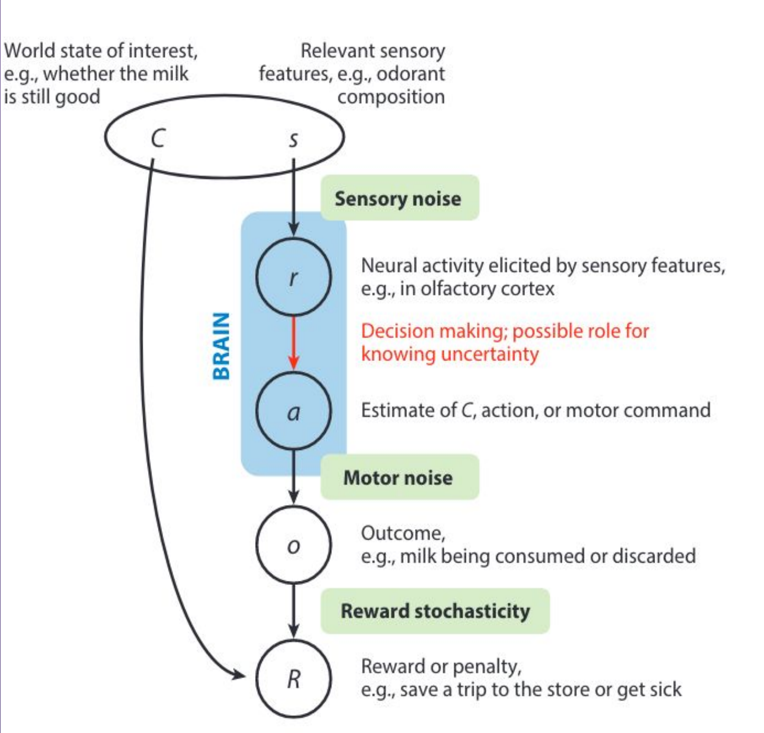
Expressive probabilistic sampling in recurrent neural networks

Can neural circuits sample from complex probability distributions?

APPLIED MATHEMATICS
UNIVERSITY of WASHINGTON

Probabilistic computation is abundant in the brain



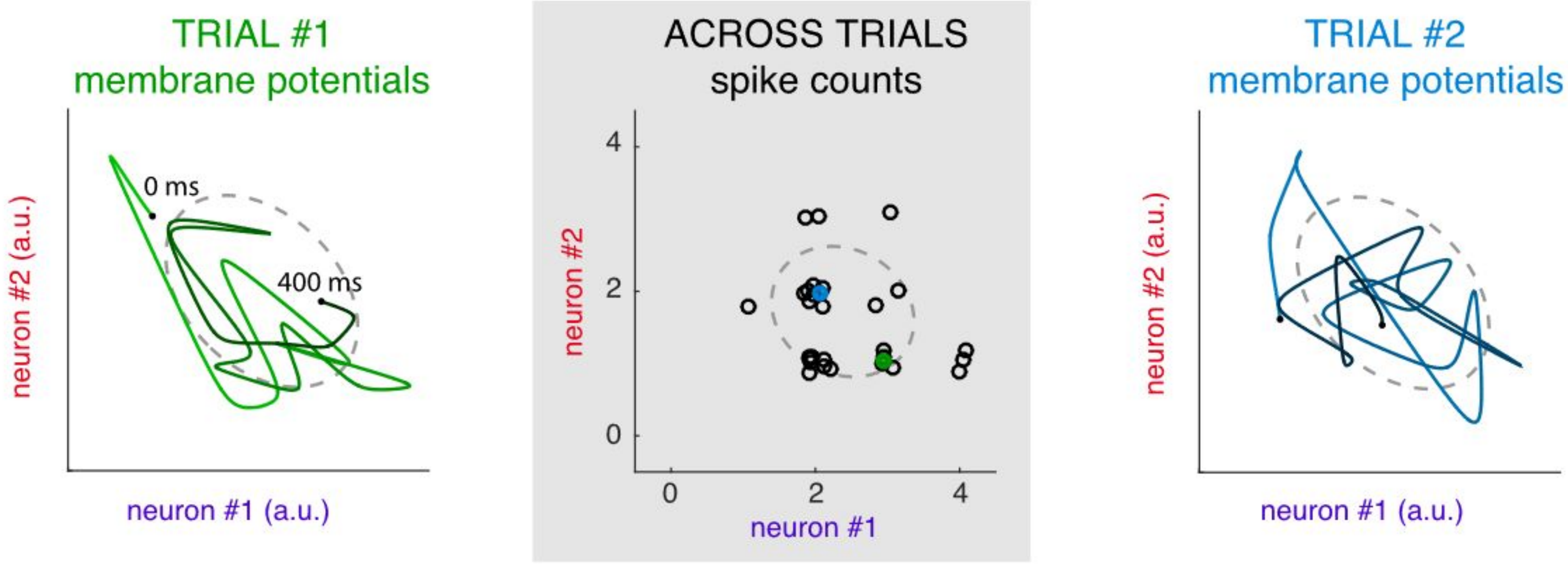
Adapted from Ma & Jazayeri 2014

How do neural circuits represent posterior distributions?

