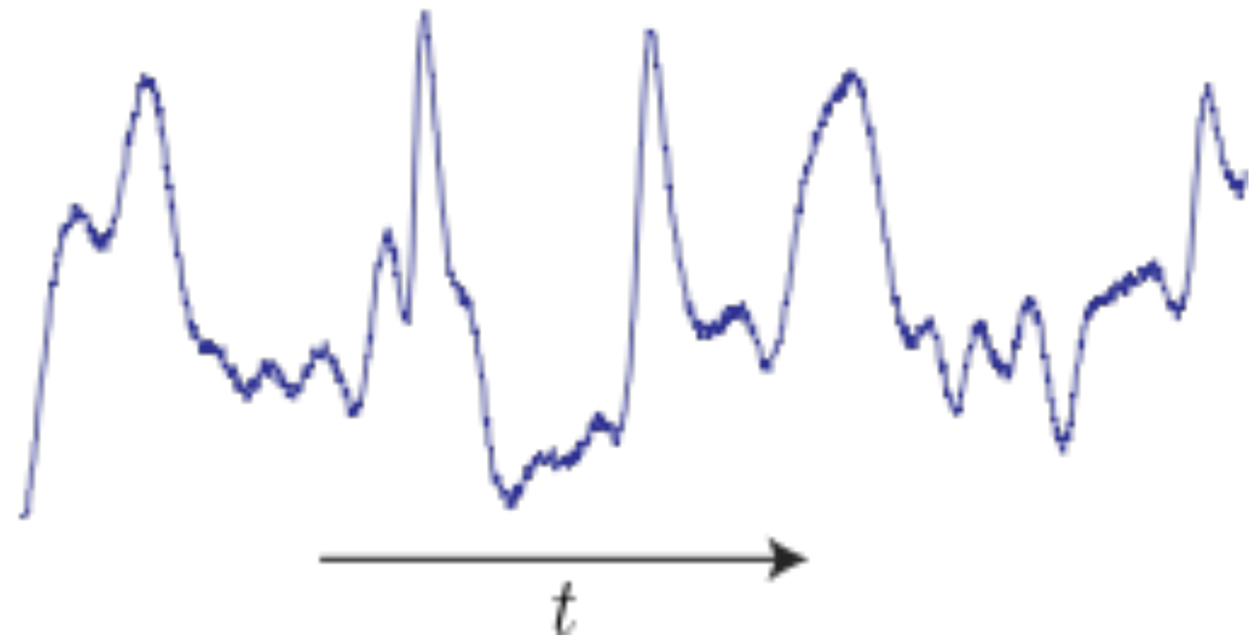
Hybrid Forecasting of Complex Systems:

Combining Machine Learning with Knowledge-Based Models

Michelle Girvan

Key Collaborators: Ed Ott, Brian Hunt, Jaideep Pathak

Introduction

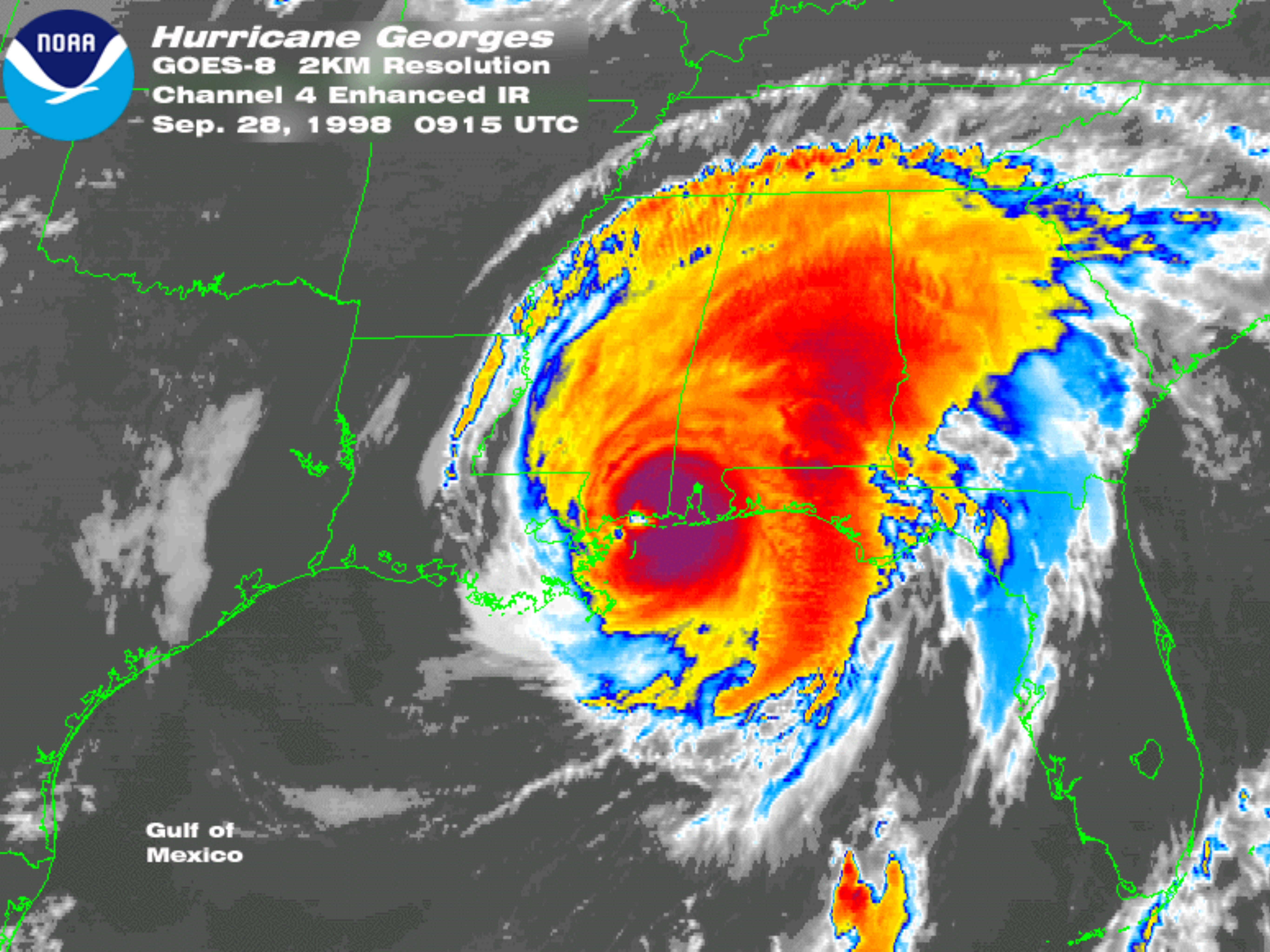


- **Main question:** How can we predict the future of some complex time series based on past history?

