FLATIRON INSTITUTE
Center for Computational Neuroscience

ADAPTIVE DECORRELATION USING GAIN-MODULATING INTERNEURONS

UT Austin

DAVID LIPSHUTZ

31 AUGUST 2023

Lyndon Duong
Flatiron Institute / NYU

David Heeger
NYU

Dmitri “Mitya” Chklovskii
Flatiron Institute / NYU

Eero Simoncelli
Flatiron Institute / NYU

FLATIRON INSTITUTE
Center for Computational Neuroscience

Statistics of natural environments

Statistics of natural environments

Dynamic range

Statistics of natural environments

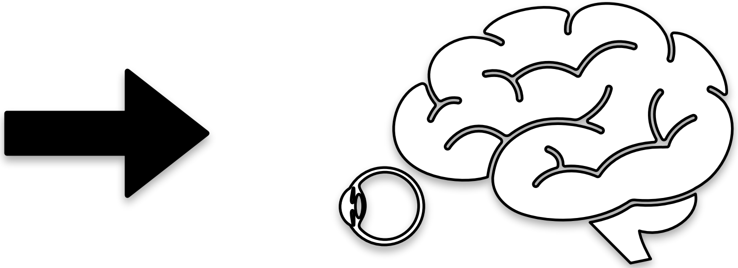
Dynamic range

Statistics of natural environments

Dynamic range

Redundancy

Efficient coding hypothesis



Attneave 1954; Barlow 1961; Laughlin 1981; Atick 1992; van Hateren 1997; Simoncell & Olshausen 2001; …

Efficient coding hypothesis

Neurons encode **maximum information** about the environment using limited resources

Attneave 1954; Barlow 1961; Laughlin 1981; Atick 1992; van Hateren 1997; Simoncell & Olshausen 2001; …

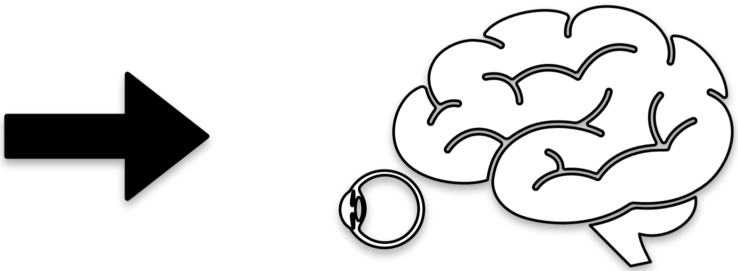
Efficient coding hypothesis

Neurons encode **maximum information** about the environment using limited resources

- Single neurons are adapted to use their entire **dynamic range**

Attneave 1954; Barlow 1961; Laughlin 1981; Atick 1992; van Hateren 1997; Simoncell & Olshausen 2001; …

Efficient coding hypothesis



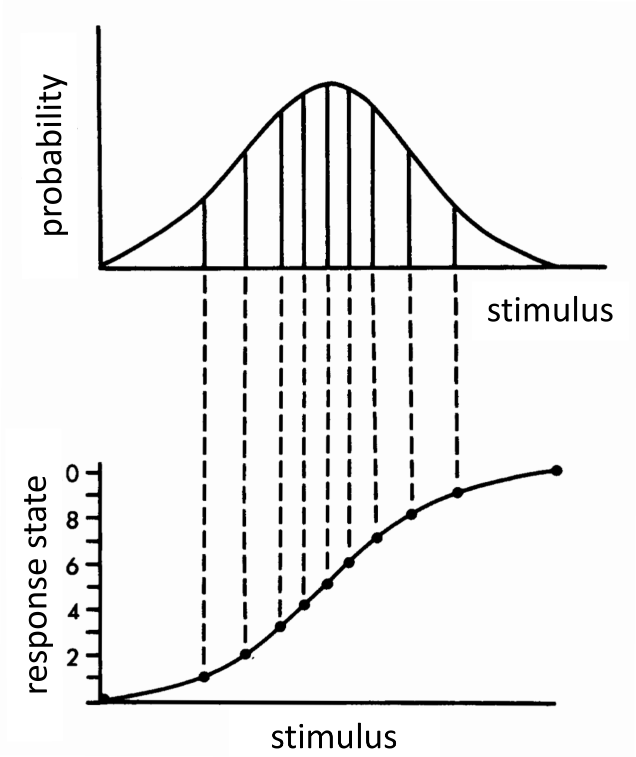
Neurons encode **maximum information** about the environment using limited resources

- Single neurons are adapted to use their entire **dynamic range**
- Neuron populations are adapted to **reduce redundancy**

Attneave 1954; Barlow 1961; Laughlin 1981; Atick 1992; van Hateren 1997; Simoncell & Olshausen 2001; …

Efficient coding in a neuron

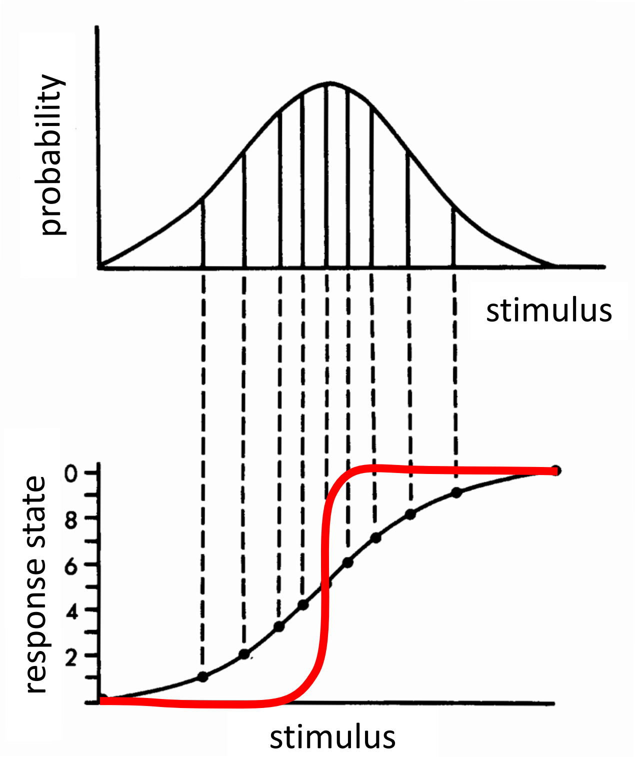
Theory



Laughlin 1981

Efficient coding in a neuron

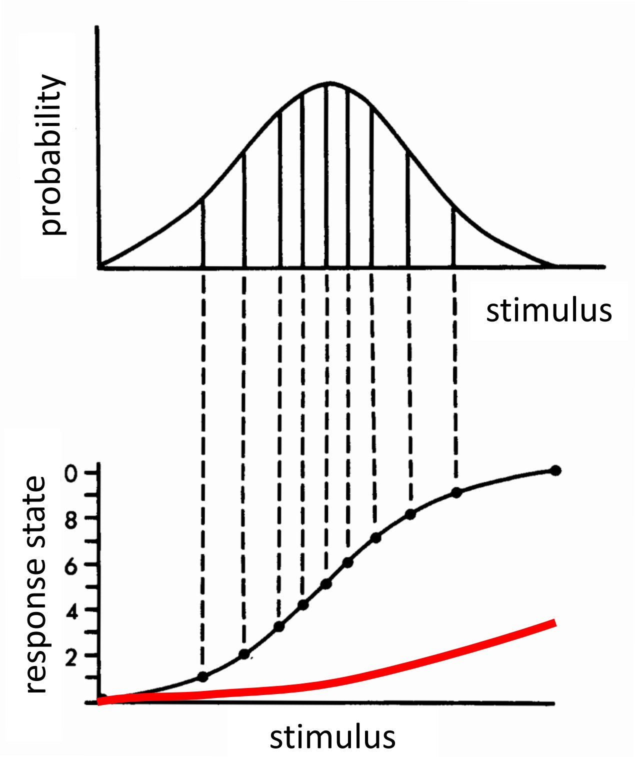
Theory



Laughlin 1981

Efficient coding in a neuron

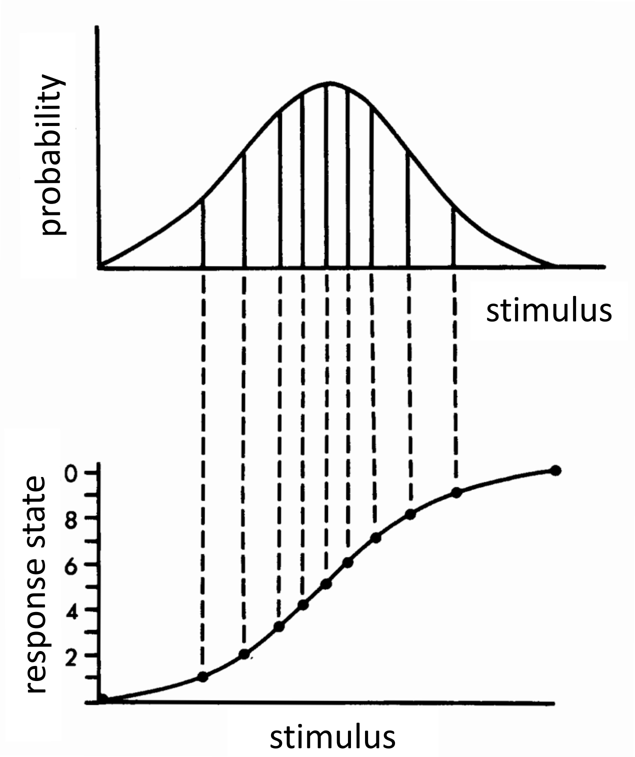
Theory



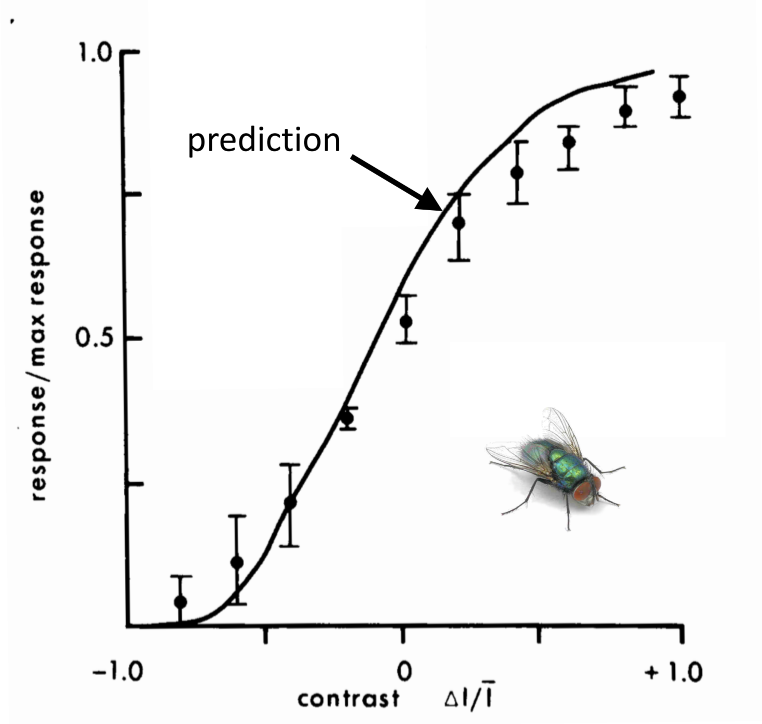
Laughlin 1981

Efficient coding in a neuron

Theory



Experiment



Laughlin 1981

Environments are dynamic!

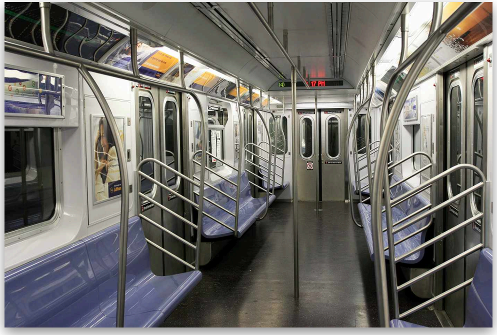
My commute

Environments are dynamic!

My commute

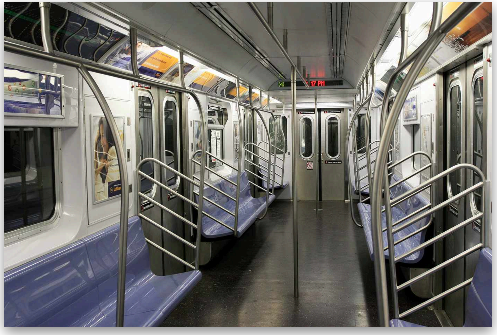
Environments are dynamic!

My commute

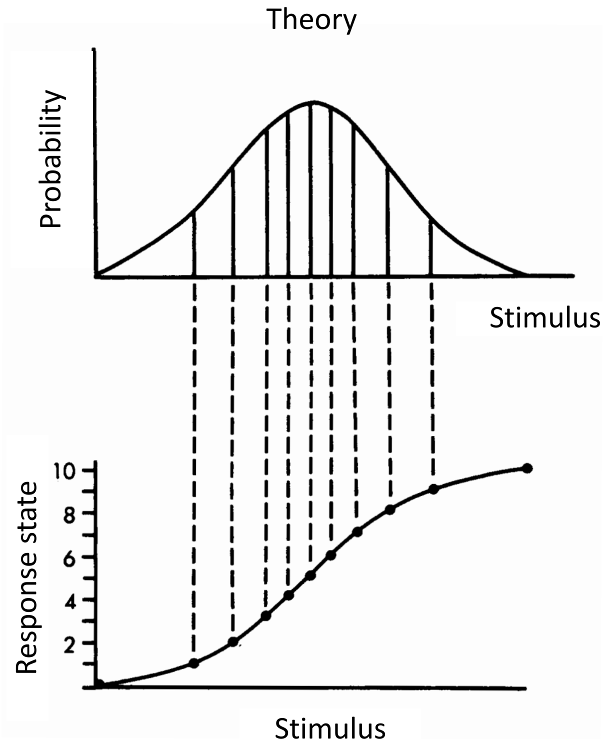


Environments are dynamic!

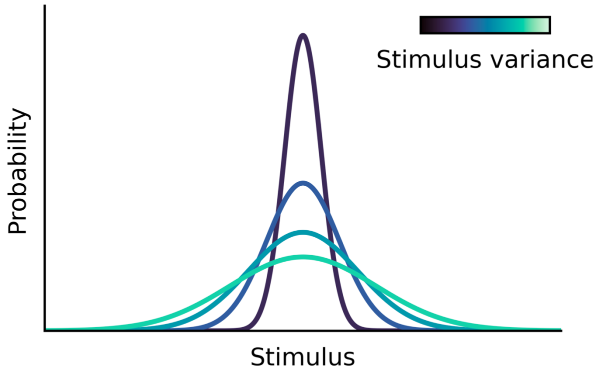
My commute



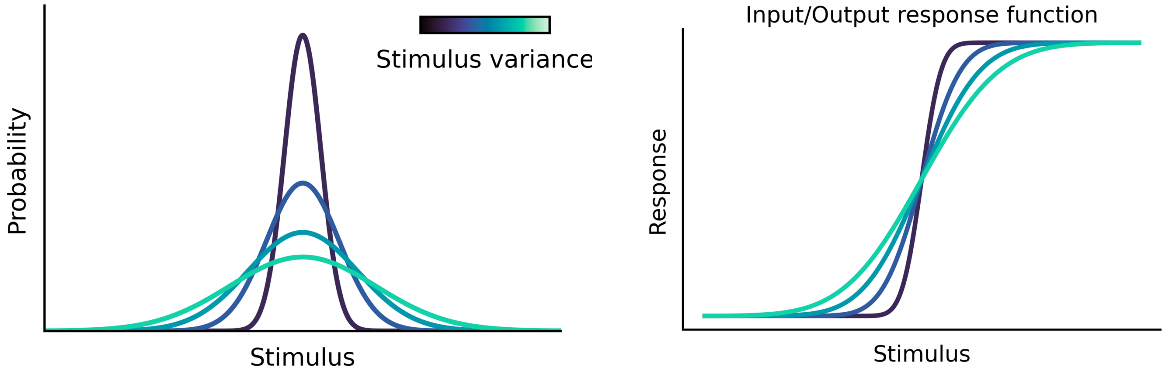
Adaptive efficient coding in single neurons



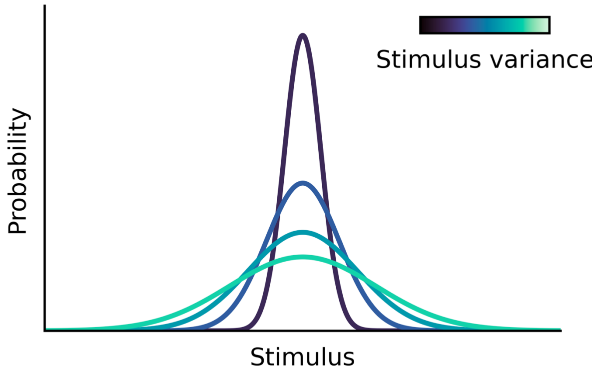
Adaptive efficient coding in single neurons

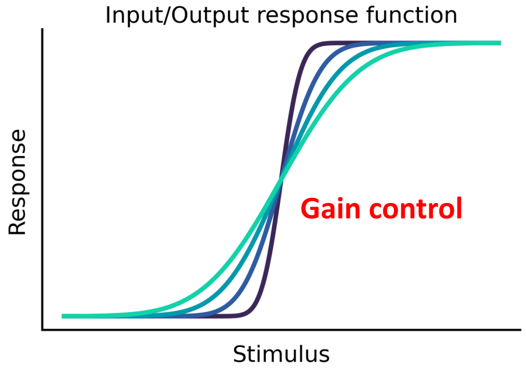


Adaptive efficient coding in single neurons

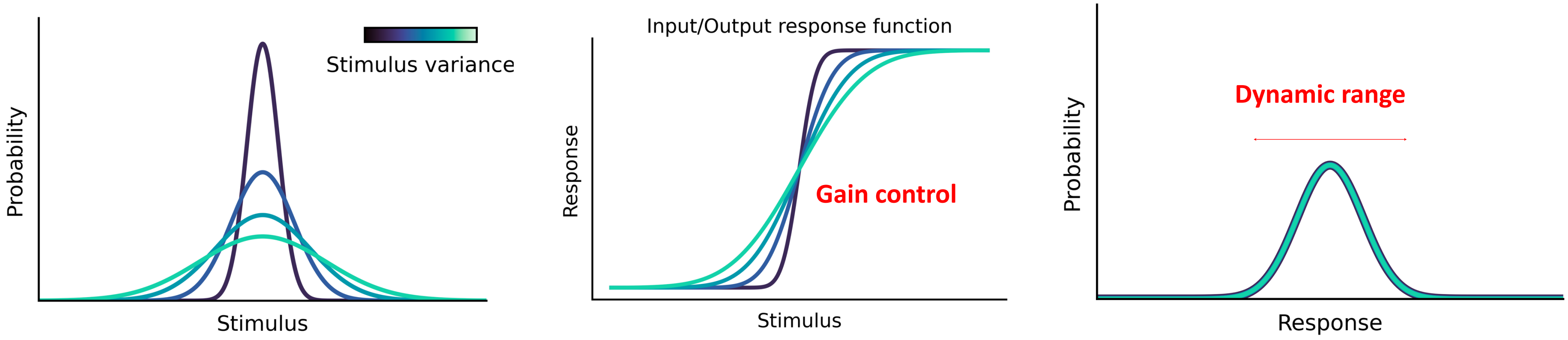


Adaptive efficient coding in single neurons

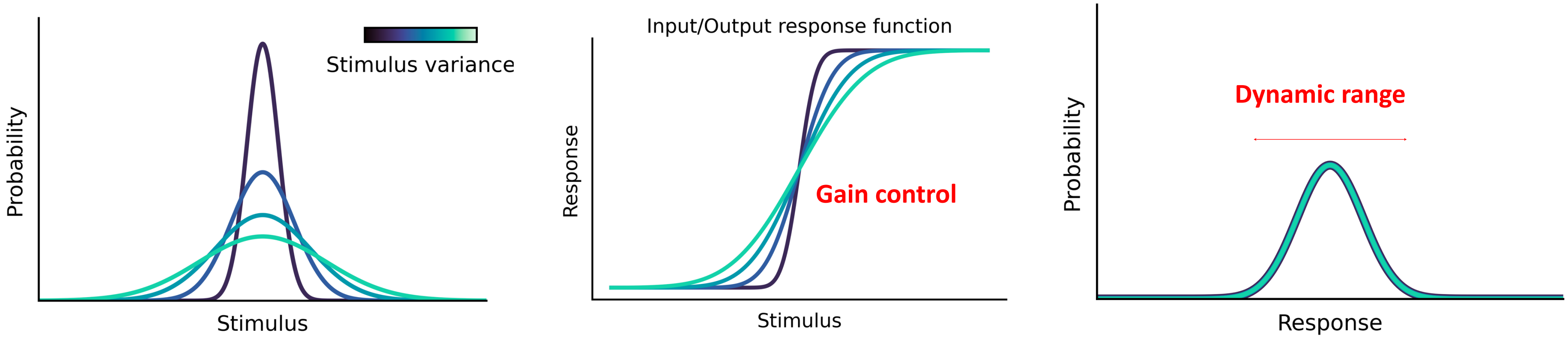




Adaptive efficient coding in single neurons



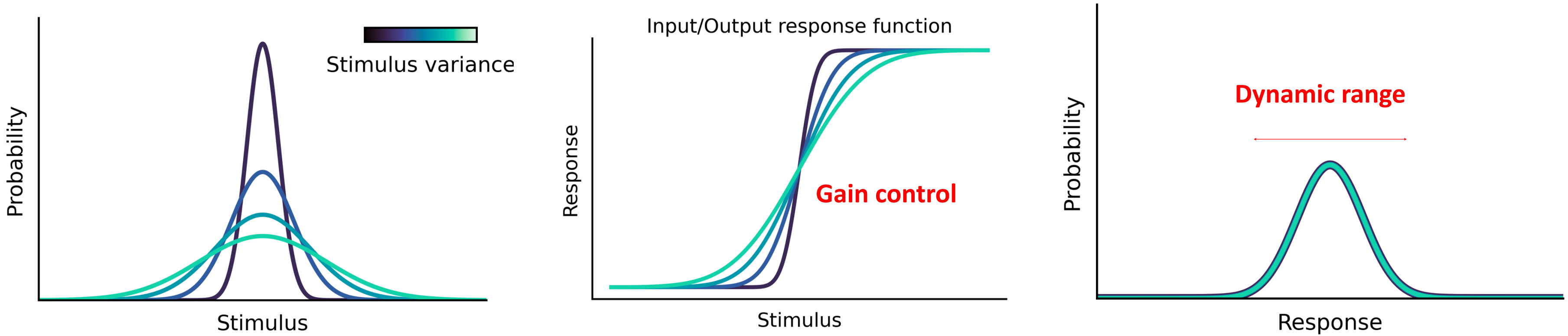
Adaptive efficient coding in single neurons



Reported in:

- Songbird auditory forebrain: [Nagel & Doupe, 2006]
- Fly vision: [Brenner et al. 2000; Fairhall et al., 2001]
- Salamander retina [Chander & Chichilnisky 2001, Baccus & Meister 2002]
- Cat LGN [Mante et al. 2005]
- & more

Adaptive efficient coding in single neurons

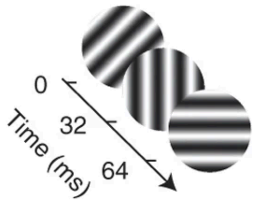


Fast & reversible!
~50ms

Reported in:

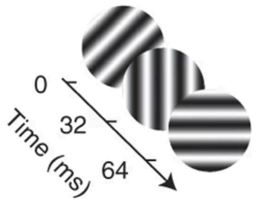
- Songbird auditory forebrain: [Nagel & Doupe, 2006]
- Fly vision: [Brenner et al. 2000; Fairhall et al., 2001]
- Salamander retina [Chander & Chichilnisky 2001, Baccus & Meister 2002]
- Cat LGN [Mante et al. 2005]
- & more

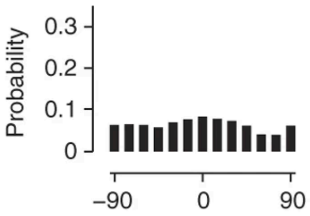
Adaptive decorrelation in a neural population



Benucci et al. 2013

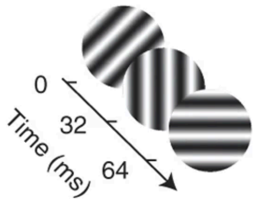
Adaptive decorrelation in a neural population

