

Uncertainty Signatures in Brain and Behaviour

Christoph Mathys

Scuola Internazionale Superiore di Studi Avanzati (SISSA)

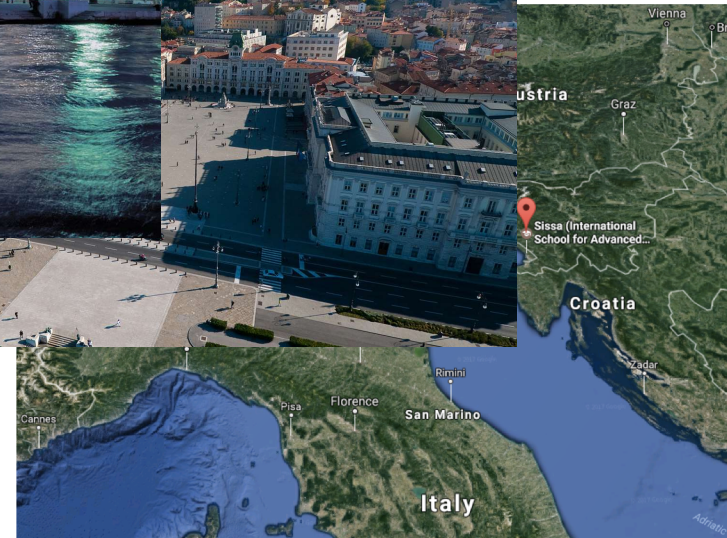
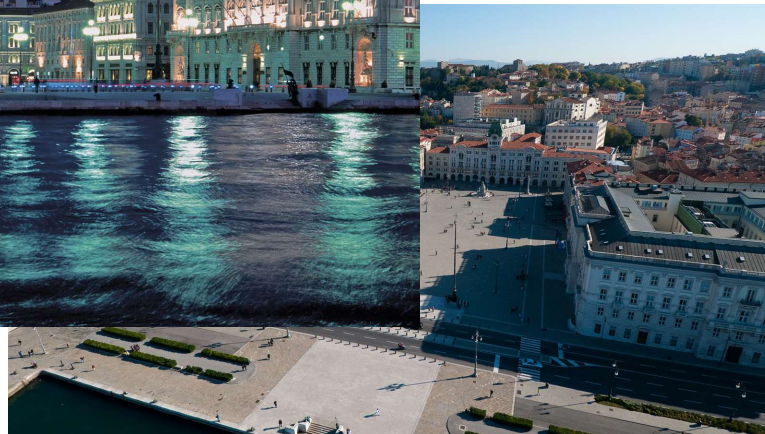
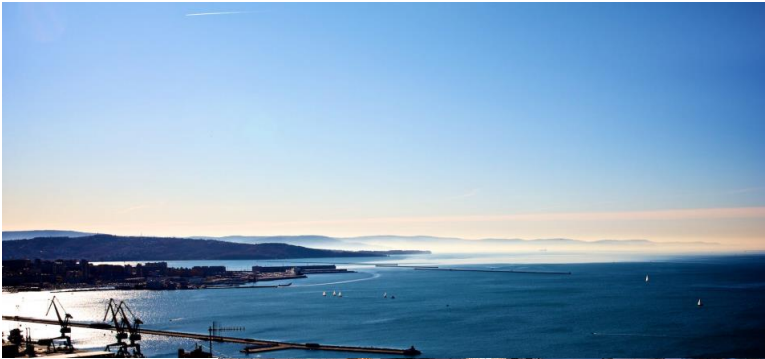
Trieste



Donders DCC Lecture

Nijmegen, 27 November 2018

- Founded 1978
- Graduate School
- 3 areas: Neuroscience, Mathematics, Physics
- 12 PhD programmes
- Cognitive Neuroscience PhD entrance exams in April and September
- 8 students admitted per year



Outline

- Reducing uncertainty by making multiple observations
- Updating the mean of multiple observations
- What if the quantity of interest changes with time?
- Rescorla-Wagner learning and exponential weighting
- Modelling observation uncertainty explicitly
- The Kalman filter
- Change in the rate of change
- Non-stationary environments and the hierarchical Gaussian filter (HGF)
- Experimental studies and results

Observations in a noisy setting

Imagine the following situation:

You're on a boat, you're lost in a storm and trying to get back to shore. A lighthouse has just appeared on the horizon, but you can only see it when you're at the peak of a wave. Your GPS etc., has all been washed overboard, but what you can still do to get an idea of your position is to measure the angle between north and the lighthouse. How exactly would you proceed?

Right. You would make multiple measurements. Why?

Right. To reduce uncertainty due to observation noise. How does that work?

Right again. Let's worry about this later and first look at an example.

Observations in a noisy setting

You're still on the boat and these are your measurements (in degrees):

76, 73, 75, 72, 77

What number are you going to base your calculation on?

Right. The mean: 74.6. Why?

Right.

- If we take the mean as our estimate, the error in our estimate is the mean of the errors in the individual measurements
- Taking the mean as maximum-likelihood estimate implies a **Gaussian error distribution**
- A Gaussian error distribution appropriately reflects our **prior** knowledge about the errors whenever we know nothing about them except perhaps their variance

Updating the mean of a series of observations

By the way: how do you calculate the mean $\bar{u}$ of $u_1, u_2, \dots, u_n$?

The usual way is to take

$$\bar{u} = \frac{1}{n} \sum_{i=1}^n u_i$$

This requires you to remember all u_i , which can become inefficient. Since the measurements arrive sequentially, we would like to update $\bar{u}$ sequentially as the u_i come in – without having to remember them.

It turns out that this is possible. After some algebra (see next slide), we get

$$\bar{u}_{n+1} = \bar{u}_n + \frac{1}{n+1} (u_{n+1} - \bar{u}_n)$$

Updating the mean of a series of observations

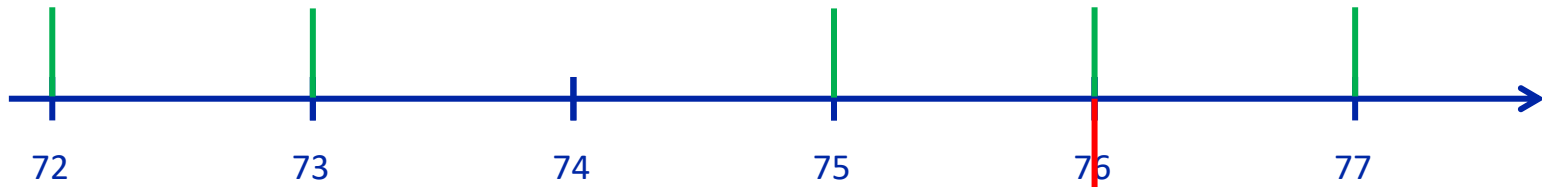
Proof of sequential update formula:

$$\begin{aligned}\bar{x}_{n+1} &= \frac{1}{n+1} \sum_{i=1}^{n+1} x_i = \frac{1}{n+1} \left(x_{n+1} + n \cdot \frac{1}{n} \sum_{i=1}^n x_i \right) = \\ &= \frac{1}{n+1} (x_{n+1} + n\bar{x}_n) = \frac{1}{n+1} (x_{n+1} - \bar{x}_n + (n+1)\bar{x}_n) \\ &= \bar{x}_n + \frac{1}{n+1} (x_{n+1} - \bar{x}_n)\end{aligned}$$

q.e.d.

Updating the mean of a series of observations

The sequential updates in our example now look like this:



$$\bar{u}_1 = 76$$

$$\bar{u}_2 = 76 + \frac{1}{2}(73 - 76) = 74.5$$

$$\bar{u}_3 = 74.5 + \frac{1}{3}(75 - 74.5) = 74.\bar{6}$$

$$\bar{u}_4 = 74.\bar{6} + \frac{1}{4}(72 - 74.\bar{6}) = 74$$

$$\bar{u}_5 = 74 + \frac{1}{5}(77 - 74) = 74.6$$

What are the building blocks of the updates we've just seen?

$$\bar{u}_{n+1} = \bar{u}_n + \frac{1}{n+1} (u_{n+1} - \bar{u}_n)$$

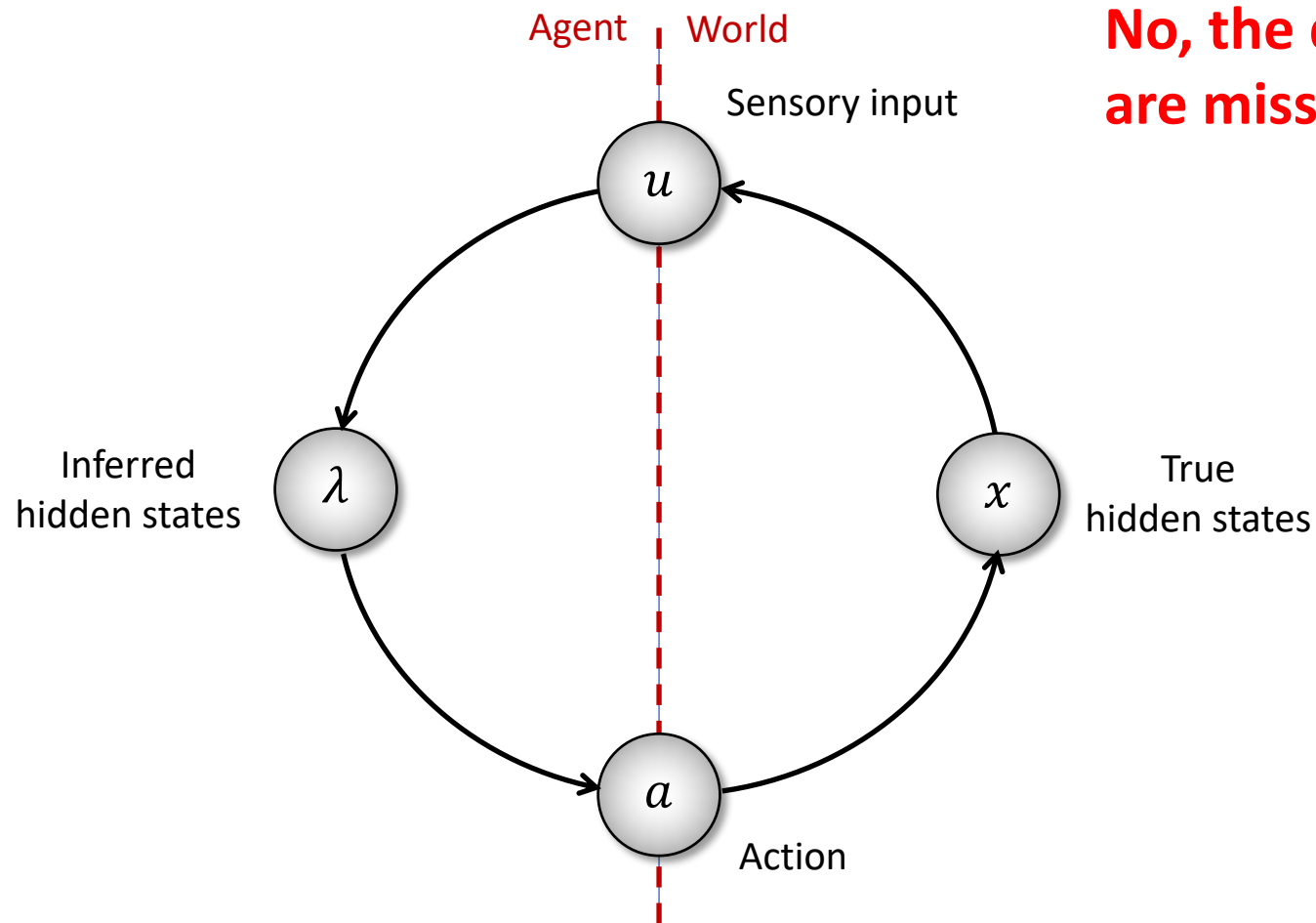
The diagram illustrates the components of the update equation $\bar{u}_{n+1} = \bar{u}_n + \frac{1}{n+1} (u_{n+1} - \bar{u}_n)$. A red arrow points from the term $\bar{u}_n$ to the label "prediction". A blue arrow points from the fraction $\frac{1}{n+1}$ to the label "weight (learning rate)". A green arrow points from the term u_{n+1} to the label "new input". A purple oval encircles the term $(u_{n+1} - \bar{u}_n)$, with a purple arrow pointing to the label "prediction error".

Is this a general pattern?

More specifically, does it generalize to Bayesian inference?

Indeed, it turns out that in many cases, Bayesian inference can be based on parameters that are updated using **precision-weighted prediction errors**.

Have we adequately described the situation of real-world agents?



No, the dynamics are missing!

What about the dynamics?

Up to now, we've only looked at inference on static quantities, but biological agents live in a continually changing world.

In our example, the boat's position changes and with it the angle to the lighthouse.

How can we take into account that old information becomes obsolete? If we don't, our learning rate becomes smaller and smaller because our equations were derived under the assumption that we're accumulating information about a stable quantity.

What's the simplest way to keep the learning rate from going too low?

Keep it constant!

So, taking the update equation for the mean of our observations as our point of departure...

$$\bar{u}_n = \bar{u}_{n-1} + \frac{1}{n}(u_n - \bar{u}_{n-1}),$$

... we simply replace $\frac{1}{n}$ with a constant $\alpha \in [0,1]$ (and the mean $\bar{u}_n$ with a generic value q_n):

$$q_n = q_{n-1} + \alpha(u_n - q_{n-1}).$$

This is called *Rescorla-Wagner learning* [although it wasn't this line of reasoning that led Rescorla & Wagner (1972) to their formulation].

How are we treating observations in Rescorla-Wagner learning?

Rewriting the RW learning rule reveals that q_n is an average weighted by α of the observation x_n and the previous value q_{n-1}

$$\begin{aligned}q_n &= q_{n-1} + \alpha(u_n - q_{n-1}) \\&= (1 - \alpha)q_{n-1} + \alpha u_n \\&= (1 - \alpha)((1 - \alpha)q_{n-2} + \alpha u_{n-1}) + \alpha u_n \\&= (1 - \alpha)^2 q_{n-2} + (1 - \alpha)\alpha u_{n-1} + \alpha u_n \\&= (1 - \alpha)^3 q_{n-3} + (1 - \alpha)^2 \alpha u_{n-2} + (1 - \alpha)\alpha u_{n-1} + \alpha u_n \\&= (1 - \alpha)^n q_0 + \alpha \sum_{i=1}^n (1 - \alpha)^{n-i} u_i\end{aligned}$$

Recursively unpacking the content of q_n reveals that **observations u_i are exponentially discounted into the past.**

How are we treating observations in Rescorla-Wagner learning?

Taking $q_0 = 0$ and $\gamma := 1 - \alpha$, we get

$$q_n = q_{n-1} + \alpha(u_n - q_{n-1})$$

$$= (1 - \alpha)q_{n-1} + \alpha u_n$$

$$= \alpha \sum_{i=1}^n (1 - \alpha)^{n-i} u_i$$

$$= (1 - \gamma) \sum_{i=1}^n \gamma^{n-i} u_i$$

$$= (1 - \gamma) \sum_{i=0}^{n-1} \gamma^i u_{n-i}$$

Does a constant learning rate solve our problems?

Partly: it implies a certain rate of forgetting because it amounts to taking only the $n = \frac{1}{\alpha}$ last data points into account. But...

... if the learning rate is supposed to reflect uncertainty due to observation noise and a changing input source, then how do we

(a) know that α reflects the right level of uncertainty at any one time, and

(b) account for changes in uncertainty if α is constant?

What we really need is an adaptive learning that accurately reflects uncertainty, and this forces us to *model uncertainty explicitly*.

Modelling uncertainty explicitly: Bayesian inference

- In mathematics, **uncertainty** about a quantity is described by the **probability distribution** over it
- Updating a probability distribution given an observation is called **Bayesian inference**.
- Bayesian inference is nothing esoteric. It is simply reasoning about uncertain quantities **without throwing any information away**.
- An agent using Bayesian inference has an advantage over one not using it since the one not using it is disregarding information.
- However, by the complete class theorem, given a likelihood and a posterior distribution, **there is always a prior distribution explaining the posterior in terms of Bayesian inference**.
- Bayesian inference is therefore a framework in whose terms we can describe the behaviour of agents. **It makes no sense to ask whether an agent (or the brain) is Bayesian** (says me - others disagree, but they're wrong).

Modelling uncertainty explicitly: updates in a simple Gaussian model

In order to understand how we can reason about uncertainty explicitly, let's **apply Bayesian inference to our little example**. Think boat, lighthouse, etc., again, but now we're doing Bayesian inference.

Before we make the next observation, our belief about the true value of the hidden state x can be described by a **Gaussian prior**:

$$p(x) = \mathcal{N}(x; \mu_x, \pi_x^{-1})$$

The **likelihood** of an observation u is also Gaussian, with precision π_u :

$$p(u|x) = \mathcal{N}(u; x, \pi_u^{-1})$$

Bayes' rule now tells us that the **posterior** is Gaussian again:

$$p(x|u) = \frac{p(u|x)p(x)}{\int p(u|x')p(x')dx'} = \mathcal{N}(x; \mu_{x|u}, \pi_{x|u}^{-1})$$

Updates in a simple Gaussian model

Here's how the updates to the sufficient statistics μ and π describing our belief look like:

$$\pi_{x|u} = \pi_x + \pi_u$$
$$\mu_{x|u} = \mu_x + \frac{\pi_u}{\pi_{x|u}} (u - \mu_x)$$

The diagram illustrates the components of the mean update equation. A red arrow points from the text 'prediction' to the term μ_x . A blue arrow points from the text 'weight (learning rate)' to the fraction $\frac{\pi_u}{\pi_{x|u}}$. A purple oval encircles the term $(u - \mu_x)$, with a blue arrow pointing from the text 'prediction error' to it. To the right of the equation, a blue arrow points from the text 'weight (learning rate) = $\frac{\text{how much we're learning here}}{\text{how much we already know}}$ ' to the fraction $\frac{\pi_u}{\pi_{x|u}}$.

The mean is updated by an uncertainty-weighted (more specifically: precision-weighted) prediction error.

The size of the update is proportional to the likelihood precision and inversely proportional to the posterior precision.

This pattern is not specific to the univariate Gaussian case, but generalizes to Bayesian updates for all exponential families of likelihood distributions with conjugate priors (i.e., to all formal descriptions of inference you are ever likely to need).

