Summary of *A tutorial on the free-energy framework for modelling perception and learning* by Rafal Bogacz

Marco Casari

University of Turin

Complex system in neuroscience, 12 December 2023

Introduction

- Predictive coding model of Rao and Ballard.

Introduction

- Predictive coding model of Rao and Ballard.
- Free-energy model of Friston.

Introduction

- Predictive coding model of Rao and Ballard.
- Free-energy model of Friston.
- Hebbian plasticity.

Introduction

- Predictive coding model of Rao and Ballard.
- Free-energy model of Friston.
- Hebbian plasticity.
- Free energy minimization.

Working hypotheses

1. Local computation.

Working hypotheses

1. Local computation.
2. Local plasticity.

Working hypotheses

1. Local computation.
2. Local plasticity.
3. Basic neuronal computation.

Single variable model

- Feature is a scalar variable $v \in \Omega_v$.
- Stimulus is a scalar variable $u \in \Omega_u$.

Single variable model

- Feature is a scalar variable $v \in \Omega_v$.
- Stimulus is a scalar variable $u \in \Omega_u$.
- Relation between feature and stimulus is a differentiable function $g : \Omega_v \to \Omega_u$.

Single variable model

- Feature is a scalar variable $v \in \Omega_v$.
- Stimulus is a scalar variable $u \in \Omega_u$.
- Relation between feature and stimulus is a differentiable function $g : \Omega_v \to \Omega_u$.
- Sensory input $p(u|v)$ is affected by gaussian noise and it has mean $g(v)$ and variance Σ_u.

Single variable model

- Feature is a scalar variable $v \in \Omega_v$.
- Stimulus is a scalar variable $u \in \Omega_u$.
- Relation between feature and stimulus is a differentiable function $g : \Omega_v \to \Omega_u$.
- Sensory input $p(u|v)$ is affected by gaussian noise and it has mean $g(v)$ and variance Σ_u.
- Prior knowledge of the feature $p(v)$ follows a gaussian distribution with mean v_p and variance Σ_p.

Exact solution to the inference problem

- Bayes theorem:

$$p(v|u) = \frac{p(v)p(u|v)}{p(u)} \quad . \tag{1}$$

Exact solution to the inference problem

- Bayes theorem:

$$p(v|u) = \frac{p(v)p(u|v)}{p(u)} \quad . \tag{1}$$

- Marginal likelihood of stimuli:

$$p(u) = \int_{\Omega_v} p(v)p(u|v)\,\mathrm{d}v \quad . \tag{2}$$

Exact solution to the inference problem

- Bayes theorem:

$$p(v|u) = \frac{p(v)p(u|v)}{p(u)} \quad . \tag{1}$$

- Marginal likelihood of stimuli:

$$p(u) = \int_{\Omega_v} p(v)p(u|v)\,\mathrm{d}v \quad . \tag{2}$$

- No implementation in simple biological systems.

Approximate solution to the inference problem

- Most likely value of the feature is a scalar variable $\phi \in \Omega_v$.

Approximate solution to the inference problem

- Most likely value of the feature is a scalar variable $\phi \in \Omega_v$.
- Equivalent to maximize negative free energy with respect to the feature:

$$F(v, u) = \ln \left(p(v) p(u|v) \right) \quad . \tag{3}$$

Approximate solution to the inference problem

- Most likely value of the feature is a scalar variable $\phi \in \Omega_v$.
- Equivalent to maximize negative free energy with respect to the feature:

$$F(v, u) = \ln \left(p(v) p(u|v) \right) \quad . \tag{3}$$

- Prediction errors:

$$\varepsilon_p = \frac{v - v_p}{\Sigma_p} \quad , \tag{4}$$

$$\varepsilon_u = \frac{u - g(v)}{\Sigma_u} \quad . \tag{5}$$

Neural implementation

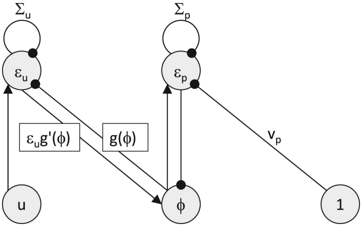

Fig. 3 from article: network implementation of the dynamical system

$$\begin{cases} \dot{\phi} = \varepsilon_u g'(\phi) - \varepsilon_p \\ \dot{\varepsilon}_p = \phi - v_p - \Sigma_p \varepsilon_p \\ \dot{\varepsilon}_u = u - g(\phi) - \Sigma_u \varepsilon_u \end{cases} . \tag{6}$$

Learning model parameters

- Choose model parameters to maximize $p(u)$.

Learning model parameters

- Choose model parameters to maximize $p(u)$.
- Equivalent to maximize negative free energy with respect to parameters:

$$\frac{\partial F}{\partial v_p} = \frac{\phi - v_p}{\Sigma_p} \quad , \tag{7}$$

$$\frac{\partial F}{\partial \Sigma_p} = \frac{1}{2}\left(\frac{(\phi - v_p)^2}{\Sigma_p^2} - \frac{1}{\Sigma_p}\right) \quad , \tag{8}$$

$$\frac{\partial F}{\partial \Sigma_u} = \frac{1}{2}\left(\frac{(u - g(\phi))^2}{\Sigma_u^2} - \frac{1}{\Sigma_u}\right) \quad . \tag{9}$$

Learning model parameters

- Choose model parameters to maximize $p(u)$.
- Equivalent to maximize negative free energy with respect to parameters:

$$\frac{\partial F}{\partial v_p} = \frac{\phi - v_p}{\Sigma_p} \quad , \tag{7}$$

$$\frac{\partial F}{\partial \Sigma_p} = \frac{1}{2}\left(\frac{(\phi - v_p)^2}{\Sigma_p^2} - \frac{1}{\Sigma_p}\right) \quad , \tag{8}$$

$$\frac{\partial F}{\partial \Sigma_u} = \frac{1}{2}\left(\frac{(u - g(\phi))^2}{\Sigma_u^2} - \frac{1}{\Sigma_u}\right) \quad . \tag{9}$$

- Hebbian plasticity is satisfied using prediction errors.