Two Hypothesis:

- Population-based coding
  - Neural responses encode **parameters** of the distribution
  - Examples: probabilistic population codes, distributed distributional codes (DDC)
- Sampling-based coding
  - Neural responses represent **samples** from the distribution
  - Examples: Langevin/Hamiltonian dynamics

Sampling-based coding



Adapted from Orban et al. 2016

Question: If we are able to write the recurrent neural dynamics as a stochastic differential equation, what are the distributions that it can sample from?

Detour: Stochastic differential equation (SDE)

This is a time-homogeneous SDE,

$$dX_t = \underbrace{\mathbf{b}(X_t)dt}_{\text{deterministic}} + \underbrace{\sigma dB_t}_{\text{noise}}$$

There is a corresponding Kolmogorov forward (Fokker-Planck) equation describes how the transition probability density p(x,t) changes with time.

$$\frac{\partial p}{\partial t} = \nabla \cdot (\Sigma \nabla p - \mathbf{b}p), \; \Sigma = \frac{1}{2}\sigma\sigma^T$$

Stationary distribution

A stationary probability distribution of an SDE is one that make the right hand side of the Fokker-Planck equation vanish, i.e. $\nabla \cdot (\Sigma \nabla p - \mathbf{b} p) = 0$

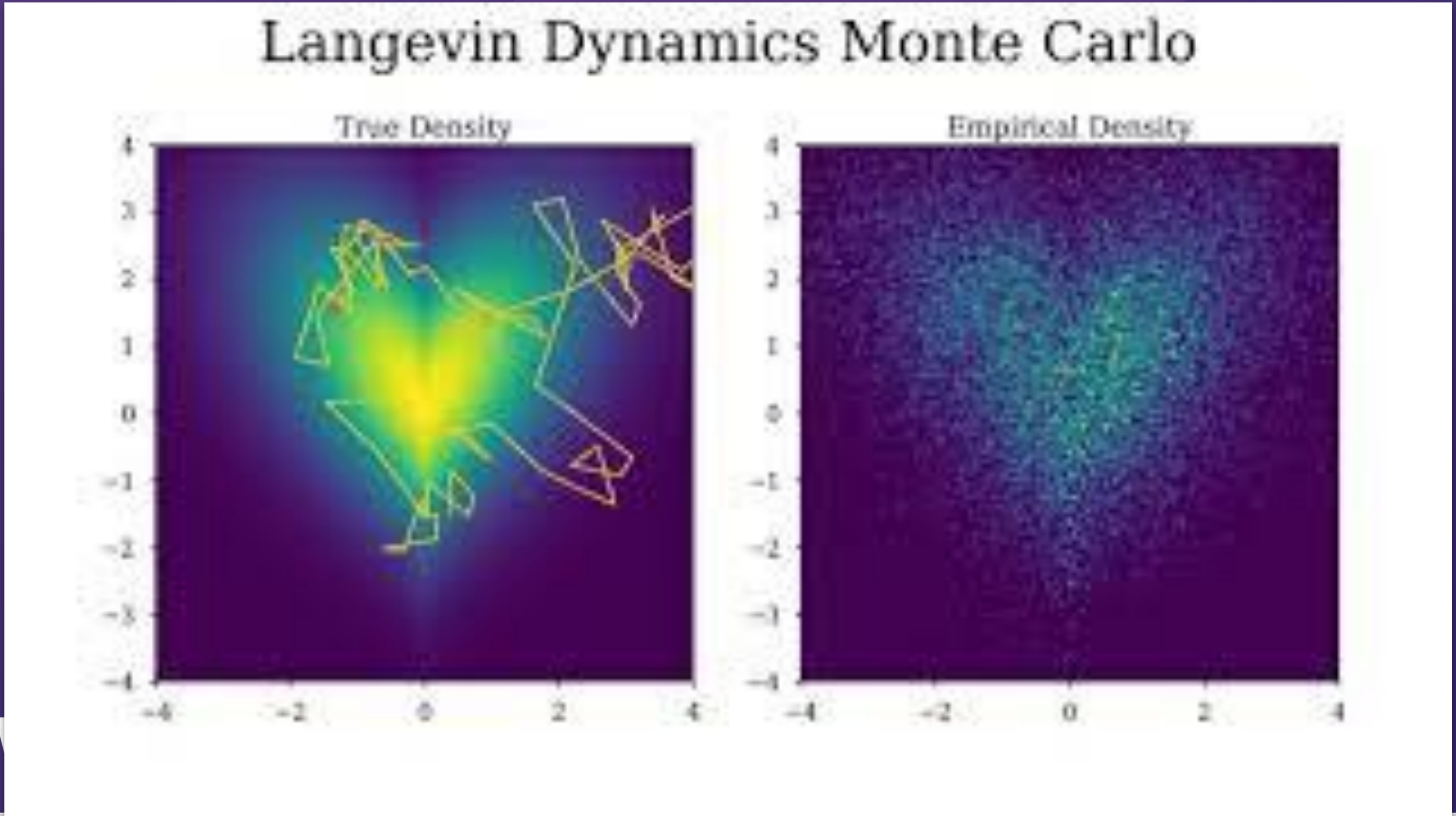
Therefore if we want to sample from the stationary distribution p, we ***hope*** that

