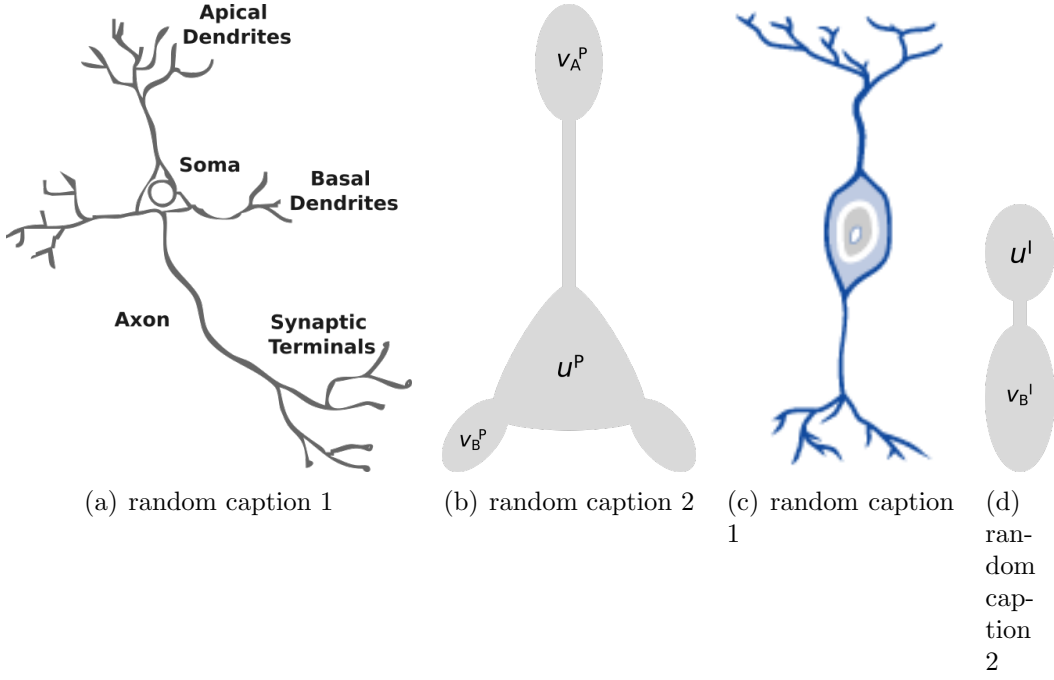


Neuro-Backprop

Paul Hill

August 2020



1 Introduction

1.1 Multicompartment-Neurons

Real nerve-cells usually have a spatial extension much larger than their actual cell-bodies (somas). To model the spatial variation of electric potentials over such cells (specifically the cell's dendrites), which is due to the finite speed with which electric signals propagate, one introduces several distinct compartments that are coupled to each other. The potential of the soma-compartment is then given by

$$\dot{u} = -g_l u + \sum_x g_x (v_x - u) + I_{syn}, \quad (1)$$

where g_l is the leakage conductance of the soma, x runs over all connected compartments of the cell and g_x is the respective conductance coupling x to the soma. I_{syn} accounts for an additional current input coupled directly into the soma. In our case we will always have $I_{syn} = g_{som}(u_{ex} - u)$, where u_{ex} is any external potential.

Note that (1) behaves as a low-pass filter for the potentials u_x and u_{ex} with time constant $\tau =$

$\frac{1}{g_{tot}} = \frac{1}{g_l + \sum_x g_x + g_{som}}$. For imprinted signals with time scale $T \gg \tau$ the soma potentials takes on the form (steady-state solution)

$$\tilde{u} = \frac{1}{g_{tot}} \left(\sum_x g_x u_x + g_{som} u_{ex} \right). \quad (2)$$

Furthermore we restrict our model to the rate-based case in which synaptic inputs are given in terms of the firing rates of their presynaptic partners r_y

$$v_x = \sum_y W_{xy} r_y = \sum_y W_{xy} \phi(u_y). \quad (3)$$

Here the vector \mathbf{W}_x denotes the coupling between the the presynaptic partners and the x -compartment and firing rates are given in terms of the corresponding soma potentials u_y via the activation function $\phi(u_y)$.