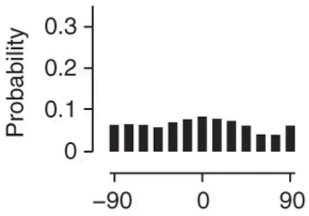




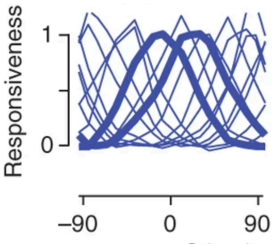
Benucci et al. 2013

Adaptive decorrelation in a neural population



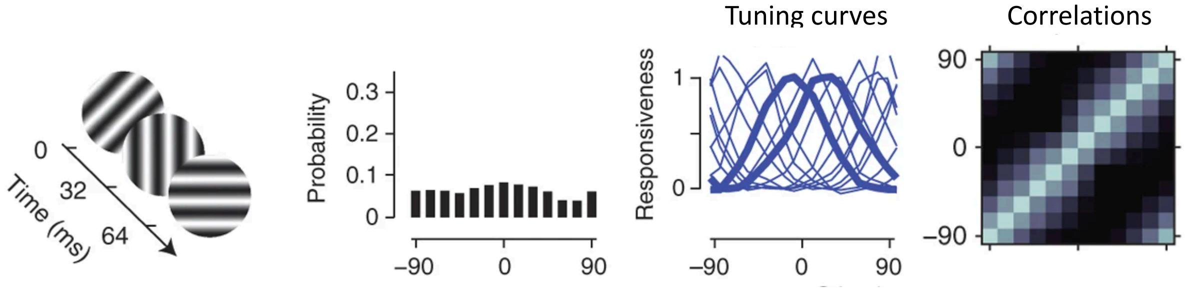


Tuning curves



Benucci et al. 2013

Adaptive decorrelation in a neural population

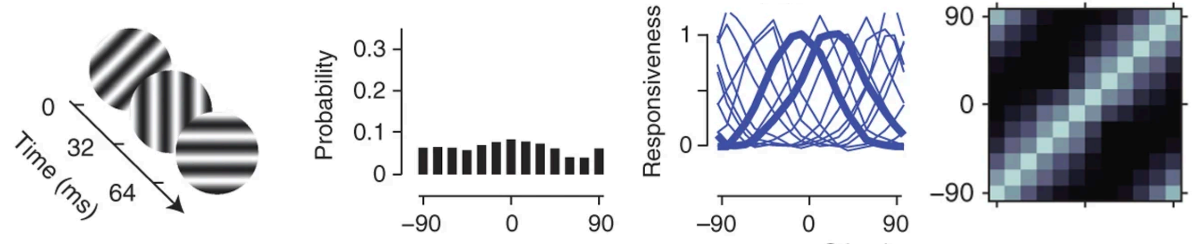


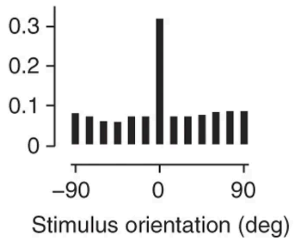
Benucci et al. 2013

Adaptive decorrelation in a neural population

Tuning curves

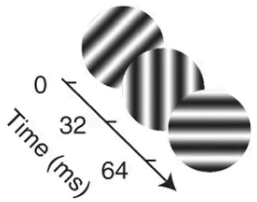
Correlations

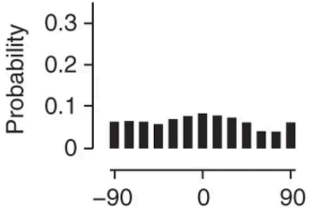




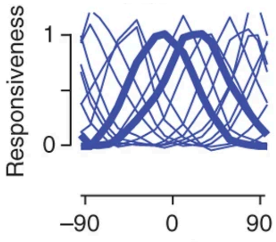
Benucci et al. 2013

Adaptive decorrelation in a neural population

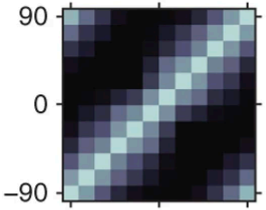




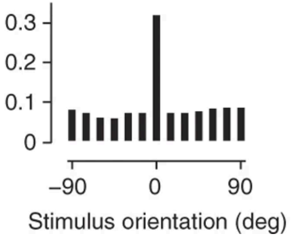
Tuning curves

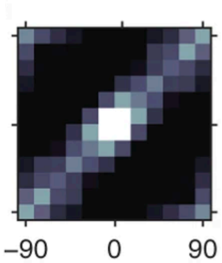


Correlations



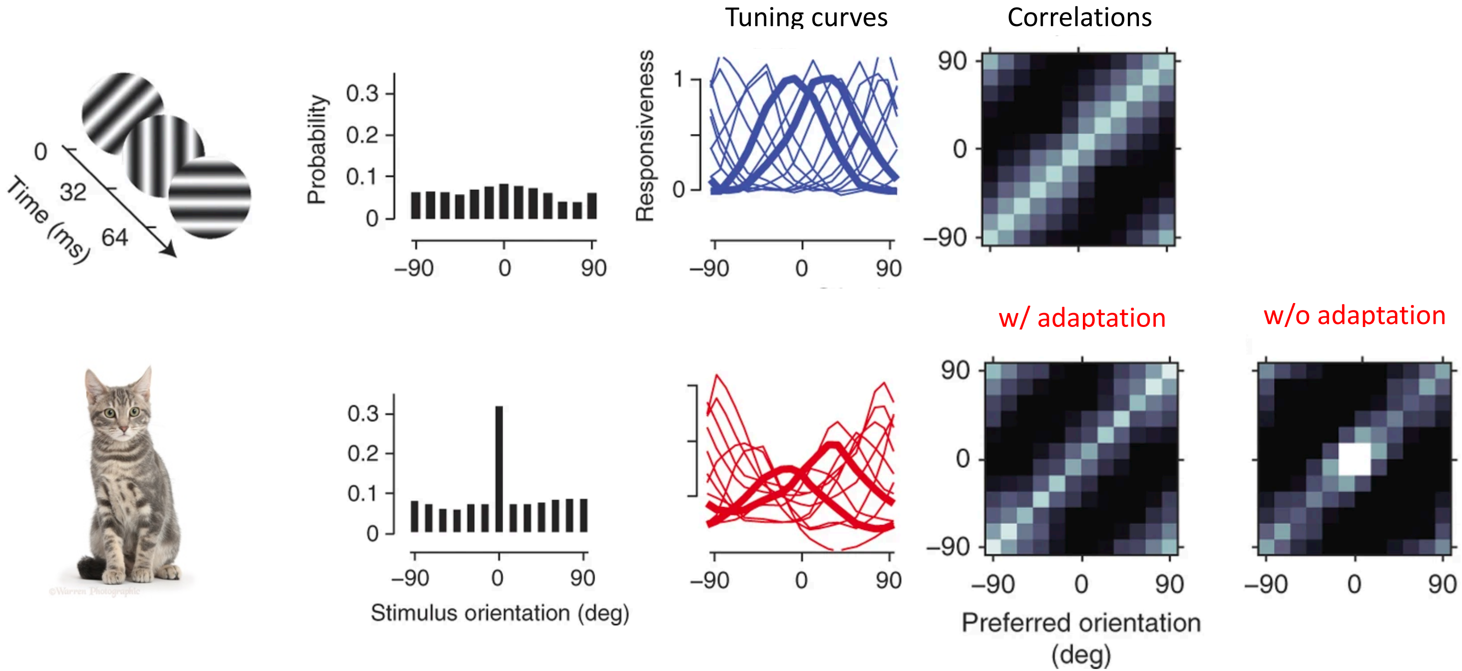
w/o adaptation





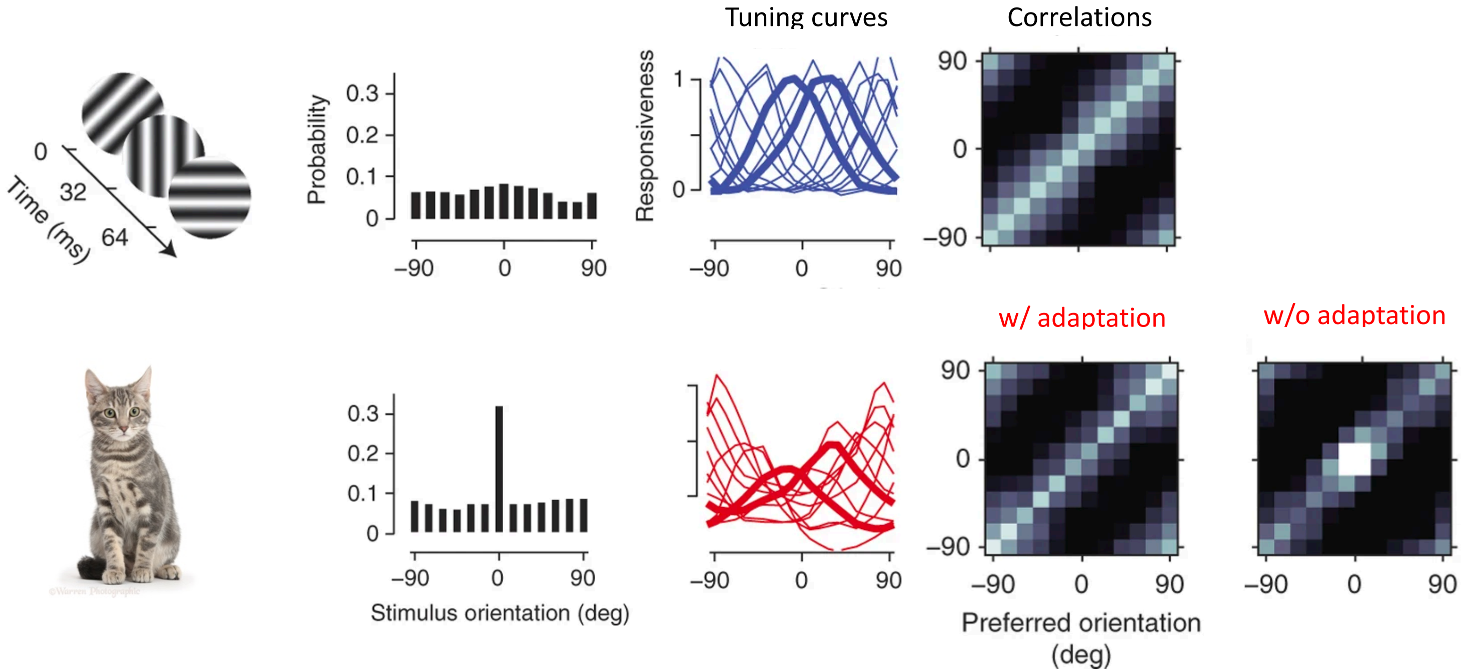
Benucci et al. 2013

Adaptive decorrelation in a neural population



Benucci et al. 2013

Adaptive decorrelation in a neural population

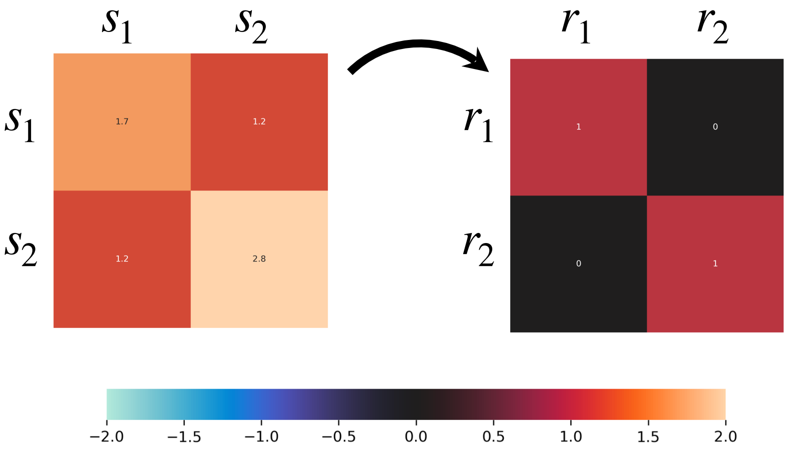


timescale ~ 1.5s / 50 stimuli

Benucci et al. 2013

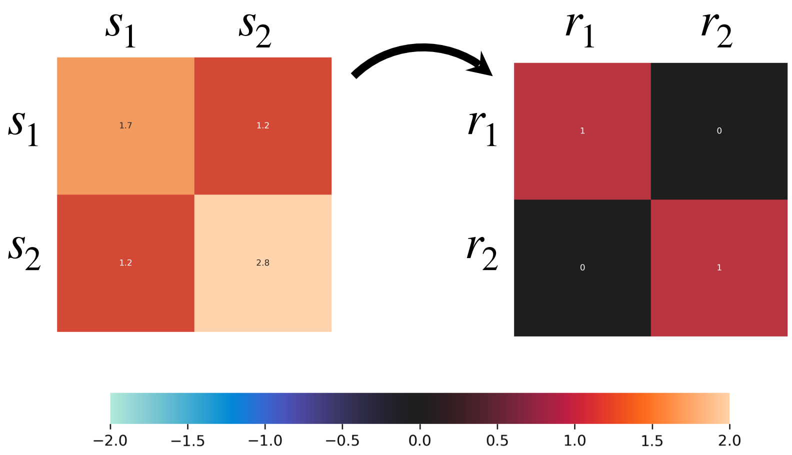
Whitening: normalization + decorrelation

Covariance matrix perspective

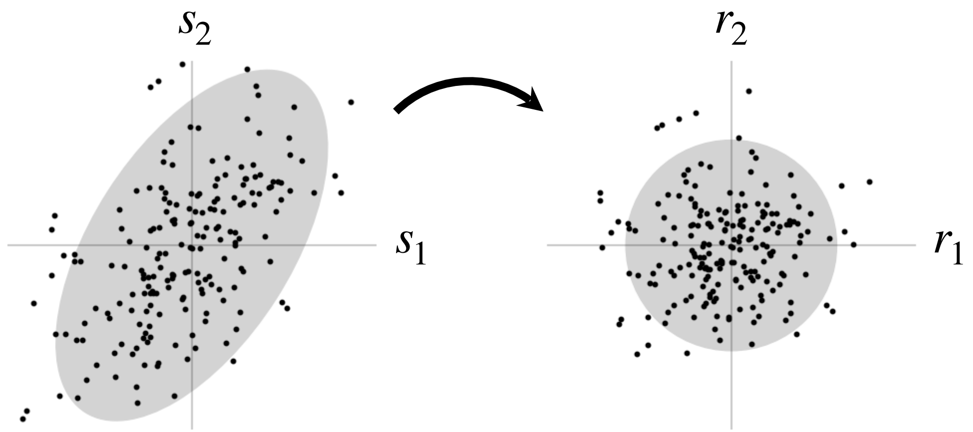


Whitening: normalization + decorrelation

Covariance matrix perspective

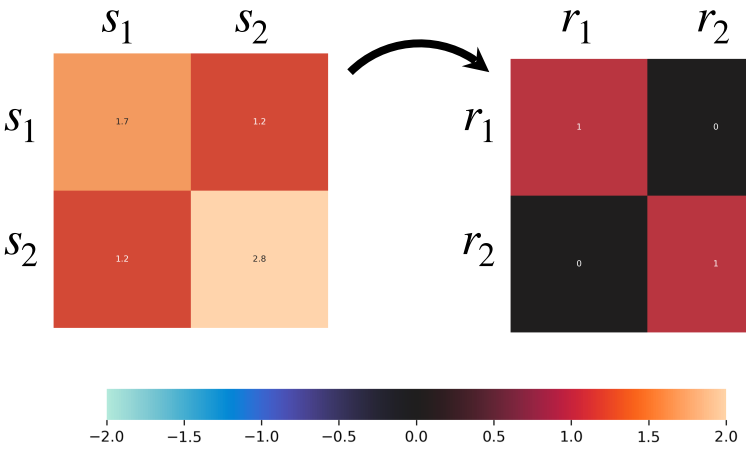


Geometric perspective

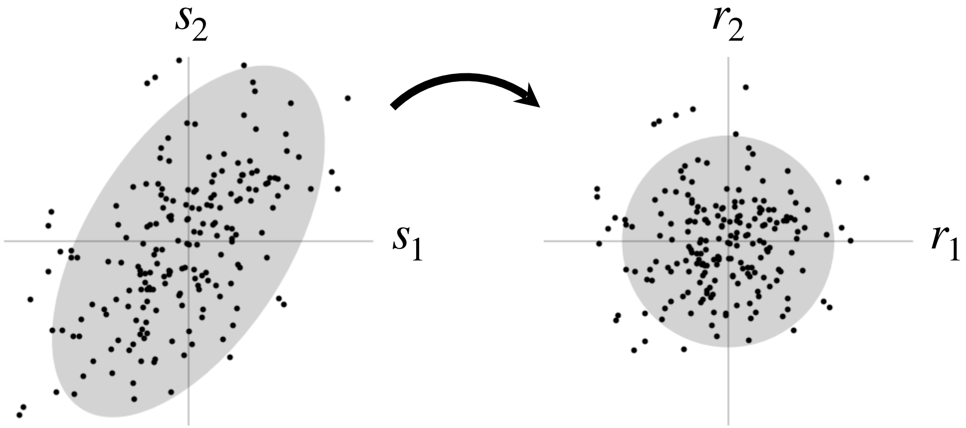


Whitening: normalization + decorrelation

Covariance matrix perspective



Geometric perspective

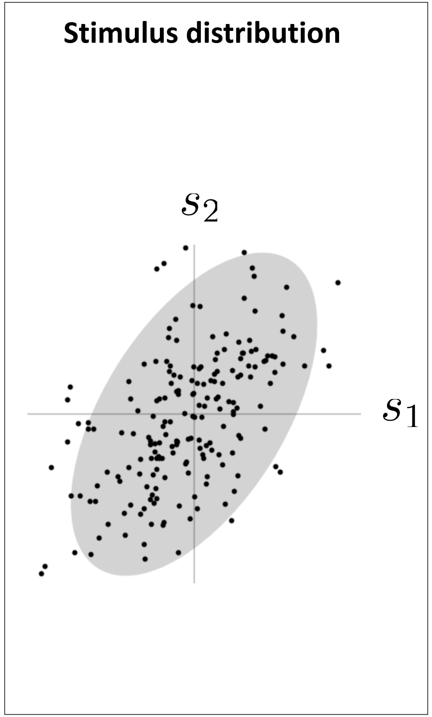


Fundamental to:

- Signal processing (e.g. ICA)
- Machine learning (unsupervised feature learning, self-supervised learning)
- **Neural computation?**
  - Cat V1 [Muller et al. 1999; Benucci et al. 2013]
  - Salamander retina [Pitkow & Meister 2012]
  - Zebrafish olfactory bulb [Wiechart et al. 2010; Wanner & Friedrich 2020]
  - Mouse olfactory bulb [Giridhar et al. 2011; Gschwend et al. 2015]

