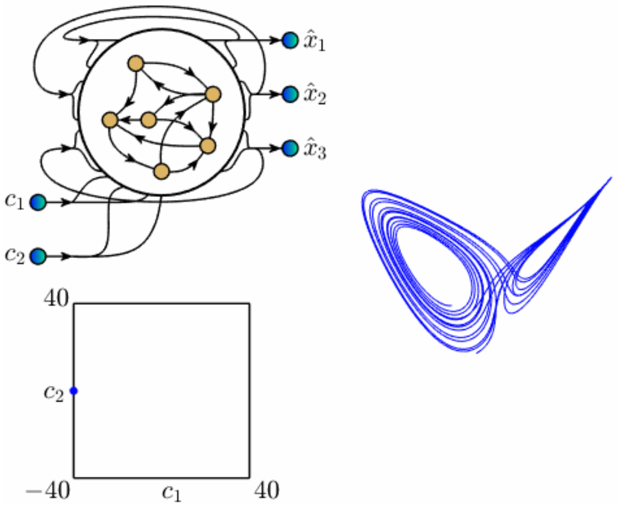
Teaching recurrent neural networks to infer global temporal structure from local examples

Jason Z Kim, Zhixin Lu, Erfan Nozari, George J. Pappas, Danielle S. Bassett

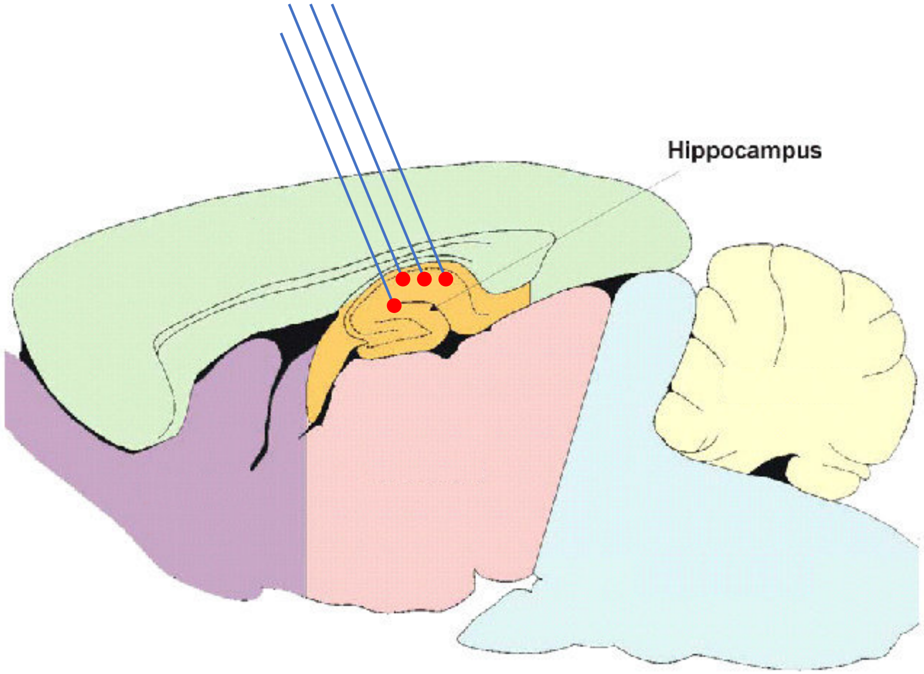


Complex Systems

Penn
University of Pennsylvania

Neural systems sustain and manipulate memories

Spatial localization and forecasting



Neural systems sustain and manipulate memories

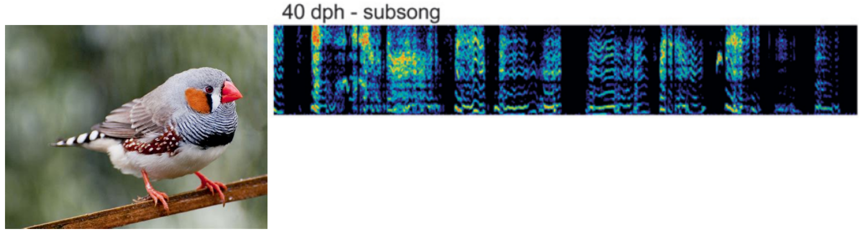
Spatial localization and forecasting

Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song

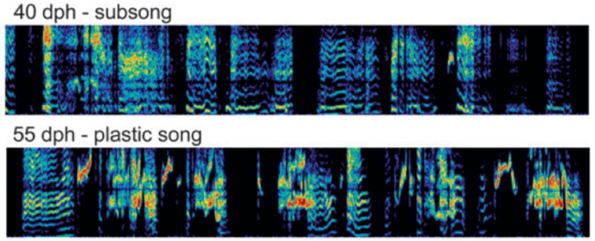


Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song

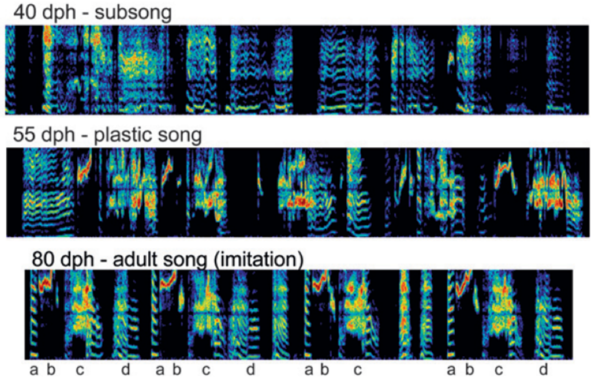


Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song

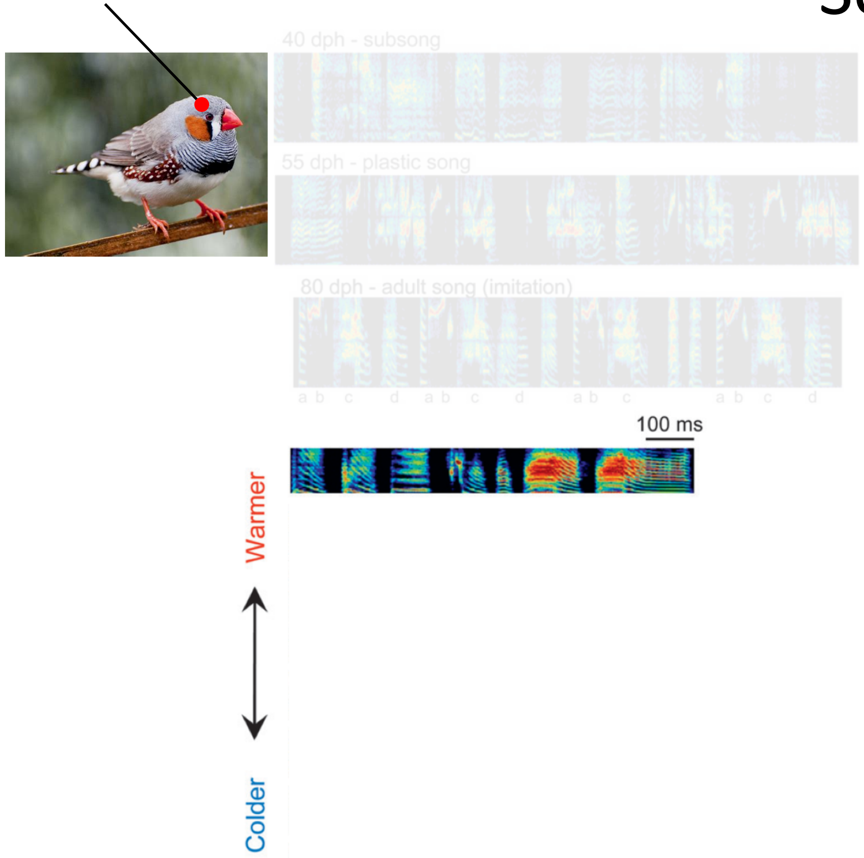


Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song

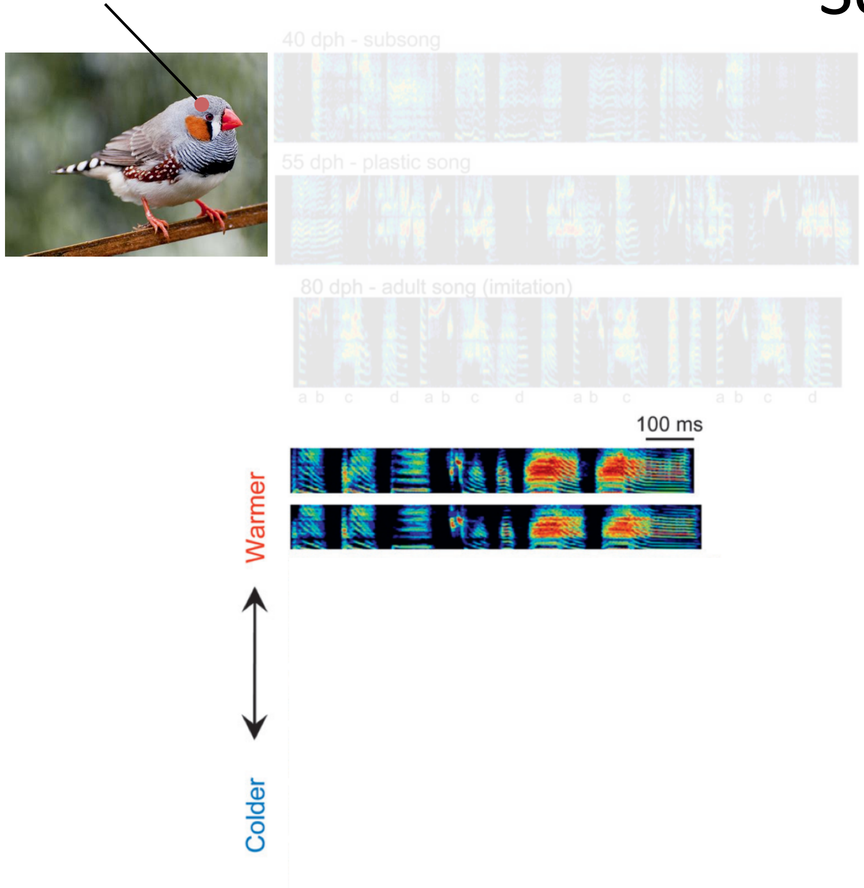


Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song



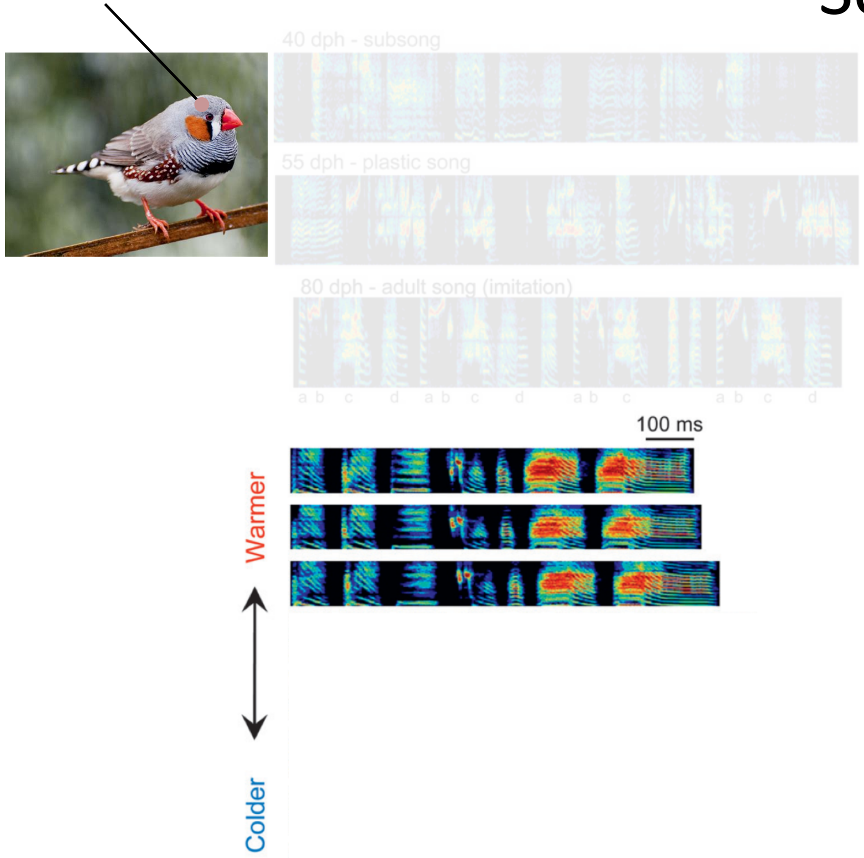
Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112

Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song

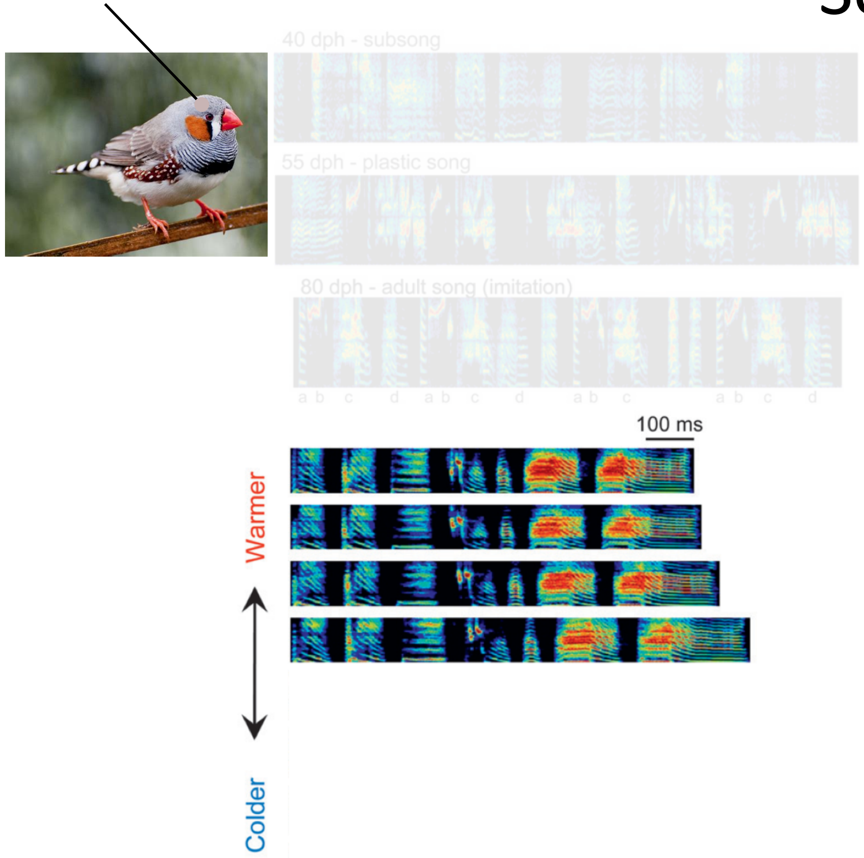


Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song

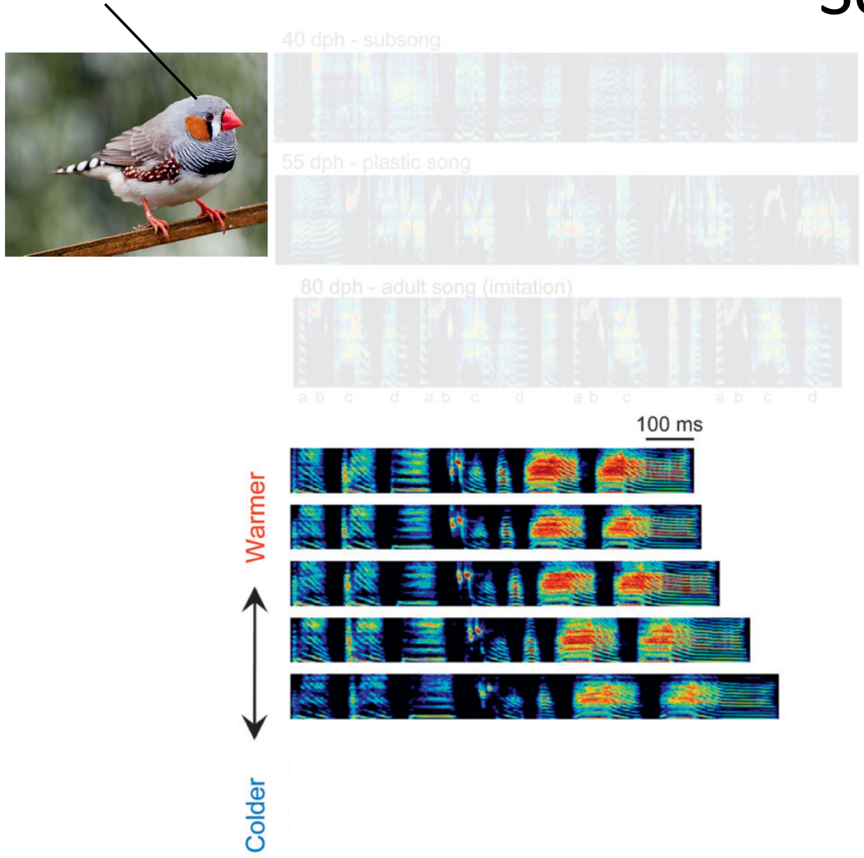


Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song

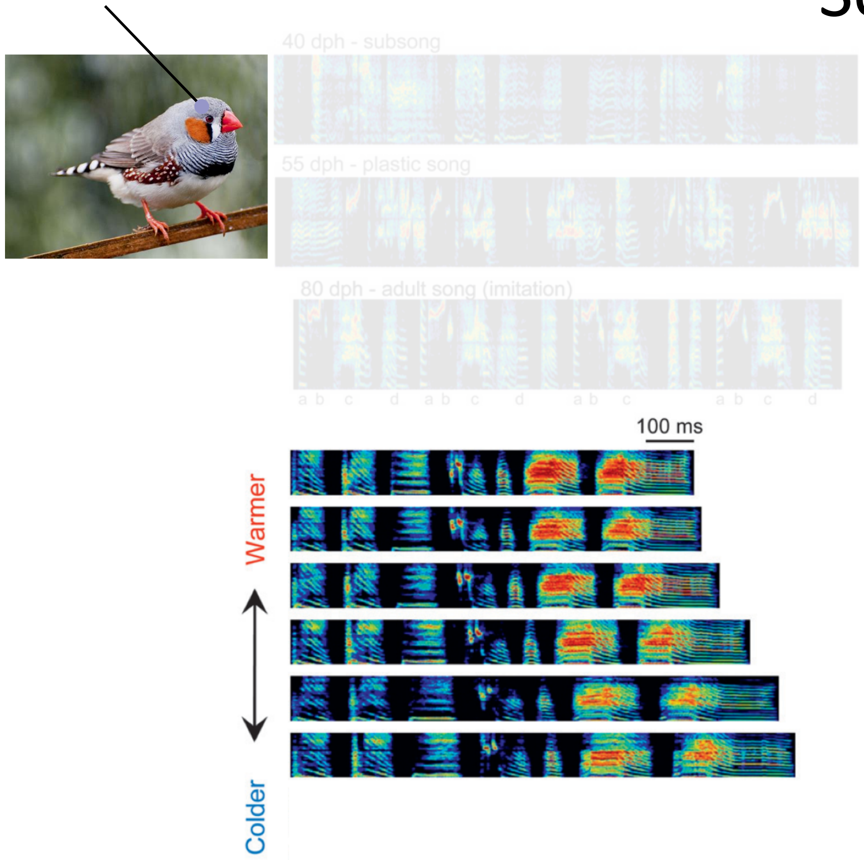


Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song

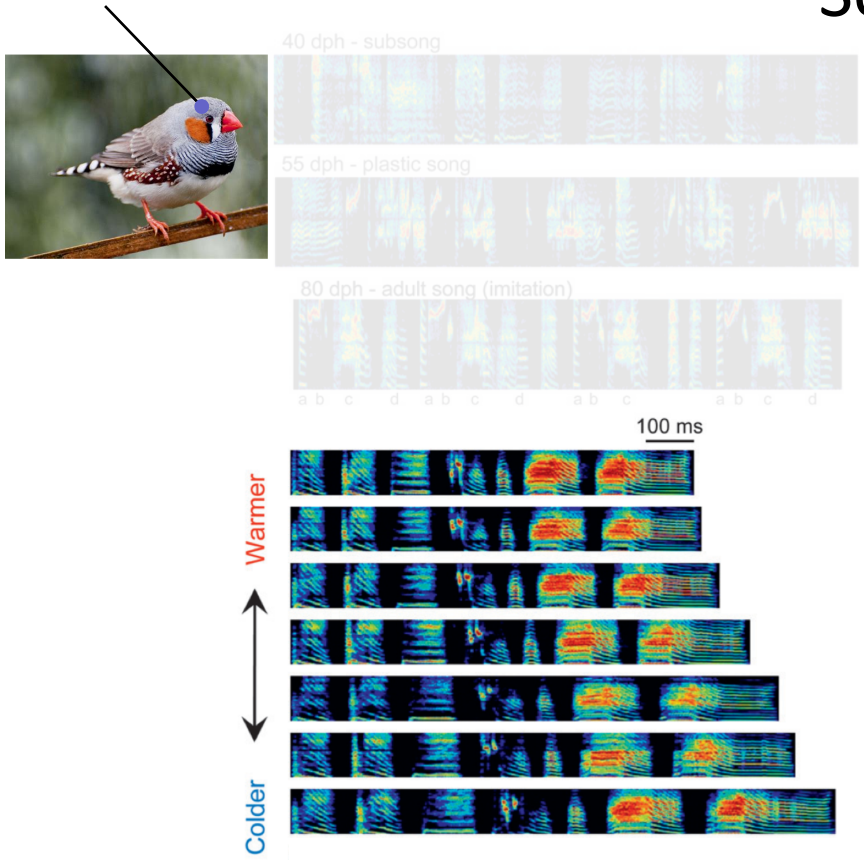


Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song



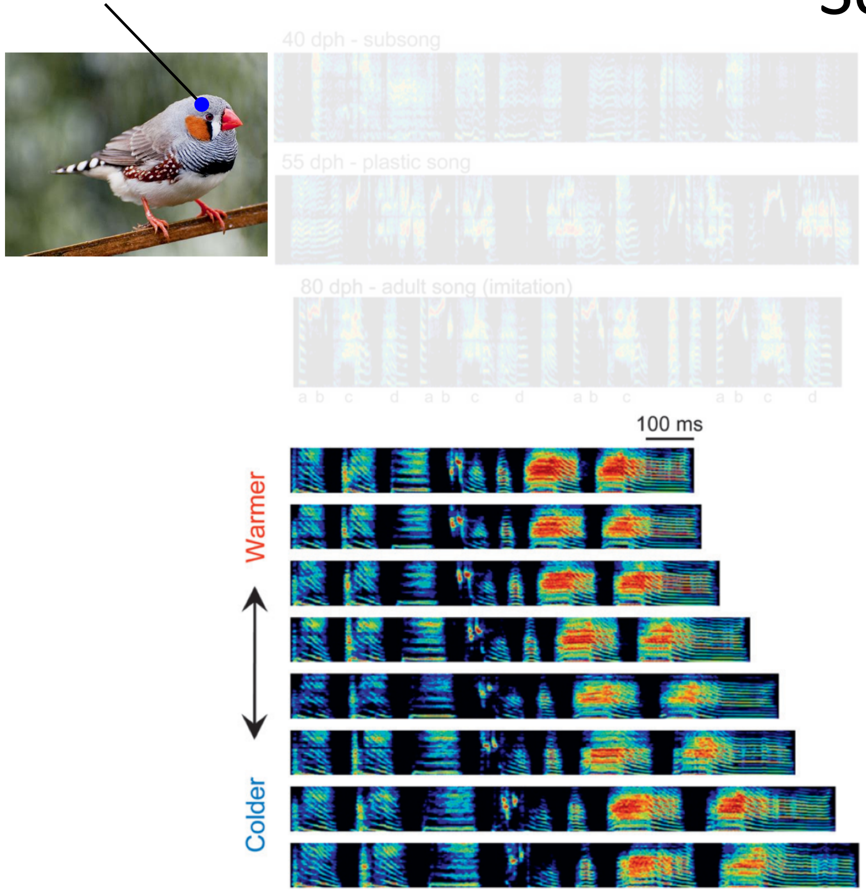
Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112

Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Neural systems sustain and manipulate memories

Spatial localization and forecasting

Song



Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79. https://doi.org/10.1038/nature12112
Fee, M. S., & Scharff, C. (2010). The Songbird as a Model for the Generation and Learning of Complex Sequential Behaviors. *ILAR Journal*, *51*(4), 362–377. https://doi.org/10.1093/ilar.51.4.362

Computers are engineered, neural systems are trained

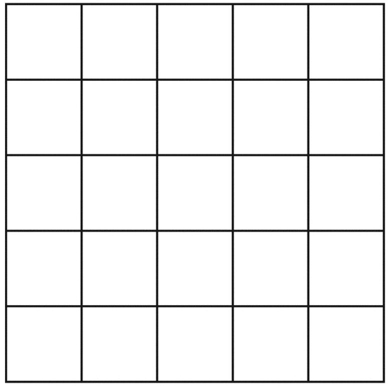
ıal

solely by observing examples?

Computers are engineered, neural systems are trained

- Computers: binary memory

al

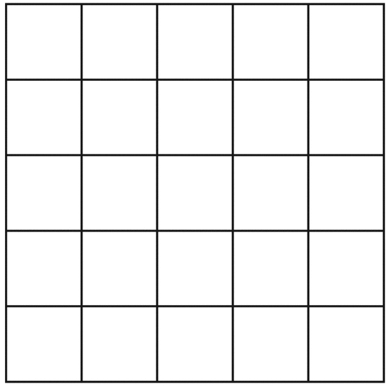


solely by observing examples?

Computers are engineered, neural systems are trained

- Computers: binary memory
- Task: 7 + 17

ıal



solely by observing examples?

Computers are engineered, neural systems are trained

- Computers: binary memory
- Task: 7 + 17
  - Encode: decimal => binary

ıal

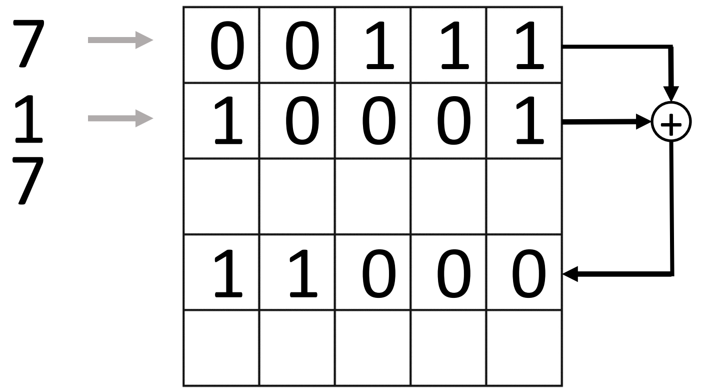
7 →	0	0	1	1	1
17 →	1	0	0	0	1

solely by observing examples?

Computers are engineered, neural systems are trained

- Computers: binary memory
- Task: 7 + 17
  - Encode: decimal => binary
  - Modify: operations in binary

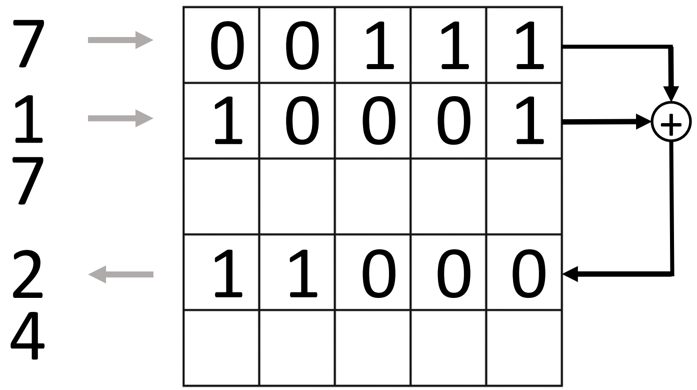
ıal



solely by observing examples?

Computers are engineered, neural systems are trained

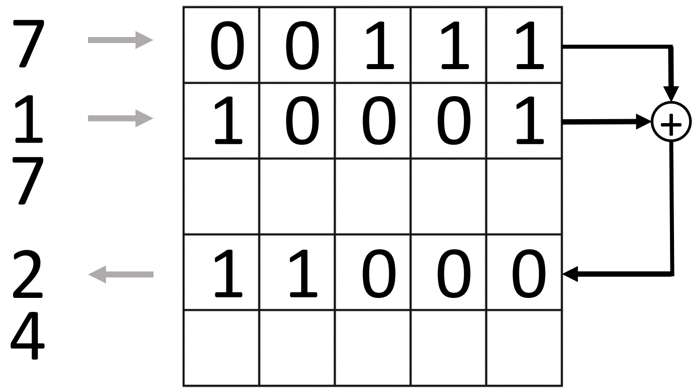
- Computers: binary memory
- Task: 7 + 17
  - Encode: decimal => binary
  - Modify: operations in binary
  - Decode: binary => decimal



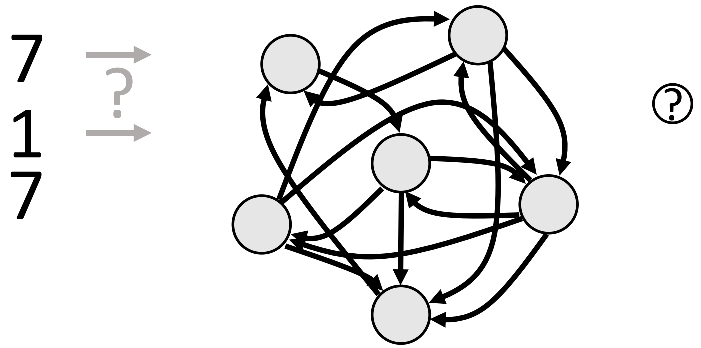
solely by observing examples?

Computers are engineered, neural systems are trained

- Computers: binary memory
- Task: 7 + 17
  - Encode: decimal => binary
  - Modify: operations in binary
  - Decode: binary => decimal



- Brain: neural memory
- Task: 7 + 17
  - Encode: ?
  - Modify: ?
  - Decode: ?

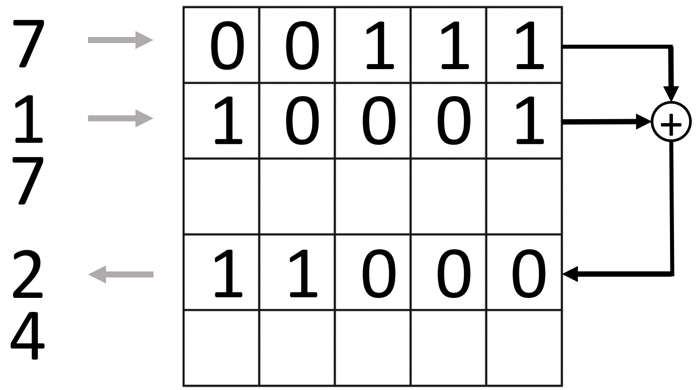


solely by observing examples?

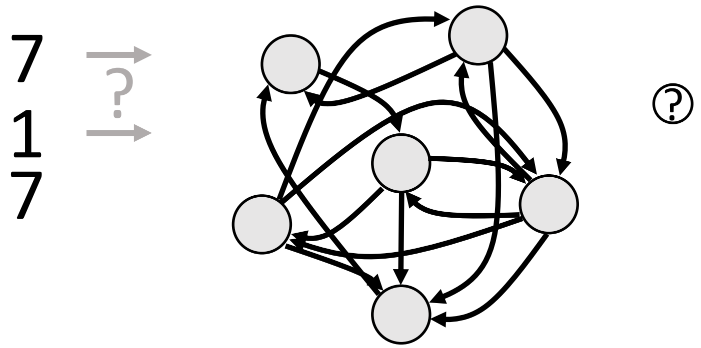
University of Pennsylvania

Computers are engineered, neural systems are trained

- Computers: binary memory
- Task: 7 + 17
  - Encode: decimal => binary
  - Modify: operations in binary
  - Decode: binary => decimal

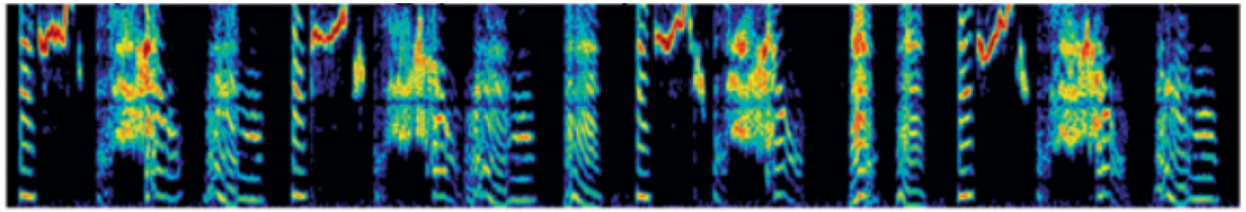


- Brain: neural memory
- Task: 7 + 17
  - Encode: ?
  - Modify: ?
  - Decode: ?



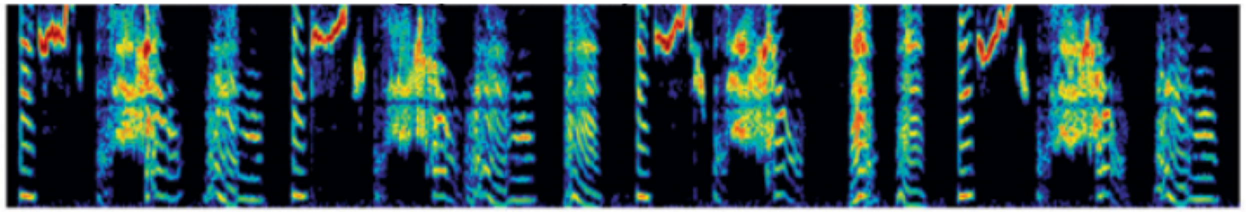
How can neural systems learn to manipulate information solely by observing examples?

Model of memory



Unpredictable yet structured data

- Temporal (changes with time)
- Structured (not completely random)
- Complex (unpredictable)



Unpredictable yet structured data

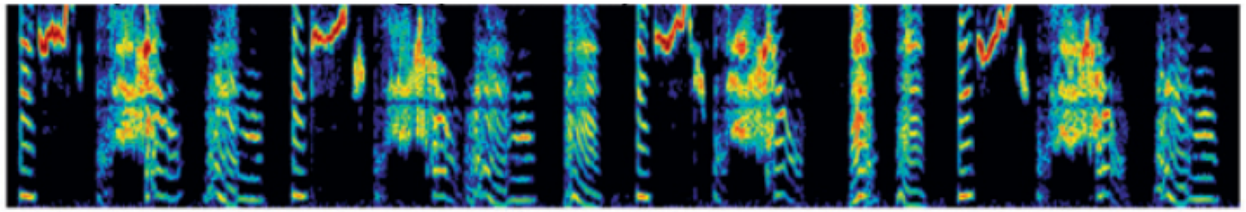
- Temporal (changes with time)
- Structured (not completely random)
- Complex (unpredictable)

Lorenz Attractor

$$\dot{x}_1 = \sigma(x_2 - x_1)$$

$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$

$$\dot{x}_3 = x_1 x_2 - \beta x_3,$$



Unpredictable yet structured data

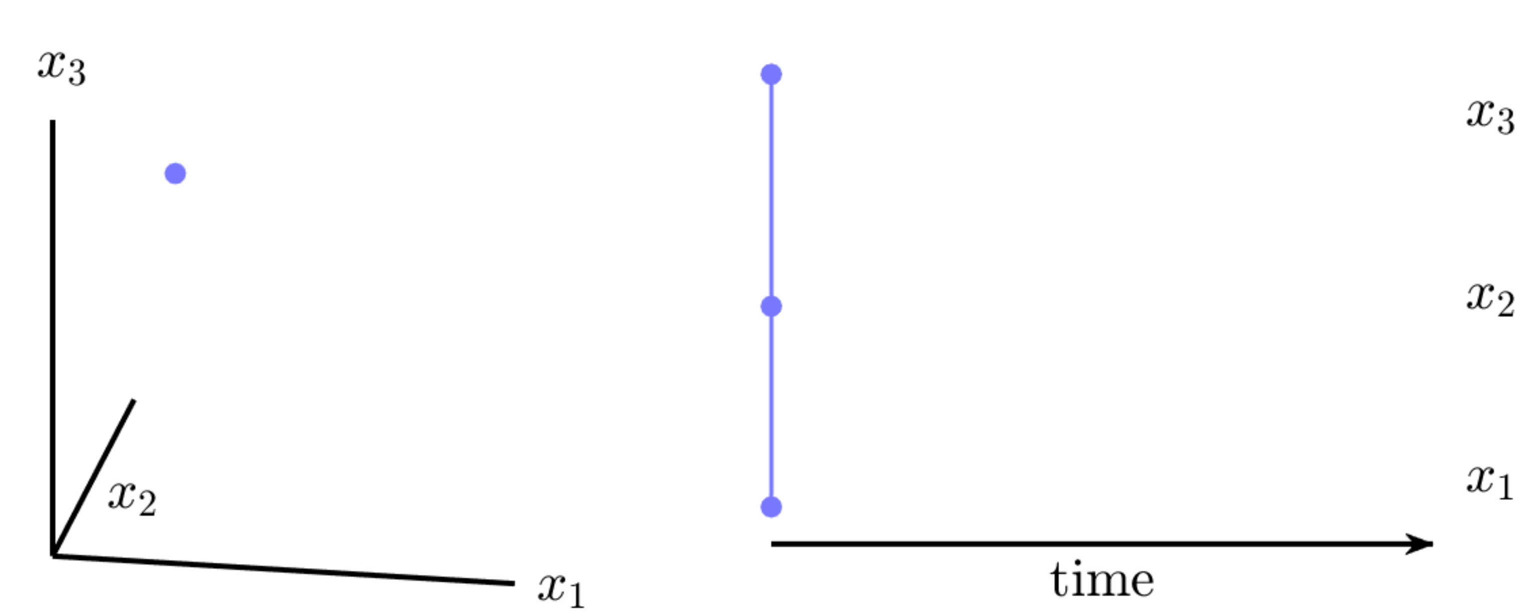
- Temporal (changes with time)
- Structured (not completely random)
- Complex (unpredictable)

Lorenz Attractor

$$\dot{x}_1 = \sigma(x_2 - x_1)$$

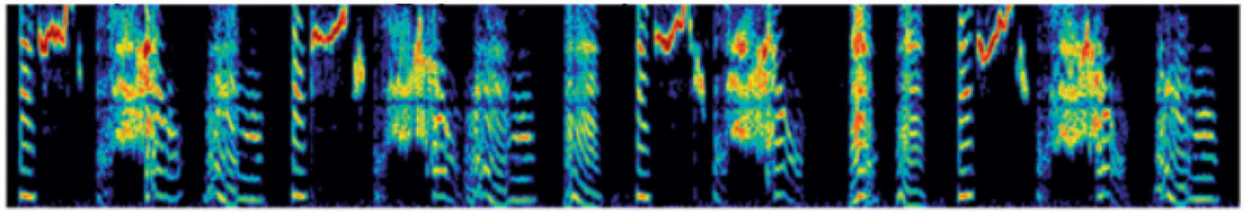
$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$

$$\dot{x}_3 = x_1 x_2 - \beta x_3,$$



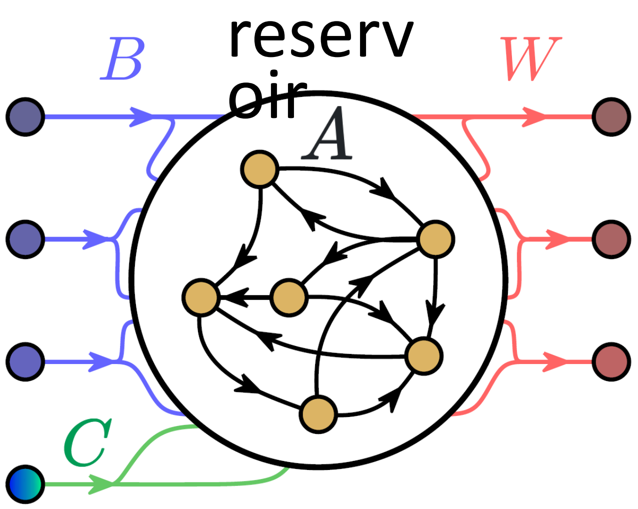
Penn
UNIVERSITY of PENNSYLVANIA

Model of neural network



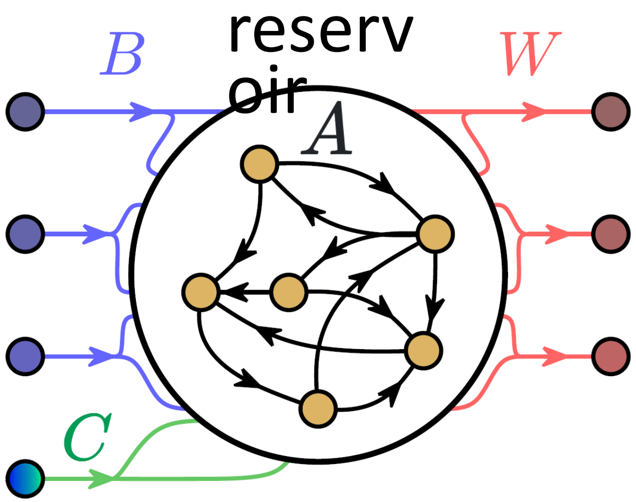
UNIVERSITY of PENNSYLVANIA

Recurrent neural network (reservoir computer)