$$\mathbf{b} = \Sigma \nabla \log p + p^{-1} G$$ For some G such that $\nabla \cdot G = 0$

An obvious solution is (Langevin dynamics):

$$\mathbf{b} = \Sigma \nabla \log p$$

$$\mathbf{b} = \nabla \log p \qquad p\left(\mathbf{x} = [x_1, x_2]^T\right) \propto \exp\left(-\frac{0.8x_1^2 + \left(x_2 - \sqrt[3]{x_1^2}\right)^2}{4}\right)$$



Ability to implement Langevin dynamics is important

Recall that b is the drift term

$$\mathbf{b}_\theta = \nabla \log p + p^{-1} G$$

With some constraint on G, it can be shown that the function space of $\{\mathbf{b}_\theta\}_\theta$ needs to have at least the same number of basis functions as the function space that $\nabla \log p$ is in.

Equivalent question: For any distribution p, is there a parametrization of the drift term such that $\mathbf{b}_\theta \approx \nabla \log p$?

Neural sampling through lens of SDE

Consider the synaptic current dynamics of a recurrent neural circuit:

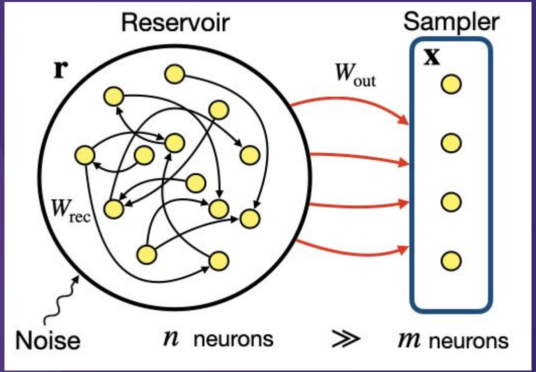
$$d\mathbf{r} = \underbrace{[-\mathbf{r} + W_{rec}\phi(\mathbf{r}) + I]}_{\mathbf{b}_\theta} dt + \sigma dB_t$$

Can the dynamics above alone sample from complex stationary probability distribution?

– No, because $\{\mathbf{b}_\theta\}_\theta$ is only spanned by $f_1(\mathbf{r}) = \mathbf{r}$ and $f_2(\mathbf{r}) = \phi(\mathbf{r})$

RNN with an output layer is a universal Langevin sampler

Reservoir-Sampler network (RSN)

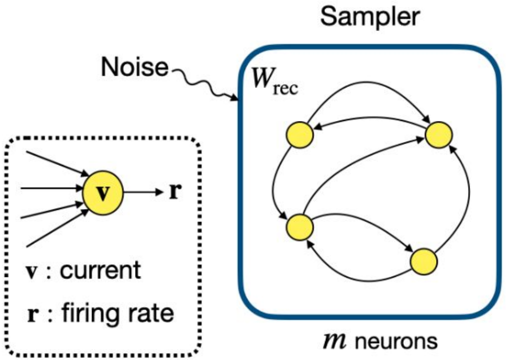


$$d\mathbf{r} = [-\mathbf{r} + \phi(W_{\text{rec}}\mathbf{r} + I)]dt + \sigma dB_t$$
$$\mathbf{x} = W_{\text{out}}\mathbf{r}$$

One can write down the SDE that is only dependent on x:

$$d\mathbf{x} = \left[-\mathbf{x} + W_{\text{out}}\phi\left(\widetilde{W}_{\text{rec}}\mathbf{x} + I\right)\right]dt + W_{\text{out}}\sigma dB_t$$

Sampler-Only Network



Neural Dynamics in SDE

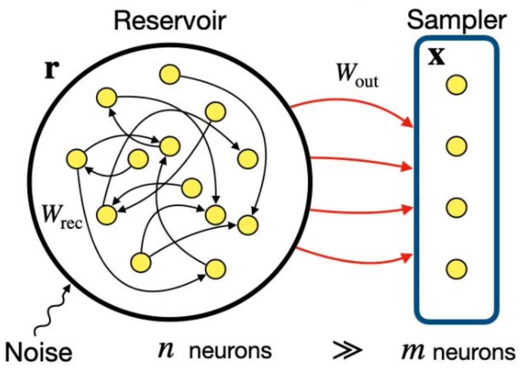
Synaptic current dynamics:

$$\tau d\mathbf{v} = \left[-\mathbf{v} + W_{\text{rec}}\phi(\mathbf{v}) + I\right] dt + \sigma dB_t$$

