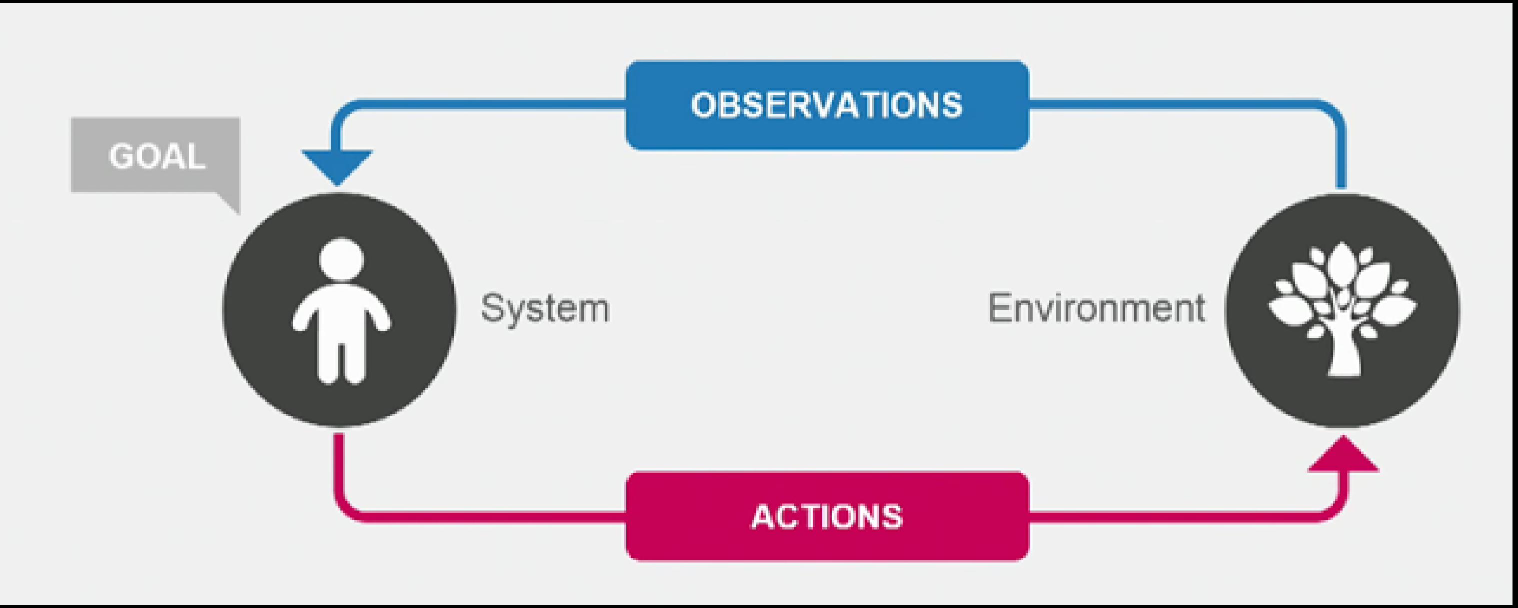


Cracking Bayesian computational modelling with Stan:
Using Rescorla-Wagner model as an example

Lei Zhang

Social, Cognitive and Affective Neuroscience Unit (SCAN-Unit), University of Vienna

lei.zhang@univie.ac.at
lei-zhang.net
@lei_zhang_lz

01.04.2022, Cognition Academy, Max Planck School of Cognition
Materials: https://git.io/Jsr1u

MAX PLANCK SCHOOL of cognition

SCAN Unit

universität wien
Fakultät für Psychologie

10.2019

01.2020

Outline

- Recap on Bayesian modeling
- Why Stan?
- What is (computational) cognitive modeling?
- The idea of the simple Rescorla-Wagner (RW) model
- Implementing RW model for one subject in Stan
- Fitting multiple subjects with the hBayesDM package
- Q&A

Outline

- Recap on Bayesian modeling
- Why Stan?
- What is (computational) cognitive modeling?
- The idea of the simple Rescorla-Wagner (RW) model
- Implementing RW model for one subject in Stan
- Fitting multiple subjects with the hBayesDM package
- Q&A

Recap on Bayesian modeling

Likelihood
How plausible is the data given our parameter is true?

Prior
How plausible is our parameter before observing the data?

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$$

Posterior
How plausible is our parameter given the observed data?

Evidence
How plausible is the data under all possible parameters?

Bernoulli Model

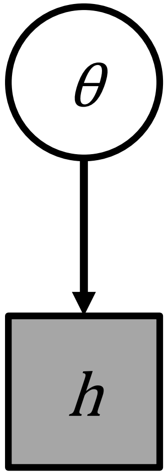
- You are interested in if a coin is biased.
- You will flip the coin.
- You will record whether it comes up a head (h) or a tail (t).
- You might observe 15 heads out of 20 flips.
- What is your degree of belief about the biased parameter ϑ?

Kruschke (2015)

Bernoulli Model

	continuous	discrete
unobserved	θ	δ
observed	y	N

$$p(h \mid \theta) = \theta^{h}(1-\theta)^{1-h}$$



$$\theta \sim \mathrm{Uniform}(0, 1)$$

$$h \sim \mathrm{Bernoulli}\ (\theta)$$

Steps of (Bayesian) Modeling?

A data story

Think about how the data might arise.

It can be *descriptive* or even *causal*.

Update

Educate your model by feeding it with data.

Bayesian Update:

- update the prior, in light of data, to produce posterior
- the updated posterior then becomes the prior of next update

Evaluate

Compare model with reality.

Revise your model.

McElreath (2016)

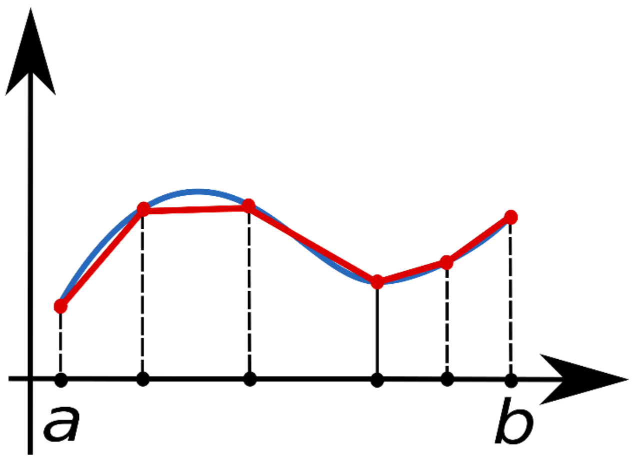
Solve it by Grid Approximation

discrete parameters

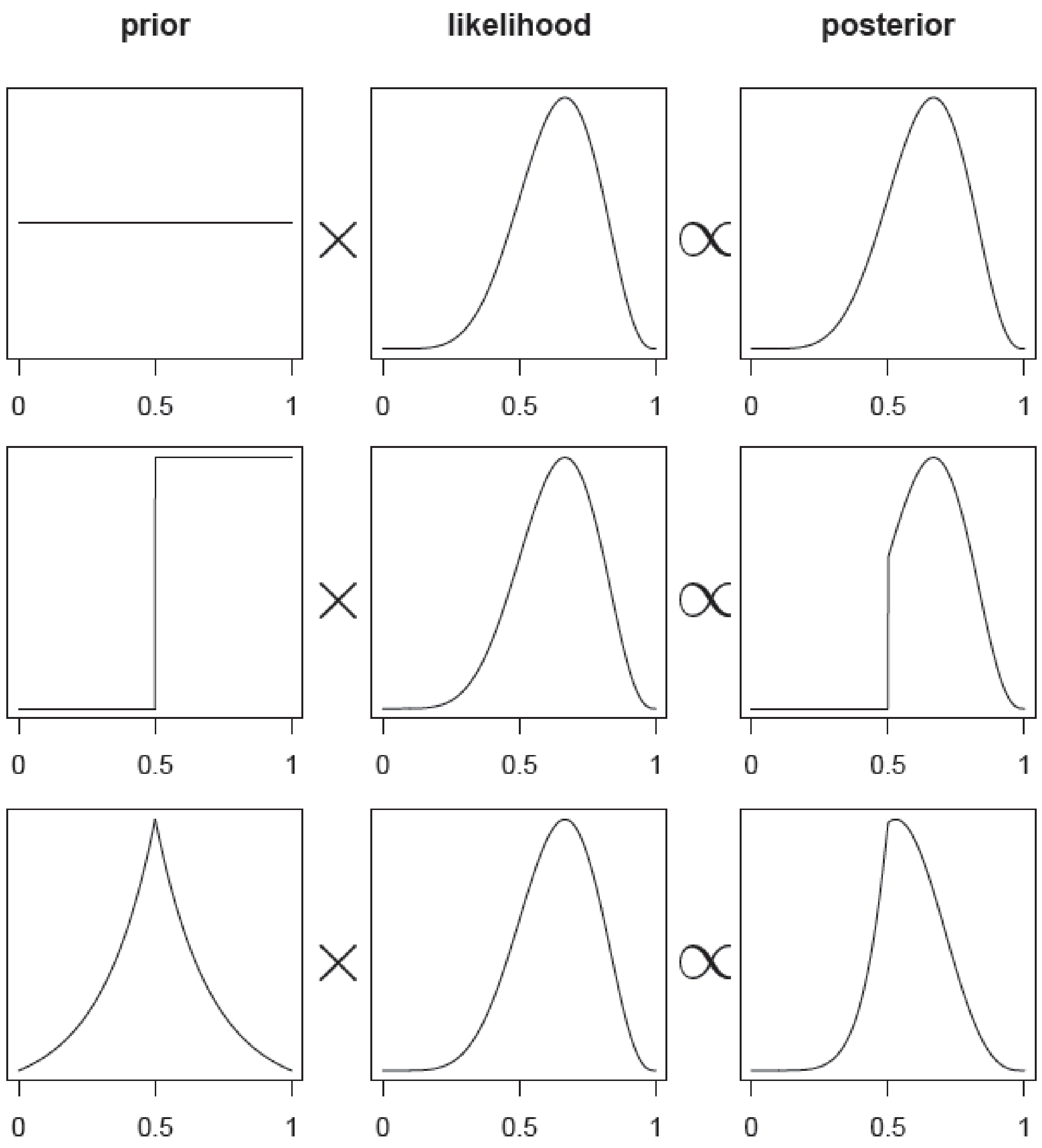
$$p(\theta \mid D) = \frac{p(D \mid \theta)\, p(\theta)}{\boxed{\sum_{\theta^*} p(D \mid \theta^*)\, p(\theta^*)}}$$

continuous parameters

$$p(\theta \mid D) = \frac{p(D \mid \theta)\, p(\theta)}{\boxed{\int p(D \mid \theta^*)\, p(\theta^*)\, d\theta^*}}$$



Bayesian estimation vs. MLE



McElreath (2016)

What if I have multiple parameters?

grid approximation for
2 parameters?
5 parameters?
10 parameters?

$$p(\theta \mid D) = \frac{p(D \mid \theta)\, p(\theta)}{\boxed{\int p(D \mid \theta^*)\, p(\theta^*)\, d\theta^*}}$$

$$p(data) = \int_{\text{All}\,\theta_1} \int_{\text{All}\,\theta_2} p(data, \theta_1, \theta_2)\, \mathrm{d}\theta_1 \mathrm{d}\theta_2$$

$$p(data) = \int_{\mu_1} \int_{\sigma_1} \cdots \int_{\mu_{100}} \int_{\sigma_{100}} \underbrace{p(data \mid \mu_1, \sigma_1, \ldots, \mu_{100}, \sigma_{100})}_{\text{likelihood}} \times \underbrace{p(\mu_1, \sigma_1, \ldots, \mu_{100}, \sigma_{100})}_{\text{prior}}$$
$$\mathrm{d}\mu_1 \mathrm{d}\sigma_1 \ldots \mathrm{d}\mu_{100} \mathrm{d}\sigma_{100},$$

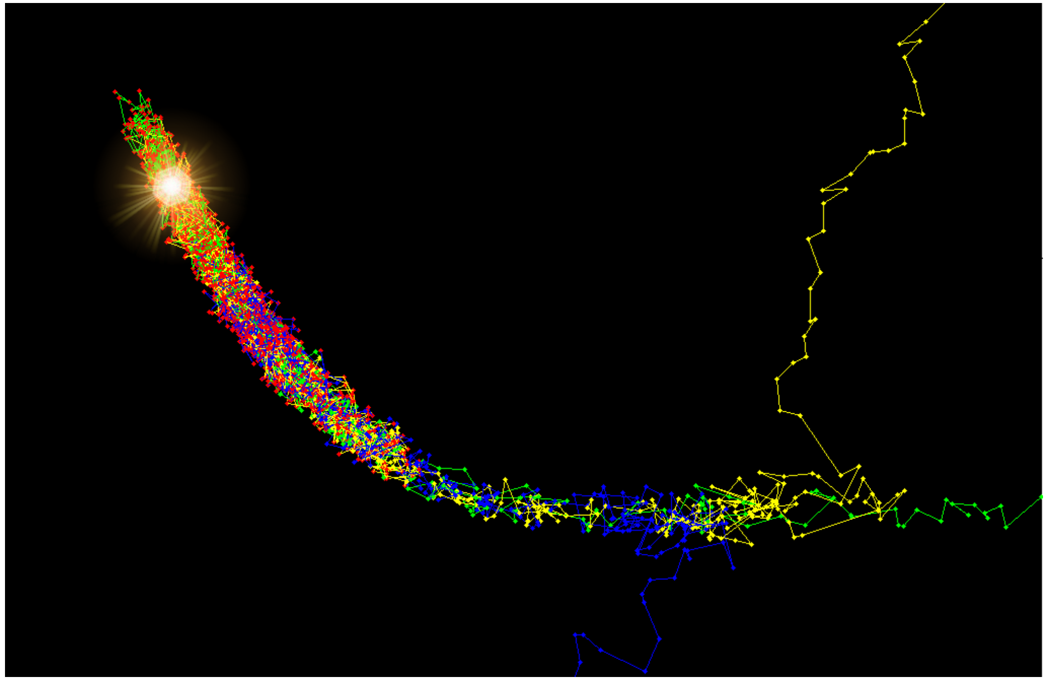
- Analytical solutions (often does not exist)
- Grid approximation (takes too long)

$$\boxed{p(\theta \mid D) \propto p(D \mid \theta)\, p(\theta)}$$

- solution: Markov Chain Monte Carlo

MCMC Sampling Algorithms

- Rejection sampling
- Importance sampling
- Metropolis algorithm
- Gibbs sampling → JAGS
- HMC sampling*



Stan!

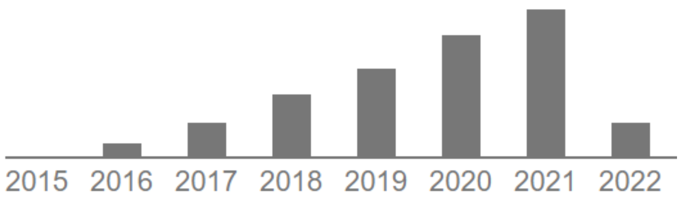
*: also PyMC3 PyMC3

Outline

- Recap on Bayesian modeling
- Why Stan?
- What is (computational) cognitive modeling?
- The idea of the simple Rescorla-Wagner (RW) model
- Implementing RW model for one subject in Stan
- Fitting multiple subjects with the hBayesDM package
- Q&A

Is Stan popular?

Cited by 5111



Stan: A Probabilistic Programming Language

Carpenter, Bob; Gelman, Andrew; Hoffman, Matthew D.; Lee, Daniel; Goodrich, Ben; Betancourt, Michael; Brubaker, Marcus A.; Guo, Jiqiang; Li, Peter; Riddell, Allen

Grantee Submission, Journal of Statistical Software v76 n1 p1-32 Jan 2017

Stan is a probabilistic programming language for specifying statistical models. A Stan program imperatively defines a log probability function over parameters conditioned on specified data and constants. As of version 2.14.0, Stan provides full Bayesian inference for continuous-variable models through Markov chain Monte Carlo methods such as the No-U-Turn sampler, an adaptive form of Hamiltonian Monte Carlo sampling. Penalized maximum likelihood estimates are calculated using optimization methods such as the limited memory Broyden-Fletcher-Goldfarb-Shanno algorithm. Stan is also a platform for computing log densities and their gradients and Hessians, which can be used in alternative algorithms such as variational Bayes, expectation propagation, and marginal inference using approximate integration. To this end, Stan is set up so that the densities, gradients, and Hessians, along with intermediate quantities of the algorithm such as acceptance probabilities, are easily accessible. Stan can be called from the command line using the "cmdstan" package, through R using the "rstan" package, and through Python using the "pystan" package. All three interfaces support sampling and optimization-based inference with diagnostics and posterior analysis. "rstan" and "pystan" also provide access to log probabilities, gradients, Hessians, parameter transforms, and specialized plotting.

Descriptors: Programming Languages, Probability, Bayesian Statistics, Monte Carlo Methods, Statistical Inference, Maximum Likelihood Statistics, Computation, Statistical Distributions, Computer Software

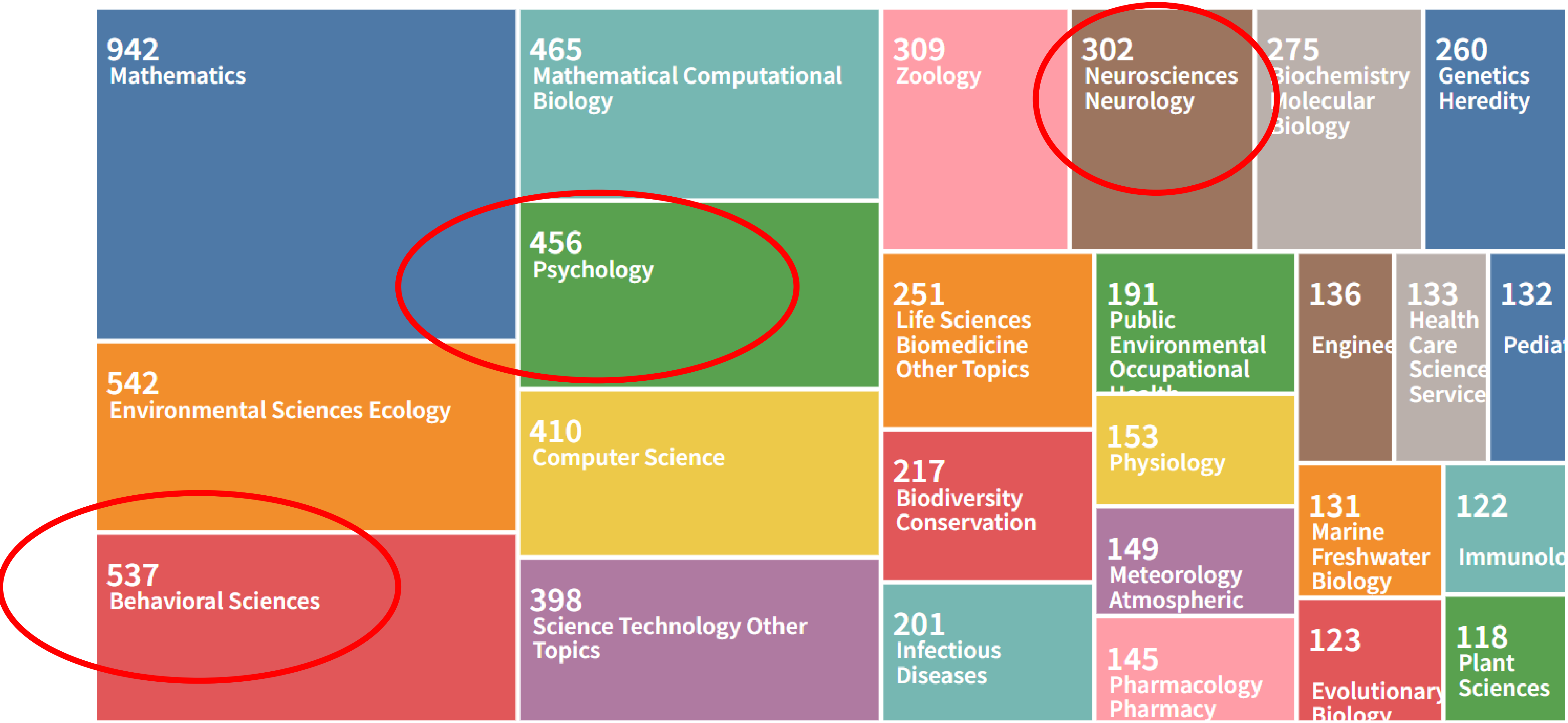
[PDF] **JAGS**: A program for analysis of Bayesian graphical models using Gibbs sampling