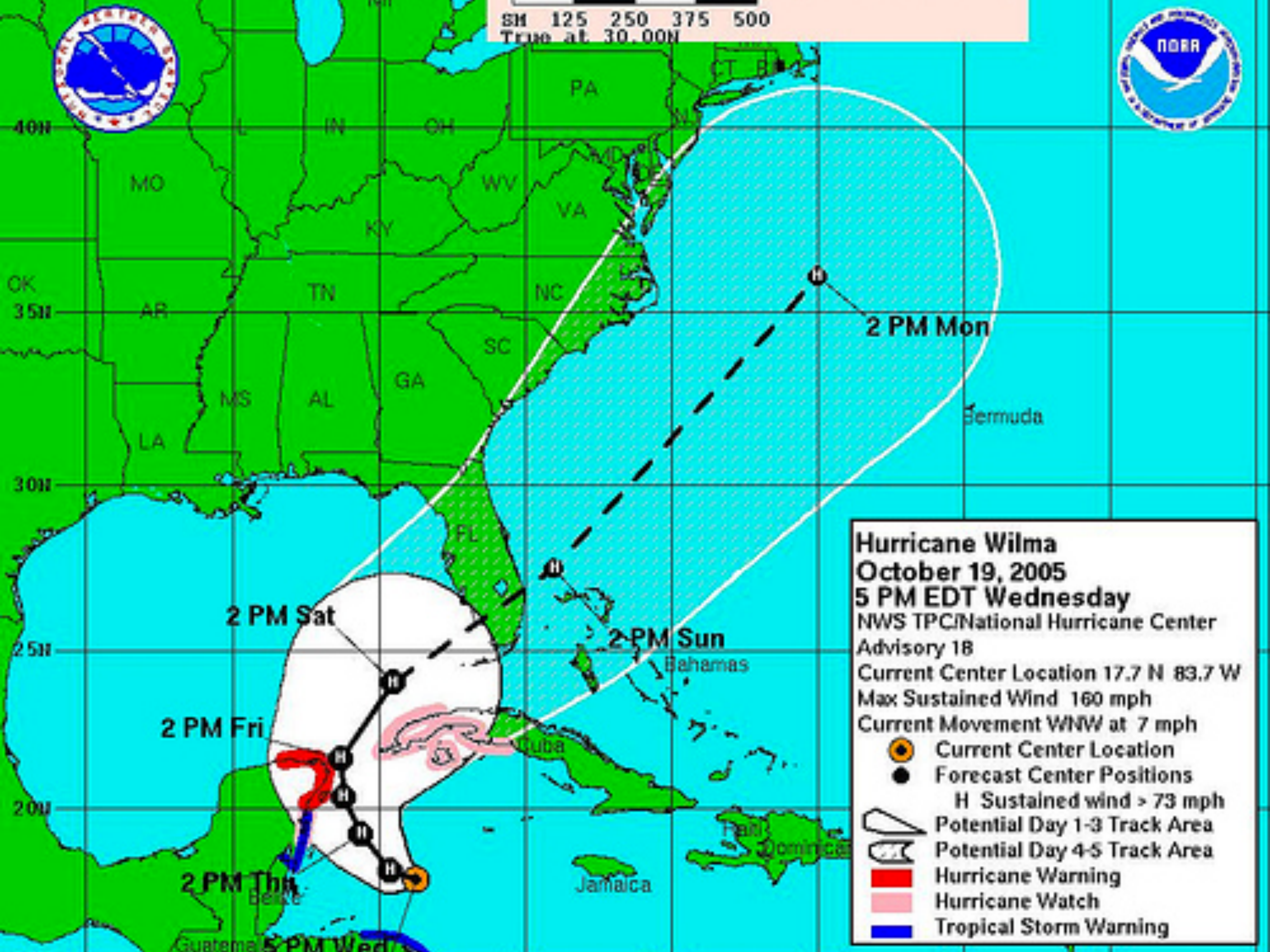
- **Applications:** Weather and climate forecasting, stock market prediction, social media behaviors, living systems, and more

- **Hybrid Approach:** Combine artificial intelligence/machine learning with mathematical modeling

What kinds of systems are hard to predict?



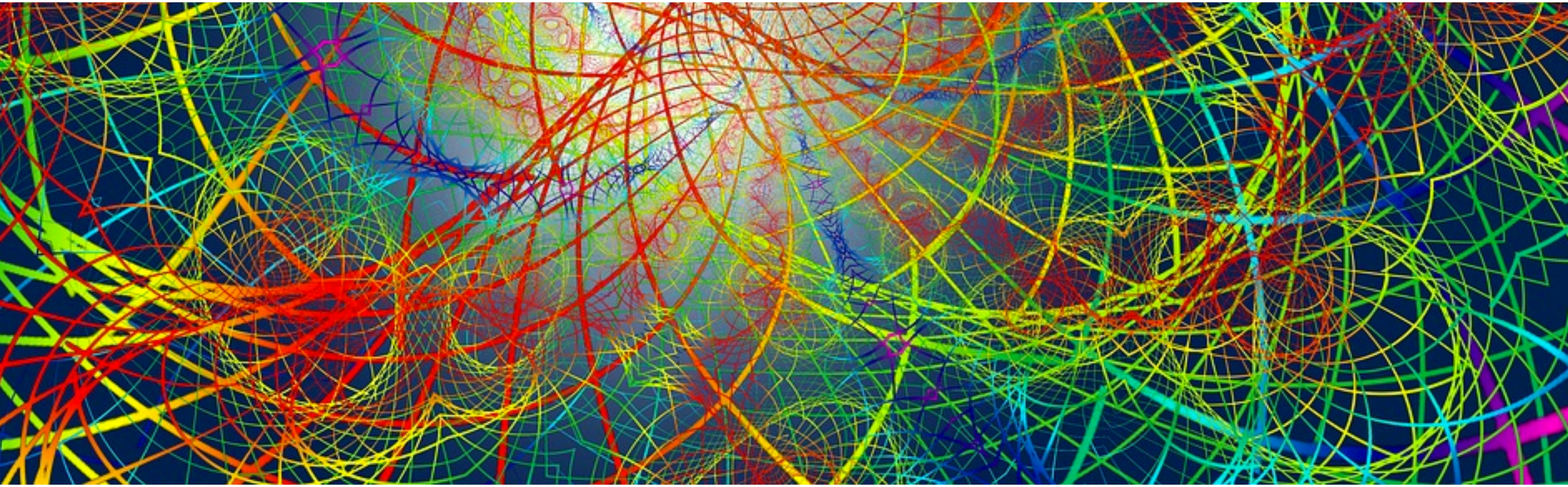
NOAA
Hurricane Georges
GOES-8 2KM Resolution
Channel 4 Enhanced IR
Sep. 28, 1998 0915 UTC
Gulf of
Mexico