Firing rate dynamics:

$$\tau d\mathbf{r} = \left[-\mathbf{r} + \phi(W_{\text{rec}}\mathbf{r} + I)\right] dt + \sigma dB_t$$

Reservoir-Sampler Network



Firing rate dynamics:

$$\tau d\mathbf{r} = \left[-\mathbf{r} + \phi(W_{\text{rec}}\mathbf{r} + I)\right] dt + \sigma dB_t$$

$$\mathbf{x} = W_{\text{out}}\mathbf{r}$$

Approximation power

Dynamics	# Func. basis	Fixed?	Closed?
Synaptic current dynamics (sampler-only)	$O(m)$	Yes	Yes
Firing rate dynamics (sampler-only)	$O(m)$	No	No
Firing rate dynamics (reservoir-sampler)	$O(n)$	No	Yes

Theorem 3. *Suppose that we are given a probability distribution with continuously differentiable density function* $p(\mathbf{x}) : \mathbb{R}^m \to \mathbb{R}^+$ *and score function* $\nabla \log p(\mathbf{x})$ *for which there exist constants* $M_1, M_2, a, k > 0$ *such that*

$$p(\mathbf{x}) < M_1 e^{-a\|\mathbf{x}\|} \tag{12}$$

$$\|\nabla \log p(\mathbf{x})\|^2 < M_2 \|\mathbf{x}\|^k \tag{13}$$

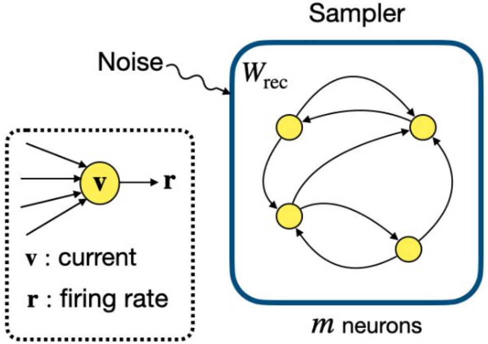
when $\|\mathbf{x}\| > L$ *for large enough* L. *Then for any* $\varepsilon > 0$, *there exists a recurrent neural network whose firing-rate dynamics are given by* (11), *whose recurrent weights, output weights and the diffusion coefficient are given by* $W_{\text{rec}} \in \mathbb{R}^{n\times n}$ *of rank* m, $W_{\text{out}} \in \mathbb{R}^{m\times n}$, *and* $\sigma \in \mathbb{R}^{n\times m}$ *respectively, such that, for a large enough* n, *the score of the stationary distribution of the output units* $s_\theta(\mathbf{x})$ *satisfies* $\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\|\nabla \log p(\mathbf{x}) - s_\theta(\mathbf{x})\|^2] < \varepsilon$.

TL; DR

A stochastic low-rank RNN with an output layer can sample from essentially any distribution

Summary of theoretical results

Sampler-Only Network



Neural Dynamics in SDE

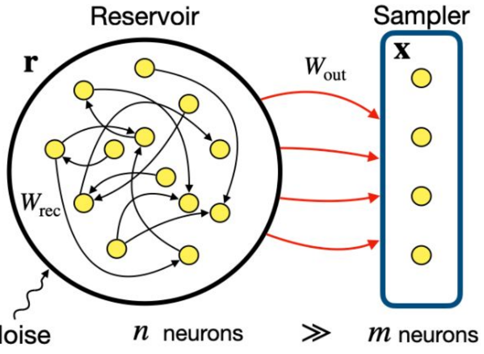
Synaptic current dynamics:

$$\tau d\mathbf{v} = \left[-\mathbf{v} + W_{\text{rec}}\phi(\mathbf{v}) + I\right] dt + \sigma dB_t$$

Firing rate dynamics:

$$\tau d\mathbf{r} = \left[-\mathbf{r} + \phi(W_{\text{rec}}\mathbf{r} + I)\right] dt + \sigma dB_t$$

Reservoir-Sampler Network



Firing rate dynamics:

$$\tau d\mathbf{r} = \left[-\mathbf{r} + \phi(W_{\text{rec}}\mathbf{r} + I)\right] dt + \sigma dB_t$$