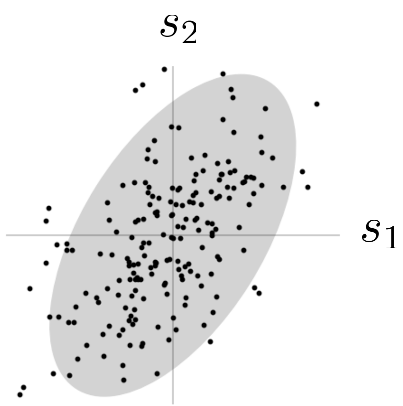
Gain modulation in neural populations

Gain modulation in neural populations

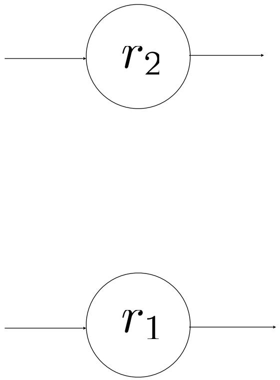


Gain modulation in neural populations

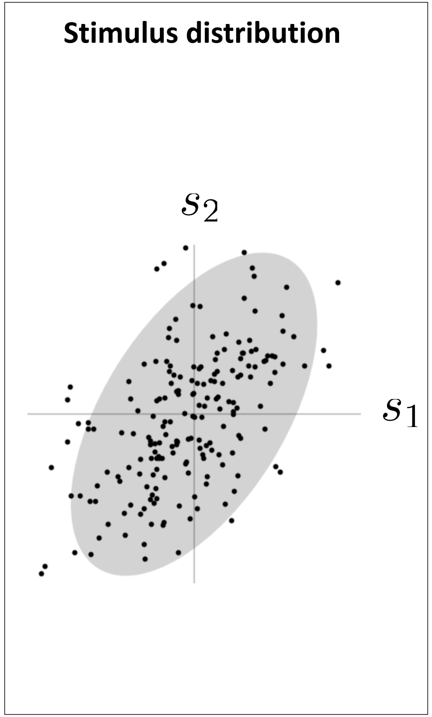
Stimulus distribution

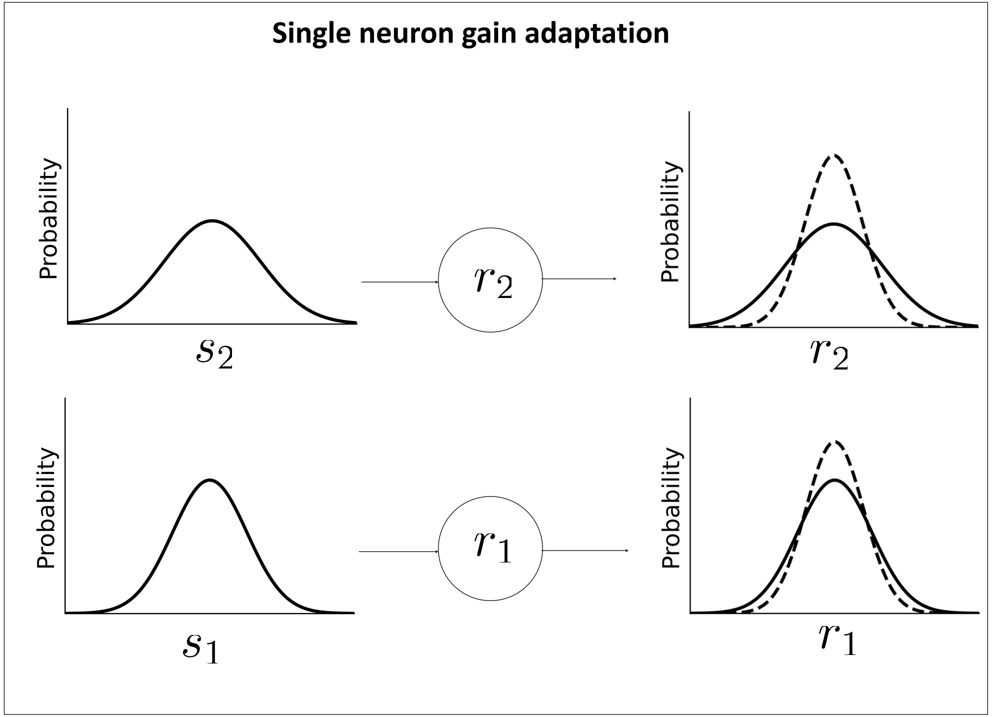


Single neuron gain adaptation

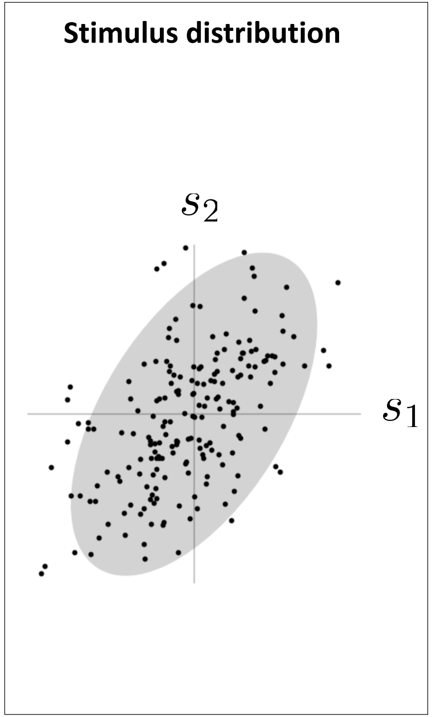


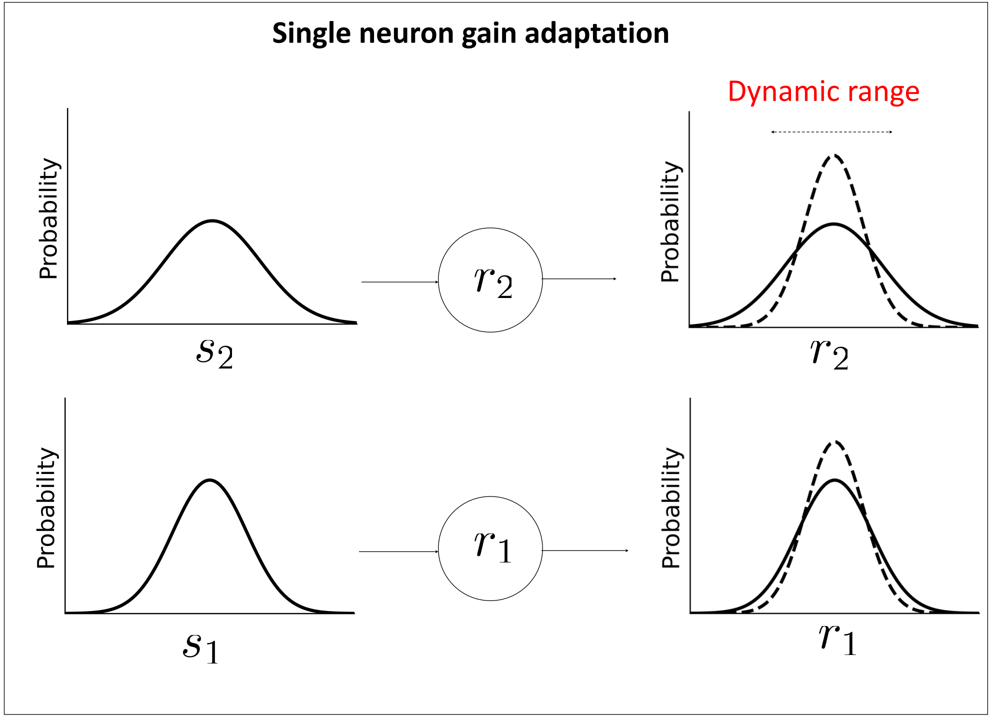
Gain modulation in neural populations



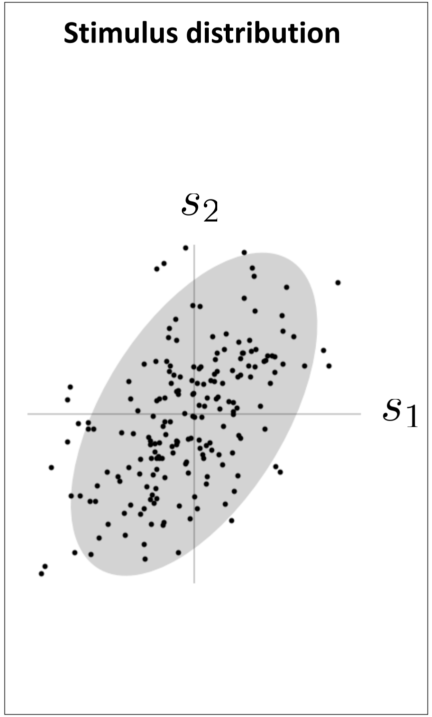


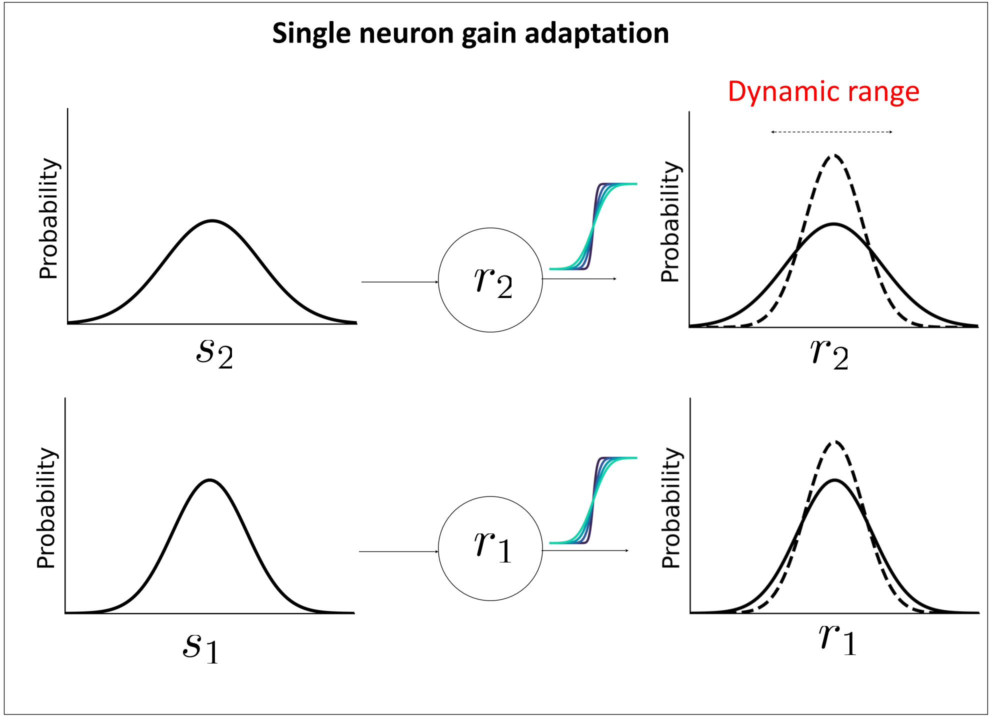
Gain modulation in neural populations



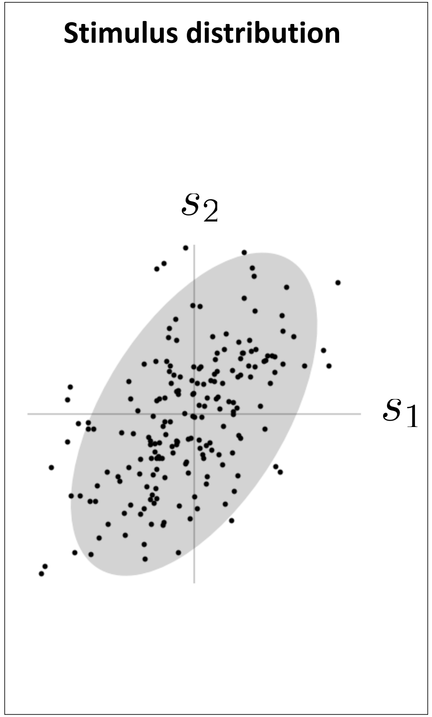


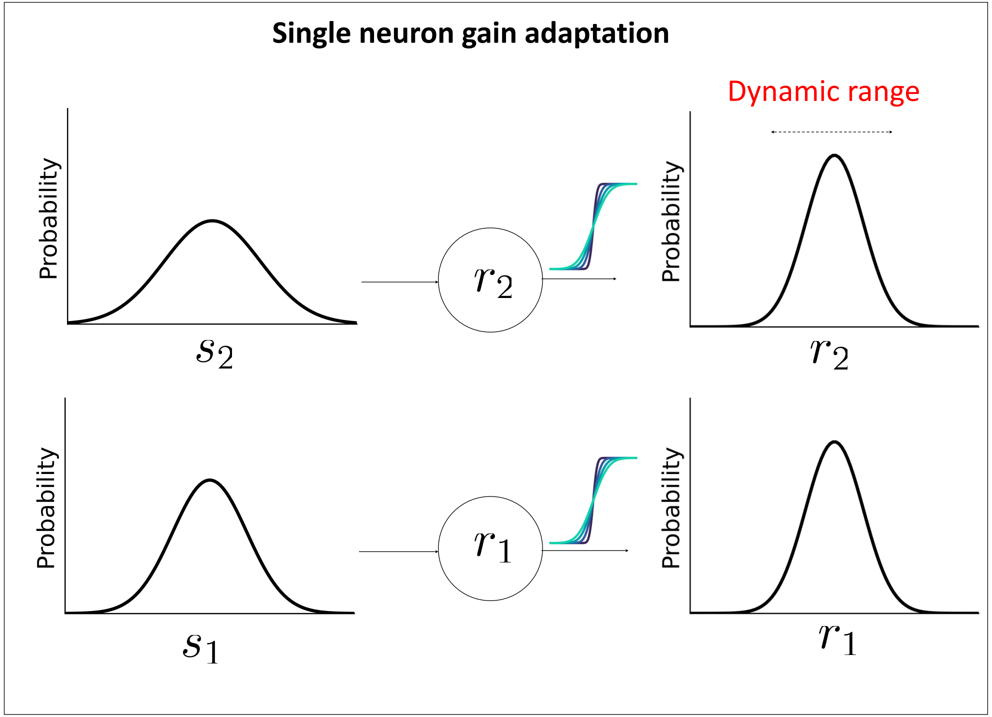
Gain modulation in neural populations



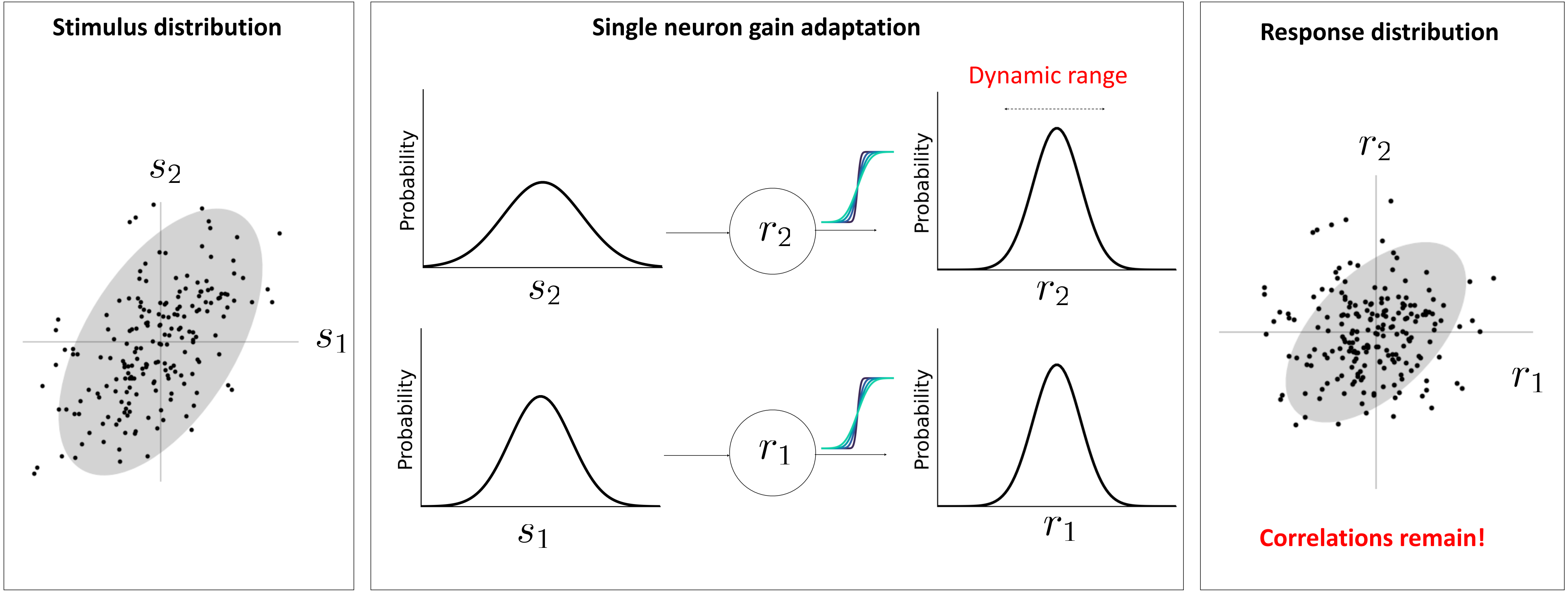


Gain modulation in neural populations





Gain modulation in neural populations

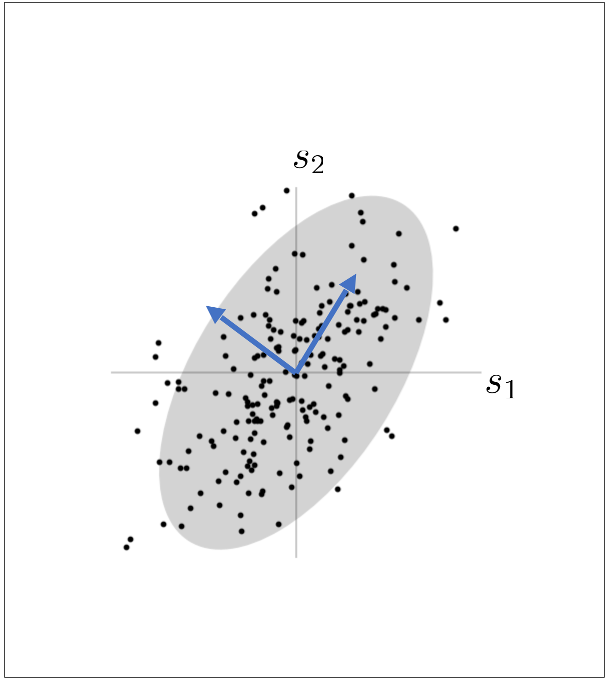


Existing adaptive neural network models

Wick et al. 2010; Pehlevan et al. 2015; Chapochnikov et al. 2023; …

Existing adaptive neural network models

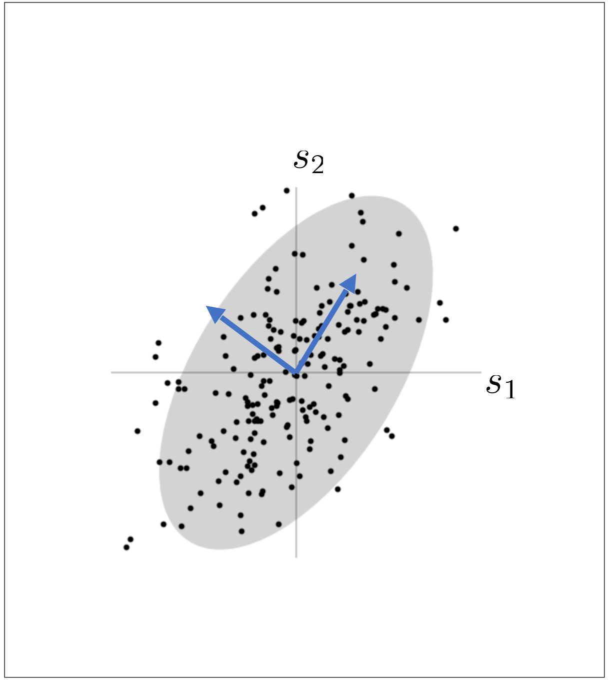
Stimulus distribution



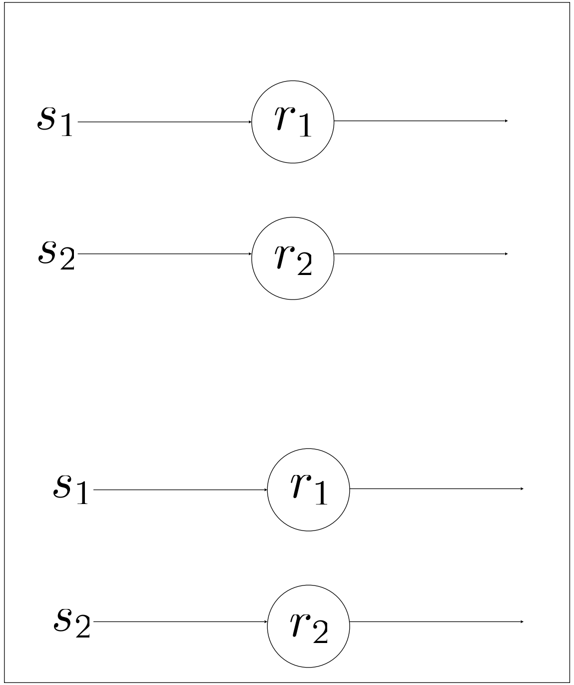
Wick et al. 2010; Pehlevan et al. 2015; Chapochnikov et al. 2023; …

Existing adaptive neural network models

Stimulus distribution



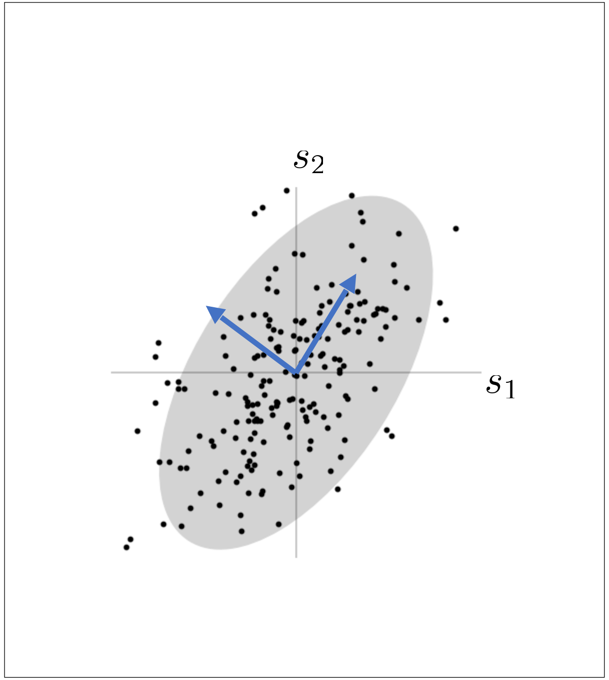
Models



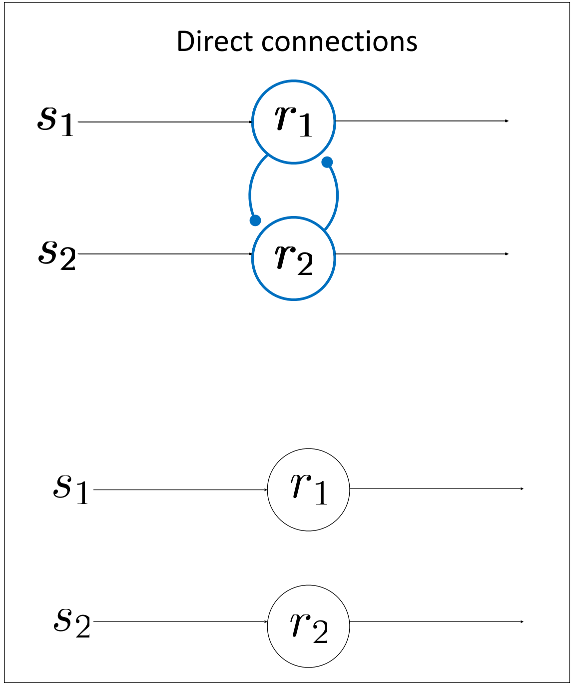
Wick et al. 2010; Pehlevan et al. 2015; Chapochnikov et al. 2023; …

Existing adaptive neural network models

Stimulus distribution



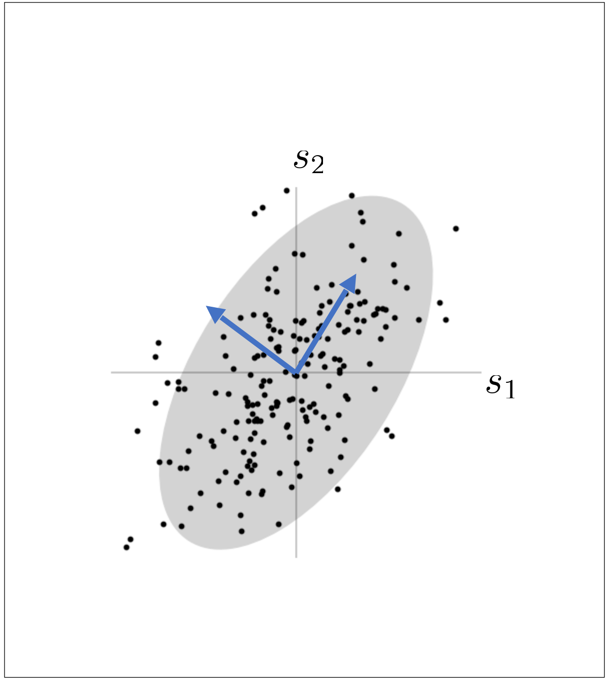
Models



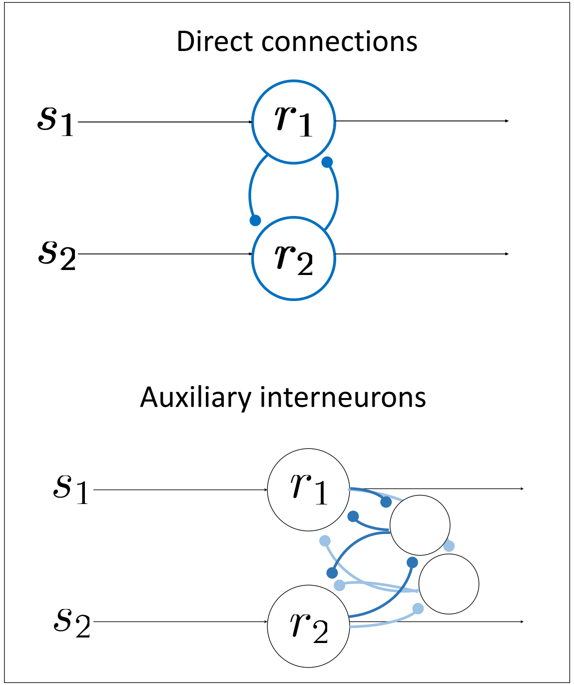
Wick et al. 2010; Pehlevan et al. 2015; Chapochnikov et al. 2023; …

Existing adaptive neural network models

Stimulus distribution



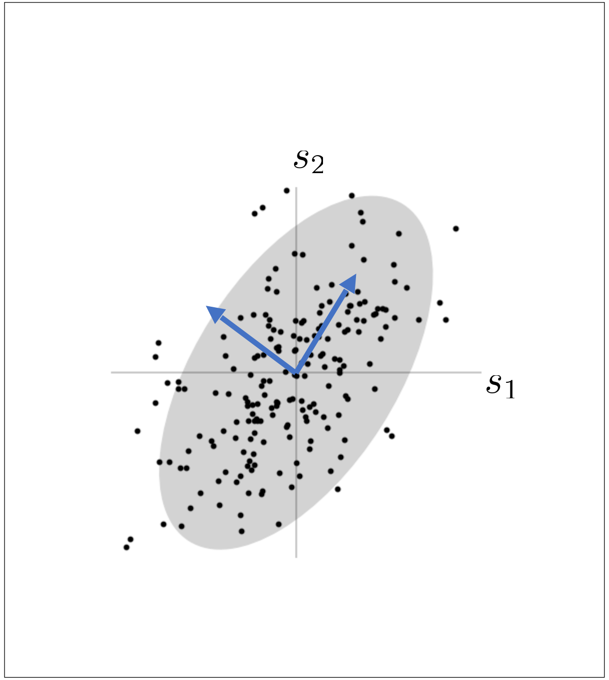
Models



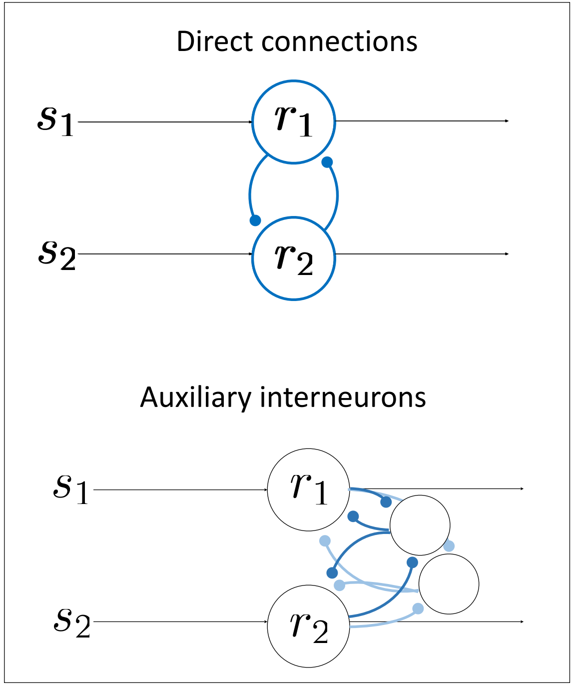
Wick et al. 2010; Pehlevan et al. 2015; Chapochnikov et al. 2023; …

Existing adaptive neural network models

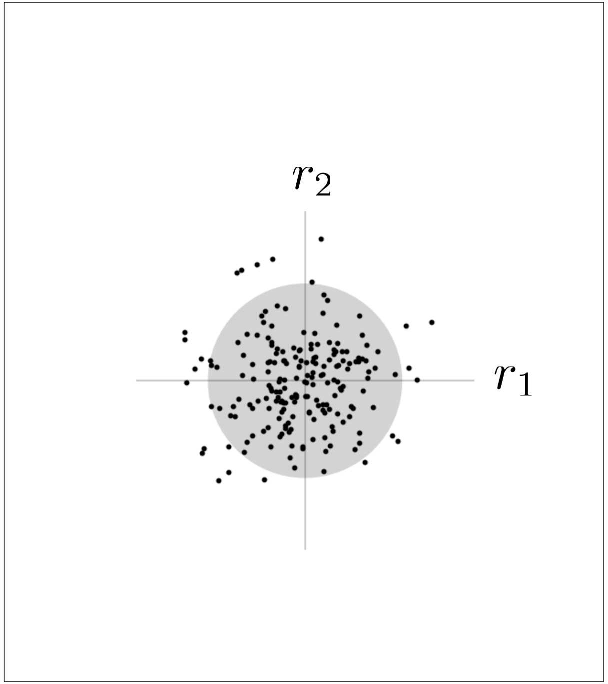
Stimulus distribution



Models



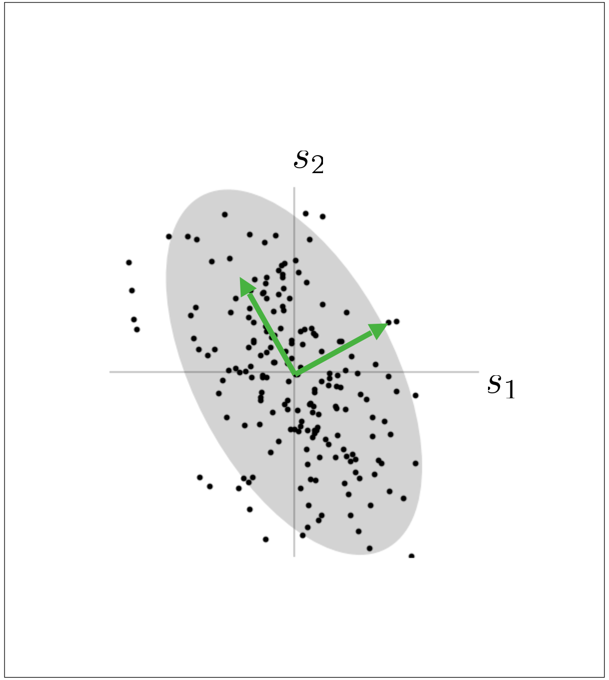
Response distribution



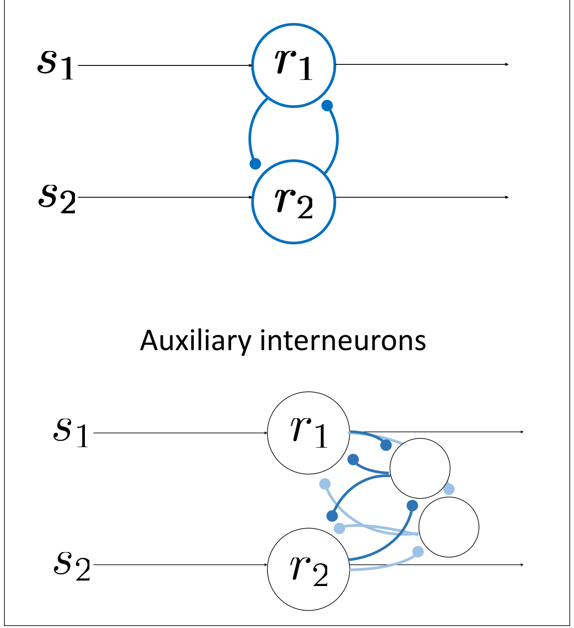
Wick et al. 2010; Pehlevan et al. 2015; Chapochnikov et al. 2023; …

Existing adaptive neural network models

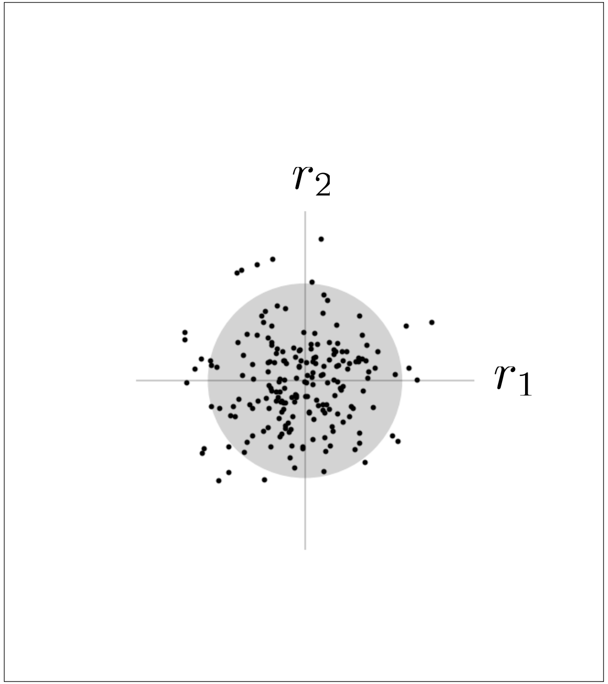
Stimulus distribution



Models



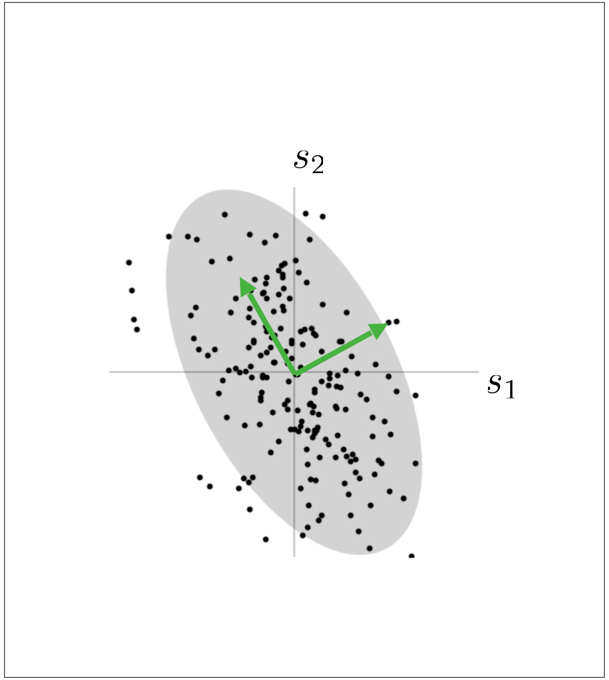
Response distribution



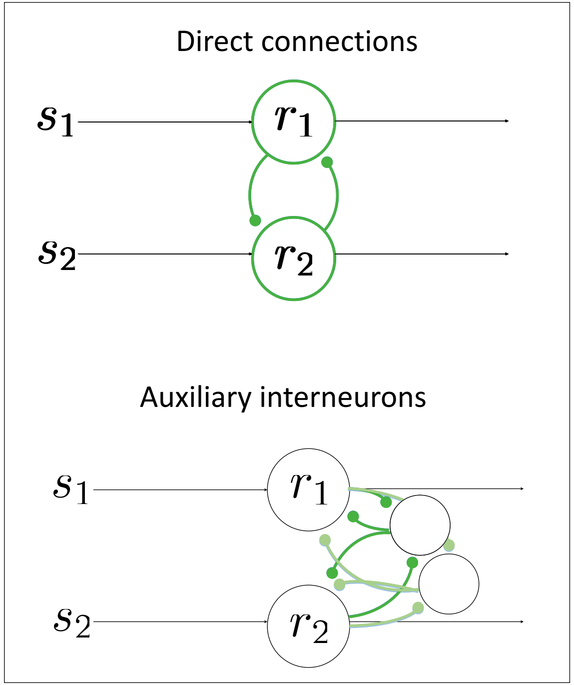
Wick et al. 2010; Pehlevan et al. 2015; Chapochnikov et al. 2023; …

Existing adaptive neural network models

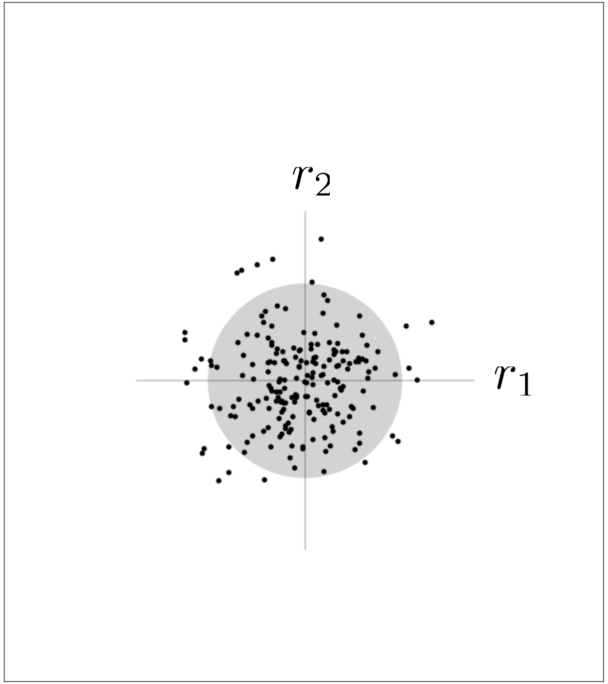
Stimulus distribution



Models



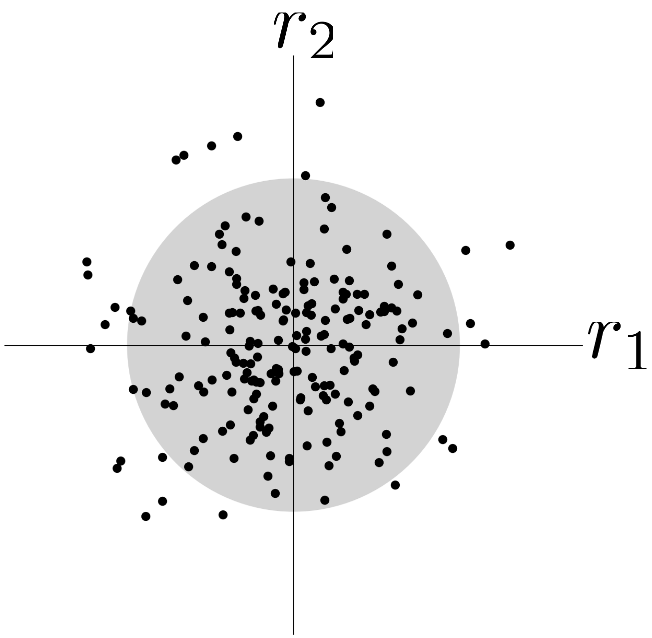
Response distribution



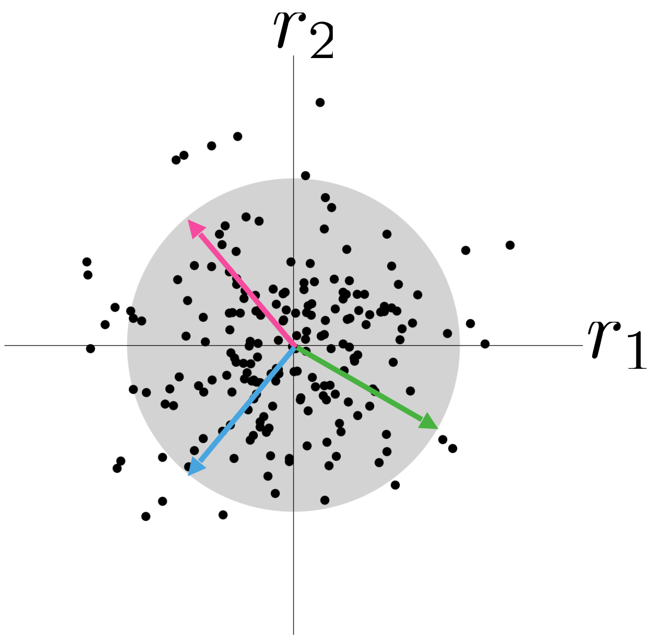
Wick et al. 2010; Pehlevan et al. 2015; Chapochnikov et al. 2023; …

Q: Can neural circuits **decorrelate** their responses using **gain modulation**?

