Title

Abstract

blahhh blahhh blah

Introduction

XXX with a special focus on the interneurons and how and hwy those may adapt yhe weighting XXX

Results

XXX

nPE and pPE neurons as the basis for computing mean and variance of sensory stimuli

We hypothesis that the unique properties of nPE and pPE neurons put them in a perfect position to support both the computation of the mean and the variance of feedforward sensory stimuli. XXX Why is that XXX.

To this end we simulated a rate-based mean-field network model the core of which represents a PE circuit with nPE and pPE neurons and three types of inhibitory interneurons (XXX). In addition, the PE circuits connect/project to a memory neuron that is modeled as perfect integrator. In accordance with XXX, we assume that the pPE neurons excite the memory neuron while the nPE neuron inhibit the memory neuron (for instance through lateral inhibition not modeled explicitly here). With this connectivity/setup/architecture, the memory neuron holds/encodes the mean of the feedforward sensory stimuli that drive the PE circuit. The memory neuron, on the other hand, connects to the apical dendrites of the PE neurons and some of the inhibitory interneurons (see methods for more details). It therefore serves as a prediction that is dynamically updated when new sensory information is available. We furthermore assume that the PE neurons excite another neuron, modeled as a leaky integrator, whose activity may represent that variance of the feedforward stimuli when both nPE and pPE neurons have a net excitatory effect on it.

To show that such a network can indeed represent mean and variance in the respective neurons, we stimulate it with sequence of step-wise constant inputs drawn from xxx. As shown in Fig. XXX, the memory neuron's activity gradually approaches the mean of the sensory stimuli, while the v neuron approaches the variance of the inputs. This is true for a wide range of inputs statistics (Fig. XXX). Deviations from the true mean occur mainly for larger input variances. The estimated variance is fairly independent of the input statistics tested. Moreover, the precise input distribution does not have a significant effect on the network's ability to estimate the mean and the variance (Supp Fig. XXX).

XXX interneuron activity increases with mean but also with variance, some faster than the other ... XXX

XXX validated that also correct for population network (beyond mean-field network) XXX

XXX assumptions (BL, gain-nPE = gain-pPE = 1) and how important really (see above)

Weighting external and internal signals requires two sets of PE neurons

Coming back to the question at hand/example ... Following up the same ideas, the weighting of internal and external signals requires a higher PE circuit that integrates over the prediction of the lower PE circuit. XXX explain set-up XXX ... XXX stimulation protocol XXX XXX two limit cases XXX XXX validation for a wide range of input statistics XXX XXX mentioning the dynamic nature of this estimation (Supp material), if you find differences in time scales of adaptation, report here XXX XXX predictions: nPE/pPE BL increase, cognitive load + trial duration XXX XXX The network does not require on the squared activation function but it is important that updating is faster in lower than in higher PE circuit XXX

XXX Do we actually propose that predictions are sent up the hierarchy?

XXX IN influence, neuromodulators ... mechanisms XXX

XXX contraction bias

Discussion

We solved the brain.

Models and methods

Network model

Network consists of two subnetworks. Each subnetwork consists of a PE circuit, a memory neuron and a neuron representing the variance. XXX The memory neuron of subnetwork feefdowradly connects to the PE circuit of the second subnetwork.

Prediction-error network model

Consider a mean field network in which each population is represented by one representative neuron. The mean-field PE network consists of an excitatory nPE and pPE neuron, as well as two inhibitory PV neurons (one receiving S, the other P), as well as inhibitory SOM and VIP neurons.

