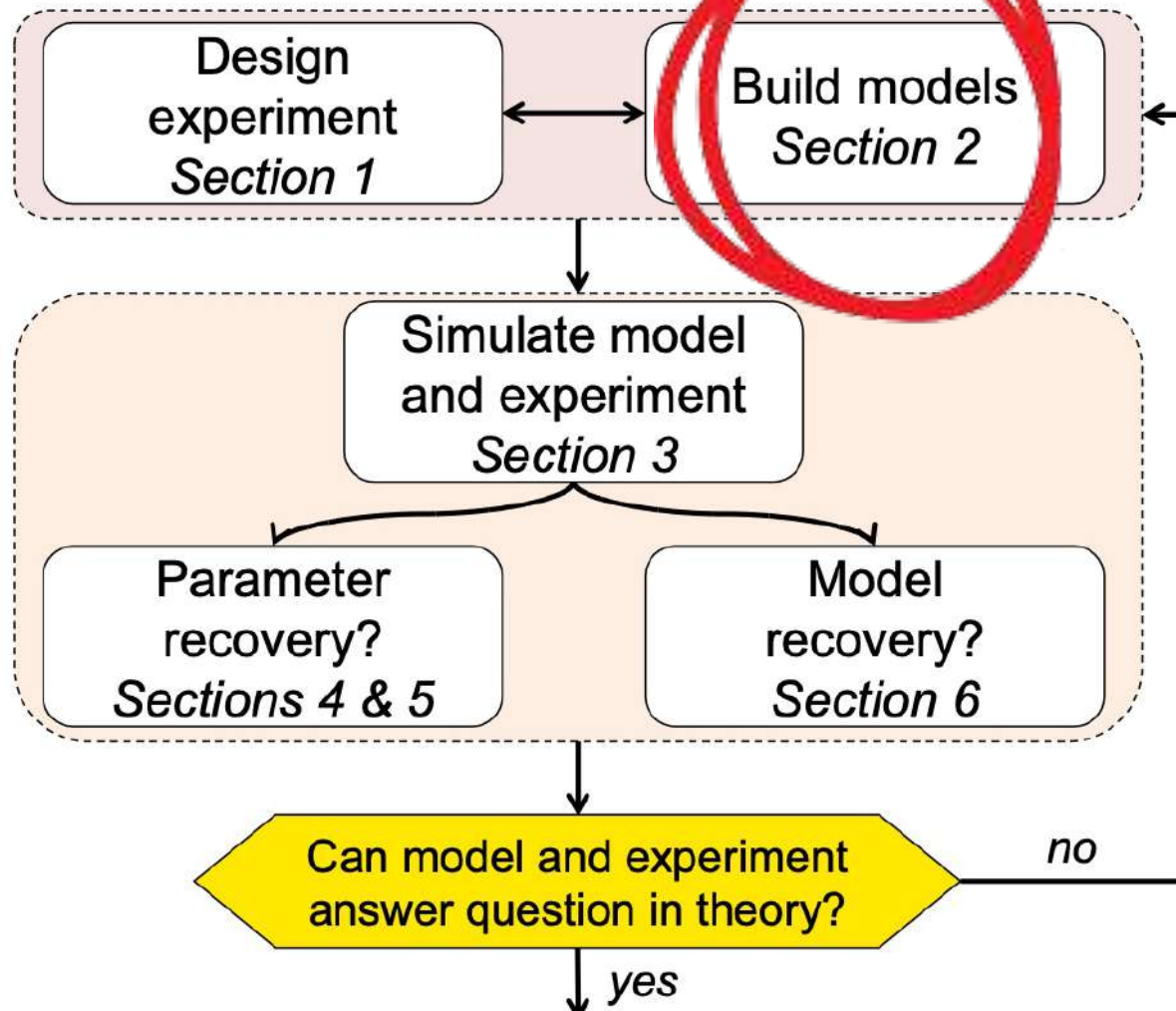


# Modeling guidelines

## **Box 2. Example: simulating behavior in the bandit task.**

To simulate behavior, we first need to define the parameters of the task. These include the total number of trials,  $T$  (=1000 in the example), as well as the number of bandits,  $K$  (= 2), and the reward probability for each bandit,  $\mu^k$  (0.2 and 0.8 for bandits 1 and 2, respectively). The experiment parameters, as used in the simulation, should match the actual parameters used in the experiment.