Recurrent neural network (reservoir computer)

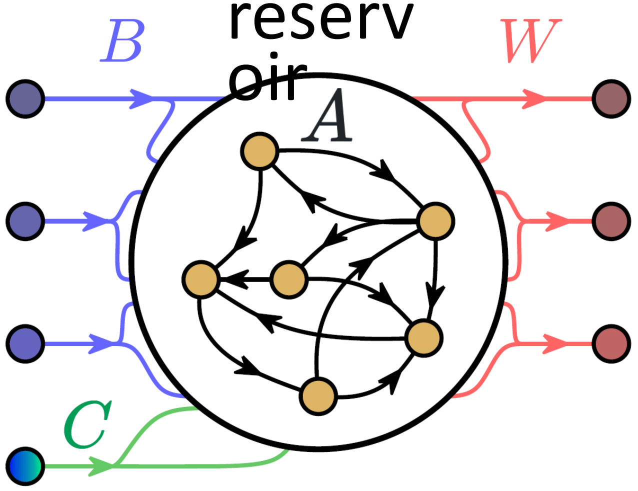
$$\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + C\boldsymbol{c} + \boldsymbol{d})$$



Recurrent neural network (reservoir computer)

reserv
oir

$$\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + C\boldsymbol{c} + \boldsymbol{d})$$

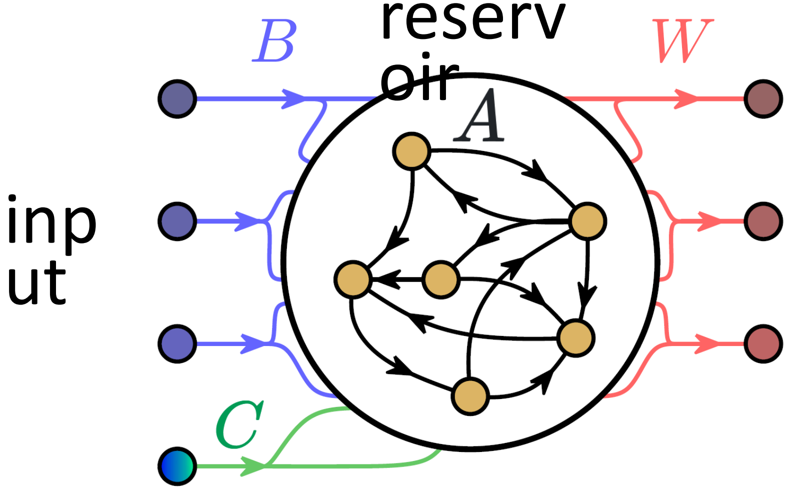


Recurrent neural network (reservoir computer)

reserv
oir

inp
ut

$$\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + C\boldsymbol{c} + \boldsymbol{d})$$



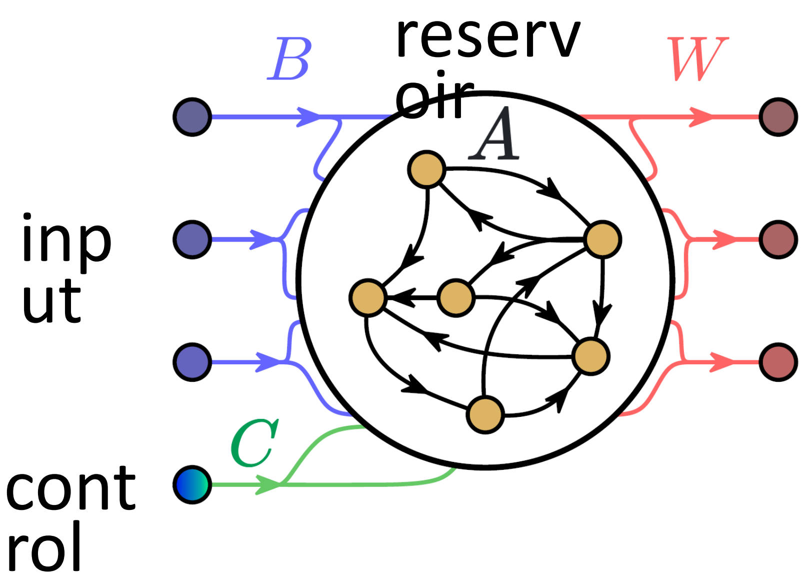
Recurrent neural network (reservoir computer)

reserv
oir

inp
ut

cont
rol

$$\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + C\boldsymbol{c} + \boldsymbol{d})$$



Recurrent neural network (reservoir computer)

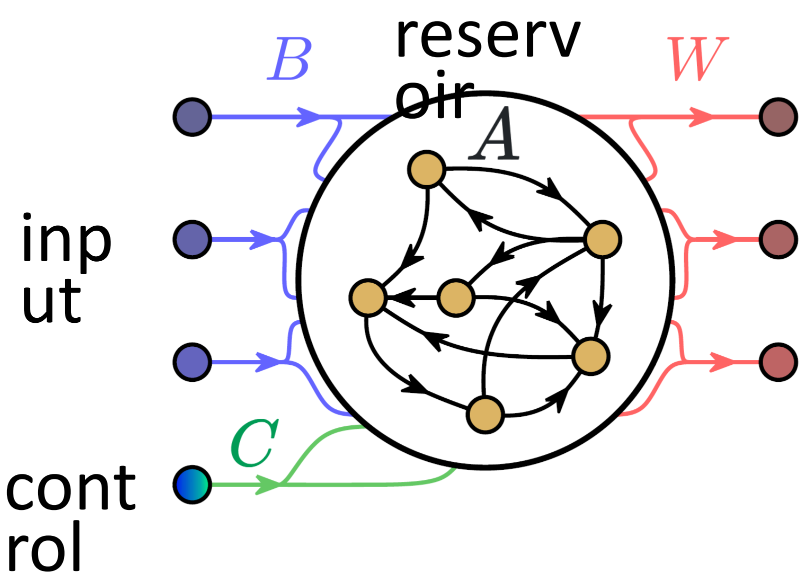
reserv
oir

inp
ut

cont
rol

bia
s

$$\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + C\boldsymbol{c} + \boldsymbol{d})$$

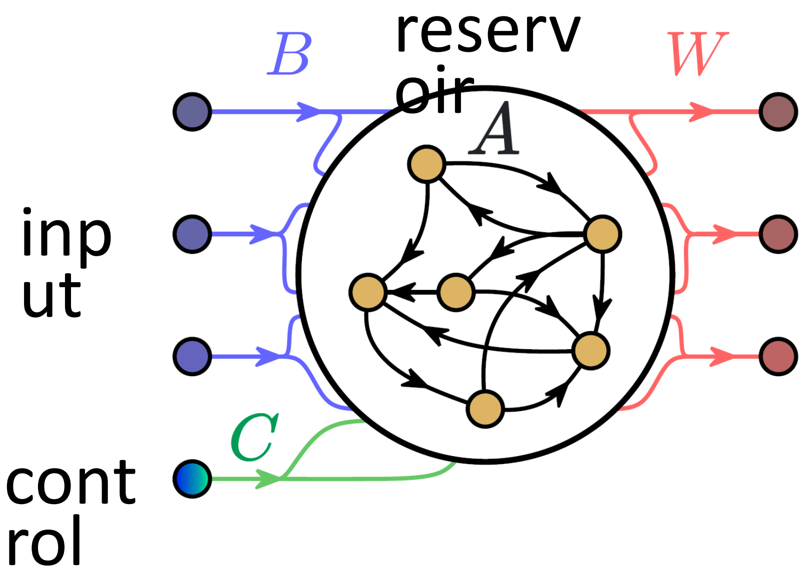


Recurrent neural network (reservoir computer)

reserv oir inp ut cont rol bia s

$$\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + C\boldsymbol{c} + \boldsymbol{d})$$

sigm oid



Recurrent neural network (reservoir computer)

reserv
oir

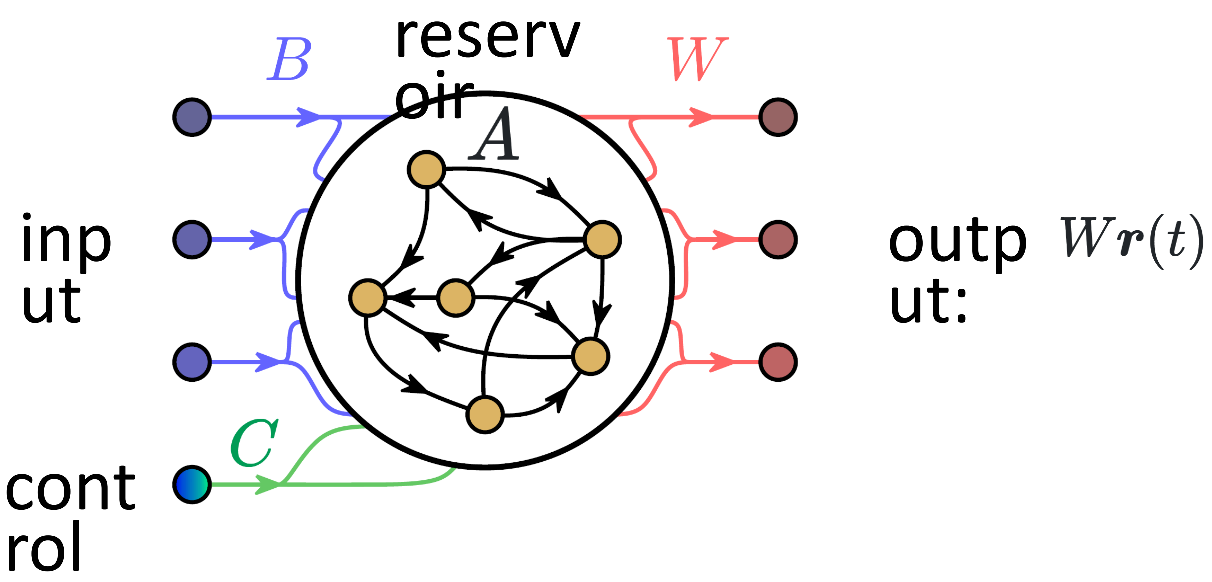
inp
ut

cont
rol

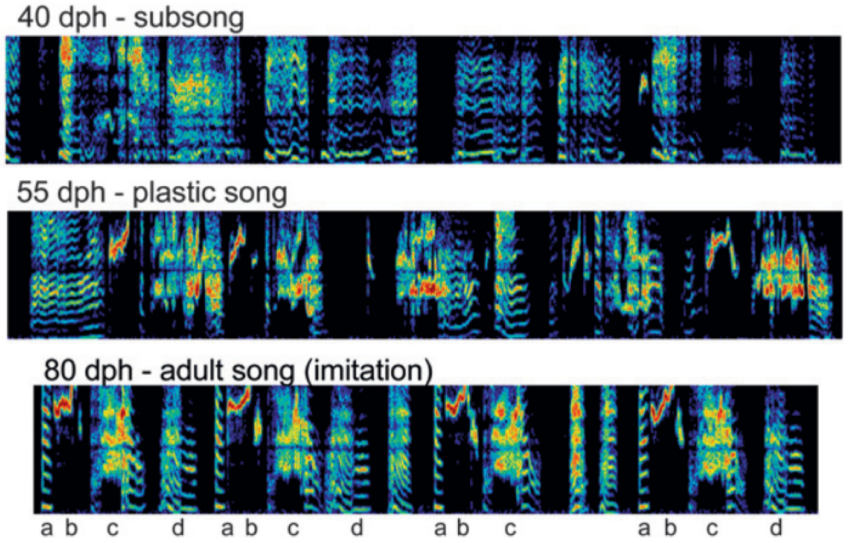
bia
s

$$\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + C\boldsymbol{c} + \boldsymbol{d})$$

sigm
oid

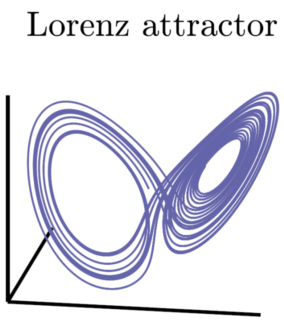


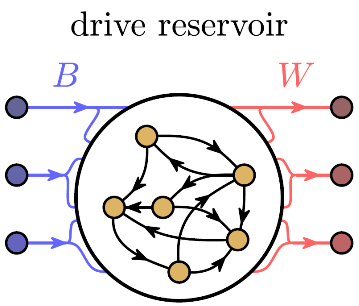
Learning memories by imitating examples



RNNs sustain chaotic memories by imitating examples

Driving $\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + \boldsymbol{d})$

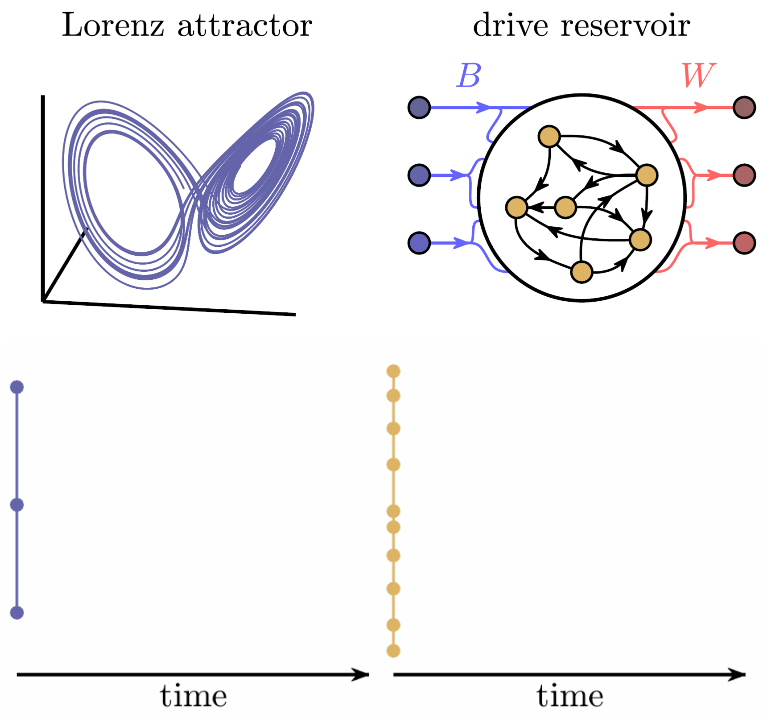




Sussillo, D., & Abbott, L. F. (2009). Generating Coherent Patterns of Activity from Chaotic Neural Networks. *Neuron*, *63*(4), 544–557.
Jaeger, H. (2010). The " echo state " approach to analysing and training recurrent neural networks – with an Erratum note. *GMD Report*, *1*(148), 1–47.

RNNs sustain chaotic memories by imitating examples

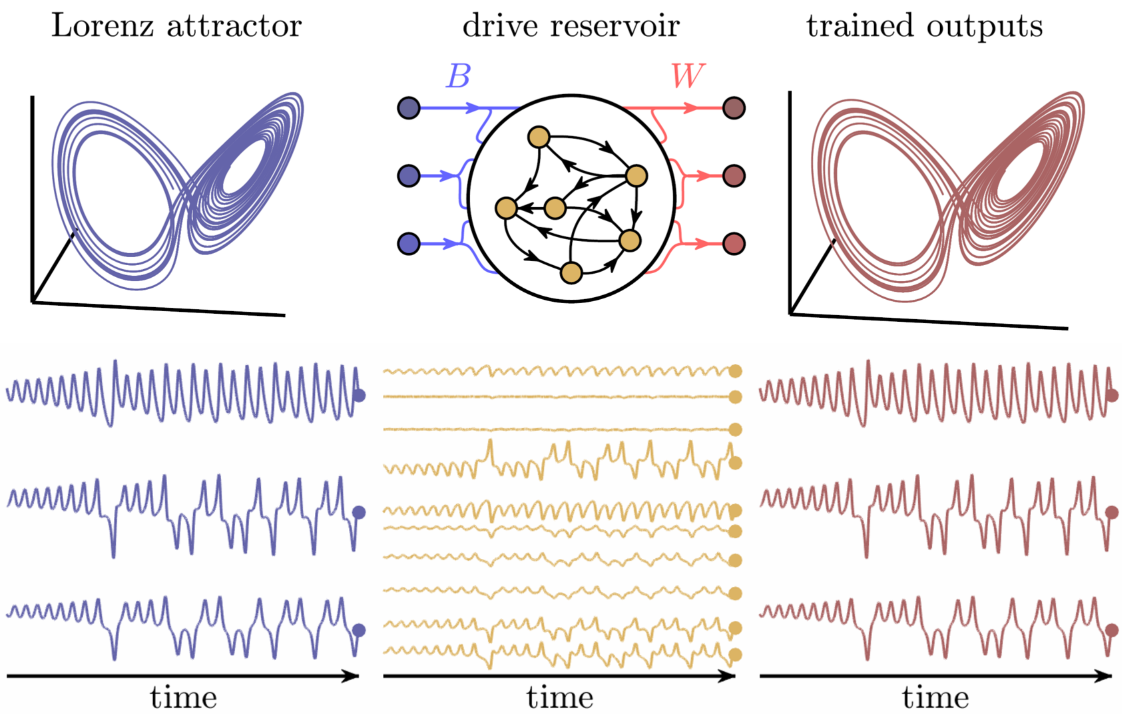
Driving $\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + \boldsymbol{d})$



Sussillo, D., & Abbott, L. F. (2009). Generating Coherent Patterns of Activity from Chaotic Neural Networks. *Neuron*, *63*(4), 544–557.
Jaeger, H. (2010). The " echo state " approach to analysing and training recurrent neural networks – with an Erratum note. *GMD Report*, *1*(148), 1–47.

RNNs sustain chaotic memories by imitating examples

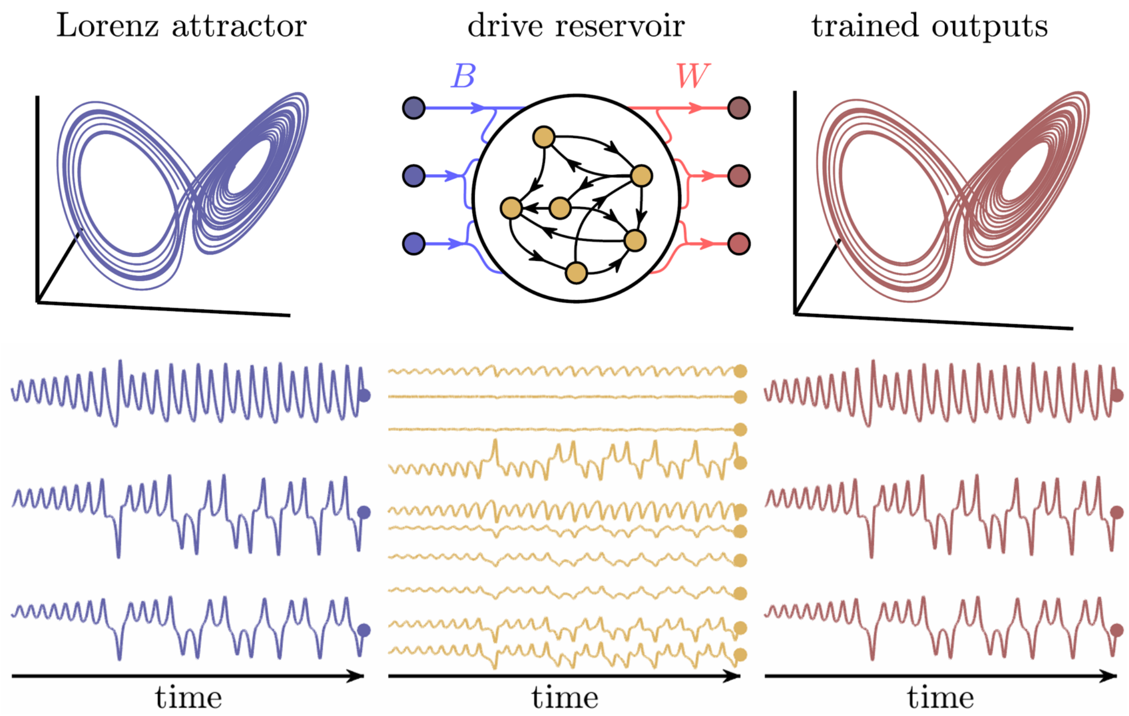
Driving $\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + \boldsymbol{d})$ → Trainin

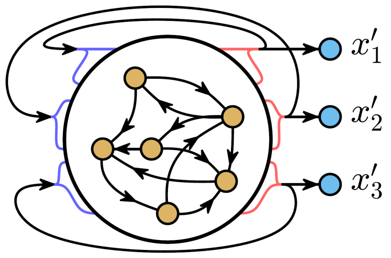


Sussillo, D., & Abbott, L. F. (2009). Generating Coherent Patterns of Activity from Chaotic Neural Networks. *Neuron*, *63*(4), 544–557.
Jaeger, H. (2010). The “ echo state ” approach to analysing and training recurrent neural networks – with an Erratum note. *GMD Report*, *1*(148), 1–47.

