Learning and Forgetting Using Reinforced Bayesian Change Detection

Vincent Moens[1], Alexandre Zénon[1,2],

1 CoAction Lab, Institue of Neuroscience, Université Catholique de Louvain, Bruxelles, Belgium
2 INCIA, Université de Bordeaux, Bordeaux, France

¤Current Address: COSY-B1.53.04, University of Louvain, 53, Avenue Mounier, 1200 Brussels, Belgium

¶Membership list can be found in the Acknowledgments section.

* vincent.moens@uclouvain.be

Abstract

Agents living in volatile environments must be able to detect changes in contingencies while refraining to adapt to unexpected events that are caused by noise. In Reinforcement Learning (RL) frameworks, this requires learning rates that adapt to past reliability of the model. The observation that behavioural flexibility in animals tends to decrease following prolonged training in stable environment provides experimental evidence for such adaptive learning rates. However, in classical RL models, learning rate is either fixed or scheduled and can thus not adapt dynamically to environmental changes. Here, we propose a new Bayesian learning model, using variational inference, that achieves adaptive change detection by the use of Stabilized Forgetting, updating its current belief based on a mixture of fixed, initial priors and previous posterior beliefs. The weight given to these two sources is optimized alongside the other parameters, allowing the model to adapt dynamically to changes in environmental volatility and to unexpected observations. This approach is used to implement the "critic" of an actor-critic RL model, while the actor samples the resulting value distributions to choose which action to undertake. We show that our model can emulate different adaptation strategies to contingency changes, depending on its prior assumptions of environmental stability, and that model parameters can be fit to real data with high accuracy. The model also exhibits trade-offs between flexibility and computational costs that mirror those observed in real data. Overall, the proposed method provides a general framework to study learning flexibility and decision making in RL contexts.

Author summary

In stable contexts, animals and humans exhibit automatic behaviour that allows them to make fast decisions. However, these automatic processes exhibit a lack of flexibility when environmental contingencies change. In the present paper, we propose a model of behavioural automatization that is based on adaptive forgetting and that emulates these properties. The model builds an estimate of the stability of the environment and uses this estimate to adjust its learning rate and the balance between exploration and

exploitation policies. The model performs Bayesian inference on latent variables that represent relevant environmental properties, such as reward functions, optimal policies or environment stability. From there, the model makes decisions in order to maximize long-term rewards, with a noise proportional to environmental uncertainty. This rich model encompasses many aspects of Reinforcement Learning (RL), such as Temporal Difference RL and counterfactual learning, and accounts for the reduced computational cost of automatic behaviour. Using simulations, we show that this model leads to interesting predictions about the efficiency with which subjects adapt to sudden change of contingencies after prolonged training.

1 Introduction

Learning agents must be able to deal efficiently with surprising events when trying to represent the current state of the environment. Ideally, agents response to such events should depend on their belief about how likely the environment is to change. When expecting a steady environment, a surprising event should be considered as an accident and should not lead to updating previous beliefs. Conversely, if the agent assumes the environment is volatile, then a single unexpected event should trigger forgetting of past beliefs and relearning of the (presumably) new contingency. Importantly, assumption on environmental volatility can also be learned from experience.

Here, we propose a general model that implements this adaptive behaviour using Bayesian inference. This model is divided in two parts: the critic which learns the environment and the actor that makes decision on the basis of the learned model of the environment.

The critic side of the model (called Hierarchical Adaptive Forgetting Variational Filter, HAFVF) discriminates contingency changes from accidents on the basis of past environmental volatility, and adapt its learning accordingly. This learner is a special case of Stabilized Forgetting (SF) [1]: practically, learning is modulated by a forgetting factor that controls the relative influence of past data with respect to a fixed prior distribution reflecting the naive knowledge of the agent. At each time step, the goal of the learner is to infer whether the environment has changed or not. In the former case, she erases her memory of past events and resets her prior belief to her initial prior knowledge. In the latter, she can learn a new posterior belief of the environment structure based on her previous belief. The value of the forgetting factor encodes these two opposite behaviours: small values tend to bring parameters back to their original prior, whereas large values tend to keep previous posteriors in memory. The first novelty of our approach lies in the fact that the posterior distribution of that forgetting factor depends on the estimated stability of past observations. The second and most crucial novelty lies in the hierarchical structure of this forgetting scheme: indeed, the posterior distribution of the forgetting factor is itself subject to a certain forgetting, learned in a similar manner. This leads to a 3-level hierarchical organization in which the bottom level learns to predict the environment, the intermediate level represents its volatility and the top level learns how likely the environment is to change its volatility. We show that this model implements a generalization of classical Q-learning algorithms.

The actor side of the model is framed as a full Drift-Diffusion Model of decision making [2] (Normal-Inverse-Gamma Diffusion Process; NIGDM) that samples from the value distributions inferred from the critic in order to select actions in proportion to their probability of being the most valued. We show that this approach predicts plausible results in terms of exploration-exploitation policy balance, reward rate, reaction times (RT) and cognitive cost of decision. Using simulated data, we also show

Algorithm 1 AC-HAFVF sketch.

for $j = 1$ **to** J **do**
  Actor: $NIGDM$ Sec 2.6
  select $a_j \sim p\left(a_j = \arg\max_{\mathbf{a}} \mu^r(s_j, \mathbf{a}) \,|\, \mathbf{x}_{<j}\right)$
  Observe $\mathbf{x}_j = \{r(s_j, a_j), s_{j+1}\}$

  Critic: $HAFVF$ Sec 2.3
  update $q_j(\mathbf{z}(s_j, a_j)) \approx p\left(\mathbf{z}(s_j, a_j) \,|\, \mathbf{x}_j; \mathbf{x}_{<j}, \boldsymbol{\theta}_0\right)$
end for

that the model can uncover specific features of human behaviour in single and multi-stage environments. The whole model is outlined in Algorithm 1: the agent first select an action given an (approximate) Q-sampling policy, which is temporally implemented as a Full DDM [2] with variable drift rate and accumulation noise, then learns based on the return of the action executed (reward $r(s_j, a_j)$ and transition $s' = T(s_j, a_j)$). Then, the critic updates its approximate posterior belief about the state of the environment $q_j \approx p$.

We apply the proposed approach to Model-Free RL contexts (i.e. to agents limiting their knowledge of the environment to a set of reward functions) in an extensive manner. We explore in detail the application of our algorithm to Temporal Discounting RL, in which we study the possibility of learning the discounting factor as a latent variable of the model. We also highlight various methods for accounting for unobserved events in a changing environment. Finally, we show that the way our algorithm optimizes the exploration-exploitation balance is similar to Q-Value Sampling when using large DDM thresholds.

Importantly, the proposed approach is very general, and even though we apply it here only to Model-Free Reinforcement Learning, it could be also extended to Model-Based RL [3], where the agent models a state-action to state transition table together with the reward functions. Additionally, other machine-learning algorithms could also benefit from this approach [4].

The paper is structured as follows: first (Sec 1.1) we review briefly the state of the art and place our work within the context of current literature. In Sec 2, we present the mathematical details of the model. We derive the analytical expressions of the learning rule, and frame them in a biological context. We then show how this learning scheme directly translates into a decision rule that constitutes a special case of the Sequential Sampling family of algorithms. In the result section Sec 3, we show various predictions of our model in terms of learning and decision making. More importantly, we show that despite its complexity, the model can be fitted to behavioural data. We conclude by reviewing the contributions of the present work, highlighting its limitations and putting it in a broader perspective.

1.1 Related work

The adaptation of learning to contingency changes and noise has numerous connections to various scientific fields from cognitive psychology to machine learning. A classical finding in behavioural neuroscience is that instrumental behaviours tend to be less and less flexible as subjects repeatedly receive positive reinforcement after selecting a certain action in a certain context, both in animals [5–7] and humans [8–11]. This suggests that biological agents indeed adapt their learning rate to inferred environmental stability: when the environment appears stable (e.g. after prolonged experience of a rewarded

stimulus-response association), they show increased tendency to maintain their model of the environment unchanged despite reception of unexpected data.

Most studies on such automatization of behaviour have focused on action selection. However, weighting new evidence against previous belief is also a fundamental problem for perception and cognition [12–14]. Predictive coding [15–20] provides a rich, global, framework that has the potential to tackle this problem, but an explicit formulation of cognitive flexibility is still lacking. For example, whereas [20] provides an elegant Kalman-like Bayesian filter that learns the current state of the environment based on its past observations and predicts the effect of its actions, it assumes a stable environment and cannot, therefore, adapt dynamically to contingency changes. The Hierarchical Gaussian Filter (HGF) proposed by Mathys and colleagues [21, 22] provides a mathematical framework that implements learning of a sensory input in a hierarchical manner, and that can account for the emergence of inflexibility in various situations. This model deals with the problem of flexibility (framed as expected "volatility") by building a hierarchy of random variables: each of these variables is distributed with a Gaussian distribution with a mean equal to this variable at the trial before and the variance equal to a non-linear transform of the variable at superior level. Each level encodes the distribution of the volatility of the level below. Although it has shown its efficiency in numerous applications [**?**, 23–27], a major limitation of this model, within the context of our present concern, is that it makes the assumption that the volatility of the environment depends deterministically on the variance of the observations. Crucially, this model cannot forget past experience after changes of contingency, but can only adapt its learning to the current contingency. Consequently, the HGF model will not be able to account for environments whose volatility change over time, i.e. where changes in contingencies occur sometimes abruptly, sometimes progressively. Moreover, HGF will have difficulties accounting for changes in the variance of the observations. These caveats limit the applicability of the HGF to a narrow range of datasets.

As will be shown in detail below, in the model proposed in the present paper, volatility is not only a function of the variance of the observations: if a new observation falls close enough from previous estimates then the agent will refine its posterior estimate of the variance and will decrease its forgetting factor (i.e. will move its prior away from the fixed initial prior and closer to the learned posterior from the previous trial), but if the new observation is not likely given this posterior estimate, the forgetting factor will increase (i.e. will move closer to the fixed initial prior) and the model will tend to update to a novel state (because of the low precision of the initial prior). In **??**, we show that our model outperforms the HGF in such situations. Another, more technical, limitation of the HGF comes from the non-conjugate form of the generative model, which makes the use of mean-field VB challenging, and requires reliance on numerous approximations. For instance, the quadratic approximations on which the HGF relies are only valid for large n [28], but the HGF is practically applicable only to small datasets because the parameters sub-space where updates are possible varies widely with the data, such that optimization fails in many instances and especially for large datasets. The model we provide, on the contrary, has no such constrain, and can be generalized to many combinations of exponential family models (e.g. Bernoulli or multivariate Gaussian distribution at the first level) without tedious derivations.

In Machine Learning and in Statistics, too, the question of whether new unexpected data should be classified as outlier or environmental change is important [29]. This problem of "denoising" or "filtering" the data is ubiquitous in science, and usually relies on arbitrary assumption on environmental stability. In signal processing and system identification, adaptive forgetting is a broad field where optimality is highly context (and prior)-dependant [1]. Bayesian Filtering (BF) [30], and in particular the Kalman

Filter [31] often lack the necessary flexibility to model real-life signals that are, by nature, changing. One can discriminate two approaches to deal with this problem: whereas Particle Filtering (PF) [32–34] is computationally expensive, the SF family of algorithms [1,35], from which our model is a special case, usually has greater accuracy for a given amount of resources [32] (for more information, we refer to [31] where SF is reviewed). Most previous approaches in SF have used a truncated exponential prior [36,37] or a fixed, linear mixture prior to account for the stability of the process [33]. Our approach is innovative in this field in two ways: first, we use a Beta prior on the mixing coefficient (unusual but not unique [38]), and we adapt the posterior of this forgetting factor on the basis of past observations, the prior of this parameter and its own adaptive forgetting factor. Second, we introduce a hierarchy of forgetting that stabilizes the learning when the training length is long.

We therefore intend to focus our present research on the very question of flexibility. We will show how flexibility can be implemented in a Bayesian framework using an adaptive forgetting factor, and what prediction this framework makes when applied to learning and decision making in Model-Free paradigms.

2 Methods

2.1 Bayesian Q-Learning and the problem of flexibility

Classical RL [39], or Bayesian RL [40,41] cannot discriminate learners that are more prone to believe in a contingency change from those who tend to disregard unexpected events and consider them as noise. To show this, we take the following example: let $p(\rho|r_{\leq j}) = \text{Beta}(\alpha, \beta)$ be the posterior probability at trial j of a binary reward $r_j \sim \text{Bern}(\rho)$ with prior probability $\rho \sim \text{Beta}(\alpha_0, \beta_0)$. It can be shown that, at the trial $j = v_j + u_j$, where v_j is the number of successes and u_j the number of failures, the posterior probability parameters read $\{\alpha_j = \alpha_0 + v_j, \beta_j = \beta_0 + u_j\}$. This can be easily mapped to a classical RL algorithm if one considers that, at each update of v and u, the posterior expectation of ρ is updated by

$$\begin{aligned}
\mathbb{E}\left[\rho|r_{\leq j}\right] &= \frac{v_{j-1} + r_j}{v_{j-1} + r_j + u_{j-1} + (1 - r_j)} \\
&= \frac{j-1}{j}\mathbb{E}\left[\rho_{j-1}\right] + \frac{r_j}{j} \\
&= \mathbb{E}\left[\rho_{j-1}\right] + \eta(r_j - \mathbb{E}\left[\rho_{j-1}\right]) \\
\text{where } \eta &\triangleq \frac{1}{j}
\end{aligned}$$

which has the form of a classical myopic Q-learning algorithm with a decreasing learning rate.

The drawback of this fixed-schedule learning rate is that, if the number of observed successes outnumbers greatly the number of failures ($v \gg u$) at the time of a contingency change in which failures become suddenly more frequent, the agent will need $v - u + 1$ failures to start considering that $p(r_j = 0|r_{\leq j}) > p(r_j = 1|r_{\leq j})$. This behaviour is obviously sub-optimal in a changing environment, and Dearden [40] suggests adding a constant forgetting factor to the updates of the posterior, making therefore the agent progressively blind to past outcomes. Consider the case in which

$$\mathbb{E}\left[\rho|r_{\leq j}\right] = \frac{wv_{j-1} + r_j}{wv_{j-1} + r_j + wu_{j-1} + (1 - r_j)}$$

with $w \in [0;1]$ being the forgetting factor. We can easily see that, in the limit case of $\alpha_0 = 0$ and $\beta_0 = 0$, $\alpha_j + \beta_j \to \frac{1}{1-w}$ as $j \to \infty$. We can define $\frac{1}{1-w}$ as the *efficient memory* of the agent, which provides a bound on the *effective memory*, represented by the total amount of trials taken into account so far (e.g. $\alpha_j + \beta_j$ in the previous example). This produces an upper and lower bound to the variance of the posterior estimate of $p(\rho|r_{\leq j})$. This can be seen from the variance of the beta distribution $\mathbb{V}\mathrm{ar}[\rho \,|\, r_{\leq j}] = \frac{(\alpha_j)(\beta_j)}{(\alpha_j+\beta_j)^2(\alpha_j+\beta_j+1)}$ which is maximized when $\alpha_j = \beta_j$, and minimized when either $\alpha_j = \alpha_0$ or $\beta_j = \beta_0$. In a steady environment, agents with larger memory are advantaged since they can better estimate the variance of the observations. But when the environment changes, large memory becomes disadvantageous because it requires longer time to adapt to the new contingency. Here, we propose a natural solution to this problem by having the agent erase its memory when a new observation (or a series of observations) is unlikely given the past experience.

2.2 General framework

Our framework is based on the following assumptions:

Assumption 1 *The environment is fully Markovian: the probability of the current observation given all the past history is equal to the probability of this observation given the previous observation.*

Assumption 2 *At a given time point, all the observations (rewards, state transitions, etc.) are i.i.d. and follow a distribution $p(\mathbf{x} \,|\, \mathbf{z})$ that is issued from the exponential family and has a conjugate prior that also belongs to the exponential family $p(\mathbf{z} \,|\, \boldsymbol{\theta}_0)$.*

For conciseness, the latent variables $\mathbf{z}$ (i.e. action value, transition probability etc.) and their prior $\boldsymbol{\theta}$ will represent the natural parameters of the corresponding distributions in what follows.

Assumption 3 *The agent builds a hierarchical model of the environment, where each of the distributions at the lower level (reward and state transitions) are independent, i.e. the reward distribution for one state-action cannot be predicted from the distribution of the other state-actions.*

Assumption 4 *The agent can only observe the effects of the action she performs.*

Model specifications We are interested in deriving the posterior probability of some datapoint-specific measure $\mathbf{z}_j$, where j indicates the point in time, given the past and current observations $\mathbf{x}_{\leq j}$ and some prior belief $\boldsymbol{\theta}_0$. Bayes theorem states that this is equal to

$$p(\mathbf{z} \,|\, \mathbf{x}) = \frac{p(\mathbf{z} \,|\, \boldsymbol{\theta}_0) \prod_{j=1}^{J} p(\mathbf{x}_j \,|\, \mathbf{z})}{\prod_{j=1}^{J} p(\mathbf{x}_j)}. \tag{1}$$

We now consider the case of a subject that observes the stream of data and updates her posterior belief on-line as data are gathered. According to Assumption 2, one can express the posterior of $\mathbf{z}$ given the current observation x_j

$$p(\mathbf{z} \,|\, x_j, \mathbf{x}_{<j}) = \frac{p(x_j \,|\, \mathbf{z}) p(\mathbf{z} \,|\, \mathbf{x}_{<j})}{p(x_j \,|\, \mathbf{x}_{<j})}. \tag{2}$$

It appears immediately that the prior $p(\mathbf{z} \,|\, \mathbf{x}_{<j})$ has the same form as the previous posterior, so that our posterior probability function can be easily estimated recursively using the last posterior estimate as a prior (yesterday's posterior is today's prior) until $p(\mathbf{z} \,|\, \boldsymbol{\theta}_0)$ is reached. Assumption 2 implies that the posterior $p(\mathbf{z} \,|\, x_j, \mathbf{x}_{<j})$ will be tractable: since $p(\mathbf{x} \,|\, \mathbf{z})$ is from the exponential family and has a conjugate prior $p(\mathbf{z} \,|\, \boldsymbol{\theta}_0)$, the posterior probability has the same form as the prior, and has a convenient expression:

$$\boldsymbol{\theta}_j = \begin{bmatrix} \boldsymbol{\theta}_j^{\xi} \\ \theta_j^{\eta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\theta}_0^{\xi} + \sum_{i=1}^{j} \mathbf{T}(\mathbf{x}_i) \\ \theta_0^{\eta} + j \end{bmatrix} \tag{3}$$

where $\mathbf{T}(\mathbf{x}_i)$ is the sufficient statistics of the ith sample, and where we have made explicit the fact that $\boldsymbol{\theta}_0 = \{\boldsymbol{\theta}_0^{\xi}, \theta_0^{\eta}\}$ can be partitioned in two parts, from which θ_0^{η} represents the prior number of observations. Consequently, θ_j^{η} is equivalent to the effective memory introduced above.

This simple form of recursive posterior estimate suffers from the drawbacks we want to avoid, i.e. it does not forget past experience. We therefore introduce a two-component mixture prior where the previous posterior is weighted against the original prior belief:

$$p(\mathbf{z} \,|\, x_{<j}; \boldsymbol{\theta}_0) \triangleq \frac{p(\mathbf{z} \,|\, x_{<j})^w p(\mathbf{z} \,|\, \boldsymbol{\theta}_0)^{1-w}}{Z(w, x_{<j}, \boldsymbol{\theta}_0)} \quad \text{with } w \in [0; 1]. \tag{4}$$

The exponential weights on this prior mixture allow us to easily write its logarithmic form, but it still demands to compute the normalizing constant $Z(w, x_{<j}, \boldsymbol{\theta}_0) = \int p(\mathbf{z} \,|\, x_{<j})^w p(\mathbf{z} \,|\, \boldsymbol{\theta}_0)^{1-w} d\,\mathbf{z}$. This constant has, however, a closed-form if both the prior and the previous posterior are from the same distribution, issued from the exponential family.

In Eq 4, we assume that the forgetting factor is unknown and that the learner needs to infer it from the data at hand. We therefore put a beta prior on this parameter, and under the assumption that the posterior probability factorizes (Mean-Field assumption), the joint probability at time j reads:

$$p(x_j, \mathbf{z}, w | x_{<j}; \boldsymbol{\theta}_0, \boldsymbol{\phi}_0) = p(x_j \,|\, \mathbf{z}) \frac{p(\mathbf{z} \,|\, x_{<j})^w p(\mathbf{z} \,|\, \boldsymbol{\theta}_0)^{1-w}}{Z(w, x_{<j}, \boldsymbol{\theta}_0)} p(w \,|\, x_{<j}, \boldsymbol{\phi}_0) \tag{5}$$

where $\boldsymbol{\phi}_0$ is the vector of the parameters of the beta prior of w. The model in Eq 5 is not conjugate, and the posterior is therefore not guaranteed to be tractable anymore.

2.3 Hierarchical Filter

Let us now analyze the expected behaviour of an agent using a model similar to the one just described, in a steady environment: if all $\mathbf{x} \in \mathbf{x}_{\leq j}$ belong to the same, unknown distribution $p(\mathbf{x} \,|\, \mathbf{z})$, the value of $\mathbb{E}_{q_j(\mathbf{z})}[\mathbf{z}]$ will progressively converge to its true value as the prior (or the previous posterior) over w will eventually put a lot of weight on the past experience (i.e. it favours high values of w), since the distribution from which $\mathbf{x}$ is drawn is stationary. We have shown that such models rapidly tend to an overconfident posterior over w [4]. In practice, when the previous posterior of w is confident on the value that w should take (i.e. has low variance), it tends to favor updates that reduce variance further, corresponding to values of $p_j(w \,|\, \mathbf{x}_{\leq t})$ that match $p_{j-1}(w \,|\, \mathbf{x}_{<t})$, even if this means ignoring an observed mismatch between $p_{j-1}(\mathbf{z} \,|\, \mathbf{x}_{<t})$ and $p_j(\mathbf{z} \,|\, \mathbf{x}_{\leq t})$. In order to deal with this issue, we enrich our model by introducing a third level in the hierarchy.

