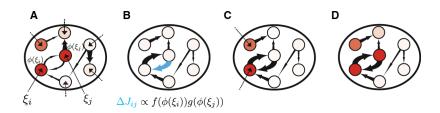
# Information bounds and attractor dynamics of an associative memory trained via spike-timing dependent plasticity

Clayton Seitz

May 20, 2021

### Introduction

#### RNNs trained with Hebbian learning rules



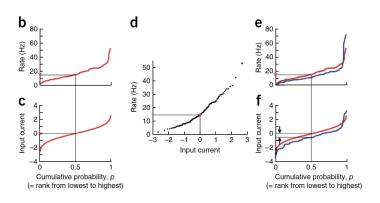
Let  $W_{ij}$  be a matrix of recurrent weights that evolves when stimulated by

$$\xi(oldsymbol{\mu},oldsymbol{\Sigma}) = rac{1}{(2\pi)^{n/2}|oldsymbol{\Sigma}|^{1/2}} \exp{-rac{1}{2}(oldsymbol{r}-oldsymbol{\mu})^Toldsymbol{\Sigma}^{-1}(oldsymbol{r}-oldsymbol{\mu})}$$

1

<sup>&</sup>lt;sup>1</sup>[Peirera and Brunel, Neuron. 2018]

#### Inferring learning rules from firing rate distributions in ITC

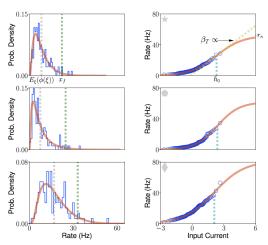


Inferring  $\Delta W_{ij}$  from ITC neurons after presentation of novel and familiar images  $^2$ 

<sup>&</sup>lt;sup>2</sup>[Lim et al., Nature Neuroscience. 2015]

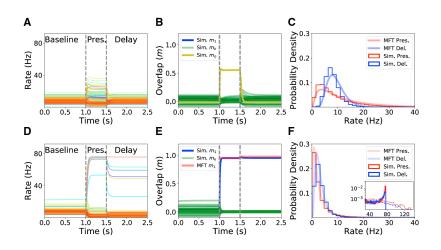
#### Inferring the transfer function from ITC data

All you can really observe is the firing rate distribution. Assume the input currents are Gaussian



<sup>3</sup>[Peirera and Brunel, Neuron. 2018]

#### Presenting novel and familiar stimuli to the network



#### Inferring learning rules from firing rate distributions in ITC

Let the synaptic update be a separable function of the presynaptic  $(\xi_i)$  and postsynaptic  $(\xi_j)$  firing rates

$$\Delta W_{ij} = f(\phi(\xi_i))g(\phi(\xi_j))$$

Evolution of the firing rate for neuron i is

$$\tau \dot{r}_i = -r_i + \phi (I + \sum_{i \neq i} W_{ij} r_j)$$

Synaptic weight updates are weighted according to

$$W_{ij} = C_{ij} \sum f(\phi(\xi_i))g(\phi(\xi_j))$$

5

#### A recurrent firing-rate model

The total input current I to a neuron with n inputs is given by

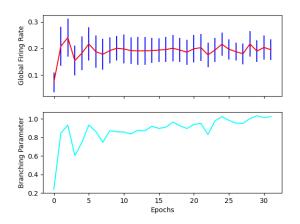
$$r = F(\sum_{n} w_{n} \int_{-\infty}^{t} d\tau K(t - \tau) u(\tau))$$

where  $K = \frac{1}{\tau} \exp(-t/\tau)$ . Say we have a white-noise input  $\boldsymbol{x}$ . The rate-vector  $\boldsymbol{v}$  evolves according to

$$\tau \frac{d\mathbf{v}}{dt} = F(\mathbf{x} + \mathbf{M} \cdot \mathbf{z}) - \mathbf{z}$$

#### Relating branching to firing rates

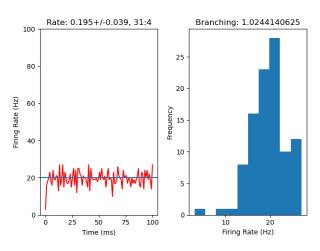
$$\mathcal{L} = \alpha \sum_{t} (r(t) - \hat{r})^2$$