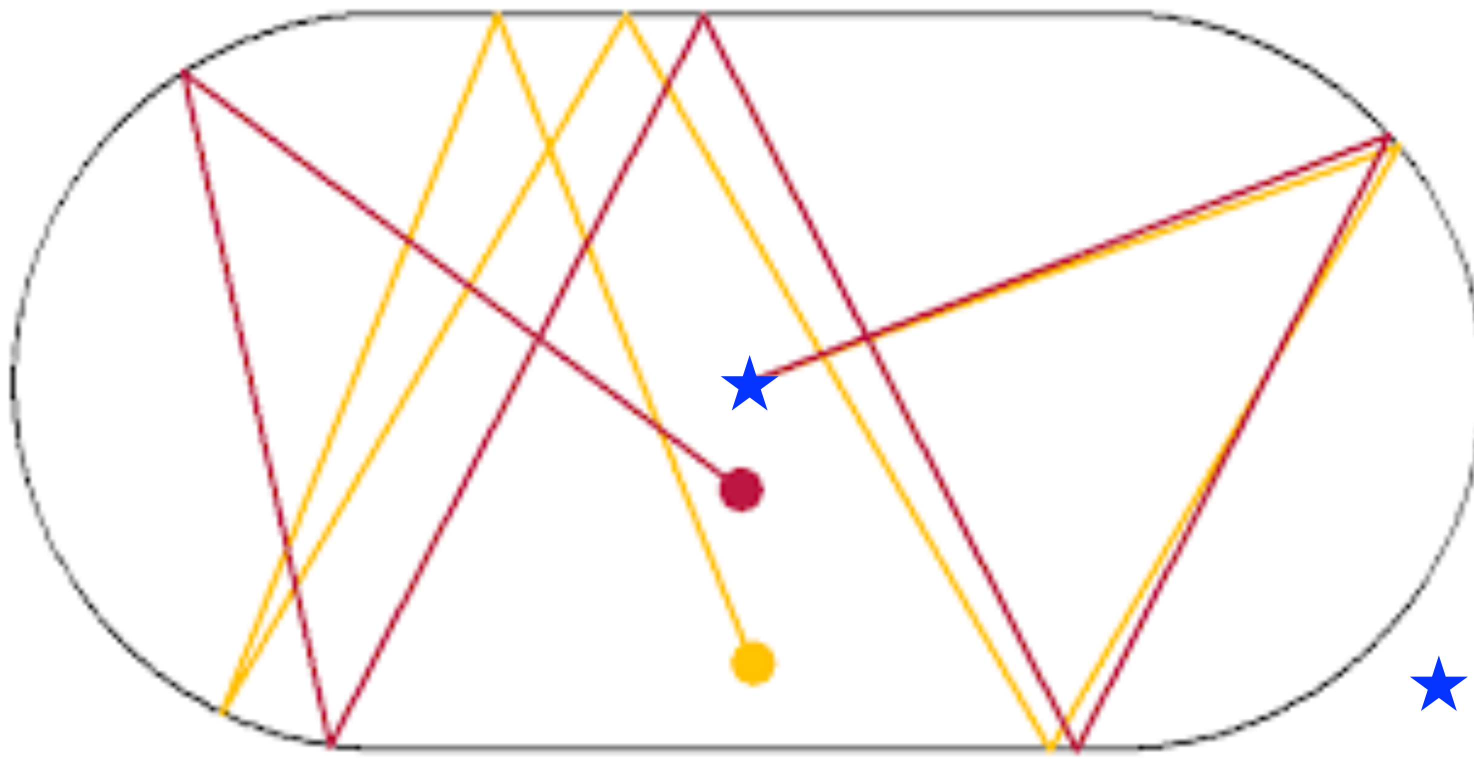
SM 125 250 375 500
True at 30.00N
40N
35N
30N
25N
20N
PA
CT
IN
OH
MO
WV
VA
KY
OK
AR
TN
NC
SC
GA
MS
AL
LA
FL
2 PM Mon
Bermuda
2 PM Sat
2 PM Sun
Bahamas
2 PM Fri
Cuba
Jamaica
2 PM Thu
Belize
Guatemala
5 PM Wed
Hurricane Wilma
October 19, 2005
5 PM EDT Wednesday
NWS TPC/National Hurricane Center
Advisory 18
Current Center Location 17.7 N 83.7 W
Max Sustained Wind 160 mph
Current Movement WNW at 7 mph
Current Center Location
Forecast Center Positions
H Sustained wind > 73 mph
Potential Day 1-3 Track Area
Potential Day 4-5 Track Area
Hurricane Warning
Hurricane Watch
Tropical Storm Warning

Question: What makes systems like the weather hard to predict?

Answer: Chaos



Features of Chaos



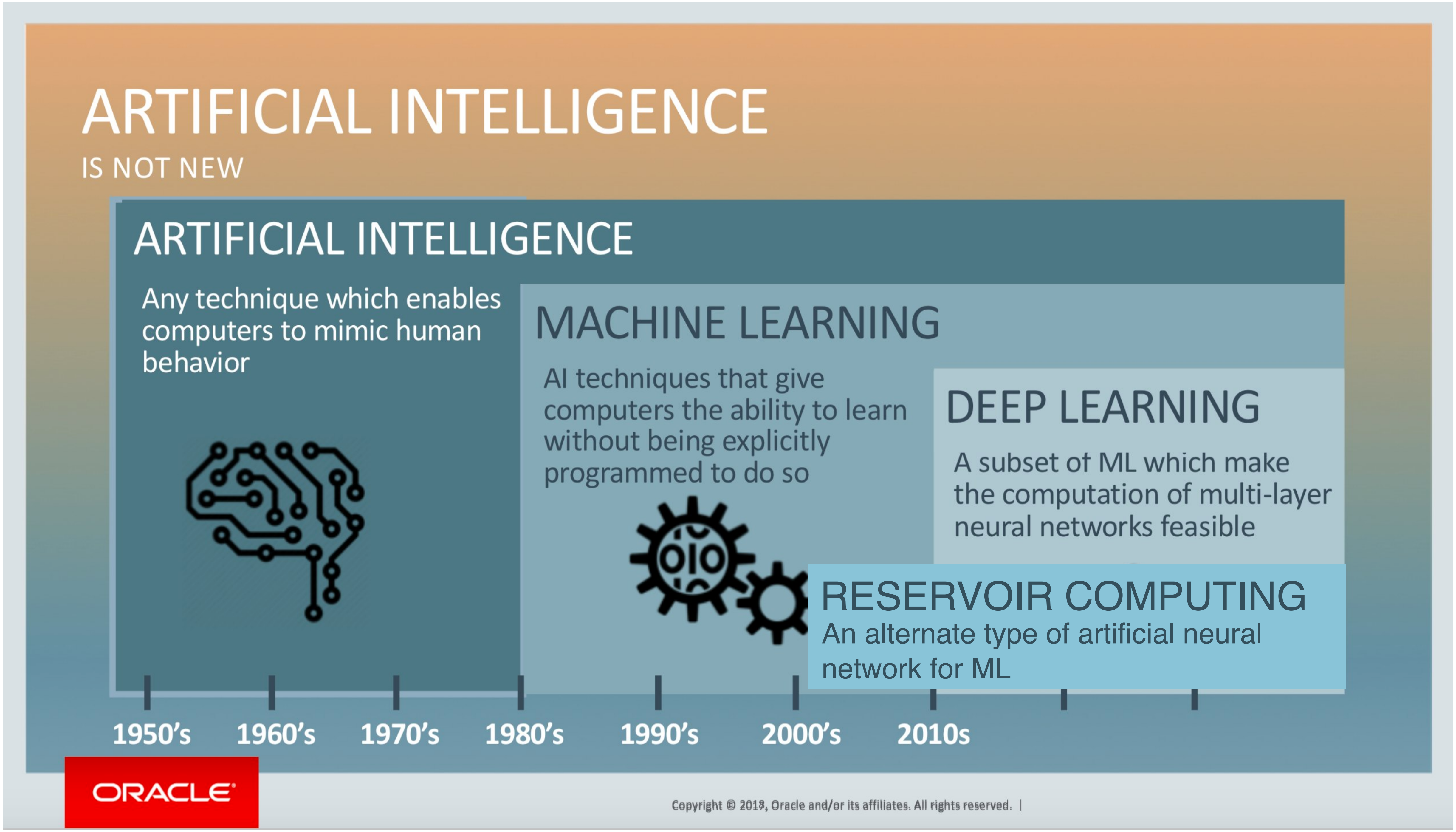
★ 2 initial conditions (yellow and red) start close together

- Sensitivity to initial conditions, i.e. the "***butterfly effect***"
- ***Lyapunov time*** = time over which the distance between initially close increases by a factor of *e* (2.78128…)
- Long term prediction is impossible even though system is ***deterministic***
- Behavior can appear random in the long term

Stochastic systems are also hard to predict

How can machine learning help us predict these systems?