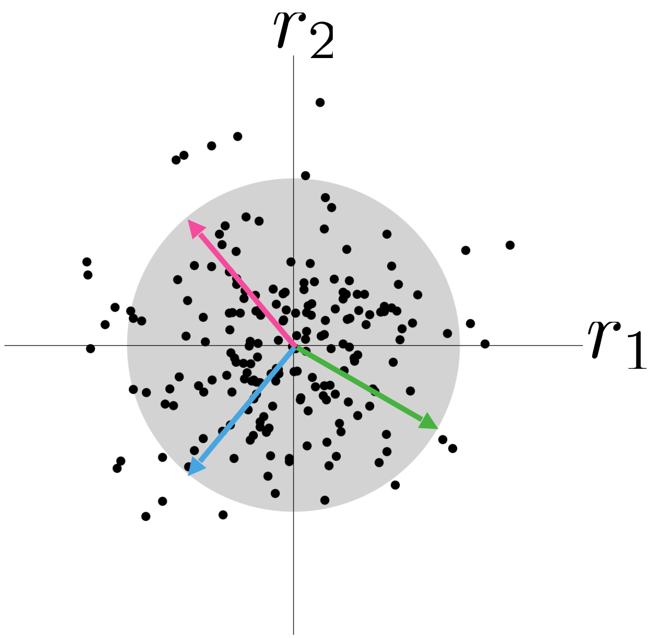
Geometric intuition

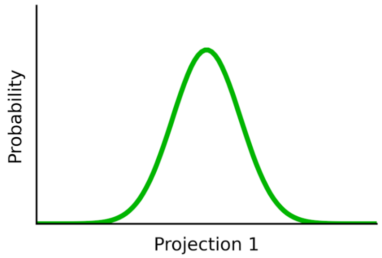


Geometric intuition

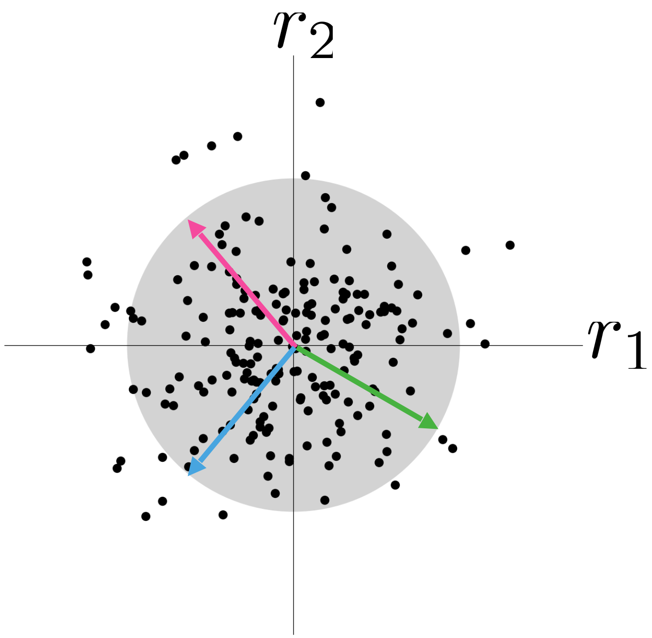


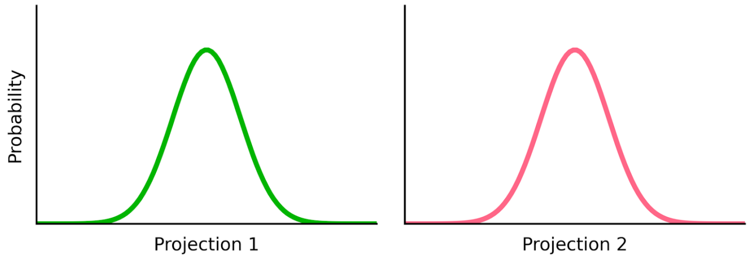
Geometric intuition



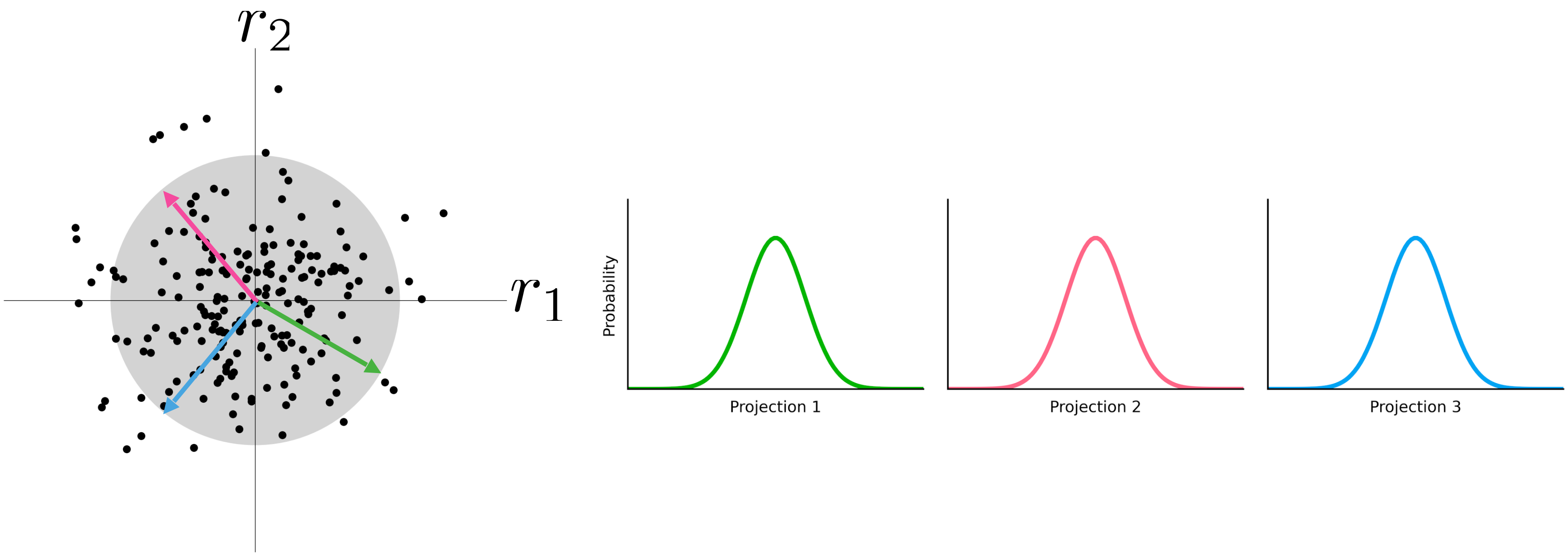


Geometric intuition

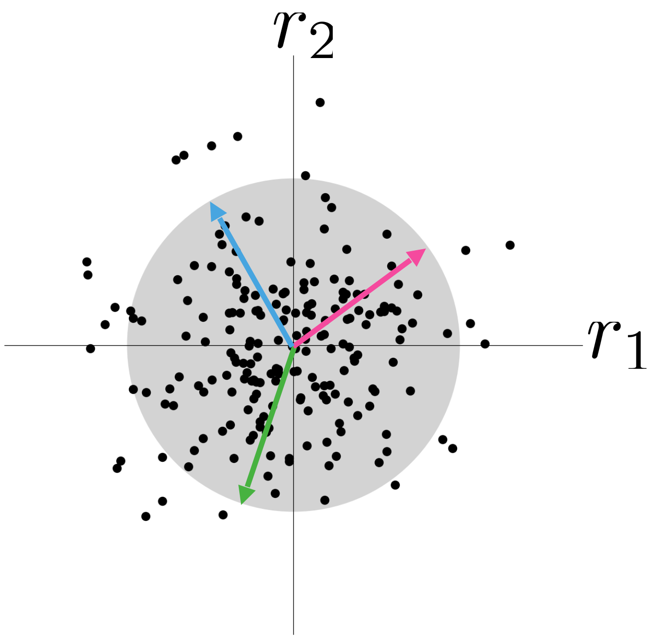


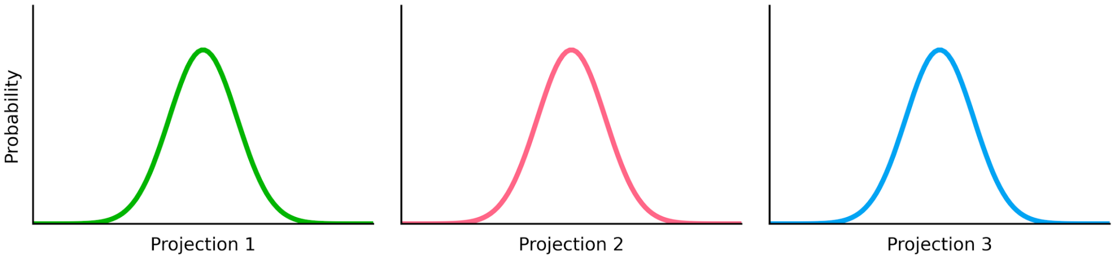


Geometric intuition

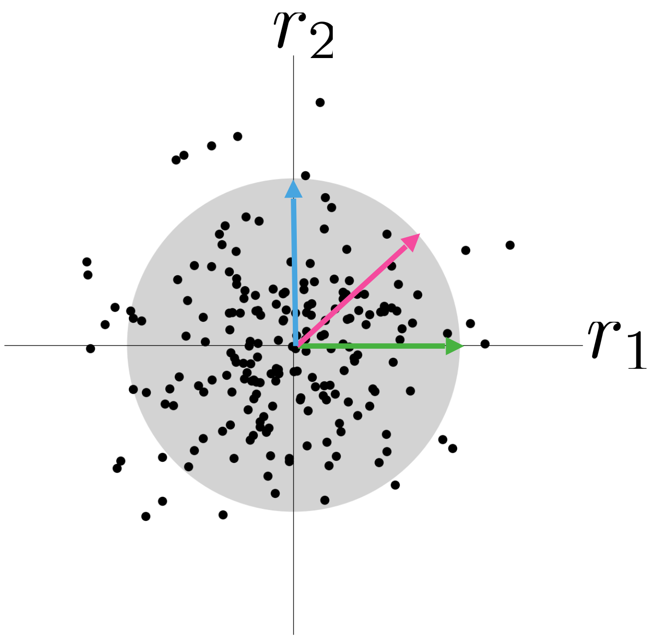


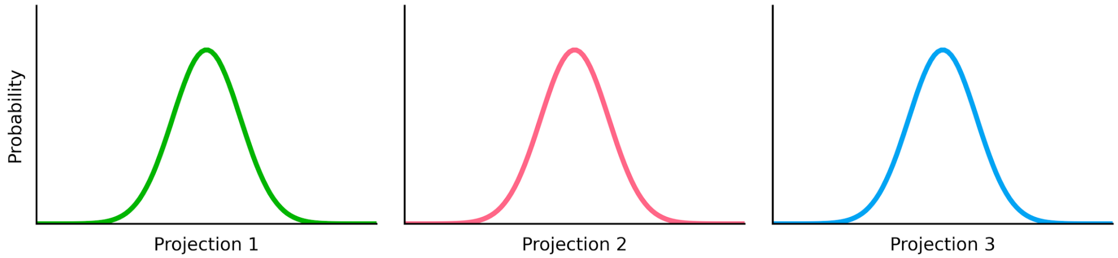
Geometric intuition



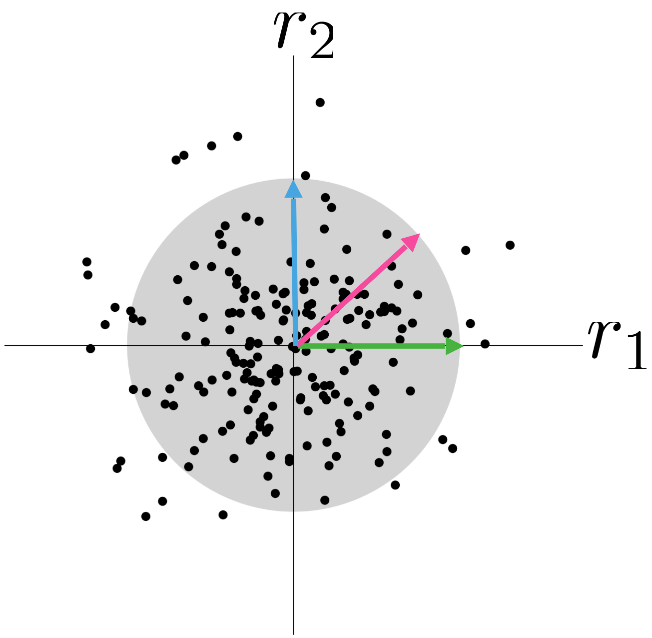


Geometric intuition

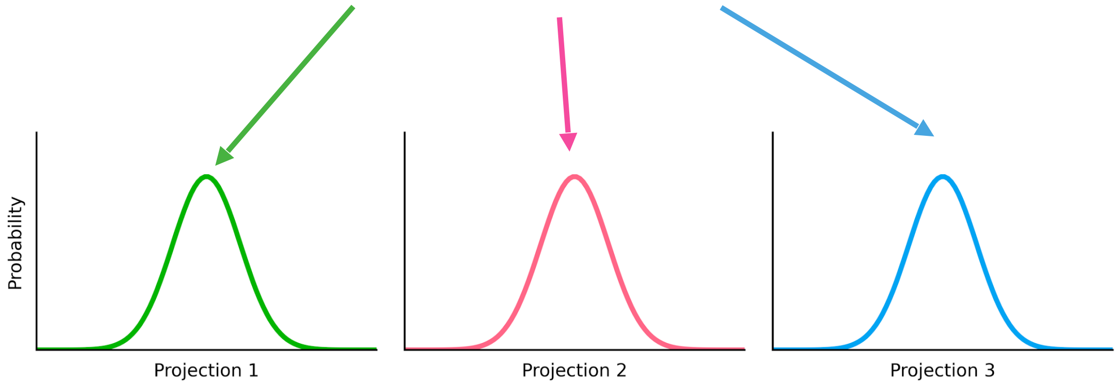




Geometric intuition



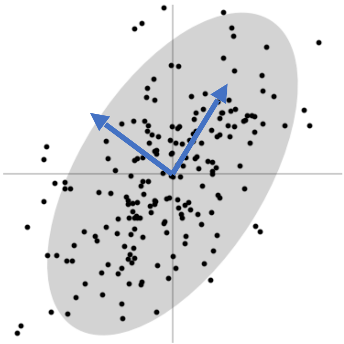
Necessary & sufficient:
3 projected distribution variances = 1!



Novel matrix factorization

Novel matrix factorization

Traditional approaches (PCA)

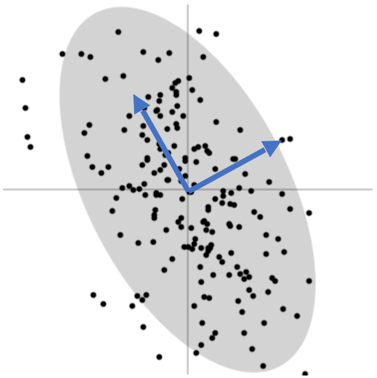


$$\mathbf{C}^{1/2} = \mathbf{V} \, \boldsymbol{\Lambda}^{1/2} \, \mathbf{V}^\top$$

Principal axes must be **relearned** for different input densities.

Novel matrix factorization

Traditional approaches (PCA)

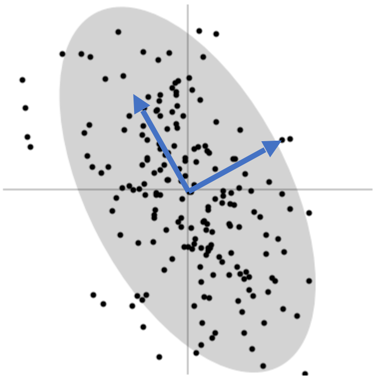


$$\mathbf{C}^{1/2} = \mathbf{V} \, \mathbf{\Lambda}^{1/2} \, \mathbf{V}^\top$$

Principal axes must be **relearned** for different input densities.

Novel matrix factorization

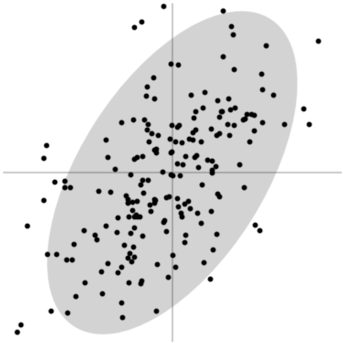
Traditional approaches (PCA)



$$\mathbf{C}^{1/2} = \mathbf{V} \quad \mathbf{\Lambda}^{1/2} \quad \mathbf{V}^\top$$

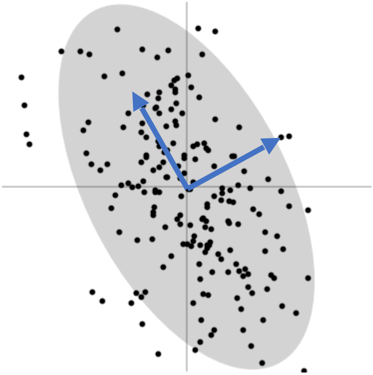
Principal axes must be **relearned** for different input densities.

Our approach



Novel matrix factorization

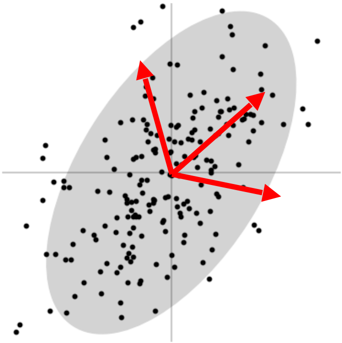
Traditional approaches (PCA)



$$\mathbf{C}^{1/2} = \mathbf{V}\,\boldsymbol{\Lambda}^{1/2}\,\mathbf{V}^\top$$

Principal axes must be **relearned** for different input densities.

Our approach

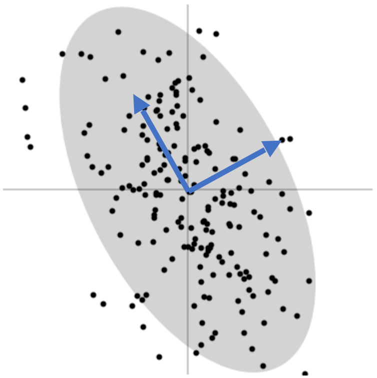


$$\mathbf{C}^{1/2} = \mathbf{W}\,\mathrm{diag}(\mathbf{g})\,\mathbf{W}^\top$$

$K = \frac{N(N+1)}{2}$ **arbitrary** axes remain **fixed for all** densities!

Novel matrix factorization

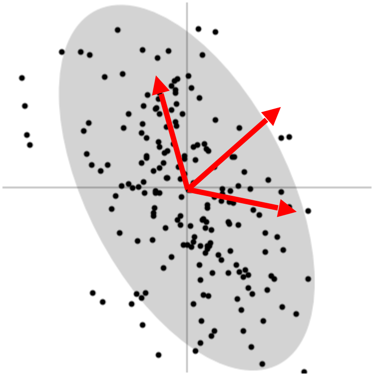
Traditional approaches (PCA)

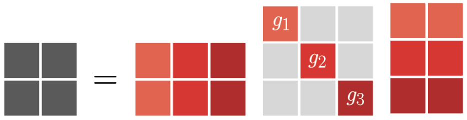


$$\mathbf{C}^{1/2} = \mathbf{V} \quad \mathbf{\Lambda}^{1/2} \quad \mathbf{V}^\top$$

Principal axes must be **relearned** for different input densities.

Our approach



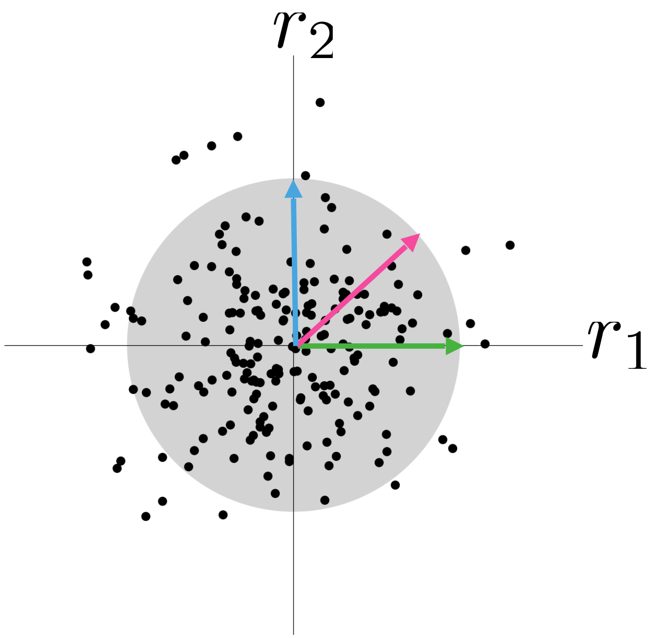


$$\mathbf{C}^{1/2} = \mathbf{W} \quad \mathrm{diag}(\mathbf{g}) \quad \mathbf{W}^\top$$

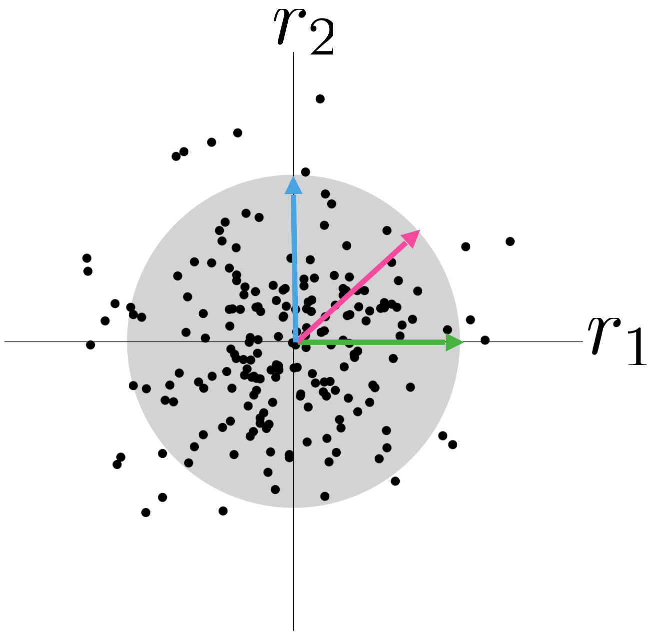
$$K = \frac{N(N+1)}{2}$$

arbitrary axes remain **fixed for all** densities!