RNNs sustain chaotic memories by imitating examples

Driving $\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + \boldsymbol{d})$ → Training: W → Pr $\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}([A + BW]\boldsymbol{r} + \boldsymbol{d})$

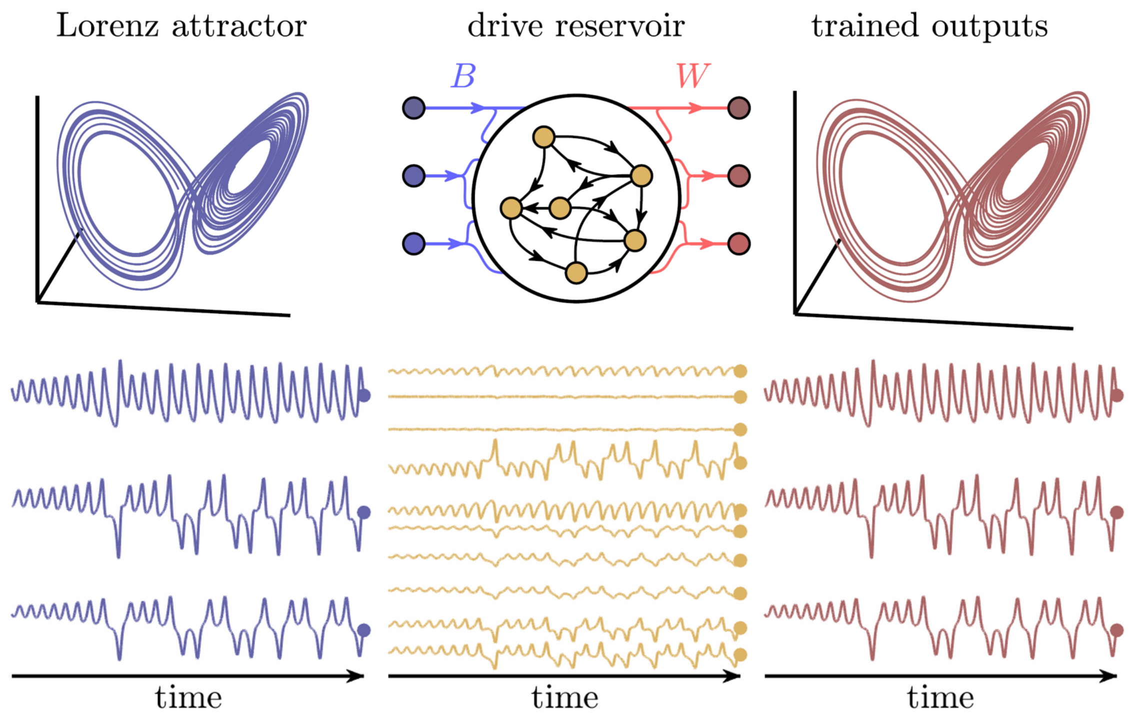


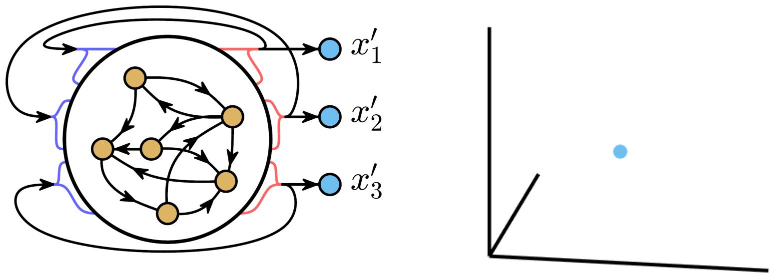


Sussillo, D., & Abbott, L. F. (2009). Generating Coherent Patterns of Activity from Chaotic Neural Networks. *Neuron*, *63*(4), 544–557.
Jaeger, H. (2010). The “ echo state ” approach to analysing and training recurrent neural networks – with an Erratum note. *GMD Report*, *1*(148), 1–47.

RNNs sustain chaotic memories by imitating examples

Driving: $\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}(A\boldsymbol{r} + B\boldsymbol{x} + \boldsymbol{d})$ → Training: W → Prediction: $\dot{\boldsymbol{r}} = -\boldsymbol{r} + \boldsymbol{g}([A + BW]\boldsymbol{r} + \boldsymbol{d})$

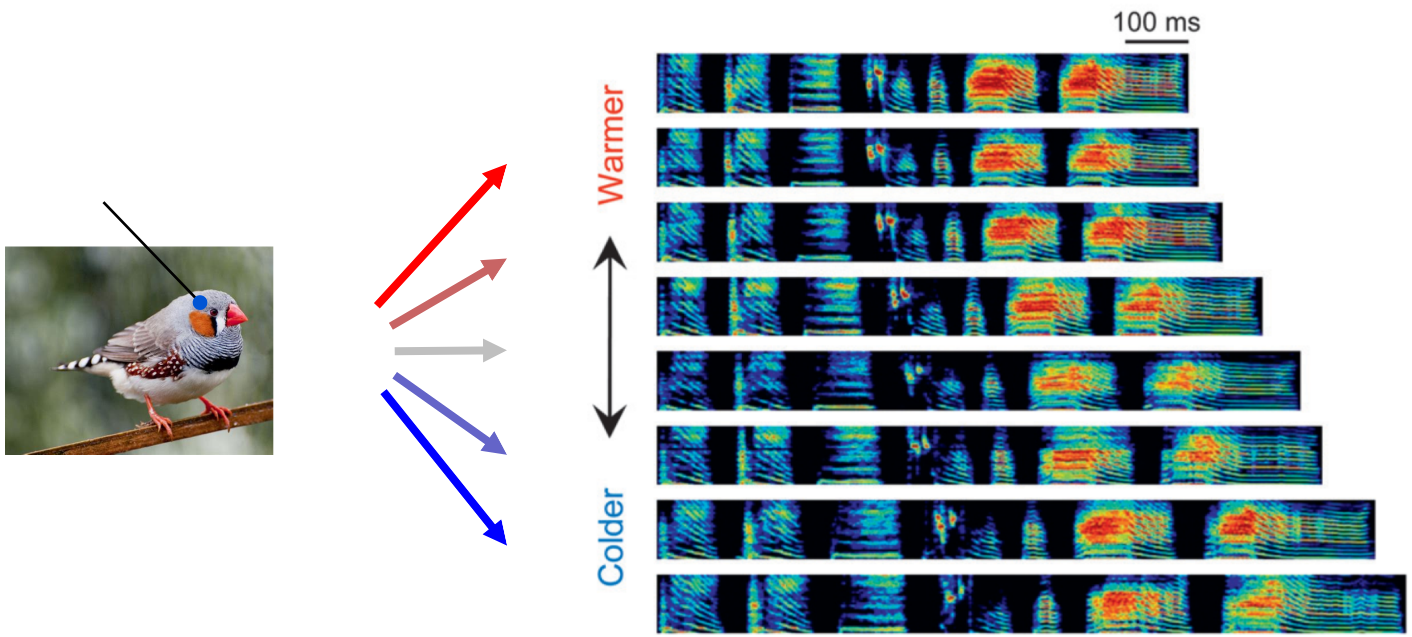




No engineered encoding or decoding!

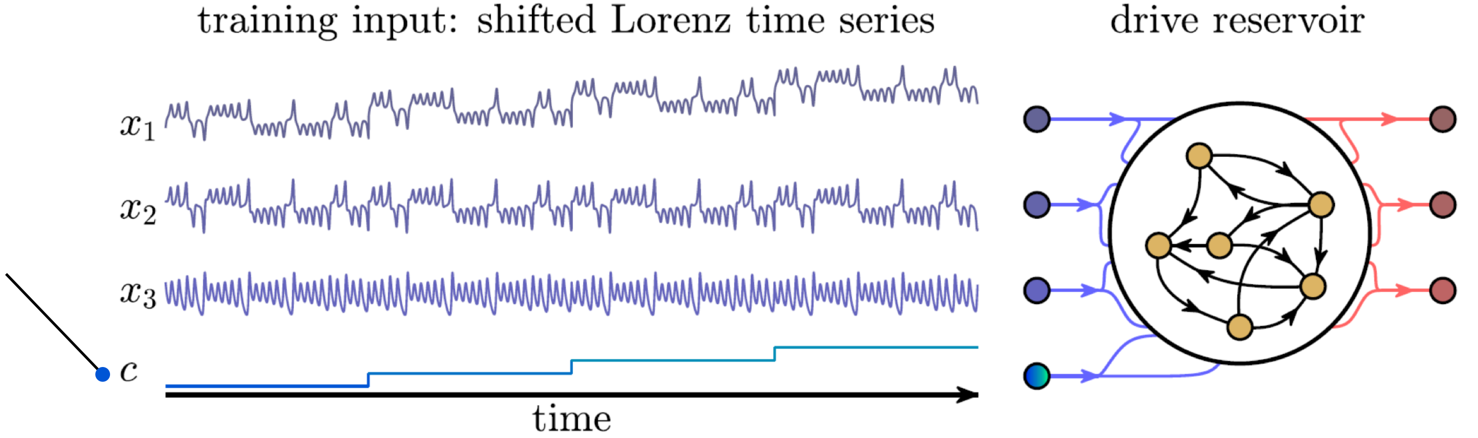
Sussillo, D., & Abbott, L. F. (2009). Generating Coherent Patterns of Activity from Chaotic Neural Networks. *Neuron*, *63*(4), 544–557.
Jaeger, H. (2010). The “ echo state ” approach to analysing and training recurrent neural networks – with an Erratum note. *GMD Report*, *1*(148), 1–47.

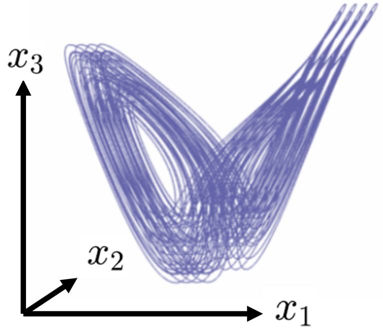
Learning computations by imitating examples



Penn
UNIVERSITY of PENNSYLVANIA

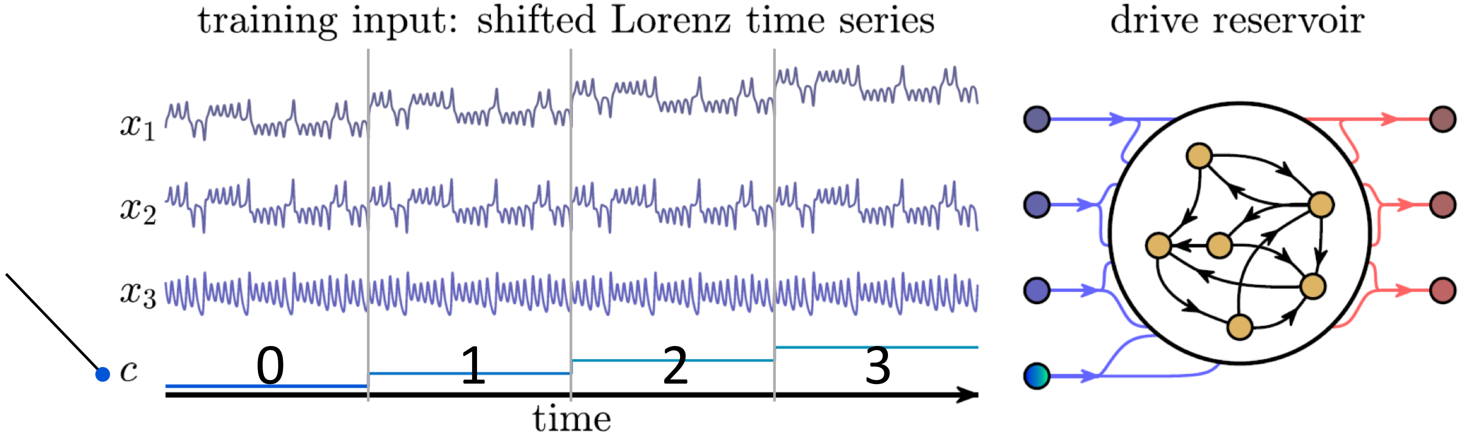
RNNs translate chaotic memories by imitating examples

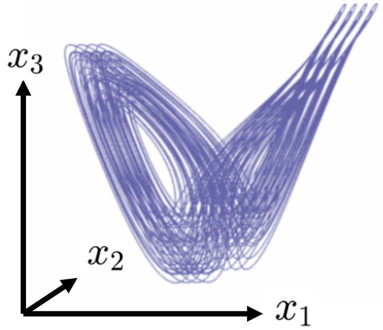




Kim, J. Z., Lu, Z., Nozari, E., Pappas, G. J., & Bassett, D. S. (2020). Teaching Recurrent Neural Networks to Modify Chaotic Memories by Example. (Accepted, *Nat. Mach. Intell.*).

RNNs translate chaotic memories by imitating examples

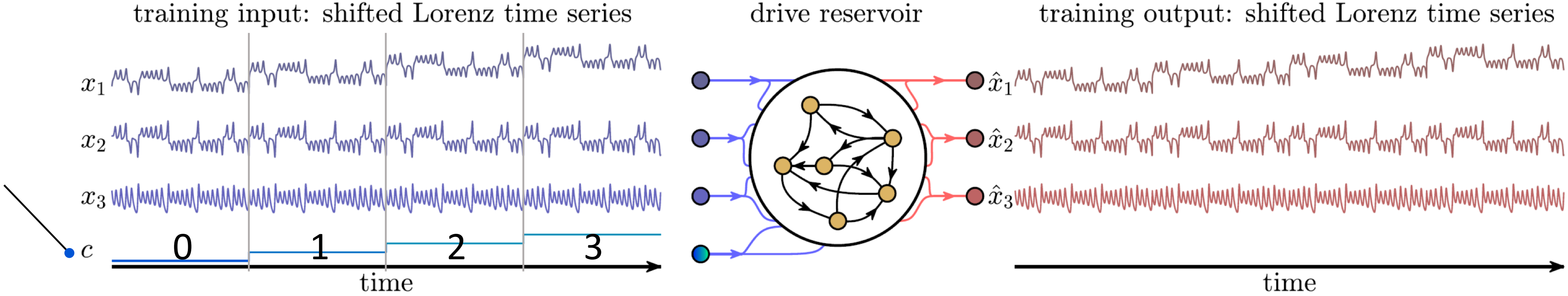


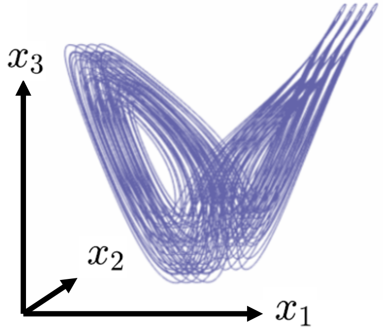


Kim, J. Z., Lu, Z., Nozari, E., Pappas, G. J., & Bassett, D. S. (2020). Teaching Recurrent Neural Networks to Modify Chaotic Memories by Example. (Accepted, *Nat. Mach. Intell.*).

UNIVERSITY of PENNSYLVANIA

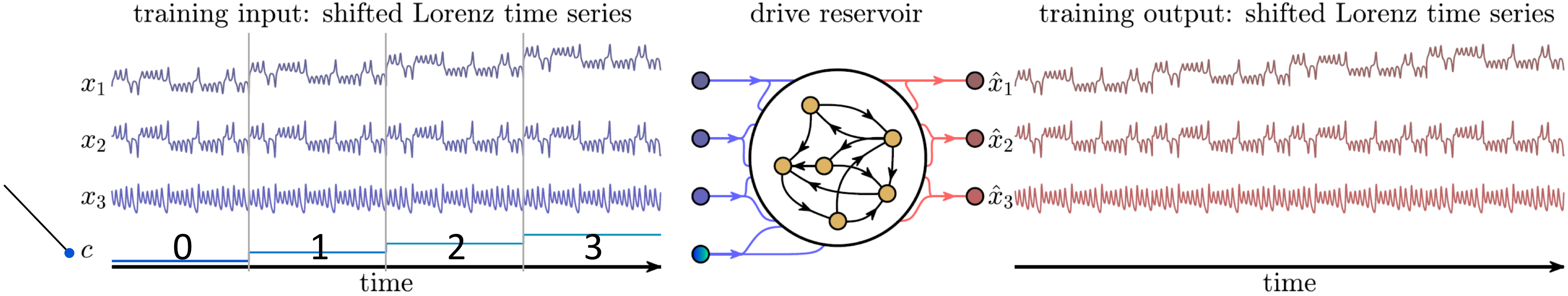
RNNs translate chaotic memories by imitating examples

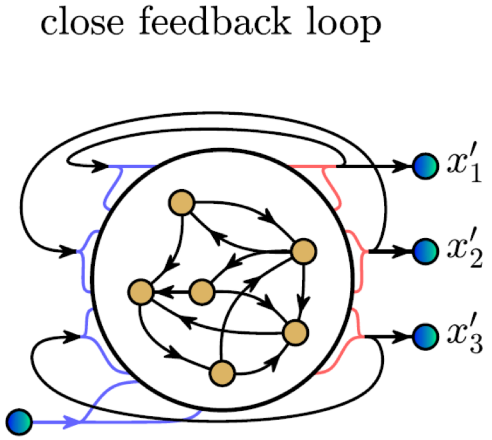


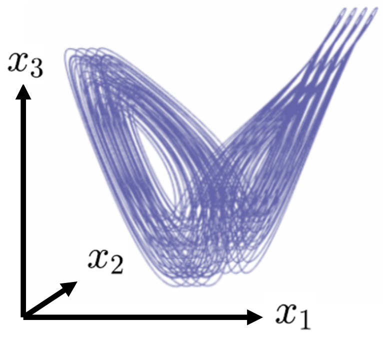


Kim, J. Z., Lu, Z., Nozari, E., Pappas, G. J., & Bassett, D. S. (2020). Teaching Recurrent Neural Networks to Modify Chaotic Memories by Example. (Accepted, *Nat. Mach. Intell.*).