# Modeling guidelines



# Modeling guidelines

## Box 2. Example: simulating behavior in the bandit task.

To simulate behavior, we first need to define the parameters of the task. These include the total number of trials,  $T$  (=1000 in the example), as well as the number of bandits,  $K$  (= 2), and the reward probability for each bandit,  $\mu^k$  (0.2 and 0.8 for bandits 1 and 2, respectively). The experiment parameters, as used in the simulation, should match the actual parameters used in the experiment.

## Box 1. Example: Modeling behavior in the multi-armed bandit task.

We consider five different models of how participants could behave in the multi-armed bandit task.

Model 1: Random responding

Model 2: Noisy win-stay-lose-shift

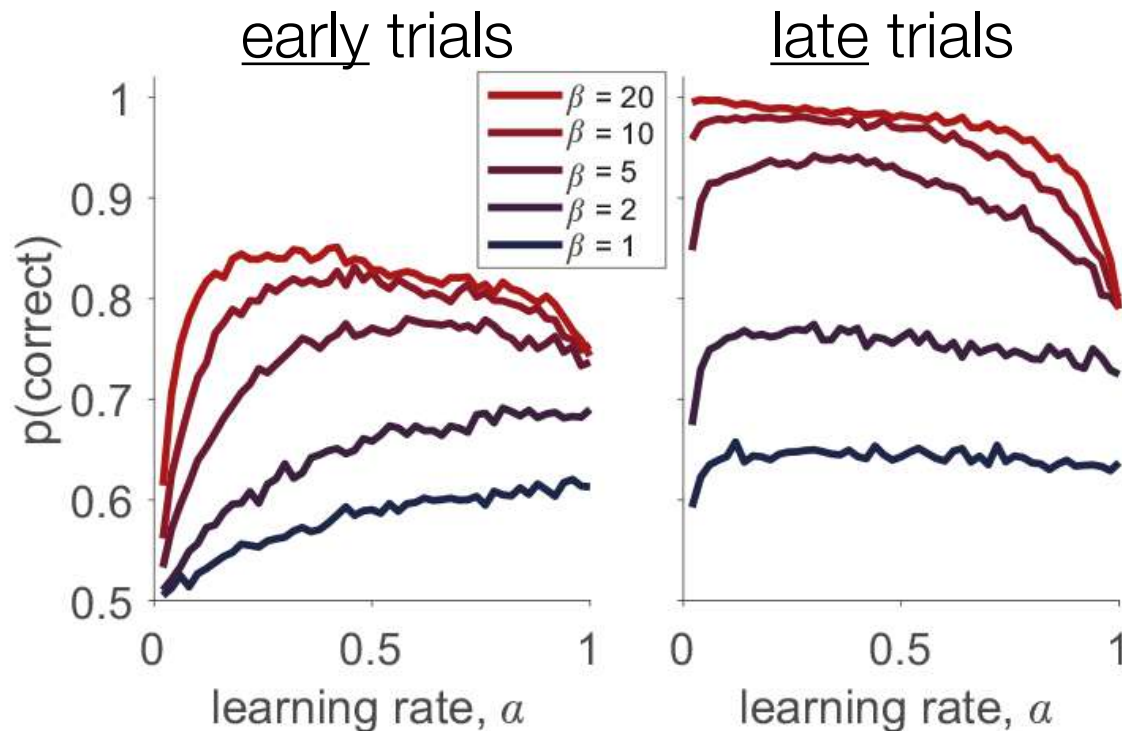
Model 3: Rescorla Wagner

Model 4: Choice kernel

Model 5: Rescorla Wagner + choice kernel

# Modeling guidelines

- Simulate **TD-based RL** in the bandit task with  $K = 2$  arms and fixed reward probabilities