1.2 PyraL-Network

The network approximating the backpropagation-algorithm is built upon two kind of neurons, pyramidal- and lateral inter-neurons. The first one consists of two compartments beside the soma, the basal and apical dendrites. (1) takes on the form

$$\dot{u}^P = -g_l u^P + g_B(v_B^P - u^P) + g_A(v_A^P - u^P), \quad (4)$$

where B denotes the basal and A the apical compartment. The inter-neurons on the other hand are governed by

$$\dot{u}^I = -g_l u^I + g_D(v_D^I - u^I) + g_{som}(u_{ex} - u^I), \quad (5)$$

where D denotes the dendritic compartment and we have an external potential u_{ex} directly coupled into the soma. Fig. ??? depicts both neuron types in their biological manifestation and abstract description, respectively.

With these two types of neurons one is now able to go on and build larger networks as the one depicted in Fig. 1.2.

The network dynamics for N layers ($N - 2$ hidden layers) are governed by the following set of equations (for $0 < k < N$)

$$\dot{\mathbf{u}}_k^P = -g_l \mathbf{u}_k^P + g_B(\mathbf{v}_{B,k}^P - \mathbf{u}_k^P) + g_A(\mathbf{v}_{A,k}^P - \mathbf{u}_k^P) \quad (6)$$

$$\dot{\mathbf{u}}_k^I = -g_l \mathbf{u}_k^I + g_D(\mathbf{v}_{D,k}^I - \mathbf{u}_k^I) + g_{som}(\mathbf{u}_{k+1}^P - \mathbf{u}_k^I) \quad (7)$$

$$\mathbf{v}_{B,k}^P = \mathbf{W}_k^{up} \phi(\mathbf{u}_{k-1}^P) \quad (8)$$

$$\mathbf{v}_{A,k}^P = \mathbf{W}_k^{pi} \phi(\mathbf{u}_k^I) + \mathbf{W}_k^{down} \phi(\mathbf{u}_{k+1}^P) \quad (9)$$

$$\mathbf{v}_{D,k}^I = \mathbf{W}_k^{ip} \phi(\mathbf{u}_k^P), \quad (10)$$

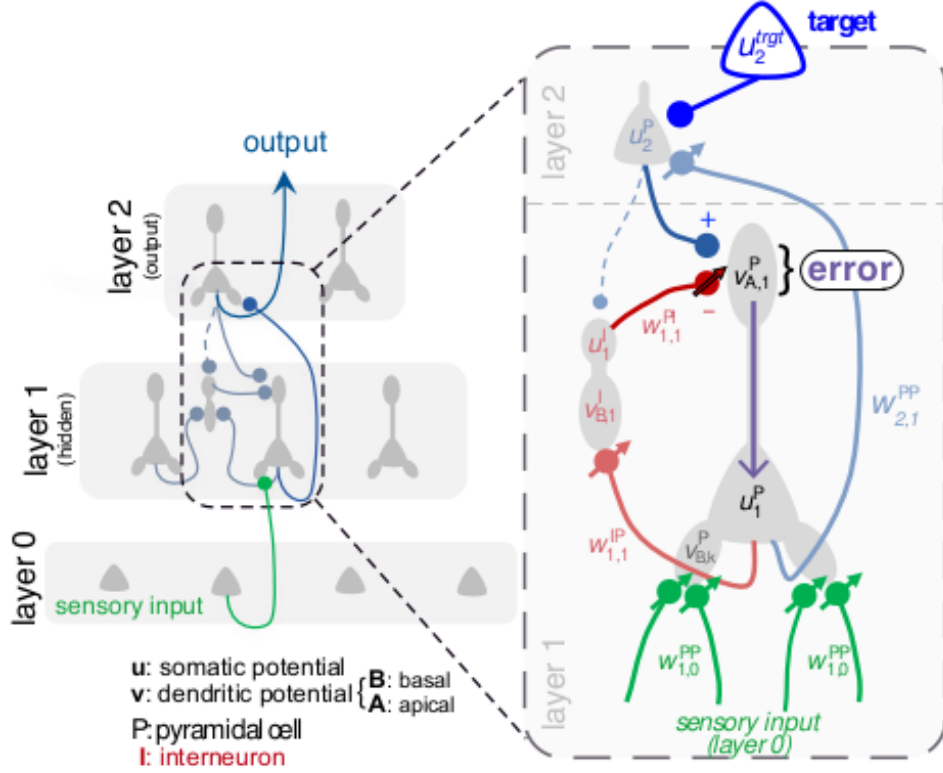
where \mathbf{W} are now matrices, \mathbf{v}, \mathbf{u} are vectors and k is the layer index ($k = 0$ is the input layer).