M Plummer - Proceedings of the 3rd international workshop on …, 2003 - ci.tuwien.ac.at

… **JAGS** is a program for Bayesian Graphical modelling which aims for compatibility … **JAGS** and so avoid having to write a new program for each application. A second motivation for **JAGS** …

Save Cite Cited by 5332 Related articles All 7 versions

Who are using Stan?



retrieved in 03/2022, Web of Science

Why using Stan?

vs. BUGS / JAGS

- Less spatial correlation → effective samples
- Time to converge and per effective sample size:

  1 - ∞ times faster
- Memory usage: 1–10%
- Language features
  - variable overwrite: `a = 4, then a = 5`
  - formal control flow (same as R)
  - full support of vectorizing & matrix calculation

http://mc-stan.org/
https://discourse.mc-stan.org/t/selling-stan/3693/6

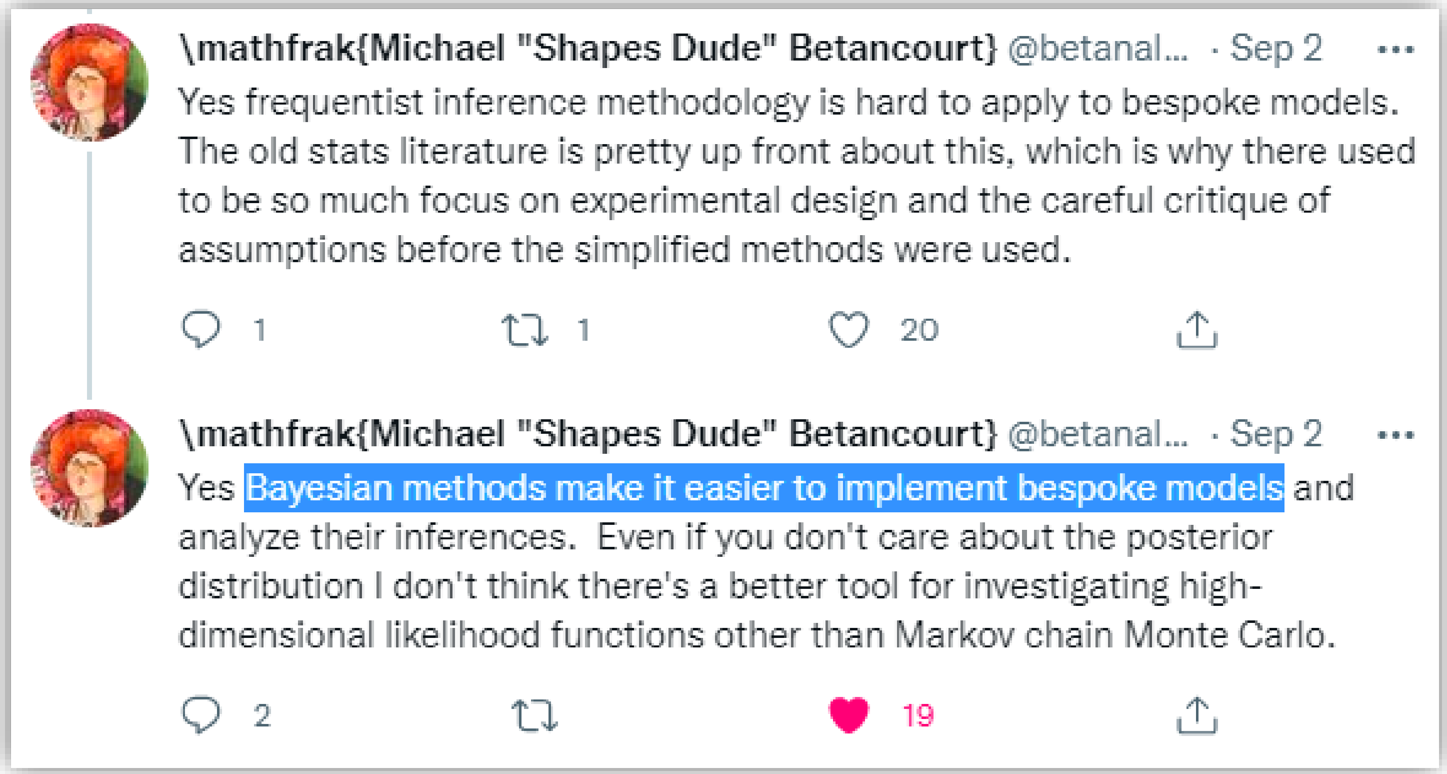
Why using Stan?

Krzysztof Sakrejda
@sakrejda

I keep getting asked why people should use @mcmc_stan so I wrote an answer:

"Selling" Stan
discourse.mc-stan.org

27.03.18, 16:01



\mathfrak{Michael "Shapes Dude" Betancourt} @betanal... · Sep 2
Yes frequentist inference methodology is hard to apply to bespoke models. The old stats literature is pretty up front about this, which is why there used to be so much focus on experimental design and the careful critique of assumptions before the simplified methods were used.

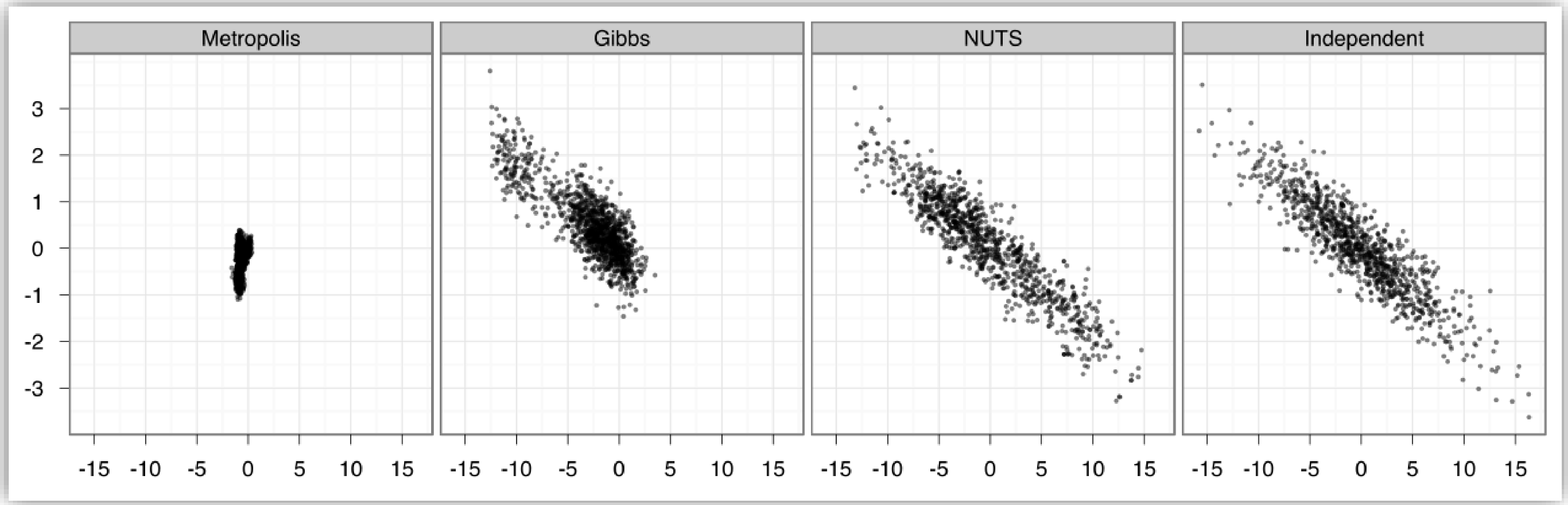
1 1 20

\mathfrak{Michael "Shapes Dude" Betancourt} @betanal... · Sep 2
Yes Bayesian methods make it easier to implement bespoke models and analyze their inferences. Even if you don't care about the posterior distribution I don't think there's a better tool for investigating high-dimensional likelihood functions other than Markov chain Monte Carlo.

2 19

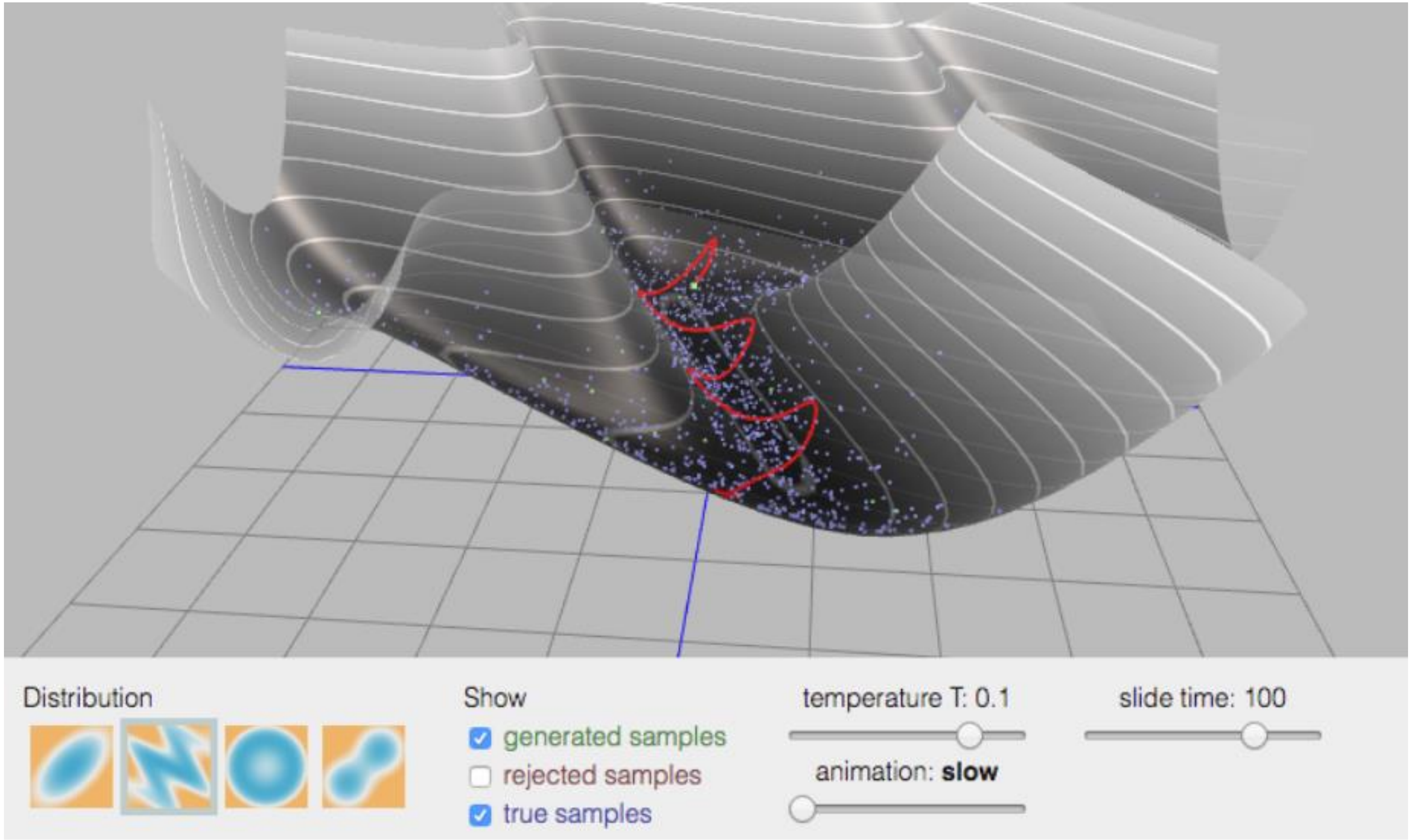
HMC vs. Gibbs and Metropolis

Hamilton MC (HMC) implements No-U-Turn Sampler (NUTS)



- Two dimensions of highly correlated 250-dim normal
- 1,000,000 draws from Metropolis and Gibbs (thin to 1000)
- 1,000 draws from NUTS; 1000 independent draws

build some intuition

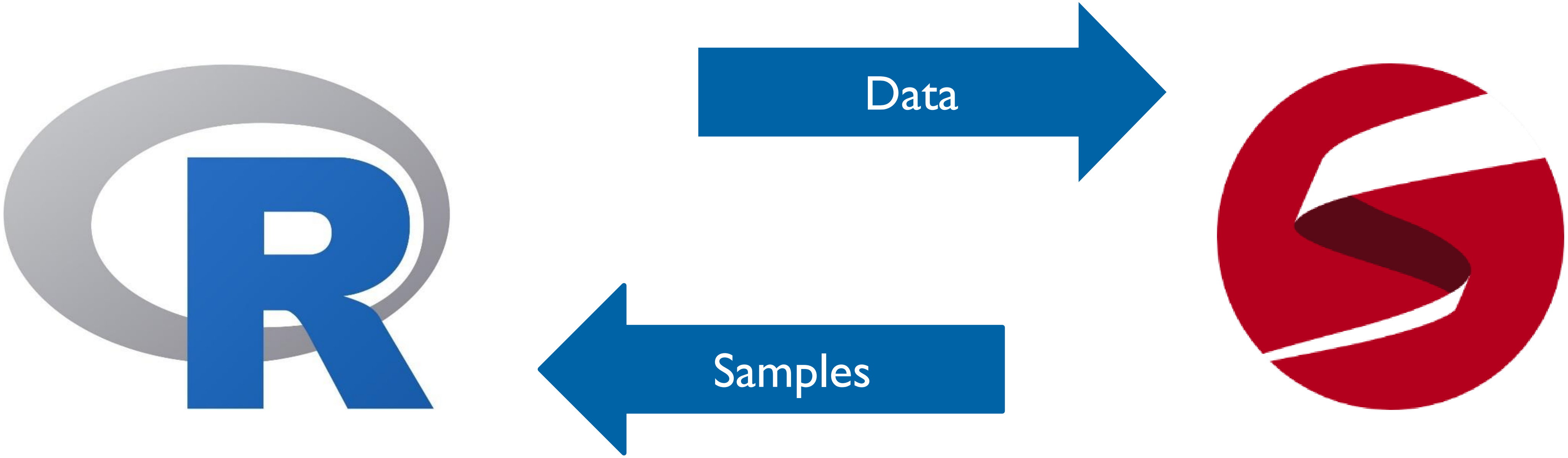


https://arogozhnikov.github.io/2016/12/19/markov_chain_monte_carlo.html

But, any cons?

- HMC provides huge improvements in computational efficiency, but mathematical foundations are more difficult to follow, at least sometimes.
- Stan cannot sample from the posterior distribution of discrete parameters (e.g., [1, 2]).
  - → with additional effort, it can be achieved through marginalisation, see the Stan User Manual.
- There are a few unwritten tips and tricks.
  - Practise makes perfect and follow discussions on the forum.
- Block-based language → might seem rigid in the first place.

Stan and RStan



`proj_main.R`

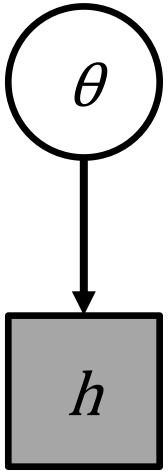
`M1.stan`
`M2.stan`
`M3.stan`

Steps of Bayesian Modeling, with Stan

A data story	Think about how the data might arise. It can be *descriptive* or even *causal*. Write a Stan program (*.stan).
Update	Educate your model by feeding it the data. Bayesian Update: update the prior, in light of data, to produce posterior. Run Stan using RStan (PyStan, MatlabStan etc.)
Evaluate	Compare model with reality. Revise your model. Evaluate in RStan and ShinyStan.

McElreath (2016)

A quick look at Stan



$$\theta \sim \text{Uniform}(0, 1)$$

$$h \sim \text{Bernoulli}\ (\theta)$$

```
data {
  int<lower=0> N;
  int<lower=0,upper=1> flip[N];
}

parameters {
  real<lower=0,upper=1> theta;
}

model {
  flip ~ bernoulli(theta);
}
```

Stan Language

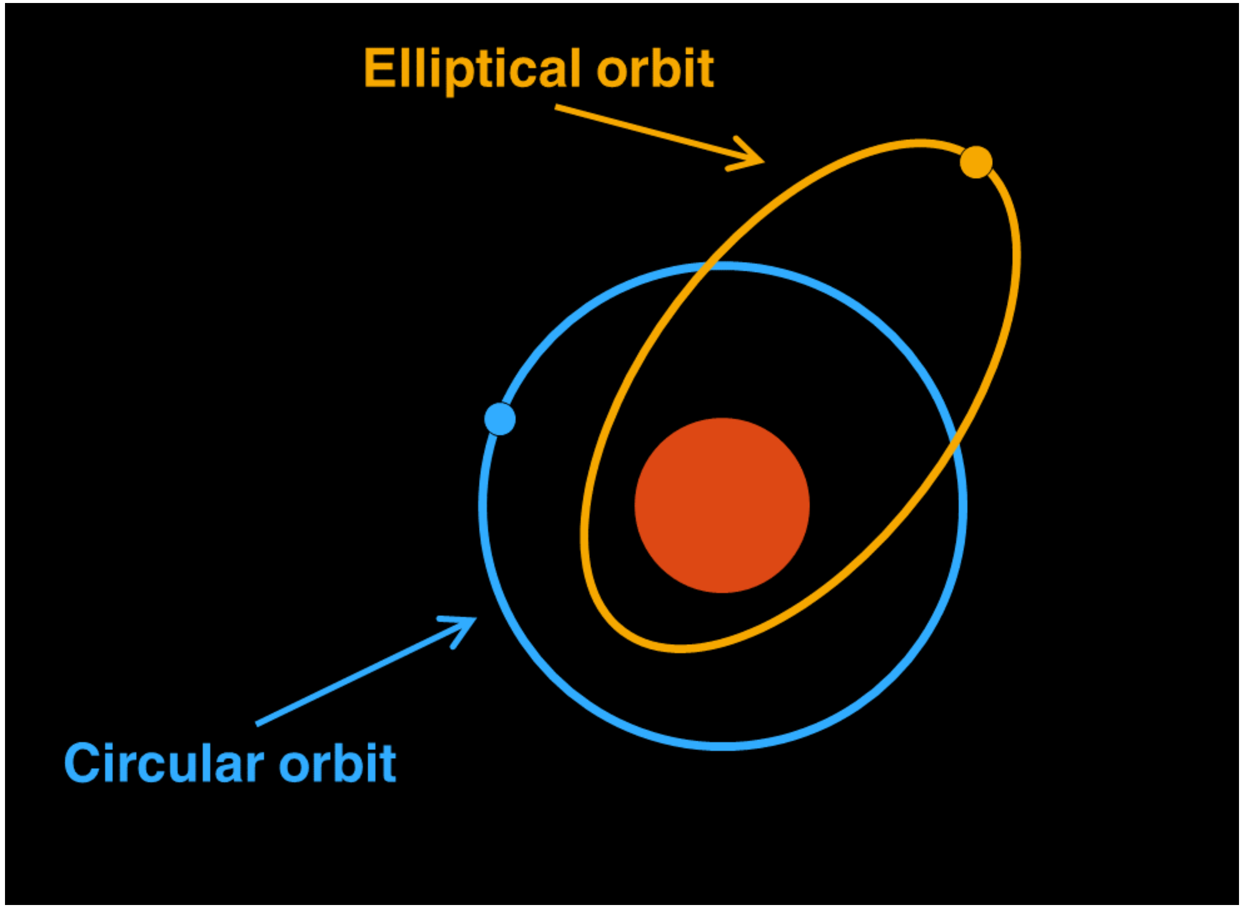
```
data {
 //... read in external data...
}

transformed data {
 //... pre-processing of data ...
}

parameters {
 //... parameters to be sampled by HMC ...
}

transformed parameters {
 //... pre-processing of parameters ...
}

model {
 //... statistical/cognitive model ...
}

generated quantities {
 //... post-processing of the model ...
}
```

Outline

- Recap on Bayesian modeling
- Why Stan?
- What is (computational) cognitive modeling?
- The idea of the simple Rescorla-Wagner (RW) model
- Implementing RW model for one subject in Stan
- Fitting multiple subjects with the hBayesDM package
- Q&A

The idea of computational modeling is never new

Scientists use mathematical models to approximate certain processes (physical or mental), in order to explain and to predict.