Dealing with nonstationary environments: the Kalman filter

So far, so good: we have learnt how to model uncertainty resulting from observation noise explicitly

However, we face the same problem as when we updated the mean of a time series: in making updates from sequential observations in this way, **we are assuming that the underlying hidden state x is stationary.**

Relaxing this assumption and replacing it with a Gaussian random walk gives us the **Kalman filter**:

$$p(x^{(k)}|x^{(k-1)}, \vartheta) = \mathcal{N}(x^{(k)}; x^{(k-1)}, \vartheta)$$

$$p(u^{(k)}|x^{(k)}, \sigma) = \mathcal{N}(u^{(k)}; x^{(k)}, \sigma)$$

This gives us the **likelihood**

$$\begin{aligned} p(u^{(k)}|x^{(k-1)}, \sigma, \vartheta) &= \int p(u^{(k)}|x^{(k)}, \sigma) p(x^{(k)}|x^{(k-1)}, \vartheta) dx^{(k)} \\ &= \int \mathcal{N}(u^{(k)}; x^{(k)}, \sigma) \mathcal{N}(x^{(k)}; x^{(k-1)}, \vartheta) dx^{(k)} \\ &= \mathcal{N}(u^{(k)}; x^{(k-1)}, \sigma + \vartheta) \end{aligned}$$

Combining this with the **prior**

$$p(x^{(k-1)}) = \mathcal{N}(x^{(k-1)}; \mu_x^{(k-1)}, 1/\pi_x^{(k-1)}), \dots$$

Dealing with nonstationary environments: the Kalman filter

..., by the same method as before, we get the **posterior**

$$p(x^{(k)}) = \mathcal{N}\left(x^{(k)}; \mu_x^{(k)}, 1/\pi_x^{(k)}\right)$$

with

$$\pi_x^{(k)} = \pi_x^{(k-1)} + \frac{1}{\sigma + \vartheta} = \pi_x^{(k-1)} + \pi_u$$

$$\mu_x^{(k)} = \mu_x^{(k-1)} + \frac{\pi_u}{\pi_x^{(k)}} \left(u^{(k)} - \mu_x^{(k-1)} \right)$$

The Kalman filter is optimal for linear dynamic systems.

Unfortunately, except for simple physical systems, **the world is not linear**. Living organisms need to be able to filter inputs whose rate of change changes, in other words: **processes whose volatility is volatile**.

Needed: an adaptive learning rate that accurately reflects uncertainty

This requires us to think a bit more about what kinds of uncertainty we are dealing with.

A possible taxonomy of uncertainty is (cf. Yu & Dayan, 2003; Payzan-LeNestour & Bossaerts, 2011):

(a) **outcome uncertainty** that remains unaccounted for by the model, called *risk* by economists (π_u in our Bayesian example); this uncertainty remains even when we know all parameters exactly,

(b) **informational** or *expected* uncertainty about the value of model parameters ($\pi_{\theta|x}$ in the Bayesian example),

(c) **environmental** or *unexpected* uncertainty owing to changes in model parameters (not accounted for in our Bayesian example, hence unexpected).

An adaptive learning rate that accurately reflects uncertainty

Various efforts have been made to come up with an adaptive learning rate:

- Kalman (1960)
- Sutton (1992)
- Nassar et al. (2010)
- Payzan-LeNestour & Bossaerts (2011)
- Mathys et al. (2011)
- Wilson et al. (2013)

Many do not model uncertainty explicitly, and if they do, they often rely on assumptions of linearity in the generative process or do not optimize an objective function like surprise.

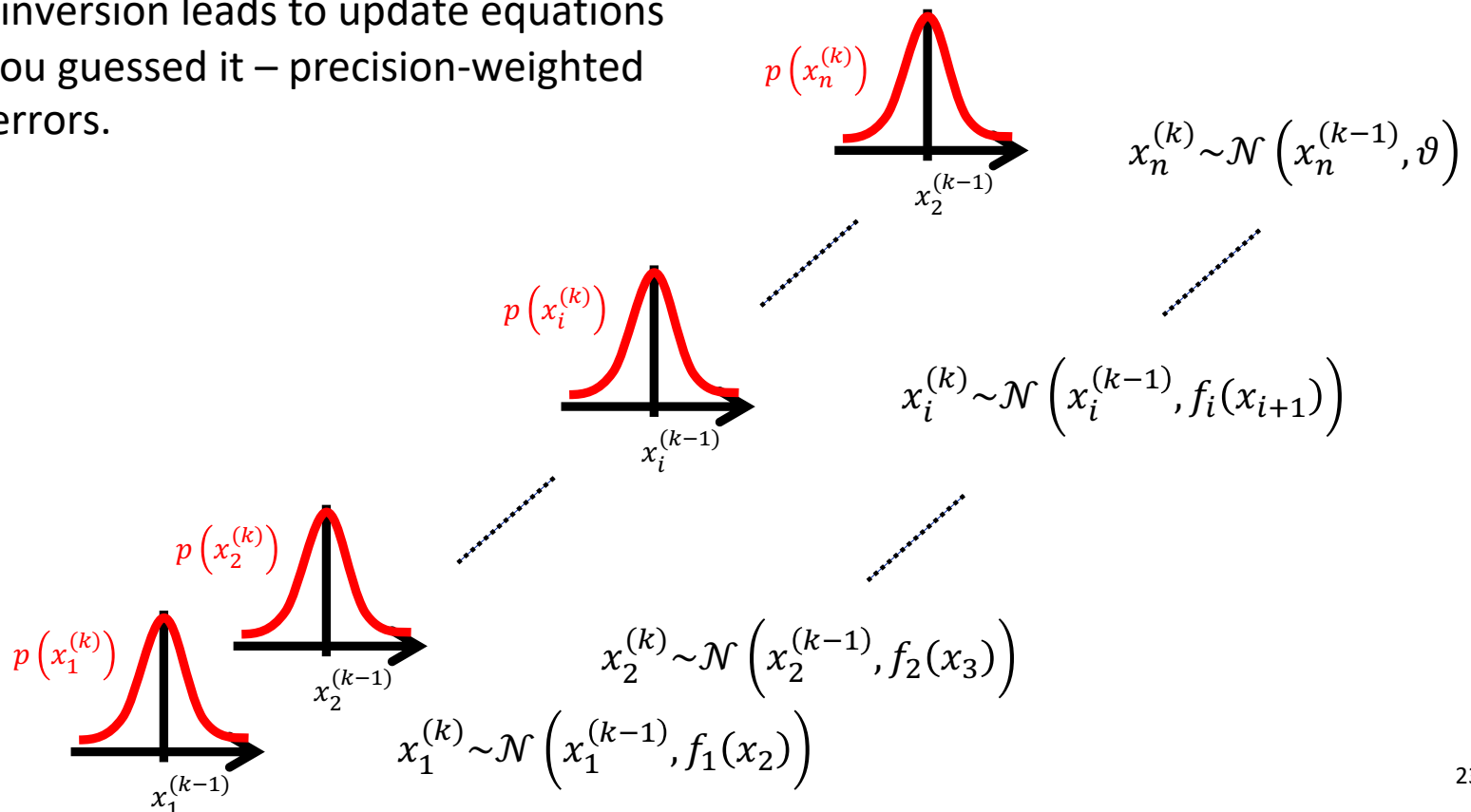
Mathys et al. use a generic non-linear hierarchical Bayesian model that allows us to derive update equations that are optimal in the sense that they minimize surprise.

The hierarchical Gaussian filter

(HGF, Mathys et al., 2011; 2014)

The HGF provides a generic solution to the problem of adapting one's learning rate in a volatile environment.

Variational inversion leads to update equations that are – you guessed it – precision-weighted prediction errors.



Variational inversion and update equations

- Inversion proceeds by introducing a mean field approximation and fitting quadratic approximations to the resulting variational energies (Mathys et al., 2011).
- This leads to **simple one-step update equations** for the sufficient statistics (mean and precision) of the approximate Gaussian posteriors of the states x_i .
- The updates of the means have the same structure as value updates in Rescorla-Wagner learning:

$$\Delta\mu_i \propto \frac{\hat{\pi}_{i-1}}{\pi_i} \delta_{i-1}$$

Predictions determine learning rate

Prediction error

- Furthermore, the updates are **precision-weighted prediction errors**.

Precision-weighting of volatility updates

Comparison to the simple non-hierarchical Bayesian update:

HGF:
$$\mu_i^{(k)} = \mu_i^{(k-1)} + \frac{1}{2} \kappa_{i-1} v_{i-1}^{(k)} \cdot \frac{\hat{\pi}_{i-1}^{(k)}}{\pi_i^{(k)}} \cdot \delta_{i-1}^{(k)}$$

Precision-weighted prediction error

Simple Gaussian:
$$\mu_{\vartheta|x} = \mu_{\vartheta} + \frac{\pi_{\varepsilon}}{\pi_{\vartheta|x}} (x - \mu_{\vartheta})$$

Updates at the outcome level

At the outcome level (i.e., at the very bottom of the hierarchy), we have

$$u^{(k)} \sim \mathcal{N} \left(x_1^{(k)}, \hat{\pi}_u^{-1} \right)$$

This gives us the following update for our belief on x_1 (our quantity of interest):

$$\pi_1^{(k)} = \hat{\pi}_1^{(k)} + \hat{\pi}_u$$
$$\mu_1^{(k)} = \mu_1^{(k-1)} + \frac{\hat{\pi}_u}{\pi_1^{(k)}} \left(u^{(k)} - \mu_1^{(k-1)} \right)$$

The familiar structure again – but now with a learning rate that is responsive to all kinds of uncertainty, including environmental (unexpected) uncertainty.

The learning rate in the HGF

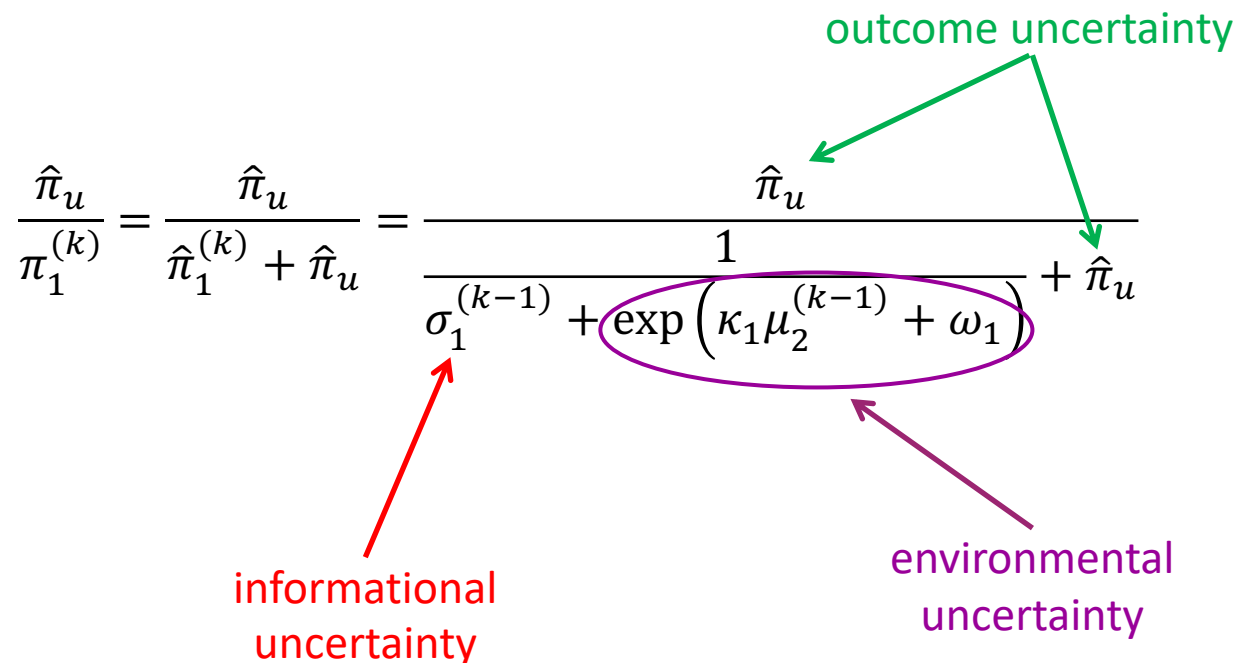
Unpacking the learning rate, we see:

$$\frac{\hat{\pi}_u}{\pi_1^{(k)}} = \frac{\hat{\pi}_u}{\hat{\pi}_1^{(k)} + \hat{\pi}_u} = \frac{\hat{\pi}_u}{\frac{1}{\sigma_1^{(k-1)} + \exp(\kappa_1 \mu_2^{(k-1)} + \omega_1)} + \hat{\pi}_u}$$

