# Sequential Bayesian Learning: A Cognitive Neuroscience Framework

#### Robert T. Lange \*

Einstein Center for Neurosciences Berlin www.rob-lange.com robert.lange17@imperial.ac.uk Miro Grundei Neurocomputation and Neuroimaging Unit Free University Berlin miro.grundei@fu-berlin.de

Sam Gijsen Neurocomputation and Neuroimaging Unit Free University Berlin samgijsen@gmail.com

#### **Abstract**

This report summarizes efforts to make Sequential Bayesian Learning accessible to the general neuroscience community. More specifically, we provide a framework for modeling the surprise of a Bayesian agent who observes a discrete-time and discrete-event sequence of trials. In order to do so we introduce a sampling process for generating sequences of trials based on a graphical model. We derive expressions for update, filtering as well as incorporate such into commonly used surprise measures.

### 1 Introduction

The world is inherently sequential due to its temporal nature. Hence, our own survival relies on making sense of temporally as well as spatially ordered sensory data. Bayesian methods thereby provide a probabilistic framework for integrating sensory information over time. More specifically, the *Bayesian Brain Hypothesis* (BBH) postulates that the brain updates its posterior belief based on integrating its prior distribution (or previous posterior) with new likelihood evidence from the most recent sensory percept. Furthermore, the *Free Energy Principle* (FEP) provides an extension to this more general hypothesis. It postulates that every self-organizing organism seeks to minimize surprise, i.e. free energy. Based on the log model evidence decomposition into the negative variational free energy and Kullback-Leibler (KL) divergence, this requires a Variational Inference (VI) scheme for approximate Bayesian model inversion.<sup>2</sup> By iteratively minimizing the free energy term, one is able to obtain an ever better approximation to the log model evidence, on which we can subsequently perform Bayesian model comparison.

In this piece of work we introduce a general paradigm for generating feature- as well as temporally-dependent sequences of trials based on a simple graphical model. Furthermore, we outline how one is able to model agents that learn the data-generating process in a sequential fashion. Based on the

<sup>\*</sup>This report got drafted during a lab rotation during the Winter of 2018/2019 and connects with an EEG study modeling mismatch-negativity in the somatosensory cortex. For further algorithmic details and replication please view https://github.com/RobertTLange/SequentialBayesianLearning.

<sup>&</sup>lt;sup>2</sup>Original work on VI originates from classical spin glass models in statistical physics. In the machine learning (ML) community the negative free energy term has been better known as the evidence lower bound (ELBO). Hence, the FEP community refers to the minimization of the free energy, while the ML community, on the other hand, speaks about the maximization of the ELBO. In order to avoid confusion: The two are operationally equivalent.

resulting posterior updating rules, we are able to compute different surprise measures which have recently become prominent in cognitive computational neuroscience literature.

The following document is structured as follows: First, we introduce the necessary background required in Bayesian modeling in order for us to introduce the special case of Sequential Bayesian Learning (SBL).

#### 1.1 Bayesian Reasoning

Probabilistic methods allow one to elegantly formulate this form of belief updating in terms of a simple computational heuristic: Bayes' theorem. Compared to traditional frequentist statistical methods it accounts for the innate uncertainty associated with the statistical relationship between measured observations  $o \in \mathbb{R}^d$  and the hidden/unobserved/latent state  $s \in \mathbb{R}^d$ .

At time t, the agent combines his prior over the distribution of the hidden state  $s_t$  with the likelihood of the observed state  $o_t$ :

$$p(s_t|o_t) = \frac{p(o_t, s_t)}{p(o_t)} = \frac{p(o_t|s_t)p(s_t)}{\int_{-\infty}^{\infty} p(o_t|s_t)p(s_t)ds_t} \propto p(o_t|s_t)p(s_t)$$

The computational procedure can often be formulated as a precision-weighted prediction error correction. The updated posterior then forms the prior distribution for time t+1:

$$p(s_{t+1}) \coloneqq p(s_t|o_t)$$

The beauty lies within the computational simplicity and its wide applicability.<sup>3</sup>

#### 1.2 Bayesian Operations: Filtering, Smoothing, Decoding and Evaluation

• Filtering:  $p(s_t|o_1,\ldots,o_t)$ 

• Smoothing:  $p(s_t|o_1,\ldots,o_T)$ 

• Decoding:  $arg \max_{s_1,\ldots,s_T} p(s_1,\ldots,s_T|o_1,\ldots,o_T)$ 

• Evaluation:  $p(o_1, \ldots, o_T)$ 

#### 1.3 Surprise

$$PS(o_t) := -\ln p(o_t|s_t)$$

$$BS(o_t) \coloneqq KL(p(s_t)||p(s_t|o_t))$$

$$CS(o_t) := KL(p(s_t)||\hat{p}(s_t|o_t))$$

### 1.4 Conjugacy

Conjugacy describes a simple mathematical relationship between the prior probability distribution and the likelihood: The updated posterior distribution which combines prior and likelihood information follows the same (differently parametrized) distribution as the prior distribution.