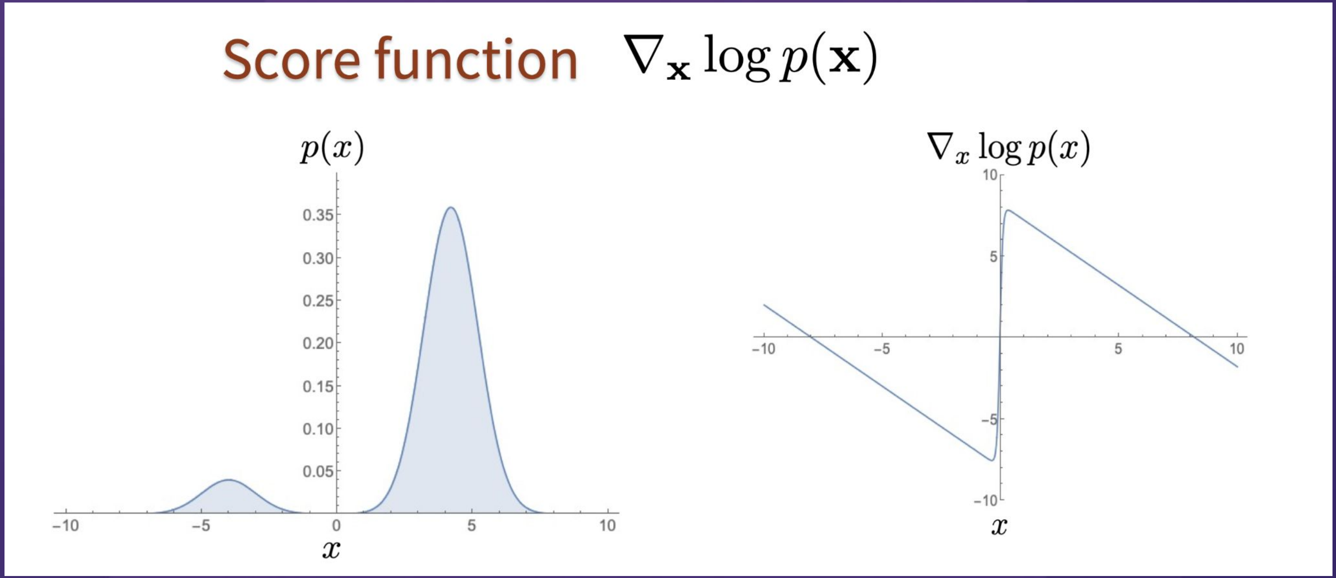
$$\mathbf{x} = W_{\text{out}}\mathbf{r}$$

Approximation power

Dynamics	# Func. basis	Fixed?	Closed?
Synaptic current dynamics (sampler-only)	$O(m)$	Yes	Yes
Firing rate dynamics (sampler-only)	$O(m)$	No	No
Firing rate dynamics (reservoir-sampler)	$O(n)$	No	Yes

How to train such an RNN? (1 / 3)

Score matching, i.e. we would like to minimize $\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\|\nabla \log p(\mathbf{x}) - s_\theta(\mathbf{x})\|^2 \right]$



Adapted from deepgenerativemodels.github.io

How to train such an RNN? (2 / 3)

Denoising Score Matching (perturb the data with noise):

$$\frac{1}{2}\mathbb{E}_{\tilde{\mathbf{x}}\sim\mathcal{N}(\mathbf{x},\,\sigma^2\mathbf{I})}\left[\|\nabla_{\tilde{\mathbf{x}}}\log q_\sigma(\tilde{\mathbf{x}}) - s_\theta(\tilde{\mathbf{x}})\|_2^2\right]$$ (Score matching loss)

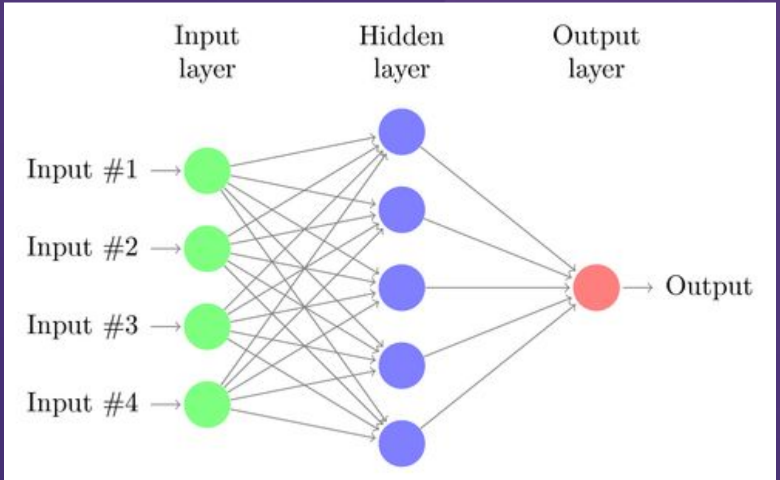
$$= \frac{1}{2}\mathbb{E}_{\mathbf{x}\sim p_{\text{data}(\mathbf{x})},\,\tilde{\mathbf{x}}\sim\mathcal{N}(\mathbf{x},\,\sigma^2\mathbf{I})}\left[\|s_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}}\log q_\sigma(\tilde{\mathbf{x}}|\mathbf{x})\|_2^2\right] + \text{const.}$$ (Denoising score matching loss)