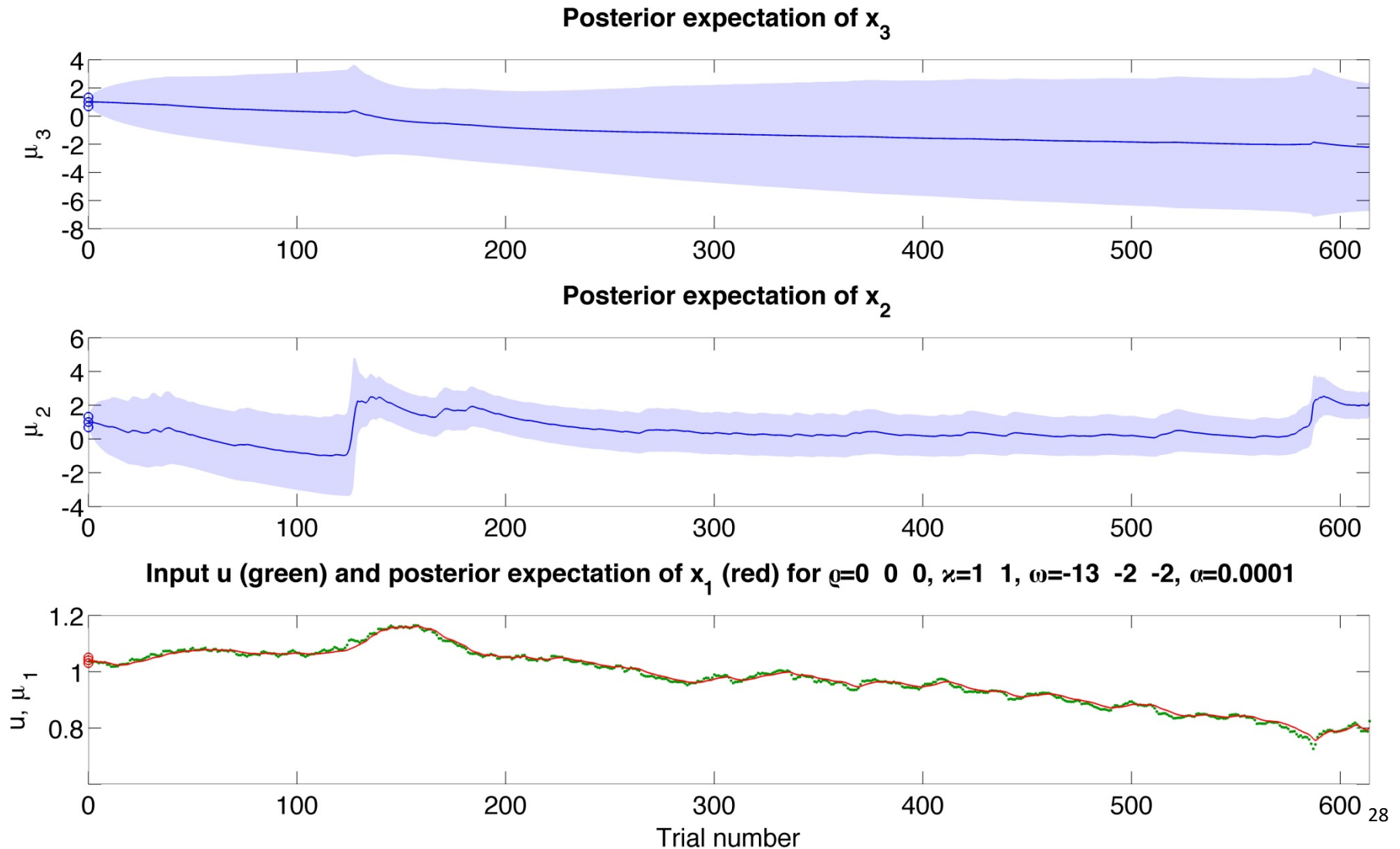
outcome uncertainty

informational uncertainty

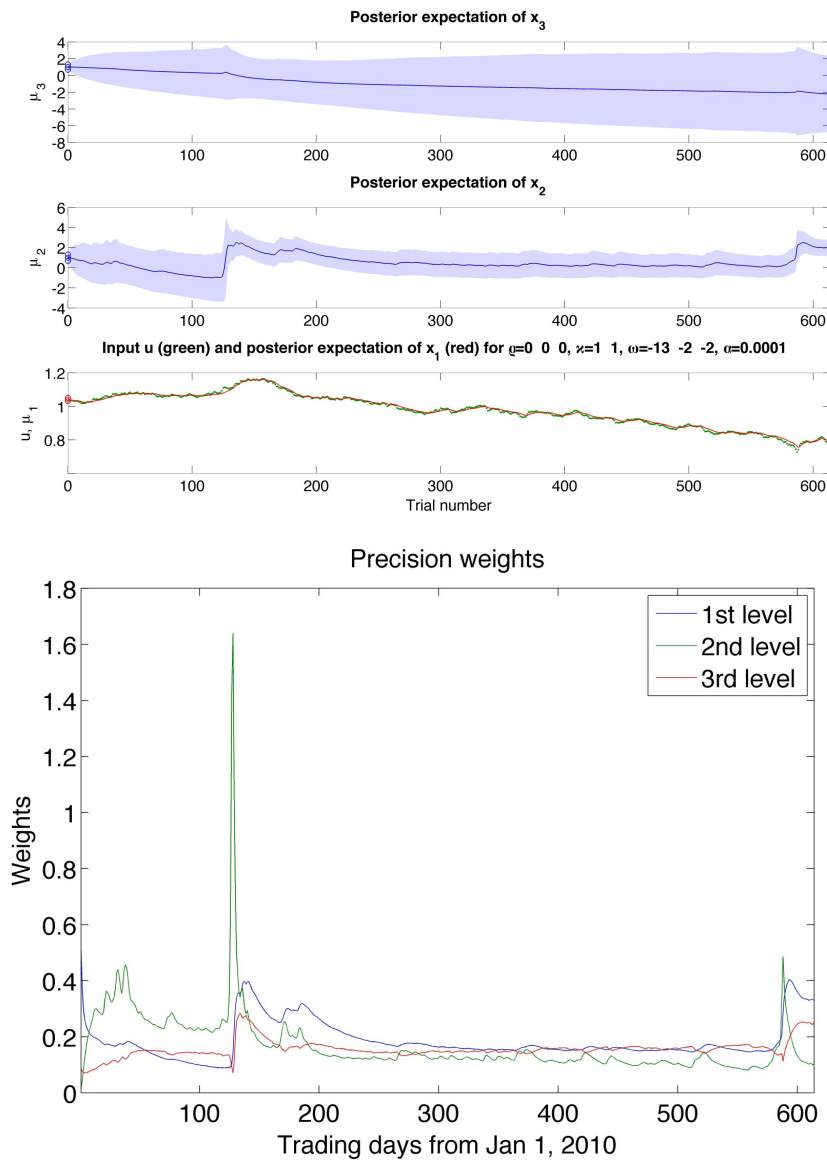
environmental uncertainty



3-level HGF for continuous observations

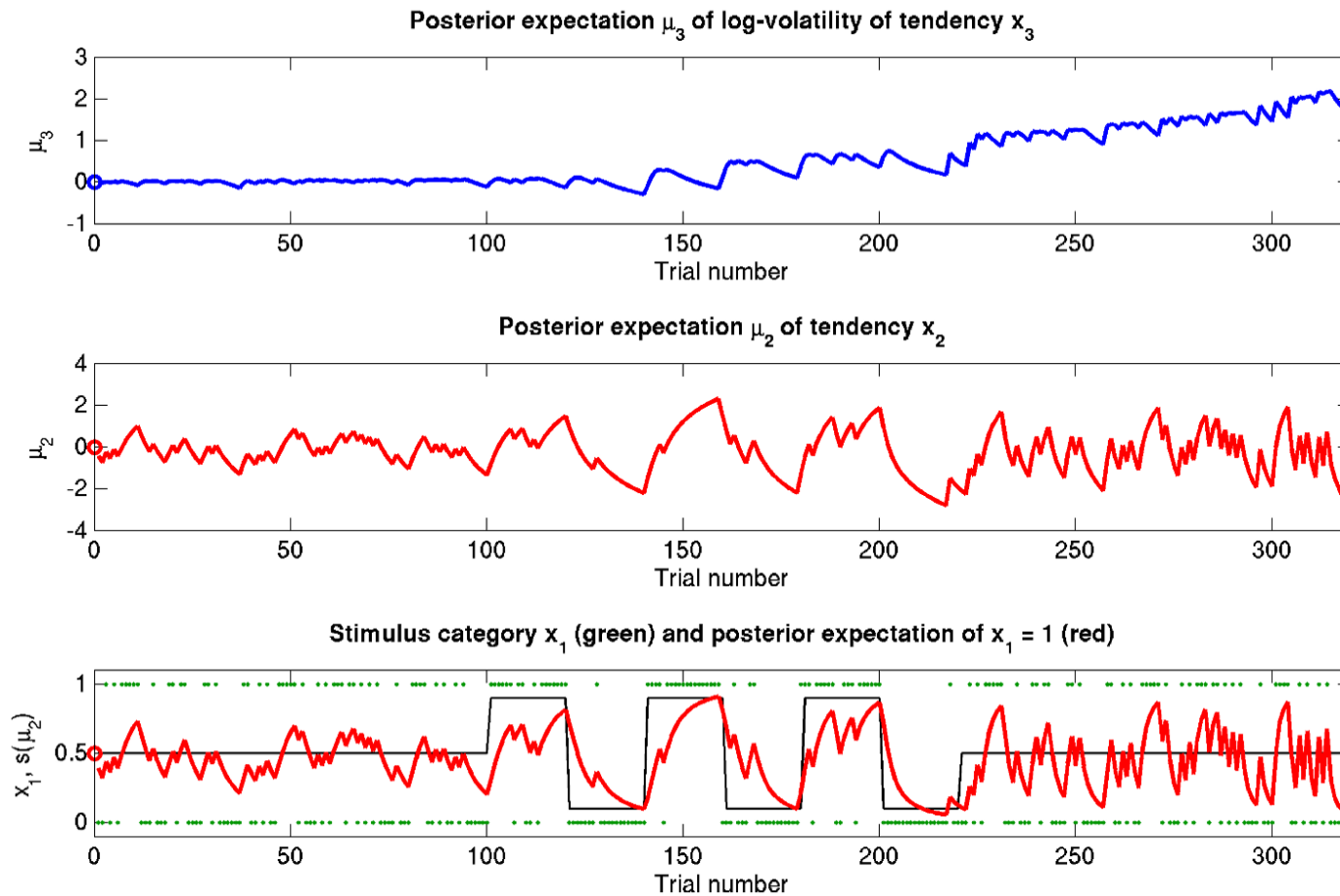


Example of precision weight trajectory

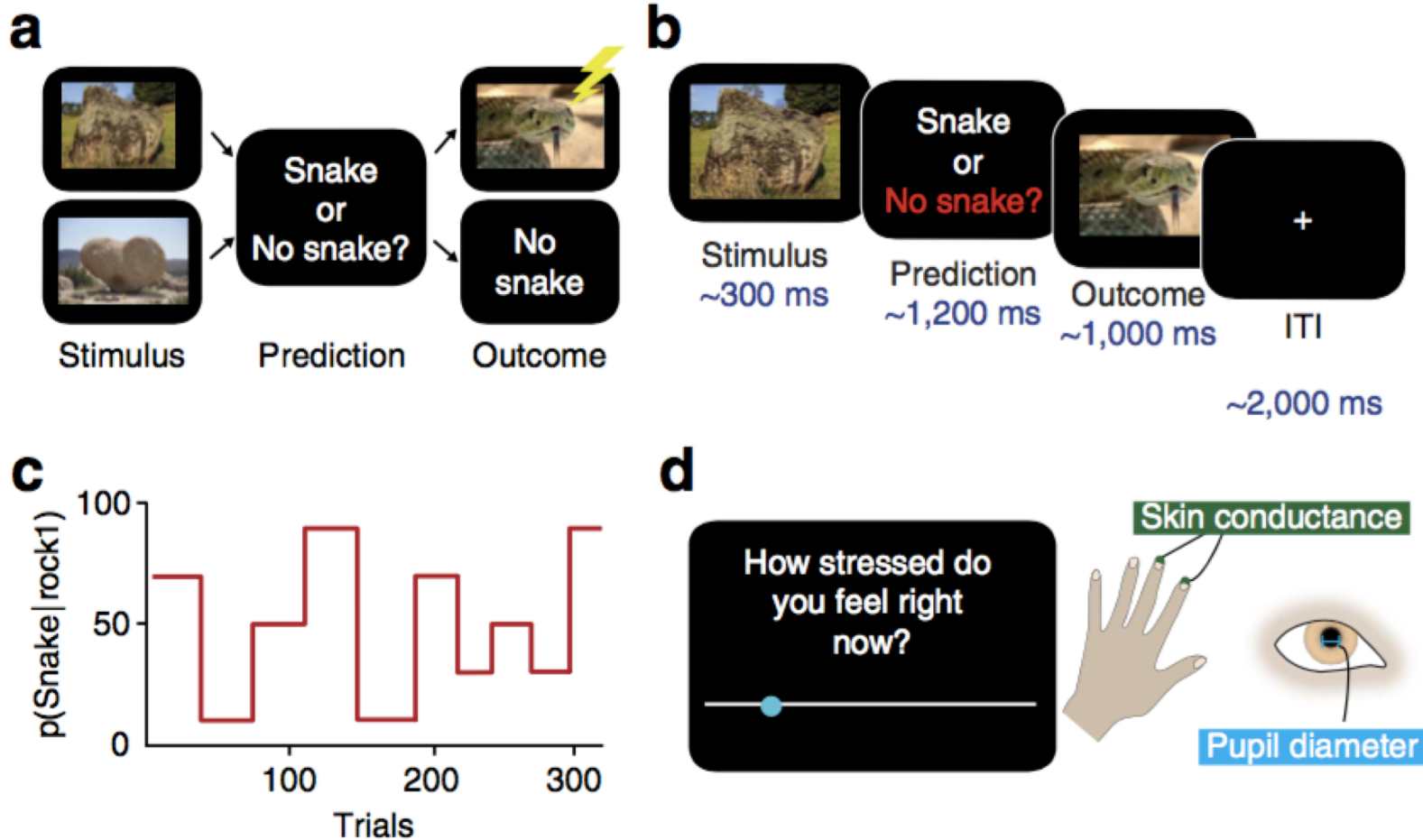


Context effects on the learning rate

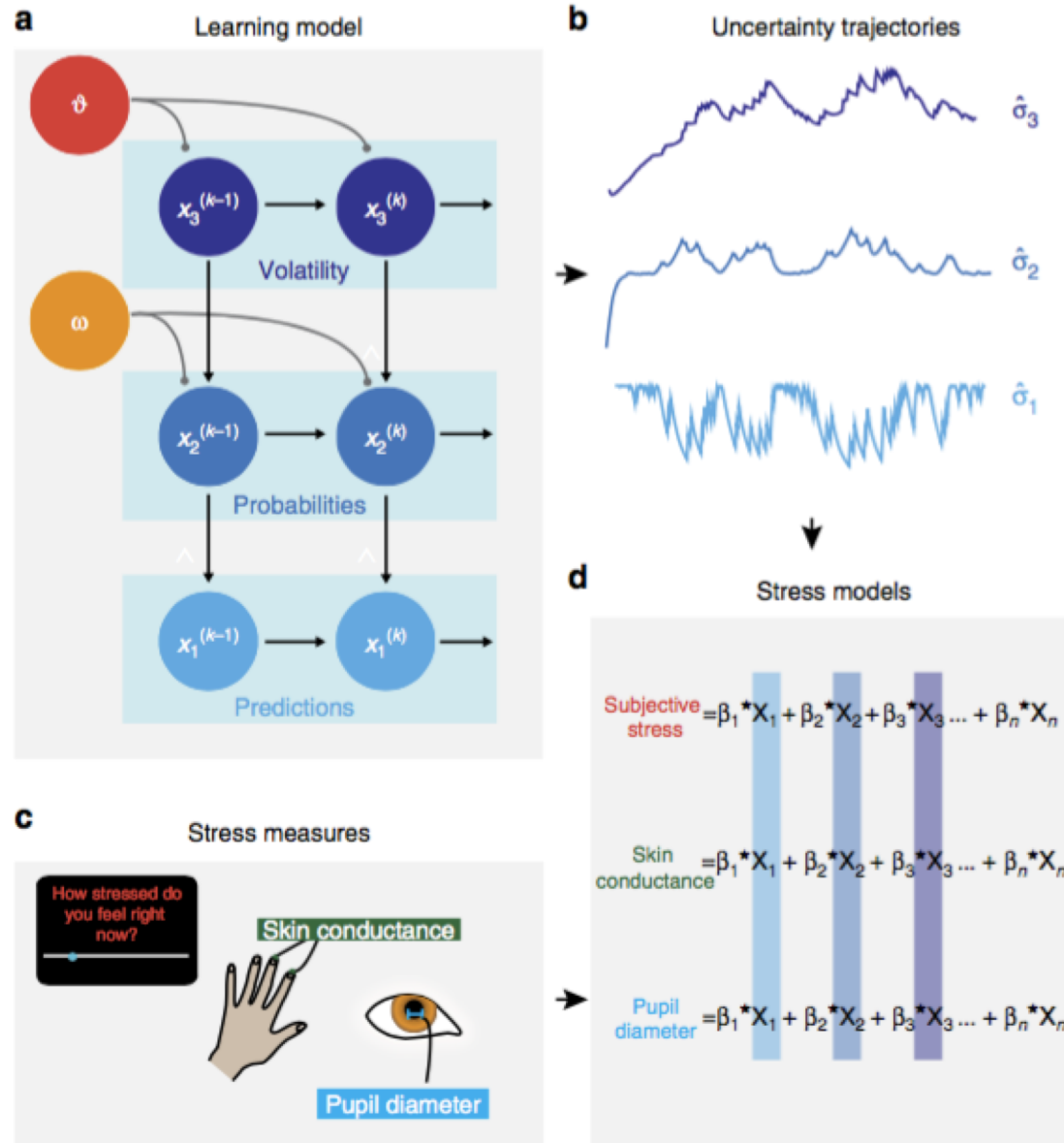
Simulation: $\mathcal{G} = 0.5$, $\omega = -2.2$, $\kappa = 1.4$



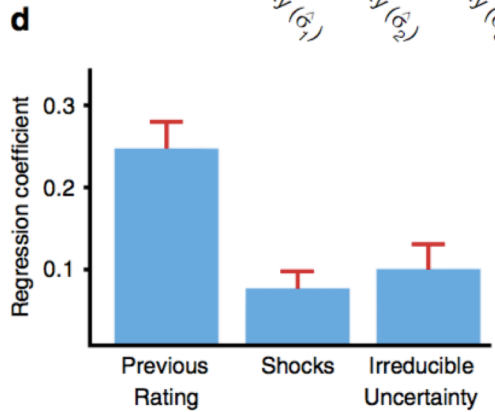
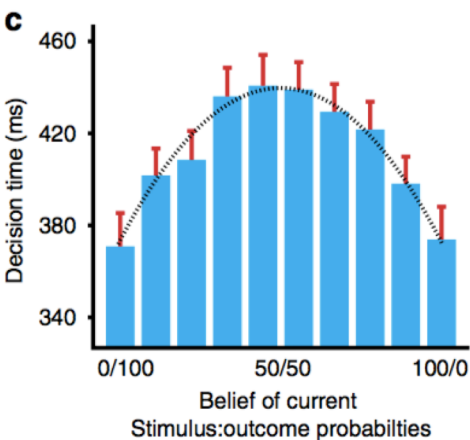
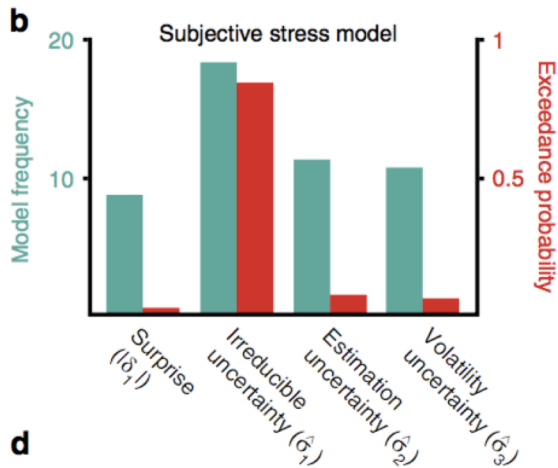
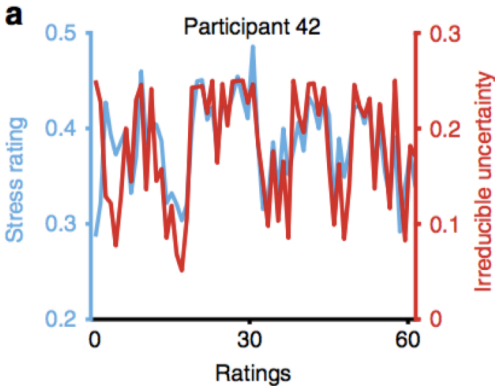
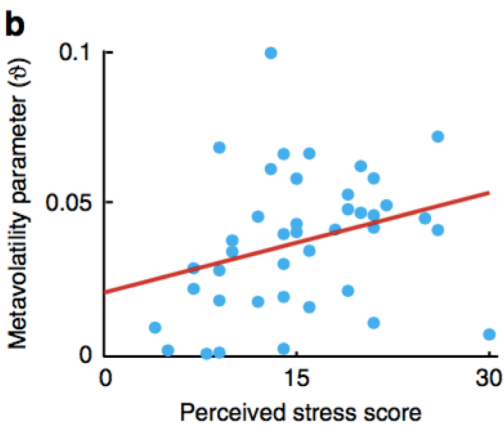
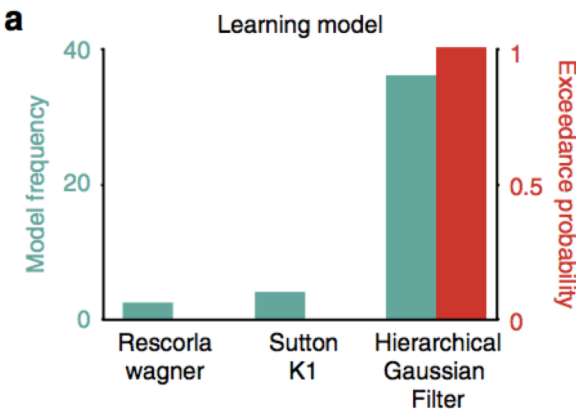
HGF: empirical evidence (De Berker et al, Nature Comms, 2016)



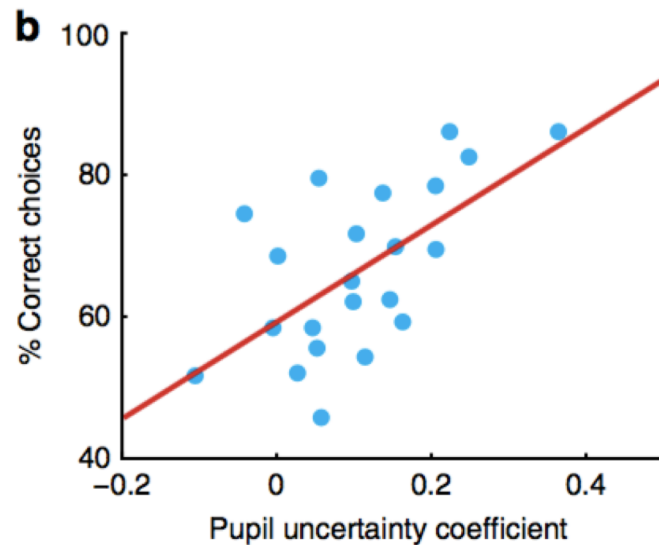
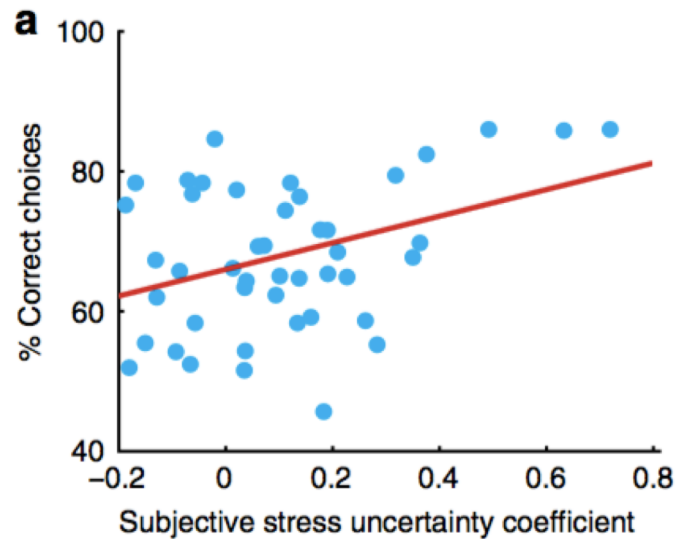
HGF: empirical evidence (De Berker et al, Nature Comms, 2016)



HGF: empirical evidence (De Berker et al, Nature Comms, 2016)