Each excitatory pyramidal cell (that is, nPE or pPE neuron) is divided into two coupled compartments, representing the soma and the dendrites, respectively. The dynamics of the firing rate r_{E} of the somatic compartment of obeys (**?**)

$$\tau_E \frac{dr_{\mathrm{E}}}{dt} = -r_{\mathrm{E}} + w_{\mathrm{ED}} \cdot r_{\mathrm{D}} - w_{\mathrm{EP}} \cdot r_{\mathrm{P}} + I_{\mathrm{E}}, \tag{1}$$

where τ_{E} denotes the excitatory rate time constant (τ_{E}=60 ms), the weight w_{ED} describes the connection strength between the dendritic compartment and the soma of the same neuron, and w_{EP} denotes the strength of somatic inhibition from PV neurons. The overall input I_{E} comprises external background and feedforward sensory inputs (see "Inputs" below). Firing rates are rectified to ensure positivity.

The dynamics of the activity r_{D} of the dendritic compartment obeys (**?**)

$$\tau_E \frac{dr_{\mathrm{D}}}{dt} = -r_{\mathrm{D}} + w_{\mathrm{DE}} \cdot r_{\mathrm{E}} - w_{\mathrm{DS}} \cdot r_{\mathrm{S}} + I_{\mathrm{D}}, \tag{2}$$

where the weight w_{DE} denotes the recurrent excitatory connections between PCs, including backpropagating activity from the soma to the dendrites. w_{DS} represents the strength of dendritic inhibition from SOM neurons. The overall input I_{D} comprises fixed, external background inputs and feedback predictions (see "Inputs" below). We assume that any excess of inhibition in a dendrite does not affect the soma, that is, the dendritic compartment is rectified at zero.

Just as for the excitatory neurons, the firing rate dynamics of each interneuron is modeled by a rectified, linear differential equation (**?**),

$$\tau_I \frac{dr_{\mathrm{X}}}{dt} = - r_{\mathrm{X}} + I_{\mathrm{X}} + w_{\mathrm{XE}} \cdot r_{\mathrm{E}} - w_{\mathrm{XP}} \cdot r_{\mathrm{P}} - w_{\mathrm{XS}} \cdot r_{\mathrm{S}} - w_{\mathrm{XV}} \cdot r_{\mathrm{V}}, \tag{3}$$

where r_{X} denotes the firing rate of neuron type X, and the weight matrices w_{XY} denote the strength of connection between the presynaptic neuron population Y and the postsynaptic neuron population X ($X, Y \in \{P, S, V\}$). The rate time constant τ_I was chosen to resemble a fast $\mathrm{GABA_A}$ time constant, and set to 2 ms for all interneuron types included. The overall input I_{X} comprises fixed, external background inputs, as well as feedforward sensory inputs and feedback predictions (see "Inputs" below).

Memory and variance neuron

$$\tau_m \cdot \frac{dr_{\mathrm{M}}}{dt} = w_{\mathrm{M}\leftarrow\mathrm{pPE}} \cdot r_{\mathrm{pPE}} - w_{\mathrm{M}\leftarrow\mathrm{nPE}} \cdot r_{\mathrm{nPE}} \tag{4}$$

$$\tau_v \cdot \frac{dr_{\mathrm{V}}}{dt} = -r_{\mathrm{V}} + (w_{\mathrm{V}\leftarrow\mathrm{pPE}} \cdot r_{\mathrm{pPE}} - w_{\mathrm{V}\leftarrow\mathrm{nPE}} \cdot r_{\mathrm{nPE}})^2 \tag{5}$$

Weighted output

$$r_{\text{out}} = \alpha \cdot S + (1 - \alpha) \cdot P \tag{6}$$

$$\begin{aligned} \alpha &= \frac{1/r_{\text{V1}}}{1/r_{\text{V1}} + 1/r_{\text{V2}}} \\ &= \left(1 + \frac{r_{\text{V1}}}{r_{\text{V2}}}\right)^{-1} \end{aligned} \tag{7}$$

Connectivity

Inputs

Simulations

Acknowledgments

Supplementary Information

Sensory weight and contraction bias

If P is rather constant, the slope in the contraction bias is exactly the sensory weight

$$r_{\text{out}} = \alpha_{\text{S}} \cdot S + (1 - \alpha_{\text{S}}) \cdot P \equiv m \cdot S + n$$