Popular conjugate pairs include the following:

Conjugate Prior	Likelihood	Posterior
Beta	Bernoulli	Beta
Dirichlet	Categorical	Dirichlet
Univ. Gaussian/Inverse Gamma	Univ. Gaussian	Univ. Gaussian/Inverse Gamma
Multiv. Gaussian/Inverse Wishart	Multiv. Gaussian	Multiv. Gaussian/Inverse Wishart

<sup>&</sup>lt;sup>3</sup>The theorem follows directly from the product and sum rule of probability.

In the following we will quickly exercise a derivation for the Beta-Bernoulli (Binomial) case:

### 1.5 Approximate Bayesian Inference

As soon as we diverge from the simple conjugate-pair case, Bayesian inference faces severe computational challenges. There is no longer a closed-form expression for the posterior distribution, since the integration needed to compute the normalizing constant becomes computationally infeasible.

Instead, one has to revert to an approximate estimation of the posterior. The two most common frameworks for doing so are Markov Chain Monte Carlo (MCMC) and Variational Inference (VI) methods.

#### 2 **A General Data Generating Process**

Throughout this text we will make use of a simple and general data-generation process. We differentiate between a simple first-order Markov paradigm in which a latent state  $s_t$  follows a Markov chain and emits observation  $o_t$  which also depend on its precessor  $o_{t-1}$ .

#### First-Order Markov Dependencies

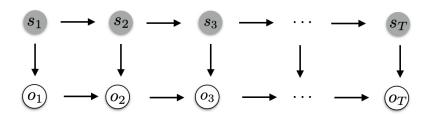


Figure 1: Graphical Model of Data-Generating Process with 1st order Markov Dependency.

$$p(s_{1:T}, o_{1:T}) = p(s_1)p(o_1|s_1) \prod_{t=2}^{T} p(s_t|s_{t-1})p(o_t|o_{t-1}, s_t)$$

- State space:  $s \in \mathcal{S} = \{1, \dots, K\}$
- Observation space:  $o \in \mathcal{O} = \{1, \dots, M\}$
- Initial state distribution:  $p(s_1) = \{\frac{1}{K}, \dots, \frac{1}{K}\} \in [0, 1]^K, \sum_{i=1}^K p(s_1 = j) = 1$
- Initial obs. distribution:  $p(o_1|s_1) = p(o_1) = \{\frac{1}{M}, \dots, \frac{1}{M}\} \in [0, 1]^M, \sum_{i=1}^M p(o_1 = j) = [0, 1]^M$
- Transitions:  $p(s_t|s_{t-1}) = \mathbf{A} \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$

- 
$$\mathbf{A}_{ij} = p(s_t = j | s_{t-1} = i); \sum_{j=1}^{|\mathcal{S}|} \mathbf{A}_{ij} = 1 \ \forall i = 1, \dots, |\mathcal{S}|$$
 and  $\mathbf{A}_{ij} \geq 0 \ \forall i, j = 1, \dots, |\mathcal{S}|$ 

- Emissions: $p(o_t|o_{t-1},s_t) = \mathbf{B} \in \mathbb{R}^{|\mathcal{O}| \times |\mathcal{O}| \times |\mathcal{S}|}$ 
  - $\mathbf{B}_{ijk} = p(o_t = j | o_{t-1} = i, s_t = k) := p_{i|i}^{(k)}$
  - $-\sum_{j=1}^{|\mathcal{O}|} \mathbf{B}_{ijk} = 1 \ \forall i = 1, \dots, |\mathcal{O}|, k = 1, \dots, |\mathcal{S}|$  $-\mathbf{B}_{ijk} \ge 0 \ \forall i, j = 1, \dots, |\mathcal{O}|, k = 1, \dots, |\mathcal{S}|$

#### 2.2 Second-Order Markov Dependencies

$$p(s_{1:T}, o_{1:T}) = p(s_1)p(o_1|s_1)p(s_2|s_1)p(o_2|s_2, o_1)\prod_{t=3}^{T}p(s_t|s_{t-1}, s_{t-2})p(o_t|o_{t-1}, o_{t-2}, s_t)$$