RNNs translate chaotic memories by imitating examples

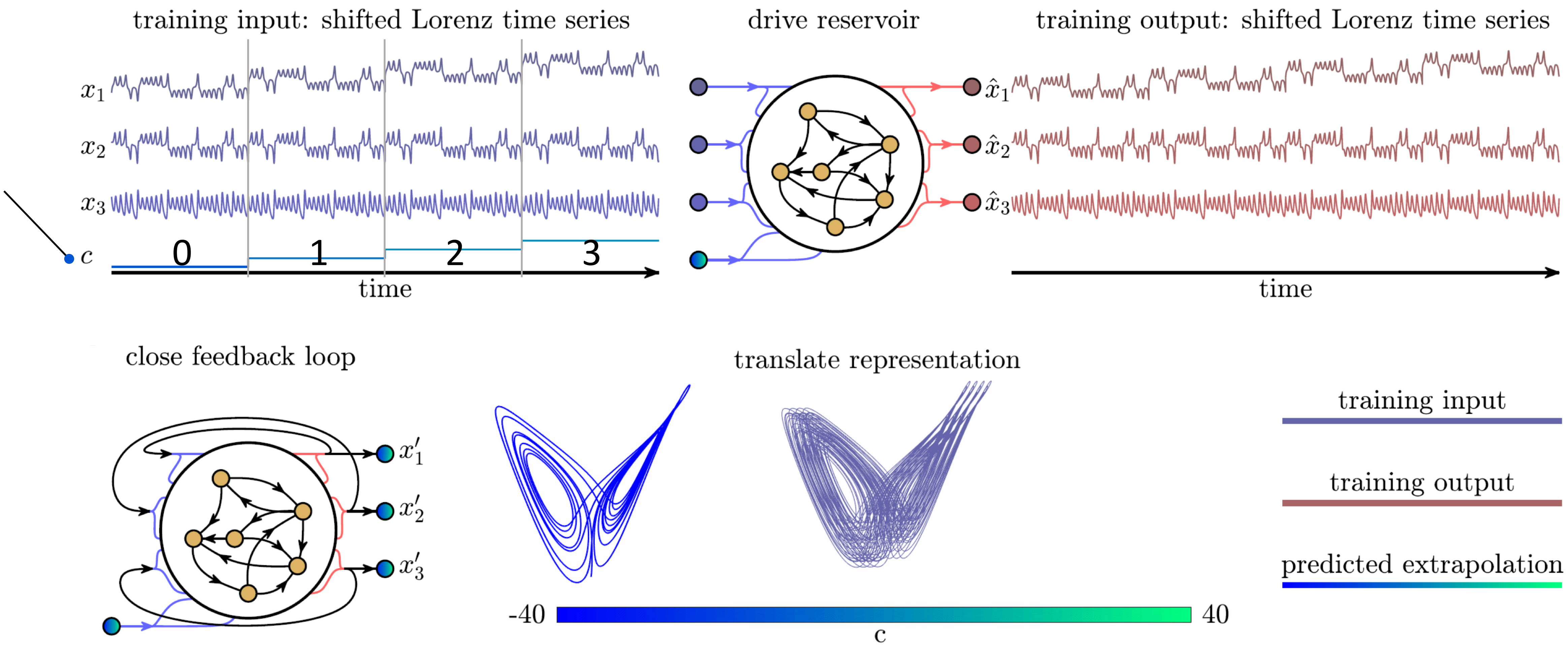






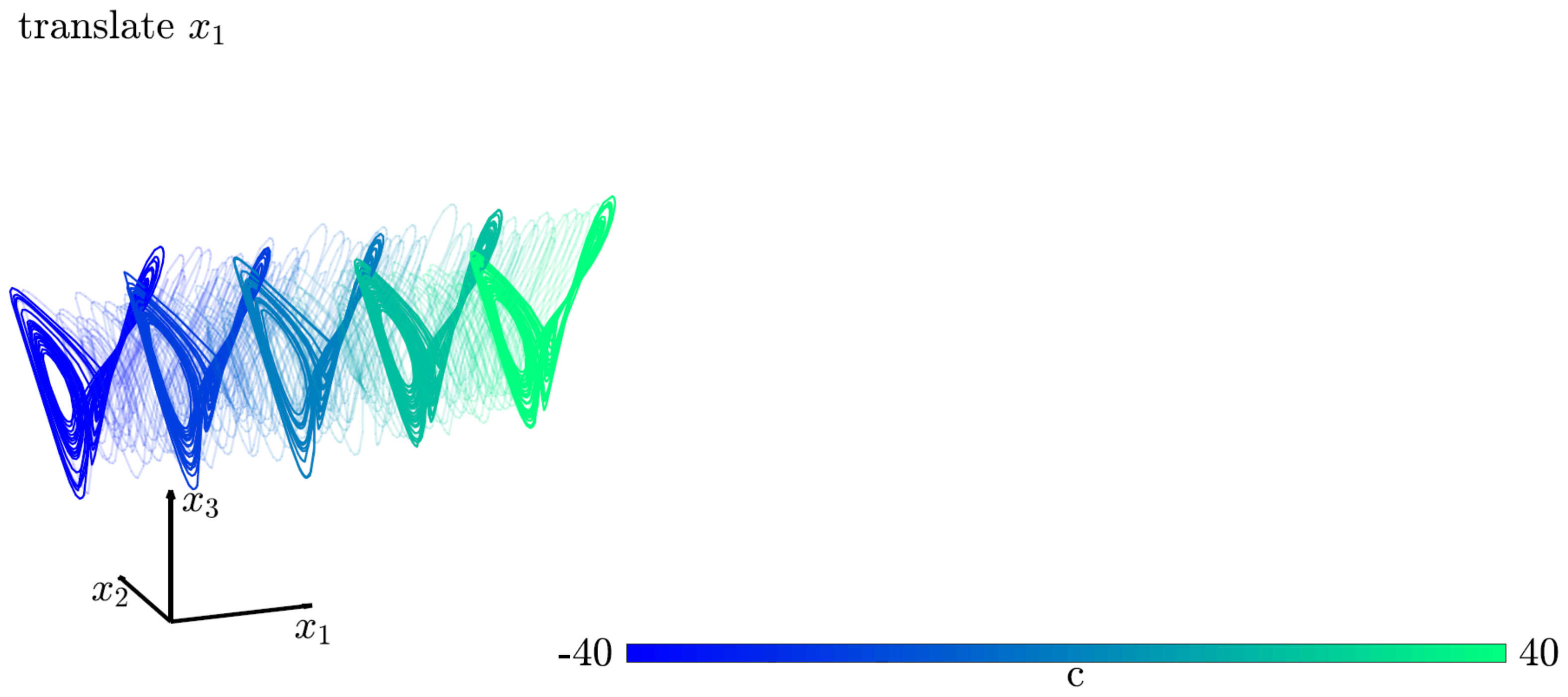
Kim, J. Z., Lu, Z., Nozari, E., Pappas, G. J., & Bassett, D. S. (2020). Teaching Recurrent Neural Networks to Modify Chaotic Memories by Example. (Accepted, *Nat. Mach. Intell.*).

RNNs translate chaotic memories by imitating examples

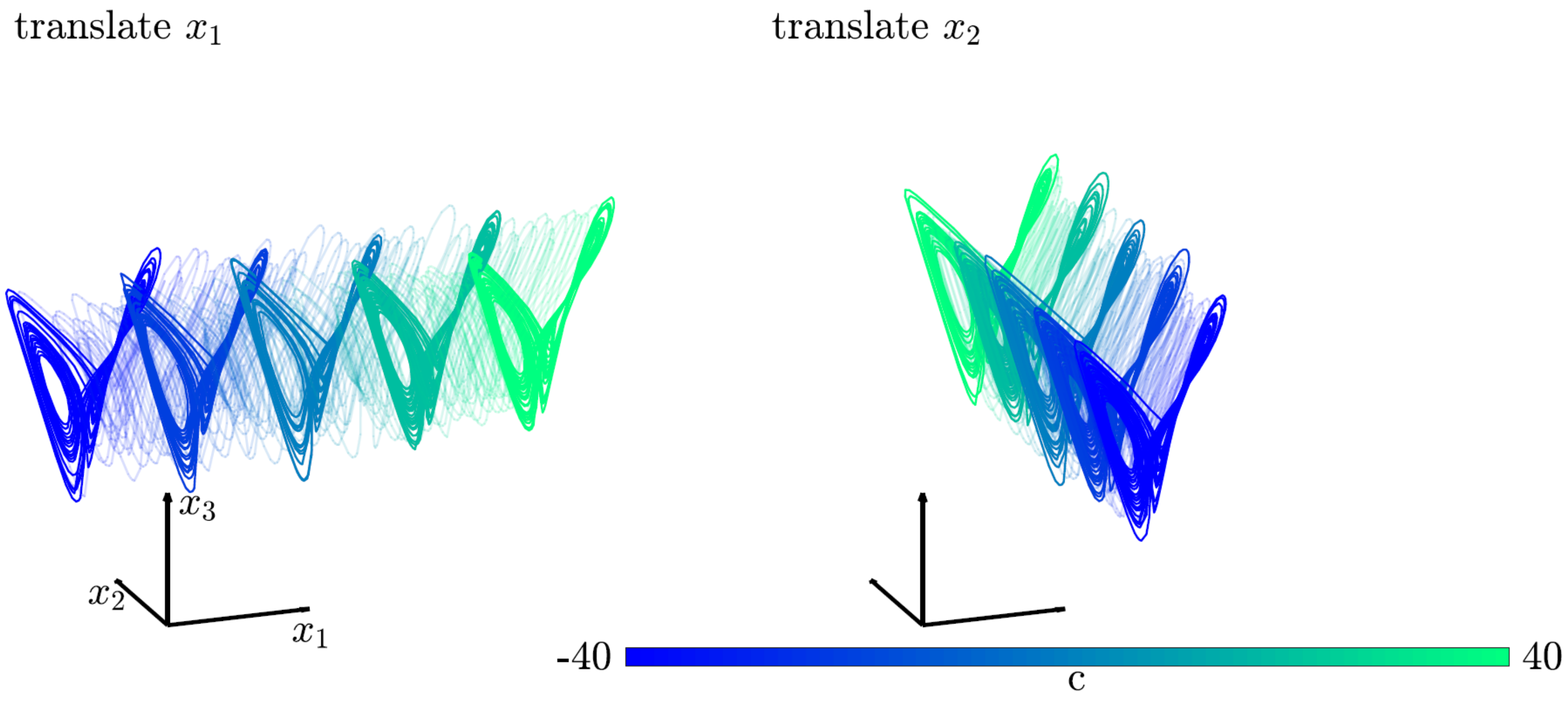


Kim, J. Z., Lu, Z., Nozari, E., Pappas, G. J., & Bassett, D. S. (2020). Teaching Recurrent Neural Networks to Modify Chaotic Memories by Example. (Accepted, *Nat. Mach. Intell.*).

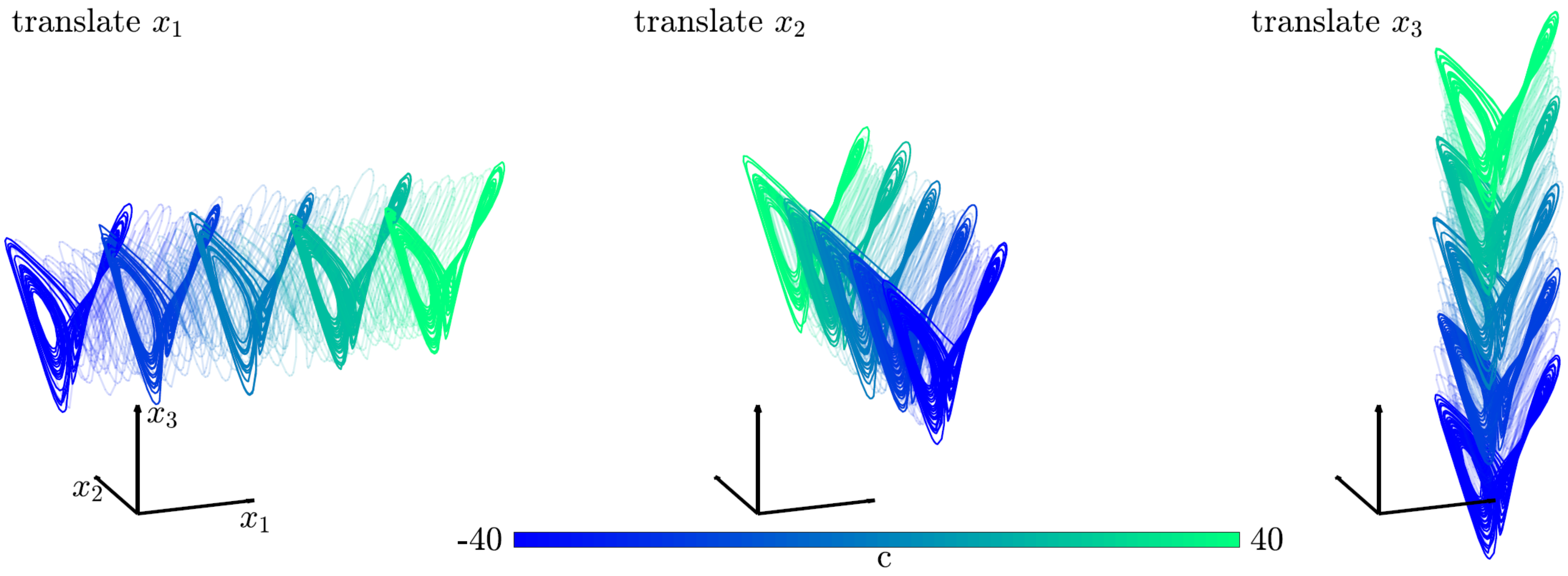
RNNs translate chaotic memories by imitating examples



RNNs translate chaotic memories by imitating examples



RNNs translate chaotic memories by imitating examples



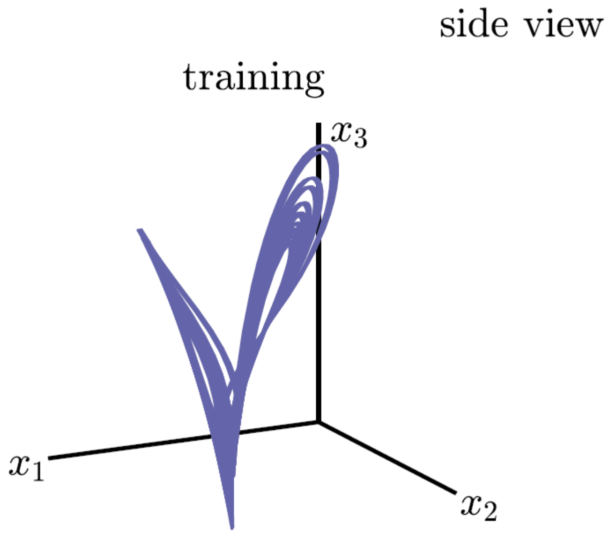
RNNs transform chaotic memories by imitating examples

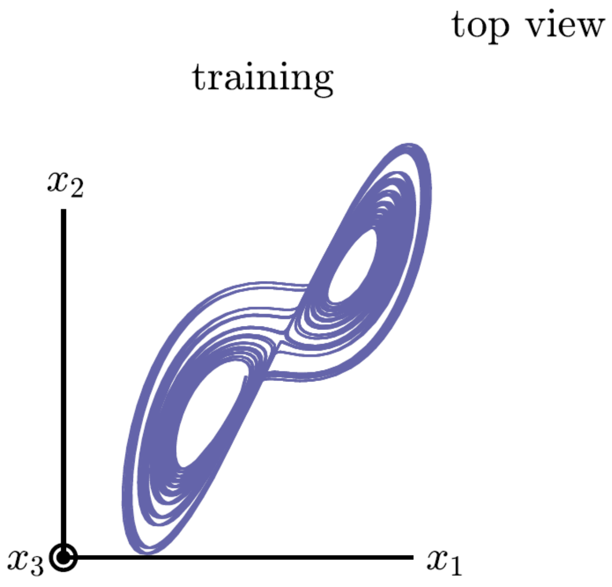
RNNs transform chaotic memories by imitating examples

- Can we change the actual geometry of the attractor manifold?

RNNs transform chaotic memories by imitating examples

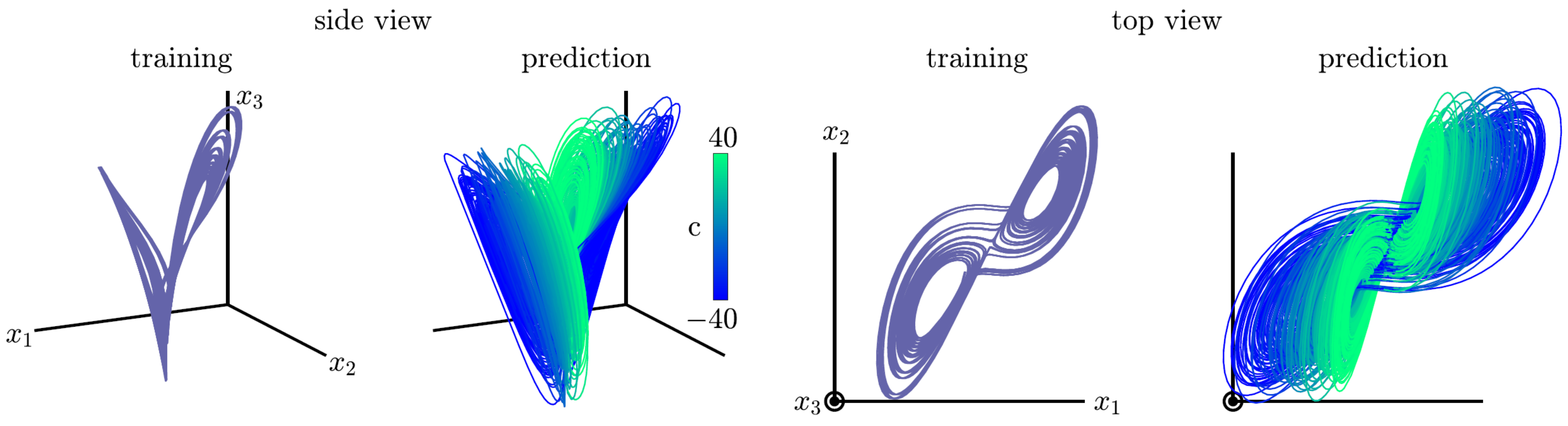
- Can we change the actual geometry of the attractor manifold?



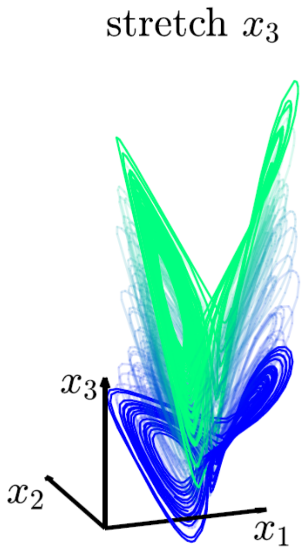


RNNs transform chaotic memories by imitating examples

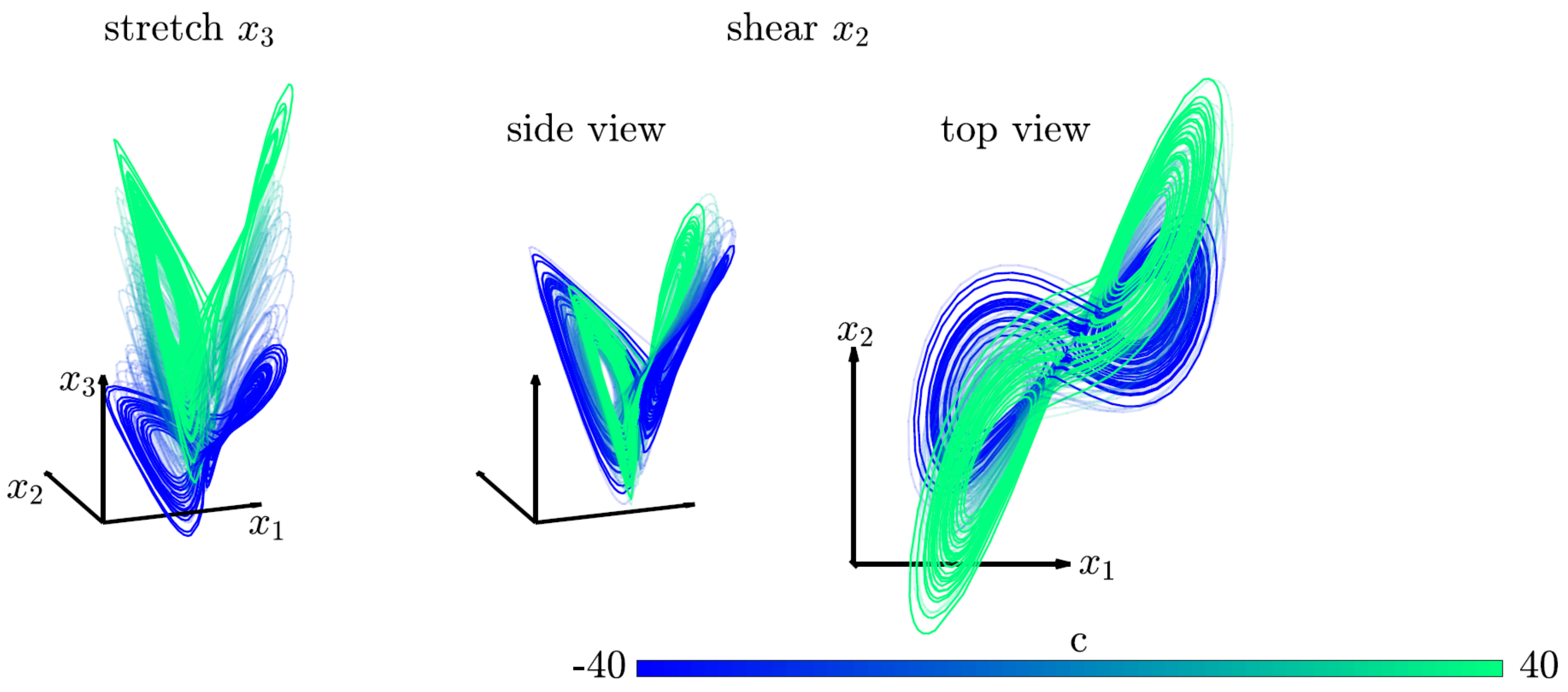
- Can we change the actual geometry of the attractor manifold?