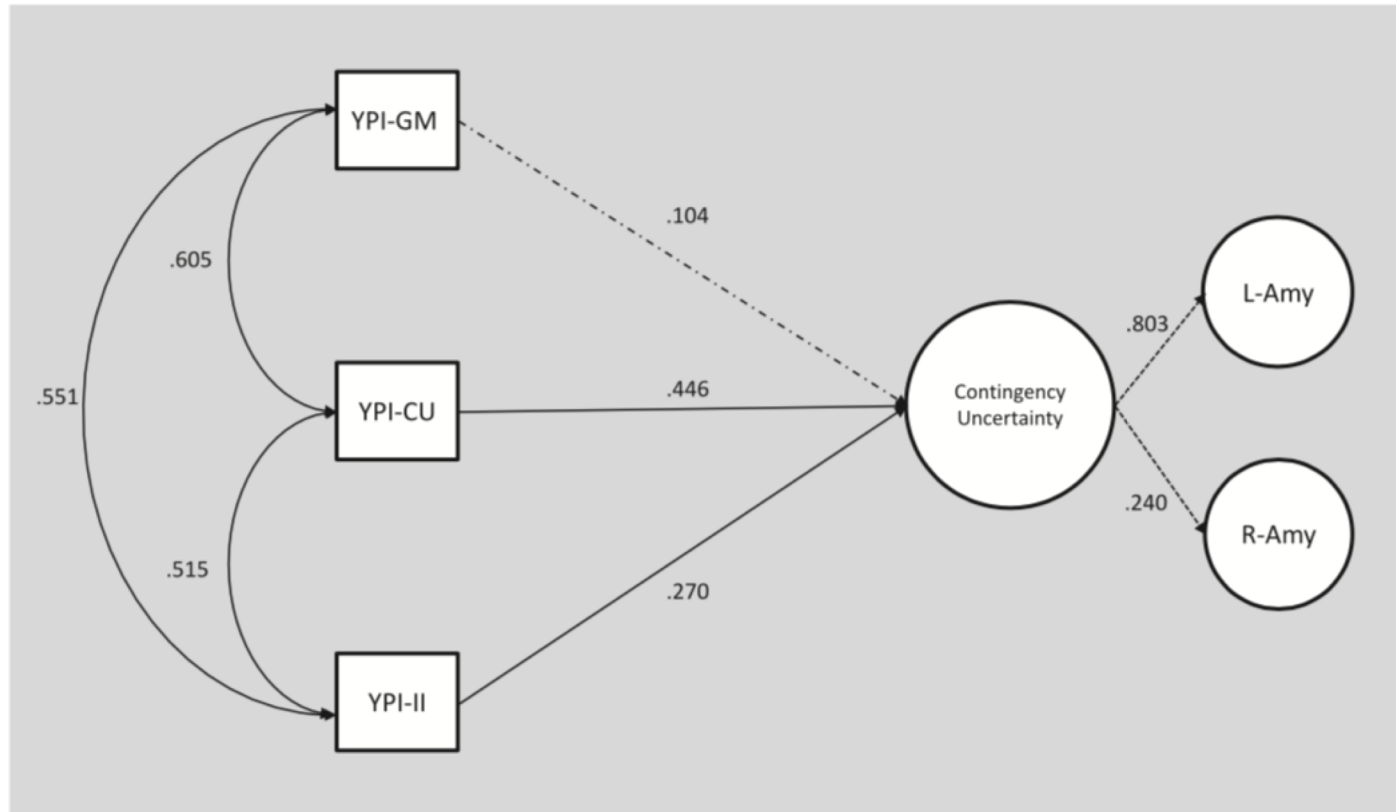


HGF: empirical evidence (De Berker et al, Nature Comms, 2016)



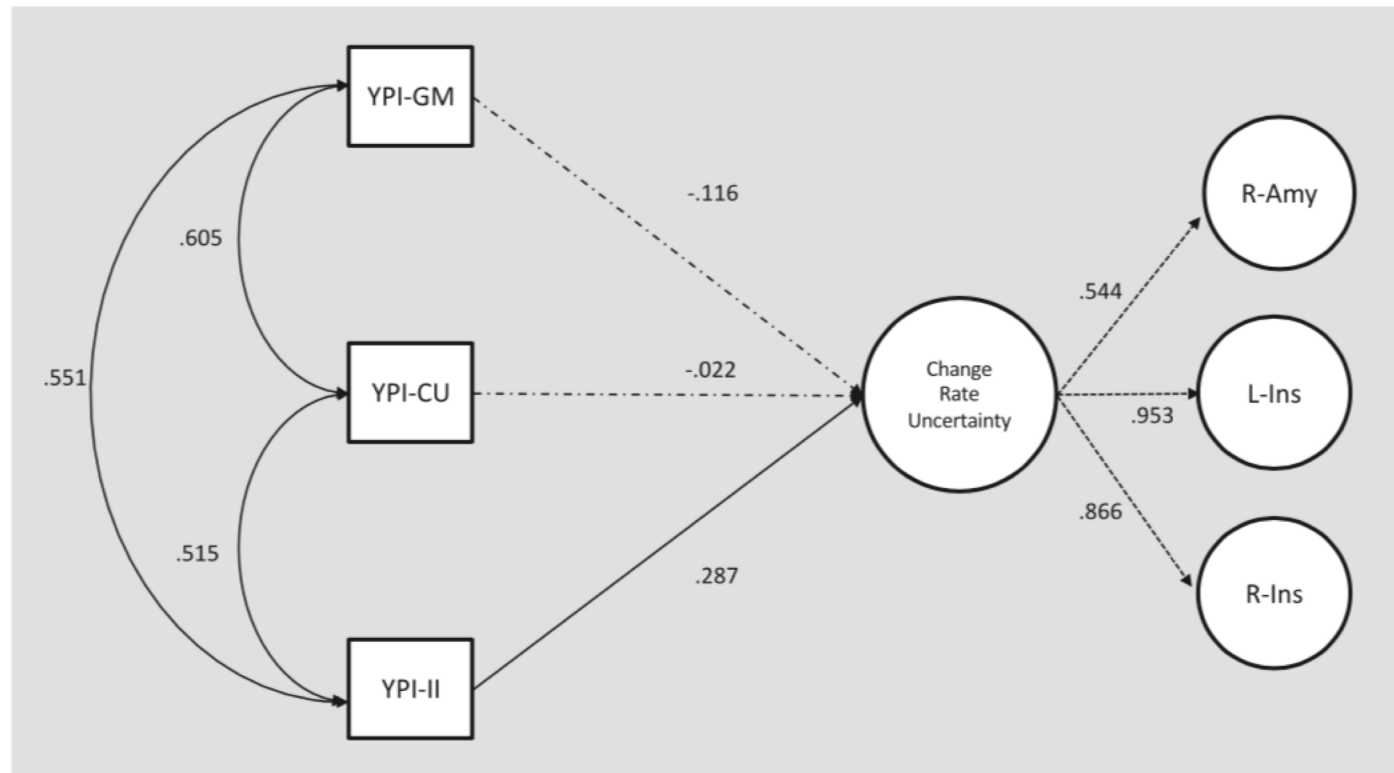
HGF: empirical evidence (Brazil et al., Biol. Psychiatry: CNI, 2017)

- **Threat conditioning paradigm:** neuronal activity time series (**fMRI**) from several regions of interest
- **Contingency uncertainty:** σ_1 from the first level of the continuous HGF
- YPI: Youth Psychopathic Traits inventory; GM - grandiose-manipulative, CU callous-unemotional, II impulsive irresponsible

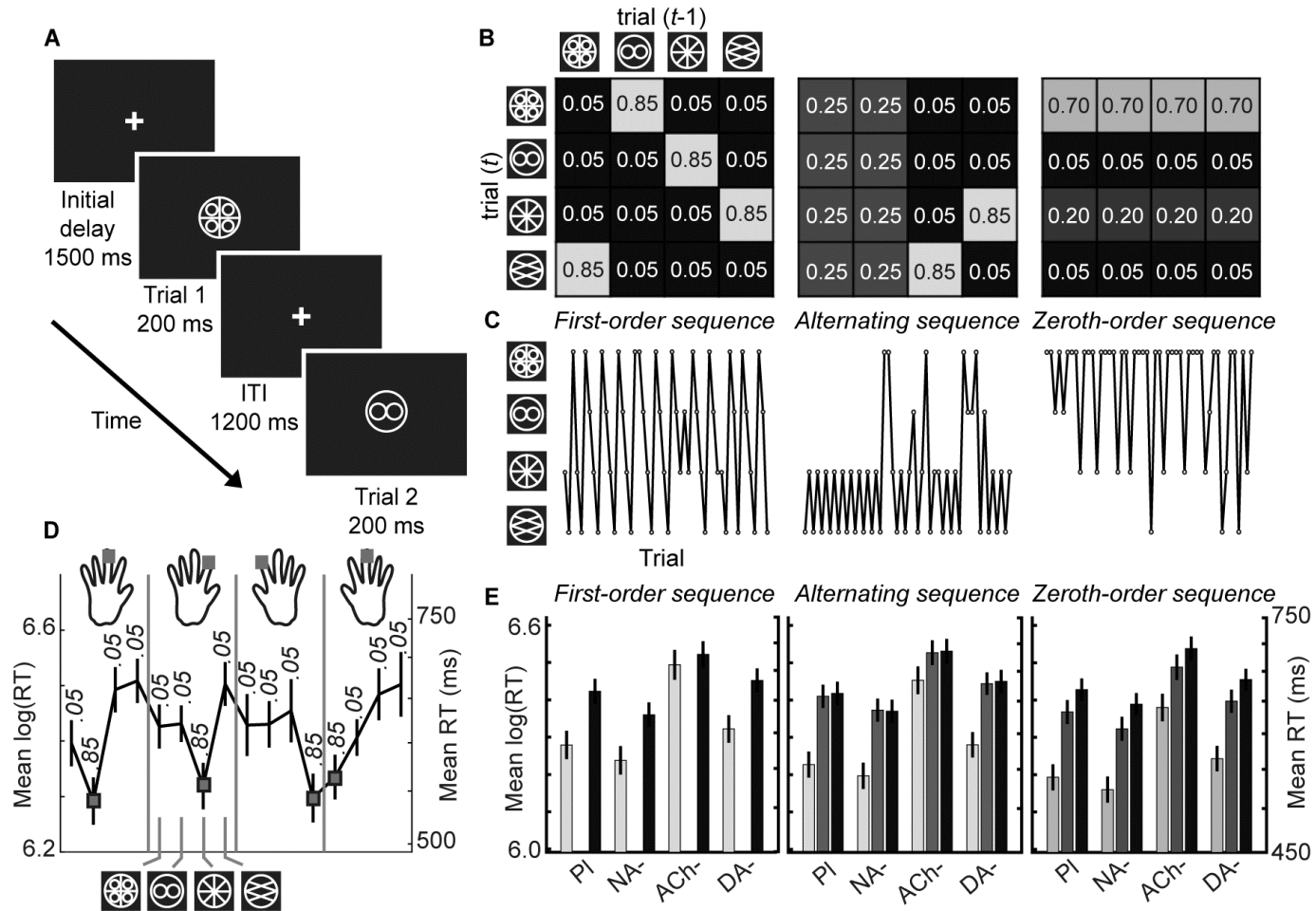


HGF: empirical evidence (Brazil et al., Biol. Psychiatry: CNNI, 2017)

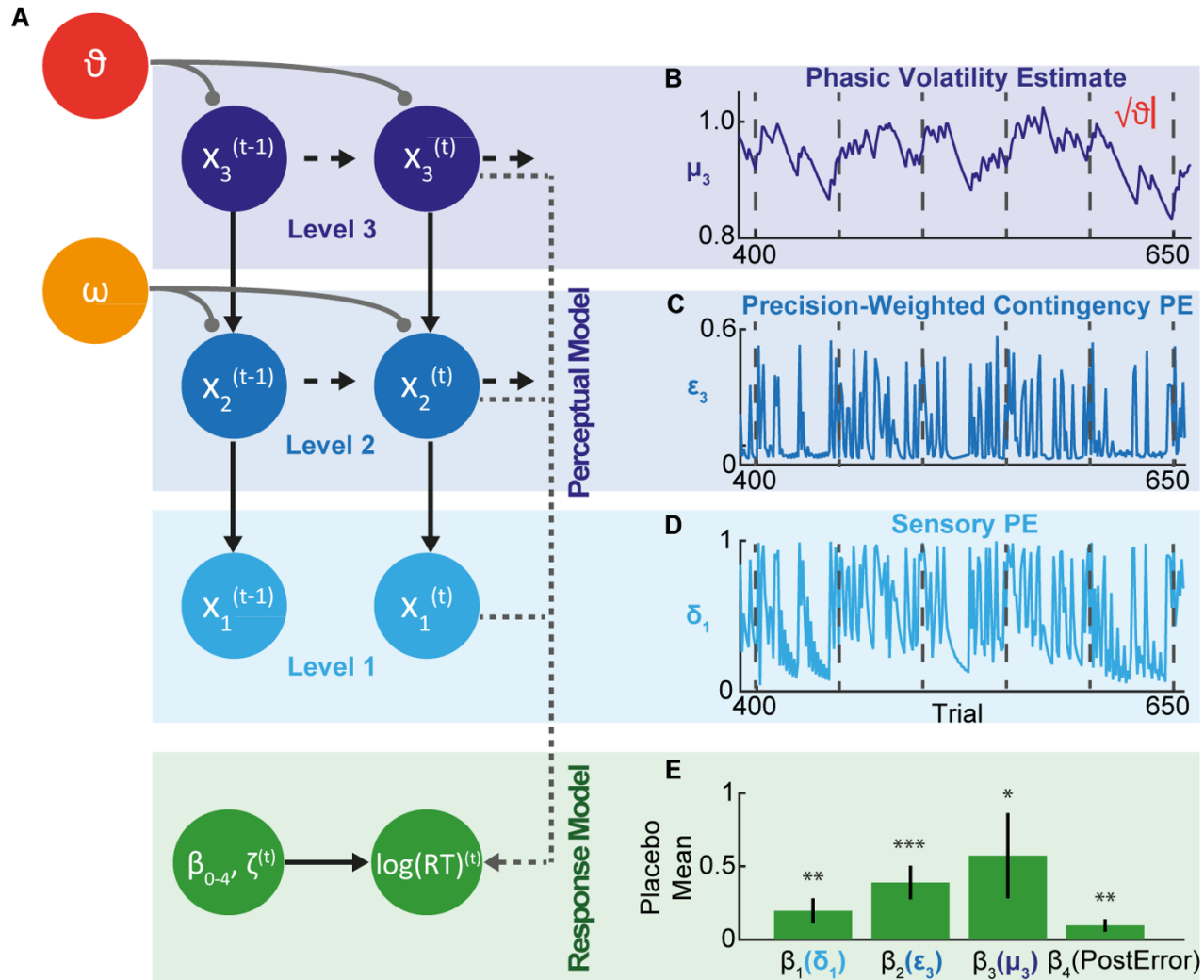
- **Threat conditioning paradigm:** neuronal activity time series (**fMRI**) from several regions of interest
- Change rate uncertainty: σ_2 from the second level of the continuous HGF
- YPI: Youth Psychopathic Traits inventory; GM - grandiose-manipulative, CU callous-unemotional, II impulsive irresponsible



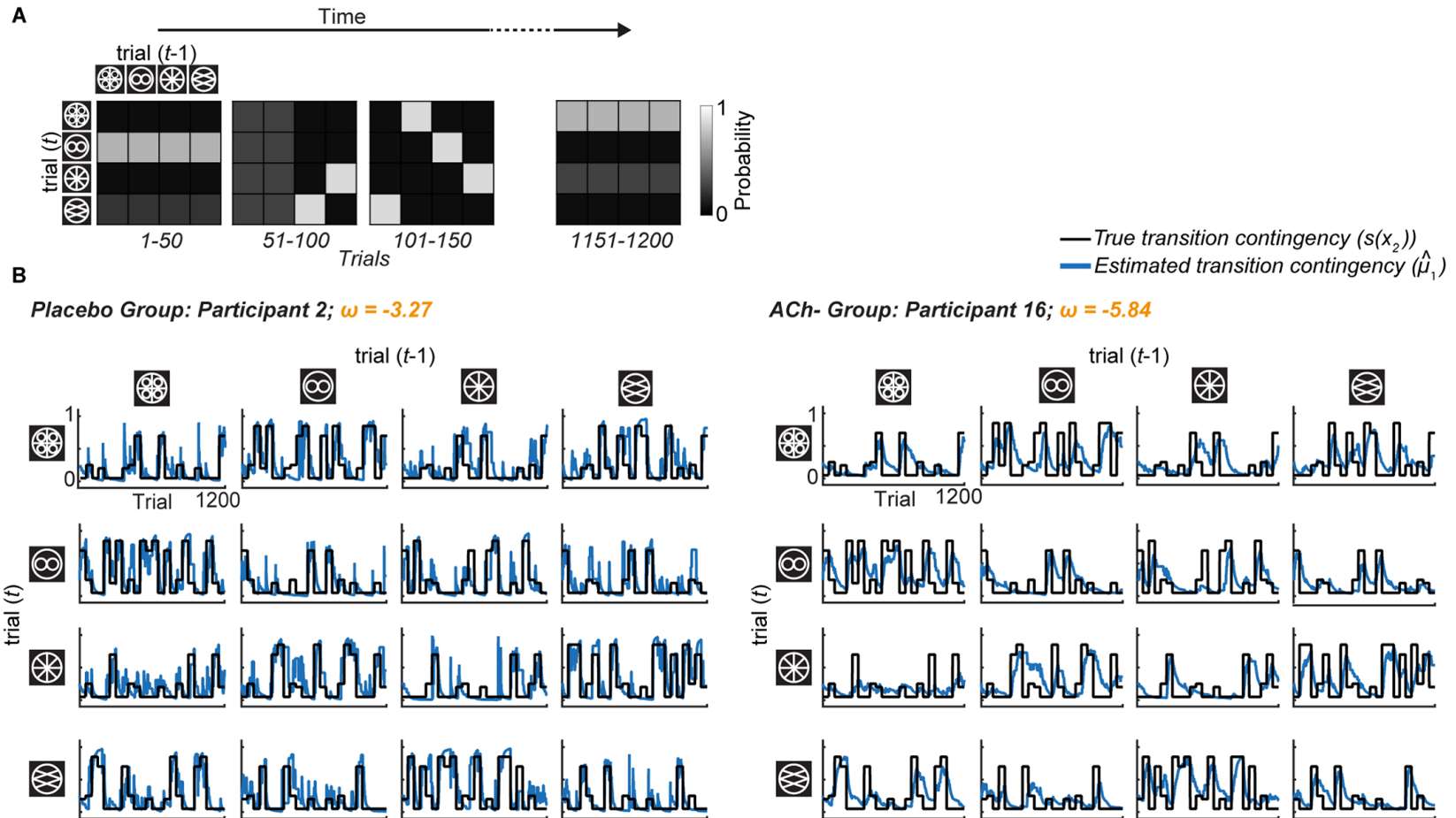
HGF: empirical evidence (Marshall & Mathys et al., PLOS Biol., 2016)



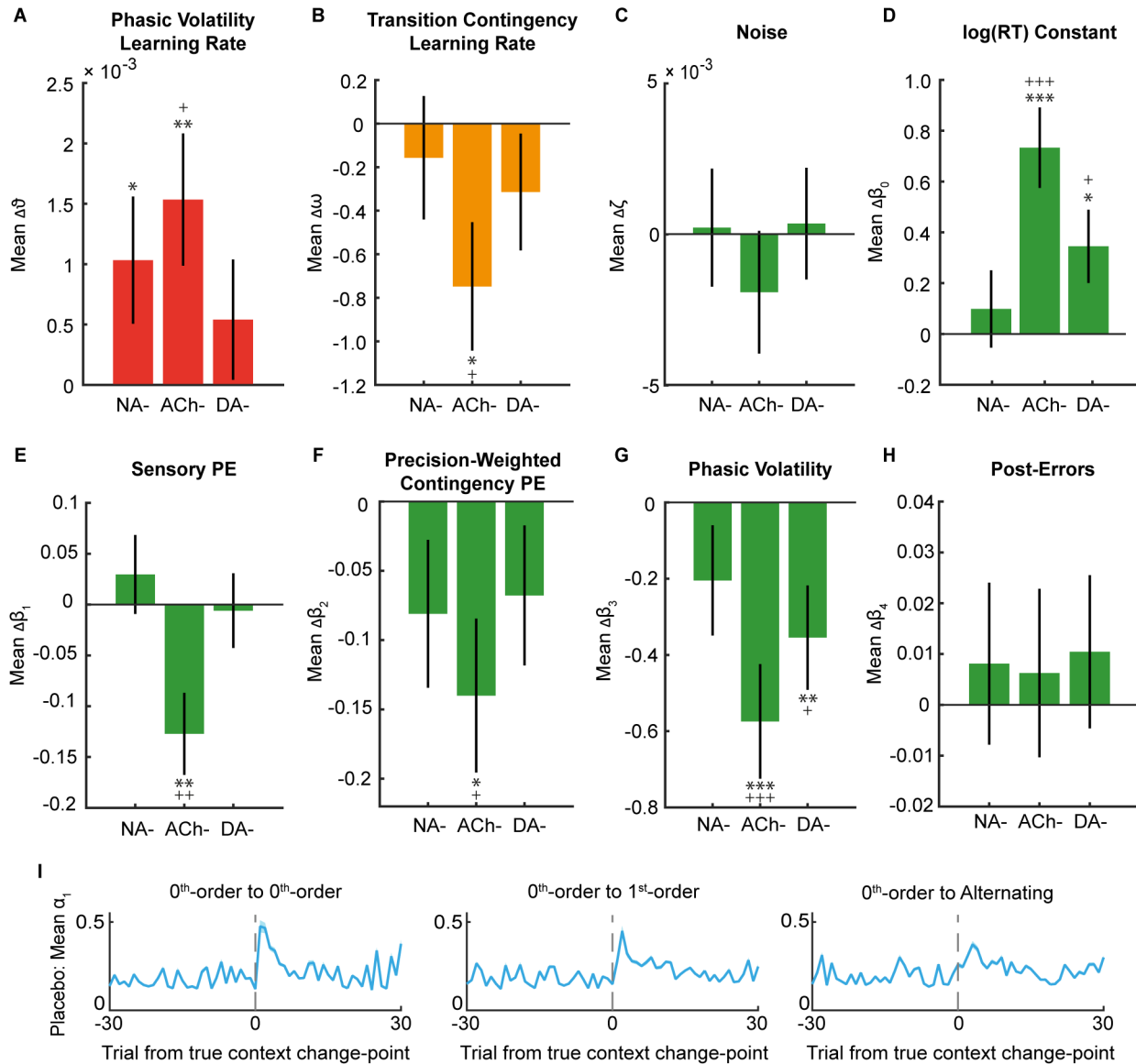
HGF: empirical evidence (Marshall & Mathys et al., PLOS Biol., 2016)