- State space:  $s \in \mathcal{S} = \{1, \dots, K\}$
- Observation space:  $o \in \mathcal{O} = \{1, \dots, M\}$
- Initial state distribution:  $p(s_1) = p(s_2|s_1) = \{\frac{1}{K}, \dots, \frac{1}{K}\} \in [0, 1]^K, \sum_{i=1}^K p(s_1 = j) = 1$
- Initial obs. distribution:  $p(o_1|s_1) = (o_2|o_1, s_1) = \{\frac{1}{M}, \dots, \frac{1}{M}\} \in [0, 1]^M, \sum_{i=1}^M p(o_1 = 0)$ (i) = 1
- Transitions:  $p(s_t|s_{t-1}) = \mathbf{A} \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}| \times |\mathcal{S}|}$

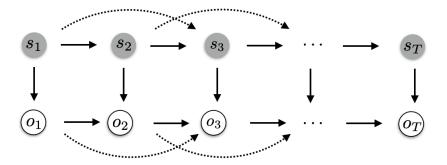


Figure 2: Graphical Model of Data-Generating Process with 2nd order Markov Dependency on the observation and hidden level.

$$- \mathbf{A}_{ijk} = p(s_t = j | s_{t-1} = i, s_{t-2} = k)$$

$$-\sum_{j=1}^{|\mathcal{S}|} \mathbf{A}_{ijk} = 1 \ \forall i, k = 1, \dots, |\mathcal{S}|$$
$$-\mathbf{A}_{ijk} \ge 0 \ \forall i, j, k = 1, \dots, |\mathcal{S}|$$

• Emissions:  $p(o_t|o_{t-1},o_{t-2},s_t)=\mathbf{B}\in\mathbb{R}^{|\mathcal{O}|\times|\mathcal{O}|\times|\mathcal{O}|\times|\mathcal{S}|}$ 

- 
$$\mathbf{B}_{ijkl} = p(o_t = j | o_{t-1} = i, o_{t-2} = k, s_t = l) := p_{j|il}^{(k)}$$
  
-  $\sum_{j=1}^{|\mathcal{O}|} \mathbf{B}_{ijkl} = 1 \ \forall i = 1, \dots, |\mathcal{O}|, k = 1, \dots, |\mathcal{S}|$   
-  $\mathbf{B}_{ijkl} \ge 0 \ \forall i, j, k = 1, \dots, |\mathcal{O}|, l = 1, \dots, |\mathcal{S}|$ 

$$-\sum_{i=1}^{|\mathcal{O}|} \mathbf{B}_{ijkl} = 1 \ \forall i = 1, \dots, |\mathcal{O}|, k = 1, \dots, |\mathcal{S}|$$

- 
$$\mathbf{B}_{ijkl} \ge 0 \ \forall i, j, k = 1, \dots, |\mathcal{O}|, l = 1, \dots, |\mathcal{S}|$$

Rob: Add examples of samples with different params/dependency orders.

### 3 Sequential Bayesian Agents

In the following we will derive updating equations as well as analytically tractable expressions for the surprise measures of interest. More specifically, we introduce two different classes of agents: A conjugate Categorical-Dirichlet (CD) agent as well as a Hidden Markov Model (HMM) agent. They differ in their degree of complexity and allow for different types of hidden state tracking:

• CD agent: The hidden state  $s_t$  is assumed to be shared across time and hence static:

$$p(s_t|s_{t-1}) = \delta_{s_{t-1}}(s_t) \Leftrightarrow s_t = s_{t-1} = s \ \forall t = 1, \dots, T.$$

• HMM agent:

We differentiate between three model settings:

- 1. **Stimulus Probability (SP) Inference**: The model of the agent does not capture any Markov dependency. The current observation  $o_t$  only depends on the hidden state s.
- 2. Alternation Probability (AP) Inference: The model captures a limited form of first-order Markov dependency, where the probability of altering observations  $o_t \neq o_{t-1}$  is estimated given the hidden state s and  $o_{t-1}$ .
- 3. **Transition Probability (TP) Inference**: The model accounts for full first-order Markov dependency and estimates separate alternation probabilities depending on the previous state  $o_{t-1}$  and  $s_t$ .

Hence, the degrees of freedom and level of abstractions with which the agent is able to model the received sequence differs between the agents as well as the tracked statistic of interest.