The output neurons look slightly different,

$$\dot{\mathbf{u}}_N^P = -g_l \mathbf{u}_N^P + g_B(\mathbf{v}_{B,N}^P - \mathbf{u}_N^P) + g_{som}(\mathbf{u}^{target} - \mathbf{u}_N^P) \quad (11)$$

$$\mathbf{v}_{B,N}^P = \mathbf{W}_N^{up} \phi(\mathbf{u}_{N-1}^P). \quad (12)$$

Note that, provided the network reaches a (unique) fix-point, i.e. a configuration in which all derivatives of the soma potentials vanish simultaneously, the network essentially maps the input



rates (or input potentials) of the 0-th layer to the output rates obtained from the N -th layer. It is this non-linear mapping between input and output rates that eventually enables the network to solve complex classification and regression tasks by adjusting the strength of its synaptic connections \mathbf{W} . However, the existence (and uniqueness) of a fix-point of the underlying network dynamics depends non-trivially on the weights \mathbf{W} and conductances g .

1.3 Learning-rules

Assuming the network indeed converges to a stable fix-point for the given input rates, there is yet no mechanism that makes the output potentials reproduce a given target signal \mathbf{u}^{target} . To solve this the concept of dendritic prediction of somatic spiking rates is introduced. Consider therefore a pyramidal neuron in the output layer when teaching is disabled ($g_{som} = 0$). The soma potential is then, in steady-state, given by

$$\hat{\mathbf{v}}_{B,N}^P \equiv \tilde{\mathbf{u}}^P = \frac{g_B}{g_I + g_B} \mathbf{W} r_{N-1}^P. \quad (13)$$

When teaching is enabled, \tilde{u}^P is nudged away from the dendritic prediction \hat{v}^P towards

$$\tilde{\mathbf{u}}_N^P = (1 - \lambda_{out}) \hat{\mathbf{v}}_{B,N}^P + \lambda_{out} \mathbf{u}^{target}, \quad (14)$$

with $\lambda_{out} = \frac{g_{som}}{g_I + g_B + g_{som}}$. The difference in predicted and actual firing rate $\phi(u^P) - \phi(\hat{v}^P)$ can now be used to update the synaptic strengths such that \hat{v}^P is itself nudged towards u^{target} and eventually $\tilde{u}^P = \hat{v}^P$ holds again.

Similar equations hold for hidden pyramidal neurons and inter-neurons as well

$$\tilde{u}_k^P = \hat{v}_{B,k}^P + \lambda_{hid} \mathbf{v}_{A,k}^P, \quad \lambda_{hid} = \frac{g_A}{g_I + g_B + g_A} \quad (15)$$

$$\tilde{u}_k^I = (1 - \lambda_I) \hat{v}_{D,k}^I + \lambda_I u_{k+1}^P, \quad \lambda_I = \frac{g_{som}}{g_I + g_D + g_{som}}, \quad (16)$$

with $\hat{\mathbf{v}}_{B,k}^P = \frac{g_B}{g_l + g_B + g_A} \mathbf{W}_k^{up} r_{k-1}^P$ and $\hat{\mathbf{v}}_{D,k}^I = \frac{g_D}{g_l + g_D} \mathbf{W}_k^{ip} r_k^P$.