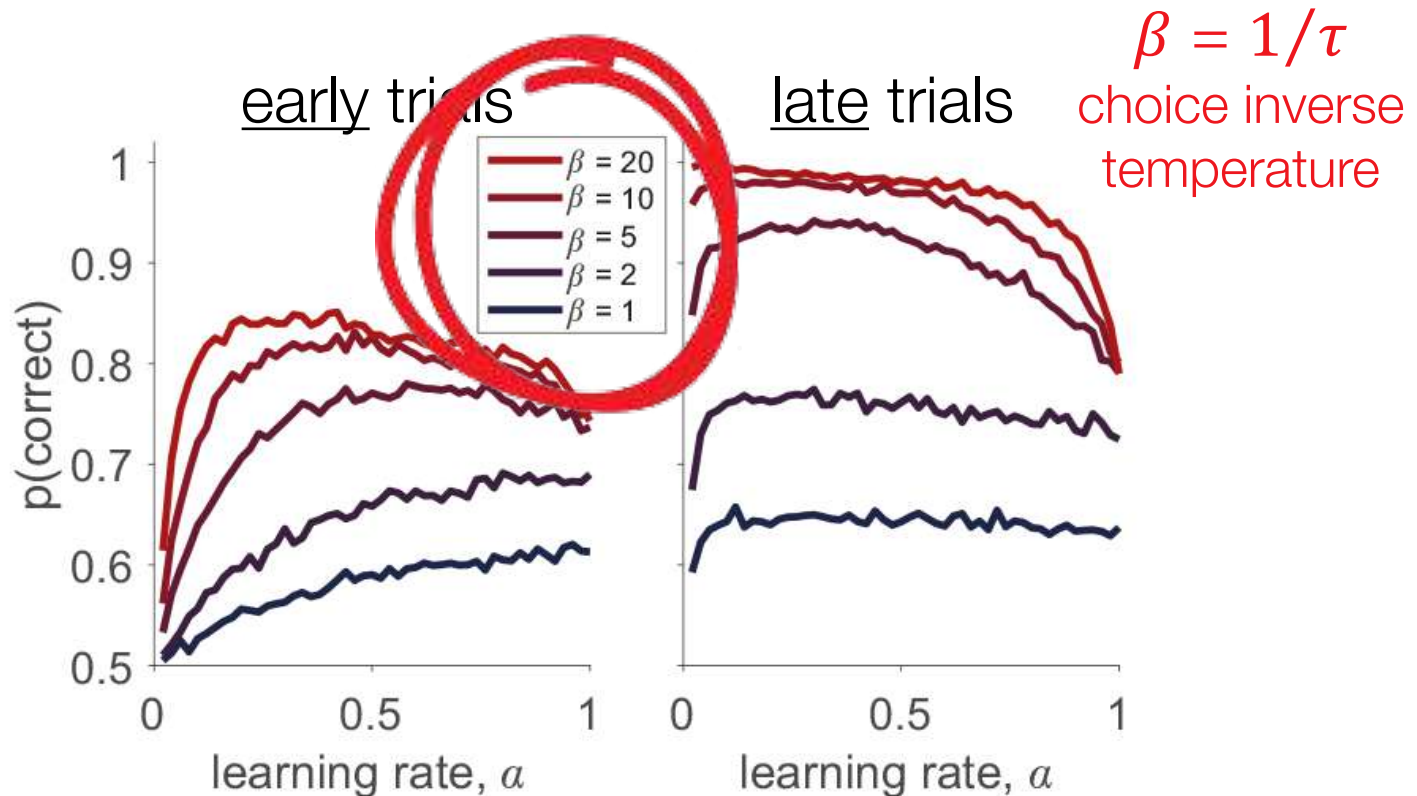


# Modeling guidelines

- Simulate TD-based RL in the bandit task with  $K = 2$  arms and fixed reward probabilities
- Questions:
  - 1/ what behavioral feature?
  - 2/ what are the axes and the different curves?
  - 3/ what are the two panels showing?
- What is the relation between model parameters and performance in this bandit task?