HGF: empirical evidence (Marshall & Mathys et al., PLOS Biol., 2016)



HGF: empirical evidence (Marshall & Mathys et al., PLOS Biol., 2016)



HGF: empirical evidence (Diaconescu et al., in preparation)

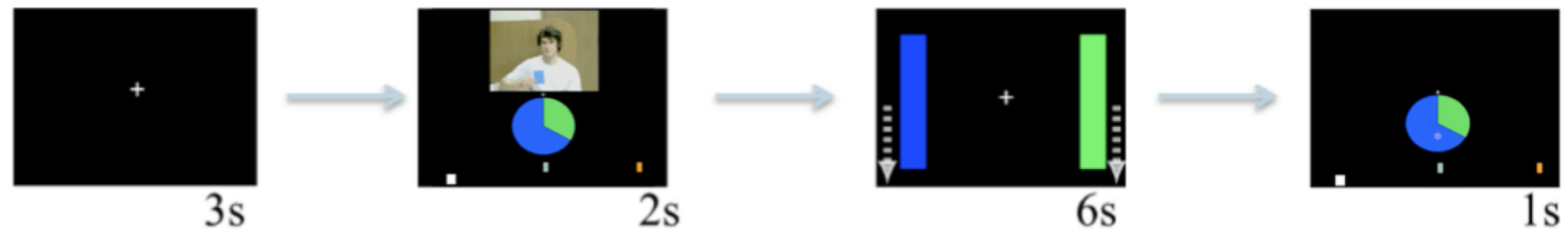
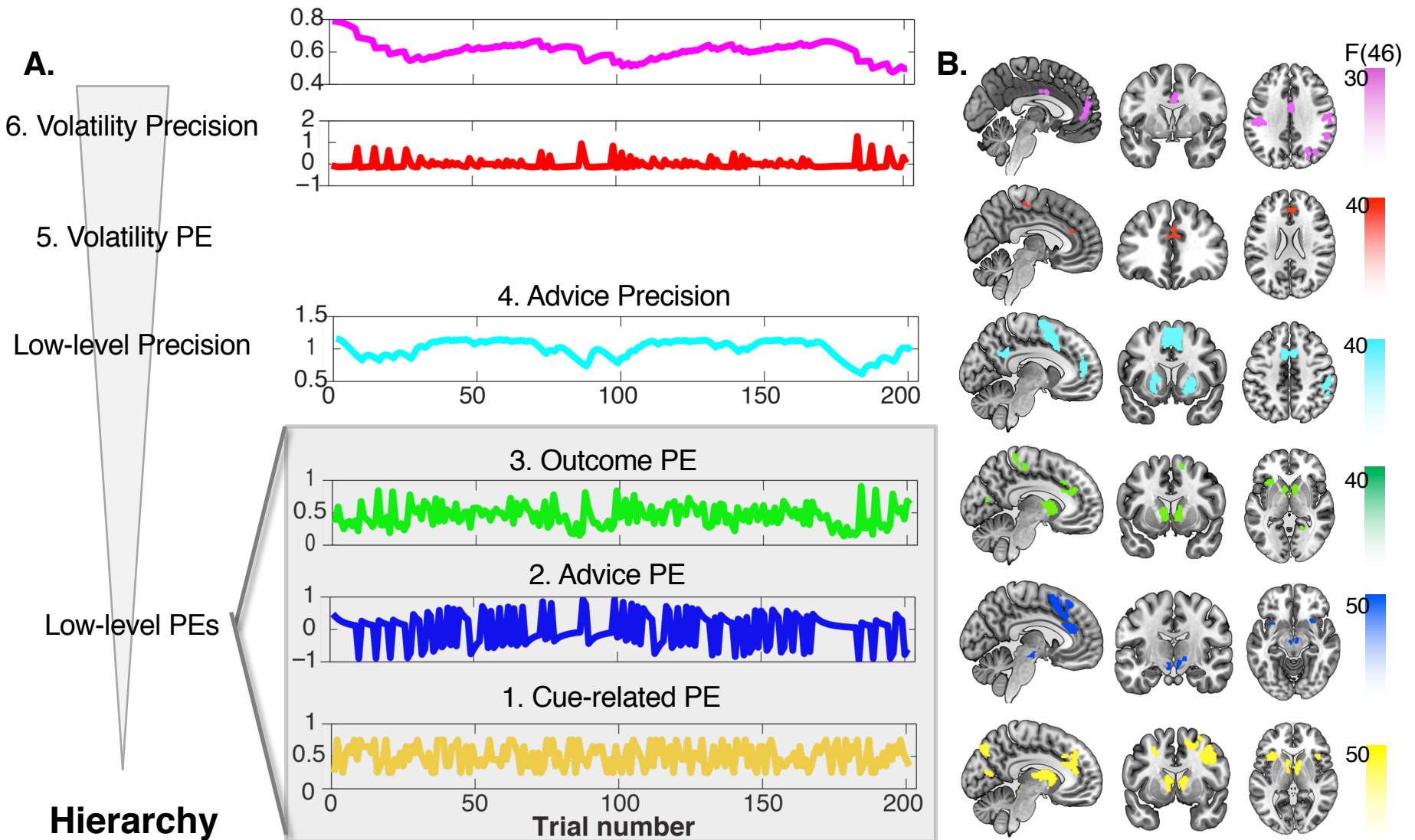
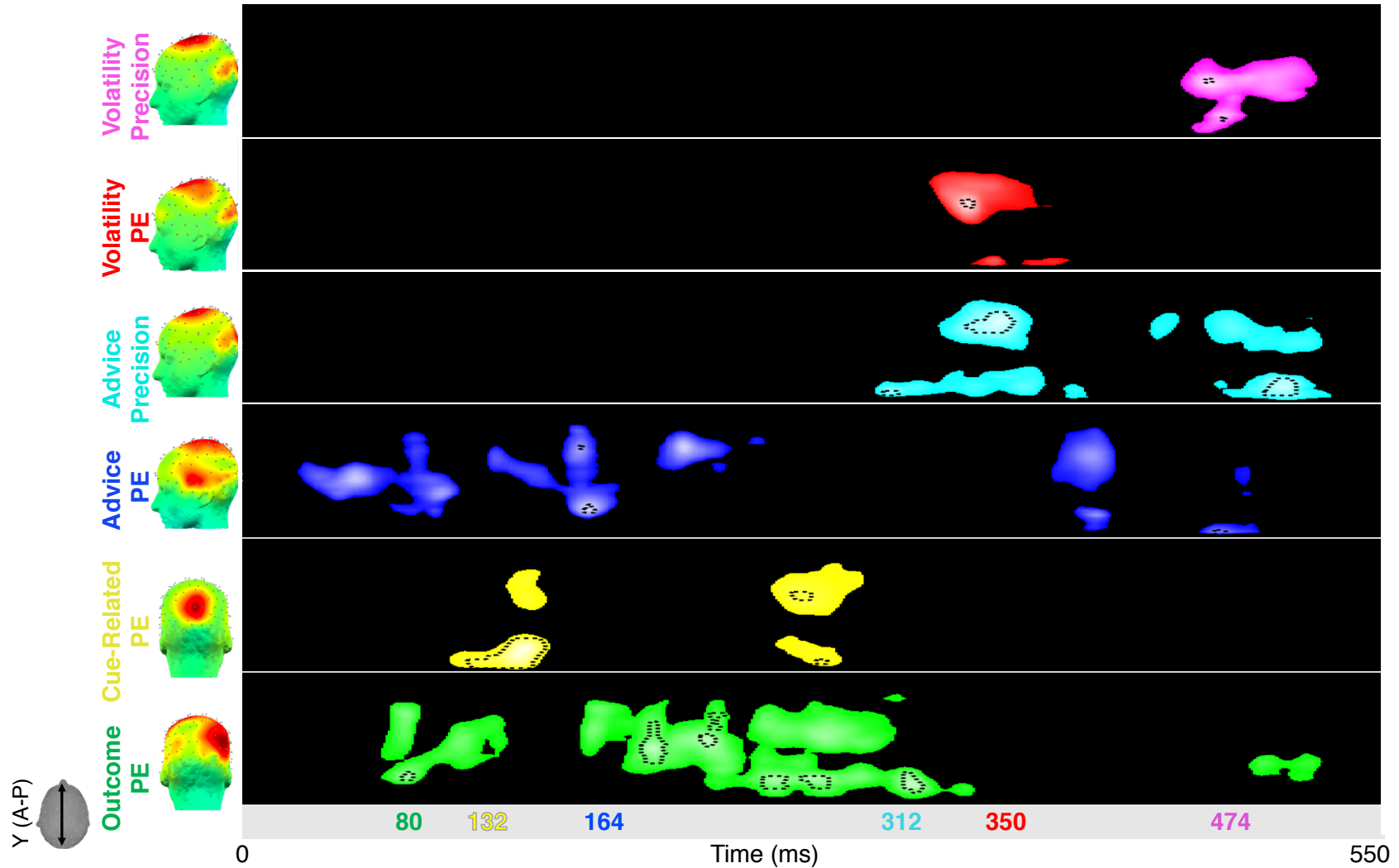


Fig. 1. Experimental Paradigm: 100 male volunteers played a binary lottery task and received advice about which option to choose from a more informed agent who was also incentivized to influence the participants' choices. To decide whether to take his advice into account, participants also inferred on the other's intentions and how they changed in time.

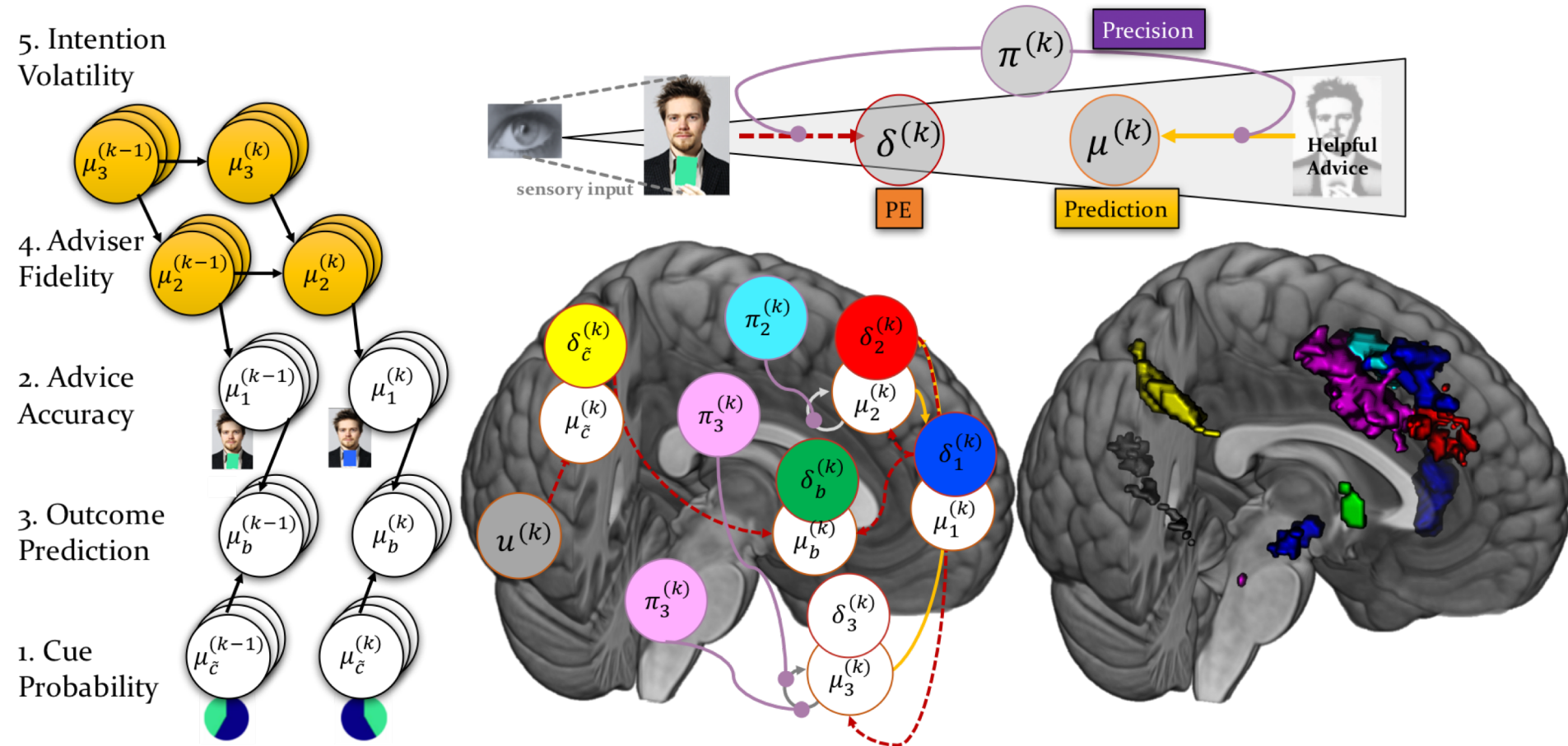
HGF: empirical evidence (Diaconescu et al., in preparation)



HGF: empirical evidence (Diaconescu et al., in preparation)

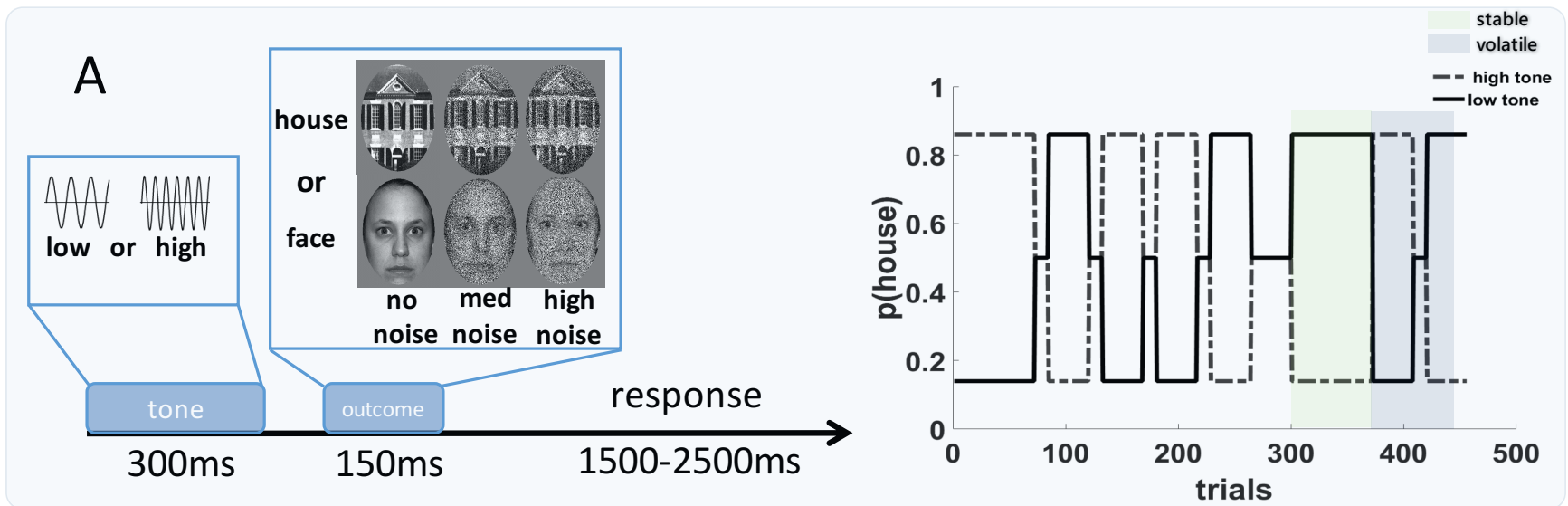


HGF: empirical evidence (Diaconescu et al., in preparation)

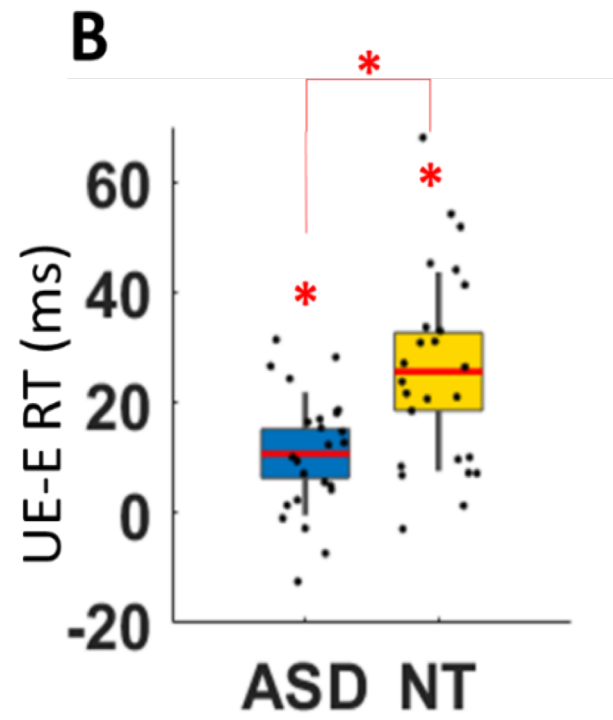
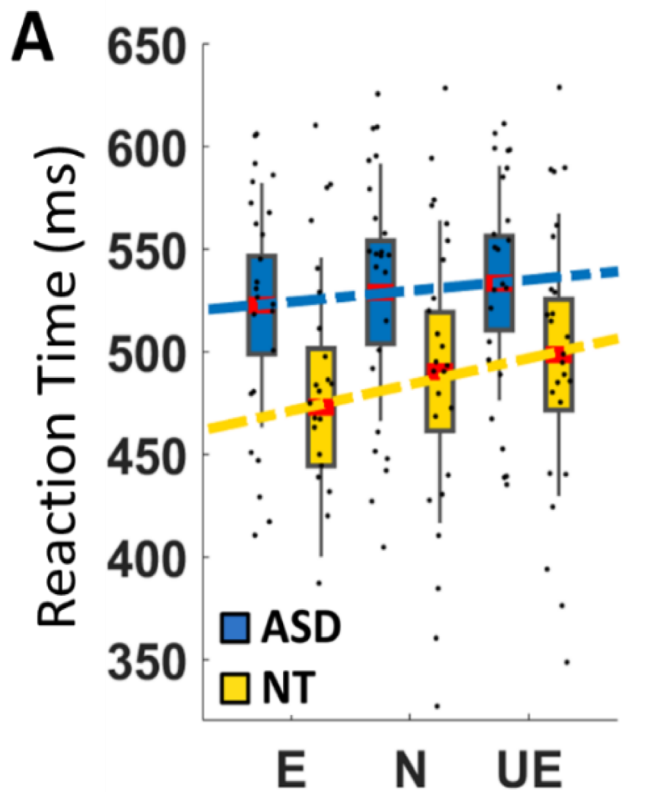