Must be overcomplete



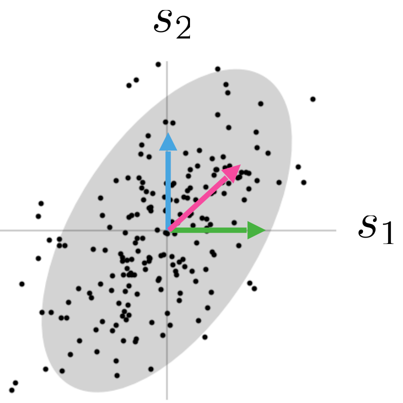
Must be overcomplete



# Primary neurons	# 1D Projections
2	3
3	6
10	55
100	5K

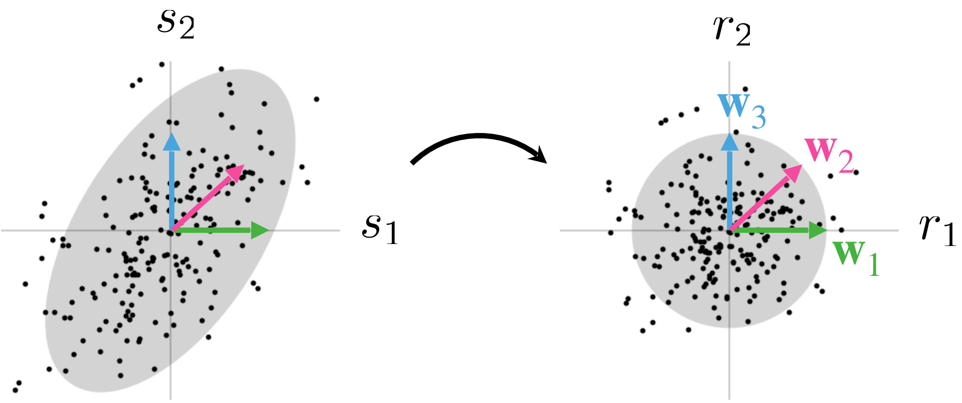
Adaptive whitening via **gain modulation**

Adaptation objective



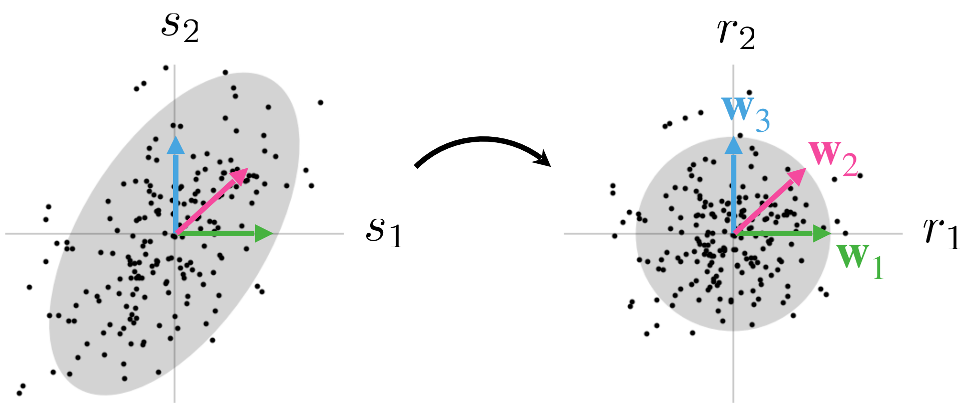
Adaptive whitening via **gain modulation**

Adaptation objective



Adaptive whitening via **gain modulation**

Adaptation objective

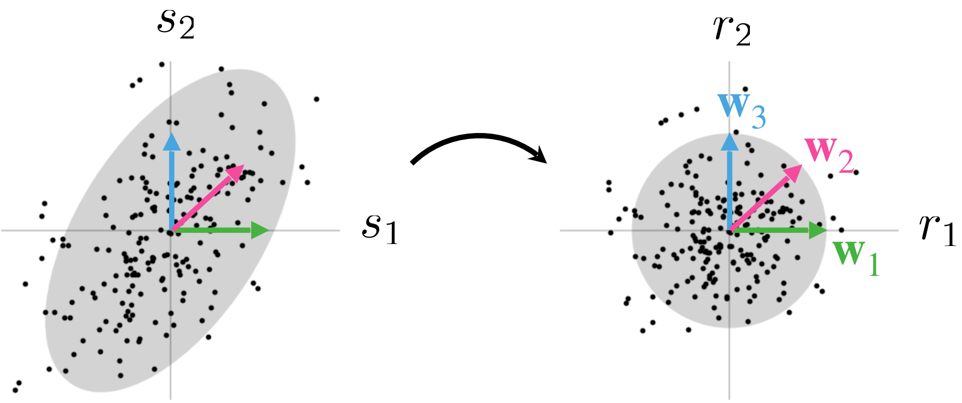


$$\max_{\mathbf{g}} \min_{\mathbf{r}_t} \langle \ell(\mathbf{g}, \mathbf{s}_t, \mathbf{r}_t) \rangle_t$$

$$\ell(\mathbf{g}, \mathbf{s}, \mathbf{r}) = \|\mathbf{r} - \mathbf{s}\|^2 + \sum_{i=1}^{K} g_i \left\{ (\mathbf{w}_i^\top \mathbf{r})^2 - 1 \right\}$$

Adaptive whitening via **gain modulation**

Adaptation objective



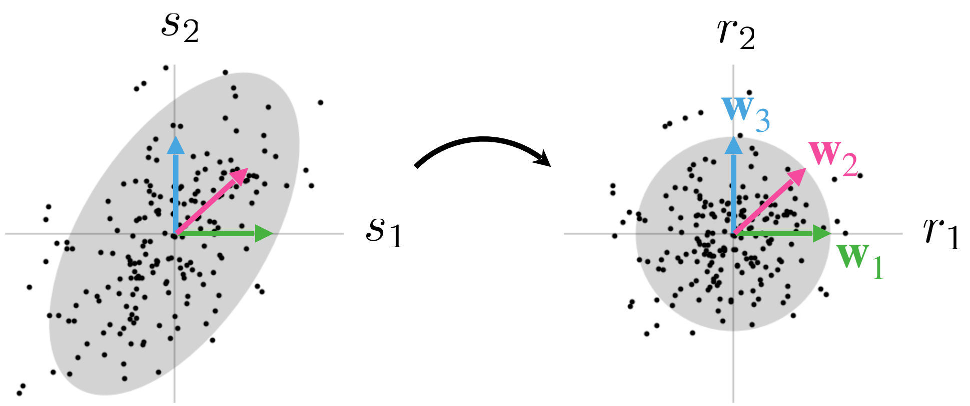
$$\max_{\mathbf{g}} \min_{\mathbf{r}_t} \langle \ell(\mathbf{g}, \mathbf{s}_t, \mathbf{r}_t) \rangle_t$$

$$\ell(\mathbf{g}, \mathbf{s}, \mathbf{r}) = \boxed{\|\mathbf{r} - \mathbf{s}\|^2} + \sum_{i=1}^{K} g_i \left\{ (\mathbf{w}_i^\top \mathbf{r})^2 - 1 \right\}$$

match response
to stimuli

Adaptive whitening via **gain modulation**

Adaptation objective



$$\max_{\mathbf{g}} \min_{\mathbf{r}_t} \langle \ell(\mathbf{g}, \mathbf{s}_t, \mathbf{r}_t) \rangle_t$$

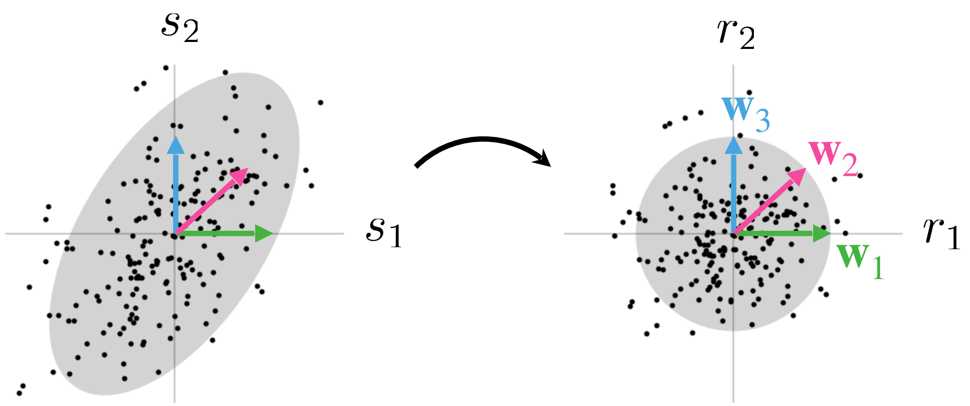
$$\ell(\mathbf{g}, \mathbf{s}, \mathbf{r}) = \|\mathbf{r} - \mathbf{s}\|^2 + \sum_{i=1}^{K} g_i \left\{ (\mathbf{w}_i^\top \mathbf{r})^2 - 1 \right\}$$

match response to stimuli

enforce variance constraints

Adaptive whitening via **gain modulation**

Adaptation objective



$$\max_{\mathbf{g}} \min_{\mathbf{r}_t} \langle \ell(\mathbf{g}, \mathbf{s}_t, \mathbf{r}_t) \rangle_t$$

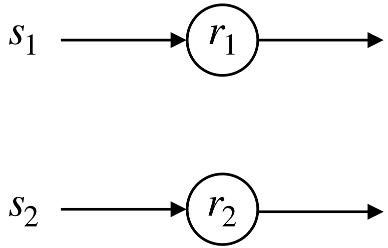
$$\ell(\mathbf{g}, \mathbf{s}, \mathbf{r}) = \underbrace{\|\mathbf{r} - \mathbf{s}\|^2}_{\text{match response to stimuli}} + \sum_{i=1}^{K} \underbrace{g_i \left\{ (\mathbf{w}_i^\top \mathbf{r})^2 - 1 \right\}}_{\text{enforce variance constraints}}$$

Adaptation algorithm

Algorithm 1: Adaptive whitening via gain modulation

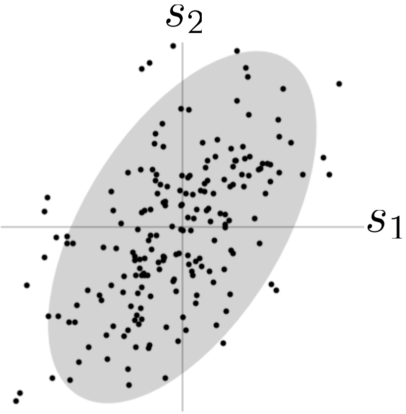
Input: Centered inputs $\mathbf{s}_1, \mathbf{s}_2, \cdots \in \mathbb{R}^N$
Initialize: $\mathbf{W} \in \mathbb{R}^{N \times K}$; $\mathbf{g} \in \mathbb{R}^K$; $\eta, \gamma > 0$
for $t = 1, 2, \ldots$ **do**
  $\mathbf{r}_t \leftarrow \mathbf{0}$
  while not converged **do**
    $\mathbf{z}_t \leftarrow \mathbf{W}^\top \mathbf{r}_t$
    $\mathbf{r}_t \leftarrow \mathbf{r}_t + \gamma \{ \mathbf{s}_t - \mathbf{W}(\mathbf{g} \circ \mathbf{z}_t) - \mathbf{r}_t \}$
  end while
  $\mathbf{g} \leftarrow \mathbf{g} + \eta \left(\mathbf{z}_t^{\circ 2} - \mathbf{1} \right)$
end for

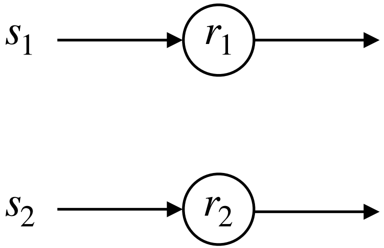
Adaptive neural circuit w/ **gain-modulating interneurons**



Adaptive neural circuit w/ **gain-modulating interneurons**

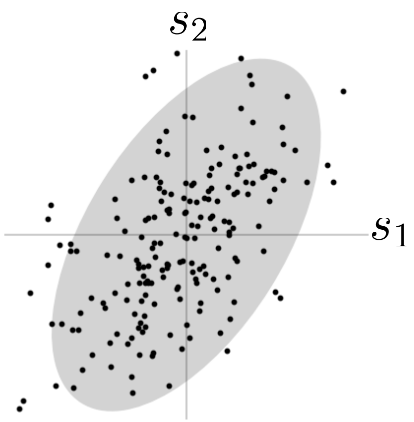
Stimulus distribution

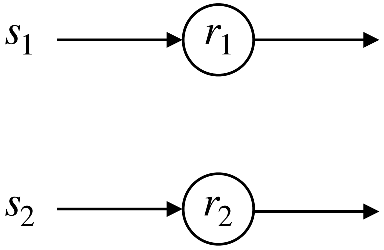




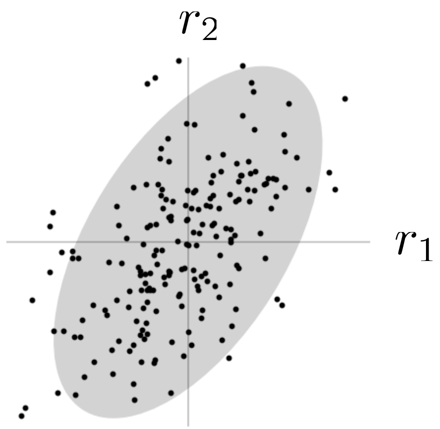
Adaptive neural circuit w/ **gain-modulating interneurons**

Stimulus distribution



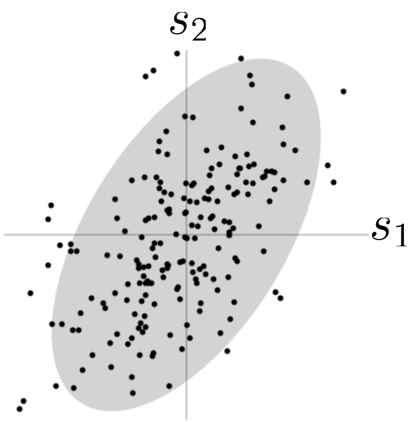


Response distribution

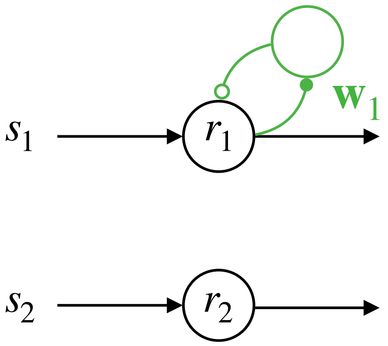


Adaptive neural circuit w/ **gain-modulating interneurons**

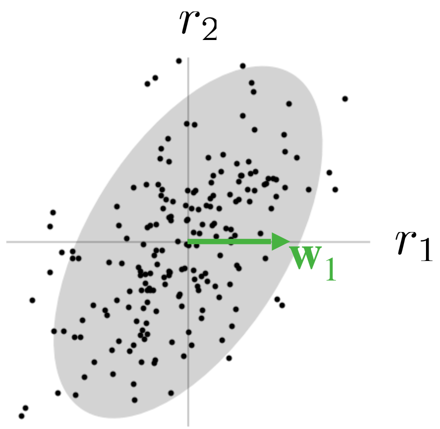
Stimulus distribution



Gain-modulating interneurons

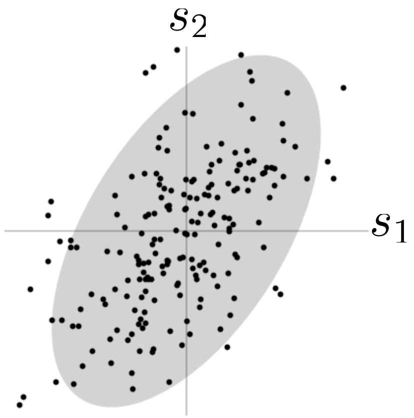


Response distribution

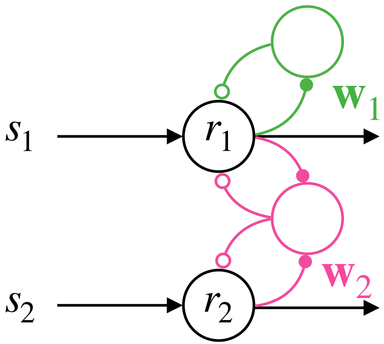


Adaptive neural circuit w/ **gain-modulating interneurons**

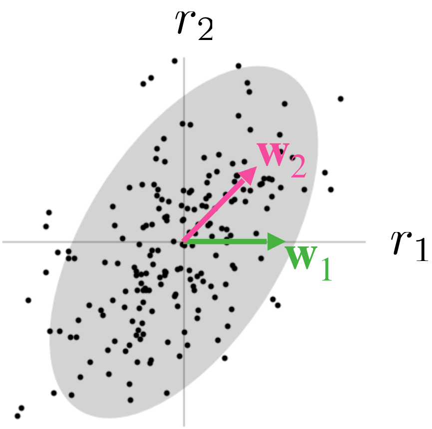
Stimulus distribution



Gain-modulating interneurons

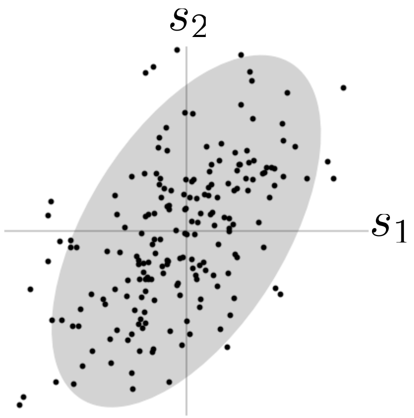


Response distribution

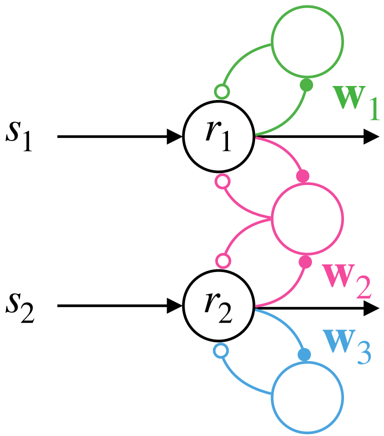


Adaptive neural circuit w/ **gain-modulating interneurons**

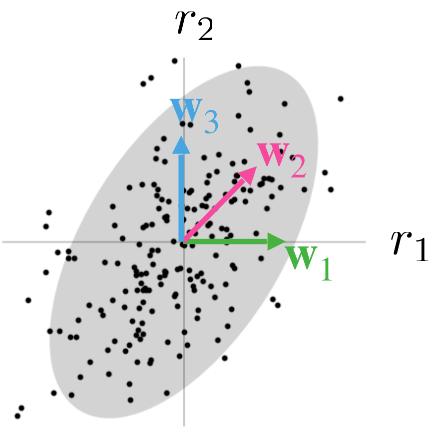
Stimulus distribution



Gain-modulating interneurons

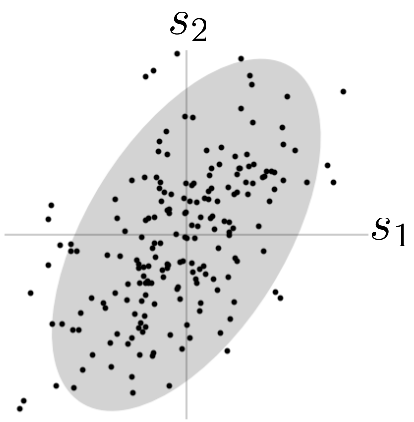


Response distribution

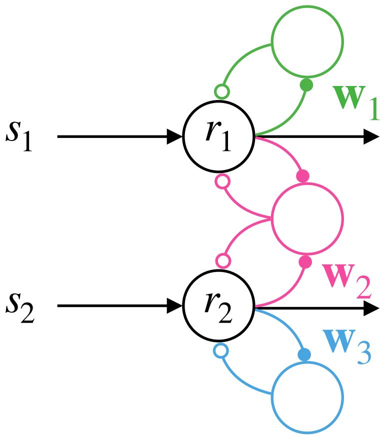


Adaptive neural circuit w/ **gain-modulating interneurons**

Stimulus distribution

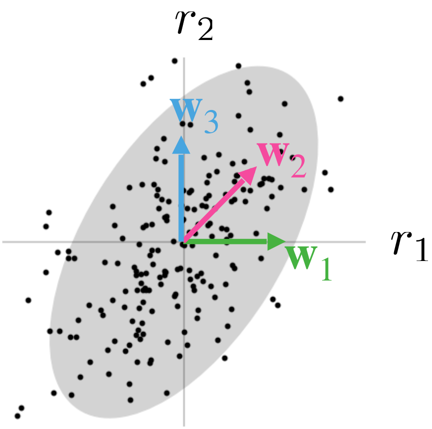


Gain-modulating interneurons



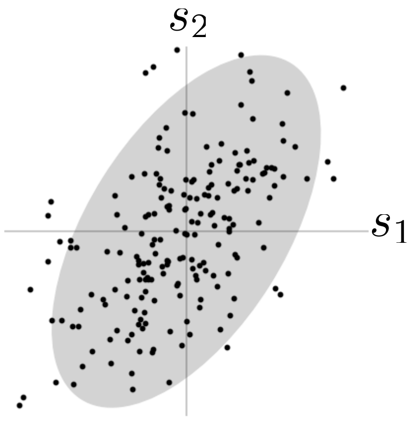
input: $z_i = \mathbf{w}_i^\top \mathbf{r}$

Response distribution

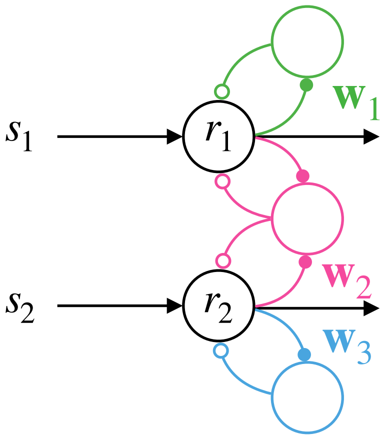


Adaptive neural circuit w/ **gain-modulating interneurons**

Stimulus distribution



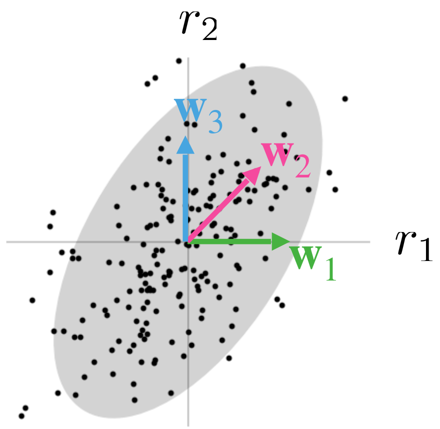
Gain-modulating interneurons



input: $z_i = \mathbf{w}_i^\top \mathbf{r}$

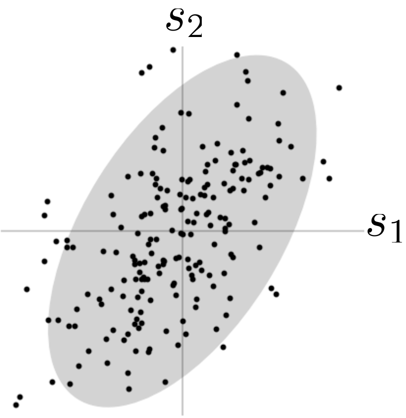
output: $n_i = g_i z_i$

Response distribution

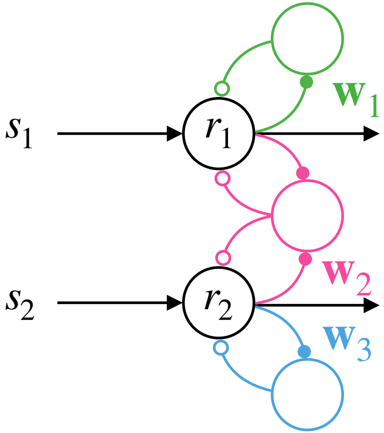


Adaptive neural circuit w/ **gain-modulating interneurons**

Stimulus distribution



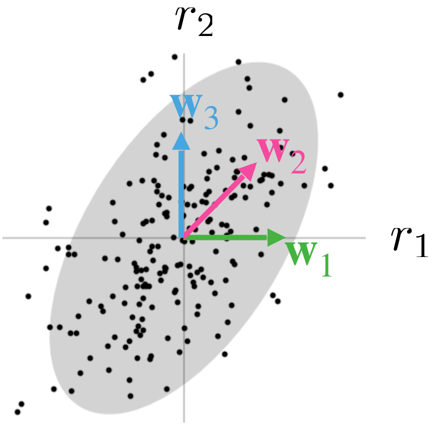
Gain-modulating interneurons



input: $z_i = \mathbf{w}_i^\top \mathbf{r}$

output: $n_i = g_i z_i$

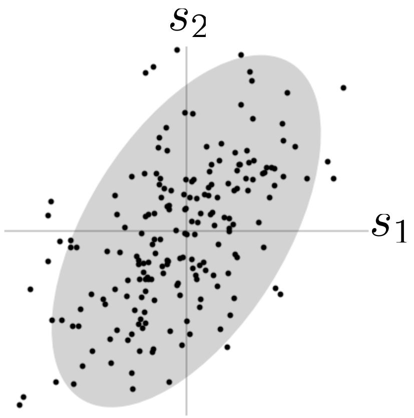
Response distribution



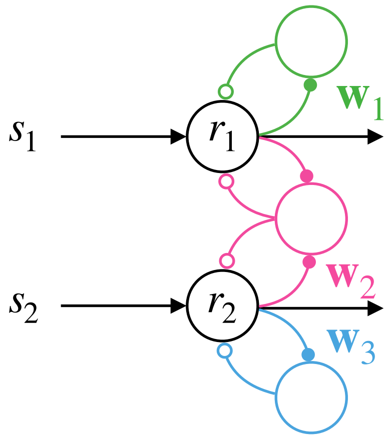
fast neural dynamics: $\frac{d\mathbf{r}}{dt} = \mathbf{s} - \mathbf{r} - \mathbf{W}\mathbf{n}$

Adaptive neural circuit w/ **gain-modulating interneurons**

Stimulus distribution



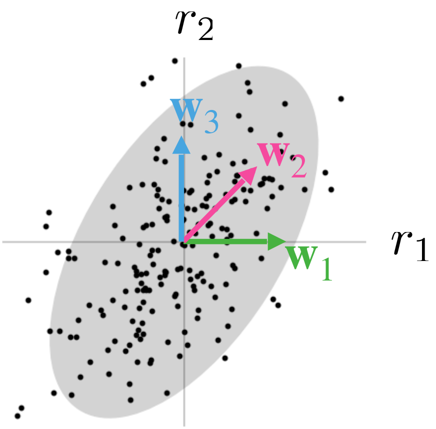
Gain-modulating interneurons



input: $z_i = \mathbf{w}_i^\top \mathbf{r}$

output: $n_i = g_i z_i$

Response distribution

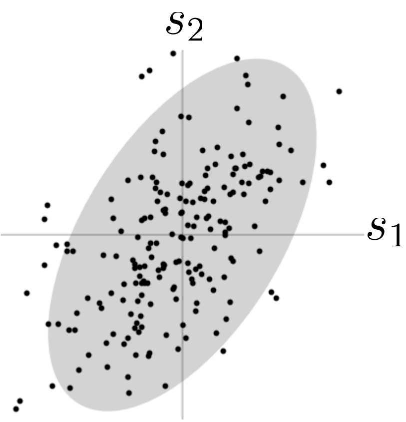


fast neural dynamics: $\dfrac{d\mathbf{r}}{dt} = \mathbf{s} - \mathbf{r} - \mathbf{W}\mathbf{n}$

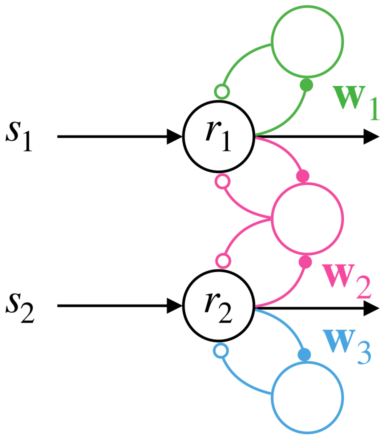
slow gain updates: $\Delta g_i = \eta_g(\mathrm{var}(z_i) - 1)$

Adaptive neural circuit w/ **gain-modulating interneurons**

Stimulus distribution



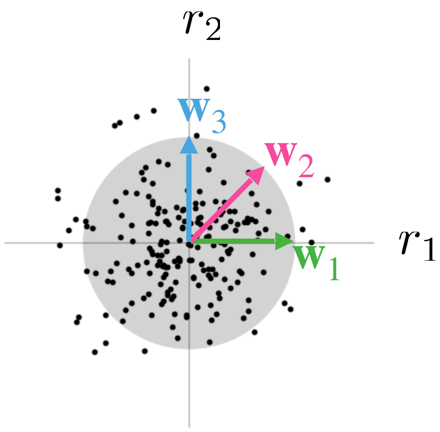
Gain-modulating interneurons



input: $z_i = \mathbf{w}_i^\top \mathbf{r}$

output: $n_i = g_i z_i$

Response distribution

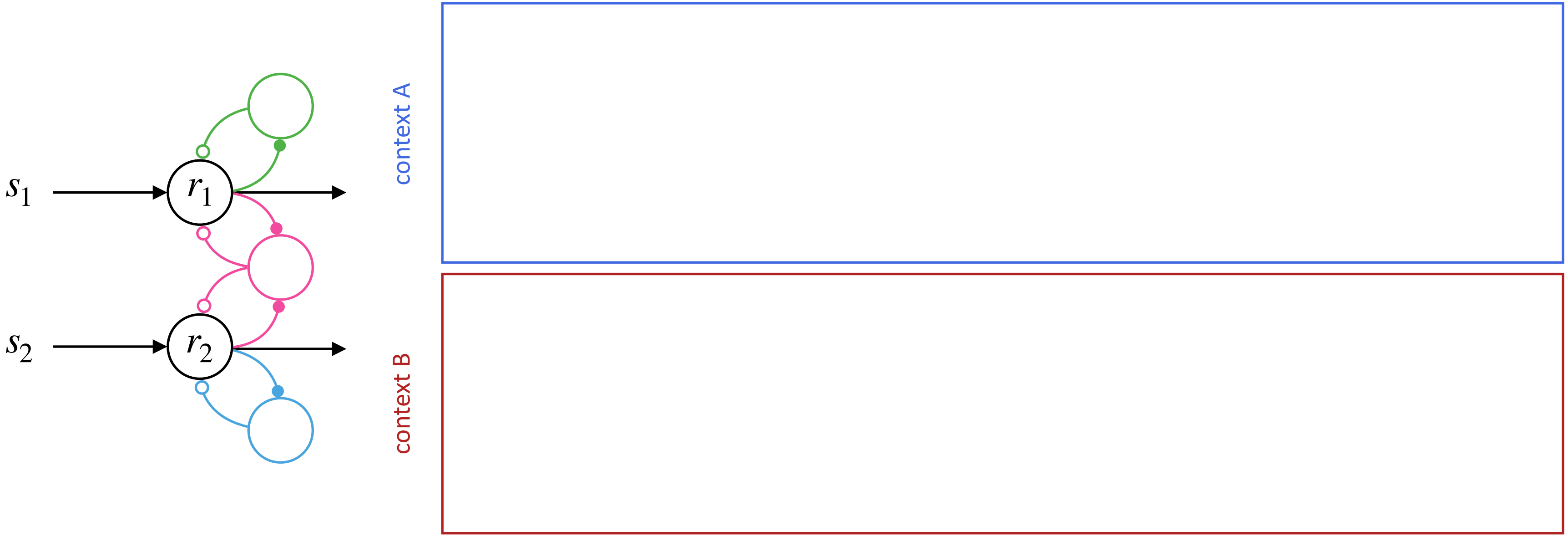


Correlations removed!