We re-define the prior over w as a two-component mixture of priors:

$$p_j(w \,|\, x_{<j}; b, \boldsymbol{\phi}_0) \triangleq \frac{p(w \,|\, x_{<j})^b p(w \,|\, \boldsymbol{\phi}_0)^{1-b}}{Z(b, x_{<j}, \alpha_0)}$$

and the full joint probability has the form

$$\begin{aligned} p(x_j, \mathbf{z}, w | x_{<j}; \boldsymbol{\theta}_0, \boldsymbol{\phi}_0) = p(x_j \,|\, \mathbf{z}) & \frac{p(\mathbf{z} \,|\, x_{<j})^w p(\mathbf{z} \,|\, \boldsymbol{\theta}_0)^{1-w}}{Z(w, x_{<j}, \boldsymbol{\theta}_0)} \\ & \frac{p(w \,|\, x_{<j})^b p(w \,|\, \boldsymbol{\phi}_0)^{1-b}}{Z(b, x_{<j}, \boldsymbol{\phi}_0)} p(b \,|\, x_{<j}, \boldsymbol{\beta}_0). \end{aligned} \tag{6}$$

2.4 Variational Inference

Two classes of methods can be used to estimate the posterior probability at the previous time step needed in Eq 6. Simulation-based algorithms [42] such as importance sampling, particle filtering or Markov Chain Monte Carlo, are asymptotically exact but computationally expensive, especially in the present case where the estimate has to be refined at each time step. The other class of methods, approximate inference [43, 44], consists in formulating, for a model with parameters $\mathbf{y}$ and data $\mathbf{x}$, an approximate posterior $q(\mathbf{y})$, that we will use as a proxy to the true posterior $p(\mathbf{y} \,|\, \mathbf{x})$. Roughly, approximate inference can be partitioned into Expectation Propagation and Variational Bayes (VB) methods. Let us consider in more detail VB, as it is the core of our learning model. In VB, optimizing the approximate posterior amounts to computing a lower-bound to the log model evidence (ELBO) $\mathcal{L}\left(q(\mathbf{y})\right) \leq \log p(\mathbf{x} \,|\, x_{<j})$, whose distance from the true log model evidence can be reduced by gradient descent [45]. Hybrid methods, that combine sampling methods with approximate inference, also exist (e.g. Stochastic Gradient Variational Bayes [46] or Markov Chain Variational Inference [47]). With the use of refined approximate posterior distributions [**?**, 48, 49], they allow for highly accurate estimates of the true posterior with possibly complex, non-conjugate models.

We define a variational distribution over $\mathbf{y}$ with parameters $\boldsymbol{\upsilon}$: $q(\mathbf{y} \,|\, \boldsymbol{\upsilon})$, which we will use as a proxy to the real, but unknown, posterior distribution $p(\mathbf{y} \,|\, \mathbf{x})$. The two distribution match exactly when their Kullback-Leibler divergences are equal to zero, i.e.

$$\begin{aligned} & D_{KL}\left[q(\mathbf{y}) \big| \big| p(\mathbf{y} \,|\, \mathbf{x})\right] = & 0 \\ \Leftrightarrow \quad & D_{KL}\left[p(\mathbf{y} \,|\, \mathbf{x}) \big| \big| q(\mathbf{y})\right] = & 0 \\ \Leftrightarrow \quad & p(\mathbf{y} \,|\, \mathbf{x}) = & q(\mathbf{y}) \end{aligned}$$

where we have omitted the approximate posterior parameters $\boldsymbol{\upsilon}$ for sparsity of the expressions. Given some arbitrary constraints on $q(\mathbf{y})$, we can choose (for mathematical convenience) to reduce $D_{KL}\left[q(\mathbf{y}) \big| \big| p(\mathbf{y} \,|\, \mathbf{x})\right]$ wrt $q(\mathbf{y})$. Formally, this can be written as

$$\begin{aligned} q^*(\mathbf{y}) &= \arg\min_{q(\mathbf{y})} D_{KL}\left[q(\mathbf{y}) \big| \big| p(\mathbf{y} \,|\, \mathbf{x})\right] \\ &= \arg\min_{q(\mathbf{y})} \int q(\mathbf{y}) \log \frac{q(\mathbf{y})}{p(\mathbf{y} \,|\, \mathbf{x})} d\,\mathbf{y} \\ &= \arg\min_{q(\mathbf{y})} \int q(\mathbf{y}) \log q(\mathbf{y}) d\,\mathbf{y} - \int q(\mathbf{y}) \log p(\mathbf{y} \,|\, \mathbf{x}) d\,\mathbf{y}\,. \end{aligned}$$

We can now substitute $\log p(\mathbf{y}\,|\,\mathbf{x})$ by its rhs in the log-Bayes formula

$$\begin{aligned} D_{KL}\left[q(\mathbf{y})\big|\big|p(\mathbf{y}\,|\,x)\right] &= \int q(\mathbf{y})\log q(\mathbf{y})d\,\mathbf{y} - \int q(\mathbf{y})\Big(\log p(\mathbf{x},\mathbf{y}) - \log p(\mathbf{x})\Big)d\,\mathbf{y} \\ \Leftrightarrow \log p(\mathbf{x}) &= \underbrace{\int q(\mathbf{y})\log p(\mathbf{x},\mathbf{y})d\,\mathbf{y} - \int q(\mathbf{y})\log q(\mathbf{y})d\,\mathbf{y}}_{\mathcal{L}(q(\mathbf{y}))} + D_{KL}\left[q(\mathbf{y})\big|\big|p(\mathbf{y}\,|\,\mathbf{x})\right]. \end{aligned} \tag{7}$$

Because $\log p(\mathbf{x})$ does not depend on the model parameters, it is fixed for a given dataset. Therefore, as we maximize $\mathcal{L}(q(\mathbf{y}))$ in Eq 7, we decrease the divergence $D_{KL}\left[q(\mathbf{y})\big|\big|p(\mathbf{y}\,|\,\mathbf{x})\right]$ between the approximate and the true posterior. When a maximum is reached, we can consider that 1) we have obtained the most accurate approximate posterior given our initial assumptions about $q(\mathbf{y})$ and 2) $\mathcal{L}(q(\mathbf{y}))$ provides a lower bound to $\log p(\mathbf{x})$. It should be noted here that the more $q(\mathbf{y})$ is flexible, the closer we can hope to get from the true posterior, but this is generally at the expense of tractability and/or computational resources.

The ELBO in Eq 7 is the sum of the expected log joint probability and the entropy of the approximate posterior. In order for the former to be tractable, one must carefully choose the form of the approximate posterior. The Mean-field assumption we have made allows us to select, for each factor of the approximate posteriors, a distribution with the same form as their conjugate prior, which is the best possible configuration in this context [50].

Applying now this approach to Eq 6, our spherical approximate posterior looks like:

$$q(\mathbf{z}, w, b) \triangleq q(\mathbf{z}\,|\,\boldsymbol{\theta})q(w\,|\,\boldsymbol{\phi})q(b\,|\,\boldsymbol{\beta}).$$

In addition, in order to recursively estimate the current posterior probability of the model parameters given the past, we make the natural approximation that the true previous posterior can be substituted by its variational approximation:

$$p(\mathbf{z}\,|\,x_{<j}) \approx q_{j-1}(\mathbf{z}) \tag{8}$$

and similarly for $p(w\,|\,x_{<j})$ and $p(b\,|\,x_{<j})$. The use of this distribution as a proxy to the posterior greatly simplifies the optimization of $q_j(\mathbf{z}, w, b)$.

The full, approximate joint probability distribution at time j therefore looks like

$$\begin{aligned} p(x_j, \mathbf{z}, w|x_{<j}) \approx p(x_j\,|\,\mathbf{z})&\frac{q_{j-1}(\mathbf{z}\,|\,\boldsymbol{\theta}_{j-1})^w p(\mathbf{z}\,|\,\boldsymbol{\theta}_0)^{1-w}}{Z(w, \boldsymbol{\theta}_{j-1}, \boldsymbol{\theta}_0)} \\ &\frac{q_{j-1}(w\,|\,\boldsymbol{\phi}_{j-1})^b p(w\,|\,\boldsymbol{\phi}_0)^{1-b}}{Z(b, \boldsymbol{\phi}_{j-1}, \boldsymbol{\phi}_0)} q(b\,|\,\boldsymbol{\beta}_{j-1}) \end{aligned}$$

where $\boldsymbol{\theta}_{j-1}$, $\boldsymbol{\phi}_{j-1}$ and $\boldsymbol{\beta}_{j-1}$ are the variational parameters at the last trial for $\mathbf{z}, w$ and b respectively. A further advantage of the approximation made in Eq 8 is that the prior of $\mathbf{z}$ and w simplifies elegantly:

$$\begin{aligned} p(x_j, \mathbf{z}, w, b|x_{<j}) \approx p(x_j\,|\,\mathbf{z})&p(\mathbf{z}\,|\,w(\boldsymbol{\theta}_{j-1} - \boldsymbol{\theta}_0) + \boldsymbol{\theta}_0)\times \\ &p(w\,|\,b(\boldsymbol{\phi}_{j-1} - \boldsymbol{\phi}_0) + \boldsymbol{\phi}_0)p(b\,|\,\boldsymbol{\beta}_{j-1}) \end{aligned} \tag{9}$$

(see Appendix A for the full derivation).

A conjugate distribution for $p(w)$ is hard to find. Šmìdl and Quinn [51] propose a uniform prior and a truncated exponential approximate posterior over w. They

interpolate the normalizing constant between two fixed value of w, which allows them to perform closed-form updates of this parameter. Here, we chose $p(w \mid \boldsymbol{\phi}_0)$ and $q(w \mid \boldsymbol{\phi}_j)$ to be both beta distributions, a choice that does not impair our ability to perform closed-form updates of the variational parameters as we will see in 2.5.1.

In this model (see Fig 1), named the Hierarchical Adaptive Forgetting Variational Filter, specific prior configurations will bend the learning process to categorize surprising events either as contingency changes, or as accidents. In contrast with other models [52], w and b are represented with a rich probability distribution where both the expected values and variances have an impact on the model's behaviour. For a given prior belief on $\mathbf{z}$, a confident prior over w, centered on high values of this parameter, will lead to a lack of flexibility that would not be observed with a less confident prior, even if they have the same expectation.

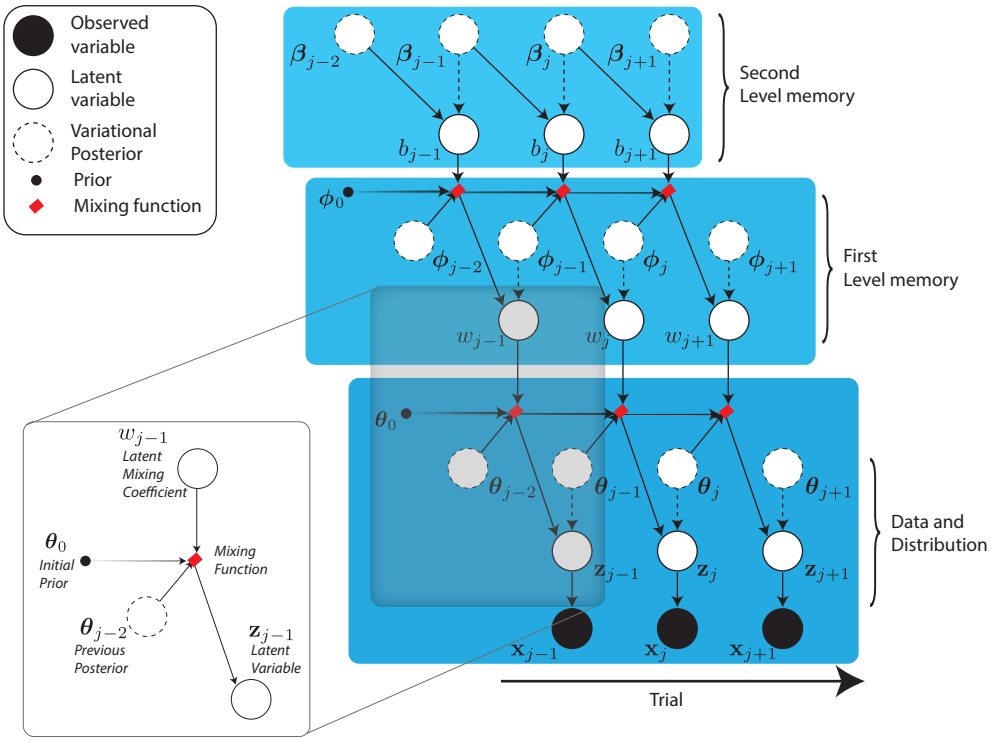


Figure 1. Directed Acyclic Graph of the HAFVF model. Plain circles represent observed variables, white circles represent latent variables and dots represents prior distribution parameters. Dashed circles and dashed arrows represent approximate posteriors and approximate posterior dependencies. A weighted prior latent node is highlighed.

2.5 The critic: HAFVF as a Reinforcement Learning Algorithm

Application of this scheme of learning to the RL case is straightforward, if one considers $\mathbf{x}_j$ as being the observed rewards and $\mathbf{z}$ as the parameters of the distribution of these rewards. In the following, we will assume that the agent models a normally distributed state-action reward function $x_j = r(s, a)$, from which she tries to estimate the posterior distribution natural parameters $\mathbf{z} \triangleq \boldsymbol{\eta}\left(\mu(s,a), \sigma(s,a)\right)$ where $\boldsymbol{\eta}(\cdot)$ is the natural

parameter vector of the normal distribution. In this context, an intuitive choice for the prior (and the approximate posterior) of these parameters is a Normal Inverse-Gamma distribution ($\mathcal{NG}^{-1}$): for the prior, we have

$$\mu(s,a) \sim \mathcal{N}\left(\mu_0^\mu(s,a), \frac{\sigma^2(s,a)}{\kappa_0^\mu(s,a)}\right)$$
$$\sigma^2(s,a) \sim \mathcal{G}^{-1}\left(\alpha_0^\sigma(s,a), \beta_0^\sigma(s,a)\right)$$

and the approximate posterior can be defined similarly with a normal component $\mathcal{N}\left(\mu_j^\mu(s,a), \frac{\sigma^2(s,a)}{\kappa_j^\mu(s,a)}\right)$ and a gamma component $\mathcal{G}^{-1}(\alpha_j^\sigma(s,a), \beta_j^\sigma(s,a))$.

2.5.1 Update equation

Even though the model is not formally conjugate, the use of a mixture of priors with exponential weights makes the variational update equations easy to implement for the first level. Let us first assert a few basic principles from the Mean-Field Variational Inference framework: it can be shown that, under the assumption that the approximate posterior factorizes in $q(y_1 y_2) = q(y_1)q(y_2)$, then the optimal distribution $q^*(y_1)$ given our current estimate of $q(y_2)$ is given by

$$q^*(y_1) = \exp\left(\mathbb{E}_{q(y_2)}\left[\log p(\mathbf{x}, \mathbf{y})\right] - \log Z(\mathbf{x})\right) \tag{10}$$

where $\log Z$ is some log-normalizer that does not depend on $\mathbf{y}$. Eq 10 states that each set of variational parameters can be updated independently given the current value of the others: this usually lead to an approach similar to EM [42], where one iterates through the updates of variational posterior successively until convergence.

Fortunately, thanks to the conjugate form of the lower level of the HAFVF, Eq 10 can be unpacked to a form that recalls Eq 3 where the update of the variational parameters of $\mathbf{z}$ reads:

$$\boldsymbol{\theta}_j = \begin{bmatrix} \widehat{\boldsymbol{\vartheta}}^\xi + \mathbf{T}(\mathbf{x}_i) \\ \widehat{\vartheta}^\eta + 1 \end{bmatrix} \tag{11}$$

where $\widehat{\boldsymbol{\vartheta}} \triangleq \mathbb{E}_{q(w)}\left[w\right](\boldsymbol{\theta}_{j-1} - \boldsymbol{\theta}_0) + \boldsymbol{\theta}_0$ is the weighted prior of $\mathbf{z}$ (see Appendix A). This update scheme can be mapped onto and be interpreted in terms of Q-learning [39] (see Appendix B). Again, $\widehat{\vartheta}^\eta + 1$ is the updated effective memory of the subject, which is bounded on the long term by the efficient memory $1/(1 - \mathbb{E}_q\left[w\right])$.

More specifically, for the approximate posterior of a single observation stream $\mathbf{x} = \{x_j \forall j \in 1 : J\}$ with corresponding parameters $\{\mu_{j+1}^\mu, \kappa_{j+1}^\mu, \alpha_{j+1}^\sigma, \beta_{j+1}^\sigma\}$, we can apply this principle easily, as the resulting distribution has the form of a $\mathcal{NG}^{-1}$ with parameters:

$$\begin{aligned}
\mu_j^\mu &= \frac{\widehat{\mathrm{w}}\ \kappa_{j-1}^\mu \mu_{j-1}^\mu + (1\text{-}\widehat{\mathrm{w}})\ \kappa_0^\mu \mu_0^\mu}{\kappa_j^\mu} + \frac{x_j}{\kappa_j^\mu} \\
\kappa_j^\mu &= \widehat{\mathrm{w}}\ \kappa_{j-1}^\mu + (1\text{-}\widehat{\mathrm{w}})\ \kappa_0^\mu + 1 \\
\alpha_j^\sigma &= \widehat{\mathrm{w}}\ \alpha_{j-1}^\sigma + (1\text{-}\widehat{\mathrm{w}})\ \alpha_0^\sigma + \frac{1}{2} \\
\beta_j^\sigma &= \widehat{\mathrm{w}}\ \beta_{j-1} + (1\text{-}\widehat{\mathrm{w}})\ \beta_0^\sigma + \frac{1}{2}\left(\widehat{\mathrm{w}}\ \kappa_{j-1}(\mu_j^\mu - \mu_{j-1}^\mu)^2 + (1\text{-}\widehat{\mathrm{w}})\ \kappa_0(\mu_j^\mu - \mu_0^\mu)^2 + (x_j - \mu_j^\mu)^2\right)
\end{aligned} \tag{12}$$

where we have used $\widehat{\mathrm{w}} = \mathbb{E}_{q(w)}[w]$.

Deriving updates for the approximate posterior over the mixture weights w and b is more challenging, as the optimal approximate posterior in Eq 10 does not have the same form as the beta prior due to the non-conjugacy of the model. Fortunately, non-conjugate variational message passing (NCVMP) [53] can be used in this context. In short, NCVMP minimizes an approximate KL divergence in order to find the value of the approximate posterior parameters that maximize the ELBO. Although NCVMP convergence is not guaranteed, this issue can be somehow alleviated by damping of the updates (i.e. updating the variational parameters to a value lying in between the previous value they endorsed and the value computed using NCVMP, see [53] for more details). The need for a closed-form formula of the expected log-joint probability constitutes another obstacle for the naive implementation of NCVMP to the present problem: indeed, computing the expected value of the log-partition functions $\log Z(w)$ and $\log Z(b)$ involves a weighted sum of the past variational parameters $\boldsymbol{\theta}_{j-1}$ and the prior $\boldsymbol{\theta}_0$, which are known, with a weight w, which is unknown. Expectation of this expression given $q(w)$ does not, in general, have an analytical expression. To solve this problem, we used the second order Taylor expansion around $\widehat{\mathrm{w}}$ (see Appendix A).

2.5.2 Counterfactual learning

As an agent performs a series of choices in an environment, she must also keep track of the actions not chosen and update her belief accordingly: ideally, the variance of the approximate posterior of the reward function associated with a given action should increase when that action is not selected, to reflect the increased uncertainty about its outcome during the period when no outcome was observed. This requirement implies counterfactual learning capability [54–56].

Two options will be considered here: the first option consists in updating the approximate posterior parameters of the non-selected action at each time step with an update scheme that pulls the approximate posterior progressively towards the prior $\boldsymbol{\theta}_0$, with a speed that depends on w, i.e. as a function of the belief the agent has about environment stability. The second approach will consist in updating the approximate posterior of the actions only when they are actually selected, but accounting for past trials during which that action was not selected. The mathematical details of these approaches are detailed in Appendix C.

Continuous Updating When an action is not selected, the agent can infer what value it would have had given the observed stability of the environment. In practice, this is done by updating the variational parameters of the selected *and* non-selected action using the observed reward for the former, and the expected reward and variance of this reward for the latter.

This approach can be beneficial for the agent in order to optimize her exploration/exploitation balance. In Appendix C.1, it is shown that if the agent has the prior belief that the reward variance is high, then the probability of exploring the non-chosen option will increase as the lag between the current trial and the last observation of the reward associated with this option increases.

Delayed Updating Even though the agent learns only actions that are selected, it can adapt its learning rate as a function of how distant in the past was the last time the action was selected. Formally, this approach considers that *if* the posterior probability had been updated at each time step and the forgetting factor w had been stable, the

impact of the observations n trials back in time would currently have an influence that would decrease geometrically with a rate $\omega \triangleq w^n$. We can then substitute the prior over $\mathbf{z}$ by:

$$p(\mathbf{z}|x_{<j}, \delta_j) \triangleq \frac{p_{j-1}(\mathbf{z} \,|\, x_{<j})^{\omega} p(\mathbf{z} \,|\, \boldsymbol{\theta}_0)^{1-\omega}}{Z(\omega, x_{<j}, \boldsymbol{\theta}_0)} \tag{13}$$

which is identical to Eq 4 except that w has been substituted by ω.

We name this strategy Delayed Approximate Posterior Updating, and present the expected results with this scheme in Sec 3.

2.5.3 Temporal Difference Learning

An important feature required for an efficient Model-Free RL algorithm is to be able to account for future rewards in order to make choices that might seem suboptimal to a myopic agent, but that make sense on the long run. This is especially useful when large rewards (or the avoidance of large punishments) can be expected in a near future.

In order to do this, one can simply sum the expected value of the next state to the current reward in order to perform the update of the reward distribution parameters. However, because the evolution of the environment is somehow chaotic, it is usually considered wiser to decay slightly the future rewards by a discount rate γ. This mechanism is in accordance with many empirical observations of animal behaviours [57–59], neurophysiological processes [60–62] and theories [63–65].

As the optimal value of γ is unknown to the agent, we can assume that she will try to estimate its posterior distribution from the data as she does for the mean and variance of the reward function. Appendix D shows how this can be implemented in the current context. An example of TD learning in a changing environment is given in Sec 3.4.

We now focus on the problem of decision making under the HAFVF.

2.6 The actor: Decision Making under the HAFVF

Bayesian Policy In a stable environment where the distribution of the action values are known precisely, the optimal choice (i.e. the choice that will maximize reward on the long run) is the choice with the maximum expected value: indeed, it is easy to see that if $\mathbb{E}\left[r(s, a_1)\right] > \mathbb{E}\left[r(s, a_2)\right]$, then $\mathbb{E}\left[\sum_n r_n(s, a_1)\right] > \mathbb{E}\left[\sum_n r_n(s, a_2)\right]$ (here and for the next few paragraphs, we will restrict our analysis to the case of single stage tasks, and omit the s input in the reward function). However, in the context of a volatile environment, the agent has no certainty that the reward function has not changed since the last time she visited this state, and she has no precise estimate of the reward distribution. This should motivate her to devote part of her choices to exploration rather than exploitation. In a randomly changing environment, there is no general, optimal balance between the two, as there is no way to know how similar is the environment wrt the last trials.The best thing an agent can do is therefore to update her current policy wrt her current estimate of the uncertainty of the latent state of the environment.

Various policies have been proposed in order to use the Bayesian belief the agent has about its environment to make a decision that maximizes expected rewards in the long run. We review briefly Q-value sampling (QS) [66]. Our framework can also be connected to another algorithm used in the study of animal RL [67, 68], based on the Value of Perfect Information (VPI) [40, 41, 69], which we describe in Appendix E.

QS [66] is an exploration policy based on the posterior predictive probability that an

action value exceeds all the other actions available:

$$p(a) \triangleq p(a = \arg\max_{\mathbf{a}} \mu(\mathbf{a})) = \mathbb{E}_{p(\mu(\mathbf{a}))}\left[p(\mu(a) > \mu(a')\,\forall a' \neg a)\right]. \tag{14}$$

The expectation of Eq 14 provides a clear policy to the agent. The QS approach is compelling in our case: in general, the learning algorithm we propose will produce a trial-wise posterior probability that should, most of the time, be easy to sample from.