Learning relation parameter

- Linear relation:

$$g(v, \theta) = \theta v \quad . \tag{10}$$

Learning relation parameter

- Linear relation:

$$g(v, \theta) = \theta v \quad . \tag{10}$$

- Nonlinear relation:

$$g(v, \theta) = \theta h(v) \quad . \tag{11}$$

Learning relation parameter

- Linear relation:

$$g(v, \theta) = \theta v \quad . \tag{10}$$

- Nonlinear relation:

$$g(v, \theta) = \theta h(v) \quad . \tag{11}$$

- Gradient of negative free energy for learning:

$$\frac{\partial F}{\partial \theta} = \frac{u - \theta h(\phi)}{\Sigma_u} h(\phi) = \varepsilon_u h(\phi) \quad . \tag{12}$$

Free energy framework

- Minimization of Kullback-Leibler divergence:

$$KL(q(v)||p(v|u)) = \int_{\Omega_v} q(v) \ln\left(\frac{q(v)}{p(v|u)}\right) \mathrm{d}v \quad . \tag{13}$$

Free energy framework

- Minimization of Kullback-Leibler divergence:

$$KL(q(v)||p(v|u)) = \int_{\Omega_v} q(v) \ln \left(\frac{q(v)}{p(v|u)} \right) \mathrm{d}v \quad . \tag{13}$$

- Definition of negative free energy:

$$F(v, u) = \int_{\Omega_v} q(v) \ln \left(\frac{p(v, u)}{q(v)} \right) \mathrm{d}v \quad . \tag{14}$$

Free energy framework

- Minimization of Kullback-Leibler divergence:

$$KL(q(v)||p(v|u)) = \int_{\Omega_v} q(v) \ln \left(\frac{q(v)}{p(v|u)} \right) \mathrm{d}v \quad . \tag{13}$$

- Definition of negative free energy:

$$F(v,u) = \int_{\Omega_v} q(v) \ln \left(\frac{p(v,u)}{q(v)} \right) \mathrm{d}v \quad . \tag{14}$$

- For the models discussed in the paper: $q(v) = \delta(v - \phi)$.

Multiple variables model

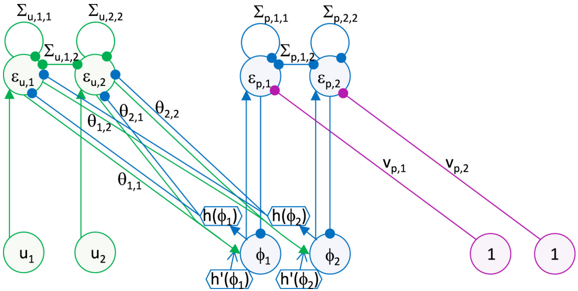

Fig. 5 from article: example of a model with 2 features and 2 stimuli. Same equations in matrix notation, but local plasticity is not satisfied.

Hierarchical structure implementation

- Parallel to structure of cortical areas.

Hierarchical structure implementation

- Parallel to structure of cortical areas.
- Generalized equations for the inference task:

$$\dot{\vec{\phi}}_i = -\vec{\varepsilon}_i + h'(\vec{\phi}_i) \times \mathbf{\Theta}_{i-1}^{\intercal} \vec{\varepsilon}_{i-1} \quad , \tag{15}$$

$$\dot{\vec{\varepsilon}}_i = \vec{\phi}_i - \mathbf{\Theta}_i h(\vec{\phi}_{i+1}) - \mathbf{\Sigma}_i \vec{\varepsilon}_i \quad . \tag{16}$$

- Generalized equations for the learning task:

$$\frac{\partial F}{\partial \mathbf{\Sigma}_i} = \frac{1}{2}(\vec{\varepsilon}_i \vec{\varepsilon}_i^{\intercal} - \mathbf{\Sigma}_i^{-1}) \quad , \tag{17}$$

$$\frac{\partial F}{\partial \mathbf{\Theta}_i} = \vec{\varepsilon}_i h(\vec{\phi}_{i+1})^{\intercal} \quad . \tag{18}$$

Recover local plasticity

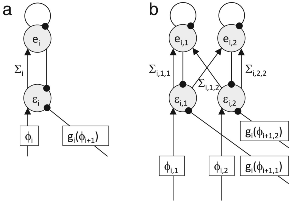

Fig. 7 from article: networks satisfying local plasticity for (a) single variable model and (b) multiple variables model. They implement the generalized dynamical system

$$\begin{cases} \dot{\vec{\varepsilon}}_i = \vec{\phi}_i - g_i(\vec{\phi}_{i+1}) - \vec{e}_i \\ \dot{\vec{e}}_i = \mathbf{\Sigma}_i \vec{\varepsilon}_i - \vec{e}_i \end{cases} . \tag{19}$$

Conclusion

- Stimuli weighted by noise.
- Learn covariance of stimuli.
- Attentional modulation.