Computational modeling

Cognition as information processing

Behavior

Output

Input

$$B = f(P, E)$$

$$y = f(x)$$

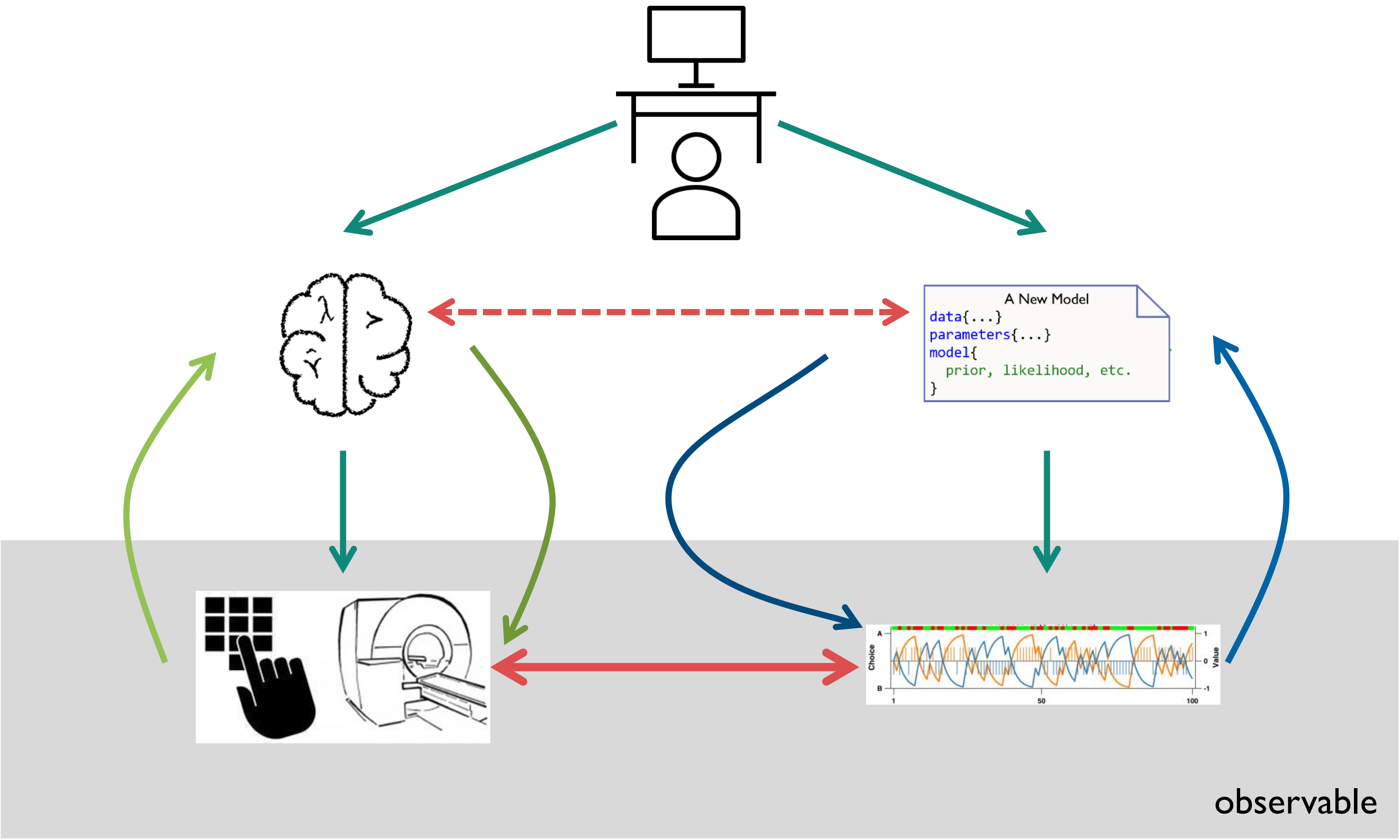
Person

Environment

'Some math happens'

Kurt Lewin, (1936)

Computational modeling of Cognition



Lewandowsky & Farrell (2011); Stephan et al. (2016); Ahn et al., (2017)

Outline

- Recap on Bayesian modeling
- Why Stan?
- What is (computational) cognitive modeling?
- The idea of the simple Rescorla-Wagner (RW) model
- Implementing RW model for one subject in Stan
- Fitting multiple subjects with the hBayesDM package
- Q&A

Simple reinforcement learning: 2-armed bandit task

a simple task often used in the laboratory:

- repeated choice between N options (N-armed bandit)
- …whose properties (reward amounts, probabilities) are learned through trial-and-error
- …with a goal in mind: maximize the overall reward

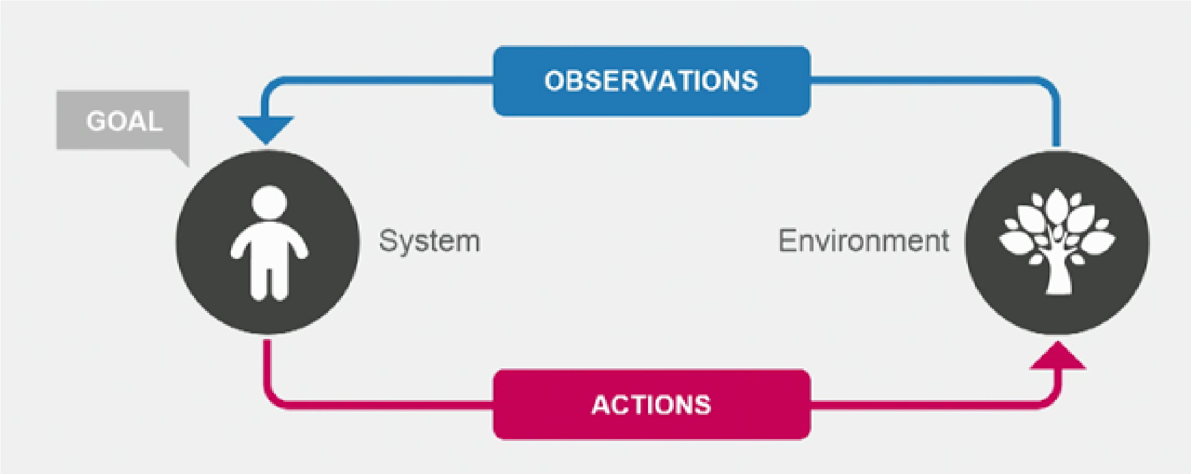
2-armed bandit task

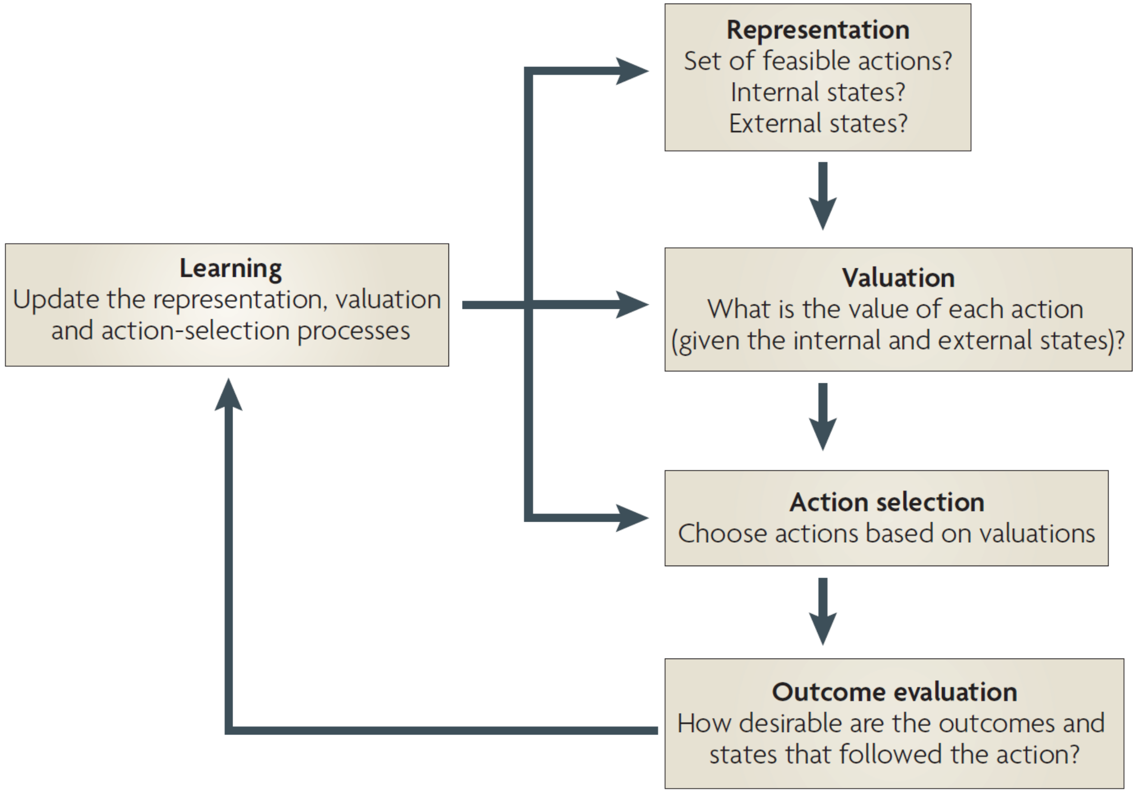
What can be your strategies:

1. predict the value of each deck
2. choose the best
3. learn from outcome to update predictions

(repeat)

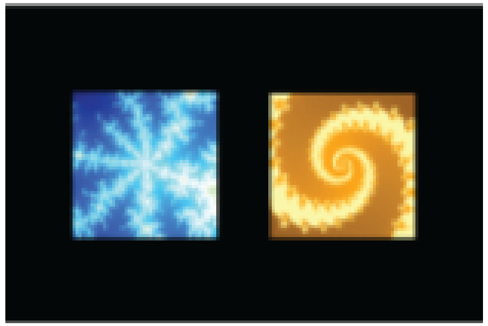
How prediction is shaped by learning?



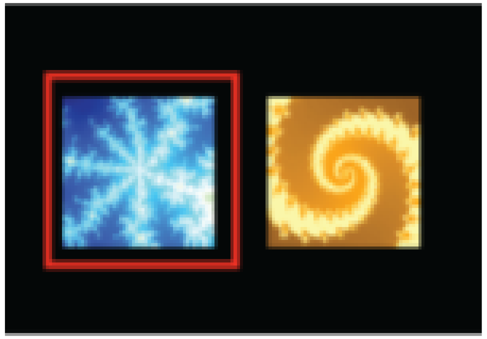


Sutton & Barto (1998); Rangel et al (2008); [Daw & Niv workshop 2015] reward contingency 80:20

One simple experiment: two choice task



choice presentation



action selection

outcome

what do we know?

choice & outcome

what can we measure?

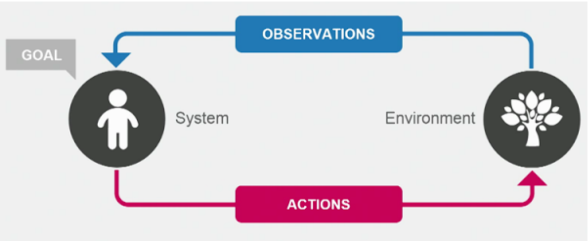
choice accuracy

what do we not know?

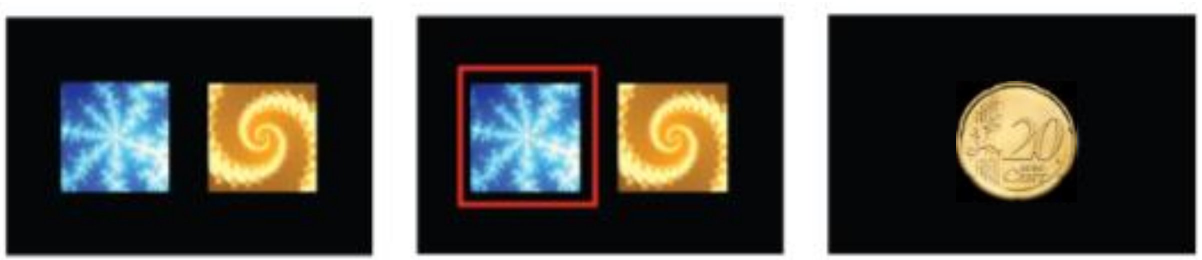
RL update

	subjID	trialID	choice	outcome
1	1	1	1	1
2	1	2	1	1
3	1	3	1	1

p(choosing the better option)



Rescorla-Wagner Value Update



Cognitive Model
- cognitive process
- using internal variables and free parameters

Observation Model (Data Model)
- relate model to observed data
- has to account for noise

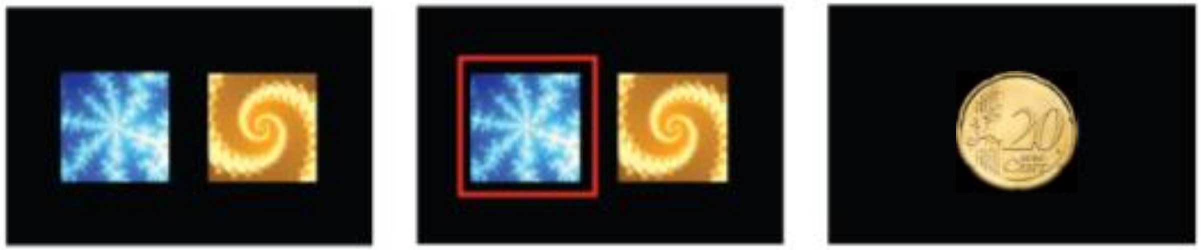
Rescorla & Wagner (1972)

Rescorla-Wagner (1972)

- The idea: error-driven learning
- Change in value is proportional to the difference between actual and predicted outcome

Robert A. Rescorla

Allan R. Wagner



Value update: $V_t = V_{t-1} + \alpha * \mathrm{PE}_{t-1}$

Prediction error: $\mathrm{PE}_{t-1} = R_{t-1} - V_{t-1}$

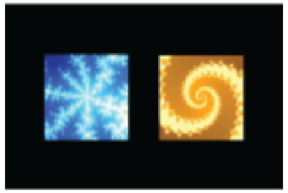
*Expectations on the next trial = the expectation on the current trial + learning rate * prediction error (reward – current expectation)*

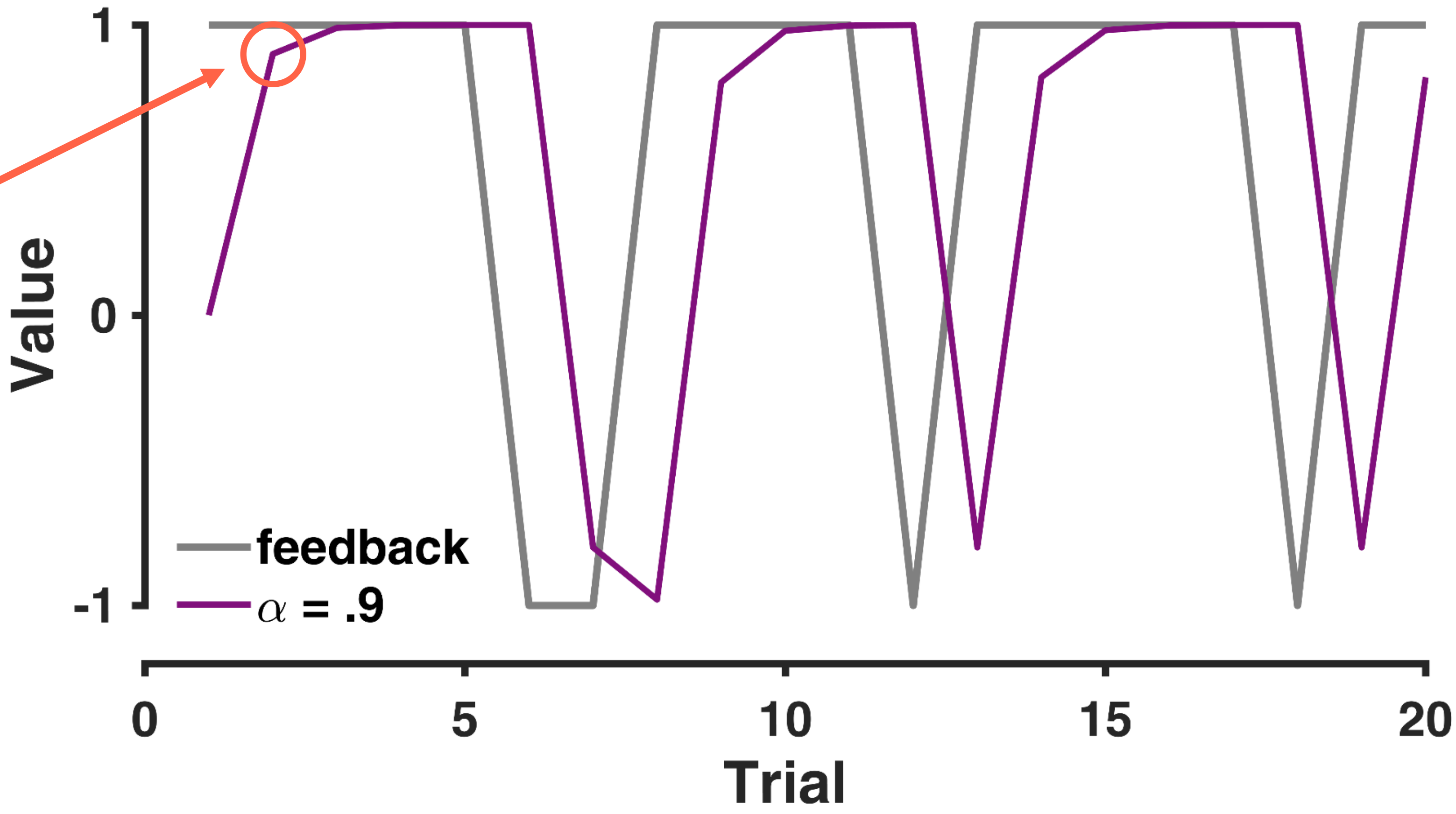
Rescorla & Wagner (1972)