# Modeling guidelines

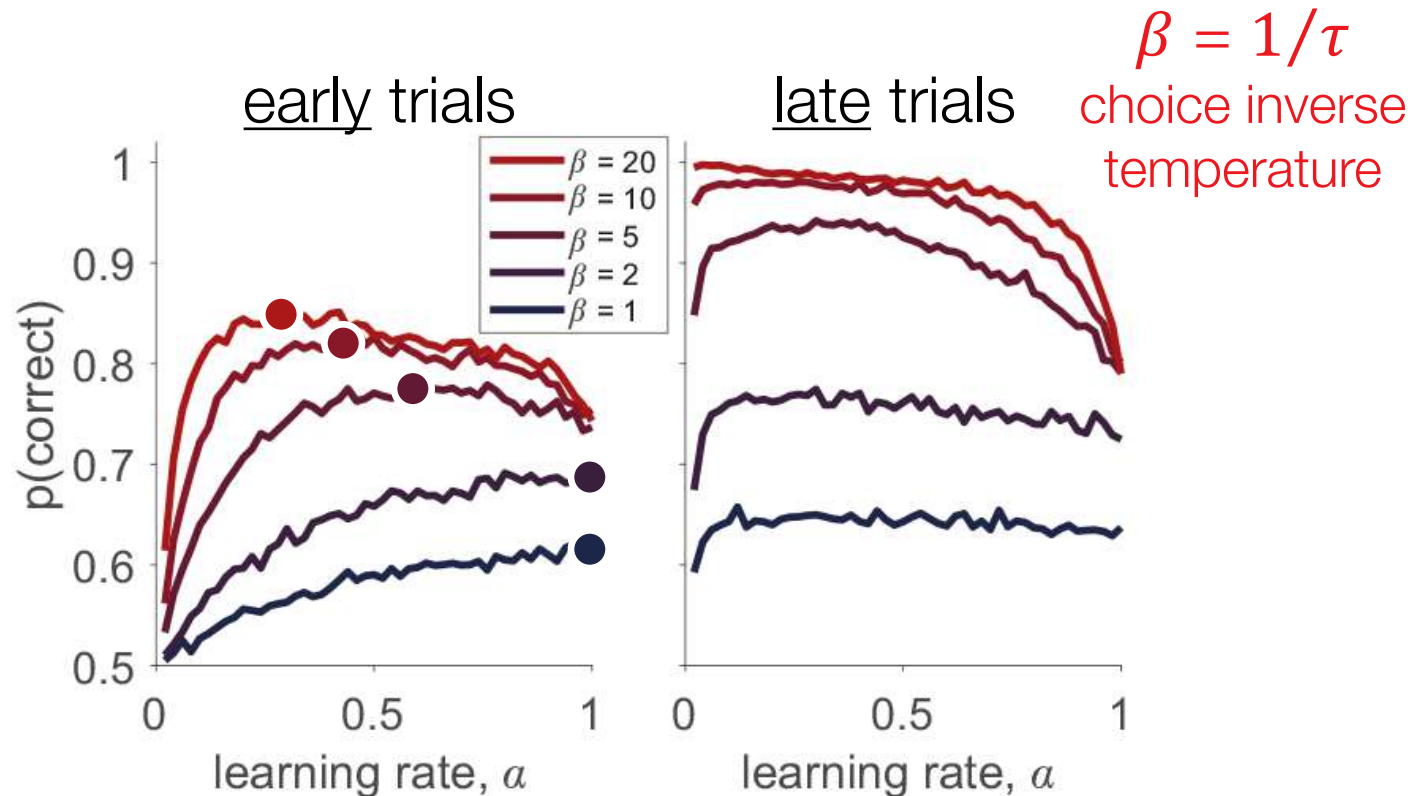
- Simulate **TD-based RL** in the bandit task with  $K = 2$  arms and fixed reward probabilities