fast neural dynamics: $\dfrac{d\mathbf{r}}{dt} = \mathbf{s} - \mathbf{r} - \mathbf{W}\mathbf{n}$

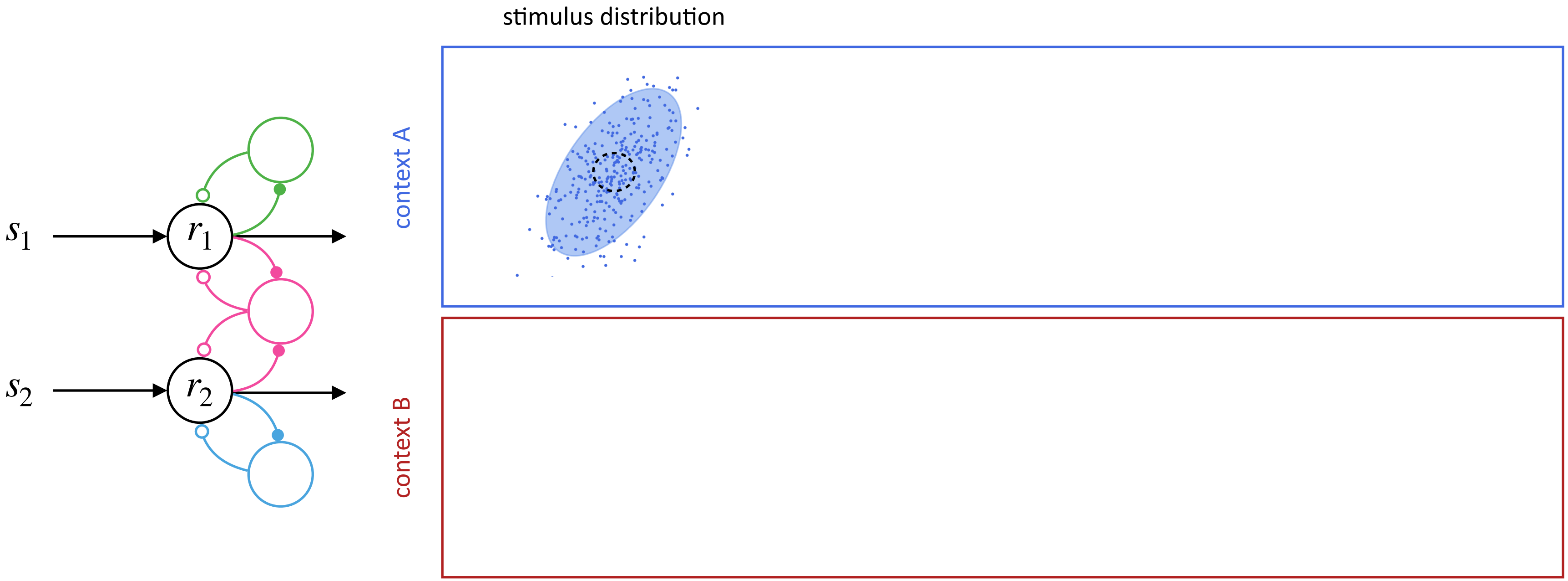
slow gain updates: $\Delta g_i = \eta_g(\mathrm{var}(z_i) - 1)$

Numerical simulations



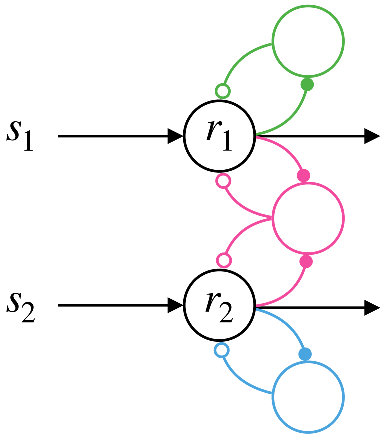
Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

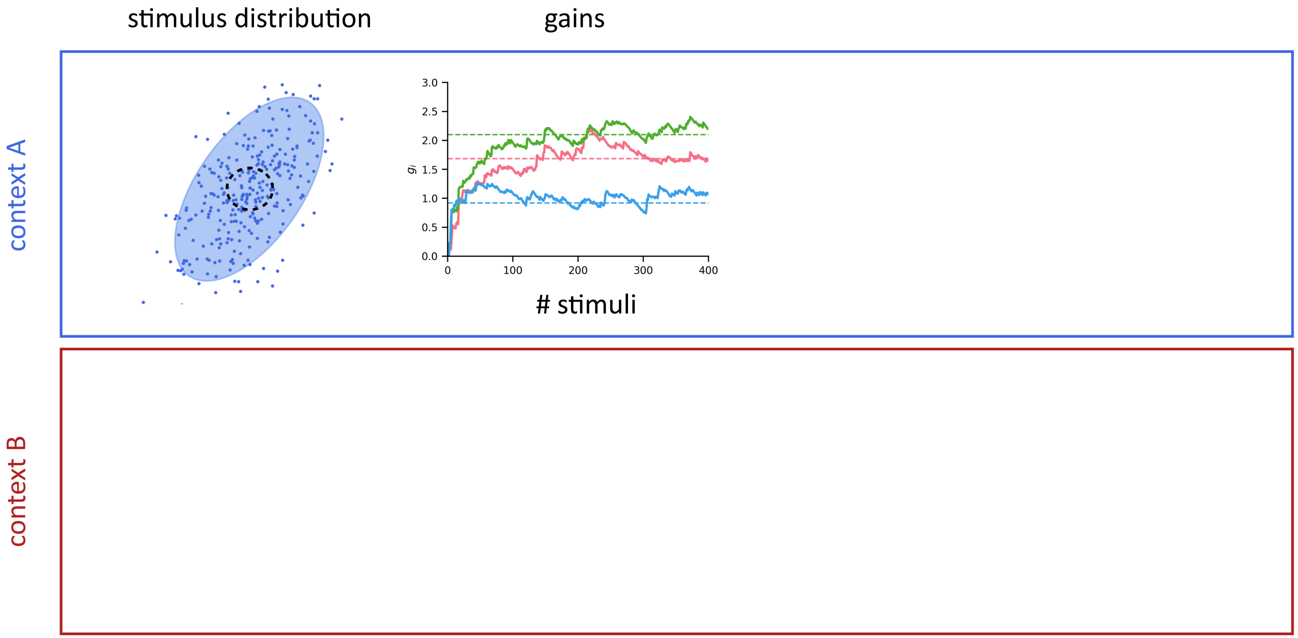
Numerical simulations



Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

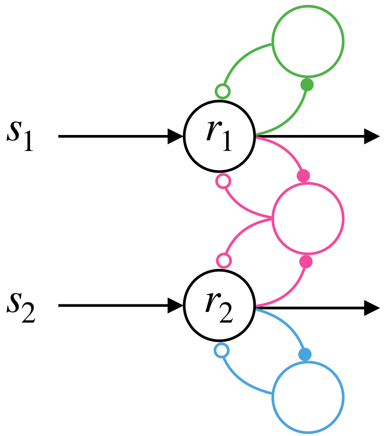
Numerical simulations

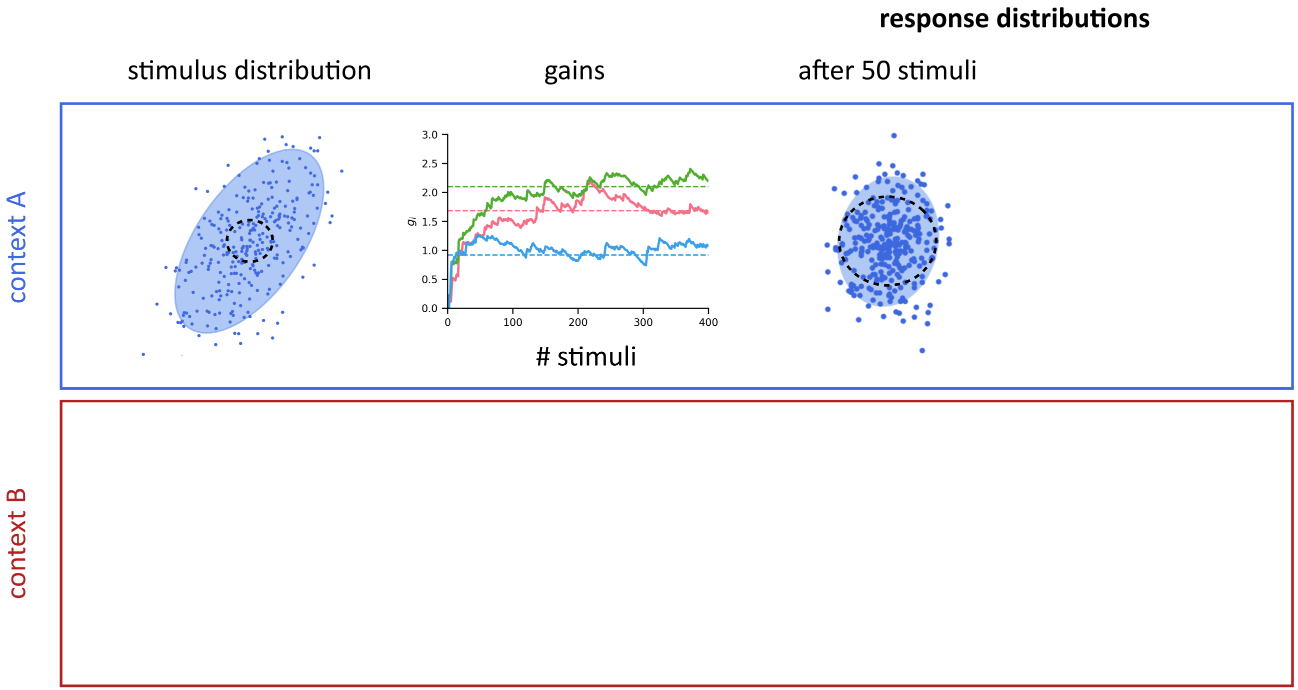




Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

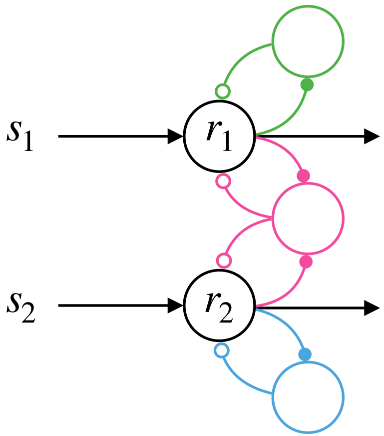
Numerical simulations

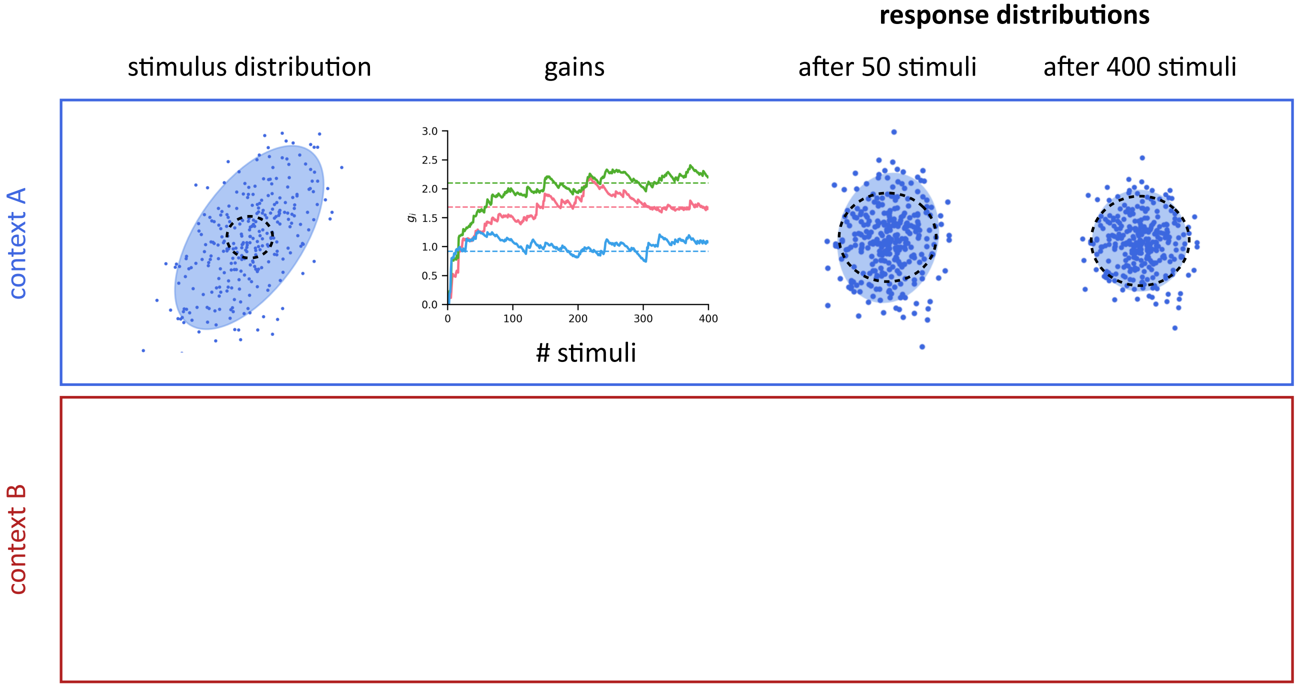




Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

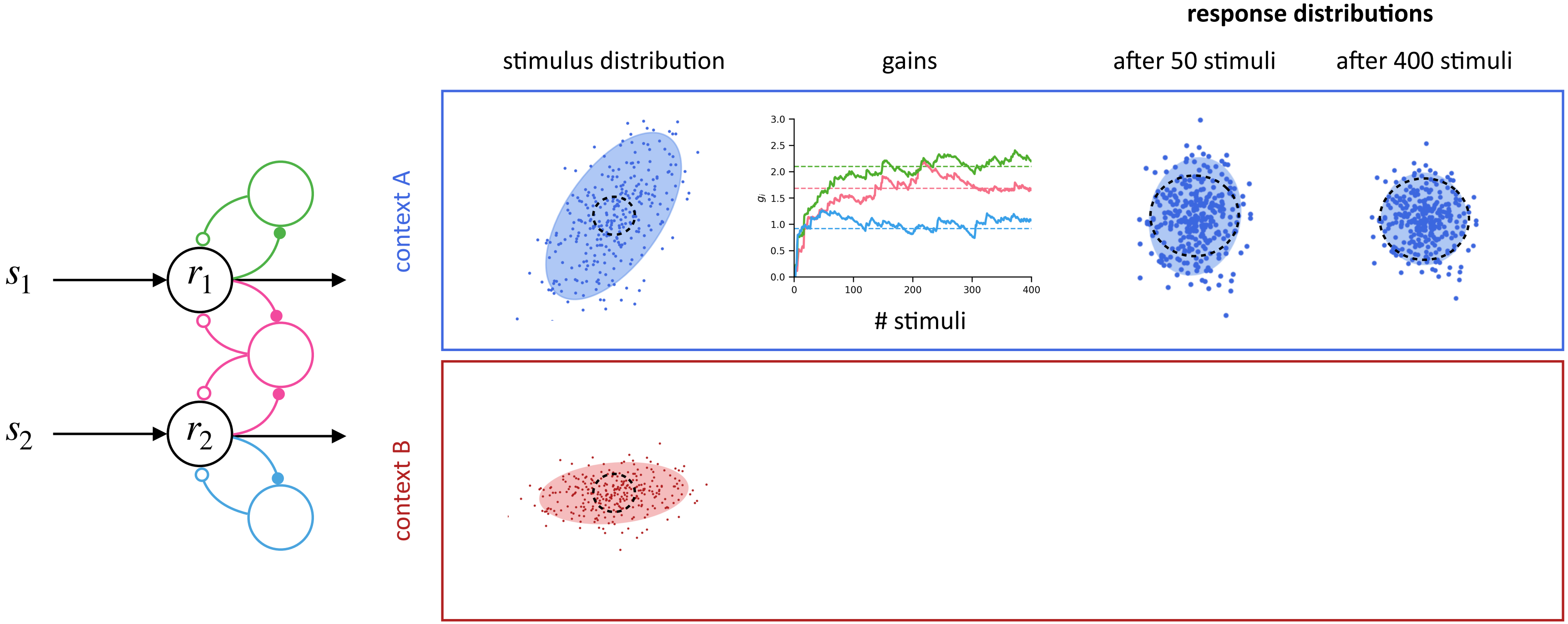
Numerical simulations





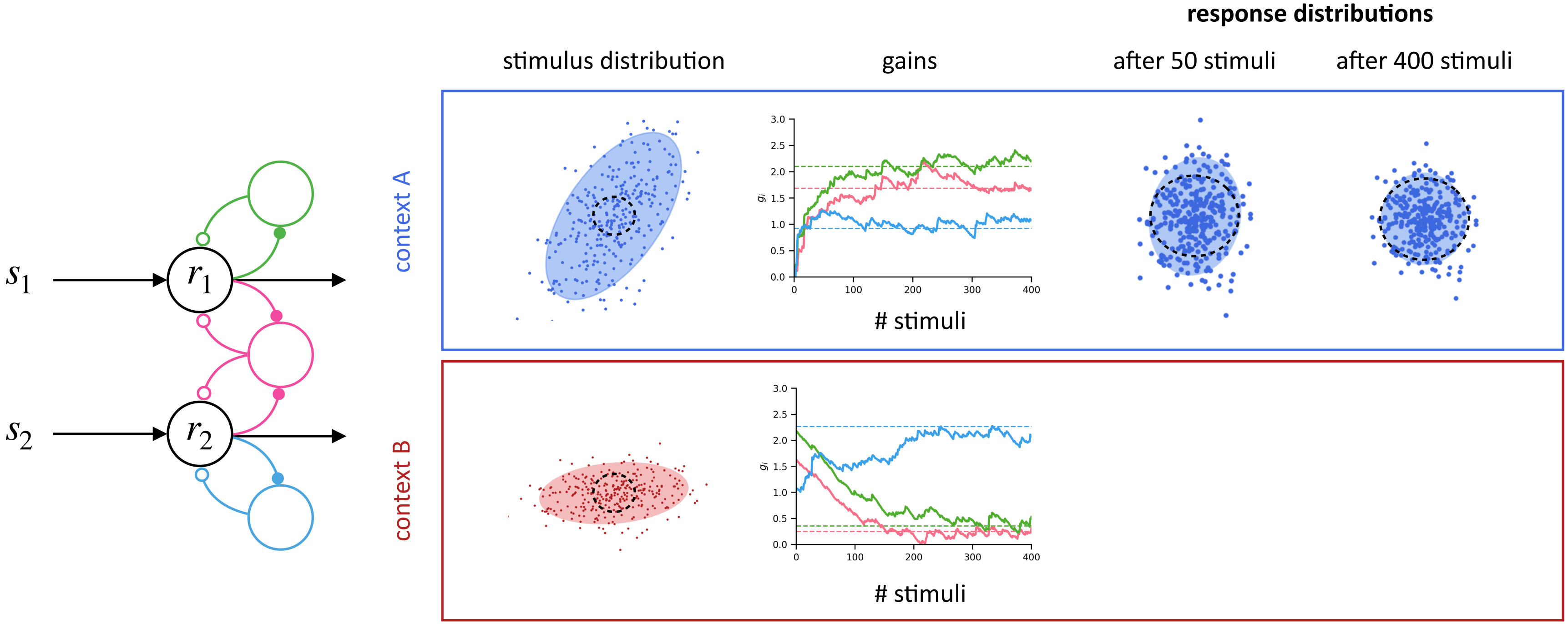
Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

Numerical simulations



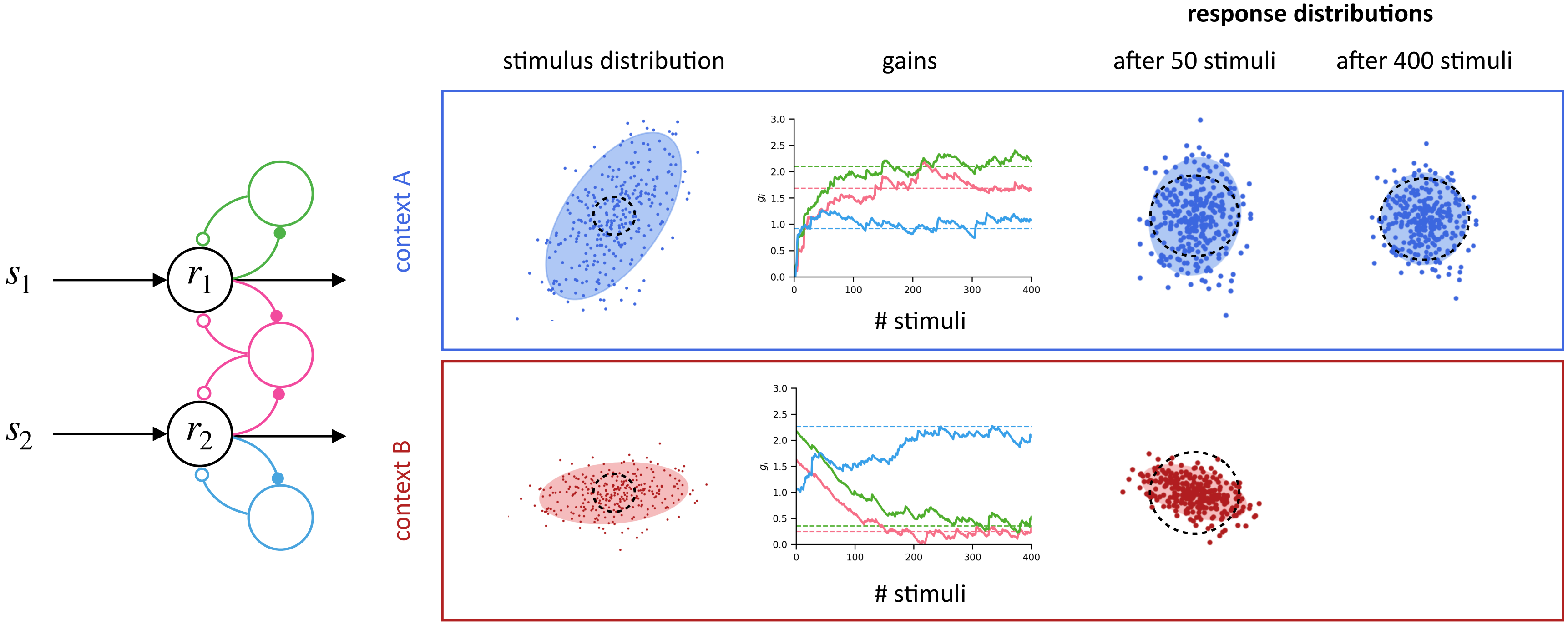
Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

Numerical simulations



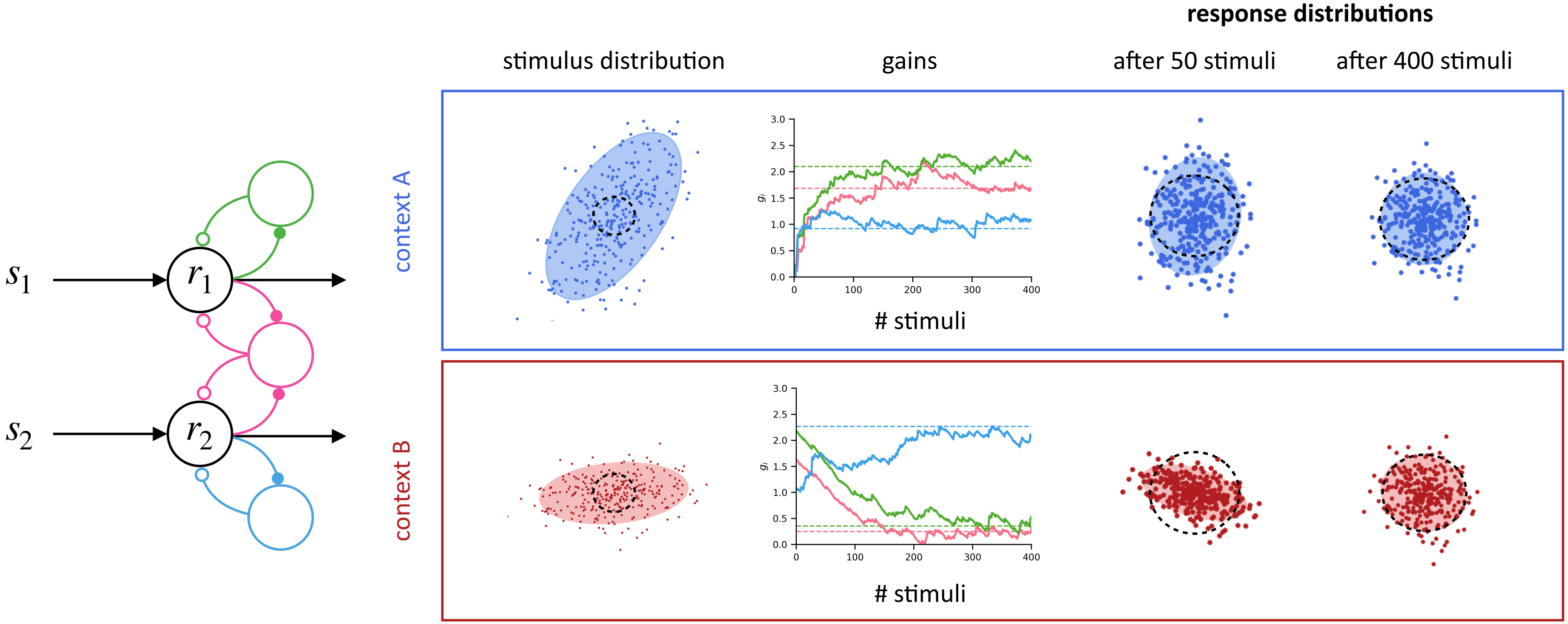
Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

Numerical simulations



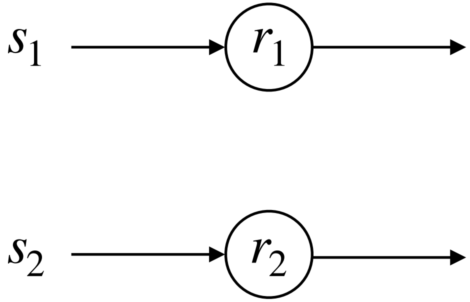
Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

Numerical simulations



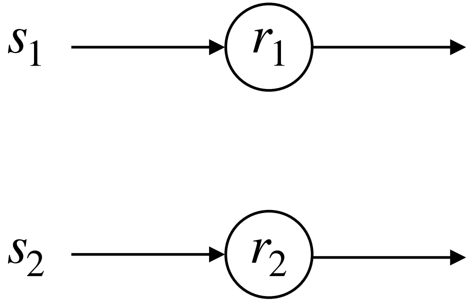
Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, *ICML*, 2023

Summary



Summary

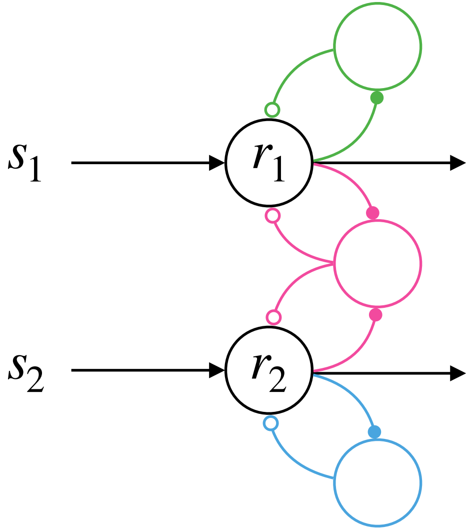
Q: Can neural circuits **decorrelate** their responses using **gain modulation**?