Association learning and autism

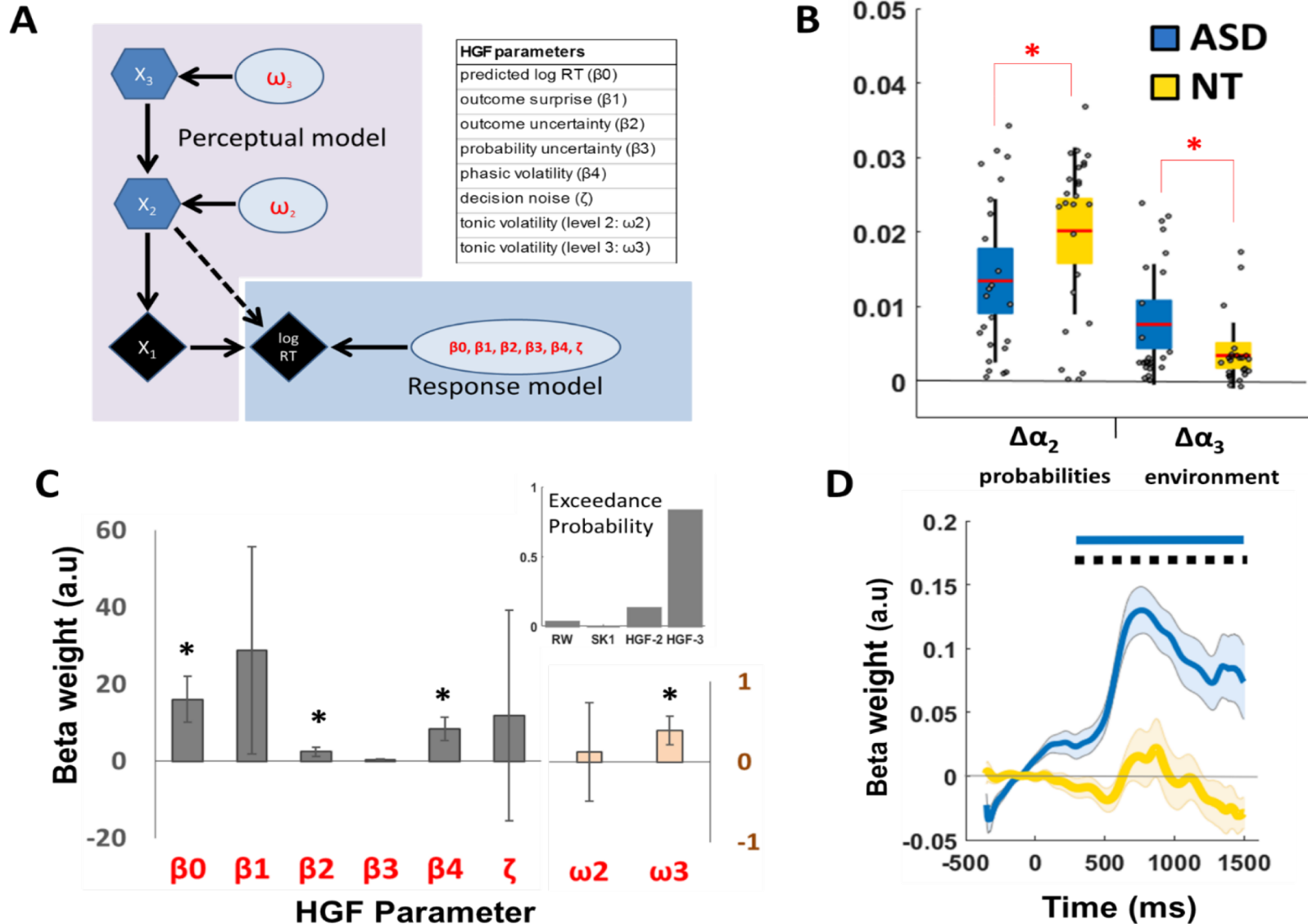
(Lawson, Mathys, & Rees, Nature Neuroscience, 2017)



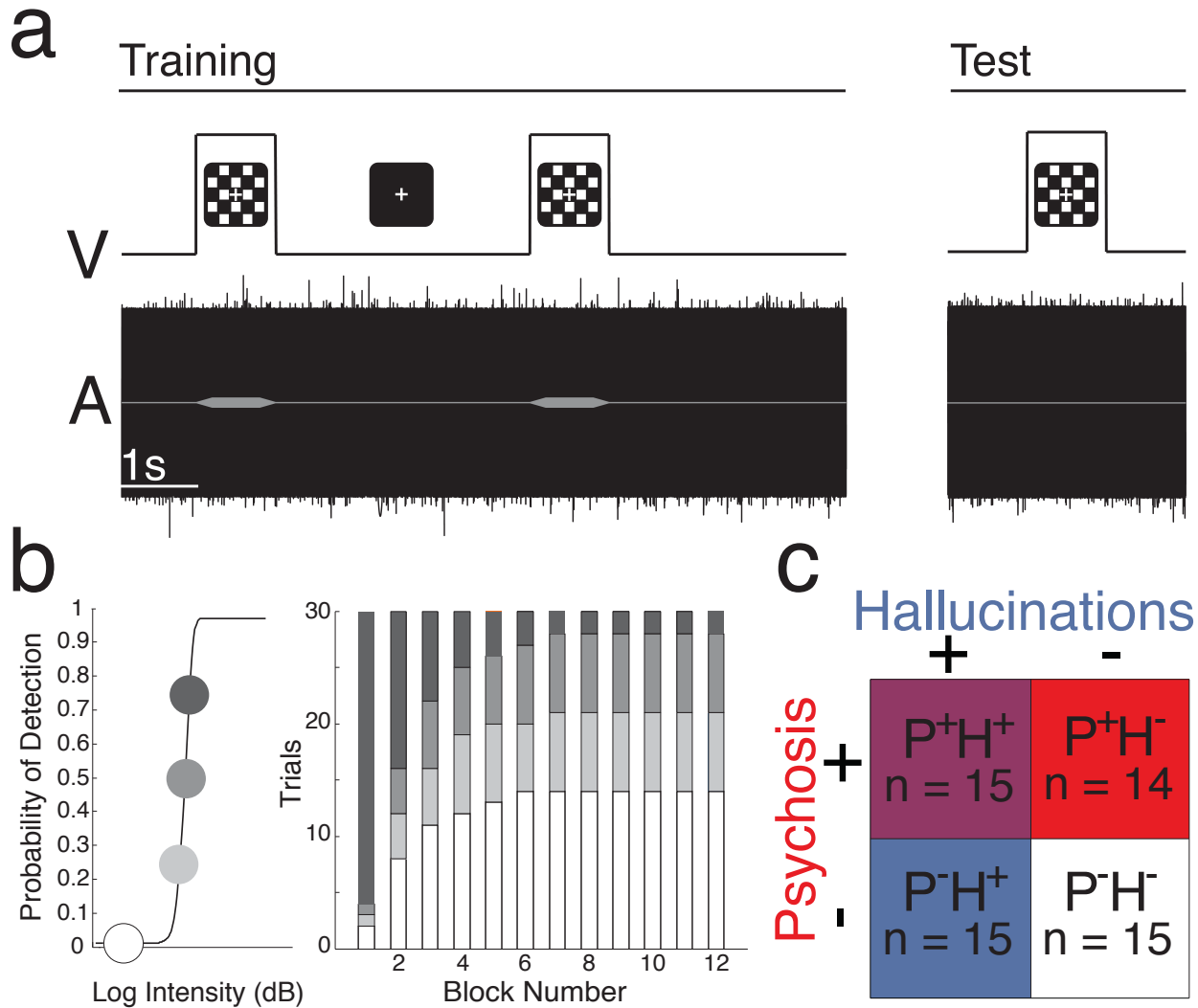
Association learning and autism



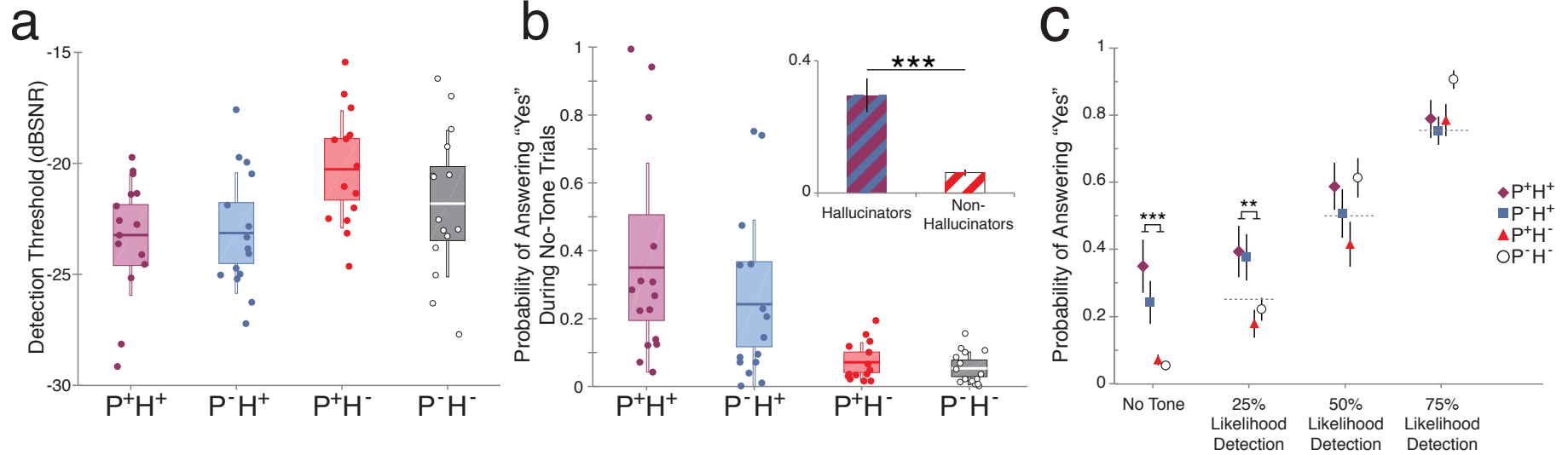
Association learning and autism



Conditioned hallucinations (Powers, Mathys, & Corlett, Science, 2017)



Conditioned hallucinations



Conditioned hallucinations

- Belief/percept formation model specifically created for this task
- Probability that the subject will respond “yes” to detection on a given trial:

$$P(\text{"yes"}|\textit{belief}) = \textit{sigmoid}(\textit{belief})$$

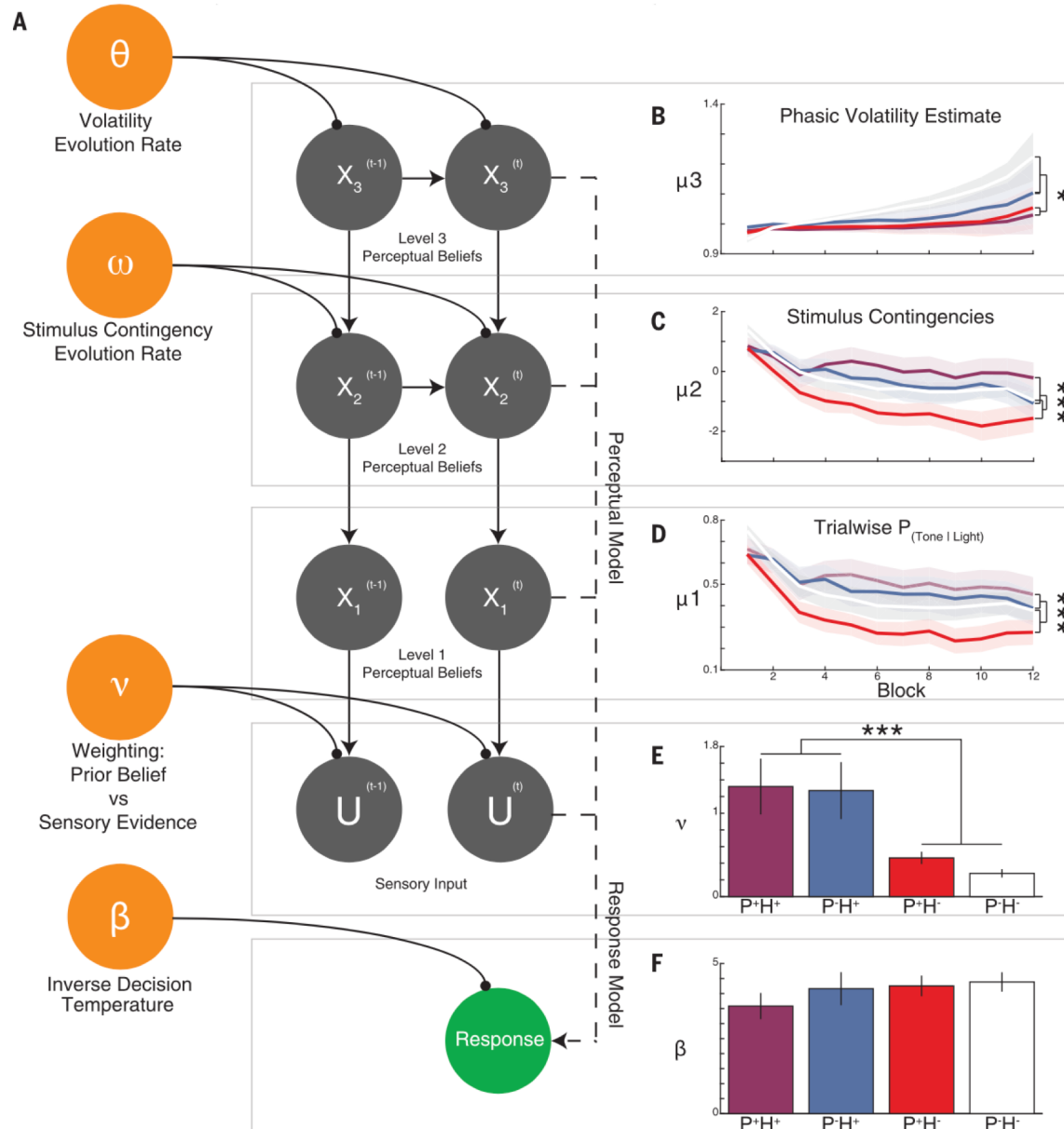
- *belief* is formalized as the Bayesian posterior mean of a beta distribution

$$\textit{belief} = \textit{prior} + \frac{1}{1 + \nu} (\textit{input} - \textit{prior})$$

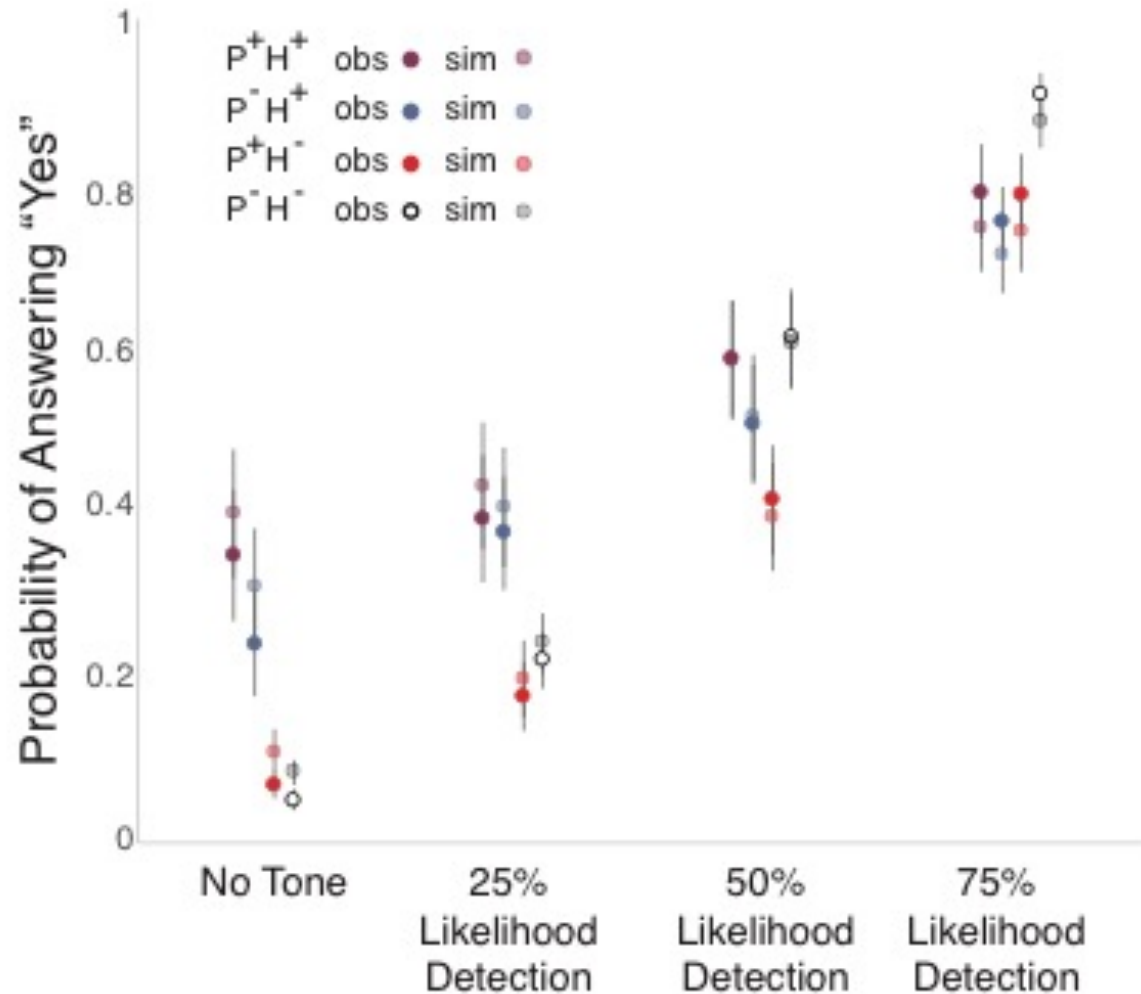
where

- *input* is given by experimental design: the true positive rate of the tone presented without light at each trial: 25%, 50%, or 75%
- *prior* is the prior from learning using the HGF: $\hat{\mu}_1$
- ν is a subject-specific parameter indicating the relative weight of the prior compared to the input.
- For $\nu = 1$, prior and input have equal weight; for $\nu > 1$ the prior has more weight than the input; and for $\nu < 1$ the input has more weight than the prior.