# Modeling guidelines

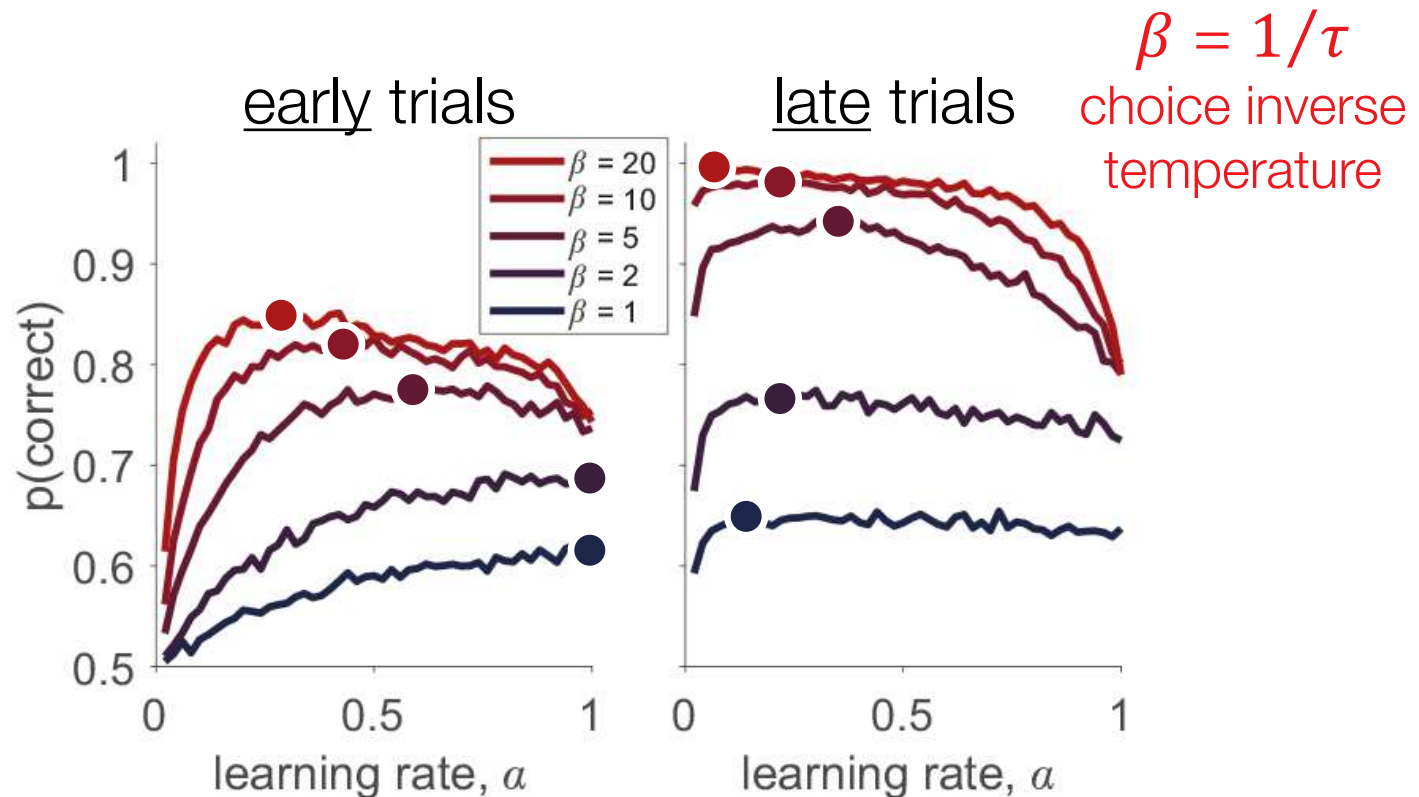
- Simulate **TD-based RL** in the bandit task with  $K = 2$  arms and fixed reward probabilities