Understand the learning rate

Value update: $V_t = V_{t-1} + \alpha * \mathrm{PE}_{t-1}$

Prediction error: $\mathrm{PE}_{t-1} = R_{t-1} - V_{t-1}$





if $\alpha = 0.9$

$V_1 = 0$

$V_2 = V_1 + 0.9 * (1 - V_1)$

$= 0 + 0.9 * (1 - 0)$

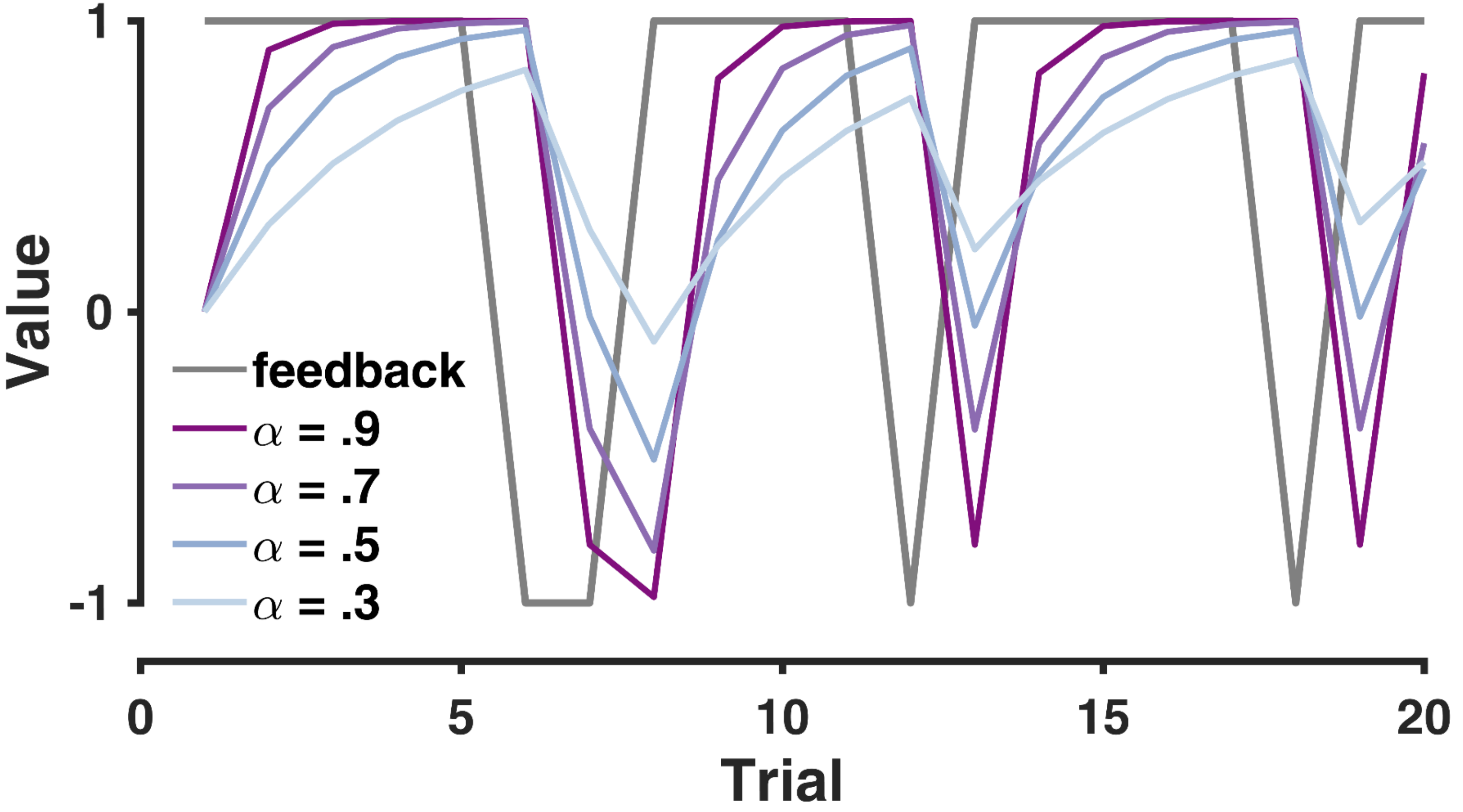
$= 0.9$

reward contingency – 80:20

Understand the learning rate

Value update: $V_t = V_{t-1} + \alpha * \mathrm{PE}_{t-1}$

Prediction error: $\mathrm{PE}_{t-1} = R_{t-1} - V_{t-1}$



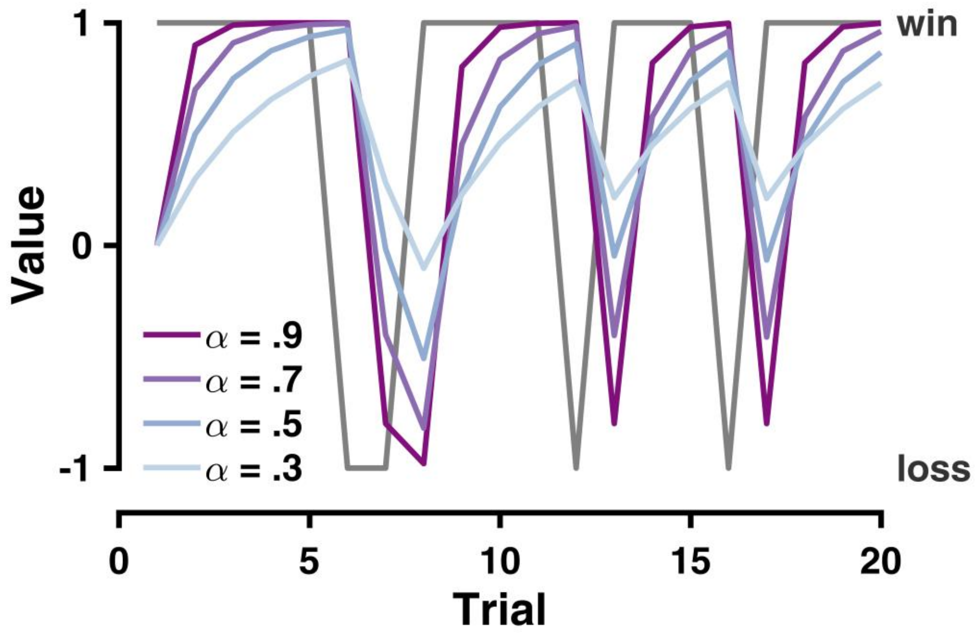
reward contingency – 80:20

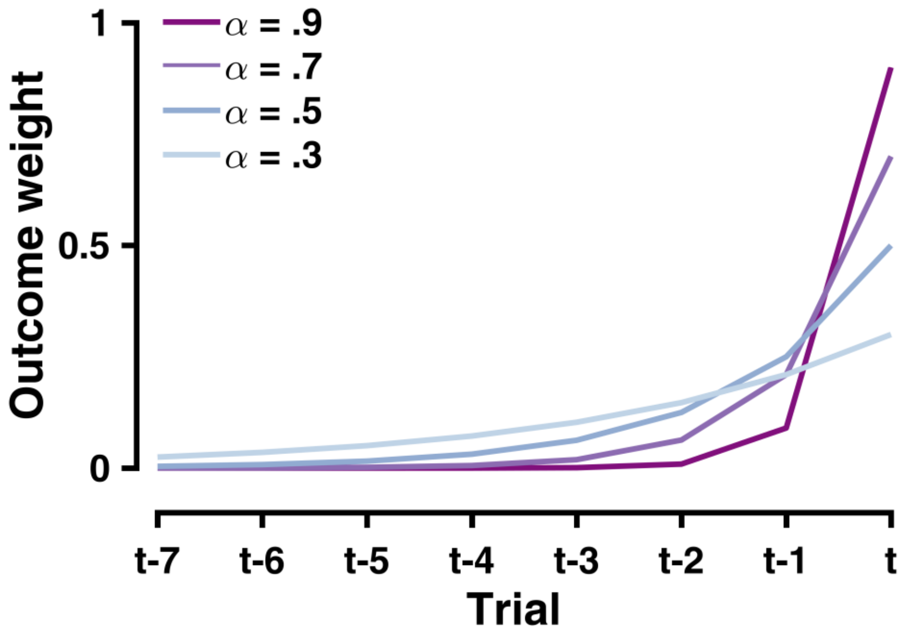
Understand the learning rate

Value update: $V_t = V_{t-1} + \alpha * \mathrm{PE}_{t-1}$

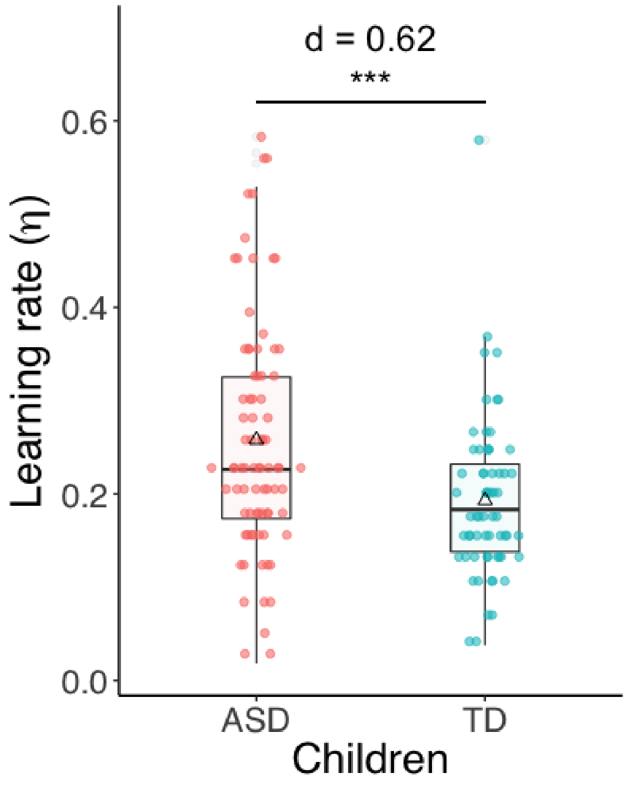
Prediction error: $\mathrm{PE}_{t-1} = R_{t-1} - V_{t-1}$

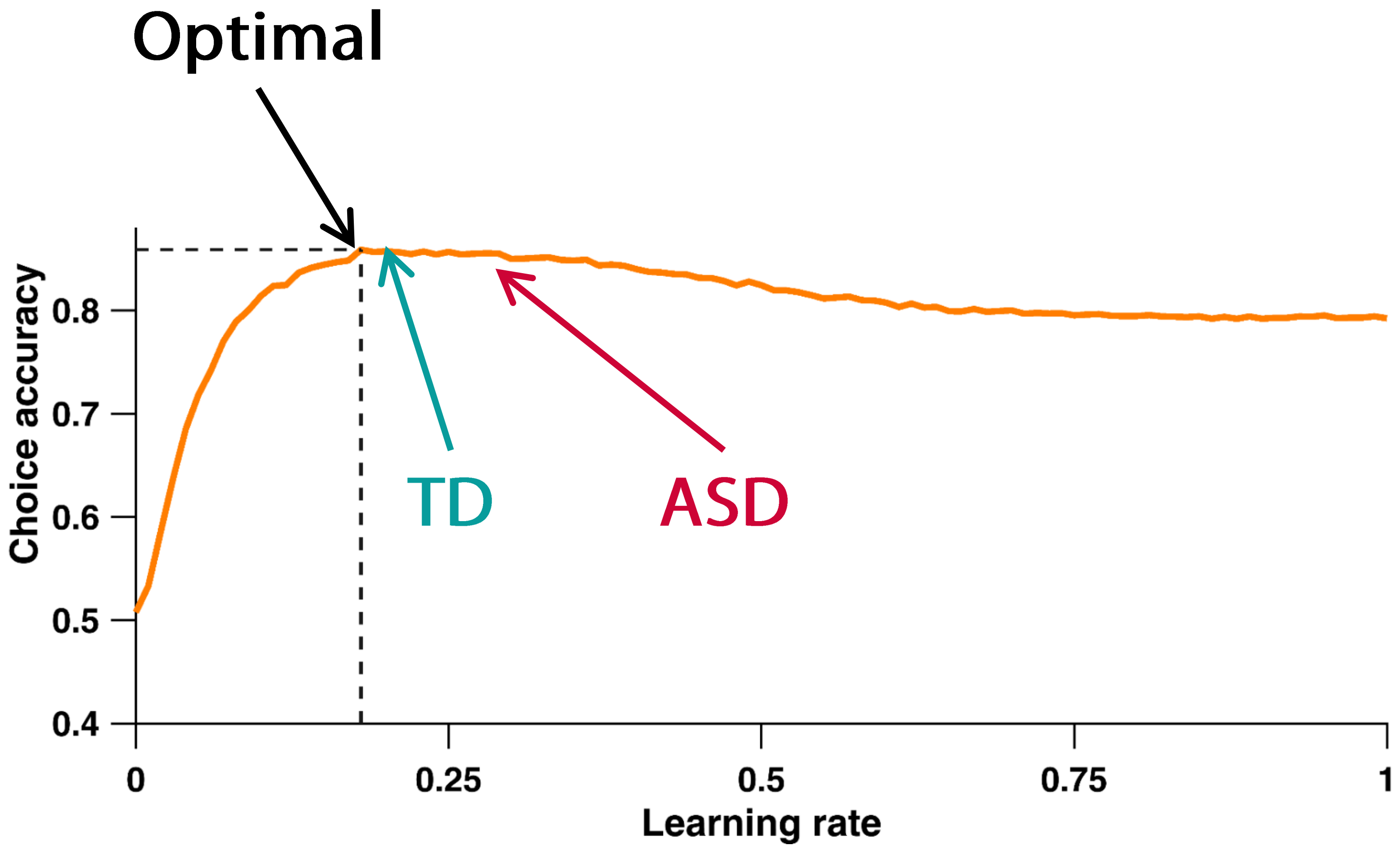
$$
\begin{aligned}
V_t &= (1-\alpha)\,V_{t-1} + \alpha R_{t-1} \\
&= (1-\alpha)\,(V_{t-2} + \alpha\,(R_{t-2} - V_{t-2})) + \alpha R_{t-1} \\
&= (1-\alpha)^t V_0 + \boxed{\sum_{i=1}^{t} (1-\alpha)^{t-i} \alpha R_i}
\end{aligned}
$$





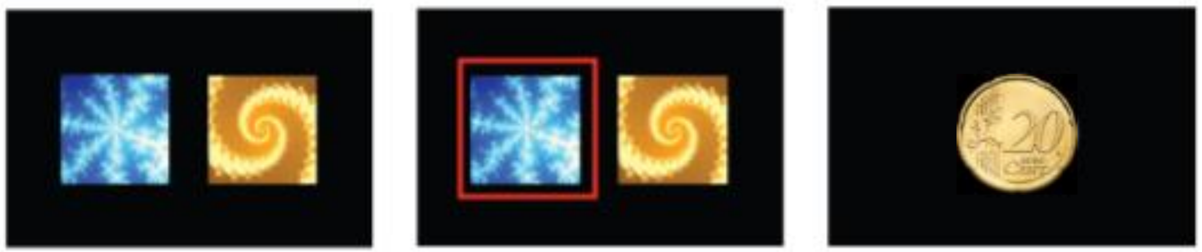
Optimal learning rate?





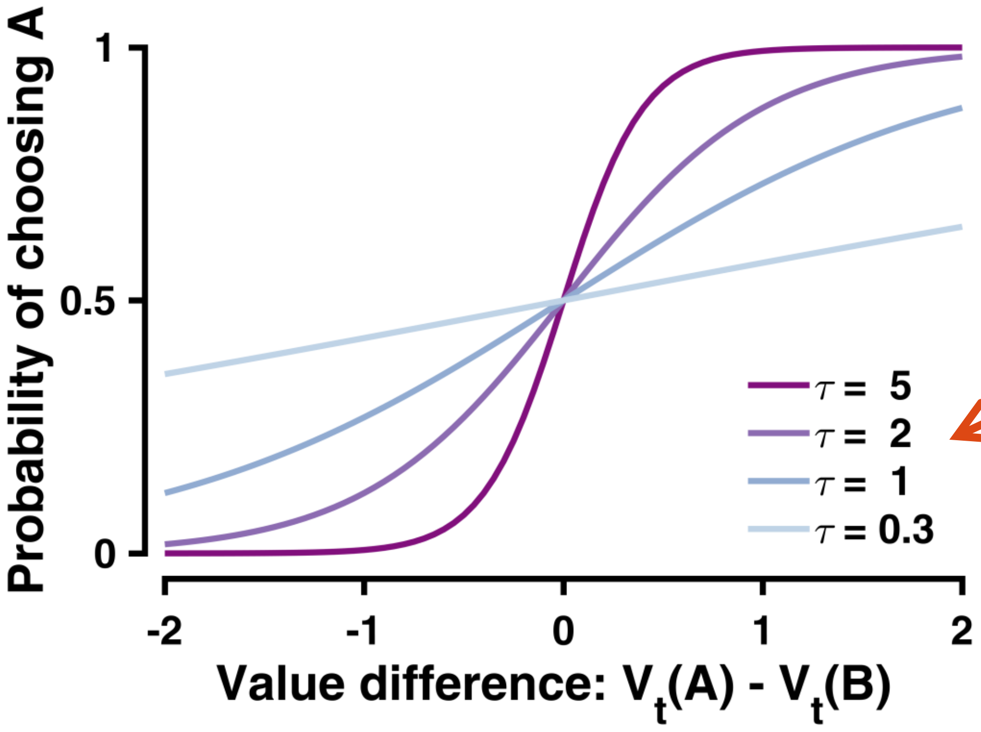
Crawley* & Zhang* et al., (2020)

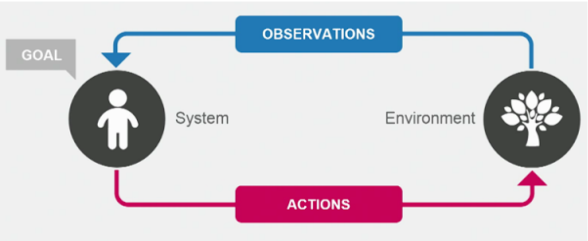
Choice rule: softmax



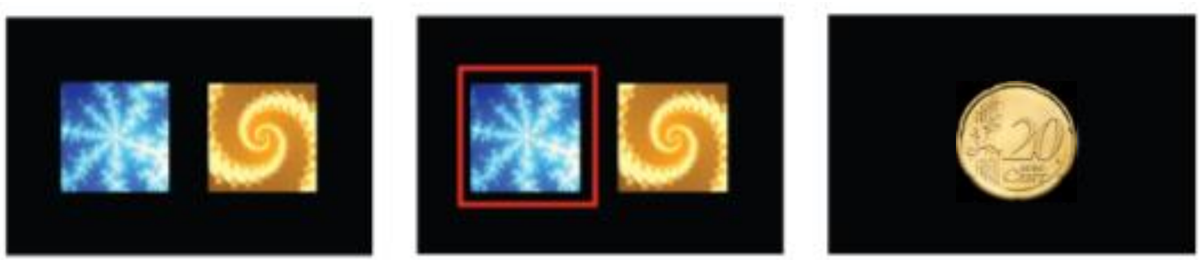
$$p_t(A) = \boxed{\frac{e^{\tau * V_t(A)}}{e^{\tau * V_t(A)} + e^{\tau * V_t(B)}}}$$

$$= \frac{1}{1 + e^{-\tau * (V_t(A) - V_t(B))}}$$





Rescorla-Wagner Value Update



Value update:

$$V_{t+1} = V_t + \alpha * PE_t$$

Prediction error:

$$PE_t = R_t - V_t$$

choice rule (sigmoid /softmax):

$$p(C{=}a) = \frac{1}{1+e^{\tau*(v(b)-v(a))}}$$

α - learning rate
PE - reward prediction error
V - value
R - reward
τ - softmax temperature

Rescorla & Wagner (1972)