Conditioned hallucinations



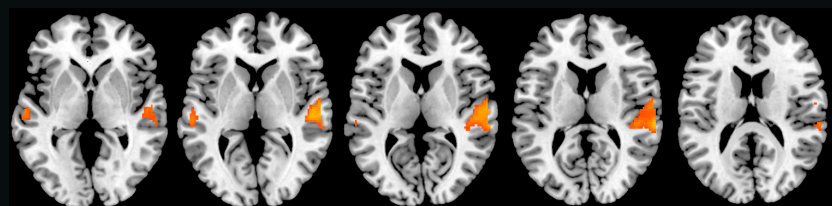
Conditioned hallucinations



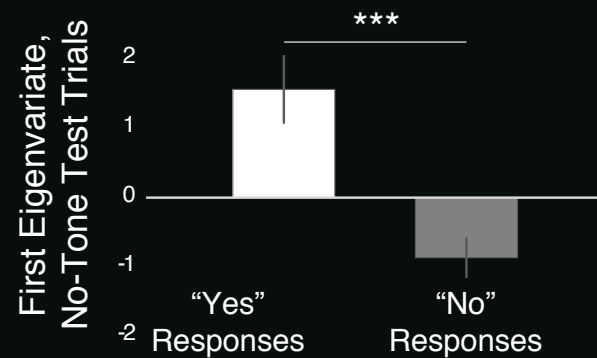
Conditioned hallucinations

a

Thresholding, Tone-Responsive Regions

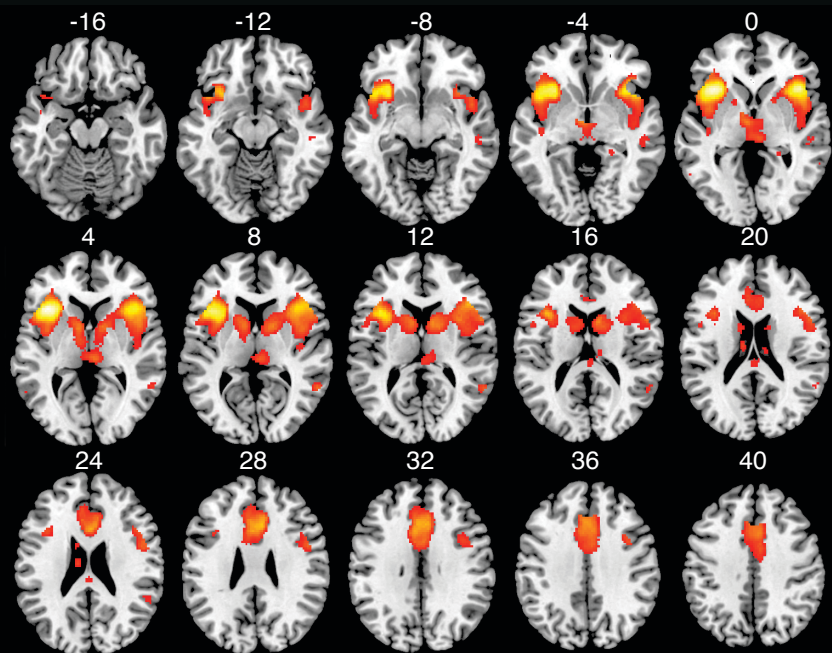


b



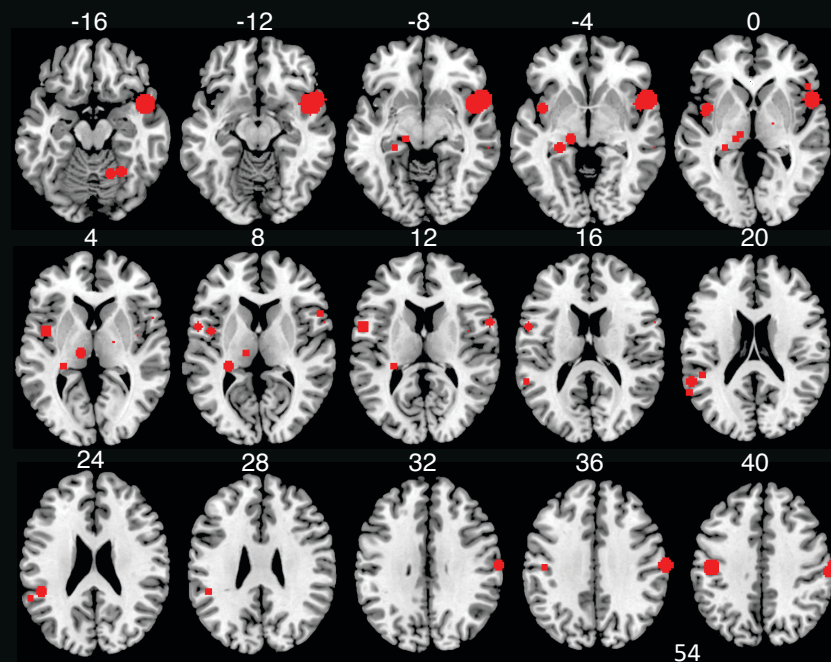
c

Test, "Yes" > "No" Responses, No-Tone Trials



d

Areas Active During AVH Symptom Capture



How to estimate and compare models: the HGF Toolbox

- Available at **<https://www.tnu.ethz.ch/tapas>**
- Start with README, manual, and interactive demo
- Modular, extensible
- Matlab-based

Thanks

Rick Adams

Archie de Berker

Sven Bestmann

Kay Brodersen

Jean Daunizeau

Andreea Diaconescu

Ray Dolan

Karl Friston

Sandra Iglesias

Lars Kasper

Rebecca Lawson

Ekaterina Lomakina

Berk Mirza

Read Montague

Tobias Nolte

Saeed Paliwal

Robb Rutledge

Klaas Enno Stephan

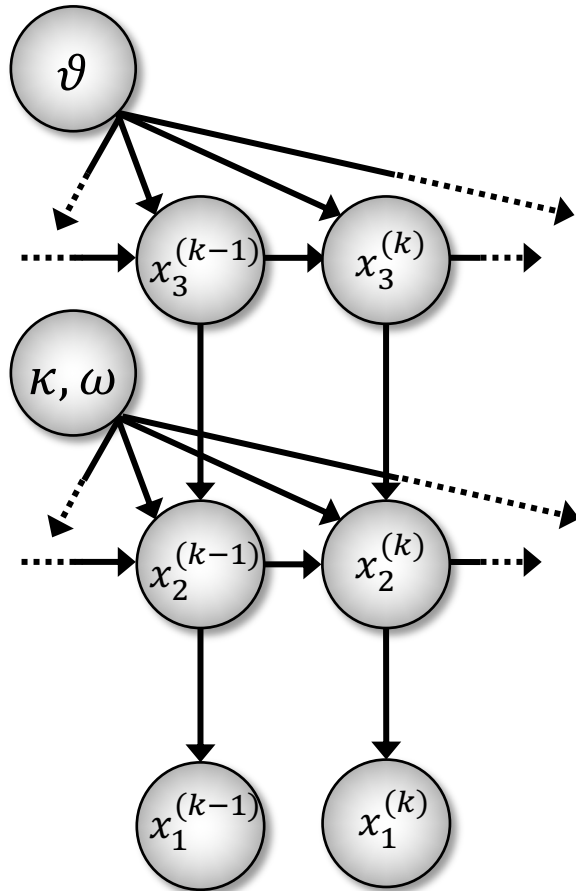
Philipp Schwartenbeck

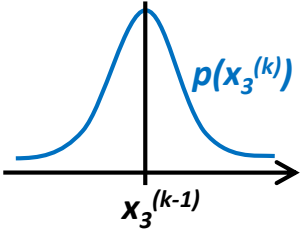
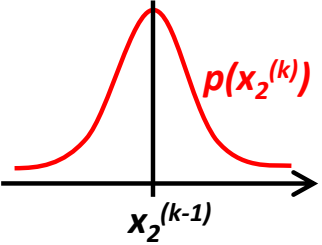
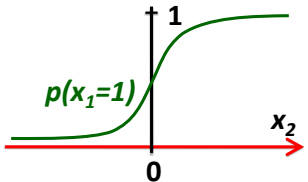
Simone Vossel

Lilian Weber



Application to binary data



State of the world	Model
Log-volatility $\mathbf{x}_3$ of tendency	$p(x_3^{(k)}) \sim N(x_3^{(k-1)}, \vartheta)$ Gaussian random walk with constant step size ϑ 
Tendency $\mathbf{x}_2$ towards category "1"	$p(x_2^{(k)}) \sim N(x_2^{(k-1)}, \exp(\kappa x_3 + \omega))$ Gaussian random walk with step size $\exp(\kappa x_3 + \omega)$ 
Stimulus category $\mathbf{x}_1$ ("0" or "1")	$p(x_1=1) = s(x_2)$ $p(x_1=0) = 1-s(x_2)$ Sigmoid transformation of x_2 

Update equation for binary observations

- $x_1 \in \{0,1\}$ is observed by the agent. Each observation leads to an update in the belief on $x_2, x_3, \dots$, and so on up the hierarchy.
- The updates for x_2 can be derived in the same manner as above.

$$I(x_2^{(k)}) = \ln s(x_2^{(k)}) + x_2^{(k)}(x_1^{(k)} - 1) - \frac{1}{2} \hat{\pi}_2^{(k)} (x_2^{(k)} - \mu_2^{(k-1)})^2$$

$$\mu_2^{(k)} = \mu_2^{(k-1)} + \sigma_2^{(k)} \delta_1^{(k)}$$

- At first, this simply looks like an uncertainty-weighted update. However, when we unpack σ_2 and do a Taylor expansion in powers of $\hat{\pi}_1$, we see that it is again proportional to the precision of the prediction on the level below:

$$\sigma_2^{(k)} = \frac{\hat{\pi}_1^{(k)}}{\hat{\pi}_2^{(k)} \hat{\pi}_1^{(k)} + 1} = \hat{\pi}_1^{(k)} - \hat{\pi}_2^{(k)} (\hat{\pi}_1^{(k)})^2 + (\hat{\pi}_2^{(k)})^2 (\hat{\pi}_1^{(k)})^3 + O(4)$$

- At all higher levels, the updates are as previously derived.

VAPes and VOPEs

The updates of the belief on x_1 are driven by value prediction errors (VAPes)

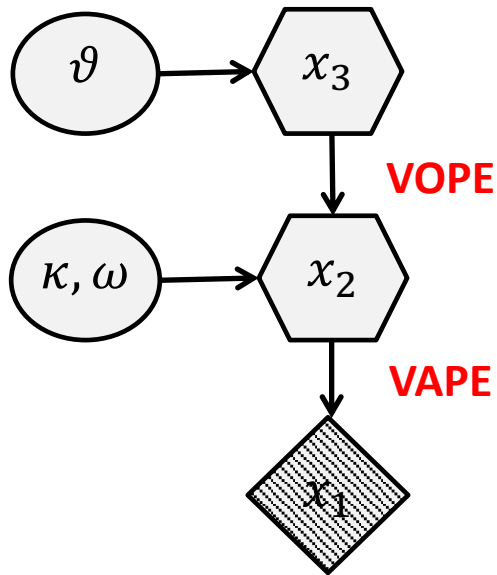
$$\mu_1^{(k)} = \mu_1^{(k-1)} + \frac{\hat{\pi}_u}{\pi_1^{(k)}} \left(u^{(k)} - \mu_1^{(k-1)} \right), \text{ VAPE}$$

while the x_2 -updates are driven by volatility prediction errors (VOPEs)

$$\mu_2^{(k)} = \mu_2^{(k-1)} + \frac{1}{2} \kappa_1 v_1^{(k)} \frac{\hat{\pi}_1^{(k)}}{\pi_2^{(k)}} \delta_1^{(k)} \text{ VOPE}$$

$$\delta_1^{(k)} \stackrel{\text{def}}{=} \frac{\sigma_1^{(k)} + \left(\mu_1^{(k)} - \mu_1^{(k-1)} \right)^2}{\sigma_1^{(k-1)} + \exp \left(\kappa_1 \mu_2^{(k-1)} + \omega_1 \right)} - 1$$

3-level HGF for binary observations



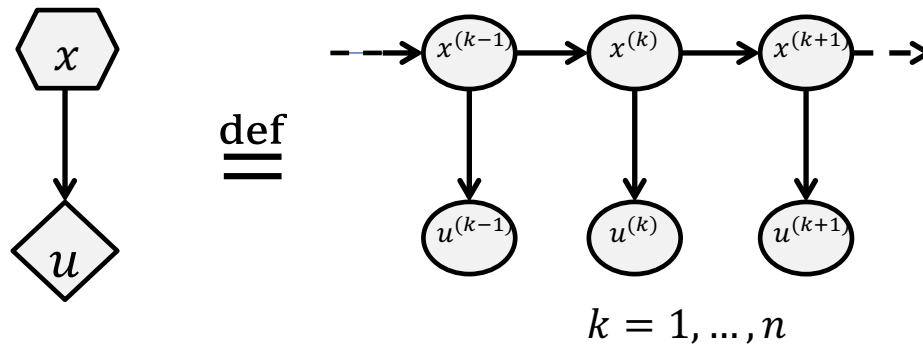
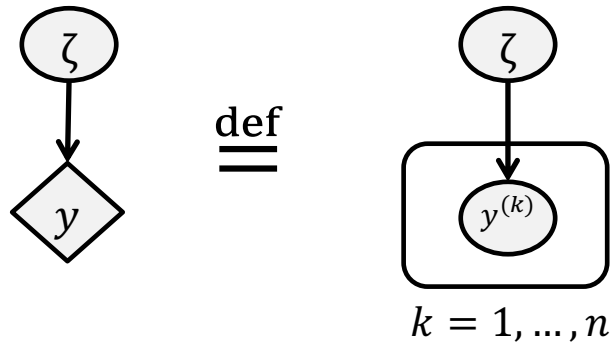
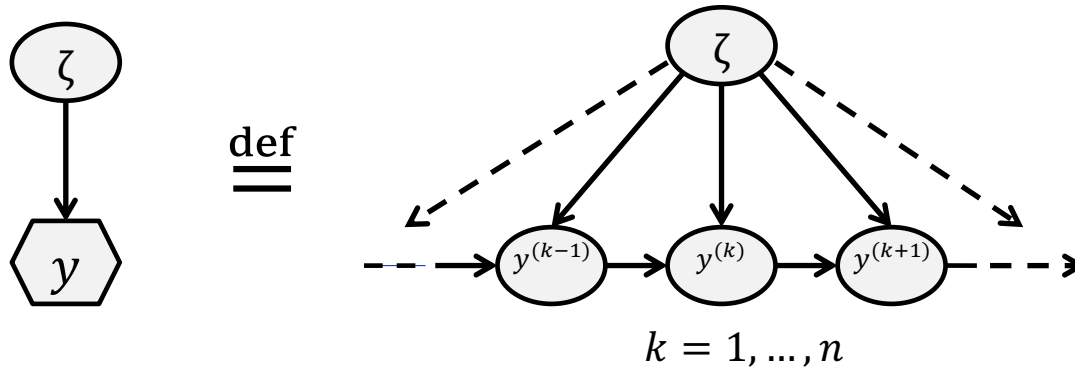
$$x_3^{(k)} \sim \mathcal{N} \left(x_3^{(k-1)}, \vartheta \right)$$

$$x_2^{(k)} \sim \mathcal{N} \left(x_2^{(k-1)}, \exp \left(\kappa x_3^{(k)} + \omega \right) \right)$$

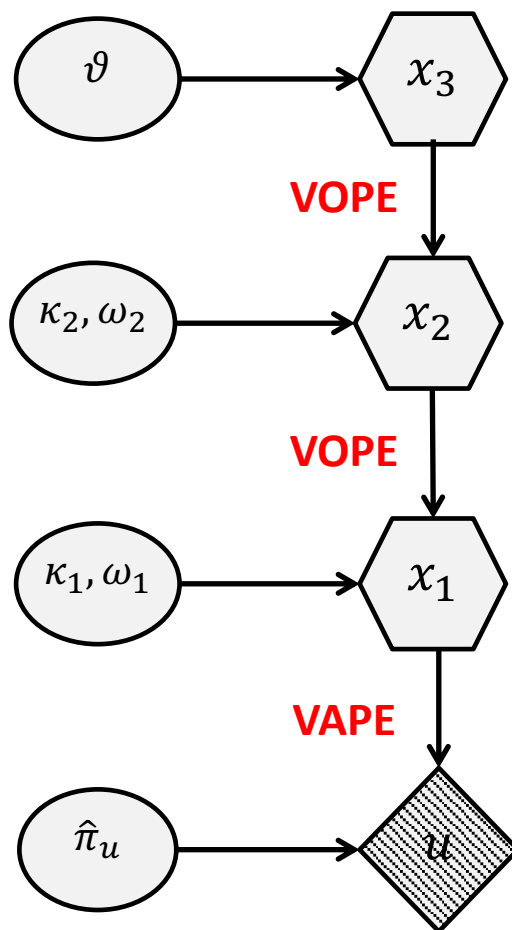
$$x_1^{(k)} \sim \text{Bern} \left(s \left(x_2^{(k)} \right) \right)$$

Mathys et al., 2011; Iglesias et al., 2013; Vossel et al., 2014a; Hauser et al., 2014; Diaconescu et al., 2014; Vossel et al., 2014b; ...