Background: Artificial Intelligence and Machine Learning



https://blogs.oracle.com/bigdata/difference-ai-machine-learning-deep-learning

Artificial Neural Networks: A kind of machine learning approach that uses networks of neuron-like units to process information

Deep Learning

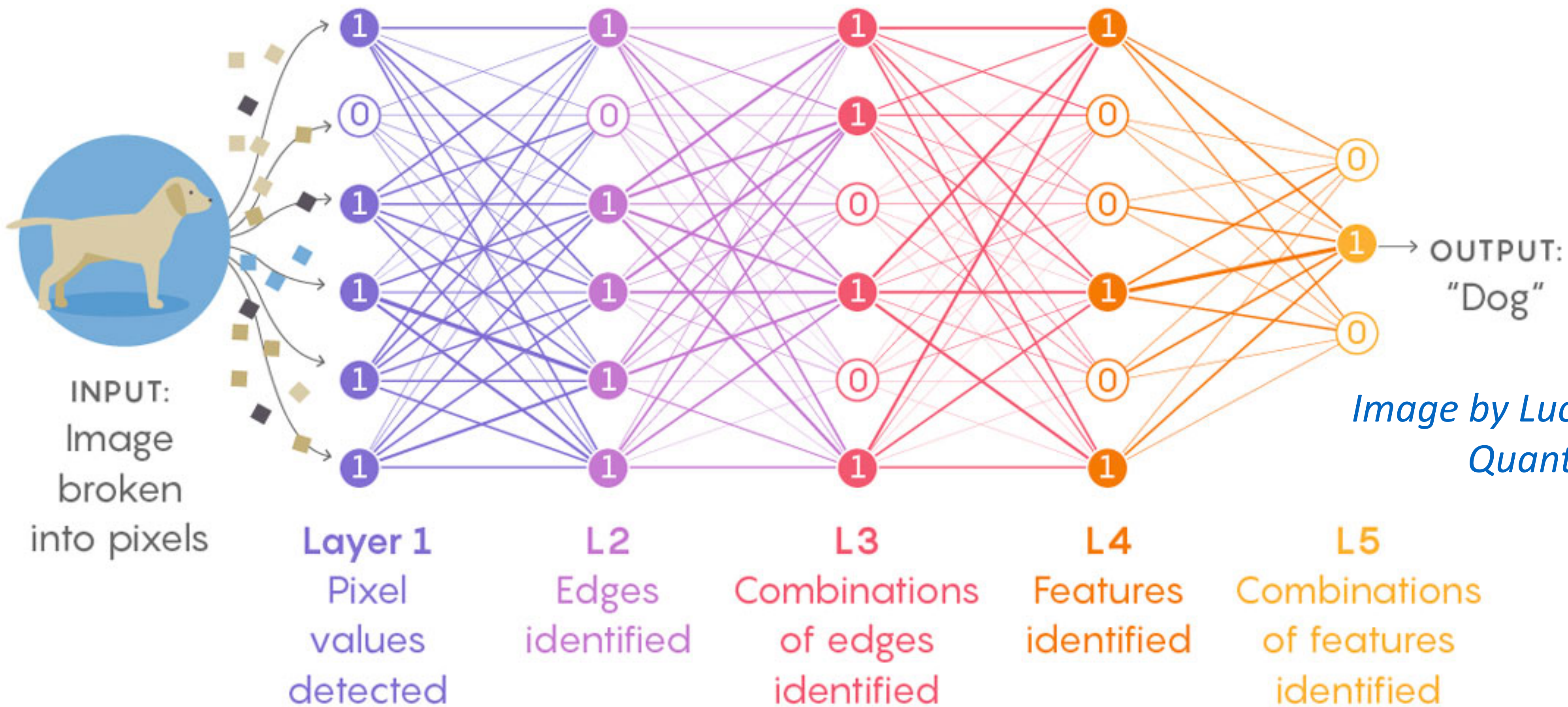


Image by Lucy Reading-Ikkanda
Quanta Magazine

- Information processed through successive layers of neuron-like units. Primarily ***feed-forward*** architecture.
- Deeper layers perform more sophisticated information processing.
- Weights between the layers are ***trained*** so that the input matches the desired output for a set of ***training data***. Large amounts of training data needed.
- Great for classification tasks, but not as good for predicting time evolution
- ***Feedback*** loops can be added to give the system memory but these make training the weights much harder and sometimes impossible

Reservoir Computing

- Provides a way to train Recurrent Neural Networks
- Independently introduced by Jaeger (2001), and Maass et al. (2002).
- Main component is a very high-dimensional system, called the reservoir.
- The reservoir provides a rich repository of dynamics. With training it offers a “universal” dynamical system
- Amenable to simple hardware implementations

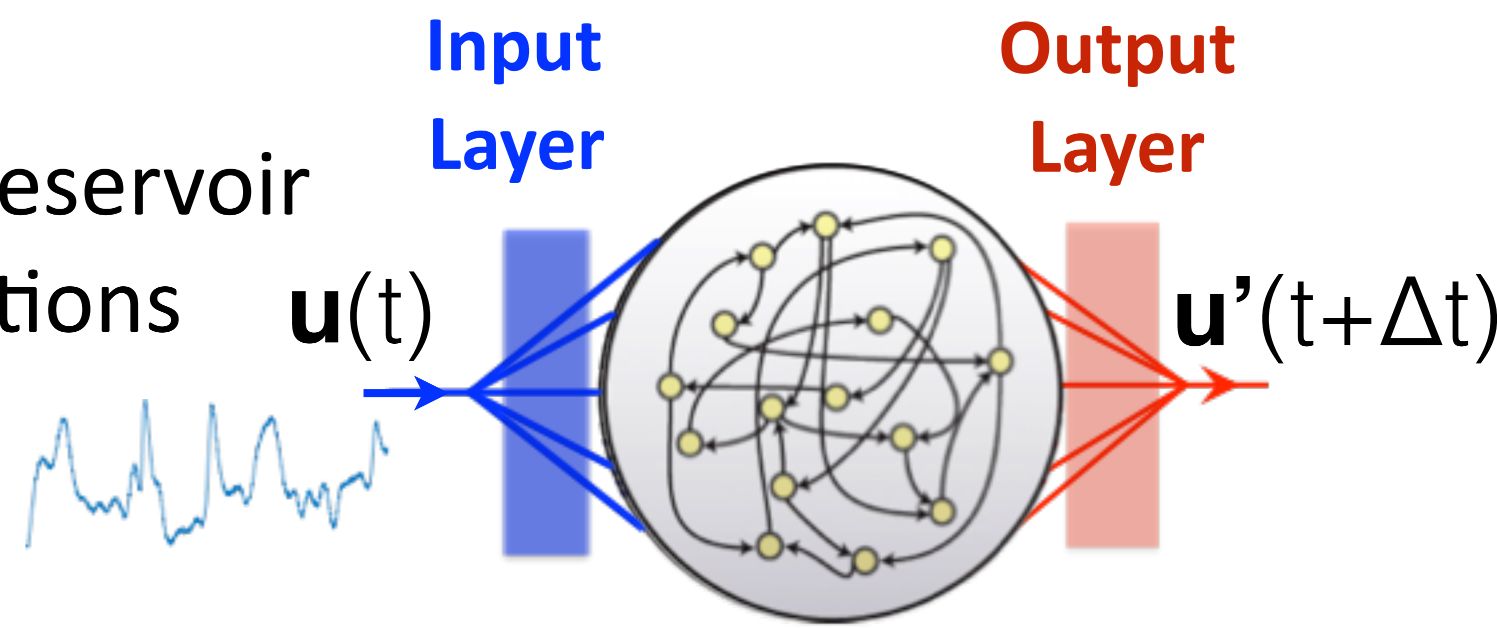
Reservoir Computing

Reservoir: Recurrent Artificial Neural Network, capable of rich dynamics

Input: Input through fed into reservoir through fixed, random connections

Reservoir Dynamics:

$$\mathbf{r}(t+\Delta t) = \tanh\left[\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{\text{in}}\mathbf{u}(t)\right]$$