However, P is usually/normally a function of S. For simplicity, let's assume that P decays exponentially to a new value of S:

$$P = P_0 \cdot \mathrm{e}^{-t/\tau} + f(S) \cdot \left(1 - \mathrm{e}^{-t/\tau}\right)$$

Within each trial with trial duration T, P can be expressed by n sections of length t in which the stimulus is constant and, for the sake of simplicity, drawn from a uniform distribution $U\left(s - \frac{\sigma_{\text{S}}}{12}, s + \frac{\sigma_{\text{S}}}{12}\right)$. P_0 is drawn from $U\left(\mu - \frac{\sigma_{\text{P}}}{12}, \mu + \frac{\sigma_{\text{P}}}{12}\right)$. P_{n} is then given by

$$P_{\text{n}} = P_0 \cdot \mathrm{e}^{-t/\tau} + \left(1 - \mathrm{e}^{-t/\tau}\right) \sum_{i=1}^{n} s_i \cdot \mathrm{e}^{-(n-1)\cdot t/\tau}$$

This needs to be averaged over all possible states

$$P_{\text{n}} = \mathrm{e}^{-t/\tau} \int_{\mu - \frac{\sigma_{\text{P}}}{12}}^{\mu + \frac{\sigma_{\text{P}}}{12}} P_0 \; f(P_0) \; dP_0 + \left(1 - \mathrm{e}^{-t/\tau}\right) \sum_{i=1}^{n} \cdot \mathrm{e}^{-(n-1)\cdot t/\tau} \int_{s - \frac{\sigma_{\text{S}}}{12}}^{s + \frac{\sigma_{\text{S}}}{12}} s \; f(s) \; ds$$

This gives

$$P_{\text{n}} = \mu \cdot \mathrm{e}^{-T/\tau} + \left(1 - \mathrm{e}^{-t/\tau}\right) \sum_{i=1}^{n} \mathrm{e}^{-(n-i)\cdot t/\tau} \cdot S$$

By making use of the geometric series, this simplifies to

$$P_{\text{n}} = \mu \cdot \mathrm{e}^{-T/\tau} + \left(1 - \mathrm{e}^{-T/\tau}\right) \cdot S$$

Together, this yields for the weighted output

$$r_{\text{out}} = \left[\alpha_{\text{S}} \; \mathrm{e}^{-T/\tau} + \left(1 - \mathrm{e}^{-T/\tau}\right)\right] \cdot S + (1 - \alpha_{\text{S}}) \; \mathrm{e}^{-T/\tau} \; \mu$$

Hence, the slope is a function of both the sensory weight and the trial duration.

In a prediction-driven input regime ($alpha_{\text{S}} \sim 0$), the slope is independent of the sensory weight and only determined by the trial duration, $m \sim \left(1 - \mathrm{e}^{-T/\tau}\right)$. In a sensory-driven input regime ($alpha_{\text{S}} \sim 0$), the contraction bias vanishes ($m \sim 1$).

If the trail duration is short ($T \to 0$), the slope is given by the sensory weight. If the trail duration approaches infinity, the slope would be 1 again (however, this seems rather unrealistic, this would only be true in an ideal system without memory decay or reproduction and accumulation noise ...).

Effect of nPE and pPE gain

In the steady state, the averaged effect of nPE and pPE must be equal (so the gain must be equal):

$$\begin{aligned} g_{pPE} \; \langle \text{nPE} \rangle &= g_{nPE} \; \langle \text{pPE} \rangle \\ g_{pPE} \langle [S - P]_+ \rangle &= g_{nPE} \langle [P - S]_+ \rangle \\ g_{pPE} \int_{P}^{b} (x - P) \; f(x) \; dx &= g_{nPE} \int_{a}^{P} (P - x) \; f(x) \; dx \end{aligned}$$

Example, uniform distribution:

$$g_{pPE} \; \left[\frac{1}{2}\left(b^2 - P^2\right) - P\,(b - P)\right] = g_{nPE} \; \left[P\,(P - a) - \frac{1}{2}\left(P^2 - a^2\right)\right]$$

which gives

$$P = \frac{g_{pPE}\ b - g_{nPE}\ a \pm \sqrt{g_{nPE}\ g_{pPE}}(a-b)}{g_{pPE} - g_{nPE}}$$

Only if $g_{\mathrm{nPE}} = g_{\mathrm{pPE}}$, the prediction P is given by $\frac{a+b}{2}$.