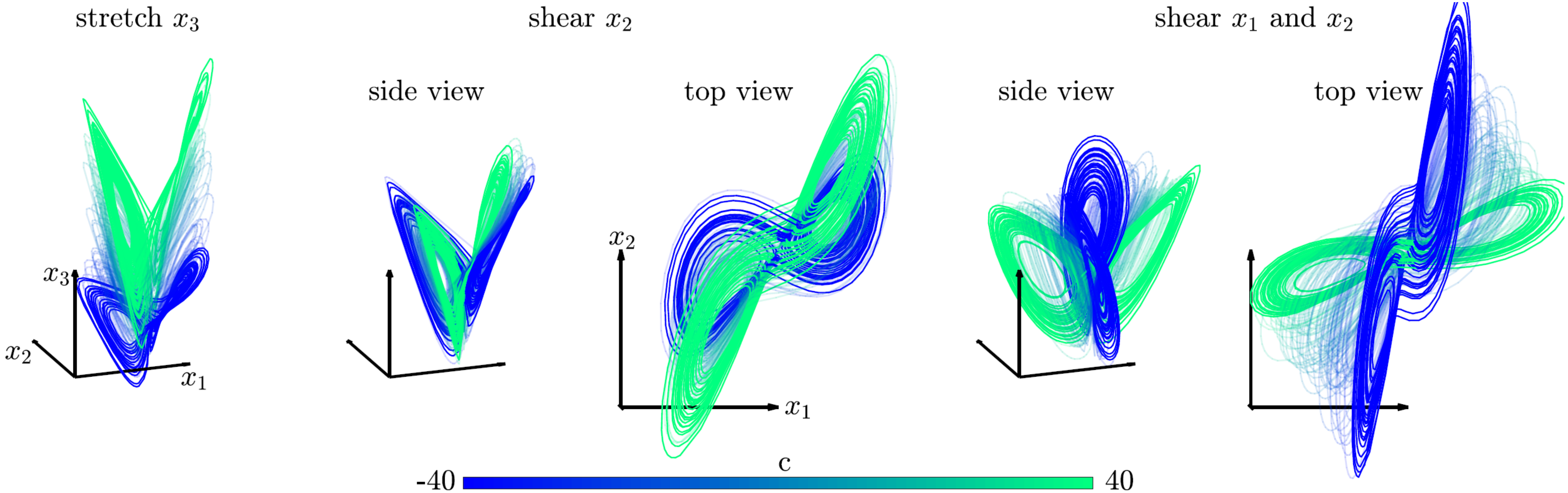
RNNs transform chaotic memories by imitating examples



RNNs transform chaotic memories by imitating examples



RNNs transform chaotic memories by imitating examples



University of Pennsylvania

RNNs predict highly nonlinear events by imitating examples

RNNs predict highly nonlinear events by imitating examples

- The Lorenz attractor undergoes a subcritical Hopf bifurcation
  - Fixed points at the wings lose stability

Lorenz system:

$$\dot{x}_1 = \sigma(x_2 - x_1)$$

$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$

$$\dot{x}_3 = x_1 x_2 - \beta x_3,$$

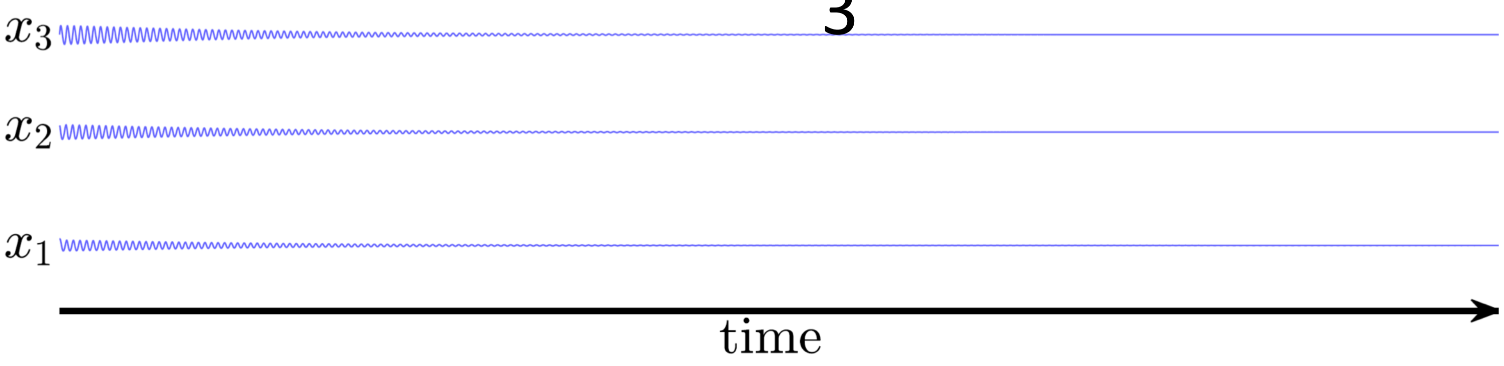
RNNs predict highly nonlinear events by imitating examples

- The Lorenz attractor undergoes a subcritical Hopf bifurcation
  - Fixed points at the wings lose stability

Lorenz system:

$$\dot{x}_1 = \sigma(x_2 - x_1)$$
$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$
$$\dot{x}_3 = x_1 x_2 - \beta x_3,$$

$\rho = 23$



RNNs predict highly nonlinear events by imitating examples

- The Lorenz attractor undergoes a subcritical Hopf bifurcation
  - Fixed points at the wings lose stability

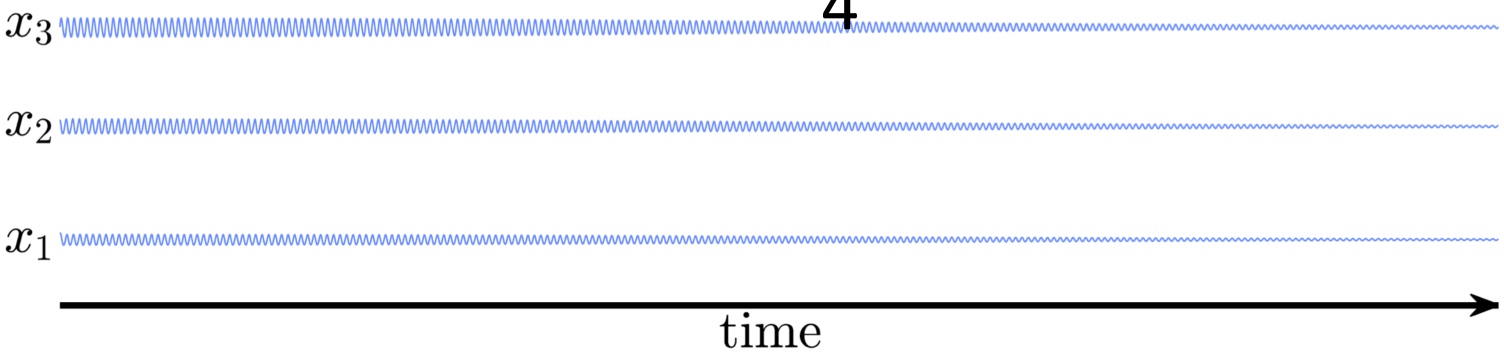
Lorenz system:

$$\dot{x}_1 = \sigma(x_2 - x_1)$$

$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$

$$\dot{x}_3 = x_1 x_2 - \beta x_3,$$

$\rho = 24$



RNNs predict highly nonlinear events by imitating examples

- The Lorenz attractor undergoes a subcritical Hopf bifurcation
    - Fixed points at the wings lose stability

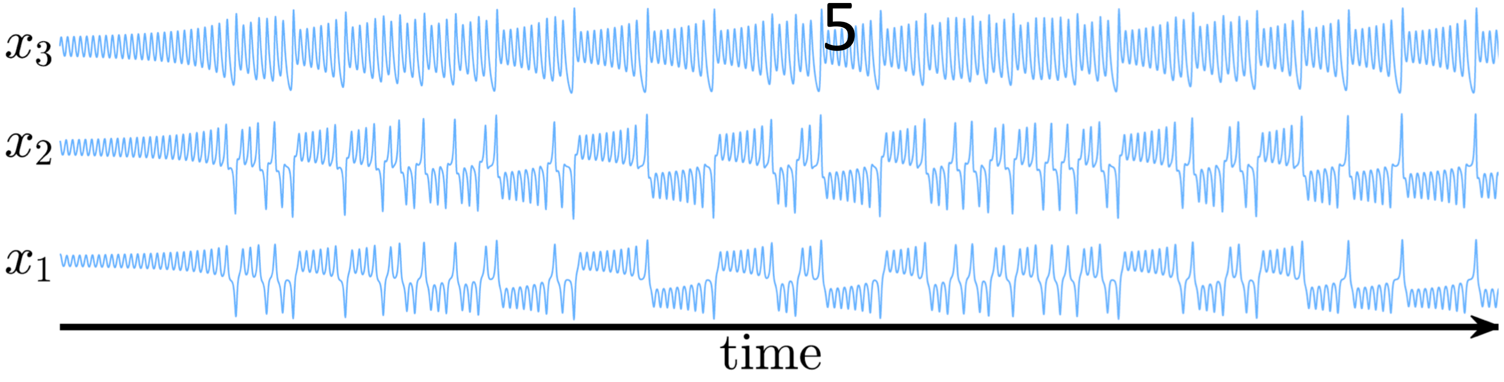
Lorenz system:

$$\dot{x}_1 = \sigma(x_2 - x_1)$$

$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$

$$\dot{x}_3 = x_1 x_2 - \beta x_3,$$

$\rho = 25$

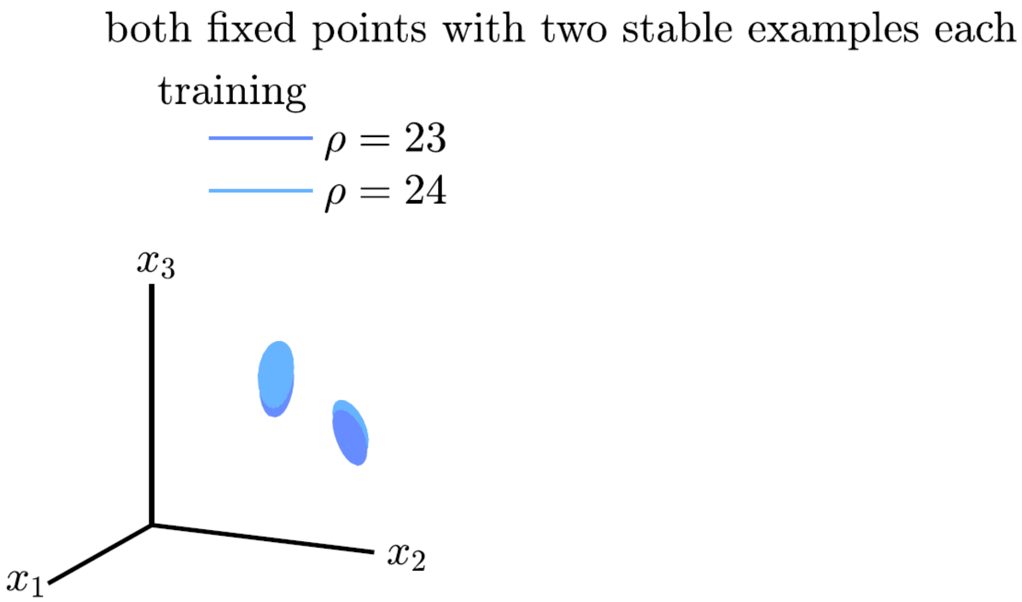


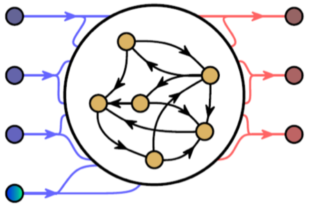
RNNs predict highly nonlinear events by imitating examples

- How much of a bifurcation can an RNN infer?

RNNs predict highly nonlinear events by imitating examples

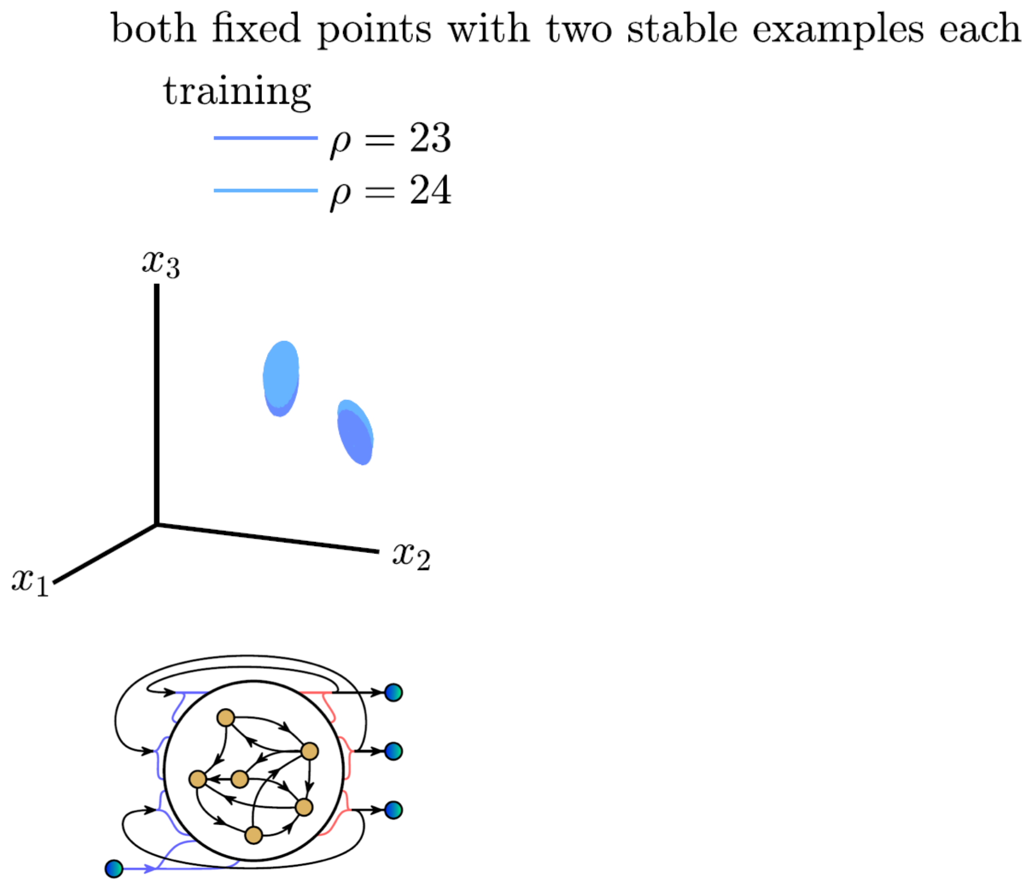
- How much of a bifurcation can an RNN infer?





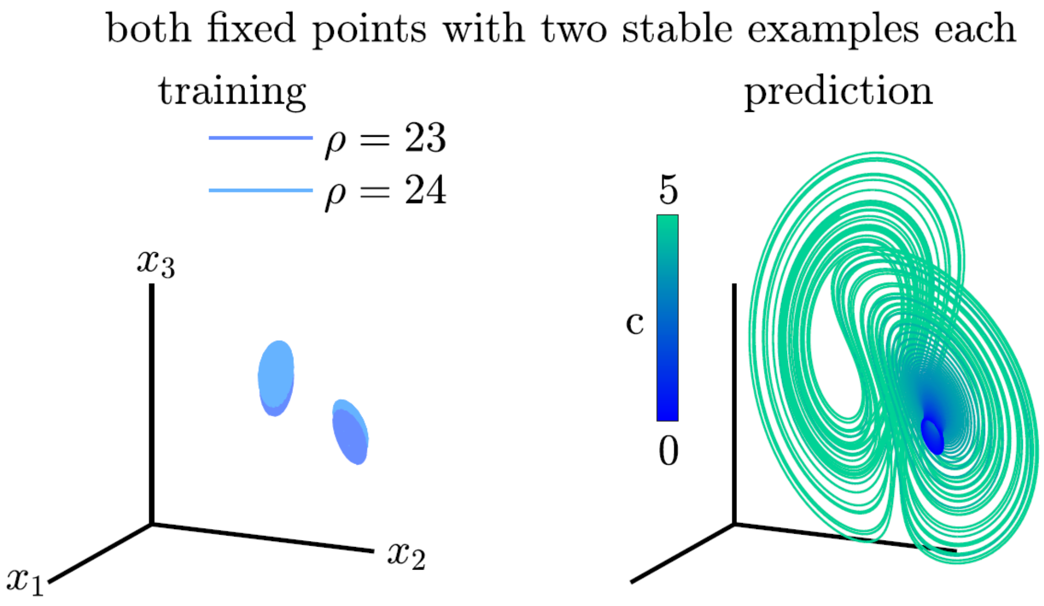
RNNs predict highly nonlinear events by imitating examples

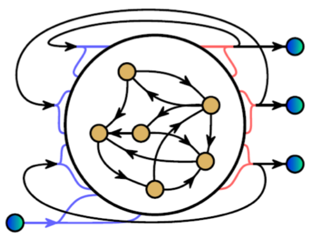
- How much of a bifurcation can an RNN infer?



RNNs predict highly nonlinear events by imitating examples

- How much of a bifurcation can an RNN infer?

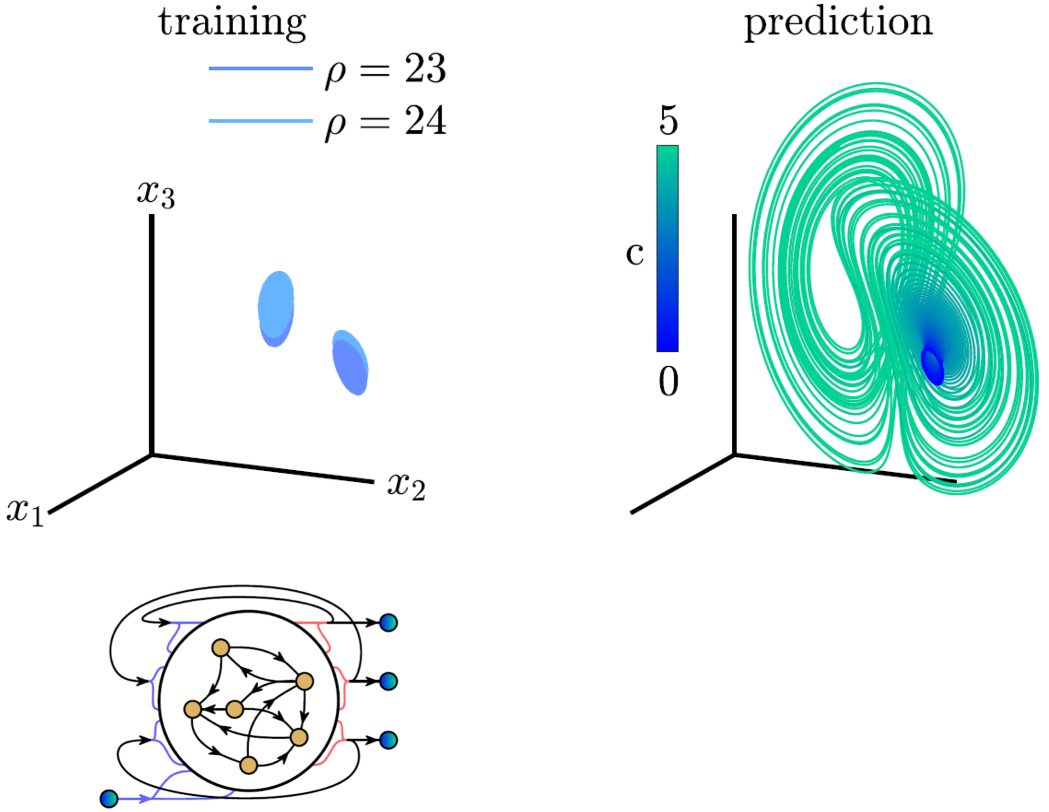




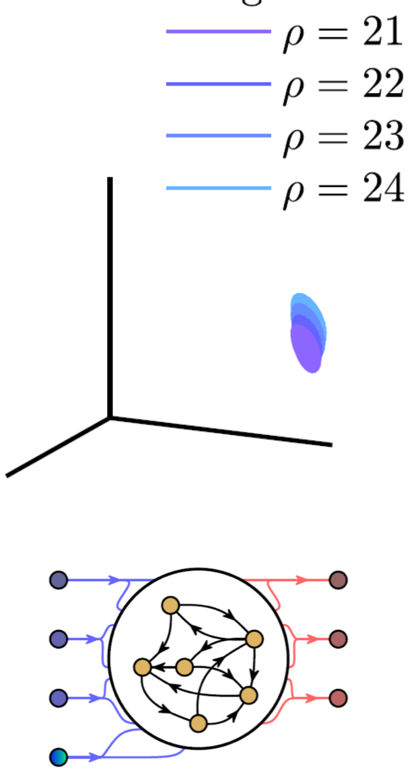
RNNs predict highly nonlinear events by imitating examples

- How much of a bifurcation can an RNN infer?