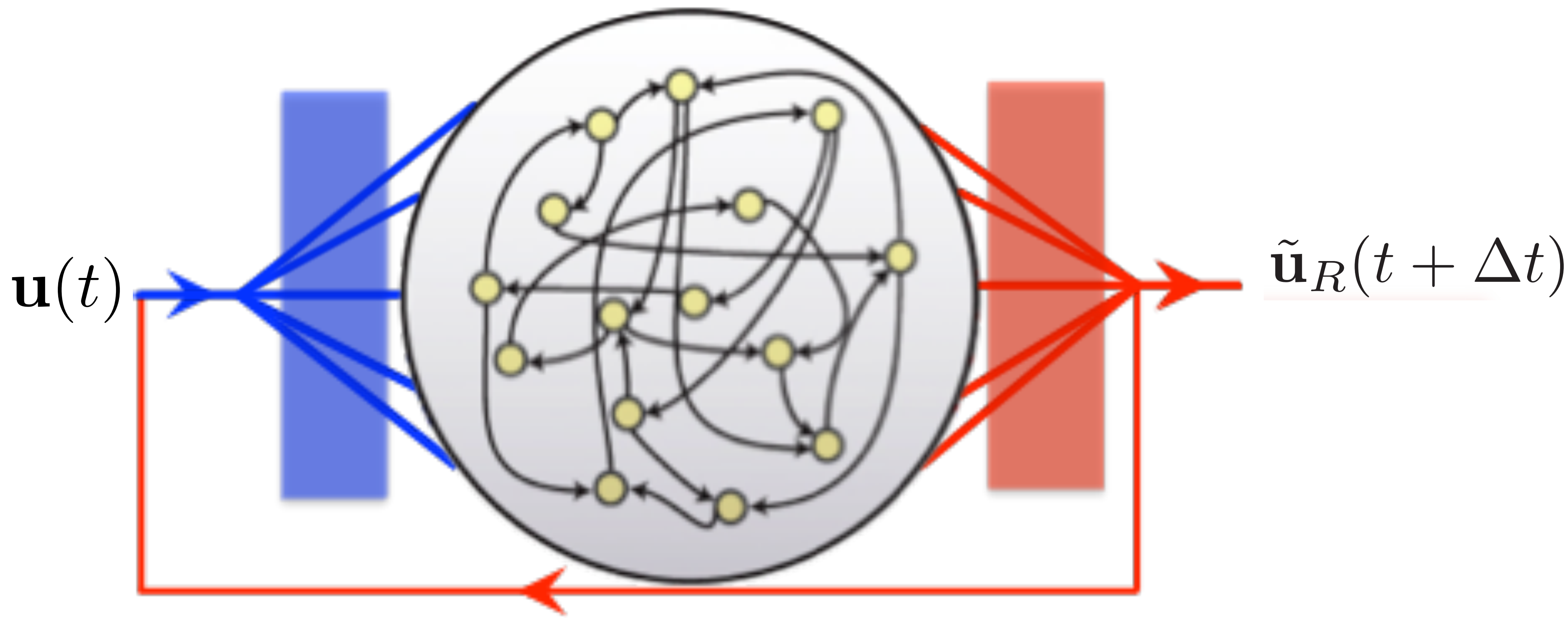
Reservoir

- neuron-like units
- random, sparse connections
- internal firing dynamics

Output: Output weights are fit to give the best outputs for the ***training*** data

Key advantage: Training is simple and efficient because only output weights are adjusted

Predicting multiple time steps forward



In prediction mode, the output of the reservoir is fed back in as input

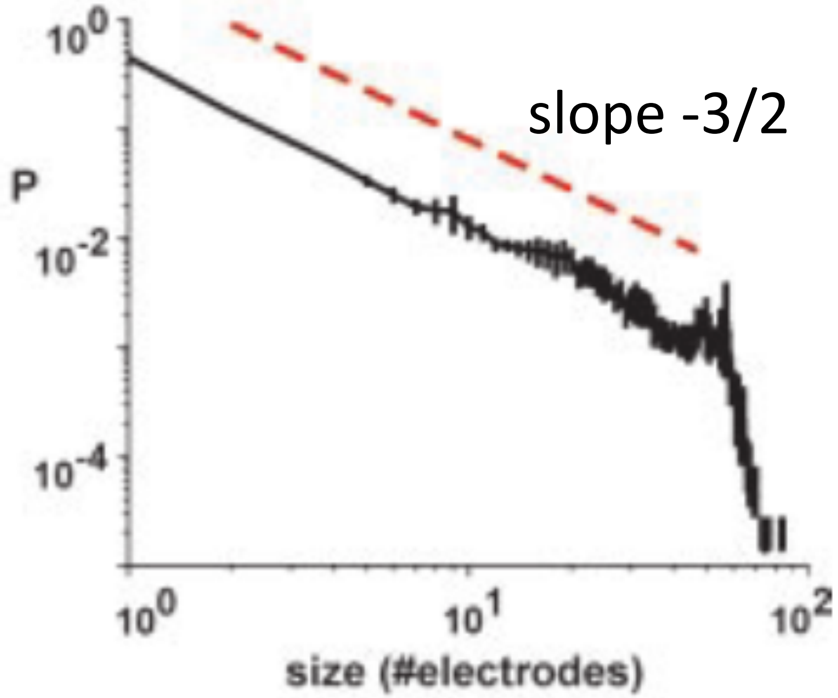
$$\tilde{\mathbf{u}}_R(t) = \mathbf{W}_{\text{out}}\mathbf{r}^\star(t)$$
$$\mathbf{r}(t+\Delta t) = \tanh\left[\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{\text{in}}\tilde{\mathbf{u}}_R(t)\right]$$

Q: Should the reservoir be critical?

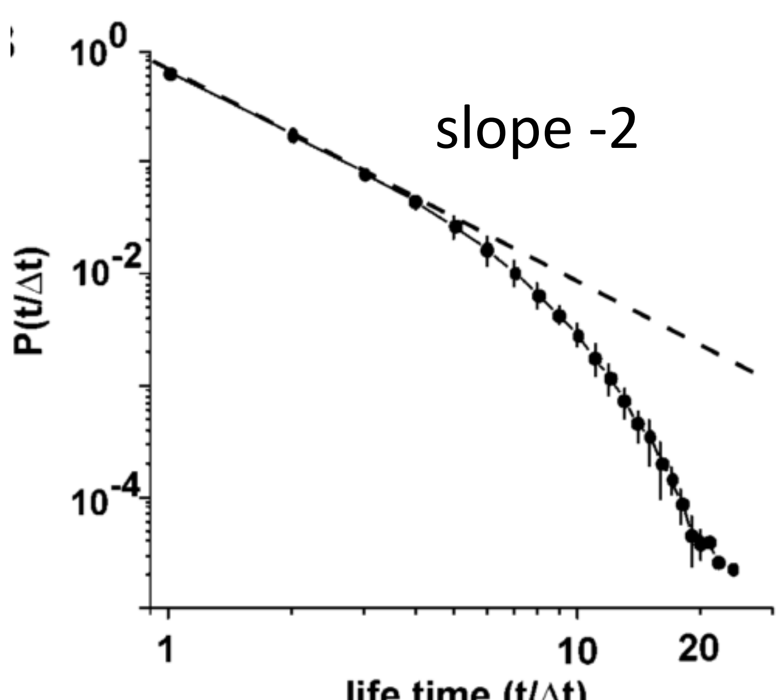
Critical Dynamics in Networks of Neurons