Summary

Q: Can neural circuits **decorrelate** their responses using **gain modulation**?

A: Yes. Using **gain-modulating interneurons** and a novel mathematical perspective.

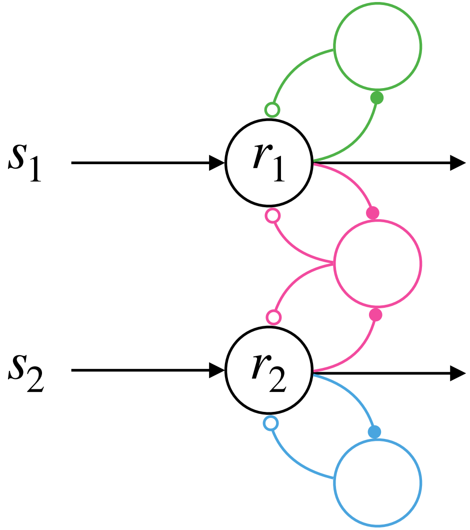


Summary

Q: Can neural circuits **decorrelate** their responses using **gain modulation**?

A: Yes. Using **gain-modulating interneurons** and a novel mathematical perspective.

Prediction: Local interneurons modulate their gains in response to changes in their **input variance**

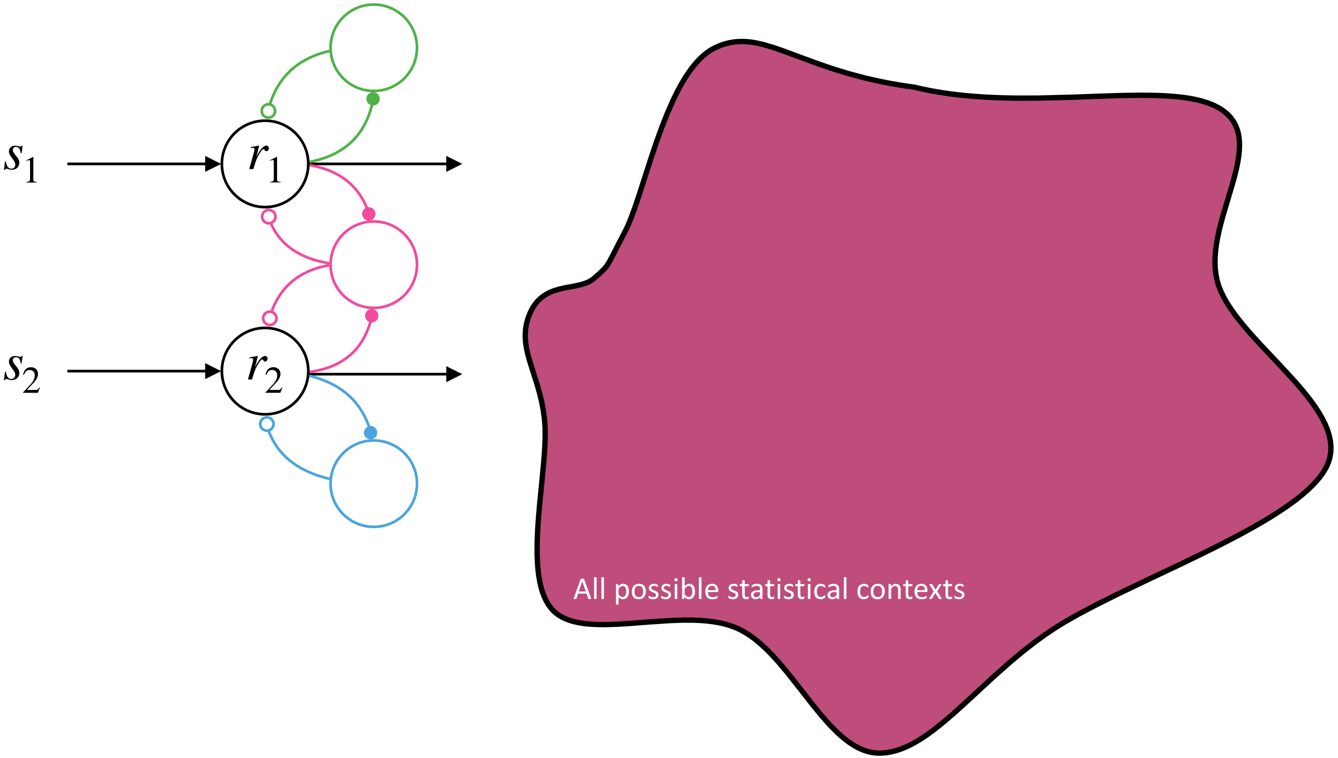


But wait…

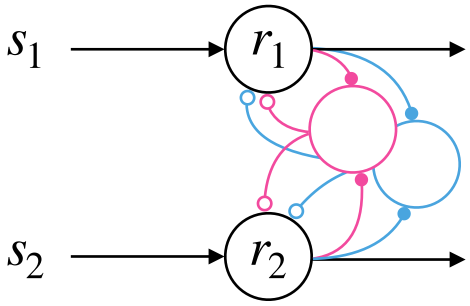
But wait…

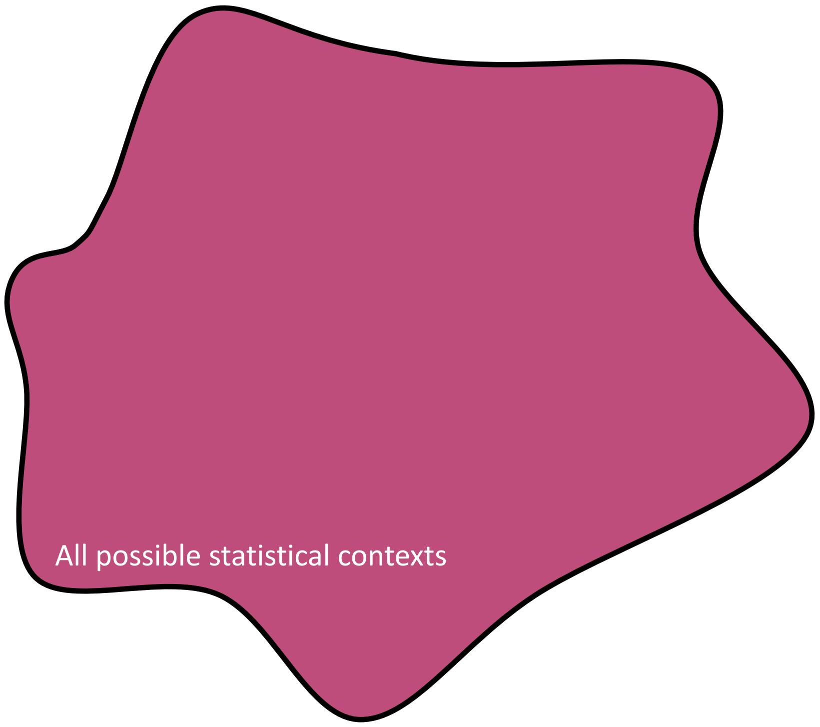
# primary neurons	# interneurons
2	3
3	6
10	55
100	5K

Multi-timescale model intuition

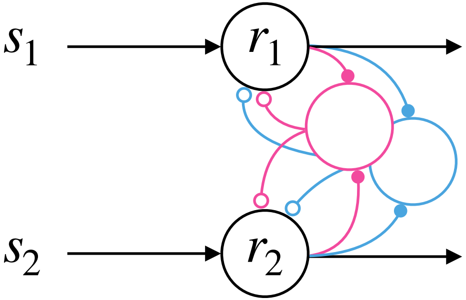


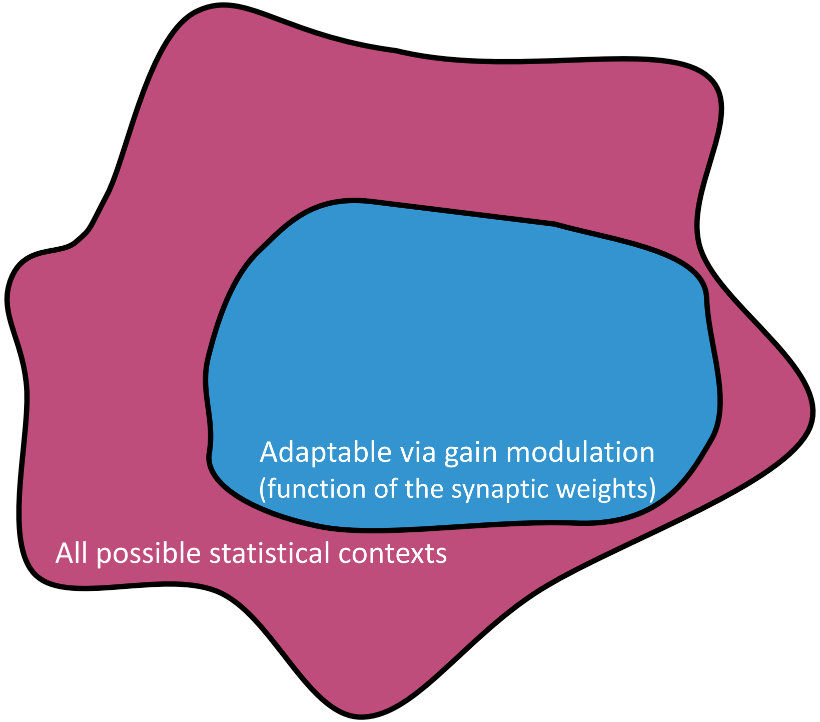
Multi-timescale model intuition



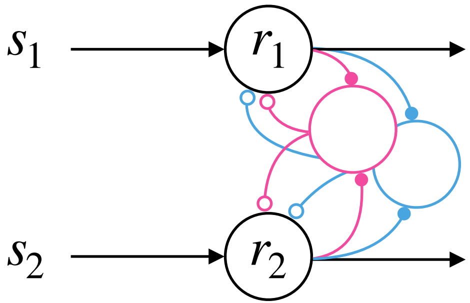


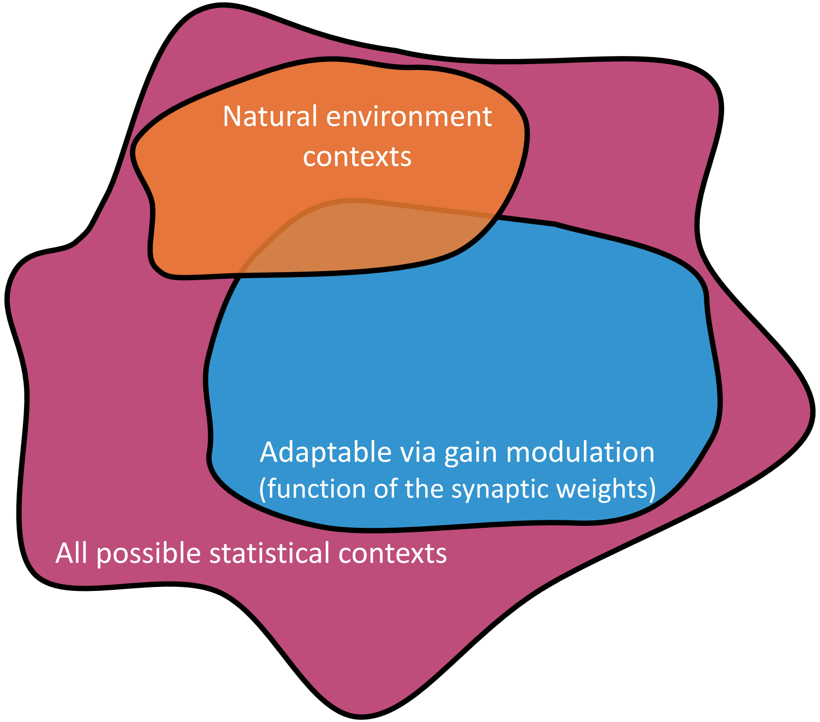
Multi-timescale model intuition



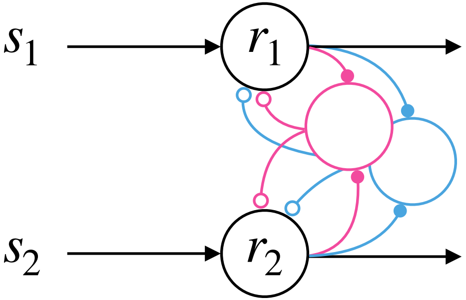


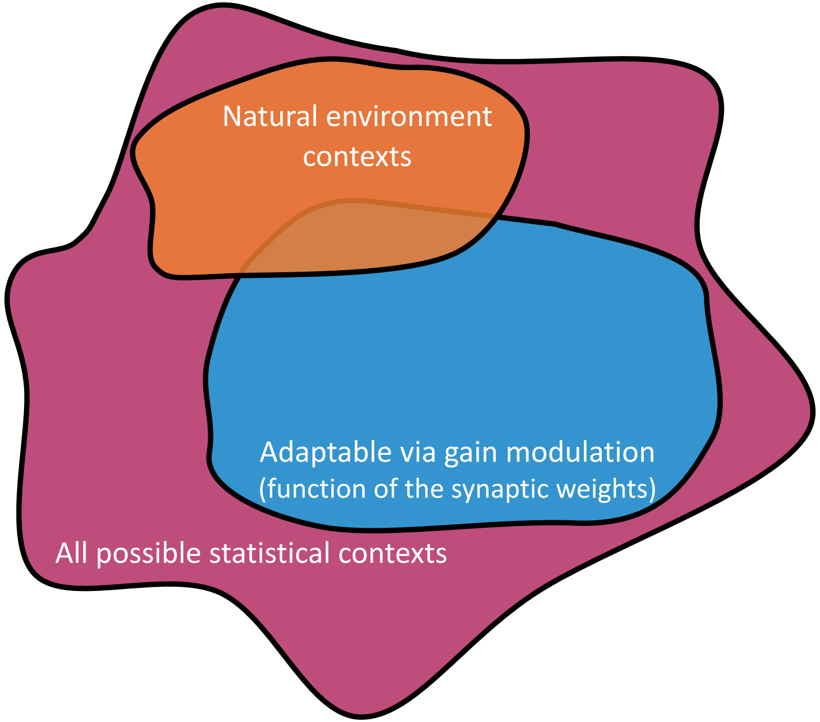
Multi-timescale model intuition



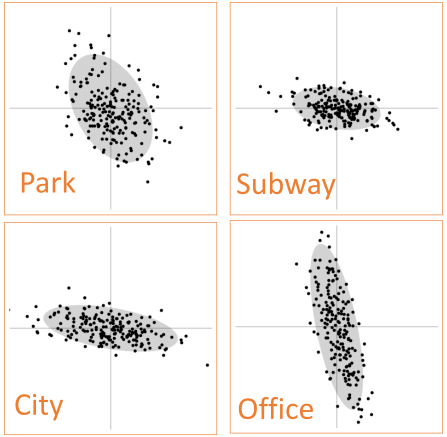


Multi-timescale model intuition

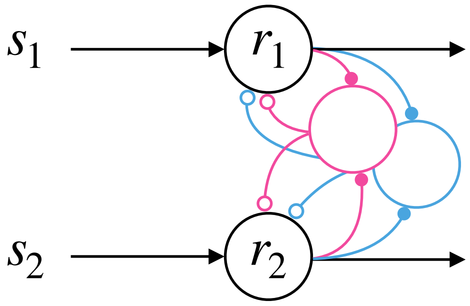


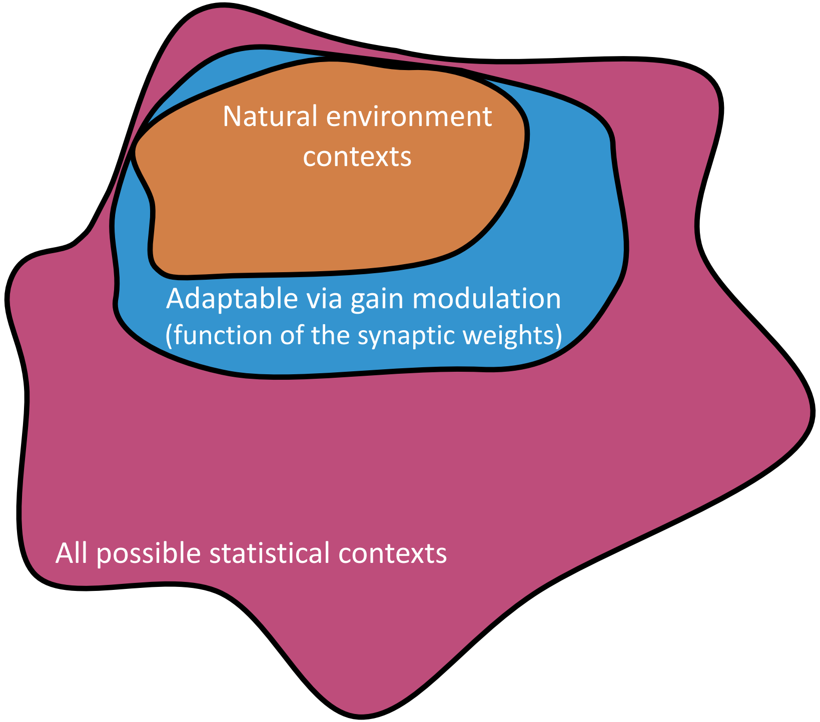


Natural context examples

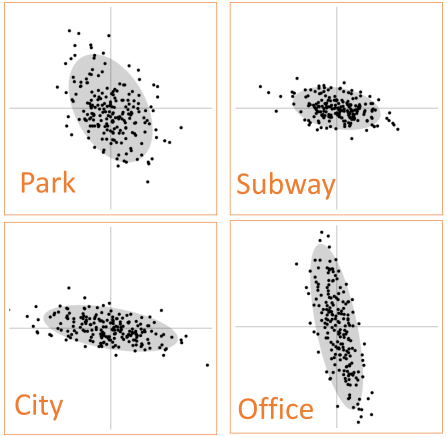


Multi-timescale model intuition





Natural context examples



Adaptive whitening via **gain modulation**

Adaptation objective

Adaptive whitening via **gain modulation**

Adaptation objective

$$\max_{\mathbf{W}} \mathbb{E}_{c \in p(c)} \left[\max_{\mathbf{g}} \mathbb{E}_{\mathbf{s} \sim p(\mathbf{s}|c)} \left[\min_{\mathbf{r}} \ell(\mathbf{W}, \mathbf{g}, \mathbf{s}, \mathbf{r}) \right] \right]$$

$$\ell(\mathbf{W}, \mathbf{g}, \mathbf{s}, \mathbf{r}) = \|\mathbf{r} - \mathbf{s}\|^2 + \sum_{i=1}^{K} g_i \left\{ (\mathbf{w}_i^\top \mathbf{r})^2 - 1 \right\}$$

Adaptive whitening via **gain modulation**

Adaptation objective

$$\max_{\mathbf{W}} \mathbb{E}_{c\in p(c)} \left[\max_{\mathbf{g}} \mathbb{E}_{\mathbf{s}\sim p(\mathbf{s}|c)} \left[\min_{\mathbf{r}} \ell(\mathbf{W}, \mathbf{g}, \mathbf{s}, \mathbf{r}) \right] \right]$$

synapses are optimized across contexts

$$\ell(\mathbf{W}, \mathbf{g}, \mathbf{s}, \mathbf{r}) = \|\mathbf{r} - \mathbf{s}\|^2 + \sum_{i=1}^{K} g_i \left\{ (\mathbf{w}_i^\top \mathbf{r})^2 - 1 \right\}$$

Adaptive whitening via **gain modulation**

Adaptation objective

$$\boxed{\max_{\mathbf{W}} \mathbb{E}_{c \in p(c)}} \left[\boxed{\max_{\mathbf{g}} \mathbb{E}_{\mathbf{s} \sim p(\mathbf{s}|c)}} \left[\min_{\mathbf{r}} \ell(\mathbf{W}, \mathbf{g}, \mathbf{s}, \mathbf{r}) \right] \right]$$

synapses are optimized across contexts

gains are optimized within each contexts

$$\ell(\mathbf{W}, \mathbf{g}, \mathbf{s}, \mathbf{r}) = \|\mathbf{r} - \mathbf{s}\|^2 + \sum_{i=1}^{K} g_i \left\{ (\mathbf{w}_i^\top \mathbf{r})^2 - 1 \right\}$$

Adaptive whitening via **gain modulation**

Adaptation objective

$$\underbrace{\max_{\mathbf{W}} \mathbb{E}_{c \in p(c)}}_{\text{synapses are optimized across contexts}} \left[\underbrace{\max_{\mathbf{g}} \mathbb{E}_{\mathbf{s} \sim p(\mathbf{s}|c)}}_{\text{gains are optimized within each contexts}} \left[\min_{\mathbf{r}} \ell(\mathbf{W}, \mathbf{g}, \mathbf{s}, \mathbf{r}) \right] \right]$$

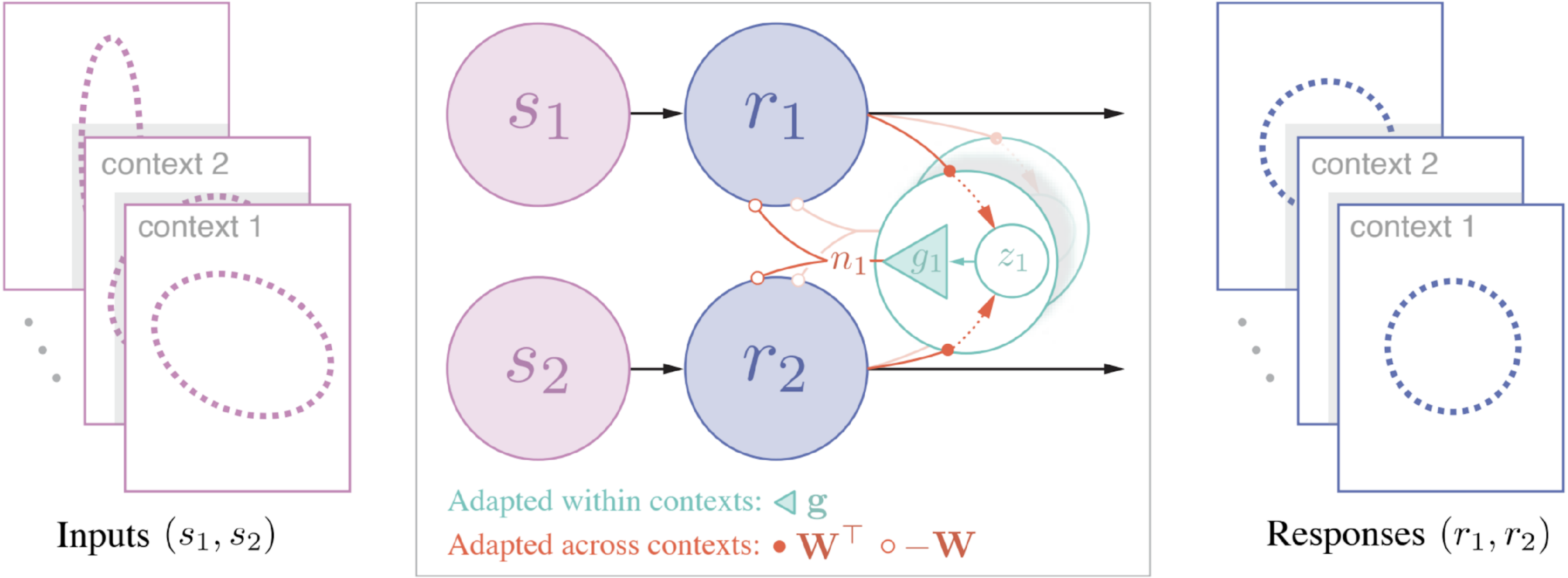
$$\ell(\mathbf{W}, \mathbf{g}, \mathbf{s}, \mathbf{r}) = \|\mathbf{r} - \mathbf{s}\|^2 + \sum_{i=1}^{K} g_i \left\{ (\mathbf{w}_i^\top \mathbf{r})^2 - 1 \right\}$$

Adaptation algorithm

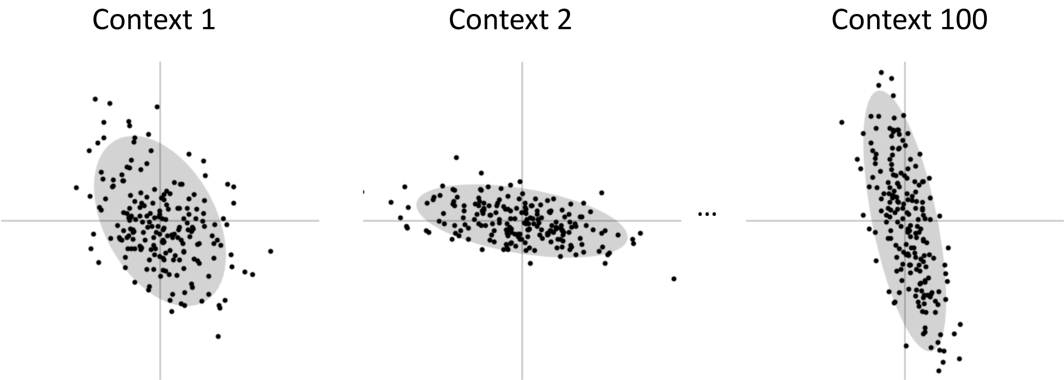
Algorithm 1: Multi-timescale adaptive whitening

1: **Input:** $\mathbf{s}_1, \mathbf{s}_2, \cdots \in \mathbb{R}^N$
2: **for** $t = 1, 2, \ldots$ **do**
3: $\quad \mathbf{r}_t \leftarrow \mathbf{0}$
4: $\quad$ **while** not converged **do**
5: $\quad\quad \mathbf{z}_t \leftarrow \mathbf{W}^\top \mathbf{r}_t$
6: $\quad\quad \mathbf{n}_t \leftarrow \mathbf{g} \circ \mathbf{z}_t$
7: $\quad\quad \mathbf{r}_t \leftarrow \mathbf{r}_t + \eta_r (\mathbf{s}_t - \mathbf{W}\mathbf{n}_t - \alpha \mathbf{r}_t)$
8: $\quad$ **end while**
9: $\quad \mathbf{g} \leftarrow \mathbf{g} + \eta_g \left(\mathbf{z}_t \circ \mathbf{z}_t - \mathrm{diag}\left(\mathbf{W}^\top \mathbf{W} \right) \right)$
10: $\quad \mathbf{W} \leftarrow \mathbf{W} + \eta_w \left(\mathbf{r}_t \mathbf{n}_t^\top - \mathbf{W}\mathrm{diag}(\mathbf{g}) \right)$
11: **end for**

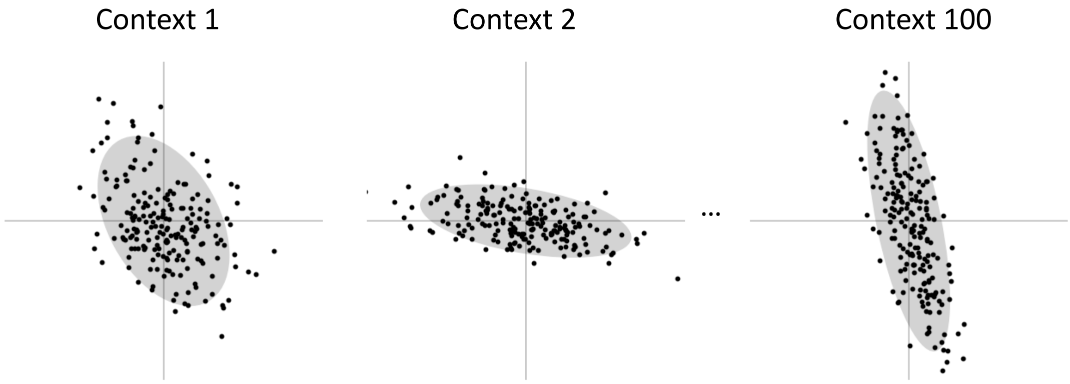
Multi-timescale adaptive RNN architecture