both fixed points with two stable examples each



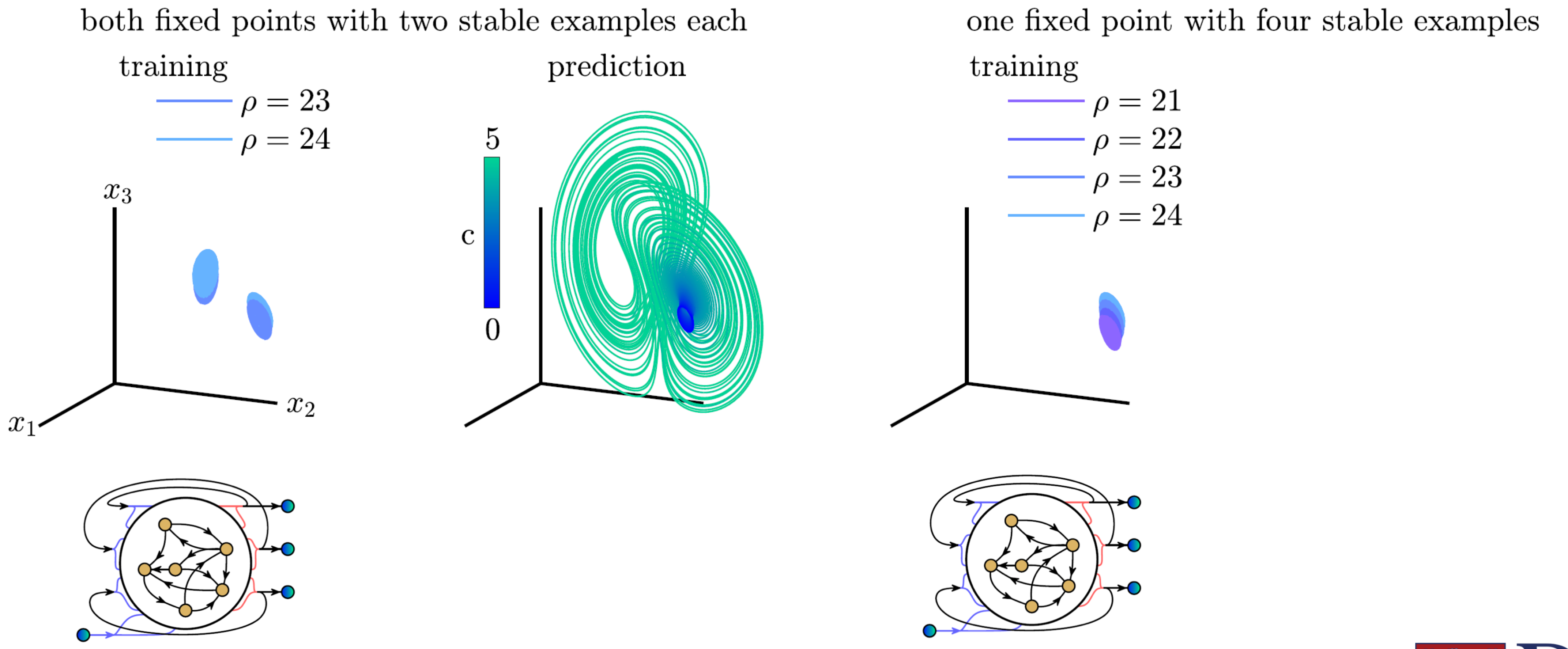
one fixed point with four stable examples



Penn
University *of* Pennsylvania

RNNs predict highly nonlinear events by imitating examples

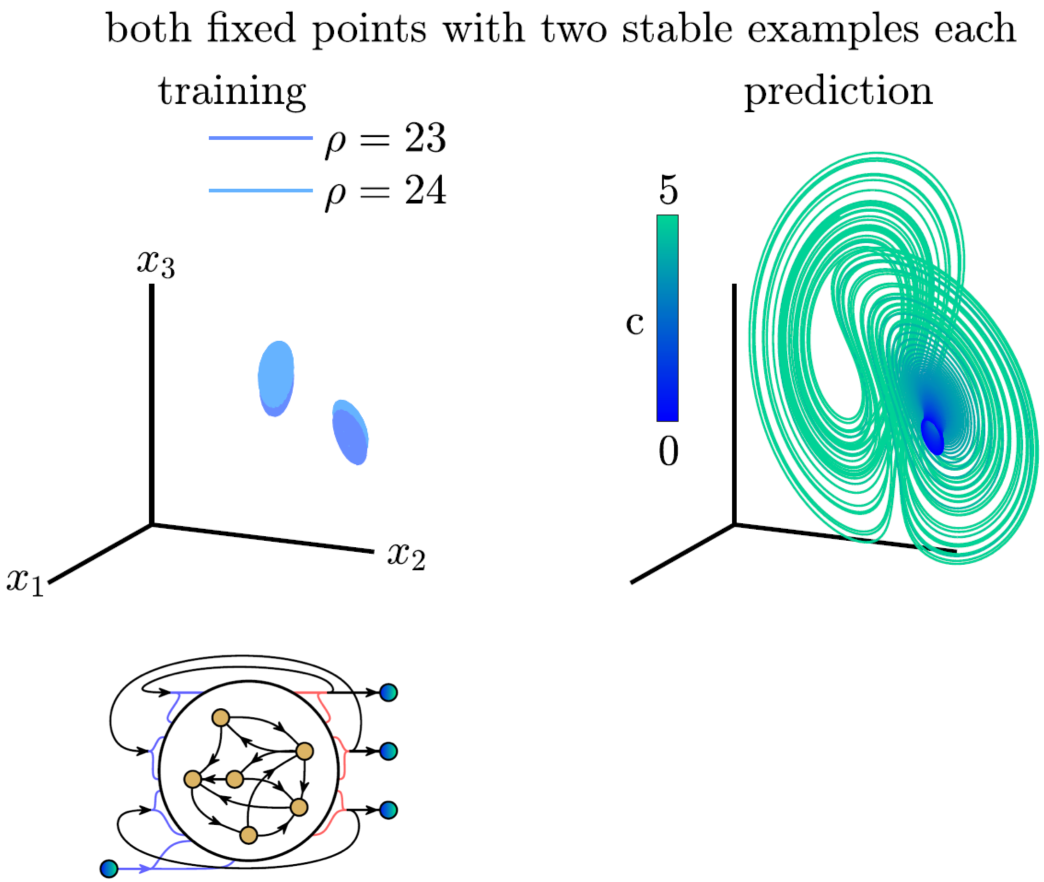
- How much of a bifurcation can an RNN infer?

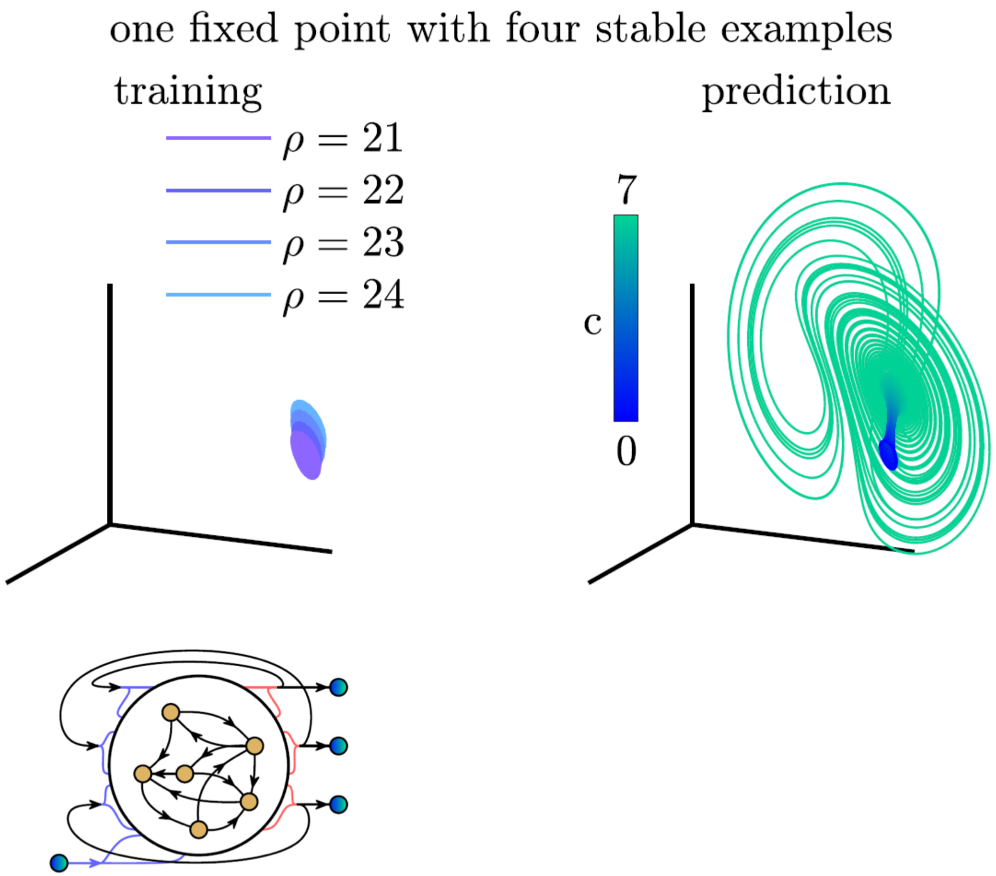


University of Pennsylvania

RNNs predict highly nonlinear events by imitating examples

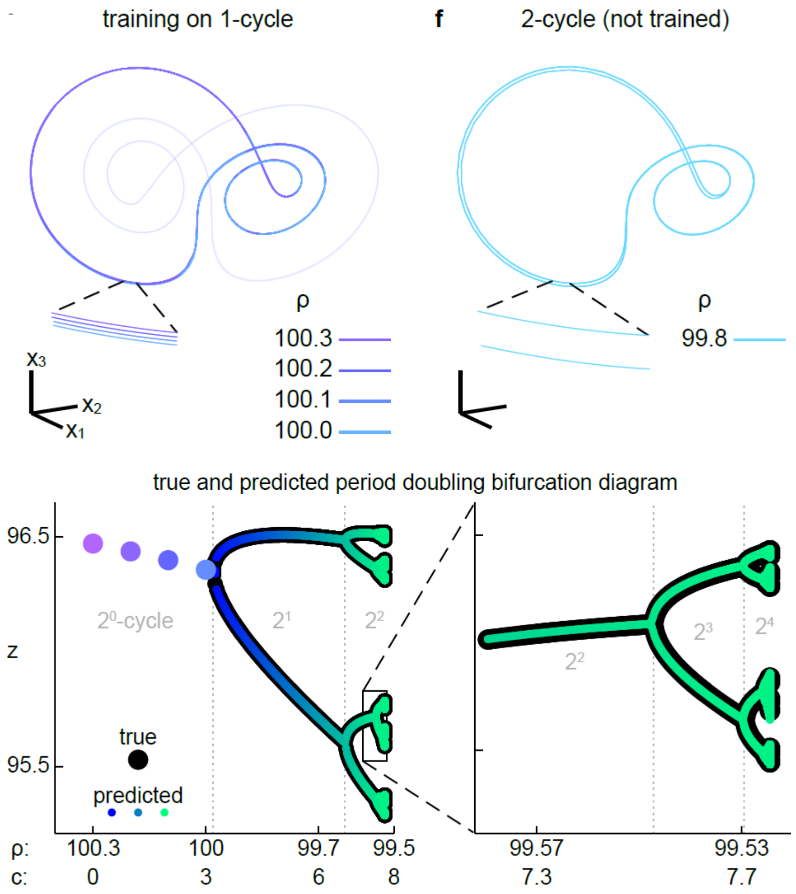
- How much of a bifurcation can an RNN infer?





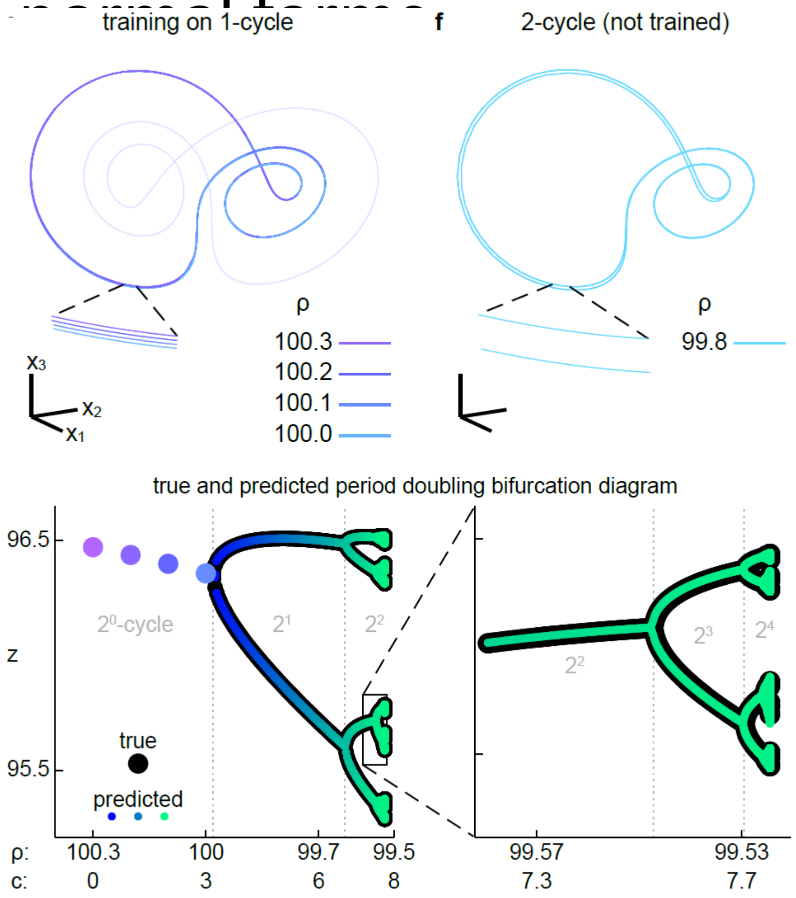
RNNs predict highly nonlinear events by imitating examples

Period doubling bifurcation

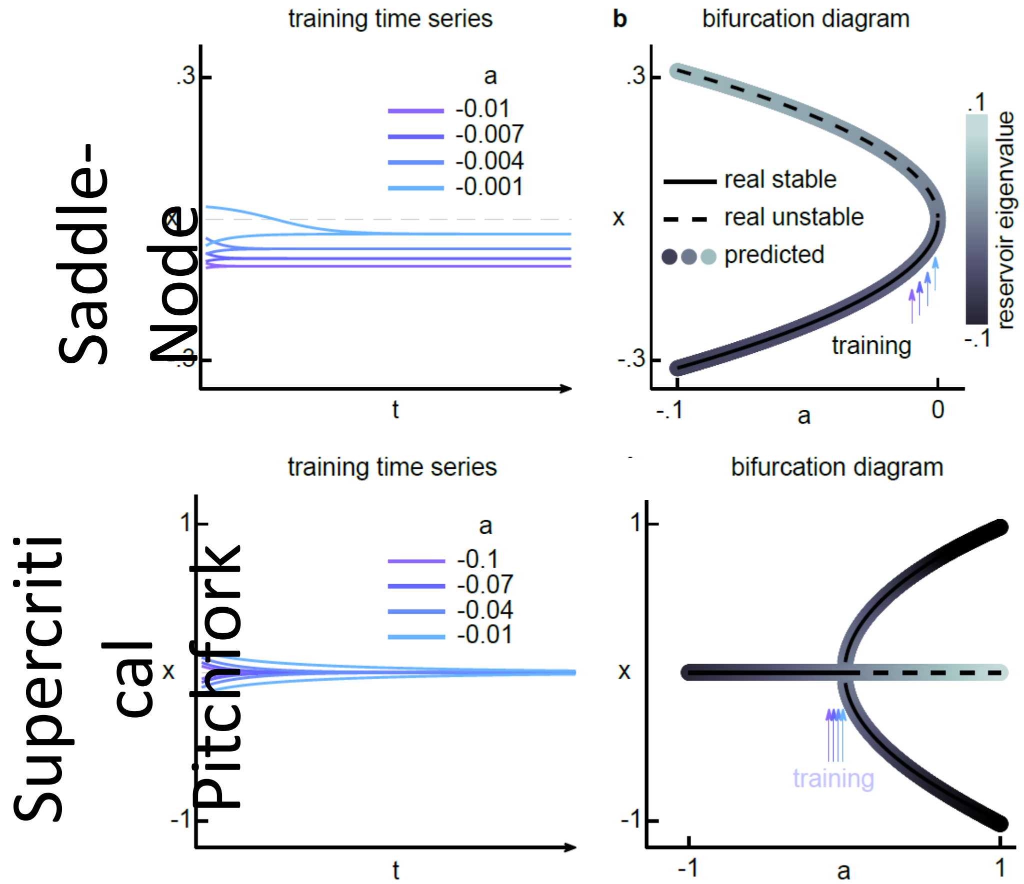


RNNs predict highly nonlinear events by imitating examples

Period doubling bifurcation

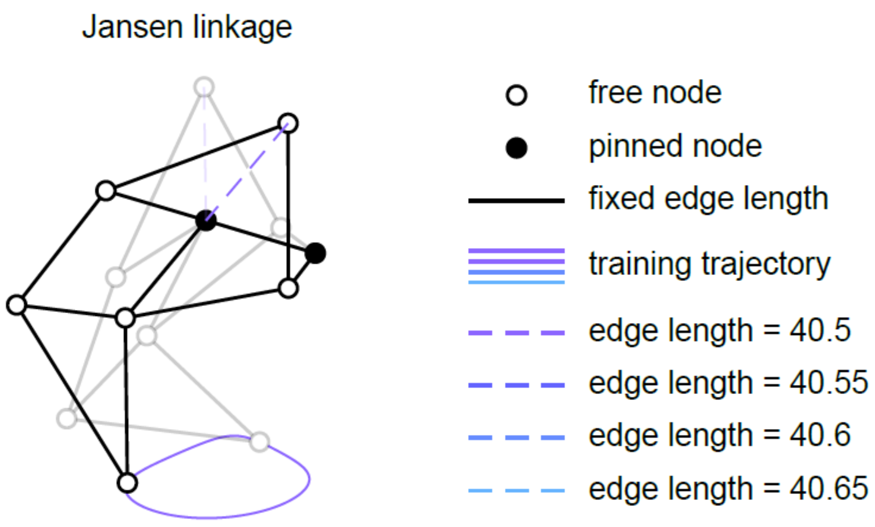


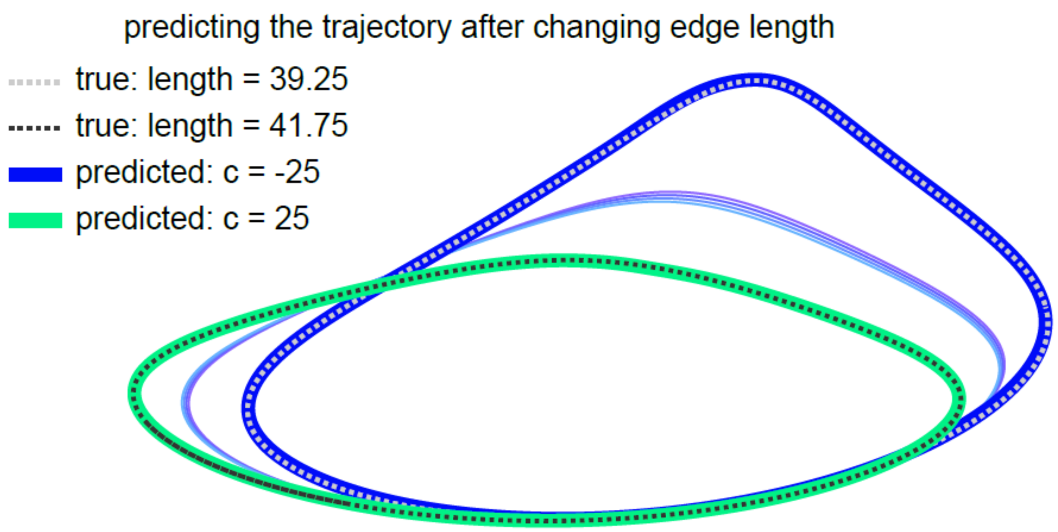
Bifurcation



RNNs predict highly nonlinear events by imitating examples

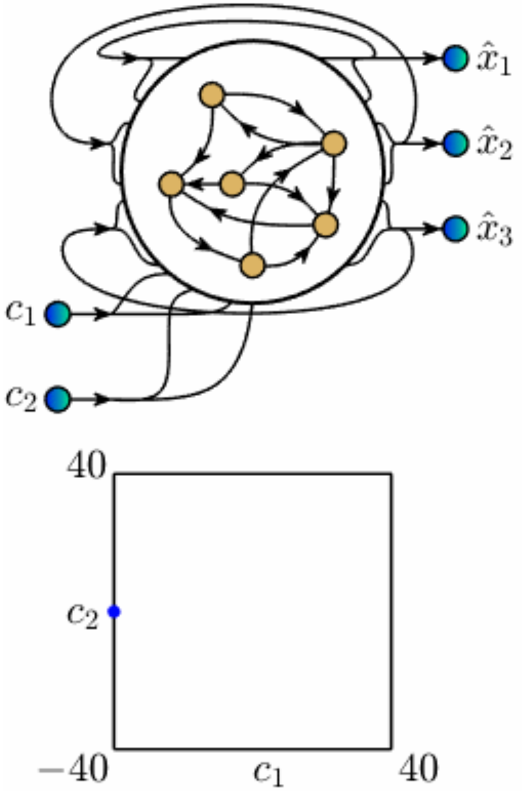
Kinematic trajectories

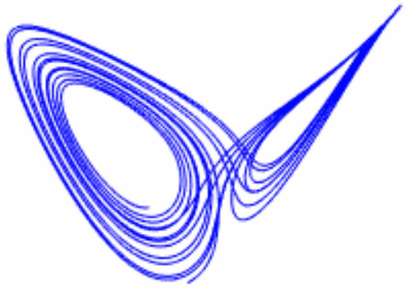




Flight of the Lorenz

- Translation in x1 and x3





How do RNNs learn translations and transformations?