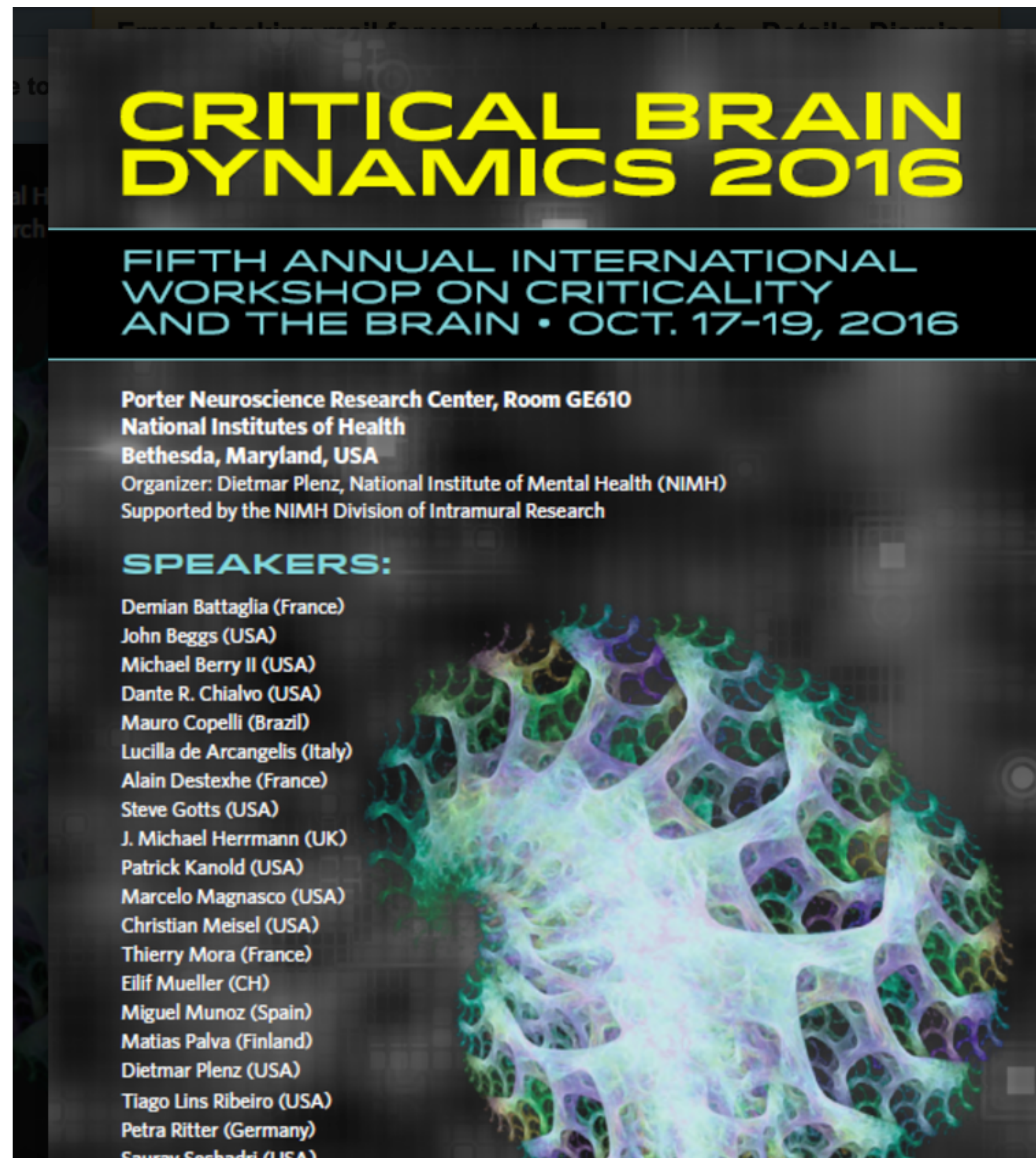
Experimental evidence: Avalanche size and duration distributions follow power-laws (first observed in cultured neurons by Beggs and Plenz, 2003)

Size distribution:



Duration distribution:

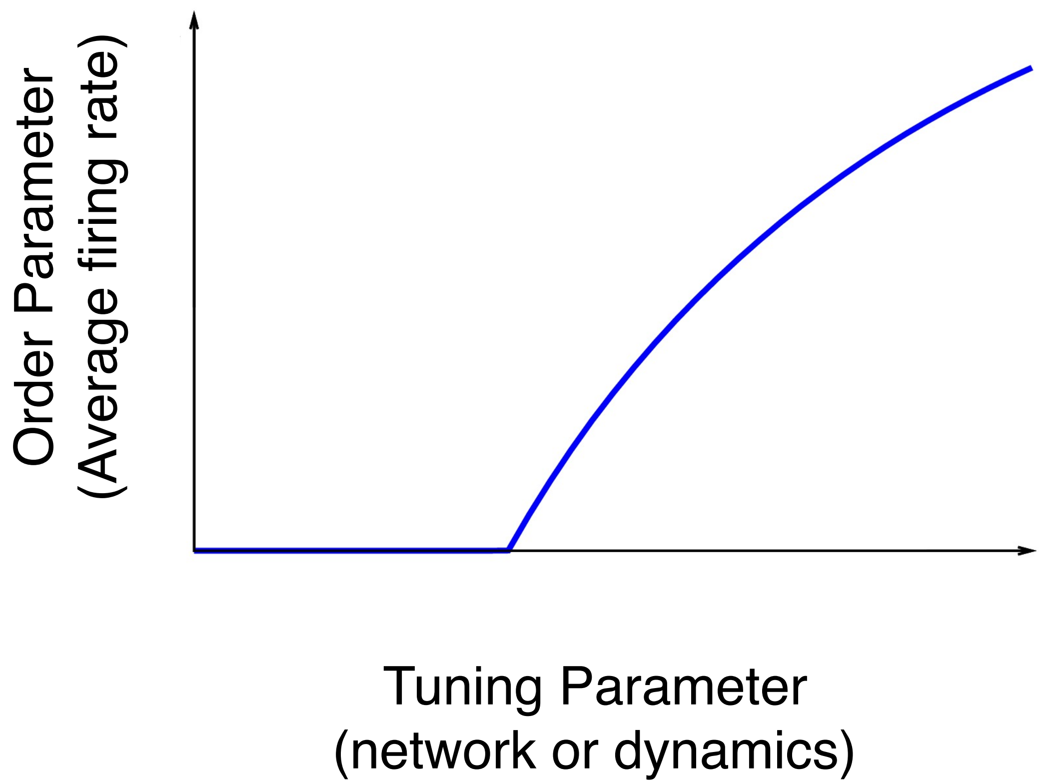




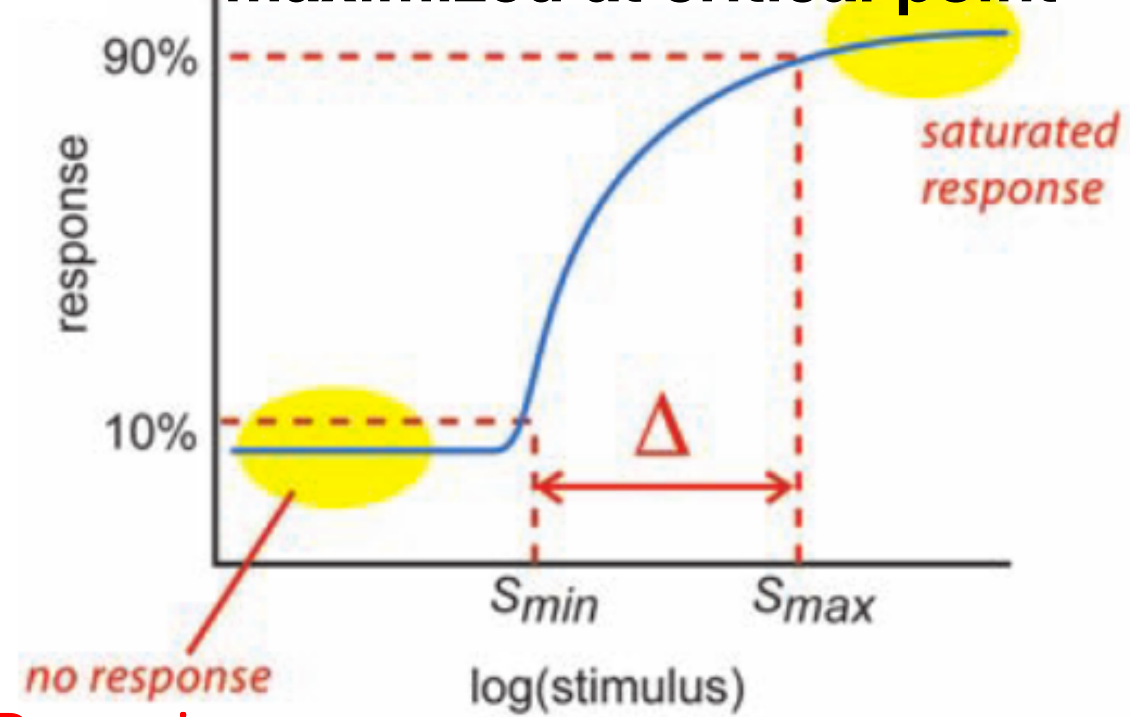
Q: Should the reservoir be critical?