The policy dictated by QS is optimal provided that the subject has an equal knowledge of all the options she has. If the environment is only partly explored, the value of some actions may be overestimated (or underestimated), in which case QS will fail to detect that exploration might be beneficial. QS can lead to the same policy when two actions (a_1 and a_2) have similar uncertainty associated with their reward distribution ($\sigma_1 = \sigma_2$) but a different reward ($\mu_1 > \mu_2$), and when one action has a much larger expected reward ($\mu_1 \gg \mu_2$) but also larger uncertainty ($\sigma_1 \gg \sigma_2$) (see [40] for an example). This can be sub-optimal, as the action with the larger uncertainty could lead to a higher (or lower) reward than expected: in this specific case, choosing the action with the largest expected reward should be even more encouraged due to the lack of of knowledge about its true reward distribution, which might be much higher than expected. A strategy to solve this problem is to correct the policy by a factor, the Value of Perfect Information, that reflects the ratio between the potential information gain that will follow the selection of an action with the reward cost of this choice. This approach, and its relationship to our algorithm, is discussed in Appendix E.

Q-Sampling as a Stochastic Process Let us get back to the case of QS, and consider an agent solving this problem using a gambler ruin strategy [70]. We assume that, in the case of a two-alternative forced choice task, this agent has equal initial expectations that either a_1 or a_2 will lead to the highest reward, represented by a start point $z_0 = \zeta/2$, where ζ will be described shortly. The gambler ruin process works as follows: this agent samples a value $\widetilde{r}\,(a_1) \sim p\,(r(a_1)\,|\,x_{<j})$ and a value $\widetilde{r}\,(a_2) \sim p\,(r(a_2)\,|\,x_{<j})$ and assess which one is higher. If $\widetilde{r}\,(a_1)$ beats $\widetilde{r}\,(a_2)$, she computes the number of wins of a_1 until now as $z_1 = z_0 + 1$, and displaces her belief the other way ($z_0 - 1$) if a_2 beats a_1. Then, she starts again and moves in the direction indicated by $\text{sign}(r(a_1) - r(a_2))$ until she reaches one of the two arbitrary thresholds situated at 0 or ζ that symbolize the two actions available. It is easy to see that the number of wins and looses generated by this procedure gives a Monte Carlo sample of $p(r(a_1) > r(a_2)\,|\,x_{<j})$. We show in Appendix F.1 that this process tends to deteministically select the best option as the threshold grows.

So far, we have studied the gambler ruin problem as a discrete process. If the interval between the realization of two samples tends to 0, this accumulation of evidence can be approximated by a continuous stochastic process [70]. When the rewards are normally distributed, as in the present case, this results in a Wiener process, or Drift-Diffusion model (DDM, [71]), since the difference between two normally distributed random variables follows a normal distribution. This stochastic accumulation model has a displacement rate (drift) that is given by

$$d\frac{z}{dt} \sim \mathcal{N}(\mu(a_1) - \mu(a_2), \sigma^2(a_1) + \sigma^2(a_2)) \qquad \text{see [72].}$$

Crucially, it enjoys the same convergence property of selecting almost surely the best option for high thresholds (see Appendix F.2).

Sequential Q-Sampling as a Full-DDM model This simple case of a fixed-parameters DDM, however, is not the one we have to deal with, as the agent does

not know the true value of $\{\boldsymbol{\mu}(a), \boldsymbol{\sigma}^2(a)\, \forall a \in \mathbf{a}\}$, but she can only approximate it based on her posterior estimate. Assuming that the approximate posterior over the latent mean and variance of the reward distribution is a $\mathcal{NG}^{-1}$ distribution, and keeping the original statement $d\frac{X}{dt} = r(a_1) - r(a_2)$, we have

$$
\begin{aligned}
& d\frac{X}{dt} \sim \mathcal{N}(\widetilde{\mu}_1 - \widetilde{\mu}_2, \widetilde{\sigma}_1^2 + \widetilde{\sigma}_1^2) \\
& \text{where} \quad \widetilde{\mu}_i, \widetilde{\sigma}_i^2 \sim \mathcal{N}(\mu_i^\mu, \sigma_i^2/\kappa_i^\mu)\, \mathcal{G}^{-1}(\alpha_i^\sigma, \beta_i^\sigma) \quad \text{for } i = \{1, 2\} \\
& \Leftrightarrow \quad \widetilde{\mu}_1 - \widetilde{\mu}_2 \sim \mathcal{N}(\mu_1^\mu - \mu_2^\mu, \tfrac{\sigma_1^2}{\kappa_1^\mu} + \tfrac{\sigma_2^2}{\kappa_2^\mu}) \\
& \qquad \widetilde{\sigma}_1^2 \sim \mathcal{G}^{-1}(\alpha_1^\sigma, \beta_1^\sigma) \\
& \qquad \widetilde{\sigma}_2^2 \sim \mathcal{G}^{-1}(\alpha_2^\sigma, \beta_2^\sigma)
\end{aligned}
\tag{15}
$$

and where, for the sake of sparsity of the notation, the indices are used to indicate the corresponding action-related variable.

To see how such evidence accumulation process evolves, one can discretize Eq 15: this would be equivalent to sample at each time t a displacement

$$
\Delta x = \Delta t \;\; \widetilde{\xi} \; + \sqrt{\Delta t} \;\; \widetilde{\varsigma} \;\; \epsilon \tag{16}
$$

where Δx stands for $x_t - x_{t-1}$. The drift $\widetilde{\xi}$ in Eq 16 is sampled as the difference between two sampled means $\widetilde{\mu}_1 - \widetilde{\mu}_2$ and the squared noise $\widetilde{\varsigma}^2$ is sampled as the sum of the two sampled variances $\widetilde{\sigma}_1^2 + \widetilde{\sigma}_2^2$. At each time step (or at each trial, quite similarly as we will show), the tuple of parameters $\{\widetilde{\mu}_1, \widetilde{\mu}_2, \widetilde{\sigma}_1^2, \widetilde{\sigma}_2^2\}$ is drawn from the current posterior distribution.

Hereafter, we will refer to this process as the Normal-Inverse-Gamma Diffusion Process, or *NIGDM*.

NIGDM as an exploration rule Importantly, and similarly to QS, this process has the desired property of selecting actions with a probability proportional to their probability of being the best option, thereby favouring exploratory behaviour. We show that the NIGDM behaves like QS in Appendix F.3. There, it is shown (Proposition 3) that, as the threshold grows, the NIGDM choice pattern resembles more and more the QS algorithm. For lower values of the threshold, this algorithm is less accurate than QS (see further discussion of the property of NIGDM for low thresholds in Appendix E).

Cognitive cost optimization under the AC-HAFVF Algorithm 2 summarizes the AC-HAFVF model. This model ties together a learning algorithm - that adapts how fast it forgets its past knowledge on the basis of its assessment of the stability of the environment - with a decision algorithm that makes full use of the posterior uncertainty of the reward distribution to balance exploration and exploitation.

Importantly, these algorithms make time and resource costs explicit: for instance, time constraints can make decisions less accurate, because they will require a lower decision threshold. Fig 2 illustrates this interpretation of the AC-HAFVF by showing how accuracy and speed of the model vary as a function of the variance of the reward estimates: the cognitive cost of making a decision using the AC-HAFVF is high when the agent is uncertain about the reward mean (low κ_j) but has a low expectation of the variance (low β_j). In these situations, the choices were also more random, allowing the agent to explore the environment. Decisions are easier to make when the difference in mean rewards is clearer or when the rewards were more noisy.

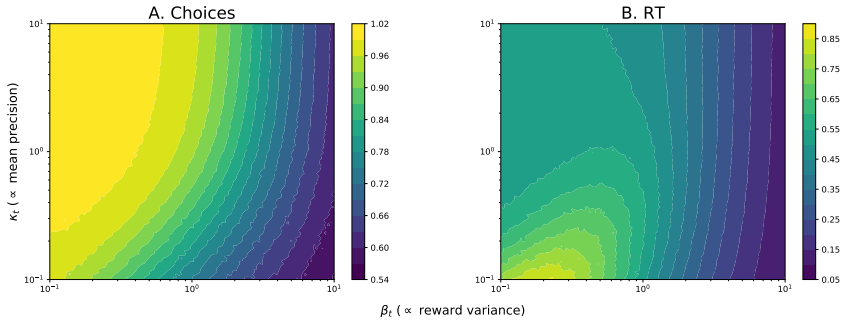


Figure 2. Policies for the AC-HAFVF as a function of reward variance β_j and number of effective observations κ_j, for a fixed value of posterior mean rewards ($\mu_1 = -\mu_2 = 1$), shape parameter $\alpha_1 = \alpha_2 = 3$ threshold $\zeta = 2$, start point $z_0 = \zeta/2$ and $\tau = 0$. **A.** Choices were more random for more noisy reward distributions (i.e. high values of β_j) and for mean estimates with a higher variance (i.e. with a lower number of observations κ_j). **B.** Decisions were faster when the difference of the means was clearer (high κ_j) and when the reward distributions was noisy (high β). Subjects were slower to decide what to do for noisy mean values but precise rewards, reflecting the high cognitive cost of the decision process in these situations.

Besides the decision stage, the computational cost of the inference step can also be determined. To this end, one must first consider that the HAFVF updates the variational posterior using a natural gradient-based [73] approach with a Fisher preconditioning matrix [53, 74, 75]: to learner computes a gradient of the ELBO wrt the variational parameters, then moves in the direction of this gradient with a step length proportional to the posterior variance of the loss function. Because of this, the divergence between the true posterior probability $p(\mathbf{z} \,|\, \mathbf{x}_{\leq j})$ and the prior probability $p(\mathbf{z})$ (i.e. the mixture of the default distribution and previous posterior) conditions directly the expected number of updates required for the approximate posterior to converge to a minimum $D_{KL}\left[\,||\,\right]$ (see for instance [76]). Also, in a more frequentist perspective and at the between-trial time scale, convergence rate of the posterior probability towards the true (if any) model configuration is faster when the KL divergence between this posterior and the prior is small [77]: in other words, in a stable environment, a higher confidence in past experience will require less observations for the same rate of convergence, because it will tighten the distance between the prior and posterior at each time step.

These three aspects of computational cost (for decision, for within-trial inference and for across-trial inference) can justify the choice (or emergence) of lower flexible behaviours as they ultimately maximize the reward rate [67, 78]: indeed, such strategies will lead in a stable environment to faster decisions, to faster inference at each time step and to a more stable and accurate posterior.

2.7 Fitting the AC-HAFVF

So far, we have provided all the necessary tools to simulate behavioural data using the AC-HAFVF. It is now necessary to show how to fit model parameters to an acquired dataset. We will first describe how this can be done in a Maximum Likelihood framework, before generalizing this method to Bayesian inference using variational methods.

Algorithm 2 AC-HAFVF. For simplicity, the NIGDM process has been discretized.

Input: prior belief $\{\boldsymbol{\theta}_0, \boldsymbol{\phi}_0, \boldsymbol{\beta}_0\}$
for $j = 1$ **to** J **do**
  Actor: $NIGDM$
  Input: Start point z_0, threshold ζ, non-decision time τ
  $k \leftarrow 0$
  sample $\widetilde{\mu}_i, \widetilde{\sigma}_i^2 \sim \mathcal{N}(\mu_i^\mu, \sigma_i^2/\kappa_i^\mu)\, \mathcal{G}^{-1}(\alpha_i^\sigma, \beta_i^\sigma)$ for $i = \{1, 2\}$
  while $0 < z_k < \zeta_j$ **do**
    $k \mathrel{+}= 1$
    sample $\delta_z \sim \mathcal{N}\left(\widetilde{\mu}_1 - \widetilde{\mu}_2, \widetilde{\sigma}_1^2 + \widetilde{\sigma}_2^2\right)$
    move $z_k \mathrel{+}= \delta_z$
  end while
  select $a_j \leftarrow \begin{cases} 1 & \text{if } z_k \geq \zeta \\ 2 & \text{otherwise} \end{cases}$
  Get reward $r_j = r(s_j, a_j)$, Observe transition $s_{j+1} = j(s_j, a_j)$

  Critic: $HAFVF$
  $ELBO \leftarrow -\infty$
  while $|\delta_L| \leq 10^{-3}$ **do**
    update $\boldsymbol{\theta}_j(s_j, a_j) = \arg\max_{\boldsymbol{\theta}_j(s_j,a_j)} \mathcal{L}(q(\boldsymbol{\theta}_j(s_j, a_j), \boldsymbol{\phi}_j, \boldsymbol{\beta}_j))$ using CVMP
    update $\{\boldsymbol{\phi}_j, \boldsymbol{\beta}_j\}$ using NCVMP
    $\delta_L \leftarrow \mathcal{L}(q(\boldsymbol{\theta}_j(s_j, a_j), \boldsymbol{\phi}_j, \boldsymbol{\beta}_j)) - ELBO$
    $ELBO \leftarrow \mathcal{L}(q(\boldsymbol{\theta}_j(s_j, a_j), \boldsymbol{\phi}_j, \boldsymbol{\beta}_j))$
  end while
end for

The problem of fitting the AC-HAFVF to a dataset can be seen as a State-Space model fitting problem. We consider the following family of models:

$$\begin{cases} \boldsymbol{\Omega}_{\mathrm{j,n}} = \underbrace{f(\boldsymbol{\Omega}_{\mathrm{j\text{-}1,n}}, \boldsymbol{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{j-1,n})}_{HAFVF} \\ \mathbf{y}_{j,n} \sim \underbrace{\mathbb{E}_{p(\mu_1,\mu_2,\sigma_1^2,\sigma_2^2 \mid \boldsymbol{\Omega}_{\mathrm{j,n}})}\left[Wiener\left(\mu_1 - \mu_2, \sqrt{\sigma_1^2 + \sigma_2^2}, \zeta_n, z_{0n}, \tau_n\right)\right]}_{NIGDM} \end{cases}$$
$$\text{where } \boldsymbol{\Omega}_{\mathrm{j,n}} = \{\boldsymbol{\mu}^\mu, \boldsymbol{\kappa}^\mu, \boldsymbol{\alpha}^\sigma, \boldsymbol{\beta}^\sigma, \boldsymbol{\phi}, \boldsymbol{\beta}\}_{j,n}$$
$$\text{and } \mathbf{y}_{j,n} = \{\mathrm{t}, a\}_{j,n} \tag{17}$$

and $\mathrm{t}_{j,n}$ stands for the reaction time associated with the state-action pair (s, a) of the subject n at the trial j. Unlike many State-Space models, we have made the assumption in Eq 17 that the transition model $\boldsymbol{\Omega}_{\mathrm{j,n}} = f(\boldsymbol{\Omega}_{\mathrm{j\text{-}1,n}}, \boldsymbol{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{j-1,n})$ is entirely deterministic given the subject prior $\boldsymbol{\Omega}_{0,\mathrm{n}}$ and by the observations $\mathbf{x}_{<j}$, which is in accordance with the model of decision making presented in Sec 2.6[1].

Quite importantly, we have made the assumption in Eq 17 that the threshold, the

[1]The Bayesian procedure we will adopt hereafter is formally identical to considering that the drift and noise are drawn according to the rules defined in Sec 2.6, making this model equivalent to:

$$\mathbf{y}_{j,n} \sim Wiener(\xi, \widetilde{\varsigma}, \zeta_n, z_{0n}, \tau_n)$$
$$\xi, \widetilde{\varsigma} \sim f(\boldsymbol{\Omega}_{\mathrm{j\text{-}1,n}}, \boldsymbol{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{j-1,n})$$

non-decision time and the start-point were fixed for each subject throughout the experiment. This is a strong assumption, that might be relaxed in practice. To simplify the analysis, and because it is not a mandatory feature of the model exposed above, we do not consider this possibility here and leave it for further developments.

We can now treat the problem of fitting the AC-HAFVF to behavioural data as two separate sub-problems: first, we will need to derive a differentiable function that, given an initial prior $\mathbf{\Omega}_{0,\mathrm{n}}$ and a set of observations $\mathbf{x}_n$ produces a sequence $\boldsymbol{\theta}_n = \{\mathbf{\Omega}_{\mathrm{j,n}} \, \forall j \in 1 : J\}$, and second (Sec 2.7.1) a function that computes the probability of the observed behaviour given the current variational parameters.

2.7.1 Maximum A Posteriori Estimate of the AC-HAFVF

The update equations described in 2.5.1 enable us to generate a differentiable sequence of approximate posterior parameters $\mathbf{\Omega}_{\mathrm{j,n}}$ given some prior $\mathbf{\Omega}_{0,\mathrm{n}}$ and a sequence of choices-rewards $\mathbf{x}$. We can therefore reduce Eq 17 to a loss function of the form

$$\log p(\mathbf{y}_{j,n} \,|\, \mathbf{x}_{\leq j,n}; \mathbf{\Omega}_{0,\mathrm{n}}, \zeta_n, z_{0n}, \tau_n)$$

whose gradient wrt $\mathbf{\Omega}_{0,\mathrm{n}}$ can be efficiently computed using the chain rule:

$$\begin{aligned} &\nabla_{\mathbf{\Omega}_{0,\mathrm{n}}} \left\{ \log p(\mathbf{y}_{j,n} \,|\, \mathbf{x}; \mathbf{\Omega}_{0,\mathrm{n}}, \zeta_n, z_{0n}, \tau_n) \right\} = \\ &\quad \nabla_{\mathbf{\Omega}_{0,\mathrm{n}}} \{ f(\mathbf{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{\leq j}) \}^T \, \nabla_{\mathbf{\Omega}_{\mathrm{j,n}}} \left\{ \log p(\mathbf{y}_{j,n} \,|\, \mathbf{\Omega}_{\mathrm{j,n}}, \zeta_n, z_{0n}, \tau_n) \right\} \\ &\quad \text{where } \mathbf{\Omega}_{\mathrm{j,n}} = f(\mathbf{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{\leq j}). \end{aligned}$$

Recall that $\nabla_{\mathbf{\Omega}_{0,\mathrm{n}}} \{ f(\mathbf{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{\leq j}) \}^T$ is the jacobian (i.e. matrix of partial derivative) of $\mathbf{\Omega}_{\mathrm{j,n}}$ wrt each of the elements of $\boldsymbol{\theta}_0$ that are optimized, and $\nabla_{\mathbf{\Omega}_{\mathrm{j,n}}} \left\{ \log p(\mathbf{y}_{j,n} \,|\, \mathbf{\Omega}_{\mathrm{j,n}}, \zeta_n, z_{0n}, \tau_n) \right\}$ is the gradient of the loss function (i.e. the NIGDM) wrt the output of $f(\cdot)$.

As the variational updates that lead to the evaluation of $\mathbf{\Omega}_{\mathrm{j,n}}$ are differentiable, the use of VB makes it possible to use automatic Differentiation to compute the Jacobian of $\mathbf{\Omega}_{\mathrm{j,n}}$ wrt $\mathbf{\Omega}_{0,\mathrm{n}}$.

The next step, is to derive the loss function $\log p(\mathbf{y}_{j,n} \,|\, \mathbf{\Omega}_{\mathrm{j,n}}, \zeta_n, z_{0n}, \tau_n)$. In this log-probability density function, the local, trial-wise parameters $\boldsymbol{\chi}_{j,n} \triangleq \left\{ \xi_{j,n}, \lambda^{-1}\left(\sigma^2_{j,n}(a_1)\right), \lambda^{-1}\left(\sigma^2_{j,n}(a_2)\right) \right\}$ have been marginalized out. This makes its evaluation hard to implement with conventional techniques. Variational methods can be used to retrieve an approximate Maximum A Posteriori (MAP) in these cases [79]. The method is detailed in Appendix G. Briefly, VB is used to compute a lower bound ($\ell_{j,n} \leq \log p(\mathbf{y}_{j,n} \,|\, \mathbf{\Omega}_{\mathrm{j,n}}, \zeta_n, z_{0n}, \tau_n)$) to the marginal posterior probability described above for each trial. Instead of optimizing each variational parameters independently, we optimize the parameters $\boldsymbol{\rho}$ of an inference network [80] that maps the current HAFVF approximate posterior parameters $\mathbf{\Omega}_{\mathrm{j,n}}$ and the data $\mathbf{y}_{j,n}$ to each trial-specific approximate posterior. This amortizes greatly the cost of the optimization (hence the name Amortized Variational Inference, AMI), as the nonlinear mapping (e.g. multilayered perceptron) $h(\mathbf{y}_{j,n}; \boldsymbol{\rho})$ can provide the approximate posterior parameters of any datapoint, even if it has not been observed yet. We chose $q_{\boldsymbol{\rho}}(\boldsymbol{\chi}_{j,n} \,|\, \mathbf{y}_{j,n})$ to be a multivariate Gaussian distribution, which leads to the following form of variational posterior:

$$\begin{aligned} q_{\boldsymbol{\rho}}(\boldsymbol{\chi}_{j,n} \,|\, \mathbf{y}_{j,n}) &\triangleq \mathcal{N}(\mu^{\boldsymbol{\chi}_{j,n}}, \Sigma^{\boldsymbol{\chi}_{j,n}}) \\ \text{where } \mu^{\boldsymbol{\chi}_{j,n}}, L^{\boldsymbol{\chi}_{j,n}} &= h(\mathbf{y}_{j,n}; \boldsymbol{\rho}); \boldsymbol{\rho}) \end{aligned} \tag{18}$$

and $L^{\boldsymbol{\chi}_{j,n}}$ is the lower Cholesky factor of $\Sigma^{\boldsymbol{\chi}_{j,n}}$. Another consideration is that, in order to use the multivariate normal approximate posterior, the unbounded variances sample $\lambda^{-1}\left(\sigma^2_{j,n}(a_i)\right)$, $i = 1, 2$ must be transformed to an unbounded space. We used the inverse softplus transform $\lambda \triangleq \log\left(\exp\left(\cdot\right) - 1\right)$, as this function has a bounded gradient, in contrasts with the exponential mapping, which prevents numerical overflow. We found that this simple trick could regularize greatly the optimization process. However, this transformation of the normally distributed $\lambda^{-1}(\cdot)$ variables requires us to correct the ELBO by the log-determinant of the Jacobian of the transform [49], which for the sofplus transform of x is simply $\log|\delta\frac{\lambda(x)}{\delta x}| = -\log\left(1 + \exp(-x)\right)$.

The same transformation can be used for the parameters of $\boldsymbol{\theta}_0$ that are required to be greater than 0 (i.e. all parameters except μ_0), which obviously do not require any log-Jacobian correction.

In Eq 18, we have made explicit that we used the three latent variables: the drift rate and the two action-specific noise parameters. This is due to the fact that, unfortunately, the distribution of the sum of two Inverse-Gamma distributed random variables does not have a closed form formula, making the use of single random variable $\varsigma^2_{j,n} = \sigma^2_{j,n}(a_1) + \sigma^2_{j,n}(a_2)$ challenging, whereas it can be done easily for $\xi_{j,n} = \mu_{j,n}(a_1) - \mu_{j,n}(a_2)$, which is normally distributed (see Eq 15).