# Modeling guidelines

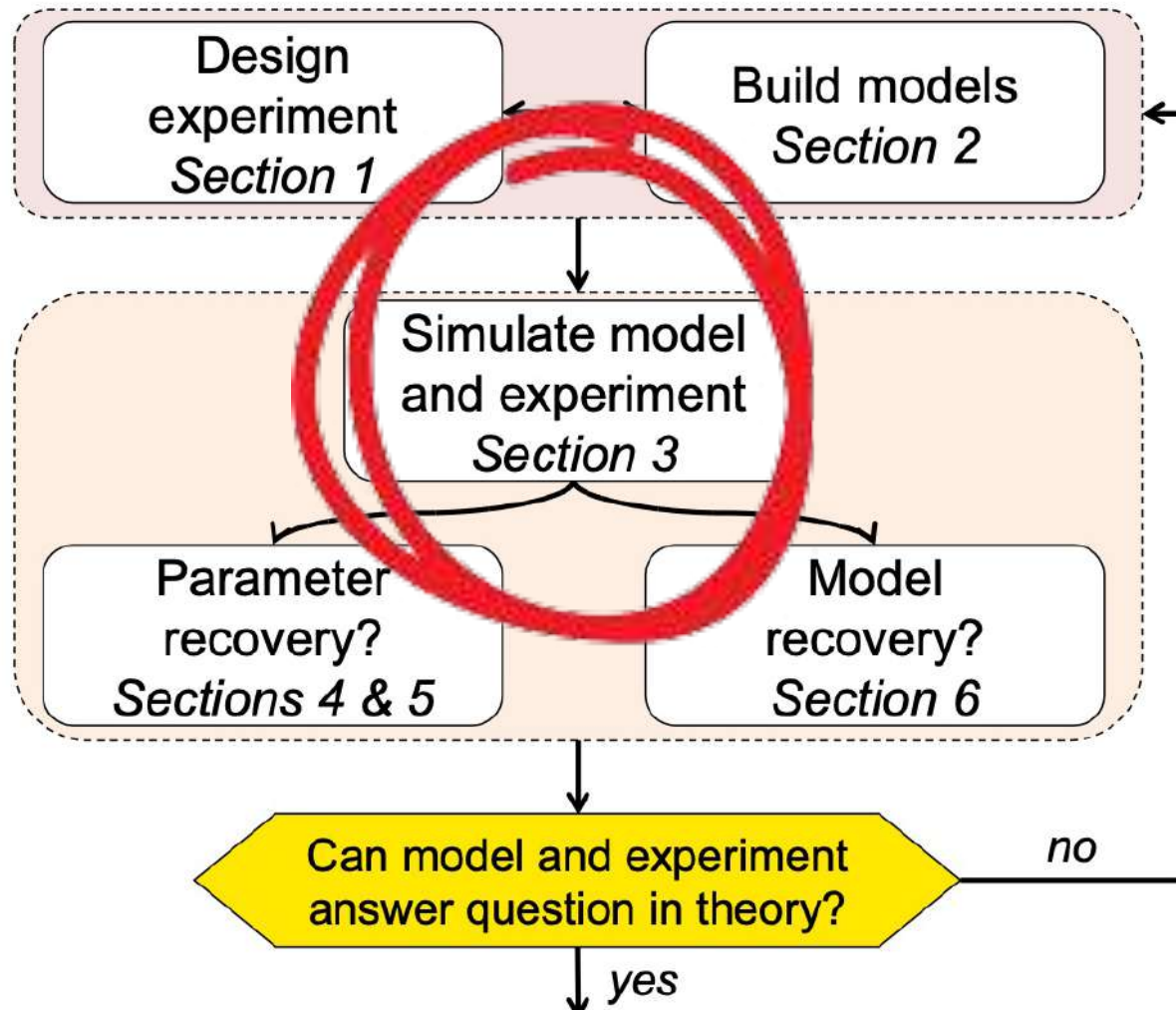
- Simulate **TD-based RL** in the bandit task with  $K = 2$  arms and fixed reward probabilities



# Modeling guidelines

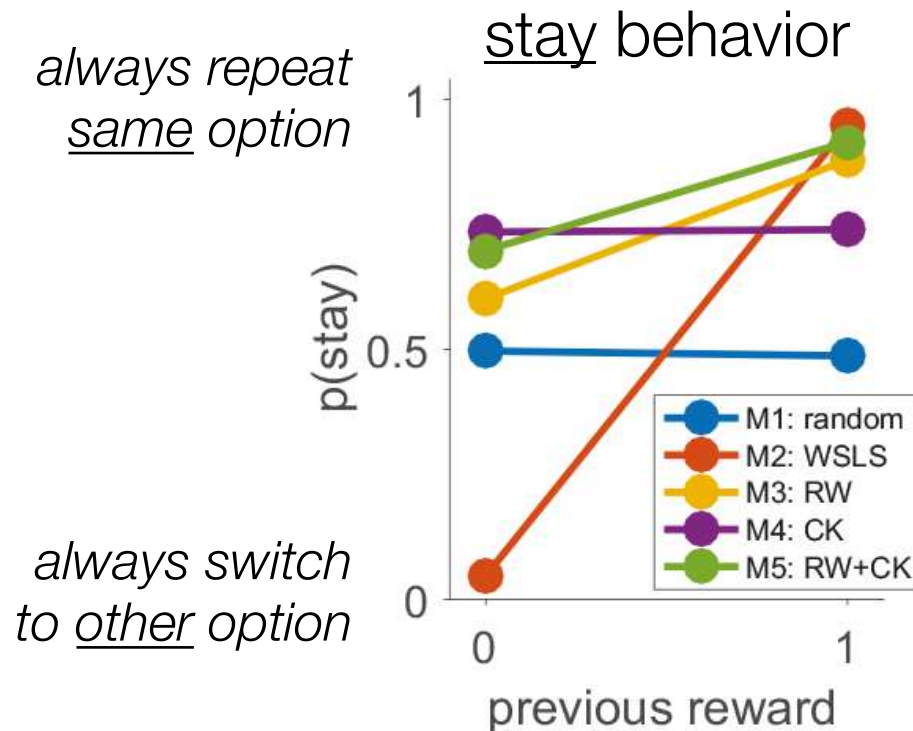
- Simulate **TD-based RL** in the bandit task with  $K = 2$  arms and fixed reward probabilities
- $p(\text{correct})$  increases monotonically with  $\beta$   
Conclusion: a ‘greedy’ policy ( $\beta \rightarrow \infty$ ) yields better performance in this bandit task.
- The best learning rate  $\alpha$  depends on  $\beta$ , and differently so, in early vs late trials!  
Conclusion: greedy agents should learn differently than exploratory agents in this bandit task.