Lessons from modeling neural networks:

Phase Transition in Dynamics



"Dynamic range" maximized at critical point



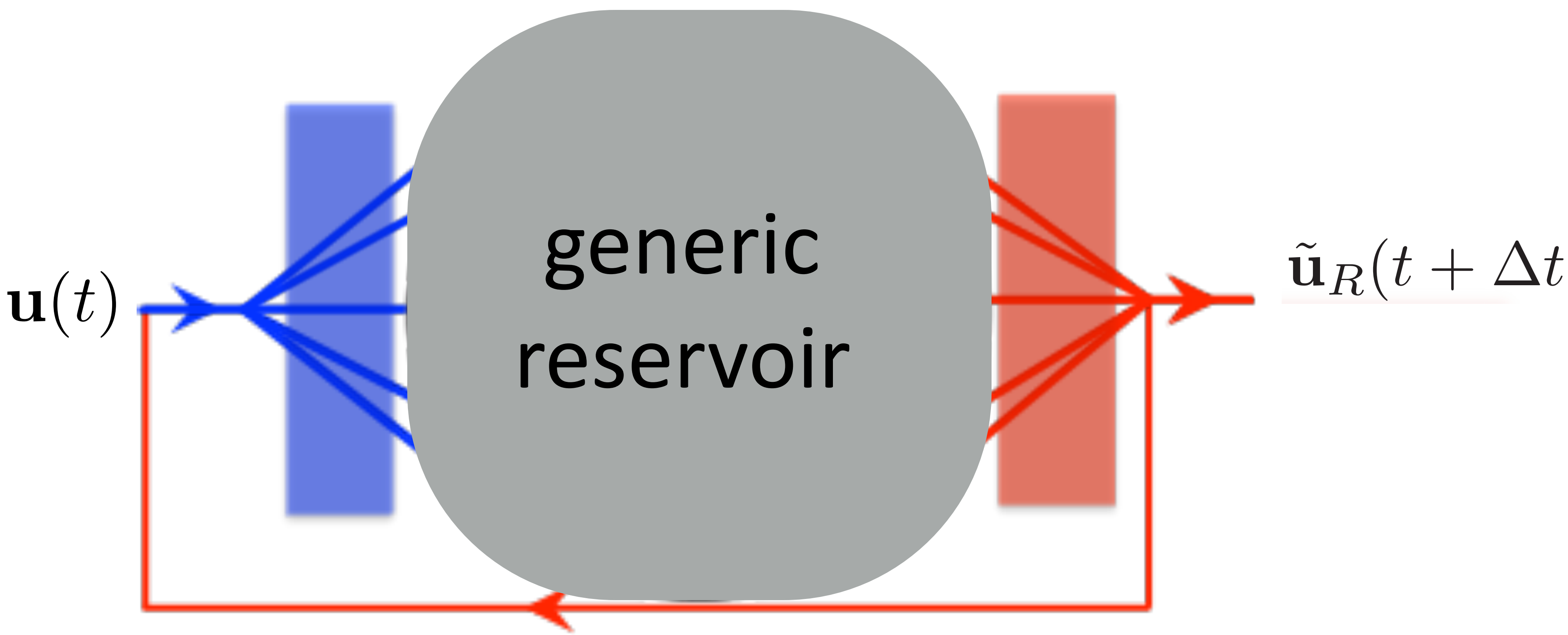
Dynamic range:

$$\Delta = \log_{10}(S_{max}) - \log_{10}(S_{min})$$

Information processing is maximized for critical avalanching dynamics: Maximized dynamic range Δ at the critical point. (Kinouchi and Copelli 2006, Shew and Plenz 2013)

A: Spectral radius near 1 (i.e. critical) gives good results, but we see robust performance over a wide range of values.