For variance, we need $g_{nPE} = g_{pPE} = 1$. In case of a uniform distribution, we get

$$\begin{aligned} V &= \langle (S-P)^2 \rangle \\ &= g_{\mathrm{pPE}}\ \langle \langle [S-P\rangle]_+ \rangle + g_{\mathrm{nPE}}\ \langle \langle [P-S\rangle]_+ \\ &= \frac{g_{\mathrm{pPE}}}{b-a} \int_P^b (u-P)^2\ du + \frac{g_{\mathrm{nPE}}}{b-a} \int_a^P (P-u)^2\ du \\ &= \frac{g_{\mathrm{pPE}}}{3} \cdot \frac{(b-P)^3}{b-a} + \frac{g_{\mathrm{nPE}}}{3} \cdot \frac{(P-a)^3}{b-a} \end{aligned}$$

if $P = (b+a)/2$

$$V = \frac{g_{\mathrm{pPE}} + g_{\mathrm{nPE}}}{24} \cdot (b-a)^2$$

Only if $g_{\mathrm{nPE}} = g_{\mathrm{pPE}} = 1$, the V neuron encodes the variance, that for a uniform distribution is given by

$$V = \frac{(b-a)^2}{12}$$

XXXXXXX add the version where P is given by the g version ... Of course, P is also a function of the gains (see above). If we take this into account, V is given by

$$V = \frac{(b-a)^2}{3\ (g_{\mathrm{pPE}} - g_{\mathrm{nPE}})^3} \cdot \left[g_{\mathrm{nPE}} \cdot (g_{\mathrm{pPE}} \mp \sqrt{g_{\mathrm{nPE}}\ g_{\mathrm{pPE}}})^3 - g_{\mathrm{pPE}} \cdot (g_{\mathrm{nPE}} \mp \sqrt{g_{\mathrm{nPE}}\ g_{\mathrm{pPE}}})^3 \right]$$

Effect of BL activity on mean an variance

$$\begin{gathered} \langle pPE \rangle = \langle nPE \rangle \\ \langle [S-P]_+ + p_0 \rangle = \langle [P-S]_+ + n_0 \rangle \\ \int_P^b (x-P)\ f(x)\ dx + p_0 \underbrace{\int_a^b f(x)\ dx}_{=1} = \int_a^P (P-x)\ f(x)\ dx + n_0 \underbrace{\int_a^b f(x)\ dx}_{=1} \end{gathered}$$

In case of a uniform distribution, that is $f(x) = \frac{1}{b-a}$ for $x \in [a,b]$, 0 otherwise, this leads to

$$P = \frac{b+a}{2} + \frac{p_0 + n_0}{b-a}$$

Variance as function of bias in mean:

$$\begin{aligned} V &= \frac{1}{n} \sum_i \left(x - (\mu \pm \delta\mu)\right)^2 \\ &= \frac{1}{n} \sum_i \{(x_i - \mu)^2 + \delta\mu^2 \mp 2\delta\mu(x_i - \mu)\} \\ &= V_{\mathrm{uniform}} + \delta\mu^2 \mp 2\ \delta\mu \left(\frac{1}{n} \sum_i x_i - \mu \right) \\ &= V_{\mathrm{uniform}} + \delta\mu^2 \end{aligned}$$