Finally, updating of the weights is realized by first low-pass filtering the dendritic prediction error with time constant τ_w

$$\tau_w \dot{\Delta}_k^{up} = (\phi(\mathbf{u}_k^P - \phi(\hat{\mathbf{v}}_{B,k}^P)) (r_{k-1}^P)^T \quad (17)$$

$$\tau_w \dot{\Delta}_k^{ip} = (\phi(\mathbf{u}_k^I - \phi(\hat{\mathbf{v}}_{D,k}^I)) (r_k^P)^T \quad (18)$$

$$\tau_w \dot{\Delta}_k^{pi} = -\mathbf{v}_A^P (r_k^I)^T \quad (19)$$

and then integrating

$$\frac{d\mathbf{W}_k^{up}}{dt} = \eta_k^{up} \Delta_k^{up} \quad (20)$$

$$\frac{d\mathbf{W}_k^{ip}}{dt} = \eta_k^{ip} \Delta_k^{ip} \quad (21)$$

$$\frac{d\mathbf{W}_k^{pi}}{dt} = \eta_k^{pi} \Delta_k^{pi}. \quad (22)$$

Note that, based on eq. (18) - (19), output neurons change to predict the target signal, inter-neurons change to predict the pyramidal neurons of the next layer, hidden pyramidal neurons are driven by their apical compartment and finally the apical compartment adjust the lateral \mathbf{W}^{pi} weights such that the top-down (see eq. (10)) is canceled and the apical potential becomes zero.

1.3.1 Self-predicting state

Especially when no target signal is applied this mechanism drives the network in what is called the self-predicting state. It is characterized by the cancellation of all apical potentials and when inter-neuron soma potentials match the pyramidal soma potentials of the next layer, i.e.

$$\mathbf{v}_A^P = 0 \quad (23)$$

$$\hat{\mathbf{v}}_{D,k}^I = \hat{\mathbf{v}}_{B,k+1}^P, \quad (24)$$

which is equivalently expressed as

$$\mathbf{W}_k^{pi} = -\mathbf{W}_k^{down} \quad (25)$$

$$\mathbf{W}_k^{ip} = \frac{g_l + g_D}{g_l + g_B + g_A} \frac{g_B}{g_D} \mathbf{W}_{k+1}^{up}, \quad (26)$$

with $g_A = 0$ for the output layer. Since in the self-predicting state all apical compartments vanish the network is guaranteed to have a fix-point and behaves like a artificial neural network with same forward weights ($\sim \mathbf{W}^{up}$) and activation function.

1.4 Approximation of the error-backpropagation algorithm

Starting from the the self-predicting state we now switch on the teaching signal \mathbf{u}^{target} , where throughout small couplings $\lambda = \lambda_{out} = \lambda_I = \lambda_{hid} \ll 1$ are assumed.

Doing so nudges the output potential away from $\hat{\mathbf{v}}_{B,N}^P = \tilde{\mathbf{u}}_{N-1}^I$, resulting in a change of the apical potential in the previous layer as

$$\mathbf{v}_{N-1}^P = \mathbf{W}_{N-1}^{down} \underbrace{(\phi(\tilde{\mathbf{u}}_N^P) - \phi(\hat{\mathbf{v}}_{B,N}^P))}_{\mathbf{e}_N} \approx \lambda \mathbf{W}_{N-1}^{down} \phi'(\hat{\mathbf{v}}_{B,N}^P) (\mathbf{u}^{target} - \hat{\mathbf{v}}_N^P) \equiv \lambda \mathbf{W}_{N-1}^{down} \mathbf{D}_N (\mathbf{u}^{target} - \hat{\mathbf{v}}_N^P), \quad (27)$$

where the prediction error \mathbf{e}_N was introduced. This error propagates further down the network

$$\mathbf{e}_{N-1} = \phi(\tilde{\mathbf{u}}_{N-1}^P) - \phi(\hat{\mathbf{v}}_{B,N-1}^P) \approx \phi'(\hat{\mathbf{v}}_{B,N-1}^P) \lambda \mathbf{W}_{N-1}^{down} \mathbf{e}_N = \lambda^2 \mathbf{D}_{N-1} \mathbf{W}_{N-1}^{down} \mathbf{D}_N (\mathbf{u}^{target} - \hat{\mathbf{v}}_N^P). \quad (28)$$