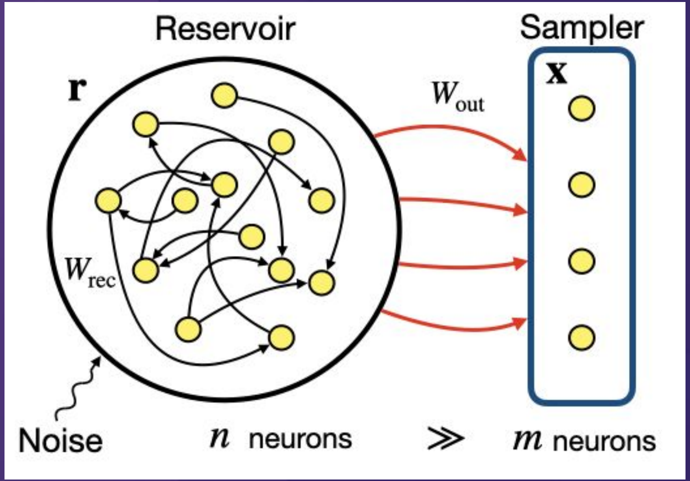
Since we use Gaussian noise, $\nabla_{\tilde{x}}\log q_\sigma(\tilde{x}\,|x) = \frac{x - \tilde{x}}{\sigma^2}$

The noise variance is gradually decreased as training proceeds

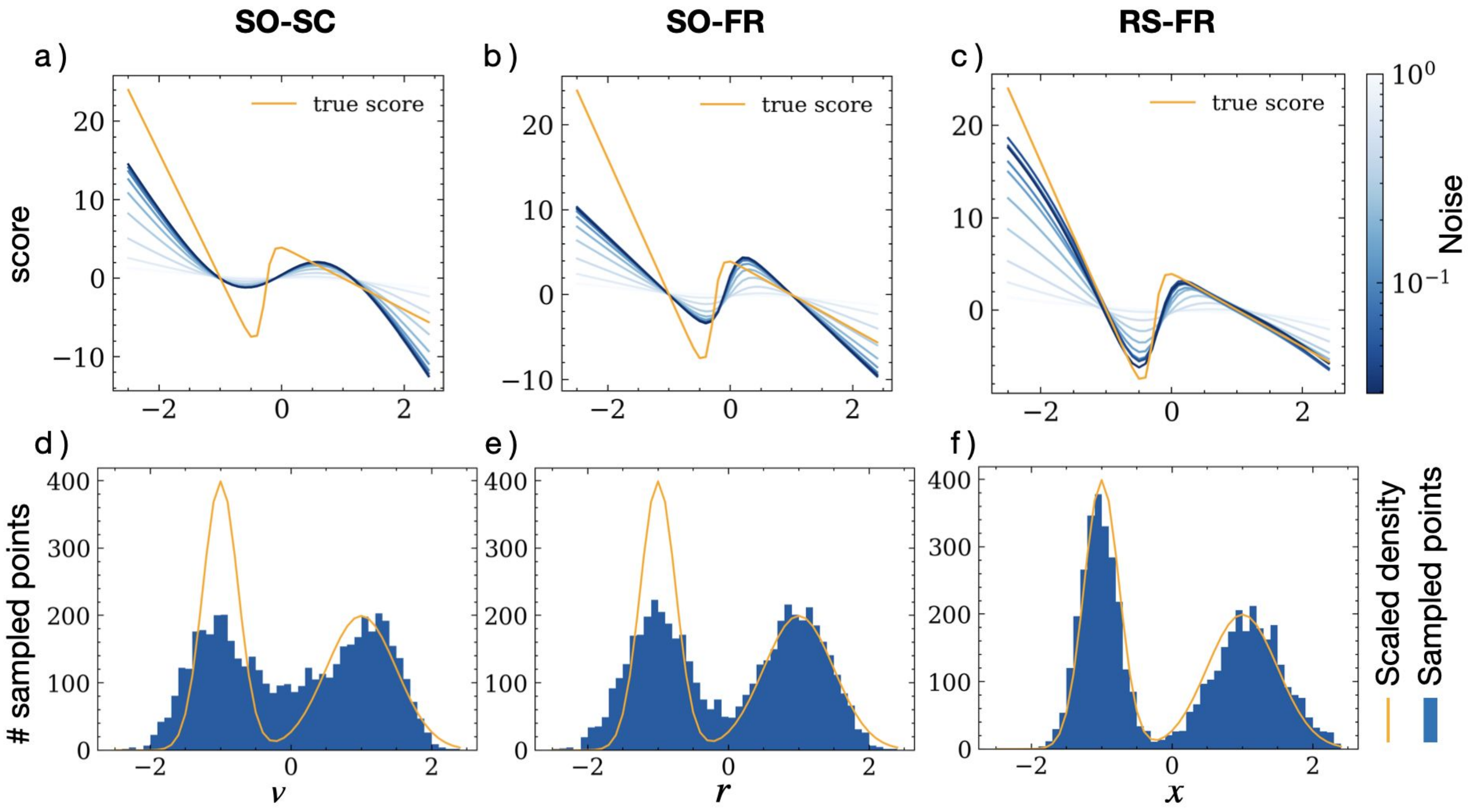
How to train such an RNN? (3/3)