# Modeling guidelines



# Modeling guidelines

- Simulate **TD-based RL** in the bandit task with  $K = 2$  arms and fixed reward probabilities

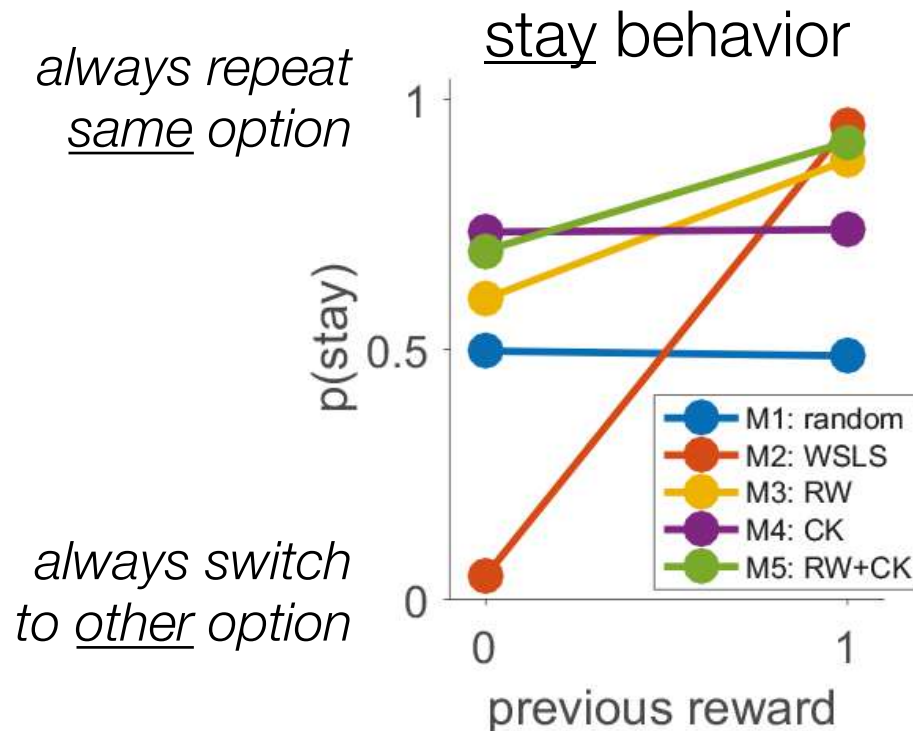


# Modeling guidelines

- Simulate **TD-based RL** in the bandit task with  $K = 2$  arms and fixed reward probabilities
- Questions:
  - 1/ do models **differ**? **how**?
  - 2/ are all models **influenced by reward**? **why**?
  - 3/ what is **set arbitrarily** to make this figure?
- How is **M1** (random responding) related to **all other models**? How is **M5** related to **M3/M4**?

# Modeling guidelines

- Simulate **TD-based RL** in the bandit task with  $K = 2$  arms and fixed reward probabilities



# Modeling guidelines

- Model **simulations** are useful to:
  - ✓ check that candidate models make **different predictions** in the **same task**
  - ✓ choose **task variables** (e.g., difficulty)
- What controls difficulty in this bandit task?
- Why is it important that all model parameters affect behavioral predictions?
- Why is it important that all candidate models make different behavioral predictions?

# Coming next

- Practical session: today, 2.00pm, same room
- Paper to read:  
Niv (2021) The primacy of behavioral research for understanding the brain. *Behav. Neuro.*  
<https://doi.org/10.1037/bne0000471> (open-access)

- Contact:

**Valentin Wyart**

[valentin.wyart@ens.psl.eu](mailto:valentin.wyart@ens.psl.eu)

**Lucas Benjamin**

[lucas.benjamin78@gmail.com](mailto:lucas.benjamin78@gmail.com)

Lab. de Neurosciences Cognitives et Computationnelles (LNC<sup>2</sup>)  
Institut National de la Santé et de la Recherche Médicale  
Ecole Normale Supérieure, Université PSL