Learning to adapt **across and within contexts**



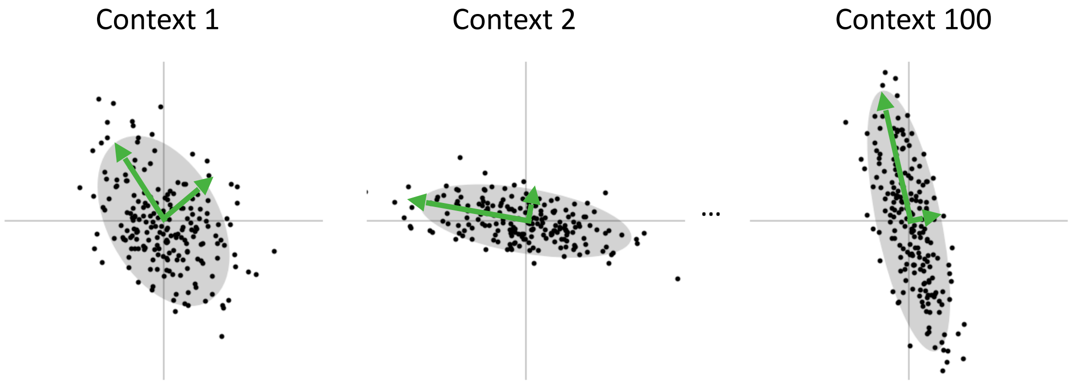
Learning to adapt **across and within contexts**



Training procedure:

1. Sample context from all possible contexts
2. Sample stimulus within context 1000x

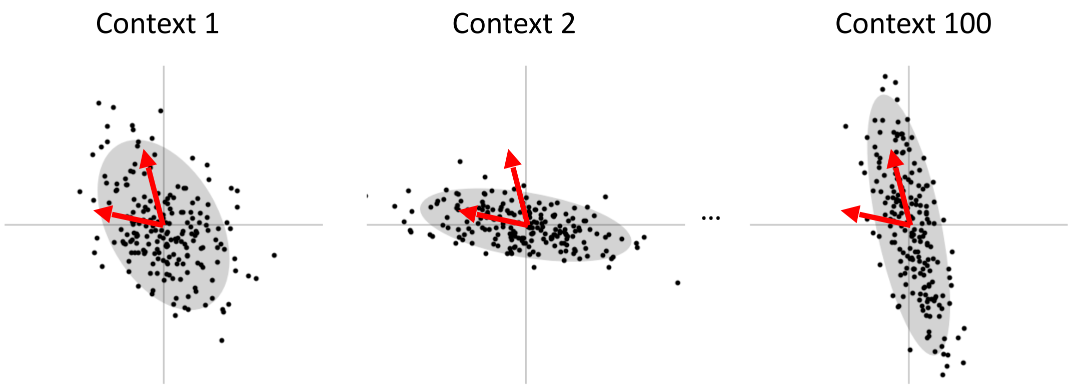
Learning to adapt **across and within contexts**



Training procedure:

1. Sample context from all possible contexts
2. Sample stimulus within context 1000x

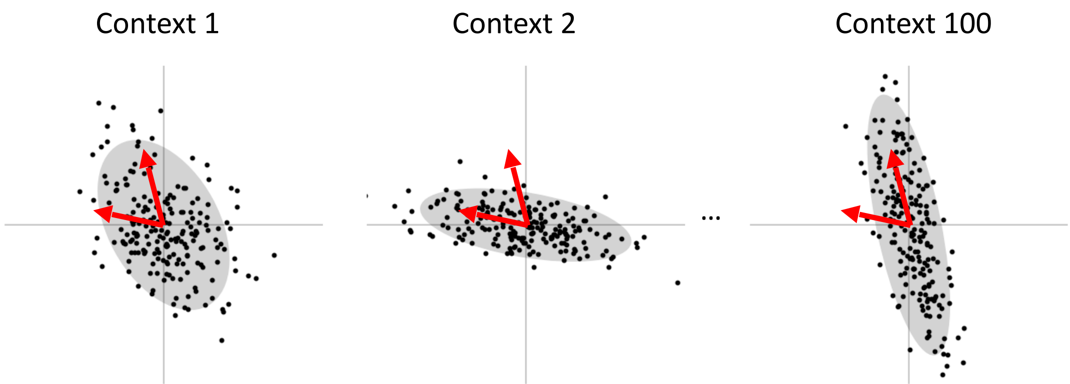
Learning to adapt **across and within contexts**



Training procedure:

1. Sample context from all possible contexts
2. Sample stimulus within context 1000x

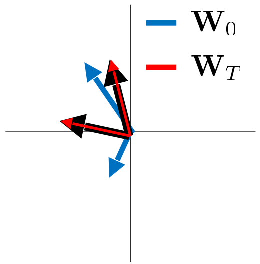
Learning to adapt **across and within contexts**



Training procedure:

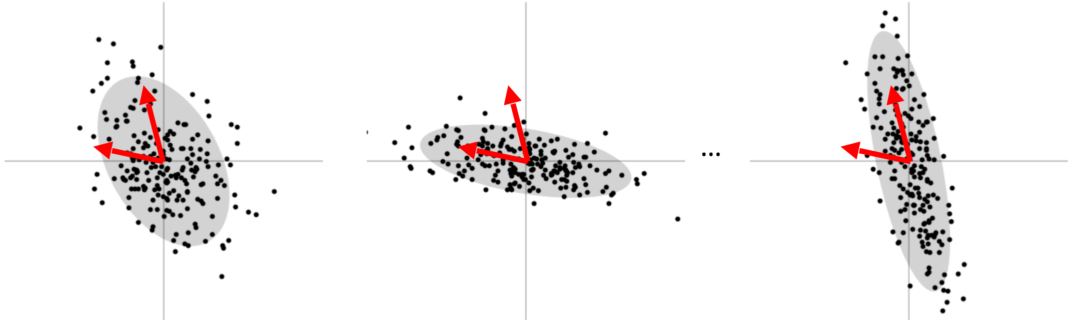
1. Sample context from all possible contexts
2. Sample stimulus within context 1000x

Weights before/after training



Learning to adapt **across and within contexts**

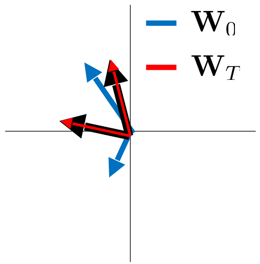
Context 1 Context 2 Context 100



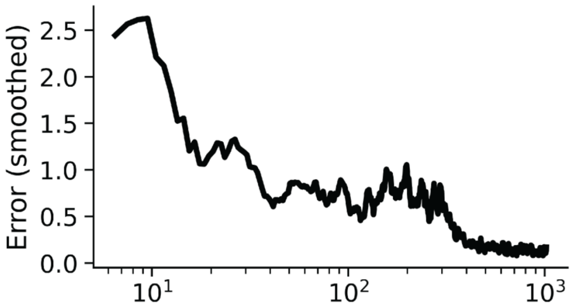
Training procedure:

1. Sample context from all possible contexts
2. Sample stimulus within context 1000x

Weights before/after training

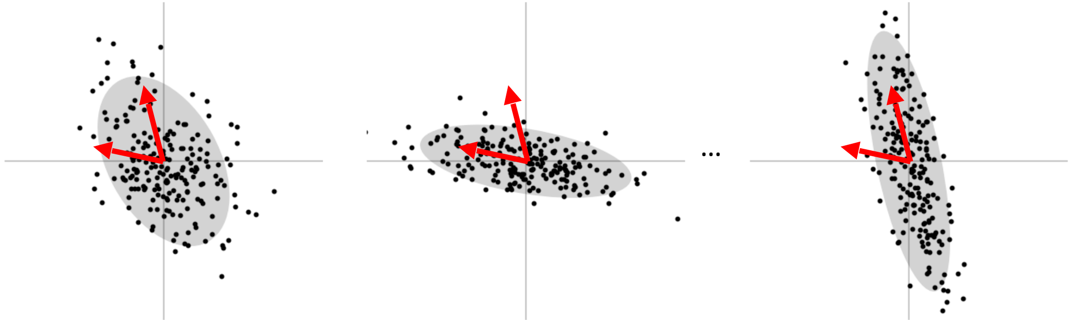


Error through training



Learning to adapt **across and within contexts**

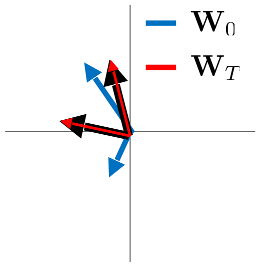
Context 1 Context 2 Context 100



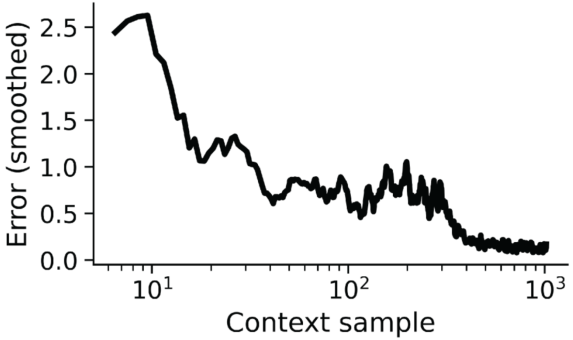
Training procedure:

1. Sample context from all possible contexts
2. Sample stimulus within context 1000x

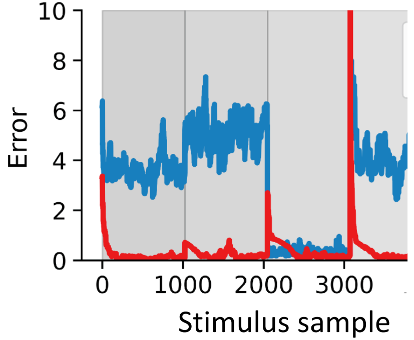
Weights before/after training



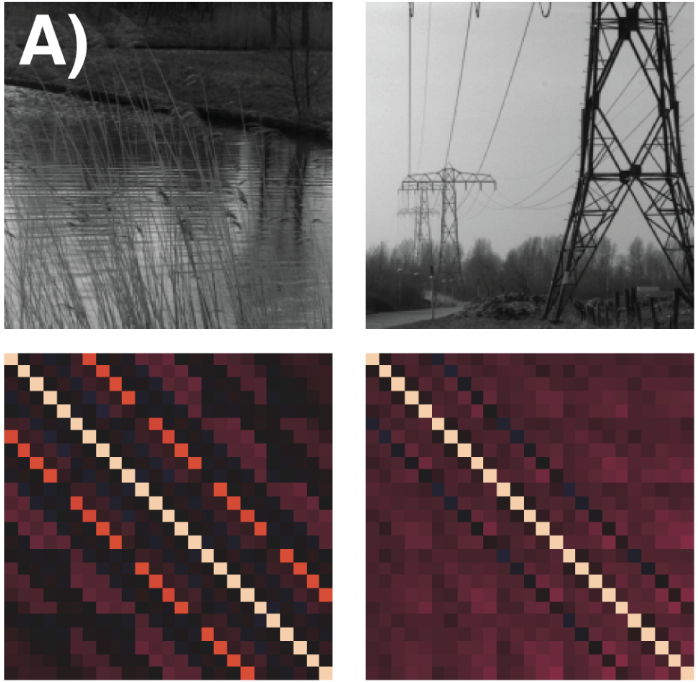
Error through training



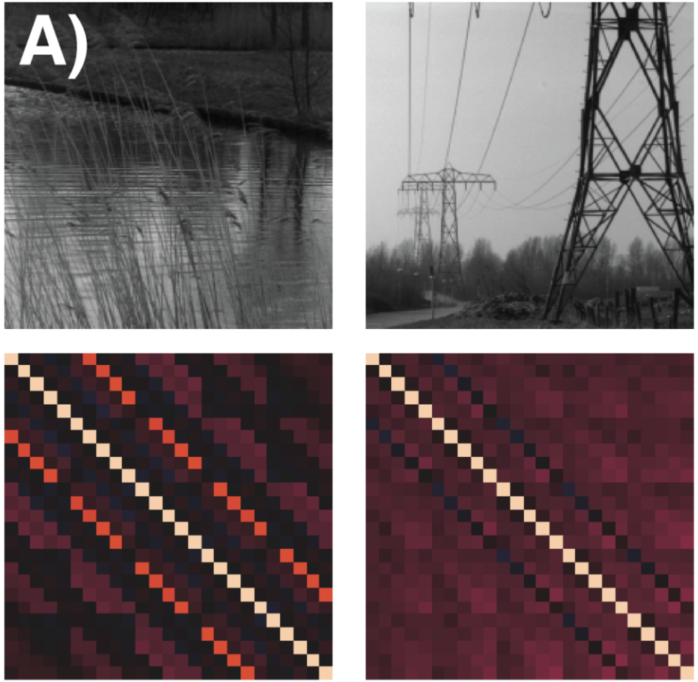
Within-context error before/after training



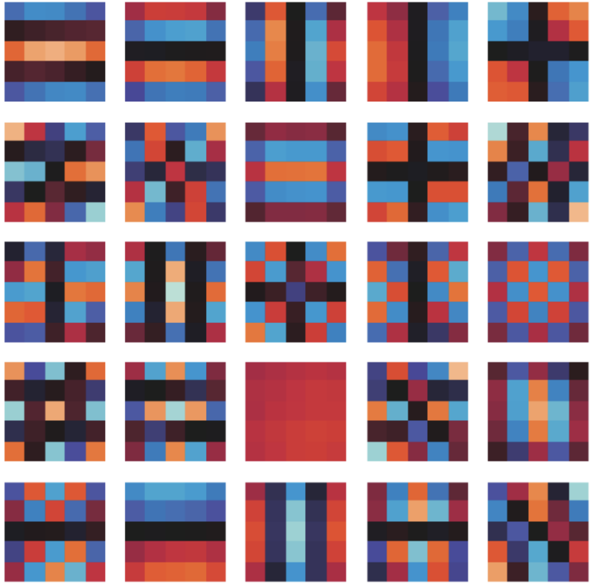
Adaptive whitening of natural images



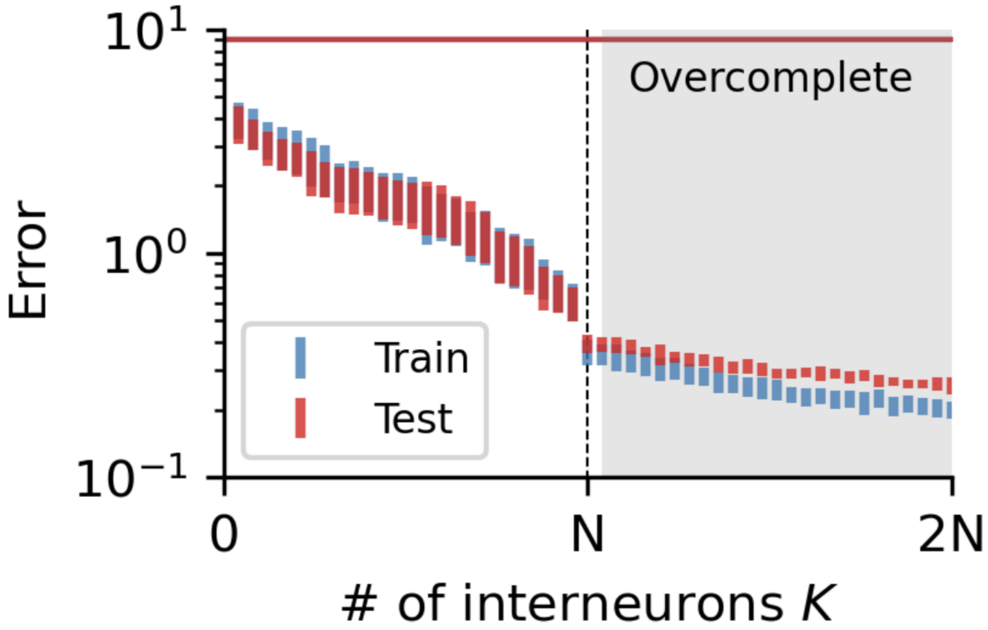
Adaptive whitening of natural images



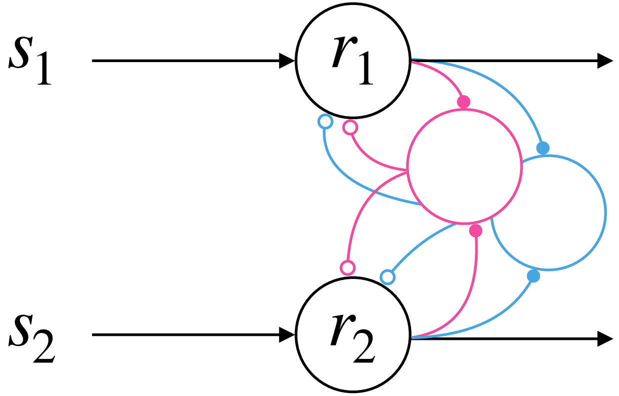
synapses learn 2D sinusoidal filters



Dependence on the # of interneurons

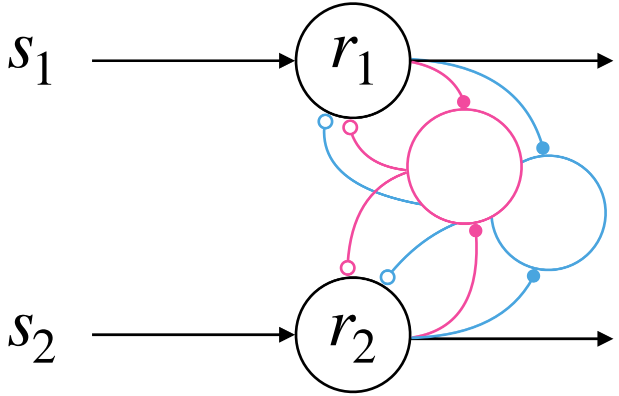


Summary



Summary

Circuit with **fast** gain modulation and **slow** synaptic plasticity

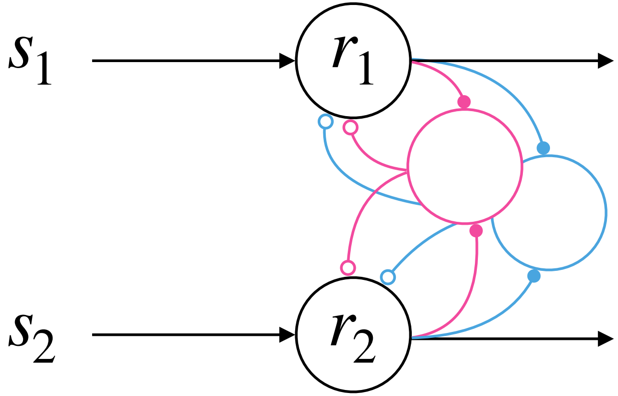


Summary

Circuit with **fast** gain modulation and **slow** synaptic plasticity

Complementary computations:

- **gains** adapt within each context to whiten responses

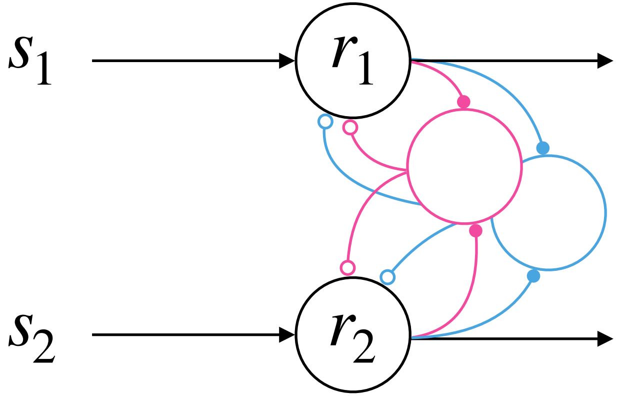


Summary

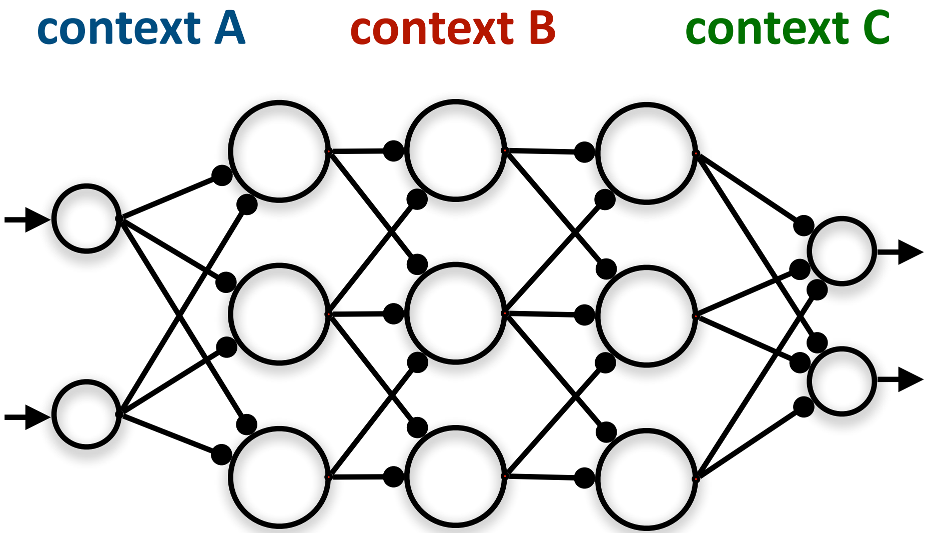
Circuit with **fast** gain modulation and **slow** synaptic plasticity

Complementary computations:

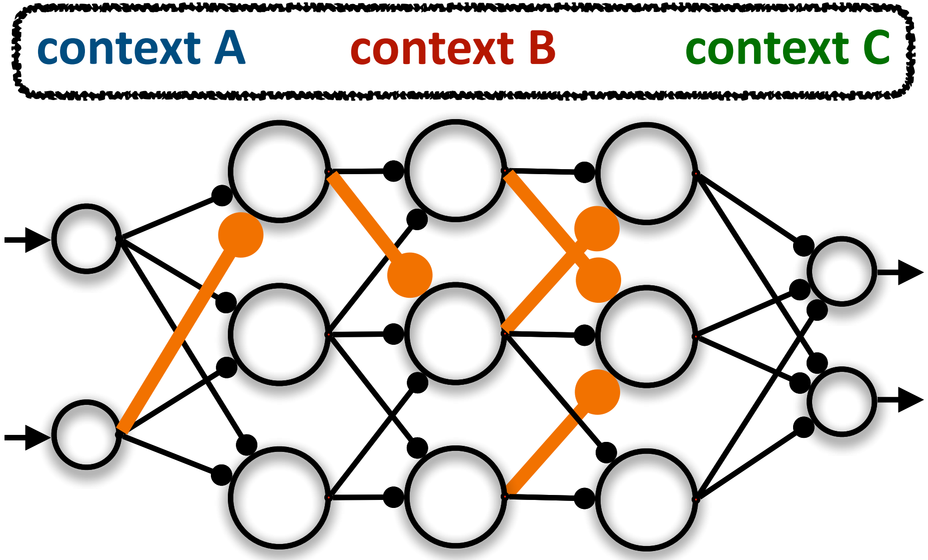
- **gains** adapt within each context to whiten responses
- **synapses** adapt across contexts to learn structural properties of the inputs



Extension: general networks

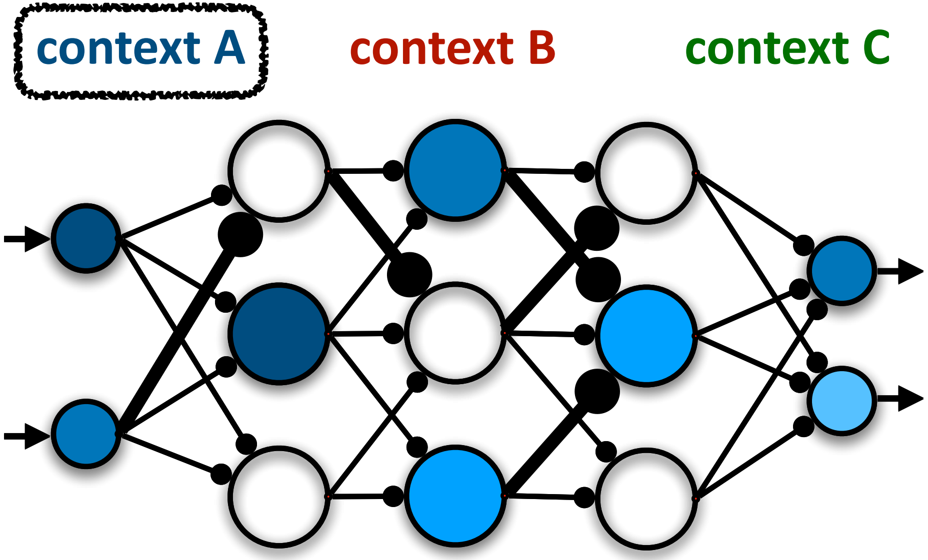


Extension: general networks



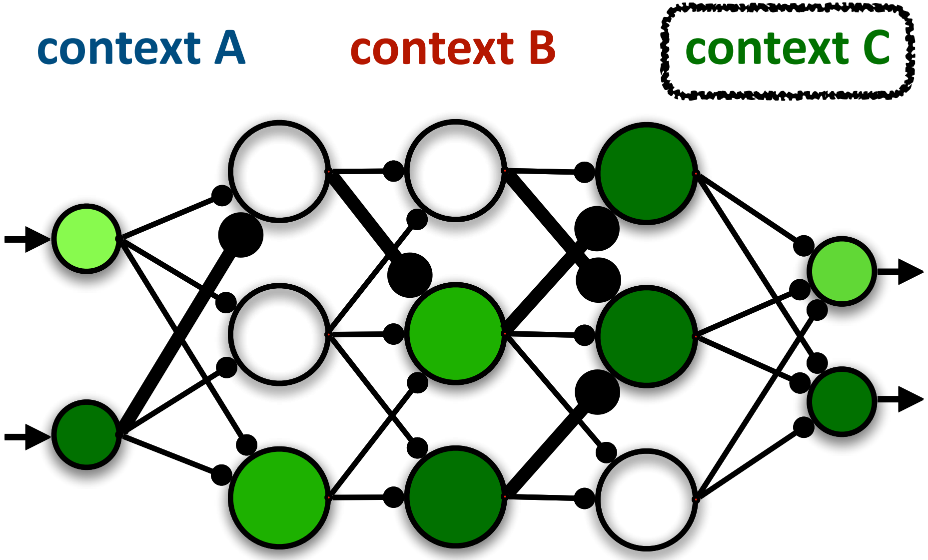
synapses learn
across contexts

Extension: general networks



gains adapts within context

Extension: general networks



gains adapts within context

Thank you

Lyndon Duong
NYU

David Heeger
NYU

Dmitri "Mitya" Chklovskii
Flatiron Institute / NYU

Eero Simoncelli
Flatiron Institute / NYU

FLATIRON INSTITUTE
Center for Computational Neuroscience

Duong*, **Lipshutz***, Heeger, Chklovskii & Simoncelli, Adaptive whitening in neural populations with gain-modulating interneurons. *ICML*, 2023

Duong, Simoncelli, Chklovskii & **Lipshutz**, Adaptive whitening with fast gain modulation and slow synaptic plasticity. *arXiv preprint*, 2023