Some terminologies

simple Rescorla-Wagner

$$V_{t+1} = V_t + \alpha\,(R_t - V_t)$$

$$p_t(\mathrm{A}) \;=\; \frac{1}{1+e^{-\tau*(\mathrm{v}(A)-\mathrm{v}(B))}}$$

$$\mathrm{LL} = \mathrm{categorical\_lpmf}(\, \mathrm{C}_t \mid p_t)$$

simple linear regression

$$\mu = \mathrm{a} + \mathrm{b}\,\mathrm{X}$$

$$\mathrm{LL} = \mathrm{normal\_lpdf}(\, \mathrm{Y} \mid \mu, \delta\,)$$

Some terminologies

simple Rescorla-Wagner

$$V_{t+1} = V_t + \alpha (R_t - V_t)$$

$$p_t(A) = \frac{1}{1+e^{-\tau*(v(A)-v(B))}}$$

$$LL = \text{categorical\_lpmf}(\ C_t \mid p_t)$$

simple linear regression

$$\mu = a + b\ X$$

$$LL = \text{normal\_lpdf}(\ Y \mid \mu, \delta\ )$$

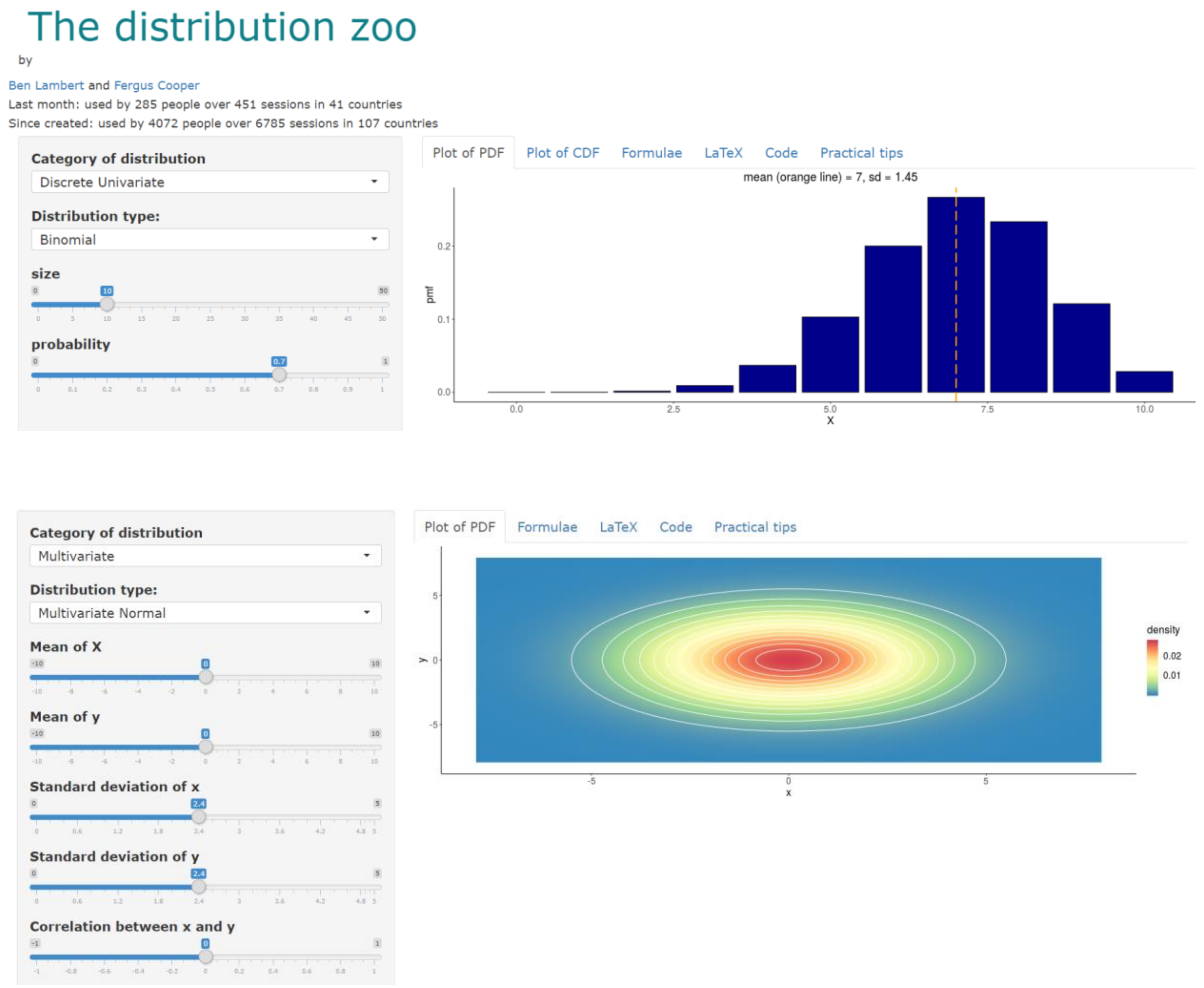
data / observation

free parameter / model parameter

internal/latent (model) variable

likelihood function

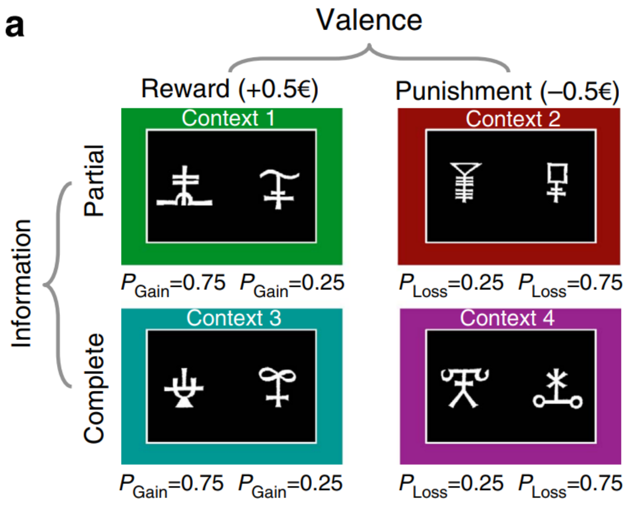
How do I know which likelihood to use?



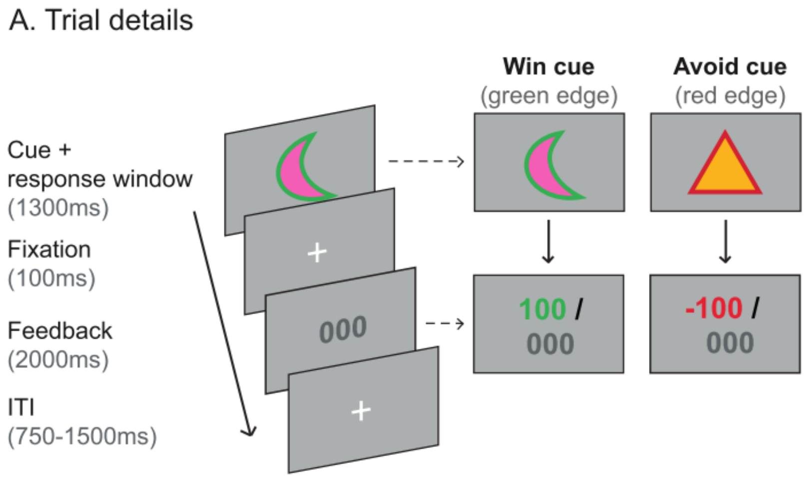
Lambert (2018)

https://ben18785.shinyapps.io/distribution-zoo/

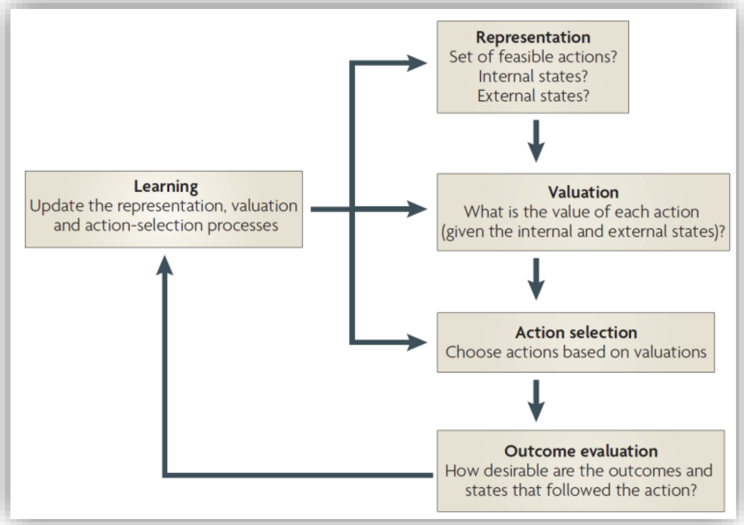
Generalizing RL framework

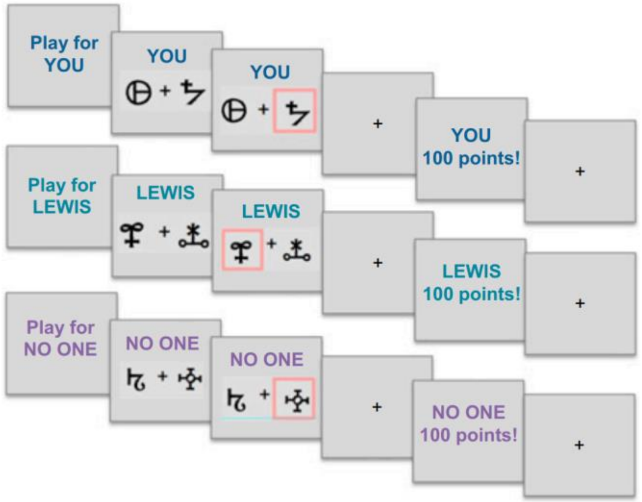


Palminteri et al. (2015)

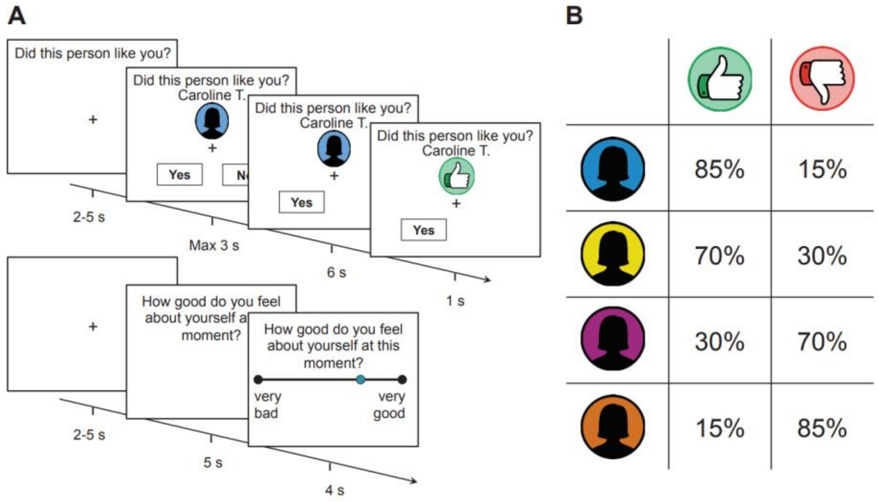


Swart et al. (2017)





Lockwood et al. (2016)



Will et al. (2017)

Some more food for thought

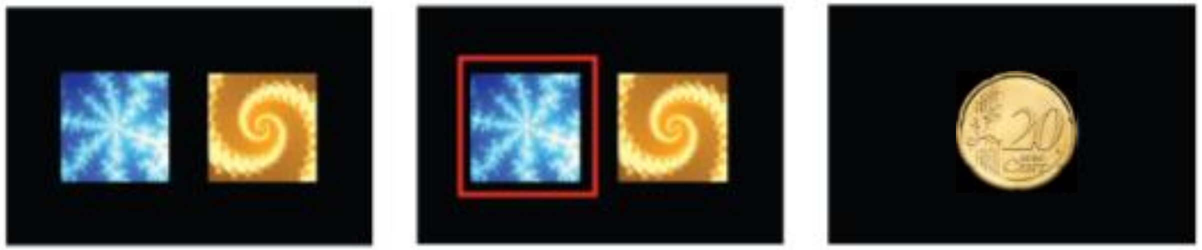
What is a social RL task?	There are commonly two variations: either reward learning in social contexts (e.g. learn to expect monetary reward for a social partner) or social feedback learning (e.g. learn to expect social status or social evaluation). When the goal is to compare different types of feedback (e.g. social vs. non-social feedback), we suggest matching the feedback as closely as possible on 'domain general' properties, such as salience or preference	Lockwood *et al.*, 2016; Will *et al.*, 2017
Is the learning rate static across trials, or dynamically adapting along the course of the experiment?	The learning rate does not necessarily have to be constant. But in the case of the Rescorla–Wagner model (and related models), the learning rate is indeed static. A dynamic learning rate, however, is possible when other types of models are applied. Note that the interpretation of the learning rate we discussed in the main text is independent of this constant *vs* dynamic property	Li *et al.*, 2011; Mathys *et al.*, 2011
Is it possible to use RL models in the absence of choice data?	At least some sort of data is needed to perform model estimation. For example, skin conductance response (SCR) or pupil size response (PSR) have been used to fit RL models in associative fear learning tasks, where choice data was not available	Li *et al.*, 2011; Tzovara *et al.*, 2018

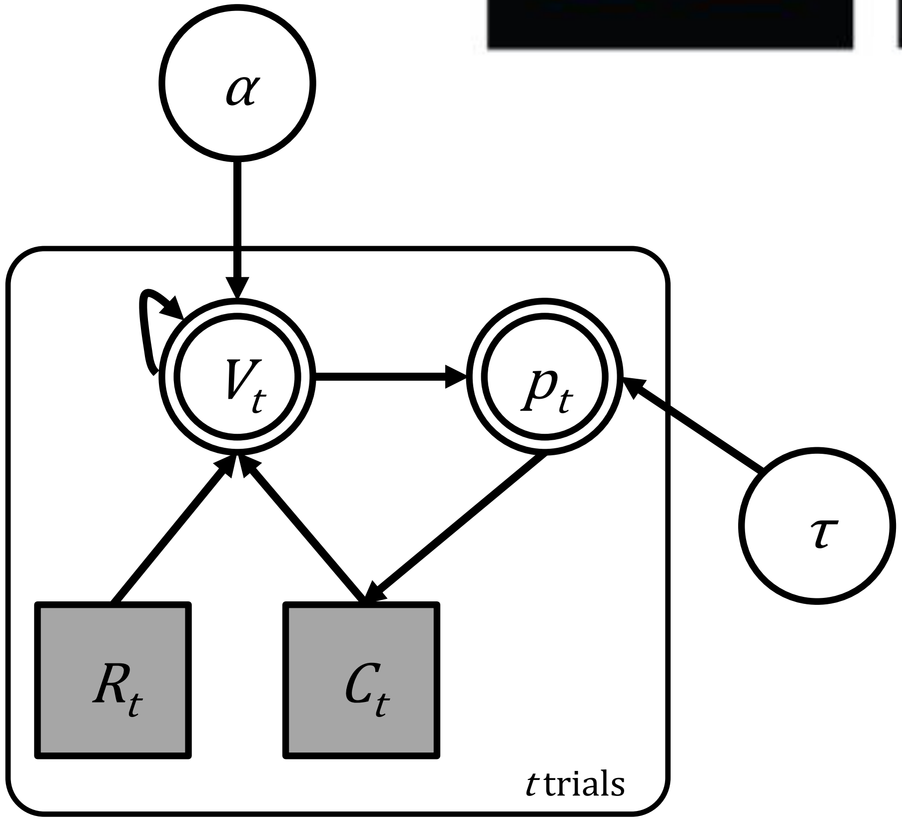
Zhang et al., (2020)

Outline

- Recap on Bayesian modeling
- Why Stan?
- What is (computational) cognitive modeling?
- The idea of the simple Rescorla-Wagner (RW) model
- Implementing RW model for one subject in Stan
- Fitting multiple subjects with the hBayesDM package
- Q&A

RL – Implementation





$$\alpha \sim Uniform\,(0,1)$$

$$\tau \sim Uniform\,(0,3)$$

$$p_t(C = A) = \frac{1}{1 + e^{\tau\,(V_t(B) - V_t(A))}}$$

$$V_{t+1}^{c} = V_t^{C} + \alpha\,(R_t - V_t^{C})$$

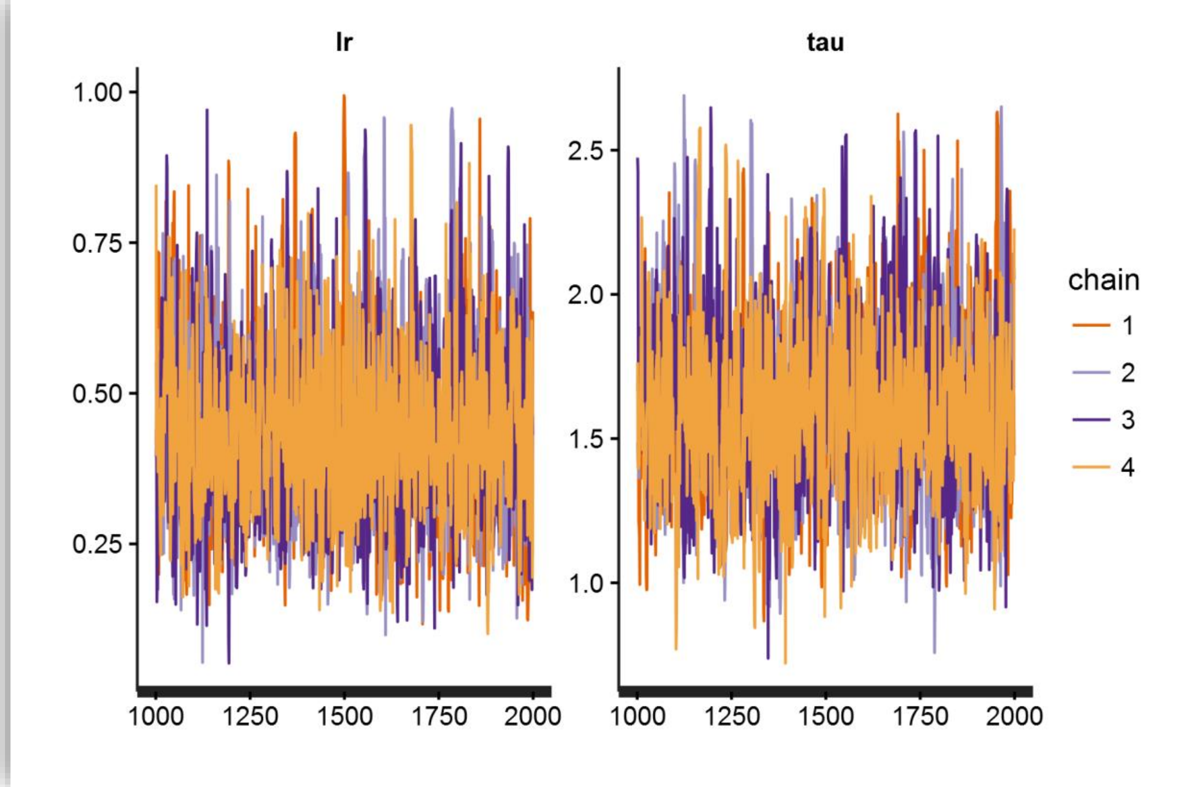
deterministic variable

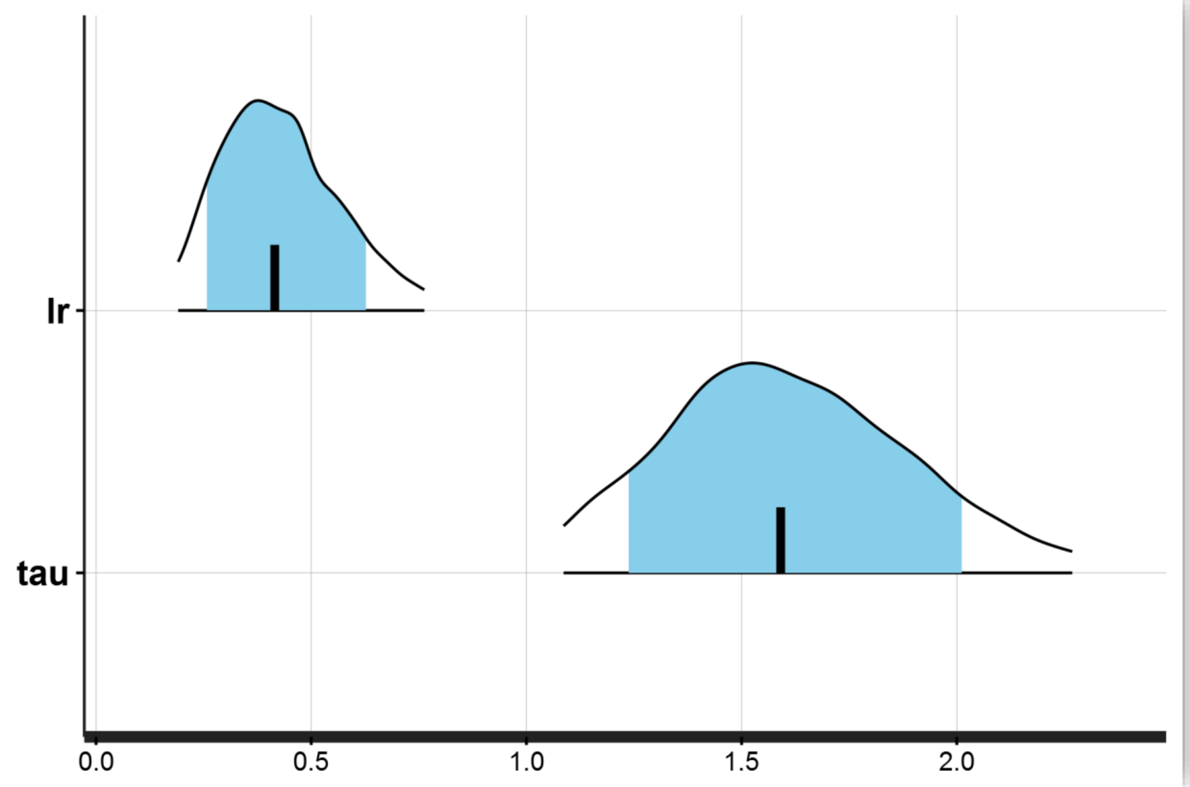
Live demo of building the RW model in Stan from scratch