The final step to implement a MAP estimation algorithm is to set a prior for the parameters of the model. We used a simple L2 regularization scheme, which consists trivially in a normal prior $\mathcal{N}(0, 1)$ over all parameters, mapped onto an unbounded space if needed.

Algorithm 3 shows how the full optimization procedure.

3 Results

We now present four simulated examples of the (AC-)HAFVF in various contexts. The first example compares the performance of the HAFVF to the HGF (REF) in a simple contingency change scenario. The second example provides various case scenarios in a changing environment, illustrating the trade-off between flexibility and the precision of the predictions (Sec 3.2), including cases where agents fail to adapt to contingency changes following prolonged training in a stable environment, as commonly observed in behavioural experiments [5]. The third example shows that the HAFVF can be efficiently fitted to a RL dataset using the method described in 2.7. The fourth and final example shows how this model behaves in multi-stage environments, and compares various implementations.

3.1 Adaptation to contingency changes and comparison with the HGF

In order to compare the performance of our model to the HGF, we generated a simple dataset consisting of a noisy square-wave signal of two periods of 200 trials, alternating between two normally distributed random variables ($\mathbb{N}(1, 0.33)$ and $\mathbb{N}(-1, 0.33)$). We fitted both a Gaussian HGF and the HAFVF to this simple dataset by finding the MAP parameter configuration for both models. The default configuration of the HGF was used, whereas in our case we put a normal prior of $\mathbb{N}(0, 1)$ on the parameters (with inverse softplus transform for parameters needing positive domains).

This fit constituted the first part of our experiment, which is displayed on the left

Algorithm 3 MAP estimate of AC-HAFVF parameters.

Input: Data $\mathbf{x} = \{r_{j,n}, \mathbf{y}_{j,n}$ for $j = 1$ **to** J, $n = 1$ **to** $N\}$
Initialize: $\boldsymbol{\Omega}_{\mathrm{n}} = \{\boldsymbol{\theta}_0, \boldsymbol{\phi}_0, \boldsymbol{\beta}_0\}_n$ **for** $n = 1$ **to** N and IN[§] weights $\boldsymbol{\rho}$
repeat
  set $L \leftarrow 0$
  for $n = 1$ **to** N **do**
    Set $\widetilde{\nabla}_{\boldsymbol{\rho},\boldsymbol{\Omega}_{0,\mathrm{n}},\zeta_n,z_{0_n},\tau_n} \leftarrow 0$
    for $j = 1$ **to** J **do**
      Learning Step: $HAFVF$
      Get $\boldsymbol{\Omega}_{\mathrm{j,n}} = f(\boldsymbol{\Omega}_{\mathrm{j\text{-}1,n}}, \boldsymbol{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{j-1,n})$
        and Jacobian wrt $\boldsymbol{\Omega}_{0,\mathrm{n}}$: $\nabla_{\boldsymbol{\Omega}_{0,\mathrm{n}}}\{f(\boldsymbol{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{\leq j})\}$ using FAD[*]
      Decision Step: $HAFVF$
      Get ELBO $\ell_{j,n}$ of $\log p(\mathbf{y}_{j,n} \,|\, \boldsymbol{\Omega}_{\mathrm{j,n}}, \zeta_n, z_{0n}, \tau_n)$ using AVI[†]
        and corresponding gradient $\nabla_{\boldsymbol{\rho},\boldsymbol{\Omega}_{\mathrm{j,n}},\zeta_n,z_{0_n},\tau_n}\{\ell_{j,n}\}$ using RAD[‡]
      Increment $L \mathrel{+}= \ell_{j,n}$
      Compute gradient wrt $\boldsymbol{\Omega}_{0,\mathrm{n}}$ using the chain rule:
        $\widetilde{\nabla}_{\boldsymbol{\Omega}_{0,\mathrm{n}}} \mathrel{+}= \nabla_{\boldsymbol{\Omega}_{0,\mathrm{n}}}\{f(\boldsymbol{\Omega}_{0,\mathrm{n}}, \mathbf{x}_{\leq j})\}\, \nabla_{\boldsymbol{\Omega}_{\mathrm{j,n}}}\{\ell_{j,n}\}$
      Increment gradient of DDM and IN parameters
        $\widetilde{\nabla}_{\boldsymbol{\rho},\zeta_n,z_{0_n},\tau_n} \mathrel{+}= \nabla_{\boldsymbol{\rho},\zeta_n,z_{0_n},\tau_n}\{\ell_{j,n}\}$
    end for
    L2-norm regularization:
      $L \mathrel{+}= -\frac{1}{2}\|\{\boldsymbol{\Omega}_{0,\mathrm{n}}, \zeta_n, z_{0n}, \tau_n\}\|_2^2$
      $\widetilde{\nabla}_{\boldsymbol{\Omega}_{0,\mathrm{n}},\zeta_n,z_{0_n},\tau_n} \mathrel{+}= -\{\boldsymbol{\Omega}_{0,\mathrm{n}}, \zeta_n, z_{0n}, \tau_n\}$
  end for
  Perform gradient step $\boldsymbol{\rho}, \boldsymbol{\Omega}_{0,\mathrm{n}}, \zeta_n, z_{0n}, \tau_n \mathrel{+}= \eta\, \widetilde{\nabla}_{\boldsymbol{\rho},\boldsymbol{\Omega}_{0,\mathrm{n}},\zeta_n,z_{0_n},\tau_n}$ for a small η
until Some convergence criterion is met

[§] IN=Inference Network, [*] FAD=Forward Automatic Differenciation, [†] AVI=Amortized Variational Inference, [‡] RAD=Reverse Automatic Differentiation (i.e. backpropagation).

part of Fig 3. Comparing the quadratic approximation of the Maximum Log-model evidence [81] of both models, we found a value of -186.42 for HAVFV and -204.73 for the HGF, making the HAVFV a better model of the data, with a Bayes Factor [82] greater than $8 * 10^7$.

The second part of the experiment consisted in adding to this 400-trial long signal a 1200-trial long signal of input situated at $y = 5$, and simulate learning with the MAP fitted parameter with the whole signal for both algorithms (Fig 3, right part). Our aim was to see how the two models would behave in such case. An optimal agent in such a situation should first account for the surprise associated with the sudden contingency change, and then progressively reduce its expected variance estimate to reflect the steadiness of the environment.

The HGF was unable to exhibit the expected behaviour: it hardly adapted its estimated variance to the contingency change and did not adjust it significantly afterwards. This contrasted with the HAFVF, in which we observed initially an increase in the variance estimate at the point of contingency change (reflecting a high surprise), followed by progressively decreasing variance estimate, reflecting the adaptation of the model to the newly stable environment.

Together, these results are informative of the comparative performance of the two

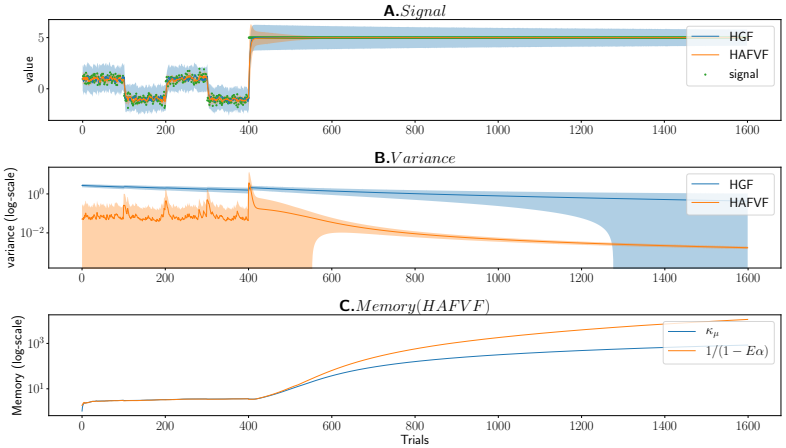


Figure 3. HAFVF and HGF performance on the same dataset. Shaded areas represent the ±3 standard error interval. **A.** Observations, mean and standard error of the mean estimated by both models. **B.** The variance estimates show that the HAFVF adapted better to the variance in the first part of the experiment, reflected better the surprise at the contingency change and adapted successfully its estimate when the environment was highly stable. The HGF, on the contrary, rapidly degenerated its estimate of the variance, and did not show a significant trace of surprise when the contingency was altered. **C.** The value of the effective memory of the HAVFV is represented by the approximate posterior parameter κ_μ, and the maximum memory (efficient memory, see Sec 2.1) allowed by the model at each trial.

algorithms. The Maximum Log-model Evidence was larger for the HAFVF than for the HGF by several orders of magnitude, showing that our approach modelled better the data at hand than the HGF. Moreover, the lack of generalization of the HGF to a simple, new signal shows that this model tended to overfit the data, as can be seen from the estimated variance at the time of the contingency change.

3.2 Learning and flexibility assessment

In the following datasets, we simulated the learning process of four hypothetical subjects differing in their prior distribution parameters $\boldsymbol{\phi}_0, \boldsymbol{\beta}_0$, whereas we kept $\boldsymbol{\theta}_0$ fixed for all of them (Table 1). The choice of the subject parameters was made to generate limit and opposite cases of each expected behaviour. With these simulations, we aimed at showing how the prior belief on the two levels of forgetting conditioned the adaptation of the subject in case of contingency change (CC, **Experiment 1**) or isolated expected event (**Experiment 2**).

In both experiments, these agents were confronted with a stream of univariate random variables from which they had to learn the trial-wise posterior distribution of the mean and standard deviation.

In **Experiment 1**, we simulated the learning of these agents in a steady environment followed by an abrupt CC, occurring either after a long (900 trials) or a short (100 trials) training. The signal $\mathbf{r} = \{r_1, r_2, ..., r_n\}$ was generated according to a Gaussian noise with mean $\mu = 3$ before the CC and $\mu = -3$ after the CC, and a constant standard deviation $\sigma = 1$. Fig 4 summarizes the results of this first simulation. During the training phase, the subjects with a long memory on the first level learned the observation value faster than others. Conversely, the SS subject took a long time to

	Subjects	μ_0	κ_0	α_0	β_0	ϕ_{1_0}	ϕ_{2_0}	β_{1_0}	β_{2_0}
(1)	LL	0	0.1	1	1	4.5	0.5	4.5	0.5
(2)	SL					0.5	4.5	4.5	0.5
(3)	LS					4.5	0.5	0.5	4.5
(4)	SS					0.5	4.5	0.5	4.5

Table 1. This table summarizes the parameters of the beta prior of the two forgetting factors w and b used in Sec 3.2, as well as the initial prior over the mean and variance. A low value of initial number of observations κ_0 was used, in order to instruct learner to have a large prior variance over the value of the mean. Each subject will be referred by its expected memory at the lower and higher level (i.e. L = long, S = short memory). For instance, the subject number 3 (LS) is expected to have a long first-level memory, but a short second-level memory, which should make her more flexible than subject 2 (SL) after a long training, whom has a short first-level memory but a long second-level memory.

	Experiment 1				Experiment 2			
Dataset	LL	SL	LS	SS	LL	SL	LS	SS
1	-1.719	-1.657	-1.62	-1.863	-1.459	-1.652	-1.501	-1.863
2	-1.526	-1.649	-1.519	-1.866	-1.453	-1.648	-1.495	-1.866

Table 2. Average ELBOs for Experiment 1 and 2. Higher ELBOs stand for more probable models.

learn the current distribution. More interesting is the behaviour of the four subjects after the CC. In order to see which strategy reflected best the data at hand, we computed the average of the ELBOs for each model. The winning agent was the Long-Short memory, irrespective of training duration, because it was better able to adapt its memory to the contingency.

The two levels had a different impact on the flexibility of the subjects: the first level indicated how much a subject should trust his past experience when confronted with a new event, and the second level measured the stability of the first level. On the one hand, subjects with a low prior on first-level memory were too cautious about the stability of the environment (i.e. expected volatile environments) and failed to learn adequately the contingency at hand. On the other hand, after a long training, subjects with a high prior on second-level memory tended to over-trust environment stability, compared to subjects with a low prior on second level memory, impairing their adaptation after the CC.

The expected forgetting factors also shed light on the underlying learning process occurring in the four subjects: $\widehat{w}$ grew or was steady until the CC for the four subjects, even for the SL subject which showed a rapid growth of $\widehat{w}$ during the first trials, but failed to reduce it at the CC. In contrast, the LS subject did not exhibit this weakness, but rapidly reduced its expectation over the stability of the environment after the CC thanks to her pessimistic prior belief over b.

In **Experiment 2**, we simulated the effect of an isolated, unexpected event ($r_j = -3$) after long and short training with the same distribution as before. For both datasets, we focused our analysis on the value of the expected forgetting factors $\widehat{w}$ and $\widehat{b}$, as well as the effective memory of the agents, represented by the parameter κ^μ. As noted earlier, the value of $\widehat{w}$ sets an upper bound (the efficient memory) on κ^μ, which represented the actual number of trials kept in memory up to the current trial.

Fig 5 illustrates the results of this experiment. Here, the flexible agents (with a low memory on either the first or second memory level, or both) were disadvantaged wrt the

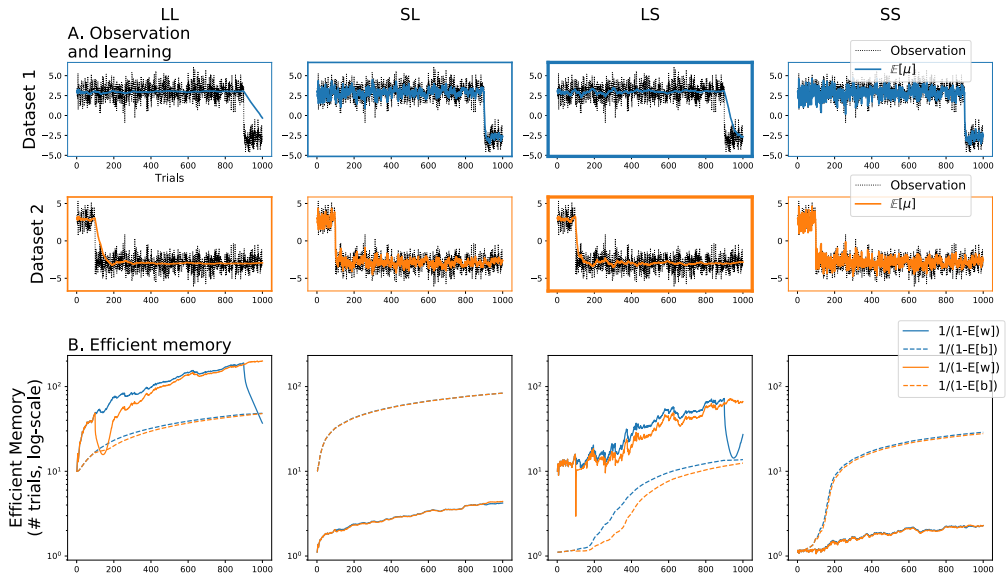


Figure 4. HAFVF predictions after a CC. Each column displays the results of a specific hyperparameter setting. The blue traces and subplots represent the learning in an experiment with a long training, the orange traces and subplots show learning during a short training experiment. **A.** The stream of observations in the two training cases are shown together with the average posterior expected value of the mean μ_j^μ. The box line width illustrates the ranking of the ELBO of each specific configuration for the dataset considered, with bolder borders corresponding to larger ELBOs. For both training conditions, the winning model (i.e. the model that best reflected the data) was the Long-Short memory model. This can be explained by the fact that the first two models trusted too much their initial knowledge after the CC, whereas the Short-Short learner was too cautious. **B.** Efficient memory (defined as $1/(1-\mathbb{E}_q[\cdot])$) for the first level ($\widehat{w}$, plain line) and second level ($\widehat{b}$, dashed line).

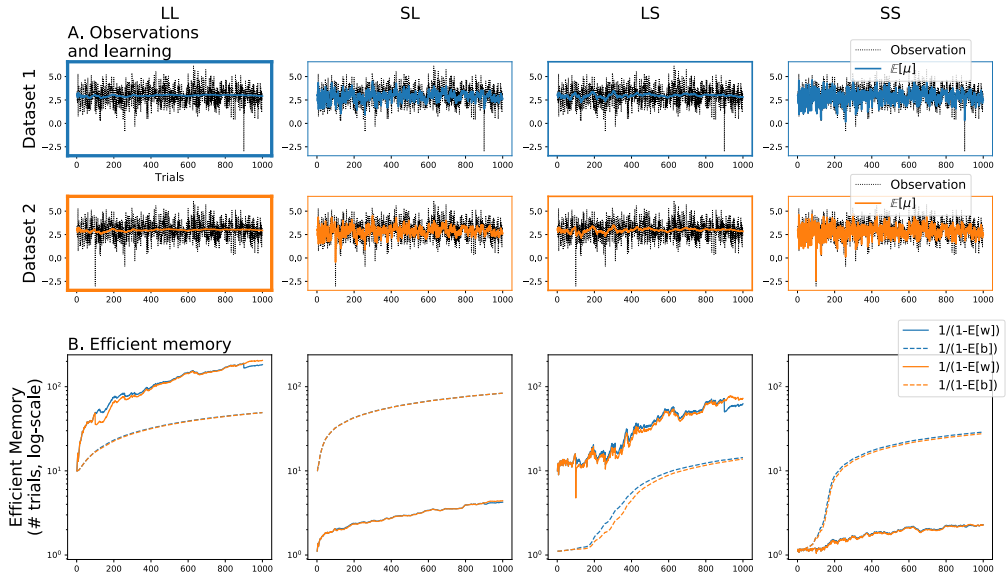


Figure 5. HAFVF predictions after an isolated unexpected event. The figure is similar to Fig 4. Here, the winning model was the one with a high memory on the first and second levels.

low flexibility agent (mostly LL). Indeed, following the occurrence of a highly unexpected observation, one can observe that the LS learner memory dropped after either long or short training. The LL learner, instead, was able to cushion the effect of this outlier, especially after a long training, making it the best learner of the four to learn these datasets (Table 2).

3.3 Fitting the HAFVF to a behavioural task

The model we propose has a large number of parameters, and overfitting could be an issue. To show that inference about the latent variables of the model depicted in Sec 2.7 is possible, we simulated a dataset of 64 subjects performing a simple one-stage behavioural task, that consisted in trying to choose at each trial the action leading to the maximum reward. In any given trial $j \in 1:1000$, the two possible actions (e.g. left or right button press) were associated to different, normally distributed, reward probabilities with a varying mean and a fixed standard deviation $\sigma^2_{1,2} = 1$: for the first action (a_1) the reward had a mean of 0 for the first 500 trials, then switched abruptly to $+2$ for 100 trials and then to -2 for the rest of the experiment. The second action value was identically distributed but in the opposite order and with the opposite sign (Fig 6). This pattern was chosen in order to test the flexibility of each agent after an abrupt CC: after the first CC, an agent discarding completely exploration in favour of exploitation would miss the CC. The second and third CC tested how fast did the simulated agents adapt to the observed CC.

Individual prior parameters $\boldsymbol{\Omega_j} \triangleq \{\mathcal{NG}^{-1}$ prior $\boldsymbol{\theta}_{0,n}$, Beta priors $\boldsymbol{\phi}_{0,n}, \boldsymbol{\beta}_{0,n}\}$ and thresholds ζ_n were generated as follows [2]: we first looked for the L2-regularized MAP

[2]With a negligible loss of generality, the prior mean μ_0 was considered to be equal to 0 for all subjects for both the data generation and the fitting procedures.

estimates of these parameters that led to the maximum total reward:

$$\Omega^*, \zeta^* = \arg\max_{\Omega,\zeta} \prod_{j=1}^{J} p(a_j = \arg\max_{\mathbf{a}_j} r_j | r_{<j}, \mathbf{a}_{<j}, \mathbf{\Omega}, \zeta) p(\mathbf{\Omega}, \zeta)$$

using a Stochastic Gradient Variational Bayes (SGVB) [83] optimization scheme.

We then simulated individual priors centered around this value with a covariance matrix arbitrarily sampled as

$$\Sigma_{\Omega,\zeta} \sim \mathcal{W}_{10}^{-1}\left(\begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & 0.1 \end{bmatrix}\right)$$

where $\mathcal{W}_n^{-1}(\cdot)$ is an inverse-Wishart distribution with n degrees of freedom. This choice of prior lead to a large variability in the values of the AC-HAFVF parameters, except for the NIGDM threshold whose variance was set to a sufficiently low value (hence the 0.1 value in the prior scale matrix) to keep the learning performance high.

This method ensured that each and every parameter set was centered around an unknown optimal policy. This approach was motivated by the desire of preventing any strong constrain on the data generation pattern while keeping a behaviour that was, on average, close to be optimal, as it might be expected with a healthy population. The other DDM parameters, ν_n and τ_n, were generated according to a Gaussian distribution centered on 0 and 0.3, respectively. Importantly, simulated subjects with a performance lower than 70% were rejected and re-sampled to avoid irrelevant parameter patterns.

Learning was simulated according to the Continuous Learning strategy (see Sec 2.5.2), because it was supposed to link more comprehensively the tendency to explore the environment with the choice of prior parameters $\mathbf{\Omega}$. Choices and RT where then generated according to the decision process described in Sec 2.6 using the algorithm described by [84]. Fig 6 shows two examples of the simulated behavioural data.

The behavioural results showed a clear tendency of subjects with large expected variance in action selection to act faster and less precisely than others. This follows directly from the structure of the NIGDM: larger variance of the drift-rate leads to faster but less precise policies. More interesting is the negative correlation between the expected stability and the reward-rate and average reaction time: this shows that the AC-HAFVF was able to encode a form of subject-wise computational complexity of the task. Indeed, large stability expectation leads subjects to trust more their past experience, thereby decreasing the expected reward variance after a long training, but it also leads to a lower capacity to adapt to CCs. For subjects with low expectation of stability, the second level memory was able to instruct the first-level to trust past experience when needed, as the positive correlation between accuracy and upper level memory shows.

Fitting results The fit was achieved with the Adam [85] Stochastic Gradient Descent optimizer with parameters $s = 0.005, \beta_1 = 0.9, \beta_2 = 0.99$, where s decreased with a rate $\sqrt{\lceil i/1000 \rceil}$ where i is the iteration number of the SGD optimizer. We used the following annealing procedure to avoid local minima: at each iteration, the whole set of parameters was sampled from a diagonal Gaussian distribution with covariance matrix $1/i$. This simple manipulation greatly improved the convergence of the algorithm.