Notation



3-level HGF for continuous observations



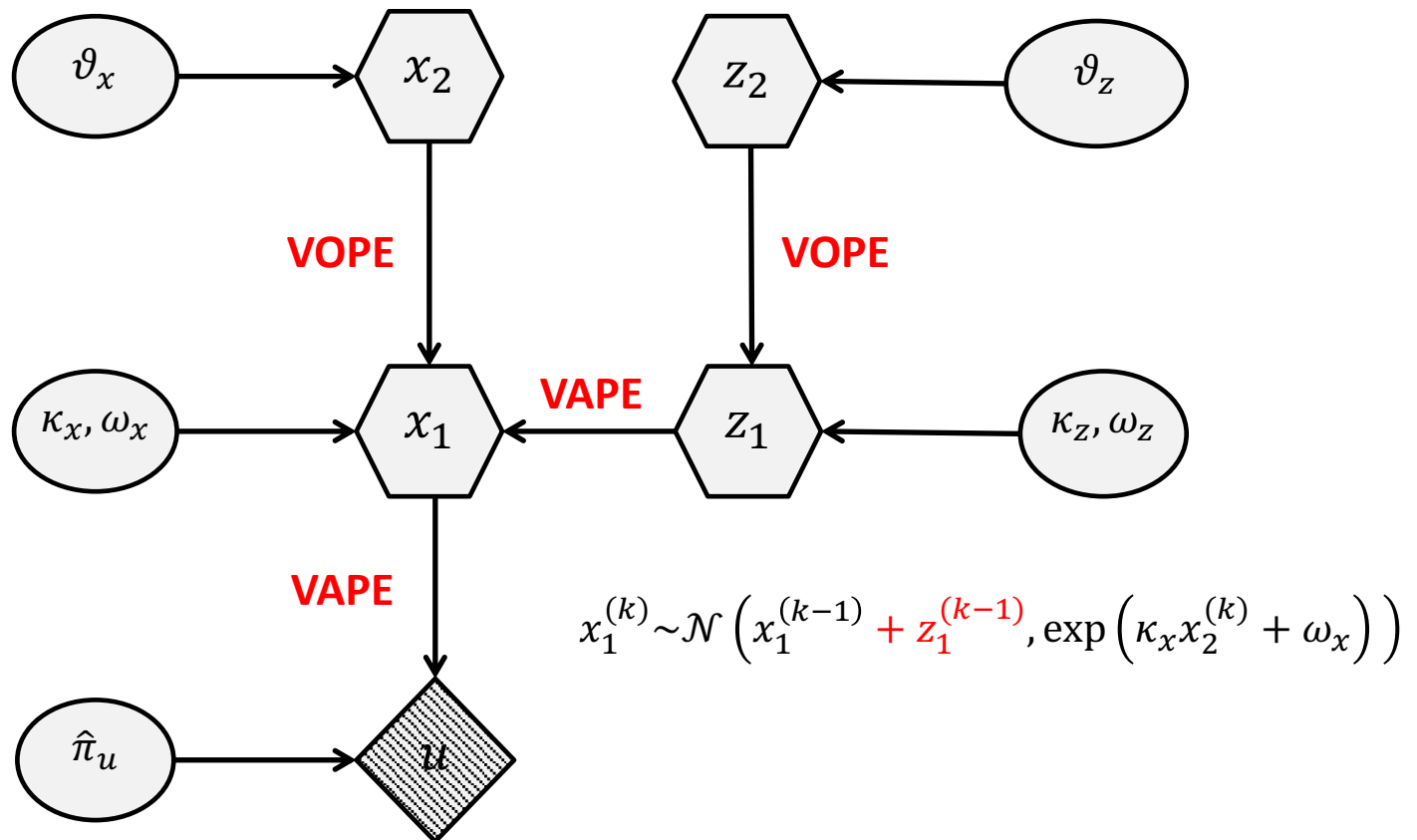
$$x_3^{(k)} \sim \mathcal{N} \left(x_3^{(k-1)}, \vartheta \right)$$

$$x_2^{(k)} \sim \mathcal{N} \left(x_2^{(k-1)}, \exp \left(\kappa_2 x_3^{(k)} + \omega_2 \right) \right)$$

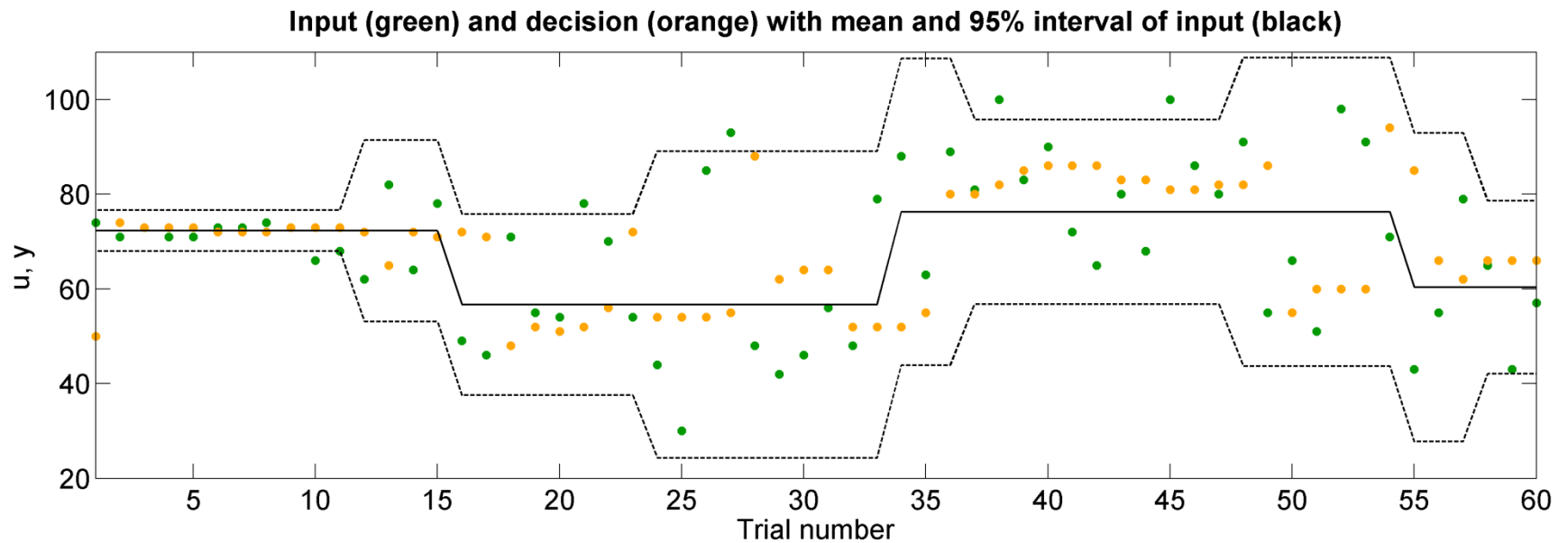
$$x_1^{(k)} \sim \mathcal{N} \left(x_1^{(k-1)}, \exp \left(\kappa_1 x_2^{(k)} + \omega_1 \right) \right)$$

$$u^{(k)} \sim \mathcal{N} \left(x_1^{(k)}, \hat{\pi}_u^{-1} \right)$$

Variable drift

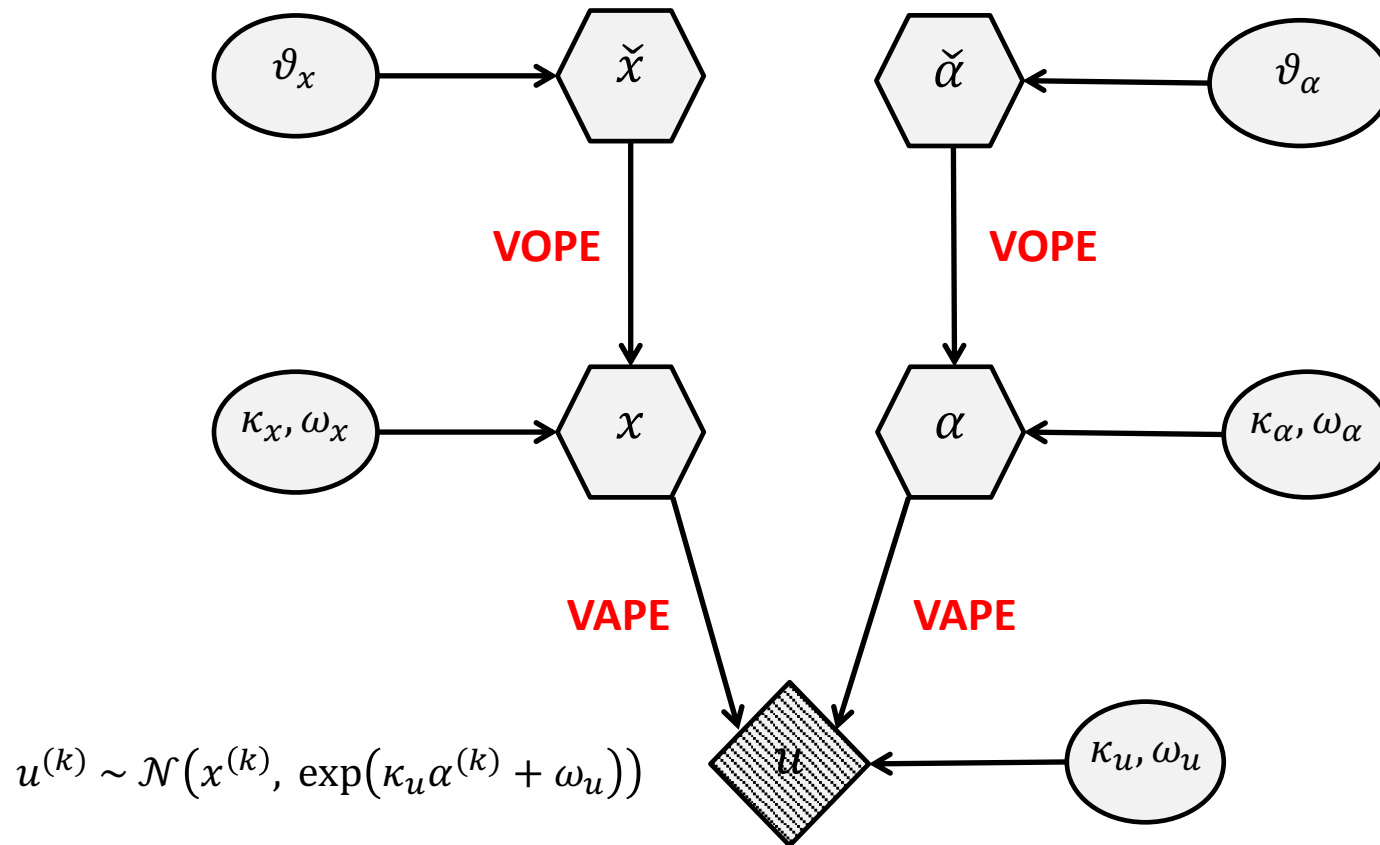


Jumping Gaussian estimation task



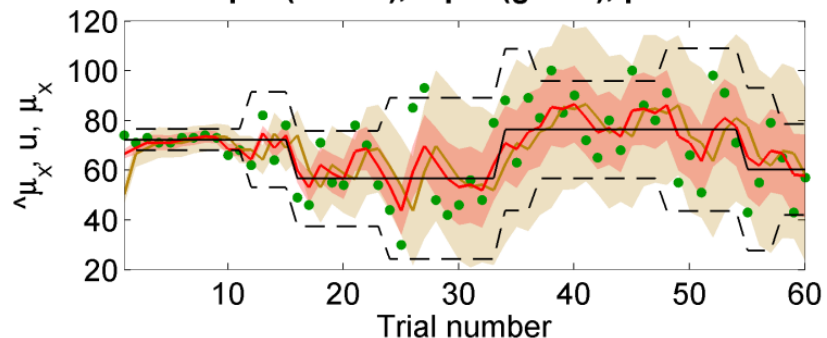
Data from Chaohui Guo

Independent mean and variance model

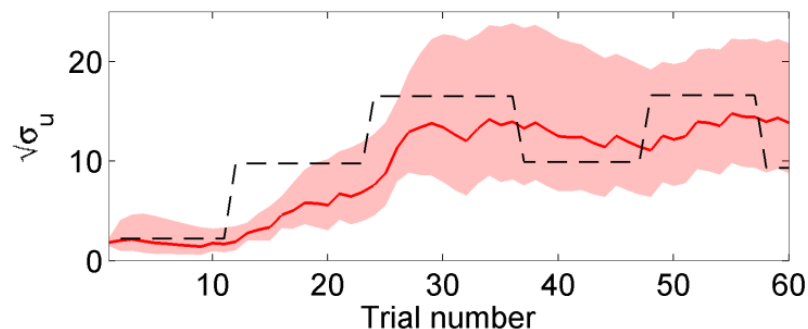


Jumping Gaussian estimation task

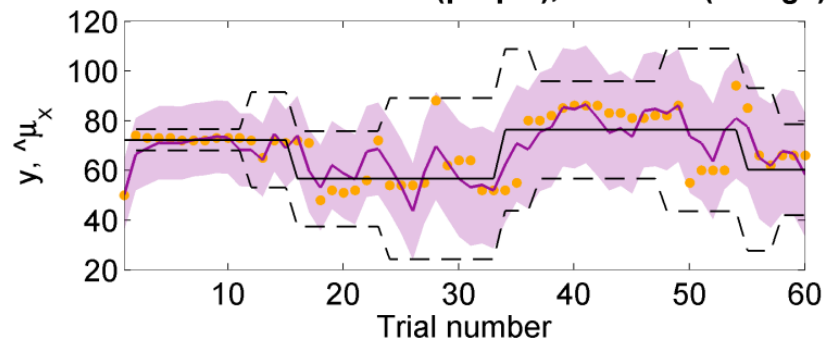
Prediction of input (brown), input (green), posterior belief (red)



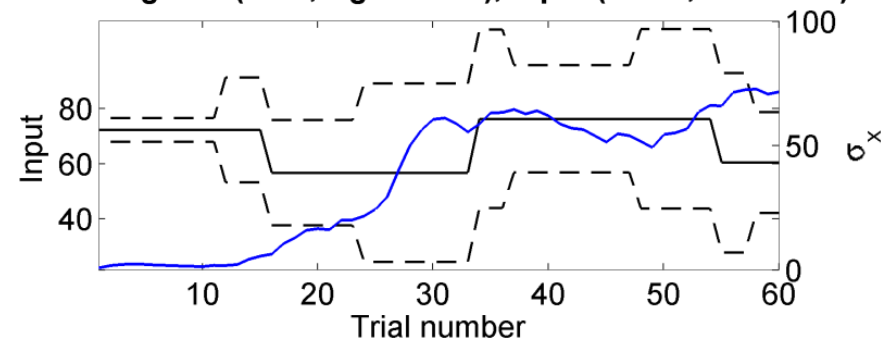
Belief on noise (red), true noise (dashed black)



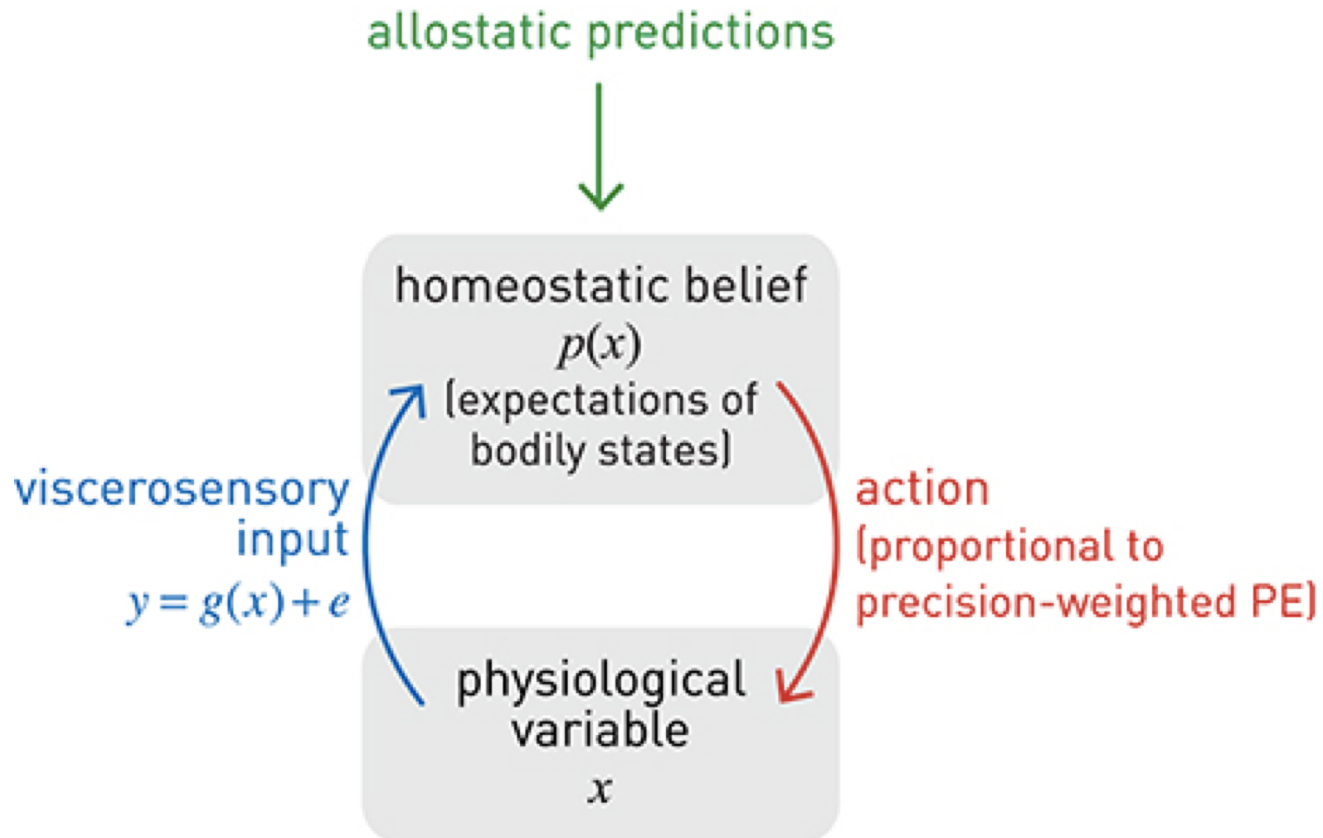
Prediction of decision (purple), decision (orange)



Learning rate (blue; right scale), input (black; left scale)



Action as active inference



Action as active inference

Model:

$$p(x) = N(x; \mu_t, \pi_t^{-1})$$

$$p(y|x) = N(y; g(x), \pi_{data}^{-1})$$

Inference (i.e., belief update):

$$\mu_{t+1} = \mu_t + \frac{\pi_{data}}{\pi_t + \pi_{data}} (y_t - g(\mu_t))$$

$$\pi_{t+1} = \pi_t + \pi_{data}$$

Action as active inference

Delta priors on mean and precision of state:

$$\begin{aligned} p(y|m_H) &= \int p(y|\mu_t, \pi_t) p(\mu_t) p(\pi_t) d\mu_t d\pi_t \\ &= \int N(y; g(\mu_t), \pi_t^{-1}) \delta(\mu_t - \mu_{prior}) \\ &\quad \delta(\pi_t - \pi_{prior}) d\mu_t d\pi_t \\ &= N(y; g(\mu_{prior}), \pi_{prior}^{-1}) \end{aligned}$$

Negative of log-evidence L is Shannon surprise S :

$$\begin{aligned} L &= \ln p(y|m_H) & S(y|m_H) &= -L \\ &= \frac{1}{2} \left(\ln \pi_{prior} - \pi_{prior} (y - g(\mu_{prior}))^2 \right) + c \\ &= \frac{1}{2} \left(\ln \pi_{prior} - \pi_{prior} (PE(y))^2 \right) + c \end{aligned}$$

Action as active inference

Definition of action:

$$\begin{aligned} a(t) &= \frac{\partial L}{\partial x} \\ &= \frac{\partial \left[-\frac{1}{2} \pi_{\text{prior}} (y(x, t) - g(\mu_{\text{prior}}))^2 \right]}{\partial x} \\ &= \frac{\partial \left[-\frac{1}{2} \pi_{\text{prior}} PE(y)^2 \right]}{\partial x} \\ &= -\frac{\pi_{\text{prior}}}{2} \frac{\partial [PE(y)^2]}{\partial y} \frac{\partial y}{\partial x} \\ &= -\pi_{\text{prior}} PE(y) \frac{\partial g}{\partial x} \end{aligned}$$

precision-weighted prediction error

Action induces gradient descent on surprise S :

$$\frac{dx}{dt} = \lambda^{-1} f(a(t)) \quad \frac{dx}{dt} = -\lambda^{-1} f\left(\frac{\partial S}{\partial x}\right)$$

Action as active inference