Variance as a function of BL in nPE and pPE:

$$\begin{aligned} V &= \langle (\mathrm{pPE} + \mathrm{nPE})^2 \rangle \\ &= \langle [S-P]_+ \rangle + \langle [P-S]_+ \rangle + (p_0 + n_0)^2 + 2\ (p_0 + n_0)\ \left(\langle [S-P]_+ \rangle + \langle [P-S]_+ \rangle \right) \end{aligned}$$

In case of a uniform distribution, this can be expressed by

$$V = \frac{1}{3\ (b-a)}\left[(b-P)^3 + (P-a)^3\right] + (p_0+n_0)^2 + \frac{(p_0+n_0)}{b-a}\left[(b-P)^2 + (a-P)^2\right]$$

Since $P = \frac{b+a}{2} + \frac{p_0+n_0}{b-a}$, this yields

$$\begin{aligned} V =& \frac{1}{3\ (b-a)}\left[\left(\frac{b-a}{2} - \frac{p_0-n_0}{b-a}\right)^3 + \left(\frac{b-a}{2} + \frac{p_0-n_0}{b-a}\right)^3\right] + (p_0+n_0)^2 \\ &+ \frac{(p_0+n_0)}{b-a}\left[\left(\frac{b-a}{2} + \frac{p_0-n_0}{b-a}\right)^2 + \left(\frac{b-a}{2} - \frac{p_0-n_0}{b-a}\right)^2\right] \end{aligned}$$

Simplifying this expression, leads to

$$V = \frac{(b-a)^2}{12} + \frac{(p_0-n_0)^2}{(b-a)^2}\left(1 + 2\ \frac{p_0+n_0}{b-a}\right) + (p_0+n_0)\left(p_0+n_0+\frac{b-a}{2}\right)$$

Influence of a population of nPE and pPE neurons

XXX

Analysis of simplified network model, effect of time constants

simplified model: dynamics and steady state of rM and rV, rM and rV as a function of time constants and trial duration etc., weighting, then use those expressions to discuss when weighting goes awry and how long transitions take from one state to another ...

Comparison to Kalman filter and Bayes Factor surprise

Kalman filter. Initialisation

$$\begin{aligned} x_{0|init} &= 0 \\ P_{0|init} &= \sigma^2\ I \end{aligned}$$

with x being the system state (in my terms the prediction), P is the covariance matrix of the errors of x (in my terms the var of the predictions) and I is the identity matrix. Then the "correction" is given by

$$\begin{aligned} K_k &= P_{k|k-1}\ H_k^T\ \left(H_k\ P_{k|k-1}\ H_k^T + R_k\right)^{-1} \\ x_k &= x_{k|k-1} + K_k\ \left(z_k - H_k\ x_{k|k-1}\right) \\ P_k &= (I - K_k\ H_k)\ \ P_{k|k-1} \end{aligned}$$

with K the kalman gain matrix, H the observation matrix ($z_k = H_k\ x_k + noise$), R the covaraince of the measurement noise and z a new observation. The last part of the Kalman filter is the "prediction":

$$\begin{aligned} x_{k|k-1} &= F_{k-1}\ x_{k-1} + B_{k-1}\ u_{k-1} \\ P_{k|k-1} &= F_{k-1}\ P_{k-1}\ F_{k-1}^T + Q_{k-1} \end{aligned}$$

with F the transition matrix ($x_{k|k-1} = F_{k-1}\ x_{k-1}$, u a deterministic perturbation, B the dynamics of the deterministic perturbation. In our terms

$$\alpha = K_k = \frac{P_{k|k-1}}{R_k + P_{k|k-1}}$$

$P_{k|k-1}$, is however σ_P^2 in my implementation and R_k is fixed variance of inputs σ_S^2. Hence, my implementation represents (?) the Kalman filter. Important to note is, that in my implementation we estimate the variance of inputs dynamically, so it is not set! Another nice advantage here is that I don't need a good estimate for P. I can basically initiate it as I want. Another difference is that I consider the optimal weighting in my "output neuron" and not the prediction itself ... .

XXX Comparison to Bayes Factor surprise