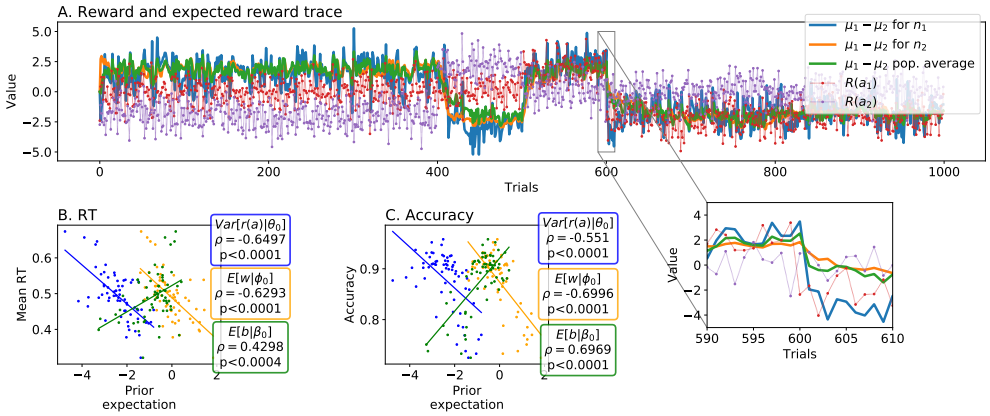


Figure 6. Simulated behavioral results. **A.** The values of the two available rewards are shown with the dotted lines. The average drift rate $\mu_1 - \mu_2$ is shown in plain lines for two selected simulated subjects n_1 and n_2, and population average. Subject n_1 was more flexible than subject n_2 on both the first and the second level, making her more prone to adapt after the CCs, situated at trials 400, 500 and 600. This result is highlighted in the underlying zoomed box. **B.** The subjects' expected variance (blue, log-valued) correlated negatively with the mean RT. The same correlation existed with the expected stability on the first level (orange, logit-valued), but not with the second level, which correlated positively with the average RT (green, logit-valued). Pearson correlation coefficient and respective p-values are shown in rounded boxes. **C.** Similarly, subjects with a higher expected variance and first-level stability had a lower average accuracy. Again, second-level memory expectation had the opposite effect.

The MAP fit of the model is displayed in Fig 7. In general, the posterior estimates of the prior parameters were well correlated with their true value, except for the prior shape parameter α_0. This lack of correlation did not, however, harm much the fit of the variance (see below), showing that the model fit was able to accurately recover the expected prior variability of each subject, which depended on α_0. The NIGDM parameters were highly correlated with their original value.

We also looked at how the true prior expected value of w and b and variance correlated with their estimate from the posterior distribution. All of these correlated well with their generative correspondent, with the notable exception of the expected value of the second-level memory $\mathbb{E}_q[b]$ (Fig 8). This poorer fit of $\mathbb{E}_q[b]$ was, however, compensated by the good correlations of the generative parameters $\boldsymbol{\beta}_0$.

3.4 TD learning with the HAFVF

To study the TD learning described in Sec 2.5.3, we built two similar Markov Decision Processes (MDP) which are described in Fig 9i and Fig 9ii. In brief, they consist of 5 different states where the agent has to choose the best action in order to reach a reward ($r = 5$) delivered once a specific state has been reached (hereafter, we will use "rewarded state" and "rewarded state-action" interchangeably). The task consisted for the agent to learn the current best policy during a 1000-trial experiment where the contingency was fixed to the first MDP during the first 500 trials, and then switched abruptly to the second MDP during the second half of the experiment.

The same prior was used for all subjects: the mean and variance prior was set to $\mu_0 = 0$, $\kappa_0 = 0.5$, $\alpha_0 = 3$, $\beta_0 = 0.5$. The forgetting factors shared the same flat prior

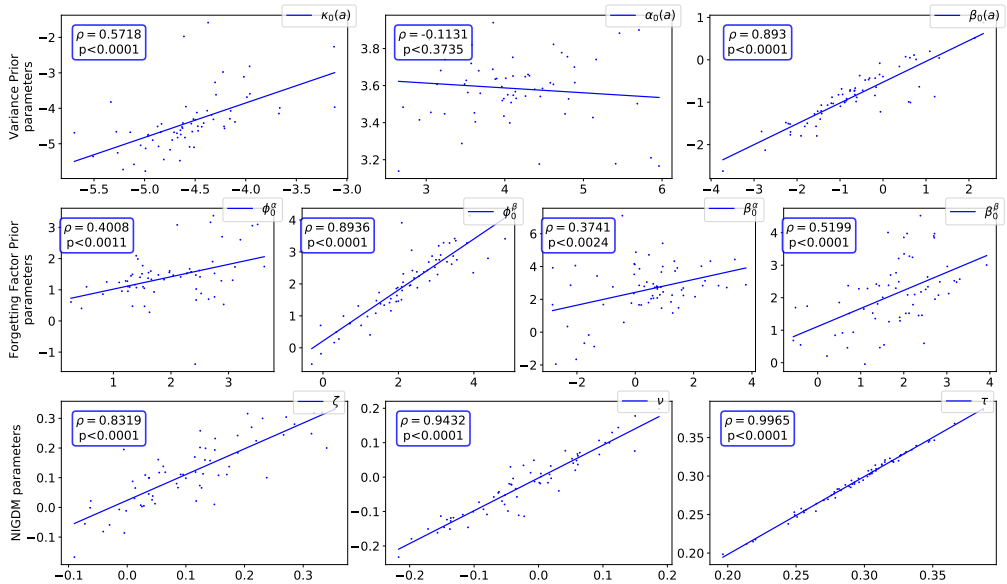


Figure 7. Correlation between the true (x axis) and the posterior estimate (y axis) of the parameters of the prior distributions across subjects. The first row displays the correlations between true value and estimated $\boldsymbol{\theta}_0$. The second row focuses on $\boldsymbol{\phi}_0$ and $\boldsymbol{\beta}_0$, whereas the third row shows the correlations for the NIGDM parameters (threshold, relative start-point and non-decision time). Correlation coefficients and associated p-value (with respect to the posterior expected value) are displayed in blue boxes. All parameters are displayed in the unbounded space they were generated from. Overall, all parameters correlated well with their true value, except for the $\alpha_0(a)$.

$\alpha_0^{w,b} = 1, \beta_0^{w,b} = 1$. The priors on the discounting factor were set to a high value $\alpha_0^{\gamma} = 9, \beta_0^{w,b} = 1$ in order to discourage myopic strategies. The policy prior (see Appendix D) was set to a high value ($\pi_0 = 5.$) in order to limit the impact of initial choices on the computation of the state-action value.

Results are displayed in Fig 10. In both experiments, agents had a similar behaviour during the first phase of the experiment: they both learned well the first contingency by assigning an accurate value to each state-action pair in order to reach the rewarded state more often. As expected, after the contingency change, the agents in experiment (i) took a longer time to adapt than they took to learn the initial contingency, which can be seen from the steeper slope of the reward rate during the first half of the experiment wrt the second. This feature can only be observed if the agent weights its belief by some measure of certainty, which is not modelled in classical, non-Bayesian RL.

In experiment (ii), the CC changes less the environment structure than the CC in experiment (i). The agents were able to take this difference into account: the value of the effective memories dropped less, and so did the reward rate.

An important feature observed in these two experiments is that the expected value of γ adapted efficiently to the contingency: although we used a prior skewed towards high values of γ, its value tended to be initially low as the agents had no knowledge of the various action values. Also, this value increased afterwards, reflecting a gain in predictive accuracy. The drop of $\mathbb{E}_q[w]$ had the effect of pushing $\mathbb{E}_q[\gamma]$ towards its prior, which in this case *increased* the posterior expectation of γ. At the same time, the uncertainty about γ increased, thereby enhancing the flexibility of this parameter.

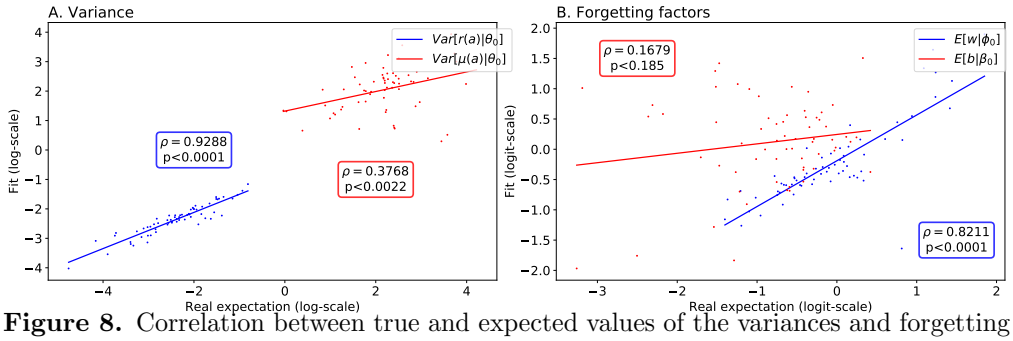
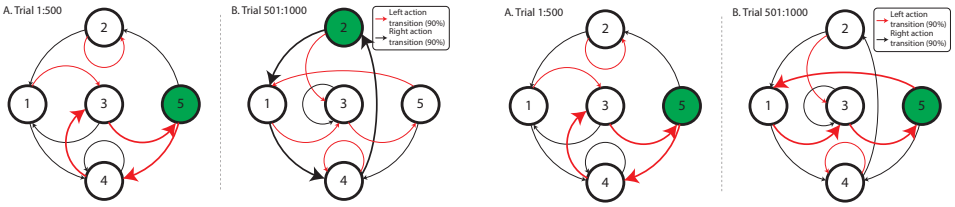


Figure 8. Correlation between true and expected values of the variances and forgetting factors. All the fitted values (y axis) are derived from the expected value of $\boldsymbol{\theta}_0$ (**A.**, variance) and $\{\boldsymbol{\phi}_0, \boldsymbol{\beta}_0\}$ (**B.**, forgetting factors) under the fitted approximate posterior distribution. Each dot represents a different subject. **A.** True (x-axis) to fit (y axis) correlation for the reward (blue) and mean reward (red) variance. Both expected values correlated well with their generative parameter, although the initial number of observations of the gamma prior α_0 did not correlate well with its generative parameter. **B.** True (x-axis) to fit (y axis) correlation for the first (blue) and second (red) level expected forgetting factor.

4 Discussion

In this paper, we propose a new Bayesian Reinforcement Learning (RL) algorithm aimed at accounting for the adaptive flexibility of learning observed in animal and human subjects. This algorithm adapts continuously its learning rate to inferred environmental variability, and this adaptive learning rate is optimal under some assumptions about statistical properties of the environment. These assumptions take the form of prior distributions on the parameters of the latent and mixing weight variables. We illustrate different types of behaviour of the model when facing unexpected contingency changes by taking extreme case scenarios. These scenarios implemented four types of assumptions on the tendency of the environment to vary over time (first-level memory) and on the propensity of this environmental variability to change itself over time (second-level memory). This approach allowed us to reproduce the emergence of inflexible behaviour following prolonged experience of a stable environment, similar to empirical observations in animals. Indeed, it has long been known that extensive training leads to automatization of behaviour, called habits [8, 86–88], or procedural "system 1" actor ([87, 89]), which is characterized by a lack of flexibility (i.e. failure to adapt to contingency changes) and by reduction of computational costs, illustrated by the capacity to perform these behaviours concomitantly to other tasks [90–92]. These automatic types of behaviour are opposed to Goal-Directed behaviours ([**?**, 91, 93]) and share the common feature of being inflexible, either in terms of planning (for Model-Free RL for instance) or in terms of adaptation in general.

Regarding the actor part, we implemented a general, Bayesian decision-making algorithm that reflects in many ways the Full-DDM proposed by [2], as it samples the reward distribution associated to each action and selects at each time step the best option. These elementary decisions are integrated until a decision boundary is reached. Therefore, the actor maps the cognitive predictions of the critic onto specific behavioural outputs such as choice and reaction time (RT). Importantly, this kind of Bayesian evidence accumulation process for decision making is biologically plausible [94, 95], well suited for decision making in RL [**?**, 96–98] and makes predictions that are in accordance with physiological models of learning and decision making [99].



(i) First experiment MDPs. **(ii)** Second experiment MDPs.

Figure 9. Rewarded states are displayed in green. Each action had a probability of 90% to lead to the end state indicated by the red and black arrows (respectively left and right action). The remaining 10% transition probabilities were evenly distributed among the other states. For clarity, the thick arrows show the optimal path the agent should aim to take during the two contingencies. Note that the only difference between experiment (i) and (ii) is the location of the rewarded state after the CC (state 2 for (i) and 5 for (ii)).

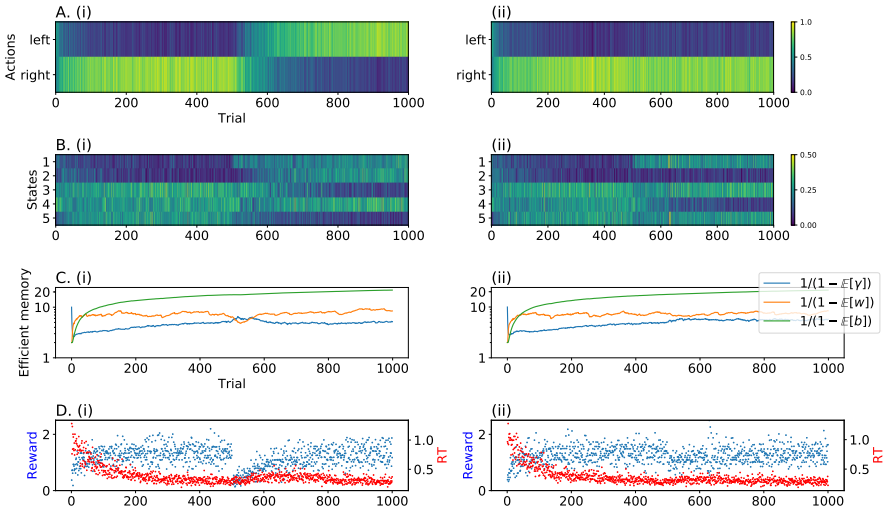


Figure 10. Behavioural results in the first (left, 9i) and second experiments (right, 9ii). **A.** and **B.** Heat plot of the probability of visiting each state and selecting each action for the 64 agents simulated. (i) Agents progressively learned the first optimal actions (left action in state 3-4-5) during the first half of the experiment, then adapted their behaviour to the new contingency (right action in states 1-4-2). (ii) Similarly, in the second experiment, agents adapted their behaviour according to the new contingency (left action in 1-3-5). **C.** Efficient memory on the first and second level, and foreseeing capacity. Since the CC was less important in (ii) than in (i), because the left action in state 3 kept being rewarded, the expected value of w dropped less. The behaviour of the foreseeing capacity ($1/(1 - \mathbb{E}_q[\gamma])$ and, therefore, of the expected value of γ, is indicative of the effect that a CC had on this parameter: when the environment became less stable, $\mathbb{E}_q[\gamma]$ tended to *increase* which had the effect of increasing the impact of future states on the current value. **D.** (i) Reward rate dropped after the CC, whereas the RT increased. The fact that subjects made slower choices after the CC can be viewed as a mark of the increased task complexity caused by the re-learning phase. Along the same line, RT decreased again when the subjects were confident about the structure of the environment. (ii) The CC had also a lower impact on the reward rate and RT in experiment (ii).

Other noteworthy attempts have been made to integrate sampling-based decision-making and RL [100, 101] using the DDM. However, the present work is the first, to our knowledge, to frame the DDM as an optimal, Bayesian decision strategy to maximize long-term utility on the basis on value distributions inferred from a RL algorithm. We show that this RL-DDM association, and especially in the framework of the Full DDM [2], finds a grounded algorithmic justification in a Bayesian perspective, as the resulting policy mimics the one of an agent trying to infer the best decision given its posterior belief about the reward distribution.

Interestingly, under some slight modifications (i.e. assuming that the sampled rewards are not simulated but retrieved from the subject memory), it is similar to the model recently proposed by Bornstein et al. [102, 103]. More specifically, while our decision making scheme used a heuristic based on the asymptotic property of MCMC, the scheme of decision making proposed by Bornstein and colleagues might recall other approximation techniques such as Approximate Bayesian Computation (ABC [104]). Following this approach, data samples are generated according to some defined rule, and only the samples that match the actual previous observations are kept in memory to approximate the posterior distribution or, in our case, to evaluate the option with the greatest reward. This simple trick in the decision making process keeps the stochastic nature of the accumulation process, while directly linking the level of evidence to items retrieved from memory.

One major advantage of our model is that its parameters are easily interpretable as reflecting hidden behavioural features such as trial-wise effective memory, prior and posterior expected stability, etc. We detail how model parameters can be fitted to data in order to recover these behavioural features at the trial, subject and population levels. This approach could be used, for instance, to cluster subjects in high-stability seeking and low-stability seeking sub-populations, and correlate these behaviours to health conditions, neurophysiological measures or training condition (stress, treatment etc.) We show that, for a simulated dataset, the fitted posterior distribution of the parameters correlate well with their original value. Moreover, each of the layers of the model (learning, first level memory and second level memory) have interesting behavioural correlates in terms of accuracy and RT. These results show that the model is identifiable and that there is a low redundancy in the various layers of the model. Moreover, different priors over the expected distribution of rewards and environment stability led also to different outcomes in terms of reward rate and RT: subjects whom assumed large environmental stability were likely to act faster, but also to be less flexible and to gain less on average, than subjects whom assumed high likelihood of contingency changes. This finding also resonates with the habitual learning literature, in which decreased flexibility is also typically associated with short average RT, reflecting lower cognitive cost. The second-level memory had the opposite effect, showing its role of regularizer of the first-level memory: some subjects might indeed be more prone to think that the stability of the environment is a steady feature, hence making them more likely to trust their past learning of the environment stability when facing a CC.

Finally, we extended our work to MDP and multi-stages tasks. More than adding a mere reparametrization of TD learning, we propose a framework in which the time discount factor is considered by the agent as a latent variable: this opens the possibility of studying how animals and humans adapt their long-term / short-term return balance in different conditions of environmental variability. We show in the results that deep CC provoke a reset of the parameters, and lead subjects to erase their acquired knowledge of the posterior value of γ. We also show that this ability to adapt γ to the uncertainty of the environment can also be determinant at the beginning of the task, where nothing is known about the long term return of each action, and where

individuals might benefit of a low expectation of γ that contrasts with the high expectation of subjects that have a deeper knowledge of the environment.

The present model is based on forgetting, which is an important feature in RL that should be differentiated from learning. In signal processing and related fields where online learning is required, learning can be seen as the capacity to build on previous knowledge (or assess a posterior probability distribution in a Bayesian context) to perform inference at the present step. Forgetting, on the contrary, is the capacity to erase the learning to come back at a naive, initial state of learning. The algorithm we propose learns a posterior belief of the data distribution and infers how likely this belief is to be valid in the next time step, on the basis of past environmental stability. This allows the algorithm to decide when and how much to forget its past belief in order to adapt to new contingencies. This feature sets our algorithm apart from previous proposals such as HGF, in which the naive prior looses its importance as learning goes on, and where the learner has no possibility of coming back to his initial knowledge. This lack of capacity to forget implies that the agent can be easily fooled by its past experience, whereas our model is more resistant in such cases, as its point of reference is fixed (which is the common feature of SF algorithms, see above). We have shown in the results that our model outperforms the HGF both in its fitting capability and its capacity to learn new observations with a given prior configuration. The AC-HAFVF should help to flexibly model learning in these contexts, and find correlates between physiological measures, such as dopaminergic signals [105, 106], and precise model predictions in term of memory and flexibility.

The SF scheme we have used, where the previous posterior is compared with a naive prior to optimize the forgetting factor, is widely diffused in the signal processing community [1, 31–33, 36, 38, 107, 108] and finds grounded mathematical justifications for error minimization in recursive Bayesian estimation [109]. However, it is the first time, to our knowledge, that this family of algorithms is applied to the study of RL in humans. We show that the two algorithms (RL and SF) share deep common features: for instance, the HAFVF and other similar algorithms ([52]) can be used with a naive prior $\boldsymbol{\theta}_0$ set to 0, in which case the update equations reduce somehow to a classical Q-learning algorithm ([4] and Appendix B). Another interesting bound between the two fields emerges when the measure of the environment volatility is built hierarchically: an interesting consequence of the forgetting algorithm we propose is that, when observations are not made, the agent erases progressively its memory of past events. This leads to counterfactual learning schedule that favors exploration over exploitation at a rate dictated by the learned stability of the environment (see Appendix C for a devlopement). Crucially, this updating scheme, and the consecutive exploration policy, flows directly from the hierarchical implementation of the SF scheme.

This work provides a tool to investigate learning rate adaptation in behaviour. Previous work has shown, for instance that the process of learning rate adaptation can be decomposed into various components that relate to different brain areas or networks [110]. Nassar and colleagues [111] have also looked at the impact of age on learning rate adaptation, and found that older subjects were more likely to have a narrow expectation of the variance of the data they were observing, impairing thereby their ability to detect true CC. The AC-HAFVF shares many similarities with the algorithm proposed by Nassar and McGuire: it is designed to detect how likely an observation is to be caused by an abrupt CC, and adapts its learning rate accordingly. Also, this detection of CC depends in both models not only on the first and second moment of the observations, but also on their prior average and variability. We think, however, that our model is more flexible and biologically plausible than the model of Nassar and McGuire for three reasons. First, it is fully Bayesian: when fitting the

model, we do not fit an expected variance of the outcome observed, but a prior distribution on this variance. This important difference is likely to predict better the observed data, as the subjects performing an experiment have probably some prior uncertainty about the variability of the outcome they will witness. Second, we considered the steadiness of the environment as another Bayesian estimate, meaning that the subjects will have some confidence (and posterior distribution) in the fact that the environment has truly changed or not. Third, we believe that our model is more general (i.e. less ad hoc) than the model of Nassar, as the general form of Eq 9 encompasses many models that can be formulated using distributions issued from the exponential family. This includes behavioural models designed to evolve in multi-stage RL tasks, such as TD learning or Model-Based RL [112].

It is important to emphasize that the model we propose does not intend to be universal. In simple situations, fitting a classical Q-learning algorithm could lead to similar or even better predictions than those provided by our model. We think, however, that our model complexity makes it useful in situations where abrupt changes occur during the experiment, or where long sequences (several hundreds of trials) of data are acquired. The necessity to account for this adaptability could be determined by comparing the accuracy of the HAFVF model (i.e. the model evidence) to the one obtained from a simpler Bayesian Q-Learning algorithm without forgetting.

The richness and generality of the AC-HAFVF opens countless possibilities of future developments. The habitual/goal-directed duality has been widely framed in terms of Model-Free/Model-Based control separation (e.g. [3, 10, 88, 113–117]). Although we do not model here a Model-Based learning algorithm, we think our work will ultimately help to discriminate various forms of inflexibility, and complete the whole picture of our understanding of human RL: the current implementation of the AC-HAFVF can model a lack of flexibility due to an overconfidence in the volatility of the environment, whereas adding a Model-Based component to the model might help to discriminate a lack of flexibility due to the overuse of a Model-Free strategy[3] that characterize the Model-Free/Model-Based paradigm. In short, implementation of Model-Based RL in an AC-HAFVF context might enrich greatly our understanding of how the balance between Model-Based and Model-Free RL works. This is certainly a development we intend to implement in the near future.

In conclusion, we provide a new Model-Free RL algorithm aimed at modelling behavioural adaptation to continuous and abrupt changes in the environment in a fully Bayesian way. We show that this model is flexible enough to reflect very different behavioural predictions in case of isolated unexpected events and prolonged change of contingencies. We also provide a biologically plausible decision making model that can be integrated elegantly in our learning algorithm, and completes elegantly the toolbox to simulate and fit datasets.

References

1. KULHAVÝ R, ZARROP MB. On a general concept of forgetting. International Journal of Control. 1993;58(4):905–924. doi:10.1080/00207179308923034.

2. Ratcliff R, Rouder JN. Modeling Response Times for Two-Choice Decisions. Psychological Science. 1998;9(5):347–356. doi:10.1111/1467-9280.00067.