RW model for 1 subj in Stan

```
data { //define our data
    int<lower=1> nTrials;
    int<lower=1, upper=2> choice[nTrials];
    int<lower=-1,upper=1> reward[nTrials];
}

parameters { // define our free parameters
    real<lower=0,upper=1> alpha;
    real<lower=0,upper=20> tau;
}

model { // the model/likelihood function
    vector[2] v; // v[1] is blue, v[2] is yellow
    real pe;
    vector[2] p;

    v = rep_vector(0, 2);

    for (t in 1:nTrials) {
        // softmax action prob
        p = softmax(tau * v);

        // choice likelihood
        choice[t] ~ categorical(p);

        // prediction error
        pe = reward[t] - v[choice[t]];

        // value update
        v[choice[t]] = v[choice[t]] + alpha * pe;
    }
}
```

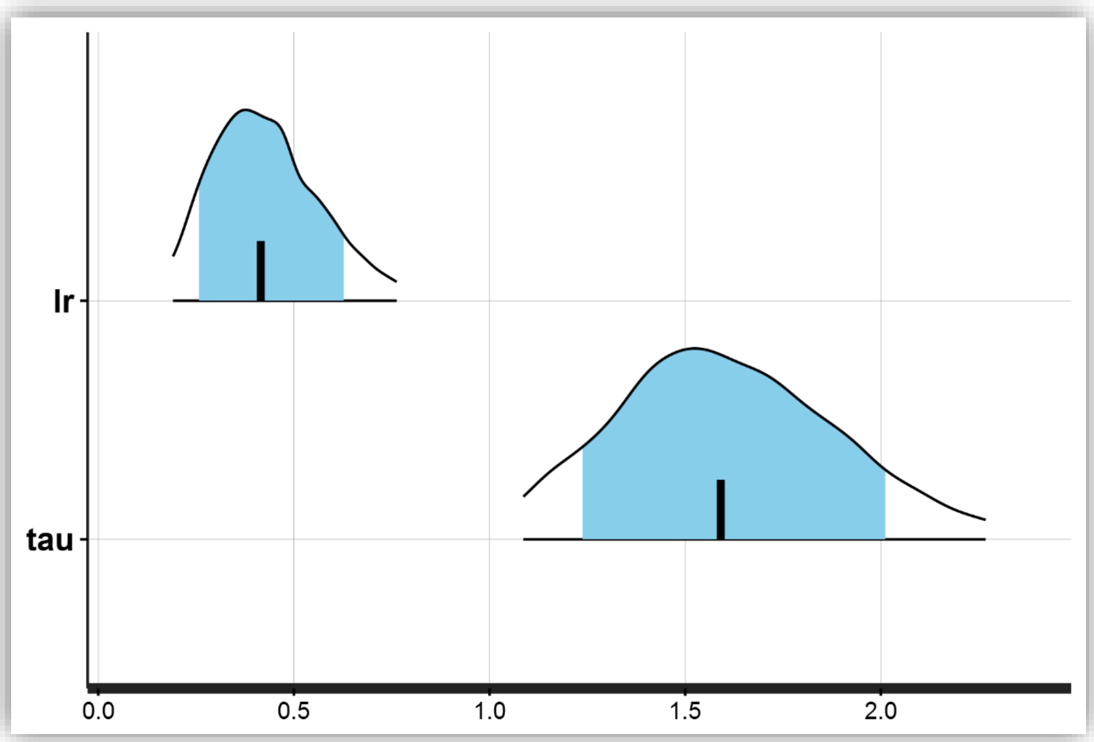
RL – MCMC Output



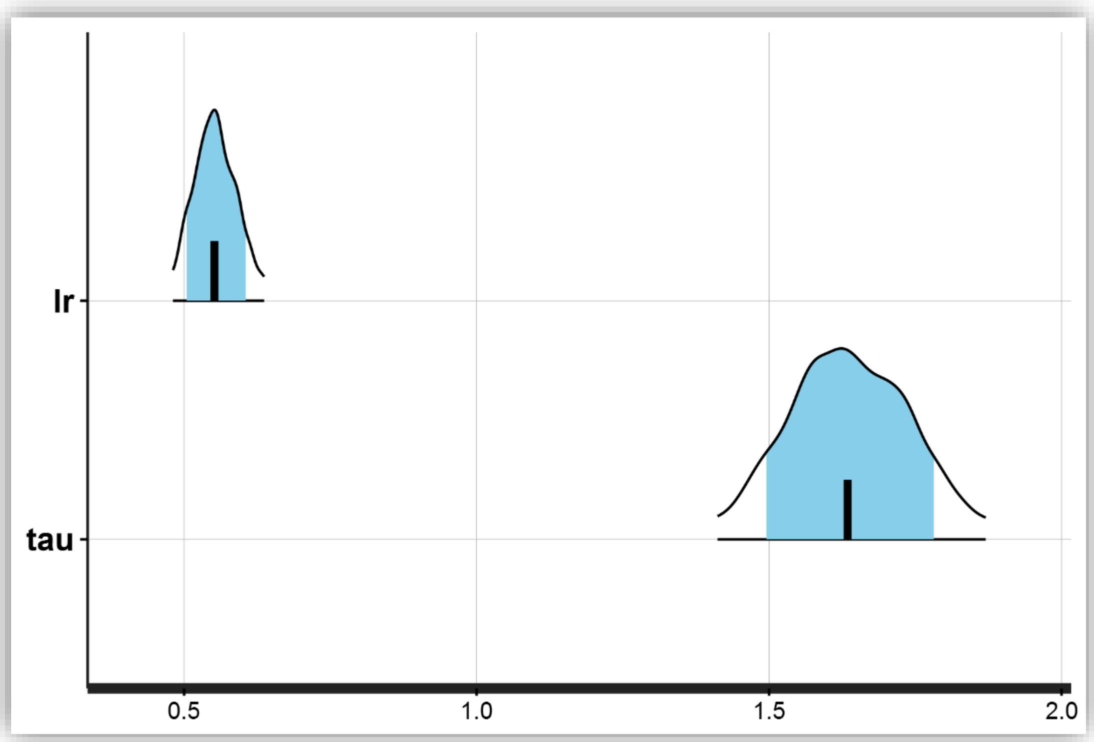


What if we have more data?

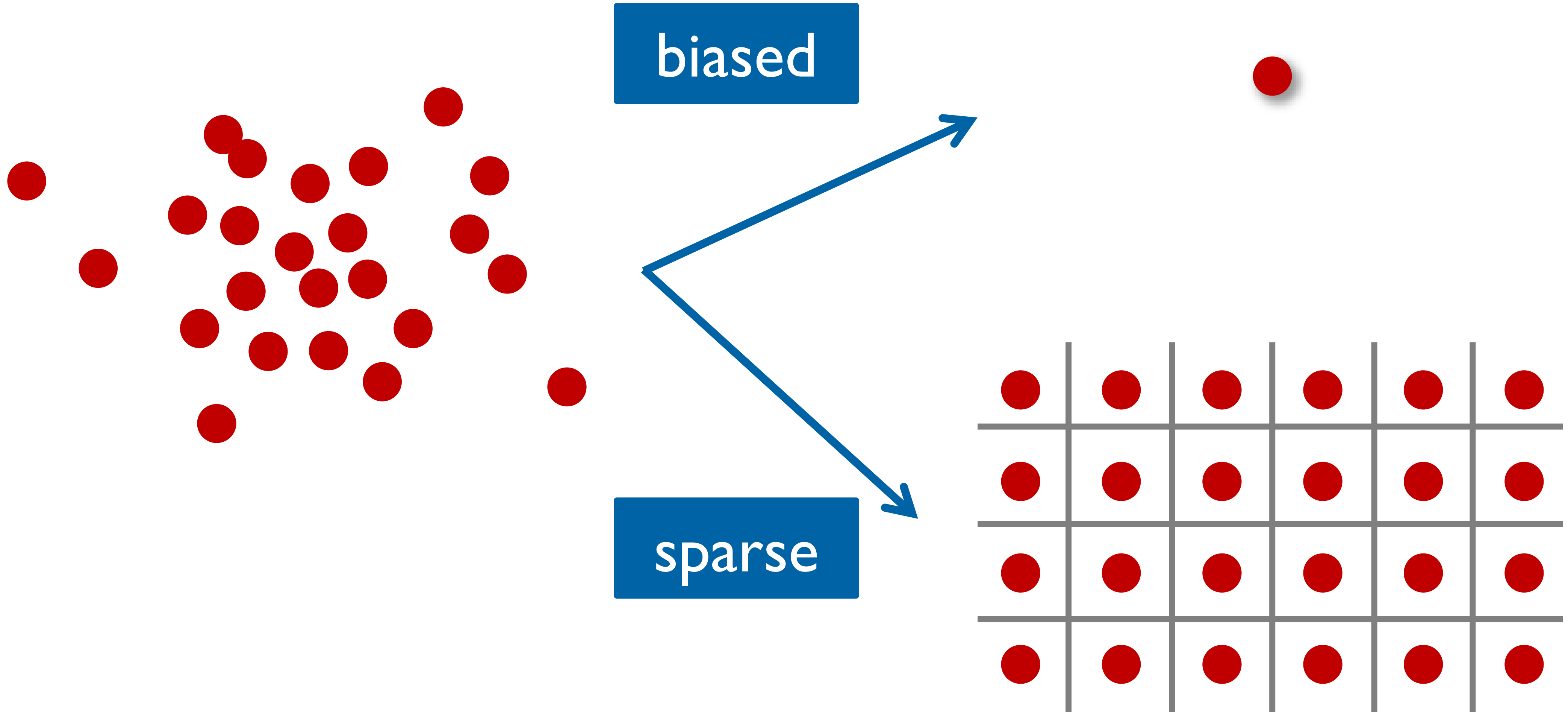
N = 1



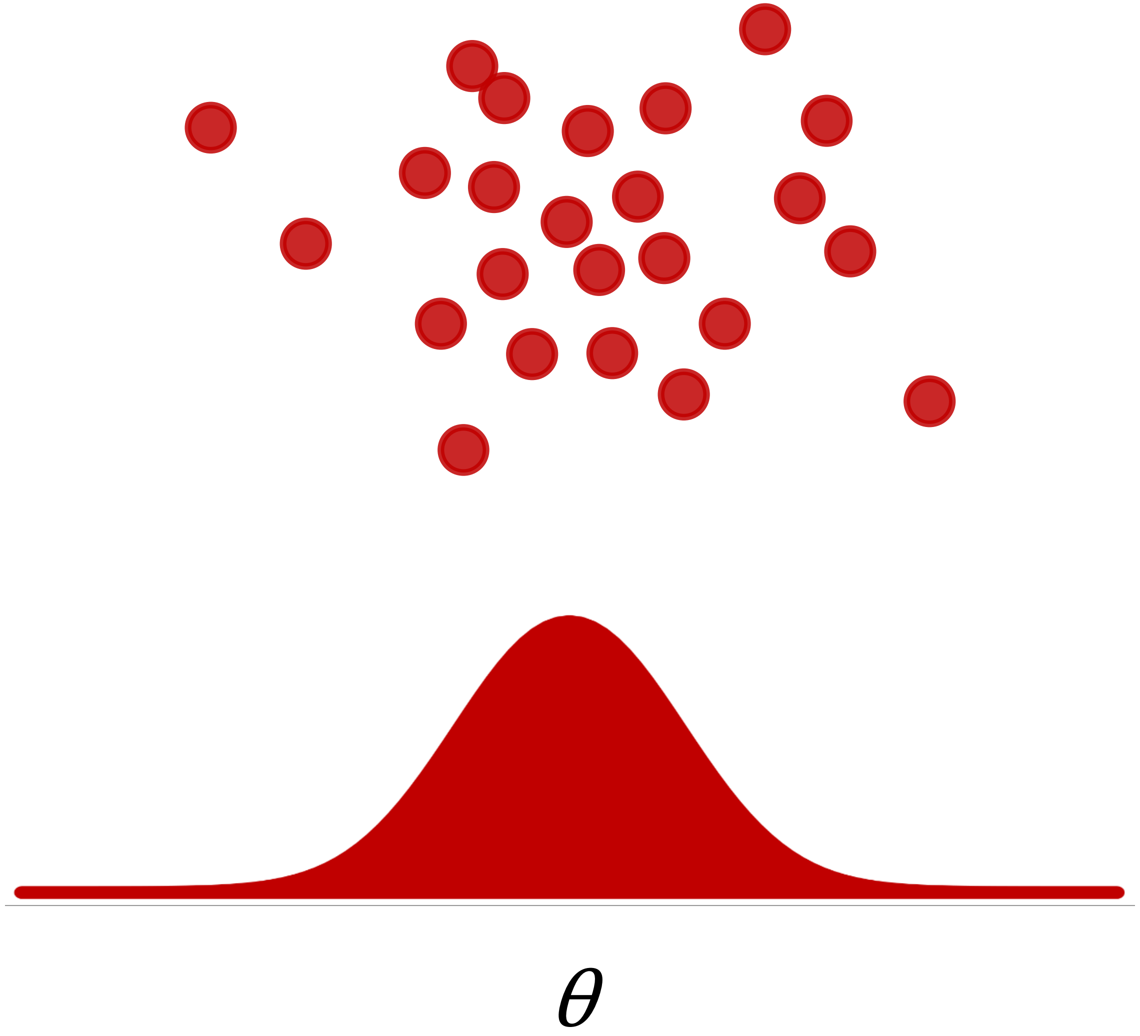
N = 10

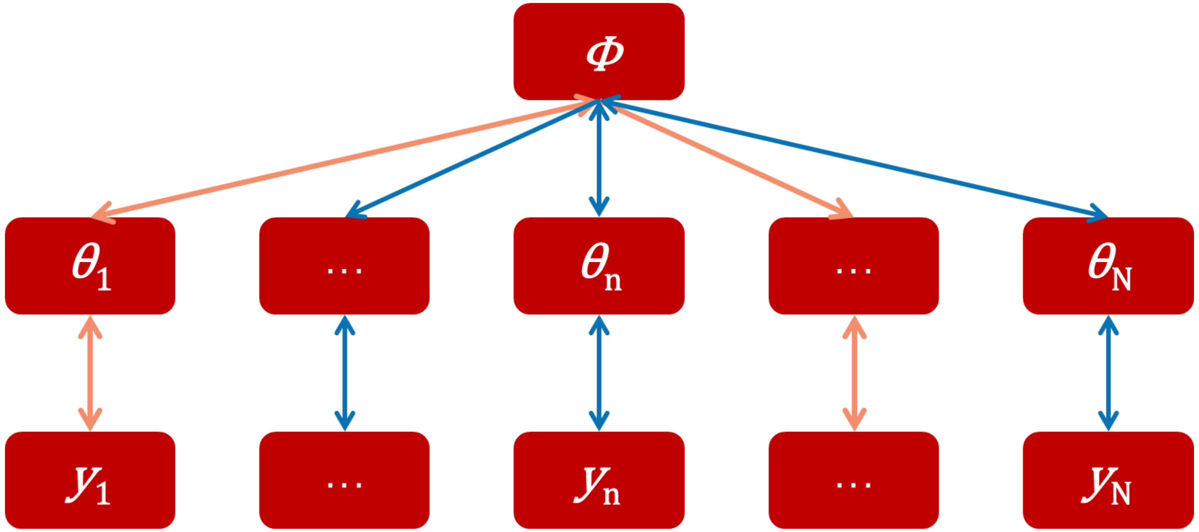


Fitting Multiple Participants

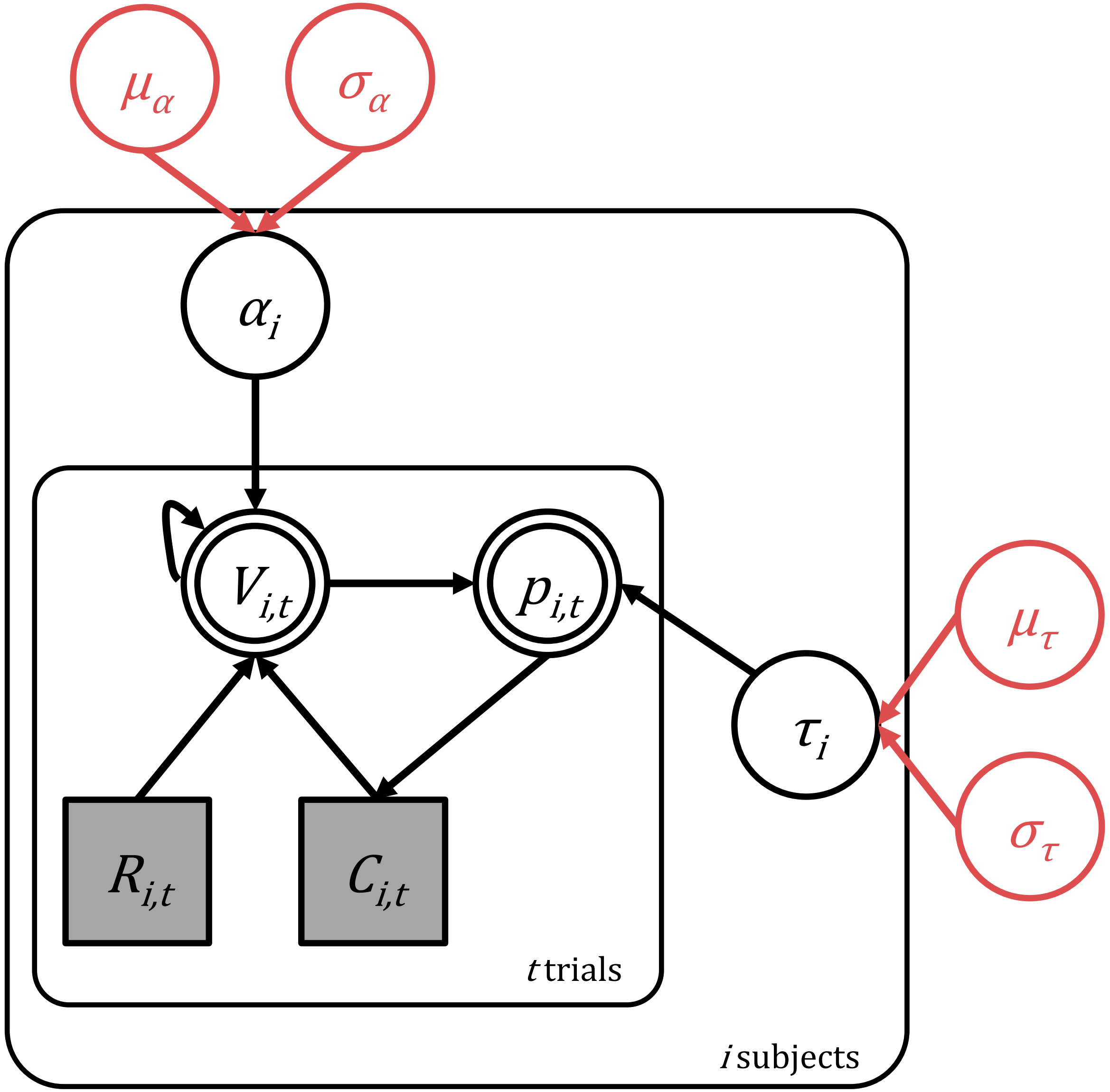


Fitting Multiple Participants with hierarchical Bayesian analysis (HBA)

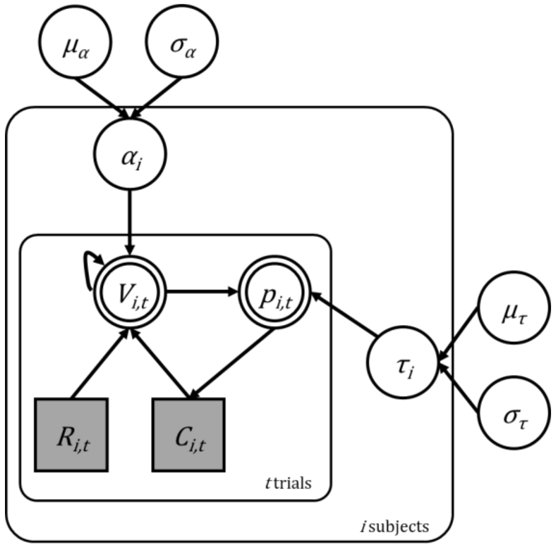




Hierarchical RL Model



HBA sounds good, but...