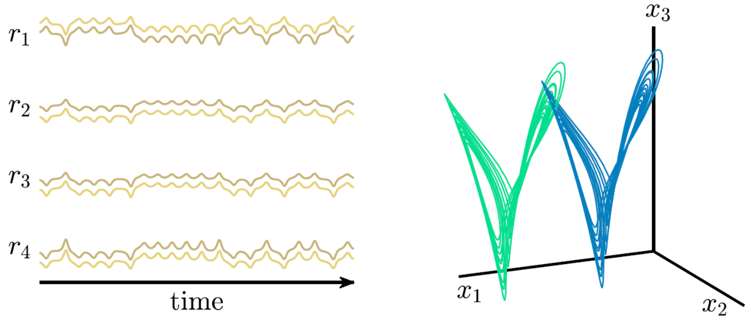
Translation: discrete

$$W\boldsymbol{r}_c(t) \approx \boldsymbol{x}(t) + \begin{bmatrix}1\\0\\0\end{bmatrix} c$$

infinitesimal

$$W\mathrm{d}\boldsymbol{r}(t) \approx \begin{bmatrix}1\\0\\0\end{bmatrix} \mathrm{d}c$$

$$\mathrm{d}\boldsymbol{r}(t) \approx f(\boldsymbol{r}(t), W)\mathrm{d}c$$



Penn
UNIVERSITY of PENNSYLVANIA

How do RNNs learn translations and transformations?

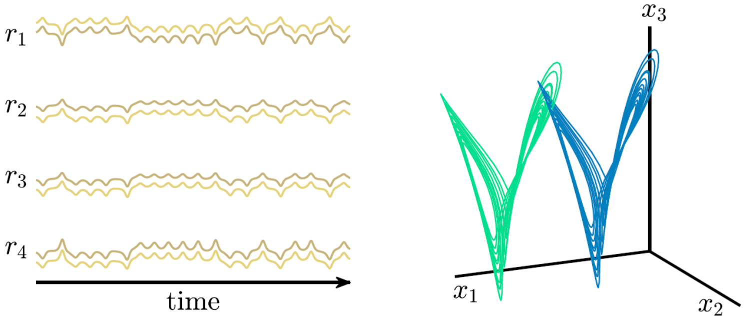
Translation: discrete

$$W\boldsymbol{r}_c(t) \approx \boldsymbol{x}(t) + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} c$$

infinitesimal

$$W\mathrm{d}\boldsymbol{r}(t) \approx \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \mathrm{d}c$$

$$\mathrm{d}\boldsymbol{r}(t) \approx f(\boldsymbol{r}(t), W)\mathrm{d}c$$



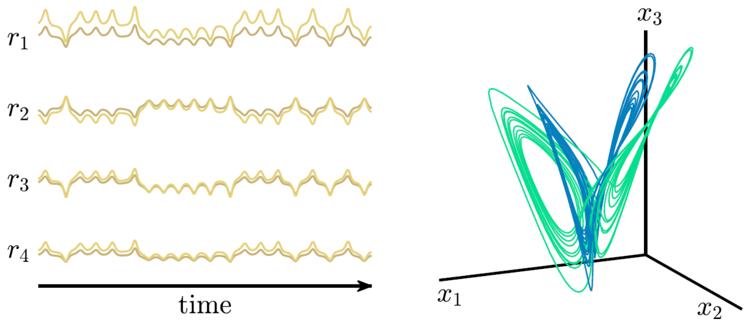
Transformation: discrete

$$W\boldsymbol{r}_c(t) \approx [I - Tc]\boldsymbol{x}(t)$$

infinitesimal

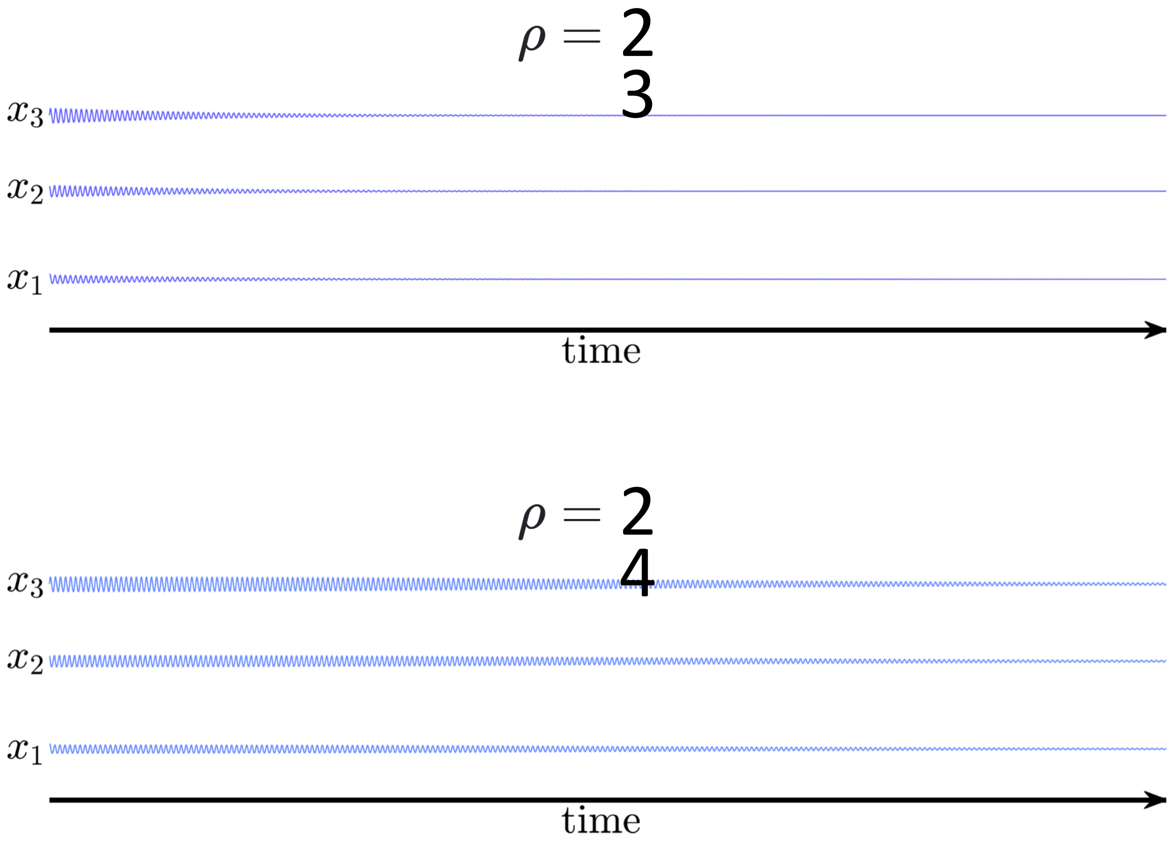
$$W\mathrm{d}\boldsymbol{r}(t) \approx -T\boldsymbol{x}(t)\mathrm{d}c$$

$$\mathrm{d}\boldsymbol{r}(t) \approx f(\boldsymbol{r}(t), W)\mathrm{d}c$$

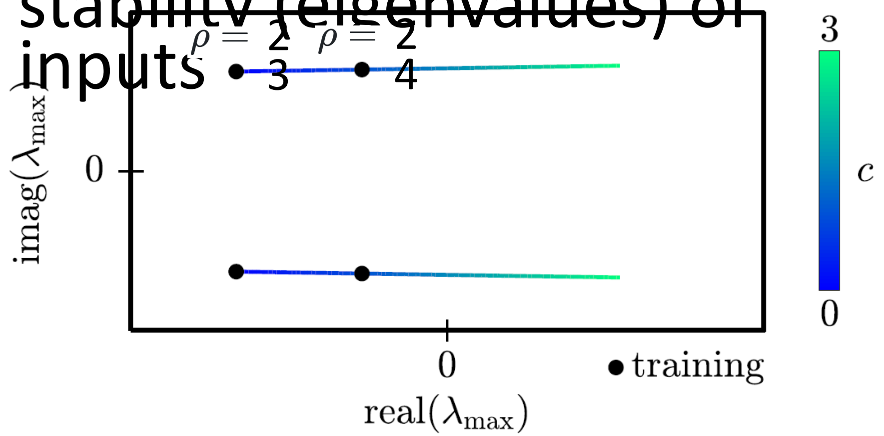


University of Pennsylvania

How do RNNs learn bifurcations?



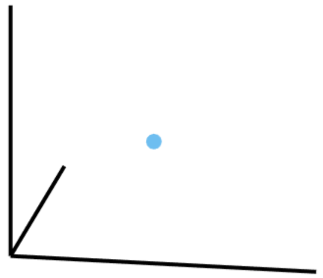
RNNs learn smooth transformations of the stability (eigenvalues) of inputs



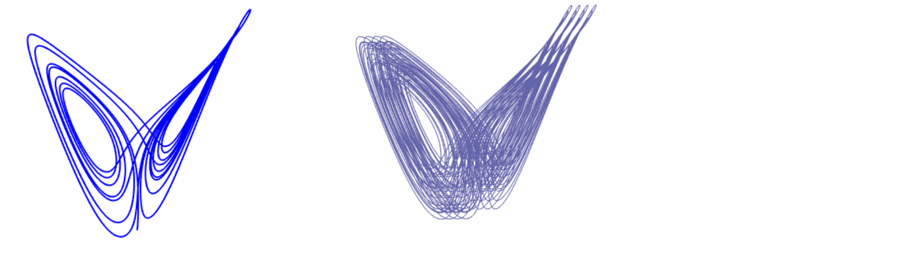
Slower rate of decay => less negative eigenvalue

Conclusions: simply by imitating inputs, reservoirs can

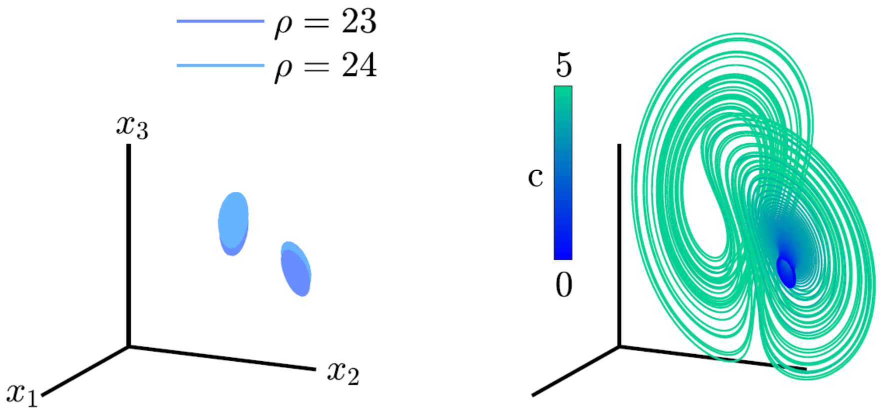
- Sustain complex temporal representations as n



- Translate and transform memo



- Infer global nonlinear structure



Penn
UNIVERSITY of PENNSYLVANIA

Thank you!

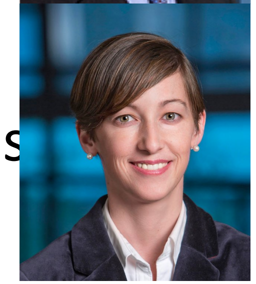
Collaborators

Zhixin Lu

Erfan Nozari

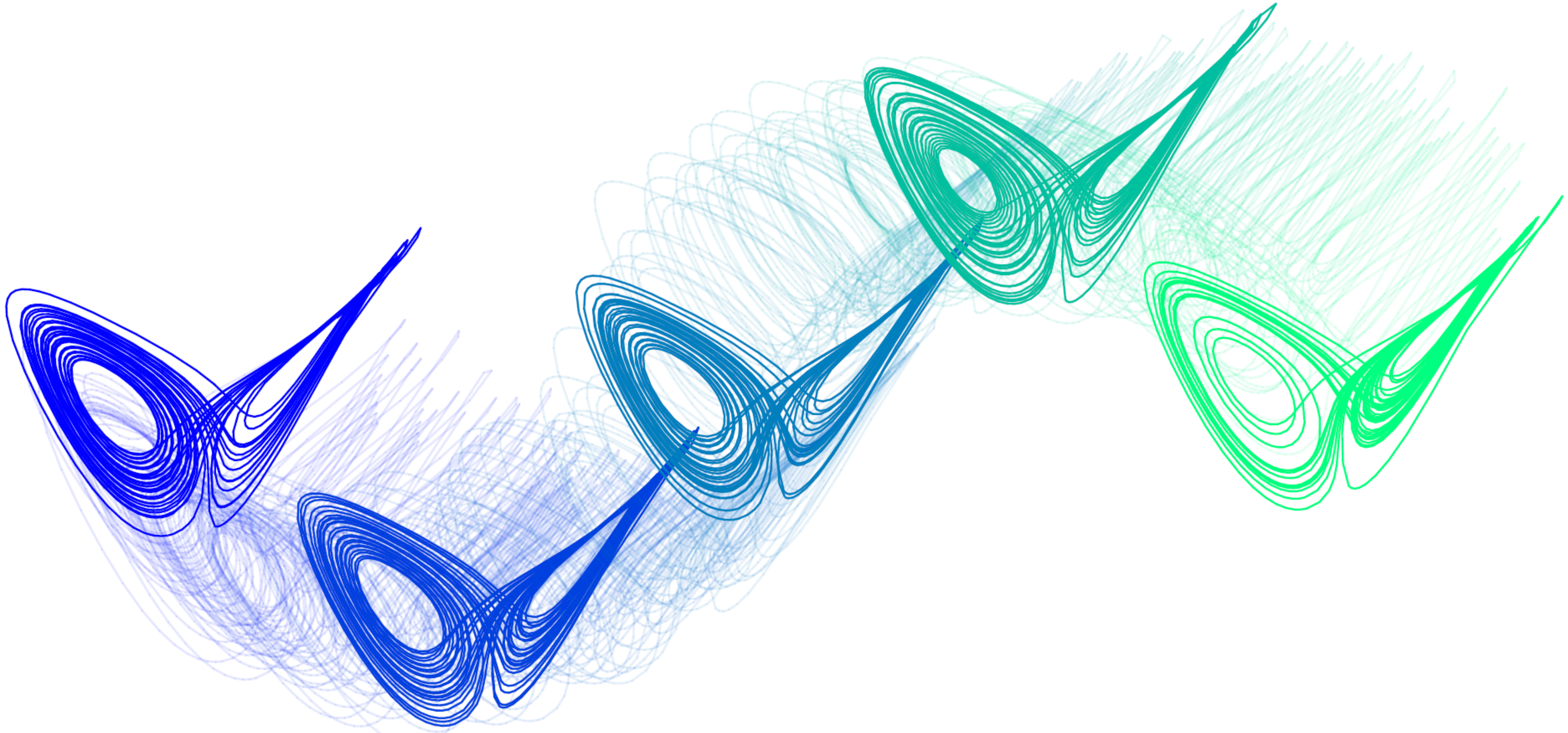
George Pappas

Danielle Bass



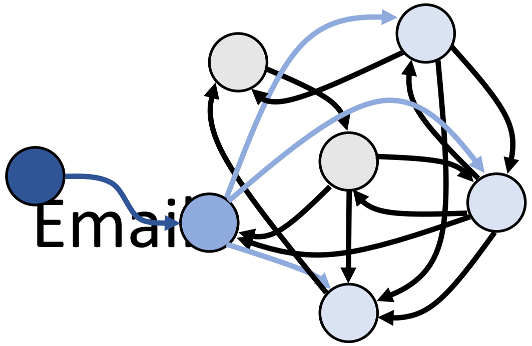
Funding

NSF GRFP: DGE-1321851



Kim, J. Z., Lu, Z., Nozari, E., Pappas, G. J., & Bassett, D. S. (2020). Teaching Recurrent Neural Networks to Infer Global Temporal Structure from Local Examples. (Accepted, *Nat. Mach. Intell.*).

Thank you!

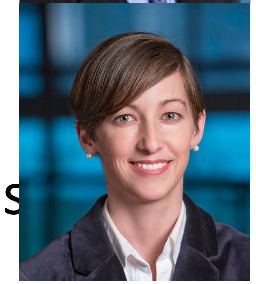


Collaborators

Zhixin Lu
jinsu1@seas. du

Erfan Nozari

George Papp

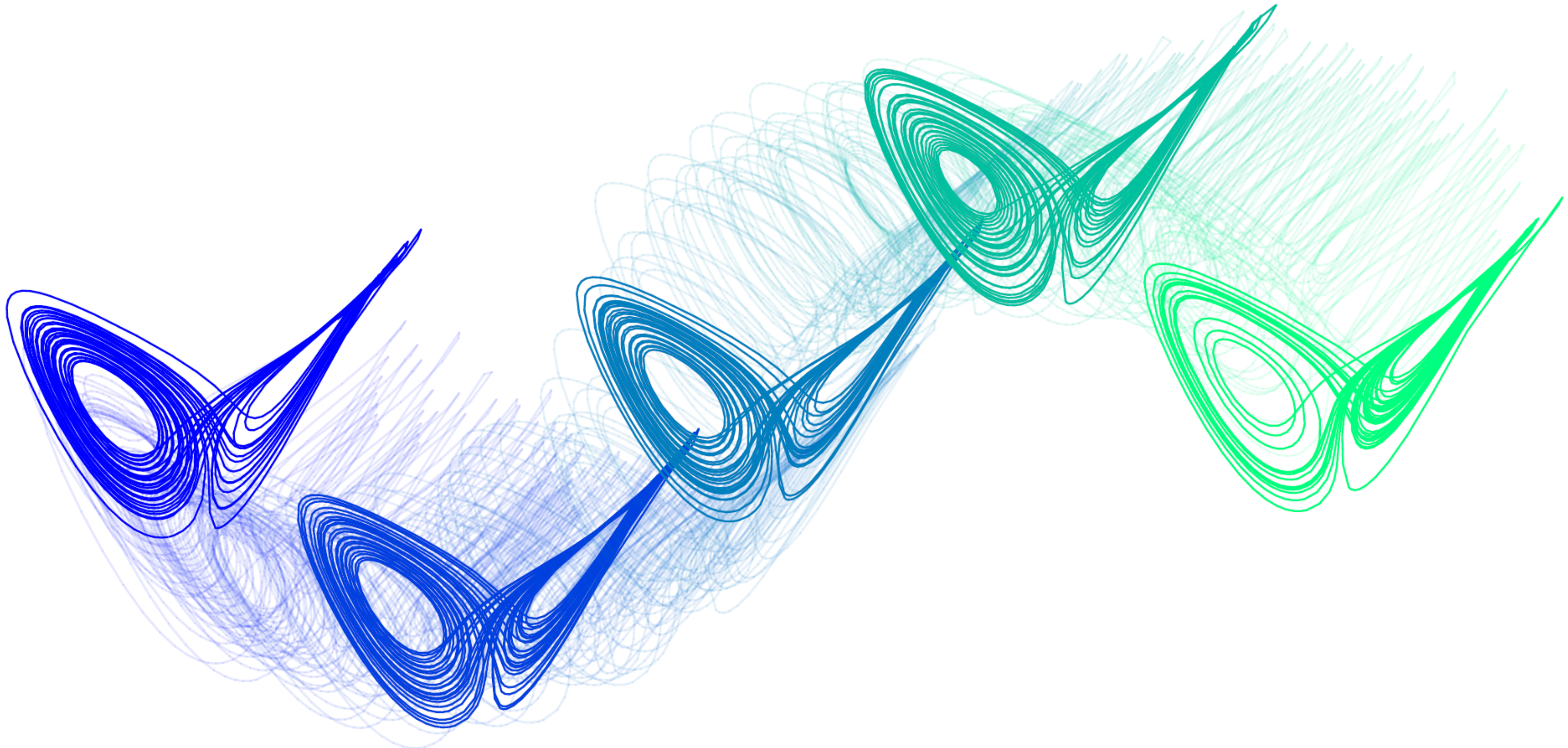


Danielle Bass

Funding

NSF GRFP: DGE-1321851

Emai



Kim, J. Z., Lu, Z., Nozari, E., Pappas, G. J., & Bassett, D. S. (2020). Teaching Recurrent Neural Networks to Infer Global Temporal Structure from Local Examples. (Accepted, *Nat. Mach. Intell.*).