[3]Overconfidence in a Model-Free controller means that the subject will need to go through the same sequences of state-action transitions over and over to downweight actions situated early in a sequence. This contrasts with Model-Based control, which can immediately adapt its policy when an action situated far from the current state is devaluated [118].

3. Daw ND, Niv Y, Dayan P. Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. Nature Neuroscience. 2005;8(12):1704–1711. doi:10.1038/nn1560.

4. Moens V. Hierarchical Adaptive Forgetting Variational Filtering. In prep. 2018;.

5. Dickinson a. Actions and Habits: The Development of Behavioural Autonomy. Philosophical Transactions of the Royal Society B: Biological Sciences. 1985;308(1135):67–78. doi:10.1098/rstb.1985.0010.

6. Dickinson A, Balleine BW, Watt A, Gonzalez F, Boakes RA. Motivational control after extended instrumental training. Animal Learning & Behavior. 1995;23(2):197–206. doi:10.3758/BF03199935.

7. Yin HH, Knowlton BJBBJ. The role of the basal ganglia in habit formation. Nature reviews Neuroscience. 2006;7(6):464–76. doi:10.1038/nrn1919.

8. Seger Ca, Spiering BJ. A critical review of habit learning and the Basal Ganglia. Frontiers in systems neuroscience. 2011;5(August):66. doi:10.3389/fnsys.2011.00066.

9. Dezfouli A, Balleine BW. Habits, action sequences and reinforcement learning. European Journal of Neuroscience. 2012;35(7):1036–1051. doi:10.1111/j.1460-9568.2012.08050.x.

10. Gillan CM, Otto aR, Phelps Ea, Daw ND. Model-based learning protects against forming habits. Cognitive, Affective, & Behavioral Neuroscience. 2015;15(3):523–536. doi:10.3758/s13415-015-0347-6.

11. Morris LS, Kundu P, Dowell N, Mechelmans DJ, Favre P, Irvine MA, et al. Fronto-striatal organization: Defining functional and microstructural substrates of behavioural flexibility. Cortex. 2016;74:118–133. doi:10.1016/j.cortex.2015.11.004.

12. Hélie S, Waldschmidt JG, Ashby FG. Automaticity in rule-based and information-integration categorization. Attention, perception & psychophysics. 2010;72(4):1013–1031. doi:10.3758/APP.72.4.1013.

13. MacLeod CM. Half a century of research on the Stroop effect: An integrative review. Psychological Bulletin. 1991;109(2):163–203. doi:10.1037/0033-2909.109.2.163.

14. Clark A. Whatever next? Predictive brains, situated agents, and the future of cognitive science. The Behavioral and brain sciences. 2013;36(3):181–204. doi:10.1017/S0140525X12000477.

15. Friston K, Kiebel S. Predictive coding under the free-energy principle. Philosophical transactions of the Royal Society of London Series B, Biological sciences. 2009;364:1211–1221. doi:10.1098/rstb.2008.0300.

16. Hesselmann G, Sadaghiani S, Friston KJ, Kleinschmidt A. Predictive coding or evidence accumulation? False inference and neuronal fluctuations. PLoS ONE. 2010;5(3):1–5. doi:10.1371/journal.pone.0009926.

17. Friston KJ, Stephan KE, Montague R, Dolan RJ. Computational psychiatry: the brain as a phantastic organ. The Lancet Psychiatry. 2014;1(2):148–158. doi:10.1016/S2215-0366(14)70275-5.

18. Mayrhauser L, Bergmann J, Crone J, Kronbichler M. Neural repetition suppression: evidence for perceptual expectation in object-selective regions. Frontiers in Human Neuroscience. 2014;8(April):1–8. doi:10.3389/fnhum.2014.00225.

19. Limongi R, Silva AM, Góngora-Costa B. Temporal prediction errors modulate task-switching performance. Frontiers in Psychology. 2015;6(August):1–10. doi:10.3389/fpsyg.2015.01185.

20. Kneissler J, Drugowitsch J, Friston K, Butz MV. Simultaneous learning and filtering without delusions: a Bayes-optimal combination of Predictive Inference and Adaptive Filtering. Frontiers in Computational Neuroscience. 2015;9(April):1–12. doi:10.3389/fncom.2015.00047.

21. Mathys C. A Bayesian foundation for individual learning under uncertainty. Frontiers in Human Neuroscience. 2011;5(May):1–20. doi:10.3389/fnhum.2011.00039.

22. Mathys CD, Lomakina EI, Daunizeau J, Iglesias S, Brodersen KH, Friston KJ, et al. Uncertainty in perception and the Hierarchical Gaussian Filter. Frontiers in Human Neuroscience. 2014;8(November):825. doi:10.3389/fnhum.2014.00825.

23. Iglesias S, Mathys C, Brodersen K, Kasper L, Piccirelli M, den Ouden HM, et al. Hierarchical Prediction Errors in Midbrain and Basal Forebrain during Sensory Learning. Neuron. 2013;80(2):519–530. doi:10.1016/j.neuron.2013.09.009.

24. Vossel S, Bauer M, Mathys C, Adams RA, Dolan RJ, Stephan KE, et al. Cholinergic stimulation enhances Bayesian belief updating in the deployment of spatial attention. The Journal of neuroscience : the official journal of the Society for Neuroscience. 2014;34(47):15735–42. doi:10.1523/JNEUROSCI.0091-14.2014.

25. Hauser TU, Iannaccone R, Ball J, Mathys C, Brandeis D, Walitza S, et al. Role of the Medial Prefrontal Cortex in Impaired Decision Making in Juvenile Attention-Deficit/Hyperactivity Disorder. JAMA Psychiatry. 2014;71(10):1165. doi:10.1001/jamapsychiatry.2014.1093.

26. Diaconescu AO, Mathys C, Weber LAE, Daunizeau J, Kasper L, Lomakina EI, et al. Inferring on the Intentions of Others by Hierarchical Bayesian Learning. PLoS Computational Biology. 2014;10(9). doi:10.1371/journal.pcbi.1003810.

27. Brazil IA, Mathys CD, Popma A, Hoppenbrouwers SS, Cohn MD. Representational uncertainty in the brain during threat conditioning and the link with psychopathic traits. Biological Psychiatry: Cognitive Neuroscience and Neuroimaging. 2017;(14):1–7. doi:10.1016/j.bpsc.2017.04.005.

28. Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, Rubin DB. Bayesian Data Analysis, Third Edition (Chapman & Hall/CRC Texts in Statistical Science). CRC; 2013.

29. Doucet A, Johansen AM. A Tutorial on Particle filtering and smoothing: Fiteen years later. The Oxford handbook of nonlinear filtering. 2011;(December 2008):656–705. doi:10.1.1.157.772.

30. Doucet a, De Freitas N, Gordon N. Sequential Monte Carlo Methods in Practice. Springer New York. 2001; p. 178–195. doi:10.1198/tech.2003.s23.

31. Azizi S, Quinn A. A data-driven forgetting factor for stabilized forgetting in approximate Bayesian filtering. In: 2015 26th Irish Signals and Systems Conference (ISSC). vol. 11855. IEEE; 2015. p. 1–6. Available from: http://ieeexplore.ieee.org/document/7163747/.

32. Smidl V, Quinn A. Variational Bayesian Filtering. IEEE Transactions on Signal Processing. 2008;56(10):5020–5030. doi:10.1109/TSP.2008.928969.

33. Smidl V, Gustafsson F. Bayesian estimation of forgetting factor in adaptive filtering and change detection. In: 2012 IEEE Statistical Signal Processing Workshop (SSP). 1. IEEE; 2012. p. 197–200. Available from: http://ieeexplore.ieee.org/document/6319658/.

34. Özkan E, Šmídl V, Saha S, Lundquist C, Gustafsson F. Marginalized adaptive particle filtering for nonlinear models with unknown time-varying noise parameters. Automatica. 2013;49(6):1566–1575. doi:10.1016/j.automatica.2013.02.046.

35. Laar TVD, Cox M, Diepen AV, Vries BD. Variational Stabilized Linear Forgetting in State-Space Models. 2017;(Section II):848–852.

36. Smidl V, Quinn A. Mixture-based extension of the AR model and its recursive Bayesian identification. IEEE Transactions on Signal Processing. 2005;53(9):3530–3542. doi:10.1109/TSP.2005.853103.

37. Masegosa A, Nielsen TD, Langseth H, Ramos-López D, Salmerón A, Madsen AL. Bayesian Models of Data Streams with Hierarchical Power Priors. International Conference on Machine Learning (ICM). 2017;70:2334–2343.

38. Dedecius K, Hofman R. Autoregressive model with partial forgetting within Rao-Blackwellized particle filter. Communications in Statistics: Simulation and Computation. 2012;41(5):582–589. doi:10.1080/03610918.2011.598992.

39. Sutton RS, Barto AG. Introduction to Reinforcement Learning. Learning. 1998;4:1–5. doi:10.1.1.32.7692.

40. Dearden R, Russell S. Bayesian Q-learning. AAAI. 1998;.

41. Dearden R, Dearden R, Friedman N, Friedman N, Andre D, Andre D. Model based Bayesian exploration. Proceedings of the fifteenth Conference on Uncertainty in Artificial Intelligence. 1999;(Howard 1966):150–159.

42. Bishop C. Pattern Recognition And Machine Learning. Jordan M, Kleinberg J, Schölkpf B, editors. Springer-Verlag New York; 2006. Available from: http://www.springer.com/us/book/9780387310732.

43. Jaakkola TS, Jordan MI. A variational approach to Bayesian logistic regression models and their extensions. Aistats. 1996;(AUGUST 2001). doi:10.1.1.49.5049.

44. Jaakkola TS, Jordan MI. Bayesian parameter estimation via variational methods. Statistics And Computing. 2000;10(1):25–37. doi:10.1023/A:1008932416310.

45. Blei DM, Kucukelbir A, McAuliffe JD. Variational Inference: A Review for Statisticians. arXiv. 2016; p. 1–33.

46. Paisley J, Blei D, Jordan M. Variational Bayesian Inference with Stochastic Search. Icml. 2012;(2000):1367—-1374.

47. Salimans T, Kingma DP, Welling M. Markov Chain Monte Carlo and Variational Inference : Bridging the Gap. International Conference on Machine Learning. 2015;.

48. Ranganath R, Tran D, Blei DM. Hierarchical Variational Models. arXiv. 2014; p. 1–9.

49. Rezende DJ, Mohamed S. Variational Inference with Normalizing Flows. Proceedings of the 32nd International Conference on Machine Learning. 2015;37:1530–1538.

50. Blei DM. Variational Inference. CsPrincetonEdu. 2002; p. 1–12. doi:10.16373/j.cnki.ahr.150049.

51. V Smidl AQ. Bayesian estimation of non-stationary AR model parameters via an unknown forgetting factor. In: 3rd IEEE Signal Processing Education Workshop. 2004 IEEE 11th Digital Signal Processing Workshop, 2004. 6. IEEE; 2004. p. 221–225. Available from: `http://staff.utia.cas.cz/smidl/files/publ/taos04.pdfhttp://ieeexplore.ieee.org/document/1437946/`.

52. Smidl V. The Variational Bayes Approach in Signal Processing; 2004. Available from: `http://staff.utia.cz/smidl/Public/Thesis-final.pdf`.

53. Knowles D, Minka TP. Non-conjugate variational message passing for multinomial and binary regression. Nips. 2011; p. 1–9.

54. Bottou L, Peters J, Ch P, Quiñonero-Candela J, Charles DX, Chickering DM, et al. Counterfactual Reasoning and Learning Systems: The Example of Computational Advertising. Journal of Machine Learning Research. 2013;14:3207–3260. doi:10.1145/2740908.2742564.

55. Foerster J, Farquhar G, Afouras T, Nardelli N, Whiteson S. Counterfactual Multi-Agent Policy Gradients. Arxiv. 2017; p. 1–12.

56. Lawrence C, Sokolov A, Riezler S. Counterfactual Learning from Bandit Feedback under Deterministic Logging: A Case Study in Statistical Machine Translation. 2017;.

57. Mischel W, Ebbesen EB, Raskoff Zeiss A. Cognitive and attentional mechanisms in delay of gratification. Journal of Personality and Social Psychology. 1972;21(2):204–218. doi:10.1037/h0032198.

58. Weatherly JN. On several factors that control rates of discounting. Behavioural Processes. 2014;104:84–90. doi:10.1016/j.beproc.2014.01.020.

59. Story GW, Vlaev I, Seymour B, Darzi A, Dolan RJ. Does temporal discounting explain unhealthy behavior? A systematic review and reinforcement learning perspective. Frontiers in behavioral neuroscience. 2014;8(March):76. doi:10.3389/fnbeh.2014.00076.

60. McClure SM. Separate Neural Systems Value Immediate and Delayed Monetary Rewards. Science. 2004;306(5695):503–507. doi:10.1126/science.1100907.

61. Schultz W. Updating dopamine reward signals. Current opinion in neurobiology. 2013;23(2):229–38. doi:10.1016/j.conb.2012.11.012.

62. Bermudez MA, Schultz W. Timing in reward and decision processes. Philosophical transactions of the Royal Society of London Series B, Biological sciences. 2014;369(1637):20120468. doi:10.1098/rstb.2012.0468.

63. Takahashi T. Loss of self-control in intertemporal choice may be attributable to logarithmic time-perception. Medical Hypotheses. 2005;65(4):691–693. doi:10.1016/j.mehy.2005.04.040.

64. Vincent BT. Hierarchical Bayesian estimation and hypothesis testing for delay discounting tasks. Behavior Research Methods. 2015;doi:10.3758/s13428-015-0672-2.

65. Kurth-Nelson Z, Bickel W, Redish AD. A theoretical account of cognitive effects in delay discounting. European Journal of Neuroscience. 2012;35(7):1052–1064. doi:10.1111/j.1460-9568.2012.08058.x.

66. Wyatt J. Exploration and Inference in Learning From Reinforcement; 1998. Available from: `https://www.era.lib.ed.ac.uk/handle/1842/532`.

67. Keramati M, Dezfouli A, Piray P. Speed/accuracy trade-off between the habitual and the goal-directed processes. PLoS computational biology. 2011;7(5):e1002055. doi:10.1371/journal.pcbi.1002055.

68. Viejo G, Khamassi M, Brovelli A, Girard B. Modelling choice and reaction time during instrumental learning through the coordination of adaptive working memory and reinforcement learning. Fourth Symposium on Biology of Decision - Making (SBDM 2014). 2014;9(August). doi:10.3389/fnbeh.2015.00225.

69. Mcallister R, Dziugaite K. Bayesian Reinforcement Learning. 2013;35(March):1–21.

70. Feller W. An Introduction to Probability Theory and Its Applications. Wiley. 1968;2:509. doi:10.2307/1266435.

71. Ratcliff R. A theory of memory retrieval. Psychological Review. 1978;85(2):59–108. doi:http://dx.doi.org/10.1037/0033-295X.85.2.59.

72. Smith PL. Stochastic Dynamic Models of Response Time and Accuracy: A Foundational Primer. Journal of Mathematical Psychology. 2000;44(3):408–463. doi:10.1006/jmps.1999.1260.

73. Amari Si. Natural Gradient Works Efficiently in Learning. Neural Computation. 1998;10(2):251–276. doi:10.1162/089976698300017746.

74. Sato MA. Online Model Selection Based on the Variational Bayes. Neural Comput. 2001;13(7):1649–1681. doi:10.1162/089976601750265045.

75. Hoffman M, Blei DM, Wang C, Paisley J. Stochastic Variational Inference. 2012;(2). doi:citeulike-article-id:10852147.

76. Martens J. New insights and perspectives on the natural gradient method. 2014;.

77. Ghosal S, Ghosh JK, van der Vaart AW. Convergence rates of posterior distributions. The Annals of Statistics. 2000;28(2):500–531. doi:10.1214/aos/1016218228.

78. Zenon A, Solopchuk O, Pezzulo G. An information-theoretic perspective on the costs of cognition. bioRxiv. 2018; p. 208280. doi:10.1101/208280.

79. Moens V, Zenon A. Variational Treatment of Trial-by-Trial Drift-Diffusion Models of Behaviour. bioRxiv. 2017;(2):220517. doi:10.1101/220517.

80. Mnih A, Gregor K. Neural Variational Inference and Learning in Belief Networks. ArXiv statML. 2014;32(October):1–20.

81. Friston K, Mattout J, Trujillo-Barreto N, Ashburner J, Penny W. Variational free energy and the Laplace approximation. NeuroImage. 2007;34(1):220–234. doi:10.1016/j.neuroimage.2006.08.035.

82. Daw ND. Trial-by-trial data analysis using computational models. Attention & Performance XXIII. 2011; p. 1–26. doi:10.1093/acprof:oso/9780199600434.003.0001.

83. Kingma DP, Welling M. Auto-Encoding Variational Bayes. 2013;.

84. Ratcliff R, Tuerlinckx F. Estimating parameters of the diffusion model: Approaching to dealing with contaminant reaction and parameter variability. Psychonomic Bulletin & Review. 2002;9(3):438–481.

85. Kingma DP, Ba JL. Adam: a Method for Stochastic Optimization. International Conference on Learning Representations 2015. 2015; p. 1–15.

86. Wood W, Ruenger D. Psychology of Habit. Annual Review of Psychology. 2015;(September):1–26. doi:10.1146/annurev-psych-122414-033417.

87. Dayan P. Goal-directed control and its antipodes. Neural Networks. 2009;22(3):213–219. doi:10.1016/j.neunet.2009.03.004.

88. Dolan RJ, Dayan P. Goals and habits in the brain. Neuron. 2013;80(2):312–25. doi:10.1016/j.neuron.2013.09.007.

89. Kahneman D, Frederick S. Representativeness Revisited: Attribute Substitution in Intuitive Judgment. In: Gilovich T, Griffin D, Kahneman D, editors. Heuristics and Biases. Cambridge University Press; 2001. p. 49–81. Available from: `http://ebooks.cambridge.org/ref/id/CBO9780511808098A012`.

90. Schneider W, Shiffrin RM. Controlled and Automatic Human Information Processing: I. Detection, Search, and Attention. Psychological Review. 1977;84(1):1–66. doi:10.1037/h0021465.

91. Moors A, De Houwer J, Houwer JD. Automaticity: A Theoretical and Conceptual Analysis. Psychological Bulletin. 2006;132(2):297–326. doi:10.1037/0033-2909.132.2.297.

92. Ashby FG, Crossley MJ. Automaticity and multiple memory systems. Wiley Interdisciplinary Reviews: Cognitive Science. 2012;3(3):363–376. doi:10.1002/wcs.1172.

93. Waldschmidt JG, Ashby FG. Cortical and striatal contributions to automaticity in information-integration categorization. NeuroImage. 2011;56(3):1791–1802. doi:10.1016/j.neuroimage.2011.02.011.

94. Hanes DP, Schall JD. Neural Control of Voluntary Movement Initiation. Science. 1996;274(5286):427–430. doi:10.1126/science.274.5286.427.

95. Roitman JD, Shadlen MN. Response of neurons in the lateral intraparietal area during a combined visual discrimination reaction time task. The Journal of neuroscience : the official journal of the Society for Neuroscience. 2002;22(21):9475–9489. doi:10.1016/S0377-2217(02)00363-6.

96. Soltani A, Wang XJ. Synaptic computation underlying probabilistic inference. Nature Neuroscience. 2010;13(1):112–119. doi:10.1038/nn.2450.

97. Gluth S, Rieskamp J, Buchel C. Deciding When to Decide: Time-Variant Sequential Sampling Models Explain the Emergence of Value-Based Decisions in the Human Brain. Journal of Neuroscience. 2012;32(31):10686–10698. doi:10.1523/JNEUROSCI.0727-12.2012.

98. Rombouts JO, Bohte SM, Roelfsema PR. Neurally Plausible Reinforcement Learning of Working Memory Tasks. Nips. 2012; p. 1–9. doi:10.1371/journal.pcbi.1004489.

99. Smith PL, Ratcliff R. Diffusion and random walk processes. International encyclopedia of the social and behavioral sciences. 2015;6:395–401. doi:10.1016/B978-0-08-097086-8.43037-0.

100. Frank MJ, Gagne C, Nyhus E, Masters S, Wiecki TV, Cavanagh JF, et al. fMRI and EEG Predictors of Dynamic Decision Parameters during Human Reinforcement Learning. Journal of Neuroscience. 2015;35(2):485–494. doi:10.1523/JNEUROSCI.2036-14.2015.

101. Pedersen ML, Frank MJ, Biele G. The drift diffusion model as the choice rule in reinforcement learning. Psychonomic Bulletin & Review. 2017;24(4):1234–1251. doi:10.3758/s13423-016-1199-y.

102. Bornstein AM, Khaw MW, Shohamy D, Daw ND. Reminders of past choices bias decisions for reward in humans. Nature Communications. 2017;8(May 2015):1–9. doi:10.1038/ncomms15958.

103. Bornstein AM, Norman KA. Reinstated episodic context guides sampling-based decisions for reward. Nature Neuroscience. 2017;20(7):997–1003. doi:10.1038/nn.4573.

104. Lintusaari J, Gutmann MU, Dutta R, Kaski S, Corander J. Fundamentals and recent developments in approximate Bayesian computation. Systematic Biology. 2017;66(1):e66–e82. doi:10.1093/sysbio/syw077.

105. Morita K, Kato A. Striatal dopamine ramping may indicate flexible reinforcement learning with forgetting in the cortico-basal ganglia circuits. Frontiers in Neural Circuits. 2014;8(April):1–15. doi:10.3389/fncir.2014.00036.

106. Kato A, Morita K. Forgetting in Reinforcement Learning Links Sustained Dopamine Signals to Motivation. PLoS Computational Biology. 2016;12(10):1–41. doi:10.1371/journal.pcbi.1005145.

107. Kulhavy R, Karny M. Tracking of slowly varying parameters by directional forgetting. Preprints 9ih IFAC Congress. 1984;10:178–183. doi:10.1016/S1474-6670(17)61051-6.

108. Kulhavý R, Kraus FJ. On Duality of Exponential and Linear Forgetting. IFAC Proceedings Volumes. 1996;29(1):5340–5345. doi:10.1016/S1474-6670(17)58530-4.

109. Kárný M. Approximate Bayesian recursive estimation. Information Sciences. 2014;285(1):100–111. doi:10.1016/j.ins.2014.01.048.

110. McGuire J, Nassar M, Gold J, Kable J. Functionally Dissociable Influences on Learning Rate in a Dynamic Environment. Neuron. 2014;84(4):870–881. doi:10.1016/j.neuron.2014.10.013.

111. Nassar MR, Bruckner R, Gold JI, Li SC, Heekeren HR, Eppinger B. Age differences in learning emerge from an insufficient representation of uncertainty in older adults. Nature Communications. 2016;7(May 2015):1–13. doi:10.1038/ncomms11609.

112. Doll BB, Simon DA, Daw ND. The ubiquity of model-based reinforcement learning. Current opinion in neurobiology. 2012;22(6):1075–81. doi:10.1016/j.conb.2012.08.003.