$$\mu_\alpha \sim Uniform(0,1)$$

$$\sigma_\alpha \sim halfCauchy(0,1)$$

$$\mu_\tau \sim Uniform(0,3)$$

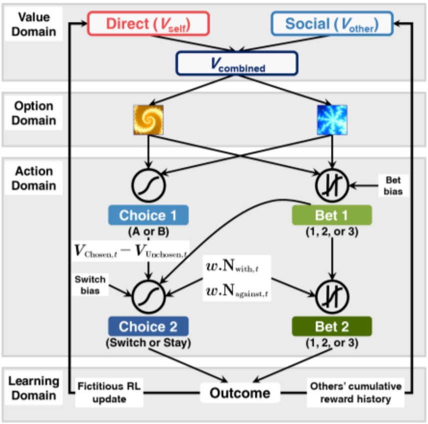
$$\sigma_\tau \sim halfCauchy(0,3)$$

$$\alpha_i \sim Normal(\mu_\alpha, \sigma_\alpha)_{\mathcal{T}(0,1)}$$

$$\tau_i \sim Normal(\mu_\tau, \sigma_\tau)_{\mathcal{T}(0,3)}$$

$$p_{i,t}(C=A) = \frac{1}{1+e^{\tau_i(V_{i,t}(B)-V_{i,t}(A))}}$$

$$V^{c}_{i,t+1} = V^{C}_{i,t} + \alpha_i (R_{i,t} - V^{C}_{i,t})$$



$$
\begin{aligned}
\mathbb{V}_{self,t} &= [V_{self,t}(\mathrm{A}), V_{self,t}(\mathrm{B})] \\
\mathbb{V}_{other,t} &= [V_{other,t}(\mathrm{A}), V_{other,t}(\mathrm{B})] \\
\mathbb{V}_t &= \beta_{vself}\, \mathbb{V}_{self,t} + \beta_{vother}\, \mathbb{V}_{other,t} \\
C1_t &\sim Categorical(Softmax(\mathbb{V}_t)) \\
U_{bet1,t} &= \beta_{bias_{b1}} + \beta_{vdiff_{b1}} (V_{chosen,C1,t} - V_{unchosen,C1,t}) \\
B1_t &\sim OrderedLogistic(U_{bet1,t} \mid \theta) \\
w.\mathrm{N}_{against,t} &= \frac{\sum_{s=1}^{K} w_{s,t}}{\sum_{s=1}^{4} w_{s,t}}, K = 0,1,\ldots,4 \\
w.\mathrm{N}_{with,t} &= \frac{\sum_{s=1}^{4-K} w_{s,t}}{\sum_{s=1}^{4} w_{s,t}} \\
V_t(\mathrm{switch}) &= \beta_{bias_{c2}} + \beta_{vdiff_{c2}} (V_{chosen,C1,t} - V_{unchosen,C1,t}) + \beta_{against}\, w.\mathrm{N}_{against,t} \\
C2 &\sim Bernoulli(V_t(\mathrm{switch})) \\
U_{bet2,t} &= \begin{cases} U_{bet1,t} + \beta_{with_{stay}}\, w.\mathrm{N}_{with,t} + \beta_{against_{stay}}\, w.\mathrm{N}_{against,t} & \text{, if } C1 = C2 \\ U_{bet1,t} + \beta_{with_{switch}}\, w.\mathrm{N}_{with,t} + \beta_{against_{switch}}\, w.\mathrm{N}_{against,t}, & \text{if } C1 \neq C2 \end{cases} \\
B2_t &\sim OrderedLogistic(U_{bet2,t} \mid \theta) \\
\Phi(x) &= \frac{1}{1+e^{-x}} \\
\delta_{self,chosen,C2,t} &= R_{self,t} - V_{self,chosen,C2,t} \\
\delta_{self,unchosen,C2,t} &= -R_{self,t} - V_{self,unchosen,C2,t} \\
V_{self,chosen,C2,t+1} &= V_{self,chosen,C2,t} + \alpha \delta_{self,chosen,C2,t} \\
V_{self,unchosen,C2,t+1} &= V_{self,unchosen,C2,t} + \alpha \delta_{self,unchosen,C2,t}
\end{aligned}
$$

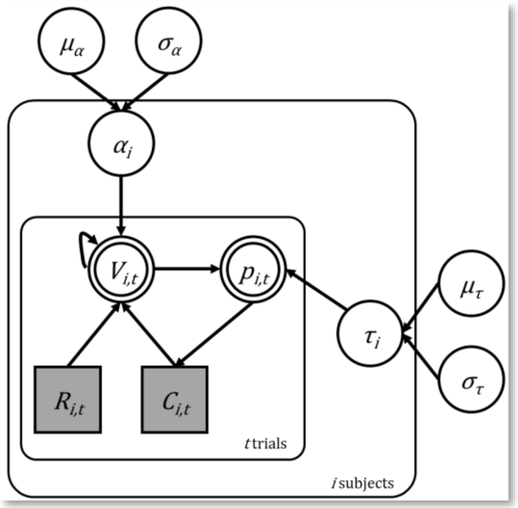
hBayesDM package

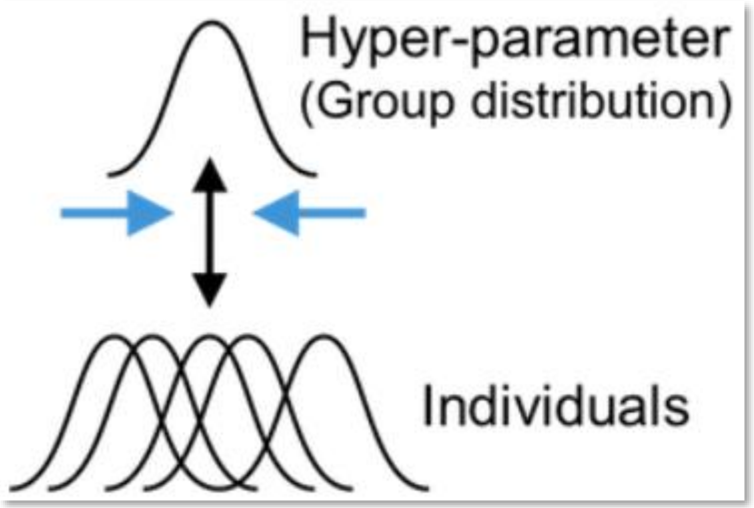
hBayesDM

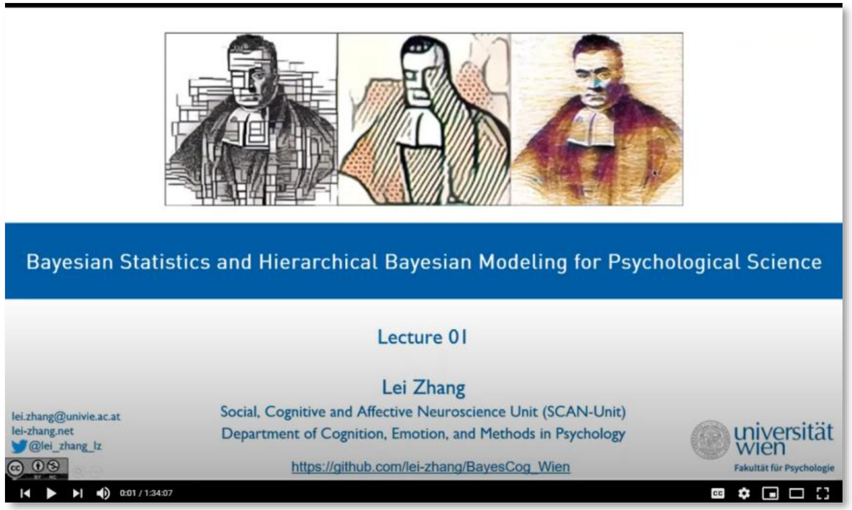
repo status Active | build passing | CRAN 1.0.2 – 2019-11-13 | downloads 33K

DOI 10.1162/CPSY_a_00002

hBayesDM (hierarchical Bayesian modeling of Decision-Making tasks) is a user-friendly package that offers hierarchical Bayesian analysis of various computational models on an array of decision-making tasks. hBayesDM uses Stan for Bayesian inference.

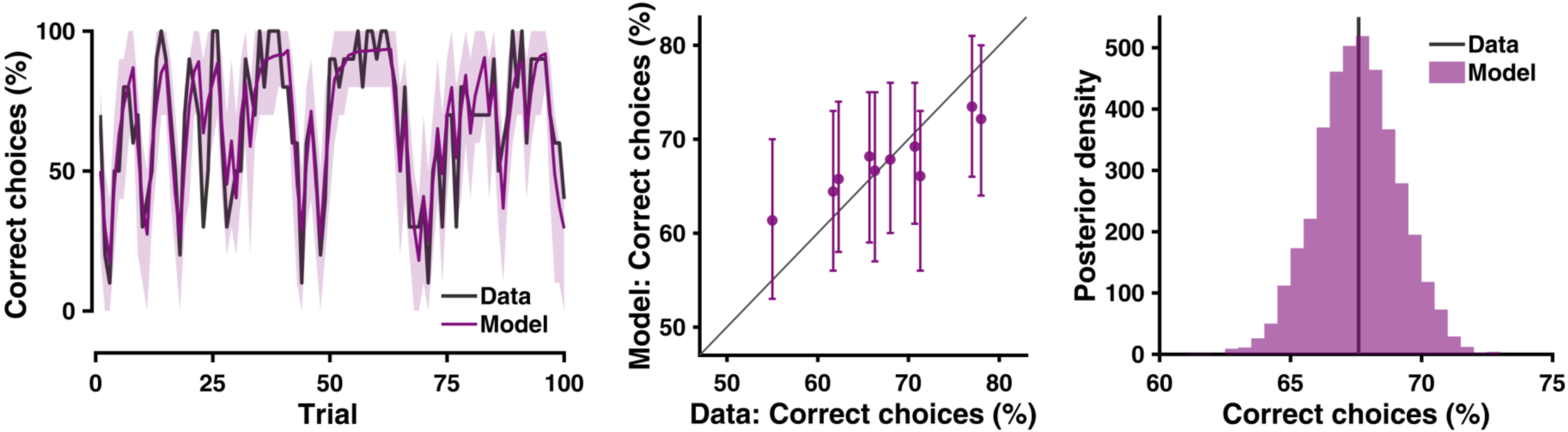






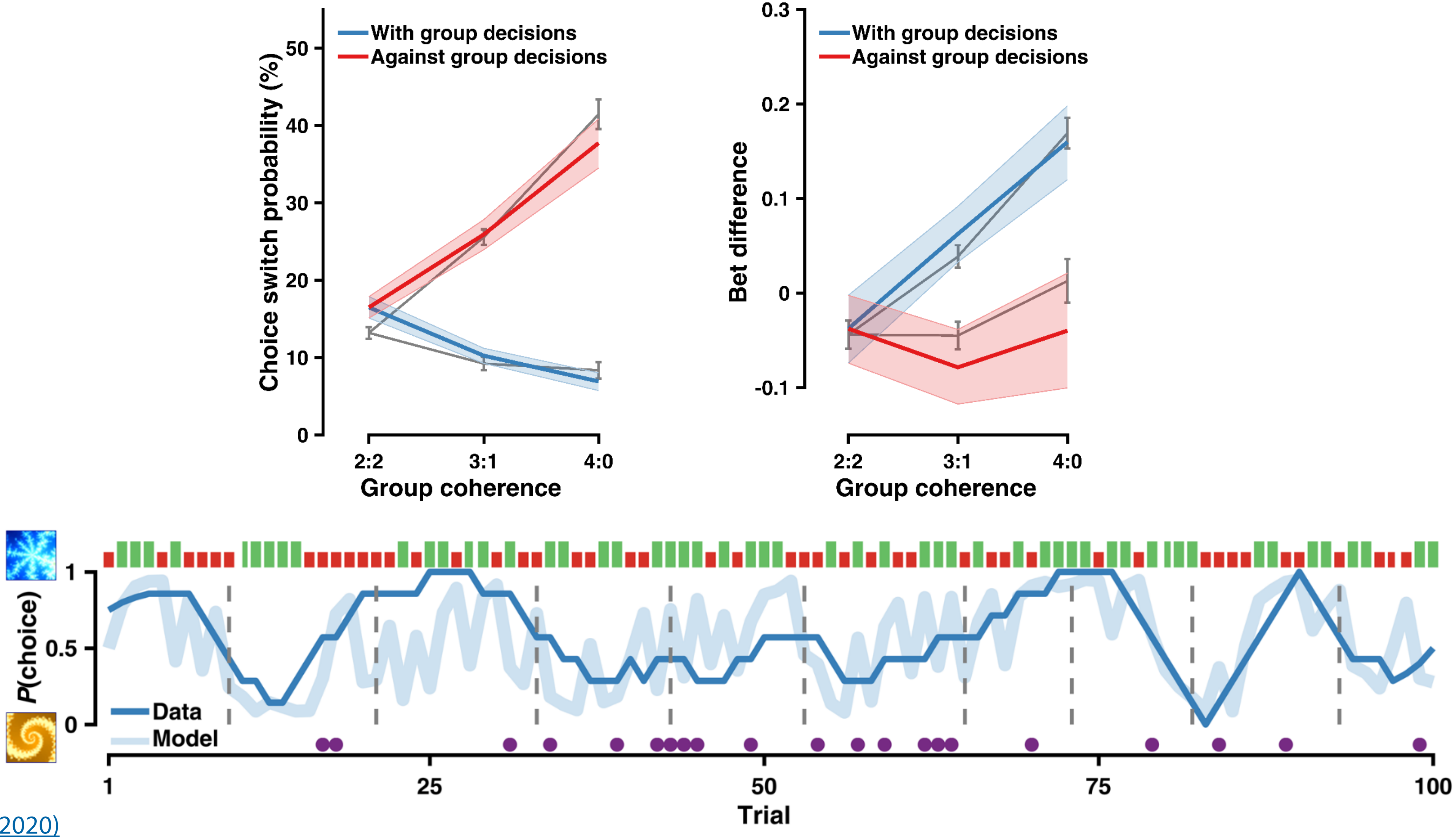
Ahn, Haines, Zhang, (2017)

Posterior predictive check:
can the model resonate with data?



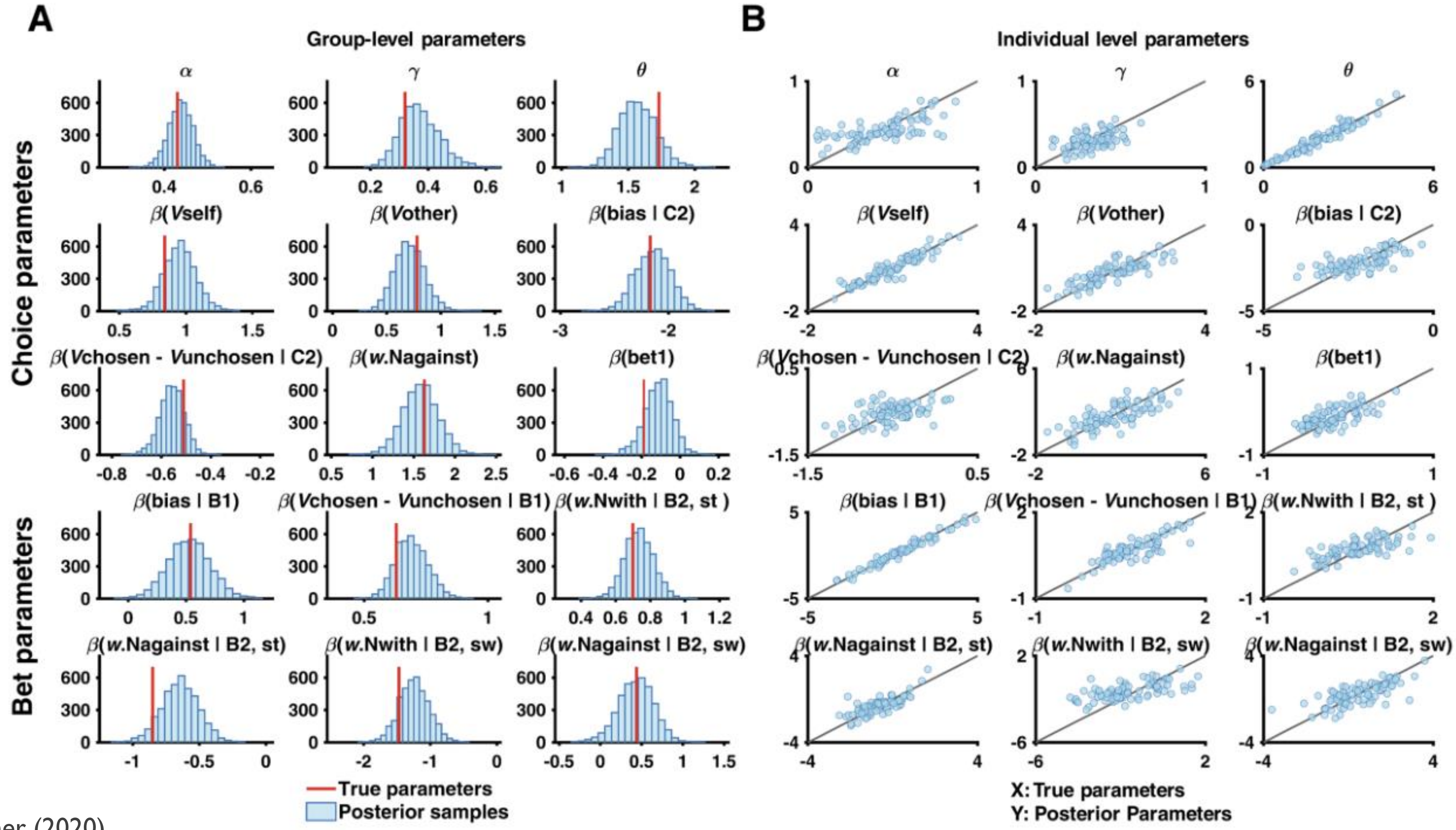
Zhang et al. (2020)

PPC One example



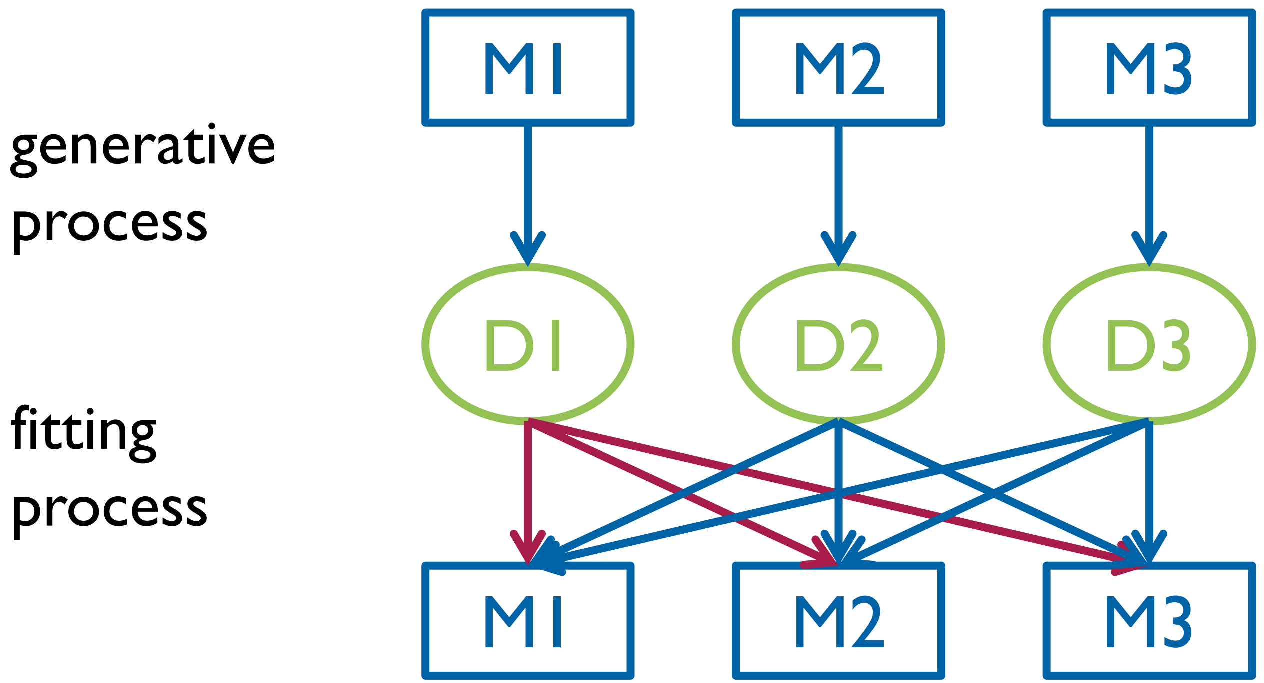
Zhang & Gläscher (2020)

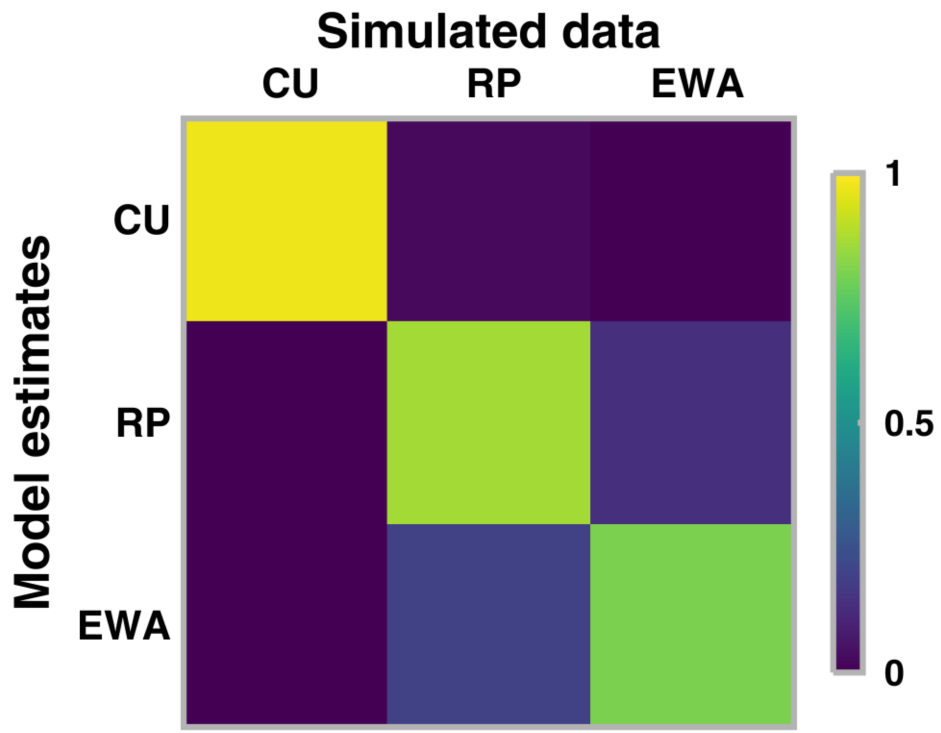
Parameter recovery:
are parameters identifiable?