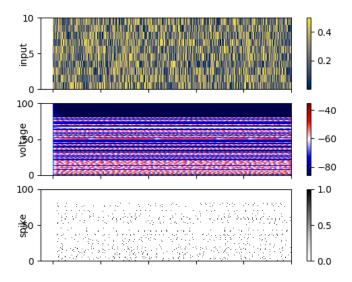
$$p_{ee} = 0.16 \ p_{ie} = 0.318 \ p_{ei} = 0.244 \ p_{ii} = 0.343$$

#### Relating branching to firing rates

$$\mathcal{L} = \alpha \sum_{t} (r(t) - \hat{r})^2$$



#### But optimization of ${\cal L}$ shows a sparse response

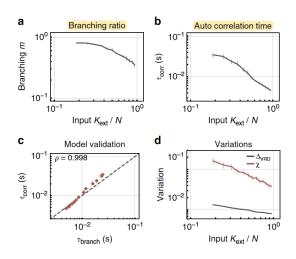


### Balancing excitation and inhibition

The above analysis says nothing about excitatory and inhibitory subpopulations. How can you get critical dynamics in this case?

#### Balancing internal and recurrent inputs

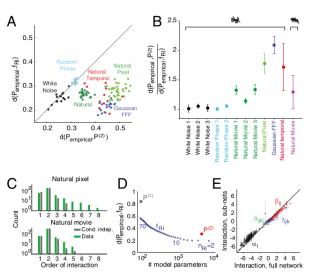
We know something about the balance of excitation and inhibition that gives critical dynamics. What about the balance between input and recurrence? (Cramer et al. 2020)



## Balancing internal and recurrent inputs

#### Higher order correlations

Does the correlation structure of the network depend on the correlation structure of the stimulus?



#### **RNN** Gradients

Say we have a model  $\Phi = (W^0, W^1)$  and want to use gradient descent to train a network to have a target rate or a target branching parameter. The rate and its associated loss for a single unit is

$$r(t) = rac{1}{\Delta t} \int_{t}^{t+\Delta t} d au \langle 
ho( au) 
angle \quad \mathcal{L} = lpha(r-r_0)^2$$

We would like the standard update

$$\Delta W_{ij} = -\eta \frac{\partial \mathcal{L}}{\partial W_{ij}}$$

But it is intractable to compute  $\frac{\partial \mathcal{L}}{\partial W_{ij}}$  since  $\rho(t)$  depends on other neurons through space and time.

#### Factorizing loss gradients for BPTT

BPTT involves unrolling an RNN into a large feedforward network where each layer is a time step.

$$\frac{\partial \mathcal{L}}{\partial W_{ij}^t} = \frac{\partial \mathcal{L}}{\partial h_j^t} \frac{\partial h_j^t}{\partial W_{ij}^t}$$

and the total gradient is a sum over the layers (time)

$$\frac{\partial \mathcal{L}}{\partial W_{ij}^t} = \sum_{t} \frac{\partial \mathcal{L}}{\partial h_j^t} \frac{\partial h_j^t}{\partial W_{ij}^t}$$

#### Deriving e-prop from BPTT

Consider the first term above. The hidden state is computed by some function  $h_j^t = F(z_j^t, h_j^{t-1}, W)$ . Backpropagating through time is then

$$\frac{\partial \mathcal{L}}{\partial h_j^t} = \frac{\partial \mathcal{L}}{\partial z_j^t} \frac{\partial z_j^t}{\partial h_j^t} + \frac{\partial \mathcal{L}}{\partial h_j^{t+1}} \frac{\partial h_j^{t+1}}{\partial h_j^t}$$

which must be expressed recursively

$$\frac{\partial \mathcal{L}}{\partial h_{j}^{t}} = \frac{\partial \mathcal{L}}{\partial z_{j}^{t}} \frac{\partial z_{j}^{t}}{\partial h_{j}^{t}} + \left(\frac{\partial \mathcal{L}}{\partial z_{j}^{t+1}} \frac{\partial z_{j}^{t+1}}{\partial h_{j}^{t+1}} + (...) \frac{\partial h_{j}^{t+2}}{\partial h_{j}^{t+1}}\right) \frac{\partial h_{j}^{t+1}}{\partial h_{j}^{t}}$$