It turns out that we can first train a 2-layer network through Backpropagation, and transform the weights of the feedforward network to the weights of the RNN.



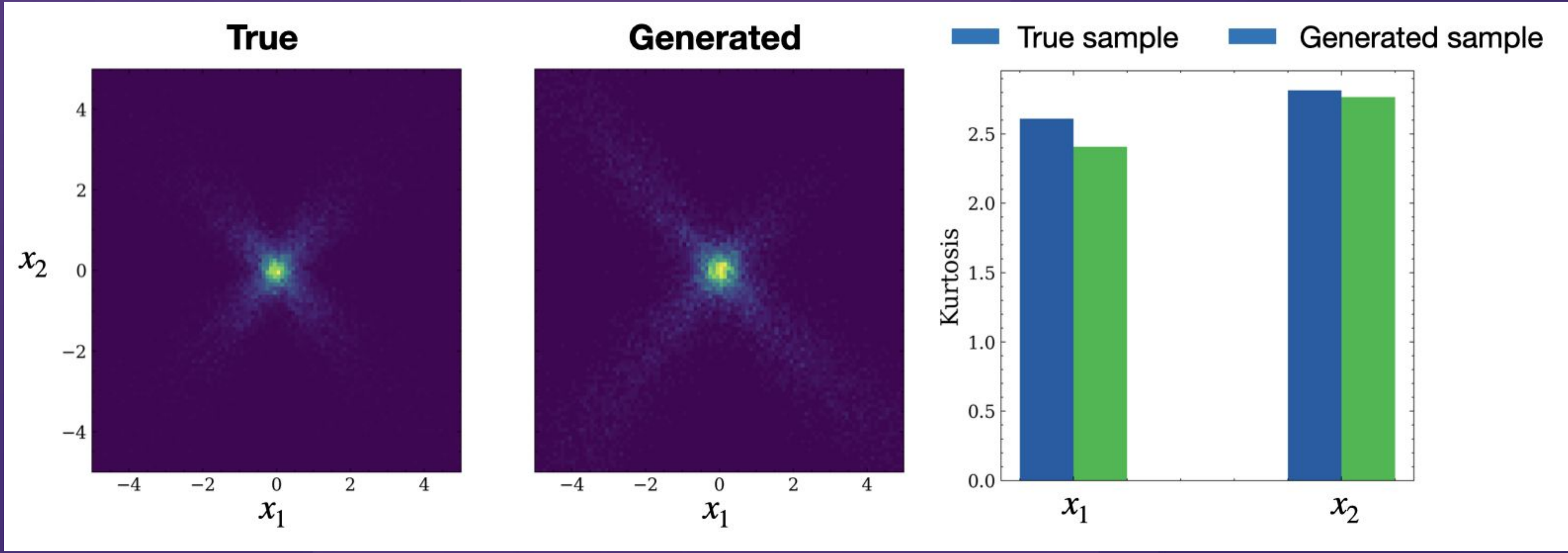


Results - mixture distribution

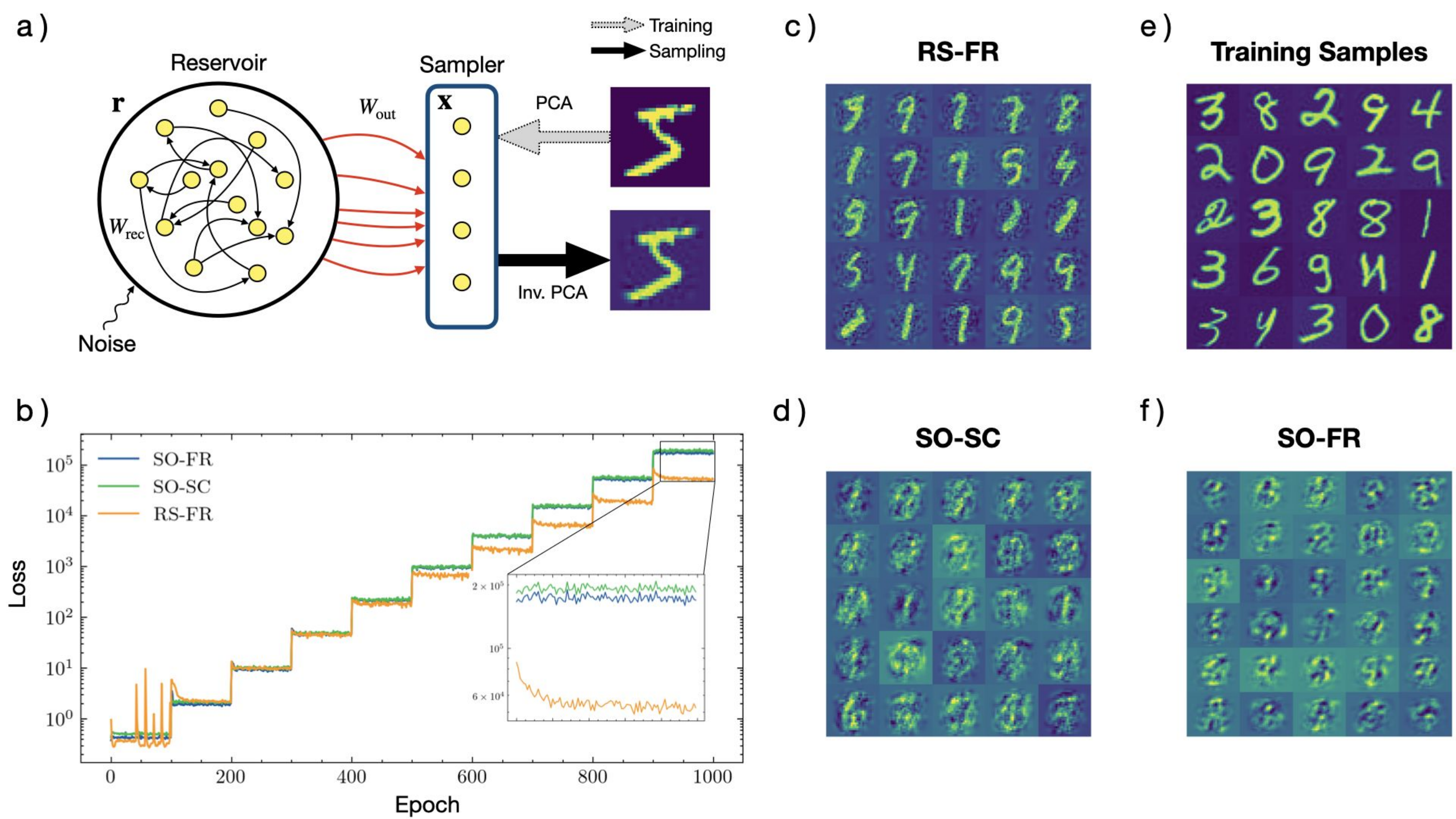


SO: Sampler-only
SC: Synaptic current
FR: Firing rate
RS: Reservoir-Sampler

Results - Heavy-tailed mixture distribution



Results - MNIST image



Discussion

- Our framework builds a bridge between variability in neural dynamics and biophysical neural circuits model

Discussion

- Our framework builds a bridge between variability in neural dynamics and biophysical neural circuits model
- Denoising score matching algorithm gives a way to reverse-engineer the probabilistic neural computation

Discussion

- Our framework builds a bridge between variability in neural dynamics and biophysical neural circuits model
- Denoising score matching algorithm gives a way to reverse-engineer the probabilistic neural computation
- Multiple ways to interpret the Reservoir-Sampler Network:
    - Sampler neurons are a part of large population of neurons or the neurons that are recorded.

Discussion

- Our framework builds a bridge between variability in neural dynamics and biophysical neural circuits model
- Denoising score matching algorithm gives a way to reverse-engineer the probabilistic neural computation
- Multiple ways to interpret the Reservoir-Sampler Network:
  - Sampler neurons are a part of large population of neurons or the neurons that are recorded.
  - Reservoir can be the hidden non-synaptic signaling network
    - pervasive neuropeptidergic signaling (Bargmann and Marder, 2013)
    - extensive aminergic signaling (Bentley et al., 2016)
    - potential extrasynaptic signaling (Yemini et al., 2021)

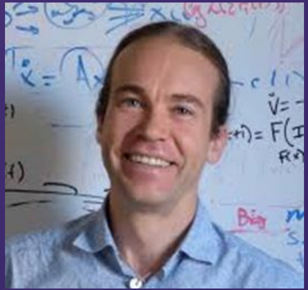
UNIVERSITY of WASHINGTON
COMPUTATIONAL NEUROSCIENCE CENTER

Thank you!

Shirui Chen

Linxing Preston Jiang

Rajesh P.N. Rao



Eric Shea-Brown

We are grateful for discussions with Hong Qian, Bamdad Hosseini and Edger Walker. We gratefully acknowledge the support of the grant NIH BRAIN R01 1RF1DA055669.

APPLIED MATHEMATICS
UNIVERSITY of WASHINGTON