113. Daw ND, Gershman SJ, Seymour B, Dayan P, Dolan RJ. Model-based influences on humans' choices and striatal prediction errors. Neuron. 2011;69(6):1204–15. doi:10.1016/j.neuron.2011.02.027.

114. Wunderlich K, Smittenaar P, Dolan RJ. Dopamine enhances model-based over model-free choice behavior. Neuron. 2012;75(3):418–24. doi:10.1016/j.neuron.2012.03.042.

115. Otto AR, Raio CM, Chiang A, Phelps EA, Daw ND. Working-memory capacity protects model-based learning from stress. Proceedings of the National Academy of Sciences. 2013;110(52):20941–20946. doi:10.1073/pnas.1312011110.

116. Schad DJ, Jünger E, Sebold M, Garbusow M, Bernhardt N, Javadi AH, et al. Processing speed enhances model-based over model-free reinforcement learning in the presence of high working memory functioning. Frontiers in Psychology. 2014;5(December):1–10. doi:10.3389/fpsyg.2014.01450.

117. Kool W, Gershman SJ, Cushman FA. Cost-Benefit Arbitration Between Multiple Reinforcement-Learning Systems. Psychological Science. 2017; p. 095679761770828. doi:10.1177/0956797617708288.

118. Daw ND, Dayan P, Doherty JPO, Gläscher J, Daw ND, Dayan P, et al. States versus rewards: dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. Neuron. 2010;66(4):585–95. doi:10.1016/j.neuron.2010.04.016.

119. Grasman RPPP, Wagenmakers EJ, Van Der Maas HLJ. On the mean and variance of response times under the diffusion model with an application to parameter estimation. Journal of Mathematical Psychology. 2009;53(2):55–68. doi:10.1016/j.jmp.2009.01.006.

120. Wagenmakers EJ, Grasman RPPP, Molenaar PCM. On the relation between the mean and the variance of a diffusion model response time distribution. Journal of Mathematical Psychology. 2005;49:195–204. doi:10.1016/j.jmp.2005.02.003.

121. Grimmett G, Welsh D. Probability, An Introduction. 2nd ed. Oxford University Press; 1986. Available from: `http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20{&}path=ASIN/0198503687`.

Appendix A Normalizing constant of the exponential mixture prior distribution

Let us consider a prior distribution from the exponential family of the form

$$f(\mathbf{z}\,|\,\boldsymbol{\theta}) = h(\mathbf{z})\exp(\boldsymbol{\theta}\,\mathbf{T}(\mathbf{z}) - A(\boldsymbol{\theta})). \tag{19}$$

Assuming that the approximate posterior $q(\mathbf{z}\,|\,\boldsymbol{\theta}_{j-1})$ has the same form as the prior distribution (which arises naturally in the VB scheme we are using), we are interested in deriving

$$\begin{aligned}
Z(w, \boldsymbol{\theta}_{j-1}, \boldsymbol{\theta}_0) &= \int f(\mathbf{z}\,|\,\boldsymbol{\theta}_{j-1})^w f(\mathbf{z}\,|\,\boldsymbol{\theta}_0)^{1-w} d\,\mathbf{z} \\
&= \int \exp\left(w\Big(\boldsymbol{\theta}_{j-1}\,\mathbf{T}(\mathbf{z}) + \log h(\mathbf{z}) - A(\boldsymbol{\theta}_{j-1})\Big) + (1-w)\Big(\boldsymbol{\theta}_0\,\mathbf{T}(\mathbf{z}) + \log h(\mathbf{z}) - A(\boldsymbol{\theta}_0)\Big)\right) \\
&= \int h(\mathbf{z})\exp\left(w\Big(\boldsymbol{\theta}_{j-1}\,\mathbf{T}(\mathbf{z}) - A(\boldsymbol{\theta}_{j-1})\Big) + (1-w)\Big(\boldsymbol{\theta}_0\,\mathbf{T}(\mathbf{z}) - A(\boldsymbol{\theta}_0)\Big)\right) \\
&\times \exp\left(A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right) - A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)\right) d\,\mathbf{z} \\
&= \underbrace{\int h(\mathbf{z})\exp\left(\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)\mathbf{T}(\mathbf{z}) - A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)\right) d\,\mathbf{z}}_{=1} \\
&\times \exp(-wA(\boldsymbol{\theta}_{j-1}) - (1-w)A(\boldsymbol{\theta}_0) + A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)) \\
&= \exp(-wA(\boldsymbol{\theta}_{j-1}) - (1-w)A(\boldsymbol{\theta}_0) + A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)).
\end{aligned} \tag{20}$$

This result simplifies when combined with the numerator of Eq 4:

$$\begin{aligned}
p(\mathbf{z}\,|\,\boldsymbol{\theta}_{j-1}, \boldsymbol{\theta}_0, w) &= \frac{h(\mathbf{z})\exp\left(\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)\mathbf{T}(\mathbf{z}) - wA(\boldsymbol{\theta}_{j-1}) - (1-w)A(\boldsymbol{\theta}_0)\right)}{\exp(-wA(\boldsymbol{\theta}_{j-1}) - (1-w)A(\boldsymbol{\theta}_0) + A(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0))} \\
&= h(\mathbf{z})\exp\left(\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)\mathbf{T}(\mathbf{z}) - A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)\right)
\end{aligned} \tag{21}$$

which has the form of a distribution from the exponential family where the natural parameters of the two parts of the mixture are weighted according to w and $1-w$.

If we are to find an analytical form to the lower bound $\mathcal{L}(q(\mathbf{z}, w)) = \mathbb{E}_{q(\mathbf{z},w)}\left[\log p(\mathbf{x}, \mathbf{z}, w) - \log q(\mathbf{z}, w)\right]$, we also need to compute the expected log-partition function given the approximate posterior $q(\mathbf{z}, w)$: $-\mathbb{E}_{q(\mathbf{z},w)}\left[A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)\right]$. This expectation has usually not a closed-form expression, but it can be approximated efficiently if we consider its second order Taylor expansion. Indeed,

$$\begin{aligned}
A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right) &= A\left(w_0\,\boldsymbol{\theta}_{j-1} + (1-w_0)\,\boldsymbol{\theta}_0\right) \\
&+ \nabla_{w_0} A\left(w_0\,\boldsymbol{\theta}_{j-1} + (1-w_0)\,\boldsymbol{\theta}_0\right)(w - w_0) \\
&+ \frac{\nabla^2_{w_0} A\left(w_0\,\boldsymbol{\theta}_{j-1} + (1-w_0)\,\boldsymbol{\theta}_0\right)}{2}(w - w_0)^2 + \mathcal{O}\big(|w_0 - w|^3\big).
\end{aligned} \tag{22}$$

If we replace w_0 by the expected value of w under the approximate posterior $w_0 \triangleq \mathbb{E}_{q(w)}[w] = \widehat{w}$, and given that $\mathbb{E}_{q(w)}[(w - \widehat{w})] = 0$, the expected value of the log-partition function can be approximated with

$$\begin{aligned}\mathbb{E}_{q(w)}\left[A\left(w\,\boldsymbol{\theta}_{j-1} + (1-w)\,\boldsymbol{\theta}_0\right)\right] \approx\; & A\left(\widehat{w}\,\boldsymbol{\theta}_{j-1} + (1-\widehat{w})\,\boldsymbol{\theta}_0\right) \\ & + \frac{1}{2}\nabla^2_{\widehat{w}} A\left(\widehat{w}\,\boldsymbol{\theta}_{j-1} + (1-\widehat{w})\,\boldsymbol{\theta}_0\right)\mathbb{V}\mathrm{ar}_{q(w)}[w] \\ \text{with } \widehat{w} = \mathbb{E}_{q(w)}[w]\,.\end{aligned} \tag{23}$$

Appendix B HAFVF update equations correspondence in classical RL

The update equations of Eq 12 have a simple correspondence in classical frequentist Q-learning. Let us first consider the case of the mean update in Bayesian Q-Learning without adaptive forgetting where we have

$$\begin{aligned}\mu^\mu_j &= \frac{\kappa^\mu_{j-1}\mu^\mu_{j-1} + \mathbf{x}_j}{\kappa^\mu_{j-1} + 1} \\ &= (1-\eta)\mu^\mu_{j-1} + \eta x_j \\ &\quad \text{where } \eta = \frac{1}{\kappa^\mu_{j-1} + 1} \\ &= \mu^\mu_{j-1} + \eta\underbrace{\left(x_j - \mu^\mu_{j-1}\right)}_{\text{RPE}}.\end{aligned} \tag{24}$$

We can see that the Reward Prediction Error (RPE) of the Rescola-Wagner (RW) update emerges naturally in one considers that the learning rate η is equal to the inverse of the sum of the number of trials observed plus the prior observation belief κ^μ_0 (meaning that the learning rate decreases each time an observation is made by a factor $\frac{\kappa^\mu_0+n}{\kappa^\mu_0+n+1}$).

We can now look at the update of the expected value of the mean reward. It can be reformulated as

$$\begin{aligned}\mu^\mu_j &= \frac{\widehat{\mathrm{w}}\;\kappa^\mu_{j-1}\mu^\mu_{j-1} + (1\text{-}\widehat{\mathrm{w}})\;\kappa^\mu_0\mu^\mu_0 + x_j}{\kappa^\mu_j} \\ &= \frac{\widehat{\mathrm{w}}\;\kappa^\mu_{j-1}}{\kappa^\mu_j}\mu^\mu_{j-1} + \frac{(1-\widehat{\mathrm{w}})\kappa^\mu_0}{\kappa^\mu_j}\mu^\mu_0 + \frac{x_j}{\kappa^\mu_j}.\end{aligned}$$

Without loss of generality, we can consider that the prior mean reward is equal to 0, which simplifies the above equation:

$$\mu^\mu_j = \frac{\widehat{\mathrm{w}}\;\kappa^\mu_{j-1}}{\kappa^\mu_j}\mu^\mu_{j-1} + \frac{x_j}{\kappa^\mu_j}.$$

One can easily see that, in this context, we can recover the classical RW update term:

$$\begin{aligned}\mu_j^\mu &= (1-\eta)\mu_{j-1}^\mu + \frac{\eta}{(1-\widehat{\mathrm{w}})\kappa_{j-1}^\mu + 1} x_j \\ &= \mu_{j-1}^\mu + \eta \underbrace{\left(\frac{x_j}{(1-\widehat{\mathrm{w}})\kappa_{j-1}^\mu + 1} - \mu_{j-1}^\mu \right)}_{\text{RPE}} \\ \text{where } \eta &= \frac{(1-\widehat{\mathrm{w}})\kappa_{j-1}^\mu + 1}{\kappa_j^\mu}.\end{aligned}$$

The current observation x_j is decayed by a factor proportional to the product of the previous effective memory κ_{j-1} and the trust allowed to the prior $1-\widehat{\mathrm{w}}$. This means that the agent learns less trials that need to be discarded (as they are more likely to be generated by the prior distribution than the previous posterior), especially when the agent has a high memory. This account for the fact that an accident is less likely after a long than a short steady training.

Appendix C Counterfactual learning update equations

C.1 Continuous Updating

We consider that the agent adopts some update policy based on a family of inferred update distributions $\widetilde{x}_{j+1} \sim f\left(\boldsymbol{\theta}_j(s,a), \boldsymbol{\theta}_0(s,a), \boldsymbol{\phi}_j, \boldsymbol{\phi}_0, \boldsymbol{\beta}_0\right)$. As our notation suggests, this update distribution is either a function of the previous approximate posterior $\boldsymbol{\theta}_j$, of the prior $\boldsymbol{\theta}_0$ or both.

If $\widetilde{x}_{j+1} \sim f(\boldsymbol{\theta}_j(s,a))$, we are in the presence of an optimistic limit case where the agent does not consider that the environment will change anymore when she has selected an action: once some $r_\tau(s,a)$ with $\tau \in 1:j$ is observed, the resulting posterior predictive distribution is used to update the variational parameters in the next trials until a is selected again (Appendix **??** lists the update equations in this and following cases of learning).

Conversely, the pessimistic model of evolution of $\widetilde{x}_{j+1}$ considers that the upcoming values of $r(s,a)$ are extremely variable, and that the agent should not trust anything more than its broad, initial marginal prior $p(\widetilde{x}_{j+1} \mid \boldsymbol{\theta}_0)$.

A third case of update distribution can be used: the agent can actually consider that the new value of $\widetilde{x}_{j+1}$ would likely be drawn from a mixture of the prior and posterior approximate marginal distributions, similar to the distribution implemented in Eq **??** (and possibly Eq 9):

$$\begin{aligned}\widetilde{x}_{j+1} &\sim p(\widetilde{x}_{j+1} \mid \mathbf{z}) \\ \mathbf{z} &\sim p(\mathbf{z} \mid w(\boldsymbol{\theta}_j(s,a) - \boldsymbol{\theta}_0(s,a)) + \boldsymbol{\theta}_0).\end{aligned} \tag{25}$$

The next section details the update equation in these three cases.

C.2 Approximate posterior updating of non-selected actions

In order to develop the update equations of the variational parameters in the case of counterfactual learning, one must first derive an ELBO for this specific case. Recall

that, if x_{j+1} were observed, the ELBO would have the form:

$$\mathcal{L}_{j+1}(q(\mu,\sigma,w,b)) = \mathbb{E}_{(\mu,\sigma,w,b)}\left[\log p(x_{j+1},\mu,\sigma,w,b|\,\boldsymbol{\theta}_{j+1},\boldsymbol{\theta}_0) - \log q(\mu,\sigma,w,b)\right].$$

In the case of an unobserved datapoint, we take the expected value of the ELBO under some model of the evolution of $x_{j+1} \sim f(\boldsymbol{\theta}_j, \boldsymbol{\theta}_0)$

$$\mathbb{E}_f\left[\mathcal{L}_{j+1}(q(\mu,\sigma,w,b))\right] = \mathbb{E}_f\left[\mathbb{E}_{(\mu,\sigma,w,b)}\left[\log p(x_{j+1},\mu,\sigma,w,b|\,\boldsymbol{\theta}_{j+1},\boldsymbol{\theta}_0) - \log q(\mu,\sigma,w,b)\right]\right].$$

One can easily see that the update of the variational parameters of the action not taken under the optimistic assumption that the environment will not change takes the form of:

$$\begin{aligned}
\mu^\mu_{j+1} &= \frac{\widehat{\mathrm{w}}\ \kappa^\mu_j \mu^\mu_j + (1\text{-}\widehat{\mathrm{w}})\ \kappa^\mu_0 \mu^\mu_0}{\kappa^\mu_{j+1}} + \frac{\widehat{x_{j+1}}}{\kappa^\mu_{j+1}} \\
\kappa^\mu_{j+1} &= \widehat{\mathrm{w}}\ \kappa^\mu_j + (1\text{-}\widehat{\mathrm{w}})\ \kappa^\mu_0 + 1 \\
\alpha^\sigma_{j+1} &= \widehat{\mathrm{w}}\ \alpha^\sigma_j + (1\text{-}\widehat{\mathrm{w}})\ \alpha^\sigma_0 + \frac{1}{2} \\
\beta^\sigma_{j+1} &= \widehat{\mathrm{w}}\ \beta^\sigma_j + (1\text{-}\widehat{\mathrm{w}})\ \beta^\sigma_0 + \\
&\quad \frac{1}{2}\left(\widehat{\mathrm{w}}\ \kappa^\mu_{j+1}(\mu^\mu_{j+1} - \mu^\mu_j)^2 + (1\text{-}\widehat{\mathrm{w}})\ \kappa^\mu_0(\mu^\mu_{j+1} - \mu^\mu_0)^2 + (\widehat{x_{j+1}} - \mu^\mu_{j+1})^2 + \mathbb{V}\mathrm{ar}[x_{j+1}]\right)
\end{aligned}$$

$$\begin{aligned}
\text{where } \widehat{x_{j+1}} &= \mathbb{E}_q\left[\mathbb{E}_p\left[x_{j+1}\right]\right] \\
&= \mu^\mu_j \\
\text{and } \mathbb{V}\mathrm{ar}[x_{j+1}] &= \mathbb{E}_q\left[\mathbb{E}_p\left[x^2_{j+1}\right]\right] - \mathbb{E}_q\left[\mathbb{E}_p\left[x_{j+1}\right]\right]^2 \\
&= \frac{\beta^\sigma_j}{\alpha^\sigma_j - 1} + \frac{\beta^\sigma_j}{\kappa^\mu_j(\alpha^\sigma_j - 1)}.
\end{aligned}$$

Note that the posterior predictive variance of x_{j+1} is equal to the sum of the expected variance and the variance of the mean. A similar update paradigm can be used for the opposite limit case where it is assumed that the distribution of x_{j+1} is totally undetermined (i.e. that it has come back to the marginal prior distribution $p(x_j|\,\boldsymbol{\theta}_0)$):

$$\begin{aligned}
\widehat{x_{j+1}} &= \mathbb{E}_q\left[\mathbb{E}_p\left[x_{j+1}\right]\right] \\
&= \mu^\mu_0 \\
\mathbb{V}\mathrm{ar}[x_{j+1}] &= \mathbb{E}_q\left[\mathbb{E}_p\left[x^2_{j+1}\right]\right] - \mathbb{E}_q\left[\mathbb{E}_p\left[x_{j+1}\right]\right]^2 \\
&= \frac{\beta^\sigma_0}{\alpha^\sigma_0 - 1} + \frac{\beta^\sigma_0}{\kappa^\mu_0(\alpha^\sigma_0 - 1)}.
\end{aligned}$$

Finally, the mixed approach consist in considering that the value x_{j+1} is a weighted

average of the two given the learned mixing coefficient:

$$\widehat{x_{j+1}} = \mu_*^\mu$$
$$\mathbb{V}\mathrm{ar}[x_{j+1}] = \frac{\beta_*^\sigma}{\alpha_*^\sigma - 1} + \frac{\beta_*^\sigma}{\kappa_*^\mu(\alpha_*^\sigma - 1)}$$

where

$$\mu_*^\mu = \frac{\widehat{\mathrm{w}}\ \kappa_j^\mu \mu_j^\mu + (1\text{-}\widehat{\mathrm{w}})\ \kappa_0^\mu \mu_0^\mu}{\kappa_*^\mu}$$
$$\kappa_*^\mu = \widehat{\mathrm{w}}\ \kappa_j^\mu + (1\text{-}\widehat{\mathrm{w}})\ \kappa_0^\mu$$
$$\alpha_*^\sigma = \widehat{\mathrm{w}}\ \alpha_j^\sigma + (1\text{-}\widehat{\mathrm{w}})\ \alpha_0^\sigma$$
$$\beta_*^\sigma = \widehat{\mathrm{w}}\ \beta_j^\sigma + (1\text{-}\widehat{\mathrm{w}})\ \beta_0^\sigma + \frac{1}{2}\left(\widehat{\mathrm{w}}\ \kappa_j^\mu(\mu_*^\mu - \mu_j^\mu)^2 + (1\text{-}\widehat{\mathrm{w}})\ \kappa_0^\mu(\mu_*^\mu - \mu_0^\mu)^2\right).$$

Using this update scheme, the agent erases his memory of the posterior distribution at a rate dictated by the stability of the environment, and $\boldsymbol{\theta}_j \to \boldsymbol{\theta}_0$ as $j \to \infty$. Together with the decision algorithm presented in 2.6, this result shows that, as the posterior distributions broadens and approaches the initial prior, the likelihood of choosing this action will also increase because the expected random noise of the evidence accumulation process will also increase.

C.3 Delayed approximate posterior updating

Another approach for the agent to consider the evolution of the environment when selecting an action whose outcome has not been seen for a long time is to simulate the expected waning of the previous update across the elapsed interval of time.

Let us consider the case where the agent has chosen an action (e.g. left) at the trial j, and then the opposite action (right) for n trials. We assume that, during these n trials, she has been updating only the value of the right action, leaving the approximate posterior over the distribution parameters of the value of the left action untouched. When selecting left at the trial $j+n$, she considers that the weight of the posterior component of the prior distribution has decreased exponentially for n trials. The prior then looks like:

$$\frac{p(\mathbf{z} \mid \boldsymbol{\theta}_{j-n})^{w^n} p(\mathbf{z} \mid \boldsymbol{\theta}_0)^{1-w^n}}{Z}$$

In order to compute the NCVMP update of the approximate posterior parameters over w, we then need to compute the expected value of w^n. If $q_j(w)$ is set to be a beta distribution with parameters $\boldsymbol{\phi}_j = \{\phi_{1j}, \phi_{2j}\}$, we have

$$\mathbb{E}_{q_j(w)}\left[w^n\right] = \frac{\Gamma(\phi_{2j})\Gamma(\phi_{1j} + n)}{B(\phi_{1j}, \phi_{2j})\Gamma(\phi_{1j} + \omega_{2j} + n)} \tag{26}$$

where $B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$. The expected value of the squared decay w^{2n} (used to compute the variance in the expected value of the second order Taylor expansion of the log-partition function, see A) can be computed readily with the same formula by replacing n by $2n$.

Unlike the previous method, the shape parameter of the gamma distribution over the variance does not need to be greater than 1 here, as we do not need to simulate the variance of the non-selected action value. However, this is true only for the single stage-case, as the multi-stage case also bases its value updates on expected squared values of rewards, thereby requiring $\alpha_0^\sigma > 1$ too.

Appendix D Discount factor inference

D.1 Model specification

The main idea we will try to develop here is that the discount factor γ can be considered as a latent random variable, so that inference is made about its value in order to get the most precise estimate of the posterior predictive distribution of the long term return of an action $p(V(s,a) \,|\, x_{\leq j})$.

For practical reasons, it is easier to model separately the factorized distribution of the current, immediate state-action reward $p(r(s,a))$ and the discounted value of the future state $p(\gamma v(s' \,|\, s,a))$. In this notation, $\gamma v(s' \,|\, s,a)$ is a random variable representing the discounted long-term value of taking the action a in state s minus the value of the reward received while performing this action $r(s,a)$:

$$V(s,a) = r(s,a) + \gamma v(s' \,|\, s,a)$$

Following this logic, we define the following factorized distribution:

$$p\big(r(s,a), v(s' \,|\, s,a)\big) = p\big(r(s,a) \,|\, \mu^r(s,a), \sigma^r(s,a)\big)\, p\big(v(s' \,|\, s,a) \,|\, \mu^v(s,a), \sigma^v(s' \,|\, s,a), \gamma\big)$$

where $r(s,a)$ is normally distributed and $v(s' \,|\, s,a)$ is distributed according to the following modified Gaussian distribution:

$$p\left(v(s' \,|\, s,a) \,|\, \mu^v(s,a), \sigma^v(s,a), \gamma\right) = \gamma \frac{\exp\left\{\frac{\left(\mu^v(s,a) - \gamma v(s' \,|\, s,a)\right)^2}{2\sigma^v(s,a)^2}\right\}}{\sqrt{2\pi}\,\sigma^v(s,a)}.$$

It becomes clear that this distribution integrates to 1 over $v(s' \,|\, s,a)$, and that it corresponds to a Normal distribution over $\gamma v(s' \,|\, s,a)$. Therefore, the probability distribution of $V(s,a) = r(s,a) + \gamma v(s' \,|\, s,a)$ can be written as:

$$p(V(s,a)) = \mathcal{N}(\mu^r(s,a) + \mu^v(s,a),\, \sigma^r(s,a)^2 + \sigma^v(s,a)^2).$$

This joint probability distribution encodes, for each state-action pair the probability distribution of the associated long-term, discounted return.