#### 3.1 Categorical-Dirichlet Agent

The Categorical-Dirichlet agent is part of the Bayesian conjugate pairs. It models the likelihood of the observations with the help of the Categorical distribution with  $o_t \in \{1,\ldots,M\}$  different possible realizations per sample. Given the probability vector  $\mathbf{s} = \{s^1,\ldots,s^M\}$  with  $s^i \geq 0$  and  $\sum s^i = 1$ , the probability density function is given by

$$p(o_t = i | s_t^1, \dots, s_t^M) = s_t^i$$

Furthermore, the prior distribution over the hidden state s is given by the Dirichlet distribution which is parametrized by  $\alpha = \{\alpha^1, \dots, \alpha^M\}$ 

$$p(s_1, \dots, s_M | \alpha_1, \dots \alpha_M) = \frac{\Gamma(\sum_{i=1}^M \alpha_i)}{\prod_{i=1}^M \Gamma(\alpha_i)} \prod_{i=1}^M s_i^{\alpha_i - 1}$$

In summary, the CD

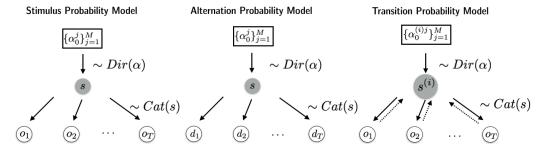


Figure 3: Categorical-Dirichlet Agent as Graphical Model. Left.Middle.Right.

Exponentially weighted forgetting

- $\mathcal{O} = \{0, 1\}, \mathcal{S} = [0, 1]$
- $\bullet \ p(s_t|s_{t-1}) = \delta_{s_{t-1}}(s_t)$
- $\bullet \ d_t = \mathbf{1}_{o_t \neq o_{t-1}}$
- SP:  $p(o_t|s_t)$ , AP:  $p(d_t|s_t)$ , TP:  $p(o_t|o_{t-1},s_t)$
- Limited memory via exponential decay in parameter updates
- Closed-form posterior/suprise via Beta-Bernoulli conjugacy

### **Stimulus Probability Model**

**Alternation Probability Model** 

**Transition Probability Model** 

### 3.2 Hidden Markov Model Agent

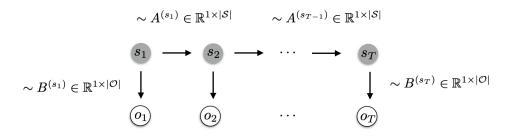


Figure 4: Hidden Markov Model Agent as Graphical Model.

Stimulus Probability Model Alternation Probability Model Transition Probability Model

### 4 Simulations

Now that we have established the theoretical framework, we are putting the above frameworks to the test.

- Catch trial (intermediate) if  $o_t = 2$  Probability:  $p^{catch}$
- Regime change if  $o_t=3$  Probability:  $p^{reg-change}$
- Let the 2-nd order dependencies be denoted by  $p_{01}^0 = p(o_t = 0 | o_{t-1} = 0, o_{t-2} = 1)$ .
- Matrix is fully specified by the four probabilities  $p_{00}^0, p$
- $\bullet\,$  E.g. given  $p^0_{00},$  we have  $p^1_{00}=1-p^0_{00}-p^{catch}/4-p^{reg-change}/4.$

$$\mathbb{P}_{s_t} = p(o_t|o_{t-1}, o_{t-2}, s_t) = \begin{bmatrix} o_{t-1} & o_{t-2} & o_{t-2} & o_{t-1} & o_{t-2} & o_{t-2} & o_{t-1} & o_{t-2} & o_{t-2} & o_{t-2} & o_{t-1} & o_{t-2} & o_{t-1} & o_{t-2} & o_{t-2}$$

- $A^{s_t}$  is row-stochastic.
- Marginalizing yields the following relationships 0-th order dependency:

$$p(o_t = i) = \sum_{j,k} p(o_t = i | o_{t-1} = j, o_{t-2} = k) = \sum_{j,k} p_{jk}^i$$

• Furthermore, 1-st order dependencies are the following:

$$p(o_t = i | o_{t-1}) = \begin{cases} p_{10}^i + p_{11}^i, & \text{for } o_{t-1} = 1\\ p_{00}^i + p_{01}^i, & \text{for } o_{t-1} = 0 \end{cases}$$

- What does slow or fast changing mean in this framework (1-st or 2-nd order)?
- How to go about catch trial? Go to third order?  $p(o_t|o_{t-1},o_{t-2}=2)=p(o_t|o_{t-1},o_{t-3})$

## 5 Conclusions

Add a table which compares the different model specifications!

### References

### **Todo list**

### **Supplementary Material**

### **Notes on Reproduction**