Zhang & Gläscher (2020)

Model recovery:
are models identifiable?

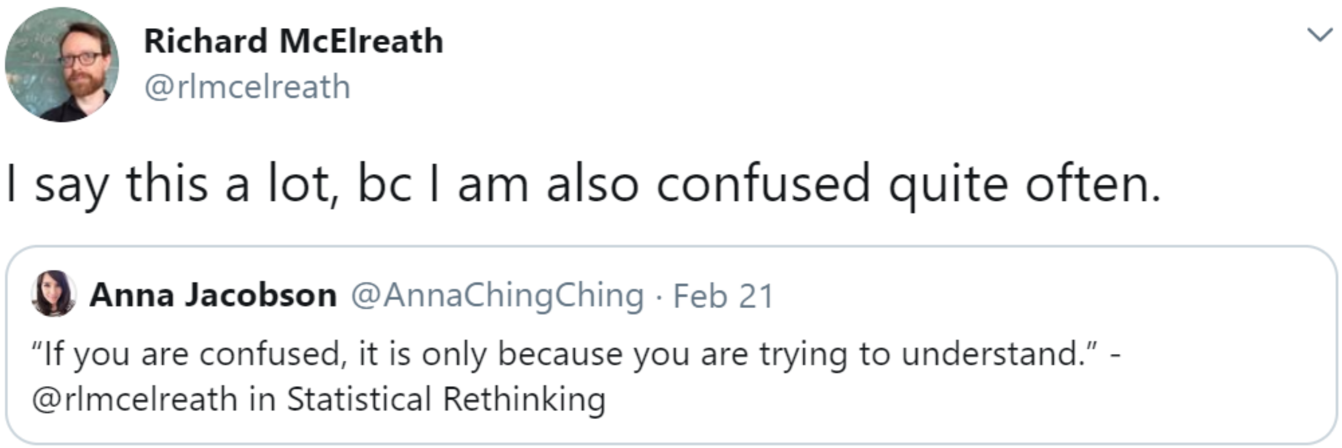




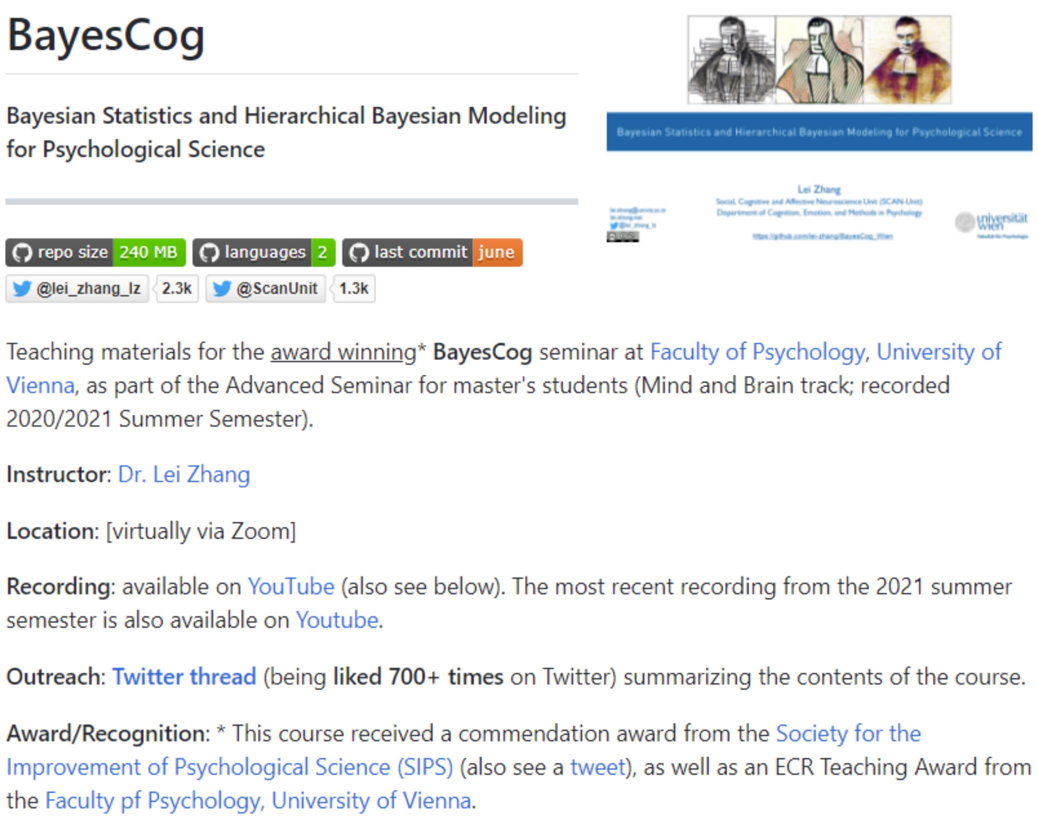
Crawley* & Zhang* et al., (2020)

Summary

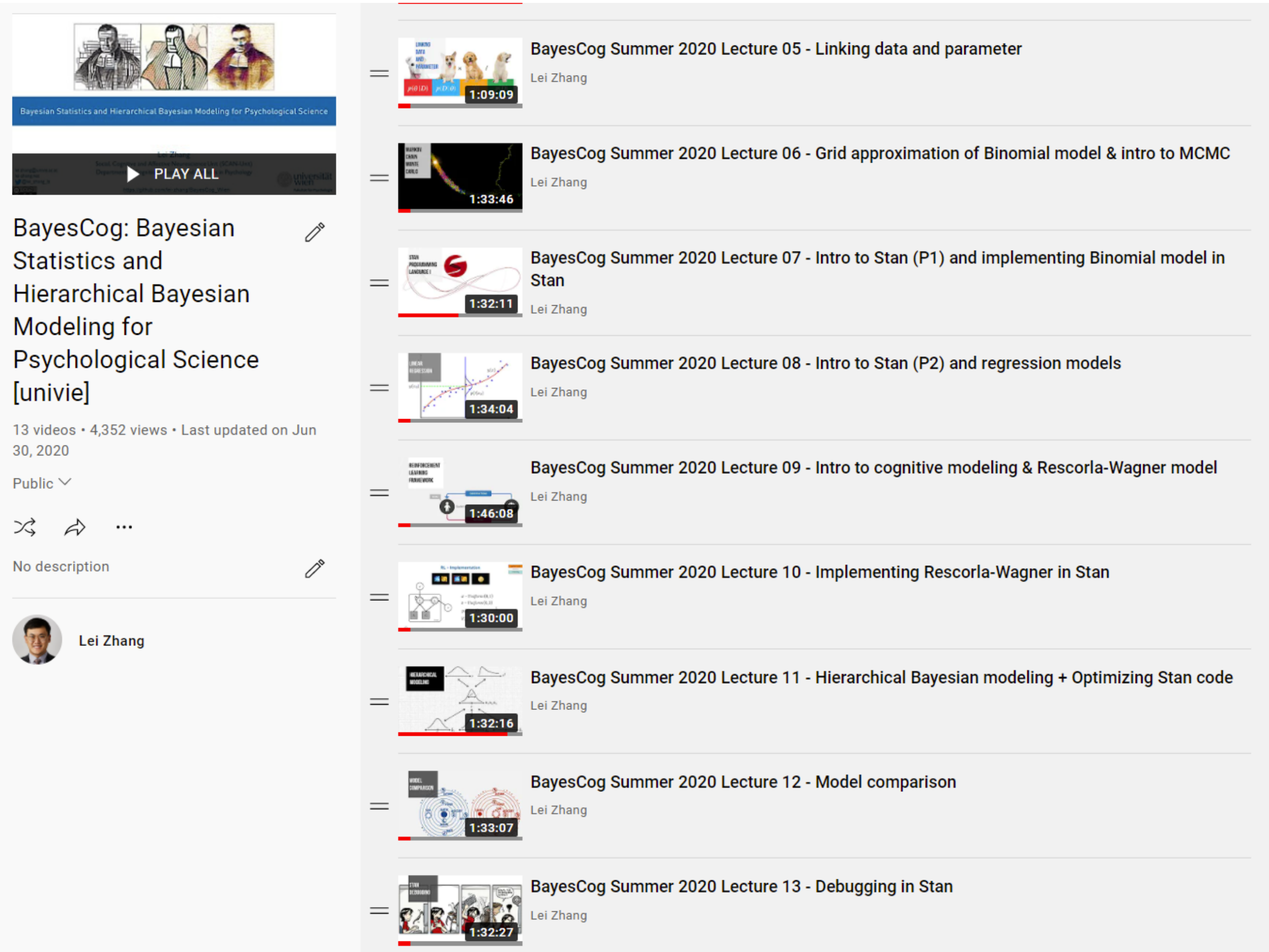
- Computational modeling is never new → don't let it fear you!
- Learn some statistics (e.g., different statistical distributions)
- Learn some math (e.g., linear algebra)
- Learn some programing (e.g., R/Python/Julia/Matlab)
- Learn to seek external help (e.g., existing packages)
- Learn in pairs; practice makes perfect!



Some more resources



https://github.com/lei-zhang/BayesCog_Wien



https://www.youtube.com/playlist?list=PLfRTb2z8k2x9gNBypgMIj3oNLF8lqM44-

Recommended reading: tutorial

Using reinforcement learning models in social neuroscience: frameworks, pitfalls and suggestions of best practices

Lei Zhang, Lukas Lengersdorff, Nace Mikus, Jan Gläscher, Claus Lamm Author Notes

Social Cognitive and Affective Neuroscience, Volume 15, Issue 6, June 2020, Pages 695–707, https://doi.org/10.1093/scan/nsaa089

https://doi.org/10.1093/scan/nsaa089

Ten simple rules for the computational modeling of behavioral data

Robert C Wilson, Anne GE Collins
University of Arizona, United States; University of California, Berkeley, United States

https://elifesciences.org/articles/49547

ACCEPTED MANUSCRIPT

Computational modelling of social cognition and behaviour—a reinforcement learning primer

Patricia L Lockwood, Miriam Klein-Flügge

Social Cognitive and Affective Neuroscience, nsaa040, https://doi.org/10.1093/scan/nsaa040
Published: 30 March 2020 **Article history** ▾

https://doi.org/10.1093/scan/nsaa040

The Importance of Falsification in Computational Cognitive Modeling

Stefano Palminteri,[1,2,*,‡] Valentin Wyart,[1,2,*,‡] and Etienne Koechlin[1,2,*]

https://doi.org/10.1016/j.tics.2017.03.011

Recommended reading: empirical work

Catecholaminergic challenge uncovers distinct Pavlovian and instrumental mechanisms of motivated (in)action

Jennifer C Swart, Monja I Froböse, Jennifer L Cook, Dirk EM Geurts, Michael J Frank, Roshan Cools, Hanneke EM den Ouden
Radboud University, The Netherlands; University of Birmingham, United Kingdom; Radboud University Medical Center, The Netherlands; Linguistic and Psychological Sciences, Brown University, United States; Brown University, United States

https://elifesciences.org/articles/22169

COGNITIVE NEUROSCIENCE

A brain network supporting social influences in human decision-making

Lei Zhang[1,2*] and Jan Gläscher[1*†]

https://advances.sciencemag.org/content/6/34/eabb4159

Social threat learning transfers to decision making in humans

Björn Lindström[a,b,c,1], Armita Golkar[c,d], Simon Jangard[c], Philippe N. Tobler[b], and Andreas Olsson[c]

[a]Department of Social Psychology, University of Amsterdam, 1018 WT Amsterdam, The Netherlands; [b]Laboratory for Social and Neural Systems Research, Department of Economics, University of Zürich, 8001 Zürich, Switzerland; [c]Section for Psychology, Department of Clinical Neuroscience, Karolinska Institutet, 171 77 Stockholm, Sweden; and [d]Department of Clinical Psychology, University of Amsterdam, 1018 WT Amsterdam, The Netherlands

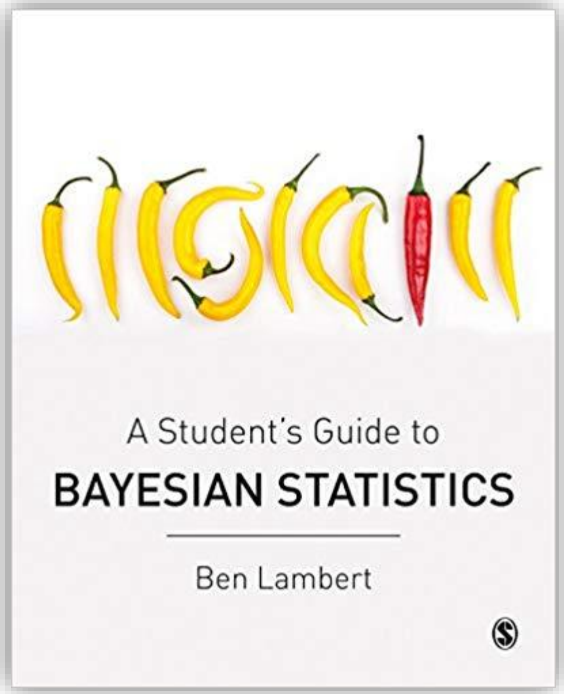
https://www.pnas.org/content/116/10/4732.abstract

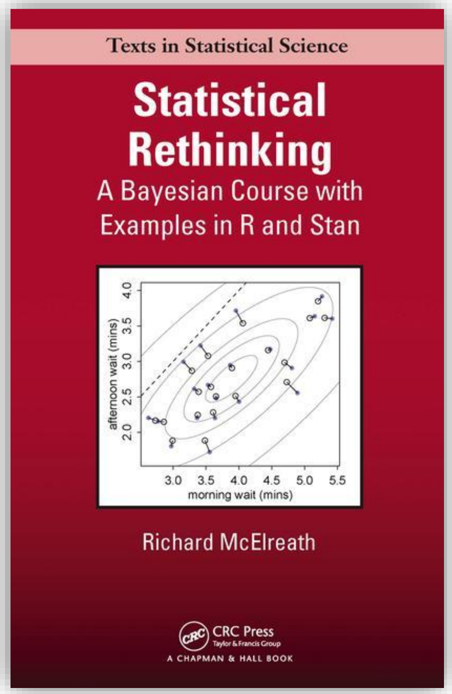
Modeling flexible behavior in childhood to adulthood shows age-dependent learning mechanisms and less optimal learning in autism in each age group

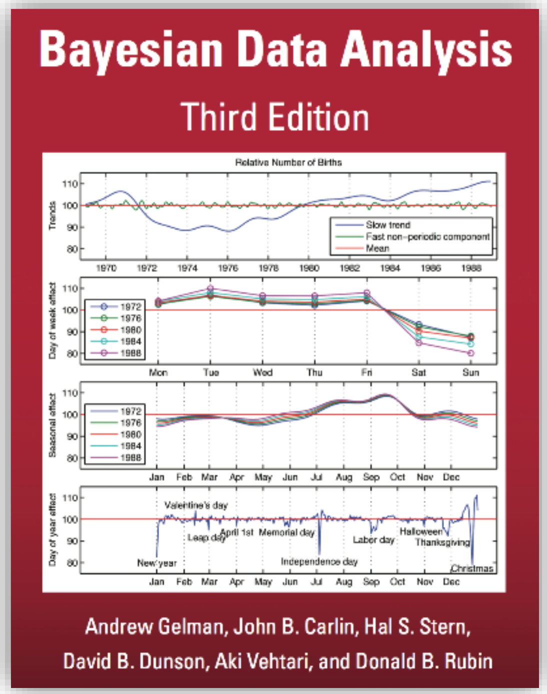
Daisy Crawley[1‡*], Lei Zhang[2,3,4‡], Emily J. H. Jones[5], Jumana Ahmad[1,6], Bethany Oakley[1], Antonia San José Cáceres[1,7], Tony Charman[8,9], Jan K. Buitelaar[10,11,12], Declan G. M. Murphy[1,9,13], Christopher Chatham[4], Hanneke den Ouden[10‡], Eva Loth[1,13‡], the EU-AIMS LEAP group[¶]

https://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.3000908

Recommended reading: book







Want to learn more about Stan?

Workshops

- StanCon 2019 Hierarchical Models
- PyData NYC 2019
- StanCon 2018 Intro Stan
- StanCon 2018 Hierarchical Models

https://mc-stan.org/workshops/

Twitter

Richard McElreath
@rlmcelreath

Quiche Lorraine,
@dan_p_simpson

\mathfrak{Michael
@betanalpha

EJ Wagenmakers
@EJWagenmakers

Stan forum

StanCon 2020. August 11-14 at Oregon State University
Announcements stancon
StanCon 2020 will be at Oregon State University! There will be two days of tutorials followed by two days of talks, open discussions, and statistical modeling. Up-to-date details at https://mc-stan.org/events/stancon202… read more

Welcome to the Stan Forums!
The Stan Forums provide a community for asking and answering questions about all aspects of Stan. Before creating a new topic please search the Forums to see if your questions has already been answered, or check out the… read more

Unable to retrieve parameters from a dynamic model
Modeling fitting-issues specification

https://discourse.mc-stan.org/

Acknowledgement

lei.zhang@univie.ac.at

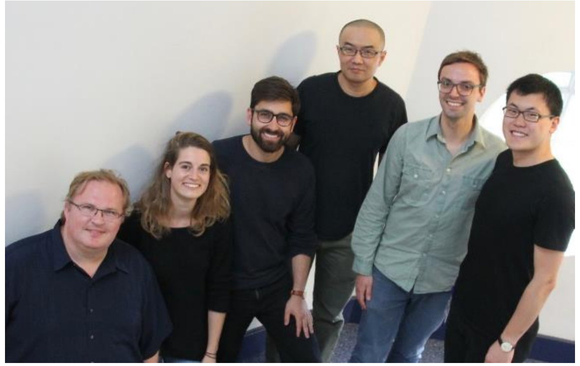
https://lei-zhang.net/

@lei_zhang_lz

@LeiZhang

@lei-zhang

Thank you!




ANY
QUESTIONS
?

Happy Computing!