We naturally define the prior probability of the discounting factor $\gamma \in [0;1]$ at $j = 1$ as a Beta distribution with parameters $\mathrm{B}(\alpha_0^\gamma, \beta_0^\gamma)$, and a $\mathcal{NG}^{-1}$ prior over the Gaussian parameters $\{\mu^r(s,a), \sigma^r(s,a), \mu^v(s,a), \sigma^v(s,a)\}$.

D.2 Learning from observed transitions and rewards

We now tackle the question of how to estimate the value of the state that has been reached $v(s' \,|\, s,a)$. Indeed, this value, contrary to the reward, is not directly observed but must be inferred from the state of arrival s'. We consider the following state-value inference model:

$$\begin{aligned} v(s' \,|\, s,a) &= \mathbb{E}_{\pi(a' \,|\, s')}\left[V(s',a')\right] \\ &= \mathbb{E}_{\pi(a' \,|\, s')}\left[r(s',a') + v(s' \,|\, s,a)\right]. \end{aligned} \tag{27}$$

Eq 27 stated that the agent assumes that the expected value of the next state is equal to the average of the various state-action values available, weighted by the probability of selection the corresponding action (the observed policy $\pi(a' \,|\, s')$). We assume that the agent does not have a direct access to the complete description of the policy (meaning that, whereas she can make a decision efficiently, she cannot retrieve the summary

statistics of this policy). This assumption is in accordance with the decision process described in 2.6. This policy can, however, be learned similarly to $r(s,a)$ using a multinomial distribution with a Dirichlet (or Beta in the case of Two-Alternative Forced Choice tasks) prior and approximate posterior: at each trial, the agent observes the action she has performed, and update her belief of the current policy.

Let us now consider how the expected log-joint probability of $v(s' \mid s,a)$ would appear in the ELBO if this value was observed under the mean-field assumption:

$$\mathbb{E}_{q_{j+1}(\mu^v(s,a),\sigma^v(s,a)^2,\gamma)}\left[\log p(v(s' \mid s,a)|\mu^v(s,a),\sigma^{v2},\gamma)\right] = -\frac{1}{2}\mathbb{E}_{q_{j+1}(\sigma^v(s,a)^2)}\left[\log \sigma^v(s,a)^2\right] - \frac{1}{2}\mathbb{E}_{q_{j+1}(\mu^v(s,a),\sigma^v(s,a)^2)}\left[\frac{(\gamma v(s' \mid s,a) - \mu^v(s,a))^2}{\sigma^v(s,a)^2}\right] + \mathbb{E}_{q_{j+1}(\gamma)}\left[\log \gamma\right] + \text{cst.} \tag{28}$$

Once the agent reaches the next state $\widehat{s'}$, inference of the value $\mathbb{E}\left[v(s' \mid s,a) \mid s' = \widehat{s'}\right]$ can be achieved by using the posterior predictive expectation of this formula:

$$\begin{aligned}
\mathbb{E}\left[v(s' \mid s,a) \mid s' = \widehat{s'}\right] &= \mathbb{E}_{p(v|x_{\leq j})}\left[v(\widehat{s'} \mid s,a)\right] \\
&\approx \mathbb{E}_{q_j(\mathbf{z})}\left[\mathbb{E}_{p(V,\pi \mid \mathbf{z})}\left[\mathbb{E}_{\pi}\left[V(\widehat{s'},a')\right]\right]\right] \\
&= \mathbb{E}_{q_j(\mathbf{z})}\left[\mathbb{E}_{p(r,v,\pi \mid \mathbf{z})}\left[\mathbb{E}_{\pi}\left[r(\widehat{s'},a') + \gamma v(s''|s',a')\right]\right]\right] \\
&= \mathbb{E}_{q_j(\mathbf{z})}\left[\mathbb{E}_{p(r,v,\pi \mid \mathbf{z})}\left[\sum_{a' \in A} \pi(a' \mid \widehat{s'})(r(\widehat{s'},a') + \gamma v(s''|\widehat{s'},a'))\right]\right] \\
&= \sum_{a' \in a_1,a_2} \frac{\delta_j^{\pi}(\widehat{s'},a')}{\sum_{a \in A} \delta_j^{\pi}(\widehat{s'},a)}(\mu_j^r(\widehat{s'},a') + \mu_j^v(\widehat{s'},a'))
\end{aligned} \tag{29}$$

where A is the set of actions available, and where we indexed with j each variational parameter to emphasize the fact that this expectation is made with respect to the previous posterior belief.

Eq 28 being linear and quadratic wrt $v(s' \mid s,a)$, we also need to solve the expected value $\mathbb{E}\left[v(s' \mid s,a)^2 \mid s' = \widehat{s'}\right]$:

$$\begin{aligned}
&\mathbb{E}_{p(v|x_{\leq j})}\left[v(s' \mid s,a)^2\right] \approx \\
&\quad \mathbb{E}_{q_j(\mathbf{z})}\left[\mathbb{E}_{p(r,v,\pi \mid \mathbf{z})}\left[\Big(\sum_{a' \in A} \pi(a' \mid s')(r(s',a') + \gamma v(s''|s',a'))\Big)^2\right]\right].
\end{aligned} \tag{30}$$

The expression in Eq 30 can be computed iteratively (Algorithm 4):

Algorithm 4 $\mathbb{E}_{p(v(s' \mid s,a)|x_{\leq j})}\left[v(s' \mid s,a)^2 | s' = \widehat{s'}\right]$ iterative computation

Input: Factorized approximate posterior distribution q_{j-1}, state reached at trial j $\widehat{s'}$
Results: Posterior predictive estimate $\nu = v(s' \mid s,a)^2$
$\nu = 0$
for $a \in A$ **do**
  for $a' \in A$ **do**
    if $a == a'$ **then**
      $\nu \mathrel{+}= \mathbb{E}_{q_{j-1}}\left[\pi(a|s')^2\right] \times$ // Eq 35
        $\left(\mathbb{E}_{q_{j-1}}\left[\mathbb{E}_p\left[r(s',a)^2\right]\right] + \mathbb{E}_{q_{j-1}}\left[\mathbb{E}_p\left[(\gamma v(s''|s',a))^2\right]\right]\right.$ // Eq 32,Eq 34
        $\left.+2\mathbb{E}_{q_{j-1}}\left[\mathbb{E}_p\left[r(s',a)\right]\right]\mathbb{E}_{q_{j-1}}\left[\mathbb{E}_p\left[\gamma v(s''|s',a)\right]\right]\right)$ // Eq 31,Eq 33
    else
      $\nu \mathrel{+}= \mathbb{E}_{q_{j-1}}\left[\pi(a|s')\pi(a' \mid s')\right] \times$ // Eq 36
        $\left(\mathbb{E}_p\left[r(s',a)\right] + \mathbb{E}_{q_{j-1}}\left[\mathbb{E}_p\left[\gamma v(s''|s',a)\right]\right]\right)\times$ // Eq 31,Eq 33
        $\left(\mathbb{E}_p\left[r(s',a')\right] + \mathbb{E}_{q_{j-1}}\left[\mathbb{E}_p\left[\gamma v(s''|s',a')\right]\right]\right)$ // Eq 31,Eq 33
    end if
  end for
end for

using the following equivalences

$$\mathbb{E}_{q_{j-1}}\left[\mathbb{E}_{p(r|\mu^r,\sigma^r)}\left[r\right]\right] = \mu^r_{j-1} \tag{31}$$

$$\mathbb{E}_{q_{j-1}}\left[\mathbb{E}_{p(r|\mu^r,\sigma^r)}\left[r^2\right]\right] = {\mu^r_{j-1}}^2 + \frac{\beta^r_{j-1}}{(\alpha^r_{j-1}-1)\kappa^r_{j-1}} + \frac{\beta^r_{j-1}}{(\alpha^r_{j-1}-1)} \tag{32}$$

$$\mathbb{E}_{q_{j-1}}\left[\mathbb{E}_{p(v|\mu^v,\sigma^v,\gamma)}\left[\gamma v\right]\right] = \mu^v_{j-1} \tag{33}$$

$$\mathbb{E}_{q_{j-1}}\left[\mathbb{E}_{p(v|\mu^v,\sigma^v,\gamma)}\left[(\gamma v)^2\right]\right] = {\mu^v_{j-1}}^2 + {\sigma^v_{j-1}}^2 \tag{34}$$

$$\mathbb{E}_{q_{j-1}}\left[\pi(a|s)^2\right] = \left(\frac{\delta^{\pi(a|s)}_{j-1}}{\delta_0}\right)^2 + \frac{\delta^{\pi(a|s)}_{j-1}(\delta_0 - \delta^{\pi(a|s)}_{j-1})}{\delta_0^2(\delta_0+1)} \tag{35}$$

$$\mathbb{E}_{q_{j-1}}\left[\pi(a|s)\pi(a' \mid s)\right] = \frac{\delta^{\pi(a|s)}_{j-1}\delta^{\pi(a' \mid s)}_{j-1}}{\delta_0^2} - \frac{\delta^{\pi(a|s)}_{j-1}\delta^{\pi(a' \mid s)}_{j-1}}{\delta_0^2(\delta_0+1)} \tag{36}$$

$$\text{with } \delta_0 = \sum_{a' \in A} \delta^{\pi(a' \mid s)}$$

where we have dropped several state-action indices for clarity.

We can now take the expectation of Eq 28 under Eq 29 and Eq 30.

This formula has the advantage that the learning of the current distribution of $V(s,a)$ takes into account the uncertainty about the policy, about the reward distribution at the next trial and about the long-term discounted return of the actions in the next state.

Appendix E VPI

VPI [40, 41, 67–69] solves this problem by giving a positive bonus to each action value corresponding to the potential payoff of selecting this action given the current uncertainty about its value distribution. Formally, this bonus is computed as the

posterior expectation of the gain of performing an action given the belief we have of the value taken by the other actions:

$$VPI(a) = \int p(\mu(a) \,|\, x_{<j}) Gain_a(\mu(a))\, d\,\mu(a)$$
$$\text{where } Gain_a(\mu(a)) = \begin{cases} \mathbb{E}\left[\mu(a_2)\right] - \mu(a) & \text{if } a = a_1 \text{ and } \mu(a) < \mathbb{E}\left[\mu(a_2)\right] \\ -\mathbb{E}\left[\mu(a_1)\right] + \mu(a) & \text{if } a \neq a_1 \text{ and } \mu(a) > \mathbb{E}\left[\mu(a_1)\right] \\ 0 & \text{otherwise} \end{cases} \tag{37}$$

and where we have used the convention that a_1 and a_2 are the actions with the highest and second highest reward respectively. Eq E shows that the bonus of a given action is proportional to the expected gain of discovering that this action leads to a higher reward than all the others when it was thought to be sub-optimal, plus the expected gain of discovering that this action is sub-optimal when it was thought to be optimal.

Interestingly, low threshold of the NIGDM 2.6 favor choices that would have been also favored by the VPI approach: if an action has currently the highest estimate, it will be more encouraged if its variance is wide than if it is narrow, and conversely for punished action with a high variance.

Moreover, the evidence accumulation process can be enriched to incorporate the expected gain of performing an action: as the agent samples the means of the two action values given its current belief, she can add the difference in the gain bonuses computed as in Eq E. The resulting process can still be modelled as a Wiener process using the Stochastic Gradient Variational Bayes approach described in Sec 2.7.

Appendix F NIGDM properties

F.1 Discrete convergence property

Importantly, as for every Monte Carlo estimate, this estimate becomes more accurate (i.e. less entropic) as the expected number of iterations (absorption time) grows. In the previous example, this happens when the distance between the absorbing boundaries (threshold) is larger, in which case the agent is more likely to select the optimal option. To see this, consider the case where $p\left(r(a_1) \,|\, x_{<j}\right)$ and $p\left(r(a_2) \,|\, x_{<j}\right)$ are both Gaussian distributions with a known mean and variance, and where $\mu(a_1) > \mu(a_2)$. We then introduce the following result:

Lemma 1 *The probability of hitting the upper boundary ζ tends to 1 as the threshold grows, and the average displacement $\widehat{\delta z}$ concomitantly approaches the true probability that a_1 is more rewarded than a_2:*
$\widehat{\delta z} = \mathbb{E}_{p(r(a_1) \,|\, x_{<j}), p(r(a_2) \,|\, x_{<j})}\left[z - z_0\right] \to p(r(a_1) > r(a_2) \,|\, x_{<j})$.

F.2 Continuous convergence property

The DDM has interesting properties that make it a good candidate to model decision making in Bayesian Reinforcement Learning. We first introduce the following result, which is a generalization of Lemma 1 to the continuous case:

Proposition 1 *If the DDM parameters (threshold ζ, relative start point $\nu \triangleq z_0/\zeta$, drift rate ξ and noise ς) are fixed within and between trials, the probability of hitting the boundary pointed by the drift approaches 1 as the distance between the boundaries grows.*

To show this, one can first start by showing that the Error Rate probability

$$P = \frac{\exp(-2\xi\zeta\nu/\varsigma^2) - \exp(-2\xi\zeta/\varsigma^2)}{1 - \exp(-2\xi\zeta/\varsigma^2)} \tag{38}$$

is monotonically increasing wrt ξ. This can be easily seen from the differential equation of this formula. Next, since $\xi \frac{dP}{d\xi} = \zeta \frac{dP}{d\zeta}$, and since $\frac{dP}{d\xi} > 0 \, \forall \xi \in \nabla$, then $\text{sign}\left(\frac{dP}{d\zeta}\right) = \text{sign}(\xi) \, \forall \xi \in \nabla$, $\zeta \in \nabla^+$ and, therefore, P is monotonically increasing wrt ζ if $\xi > 0$ and decreasing if $\xi < 0$.

F.3 NIGDM similarity to QS

In order to show that the NIGDM mimics a QS algorithm for high thresholds, we need to consider example choices and reaction times generated by this process.

To this end, we first introduce an important result:

Proposition 2 *If a data collection is generated according to a classical Wiener Diffusion process where the drift and squared noise are sampled* ***at each trial*** *following some distribution* $\widetilde{\xi}, \widetilde{\varsigma}^2 \sim p(\xi, \varsigma^2 \,|\, \boldsymbol{\theta}_j)$*, then this dataset cannot be distinguished from a dataset where the drift and noise are sampled similarly* ***within*** *trials.*

This can be proven trivially by considering the moment generating function of these two distributions. For the trial-by-trial process, we have:

$$\mathbb{E}\left[\mathrm{t}, a \,|\, \xi_j, \varsigma_j, \zeta\right] = \sum_{i \in \{1,2\}} e^{la_i} \int e^{l\mathrm{t}} p(\mathrm{t}, a_i \,|\, \xi_j, \varsigma_j, \zeta, \tau, \nu) d\,\mathrm{t}\Big|_{l=1} \tag{39}$$

where t and a are the reaction time and choice, respectively. Now, as ξ and ς are unknown to us, we marginalize over their prior distribution (i.e. the current approximate posterior of $\boldsymbol{\theta}_j$):

$$\mathbb{E}\left[\mathrm{t}, a \,\middle|\, \boldsymbol{\theta}_j, \zeta\right] = \sum_{i \in \{1,2\}} e^{lc_i} \int e^{l\mathrm{t}} \iint p\left(\mathrm{t}, a_i \,|\, \zeta, \xi_j, \varsigma_j\right) p\left(\xi_j, \varsigma_j^2 \,|\, \boldsymbol{\theta}_j\right) dv_j \, d\varsigma_j^2 \, d\mathrm{t}\Big|_{l=1}.$$

One can see that this expression can be re-arranged into the moment generating function of the within-trial process. Because the two distributions have the same moments and the moment generating function exist for the DDM [119, 120], the two distributions are equal [121].

Proposition 2 is important because it states that, if we are able to generate from the NIGDM or fit this probability distribution to a dataset under the assumption of a between-trial variability, then this fit will also be valid in the case of a within-trial variability.

We will show in Sec 2.7 how this model can be optimized in a Maximum Likelihood and a Bayesian context.

We can now use Proposition 1 and Proposition 2 to show that this process does not automatically hit the current best estimate when the boundaries are distant enough from each other. During some trials, the difference between the sampled means $\widetilde{\mu}_1$ and $\widetilde{\mu}_2$ will have a sign opposite to the sign of the current estimate of $\mu_1^\mu - \mu_2^\mu$, and the agent will choose the action with the lowest value. More formally:

Proposition 3 *Under the NIGDM, the proportion of choices* $p(a_1)$ *tends to the posterior predictive probability* $p(\mu_1 > \mu_2 \,|\, x_{<j})$ *as the threshold tends to infinity.*

This can be shown by considering Proposition 2 and Proposition 1 in the fixed-parameters case, and then considering that the expected probability of hitting the positive (or similarly the negative) boundary has the following distribution when ζ tends to infinity:

$$\lim_{\zeta\to\infty} \mathbb{E}\left[p(a_1)\,|\,\boldsymbol{\theta}_j,\zeta\right] = \iiiint \mathbb{1}\left[\text{sign}(\mu_1-\mu_2)=+1\right]\times \tag{40}$$

$$p\left(\mu_1,\mu_2,\sigma_1,\sigma_2\,|\,\boldsymbol{\theta}_j\right)\,d\mu_1\,d\mu_2\,d\sigma_1^2\,d\sigma_2^2$$

$$= p(\mu_1>\mu_2\,|\,\boldsymbol{\theta}_j)$$

$$\text{where } \boldsymbol{\theta}_j = \{\boldsymbol{\mu}^\mu,\boldsymbol{\kappa}^\mu,\boldsymbol{\alpha}^\sigma,\boldsymbol{\beta}^\sigma\} \tag{41}$$

where $\mathbb{1}[x]$ is an operator that takes the value of 1 if its condition x is met and zero otherwise.

Appendix G Sampled Drift Optimisation

Under the assumption that decisions are made according to the scheme described in 2.6, the HAFVF can be optimized using a simple, informative model. Indeed, we can express the First Passage Time probability density function (and similarly the cumulative distribution function) as

$$p(\mathbf{y}_j\,|\,\boldsymbol{\theta}_j,\zeta,\nu,\tau) = \int p(\mathbf{y}_j\,|a_j,\mu_j,w_j,\tau_j,\sigma_j^2)\,p(\mu_j,\sigma_j^2|\,\boldsymbol{\theta}_j)d\mu_j,\sigma_j^2. \tag{42}$$

The central idea of the Sampled Drift Optimisation technique is to use the joint distribution (and not the marginal) in a VB framework. Considering that $\boldsymbol{\theta}_j = f(\boldsymbol{\theta}_0, x_{<j})$, we can write the posterior probability of the DDM parameters at the trial j as:

$$p(\boldsymbol{\theta}_0,\mu_j,\sigma_j^2|\,\mathbf{y}_j,\zeta,\nu,\tau) = \frac{p(\mathbf{y}_j\,|\zeta,\nu,\tau,\mu_j,\sigma_j^2)p(\mu_j,\sigma_j^2|f(\boldsymbol{\theta}_0,x_{<j}))}{p(\mathbf{y}_j,\zeta,\nu,\tau)}. \tag{43}$$

Let us now consider the posterior predictive probability density of the difference between the value of two action values at a specific trial. In the case of a single step task (i.e. without temporal discounting), the posterior predictive distribution of the reward difference is:

$$\begin{aligned} p(R_1-R_2|x_{1:j-1}) &\approx \int p(R_1-R_2|\,\mathbf{z}_j)q(\mathbf{z}_j\,|\,\boldsymbol{\theta}_{j-1})d\,\mathbf{z}_j \\ &\quad \text{with } R_i = r(s,a_i), \qquad \mathbf{z}_j = \{\mu_1^r,\mu_2^r,{\sigma_1^r}^2,{\sigma_2^r}^2\} \\ &= \iint p(\phi+R_2|\mu_1^r,{\sigma_1^r}^2)p(R_2|\mu_2^r,{\sigma_2^r}^2)dR_2\,q(\mathbf{z}_j\,|\,\boldsymbol{\theta}_{j-1})d\,\mathbf{z}_j. \\ &\quad \text{where } \phi = R_1-R_2 \\ &= \int p(\phi|\mu_1^r-\mu_2^r,{\sigma_1^r}^2+{\sigma_2^r}^2)q(\mathbf{z}_j\,|\,\boldsymbol{\theta}_{j-1})d\,\mathbf{z}_j \end{aligned} \tag{44}$$

We see that we can express the posterior predictive probability of the difference of the Reward values as a normal distribution of mean $\mu^r-\mu^r$ and standard deviation $\sqrt{\sigma^{r2}+\sigma^{r2}}$, marginalized over these parameters given their posterior value at the present trial. We can plug the result of 44 into 42 and then into 43 to get:

$$p(\boldsymbol{\theta}_0\,|\,\mathbf{y}_j) = \frac{p(\mathbf{y}_j\,|\zeta,\nu,\tau,\mathbf{z}_j)p(\mathbf{z}_j\,|f(\boldsymbol{\theta}_0,x_{<j})p(\zeta,\nu,\tau)}{p(\mathbf{y}_j)}. \tag{45}$$

This model is easy to optimize using SGVB provided that we can estimate the Jacobian $\nabla_{\boldsymbol{\theta}_0} f(\boldsymbol{\theta}_0, x_{<j})$.

It can also be applied to the discounted value $V(s,a)$ in multistep tasks:

$$\begin{aligned} p\left(V(s,a) - V(s,a')|x_{1:j-1}\right) &\approx \int p\left(V(s,a) - V(s,a')|\,\mathbf{z}_j\right) q(\mathbf{z}_j\,|\,\boldsymbol{\theta}_{j-1}) d\,\mathbf{z}_j \\ &= \iint p\left(\phi + V(s,a')|\mu_1^R + \mu_1^v, {\sigma_1^R}^2 + {\sigma_1^v}^2\right) \times \\ &\quad p\left(V(s,a')|\mu_2^R + \mu_2^v, {\sigma_2^R}^2 + {\sigma_2^v}^2\right) dr(s,a')\, q(\mathbf{z}_j\,|\,\boldsymbol{\theta}_{j-1}) d\,\mathbf{z}_j \quad . \\ \text{where } \phi &= V(s,a) - V(s,a') \\ &= \int p\left(\phi \Big| \sum_{u\in\{r,v\}} \mu_1^u - \mu_2^u, \sum_{u\in\{r,v\}} \sum_{a\in A} {\sigma_a^u}^2\right) q(\mathbf{z}_j\,|\,\boldsymbol{\theta}_{j-1}) d\,\mathbf{z}_j \end{aligned} \tag{46}$$

Acknowledgments

This work was performed at the Institute of Neuroscience (IoNS) of the Université catholique de Louvain (Brussels, Belgium); it was supported by grants from the ARC (Actions de Recherche Concerté es, Communauté Francaise de Belgique), from the Fondation Médicale Reine Elisabeth (FMRE), and from the Fonds de la Recherche Scientifique (F.R.S.-FNRS) to A.Z. A.Z. was a Senior Research Associate supported by INNOVIRIS and is currently 1st grade researcher for the french National Center for Scientific Research (CNRS).