$$= \mathcal{L}_{j}^{t} \frac{\partial z_{j}^{t}}{\partial h_{j}^{t}} + \left(\mathcal{L}_{j}^{t+1} \frac{\partial z_{j}^{t+1}}{\partial h_{j}^{t+1}} + (...) \frac{\partial h_{j}^{t+2}}{\partial h_{j}^{t+1}}\right) \frac{\partial h_{j}^{t+1}}{\partial h_{j}^{t}}$$

$$= \mathcal{L}_{j}^{t} \frac{\partial z_{j}^{t}}{\partial h_{j}^{t}} + \left(\mathcal{L}_{j}^{t+1} \frac{\partial z_{j}^{t+1}}{\partial h_{j}^{t+1}} + (...) \frac{\partial h_{j}^{t+2}}{\partial h_{j}^{t+1}}\right) \frac{\partial h_{j}^{t+1}}{\partial h_{j}^{t}}$$

#### Deriving e-prop from BPTT

Plugging into the original factorization gives

$$\frac{\partial \mathcal{L}}{\partial W_{ij}} = \left(\sum_{t} L_{j}^{t} \frac{\partial z_{j}^{t}}{\partial h_{j}^{t}} + \left(L_{j}^{t+1} \frac{\partial z_{j}^{t+1}}{\partial h_{j}^{t+1}} + (...) \frac{\partial h_{j}^{t+2}}{\partial h_{j}^{t+1}}\right) \frac{\partial h_{j}^{t+1}}{\partial h_{j}^{t}}\right) \frac{\partial h_{j}^{t'}}{\partial W_{ij}}$$

You can then collect terms that are multiplied  $\mathcal{L}_{j}^{t}$ 

$$\frac{\partial \mathcal{L}}{\partial W_{ij}} = \sum_{t} L_{j}^{t} \frac{\partial z_{j}^{t}}{\partial h_{j}^{t}} \left( \sum_{t' \leq t} \left( \prod_{t'} \frac{\partial h_{j}^{t'+1}}{\partial h_{j}^{t'}} \right) \frac{\partial h_{j}^{t'}}{\partial W_{ij}} \right)$$
$$= \sum_{t} L_{j}^{t} \frac{\partial z_{j}^{t}}{\partial h_{j}^{t}} \epsilon_{ij}^{t} = \sum_{t} L_{j}^{t} e_{ij}^{t}$$

#### Constraining the global firing rate distribution

We can define a constraint on the variance of the global firing rate (which simultaneously constrains the mean)

$$\mathcal{L} = \beta(\sigma - \sigma_r)^2$$
  $\sigma = \frac{1}{T} \sum_{r} (r - \mu_r)^2$ 

where we constrain branching by constraining the variance s of the global firing rate where branching  $\to 1$  as  $s \to 0$ .

$$L_j^t = \frac{\partial \mathcal{L}}{\partial z_j^t} = \frac{\partial \mathcal{L}}{\partial \sigma} \frac{\partial \sigma}{\partial n} \frac{\partial n}{\partial z_j^t} = \pm \beta (\sigma - \sigma_r) \cdot (r - \mu_r)$$

Think push-pull. Some variation is necessary for refractoriness.

#### Relating branching to firing rates and information

We have an ensemble of neurons with firing rates  $\mathbf{r} = (r_1, r_2..., r_K)$ 

$$\langle N \rangle = \sum_{k} r_{k}^{t} \Delta t = K \Delta t \langle r \rangle$$

and we would like  $\langle N \rangle$  to be constant. We draw a rate-vector from the joint distribution  $r \sim R(\mu, \Sigma)$ 

$$H(R) = H(P(r_1, r_2, ...r_k)) \leq H\left(\prod_k P(r_k)\right)$$

the upper bound maximizes mutual information at low noise

$$I(X;R) = H(R) - H(R|X)$$

#### Introduction