Reservoir doesn't have to be an Artificial Neural Network

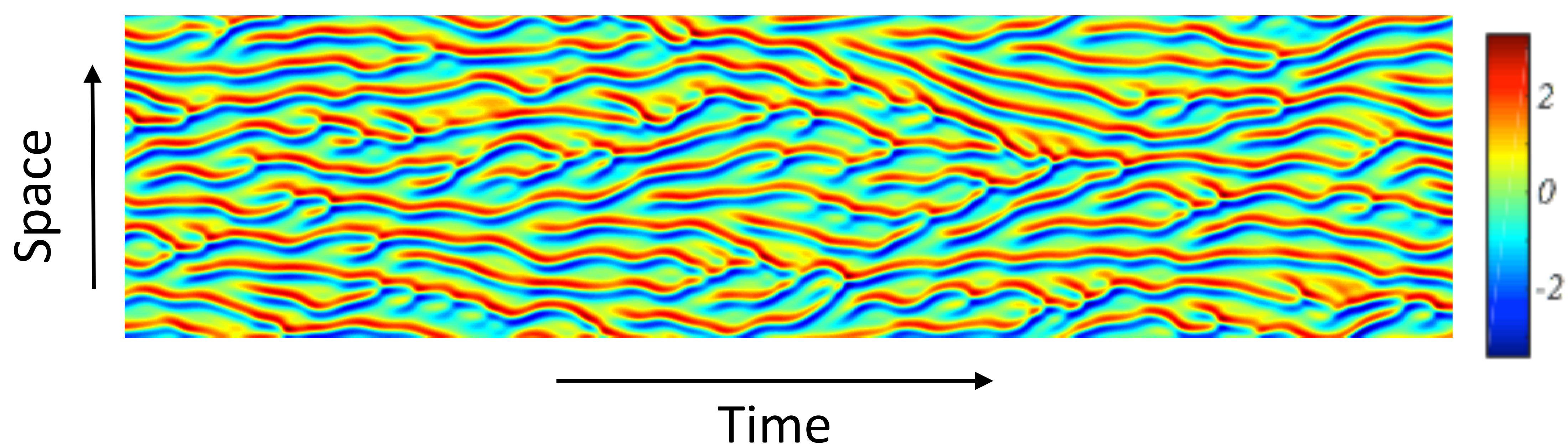


Reservoir requirements

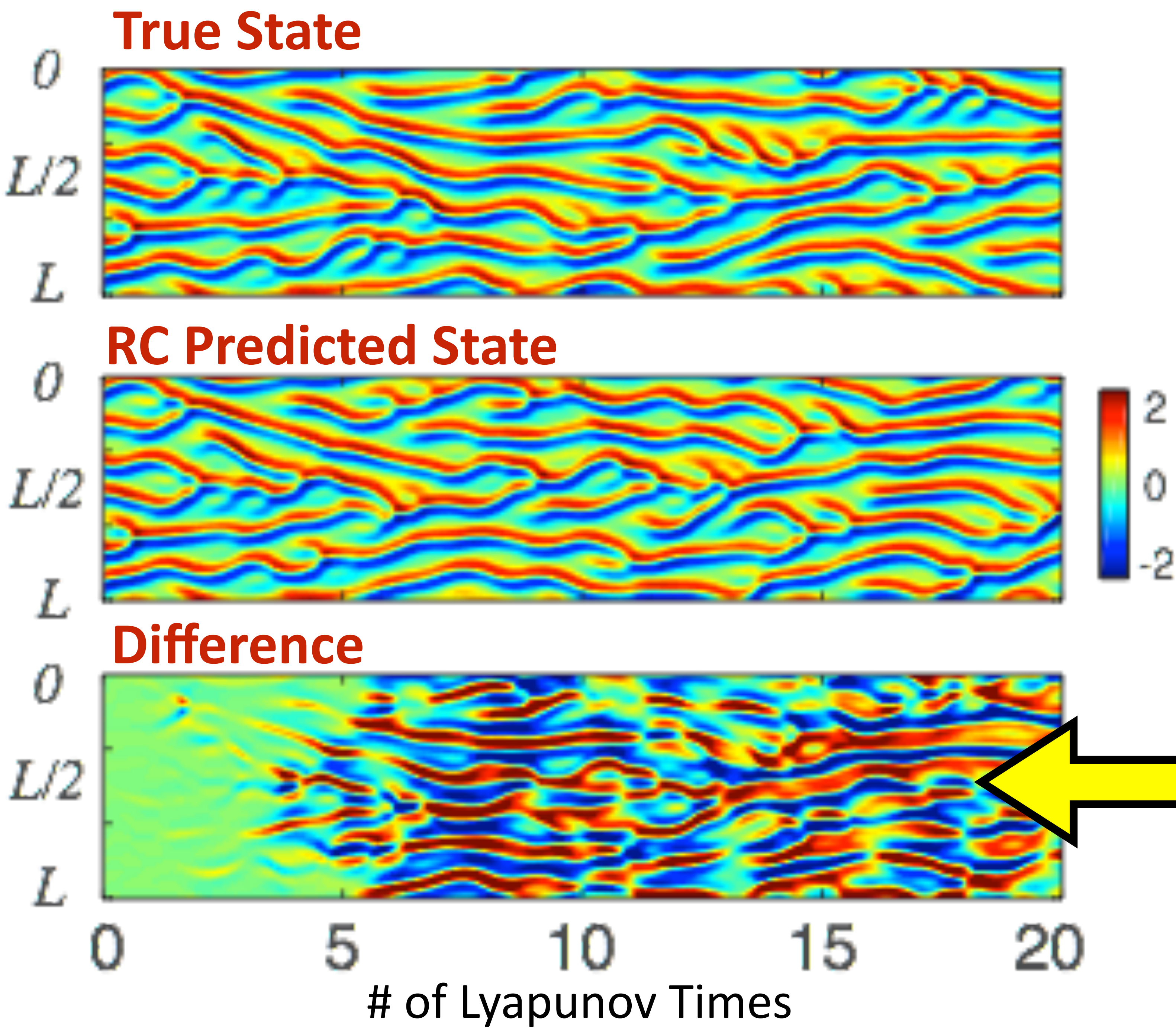
- Input-output system
- Complex repository of dynamics
- Internal memory

Reservoir Computing for Predicting Spatiotemporally Chaotic Systems

- **Spatiotemporal Chaos:** Pattern evolves in space and time. Time evolution is chaotic. Example weather.
- **Our test system:** Kuramoto Sivasinshy (KS) system, designed to model the chaotic propagation of waves in space and time $y_t = -yy_x - y_{xx} - y_{xxxx}$



Results

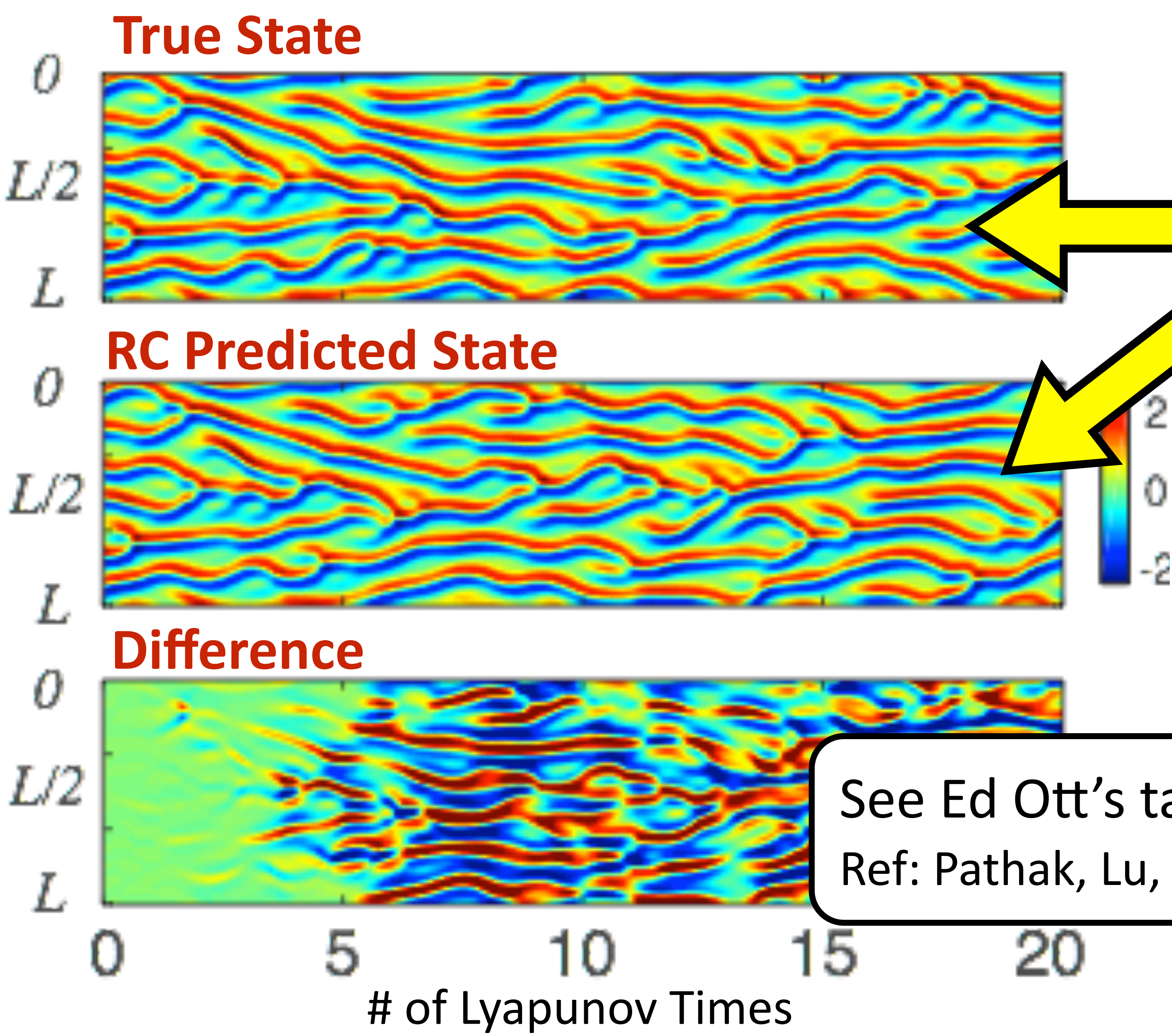


We obtain high quality predictions for about 5 multiples of the Lyapunov time.

In that time, the distance between initially close trajectories grows by a factor of about 150 (e^5).

Ref: J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott. *Phys. Rev. Lett.* (2018).

Results



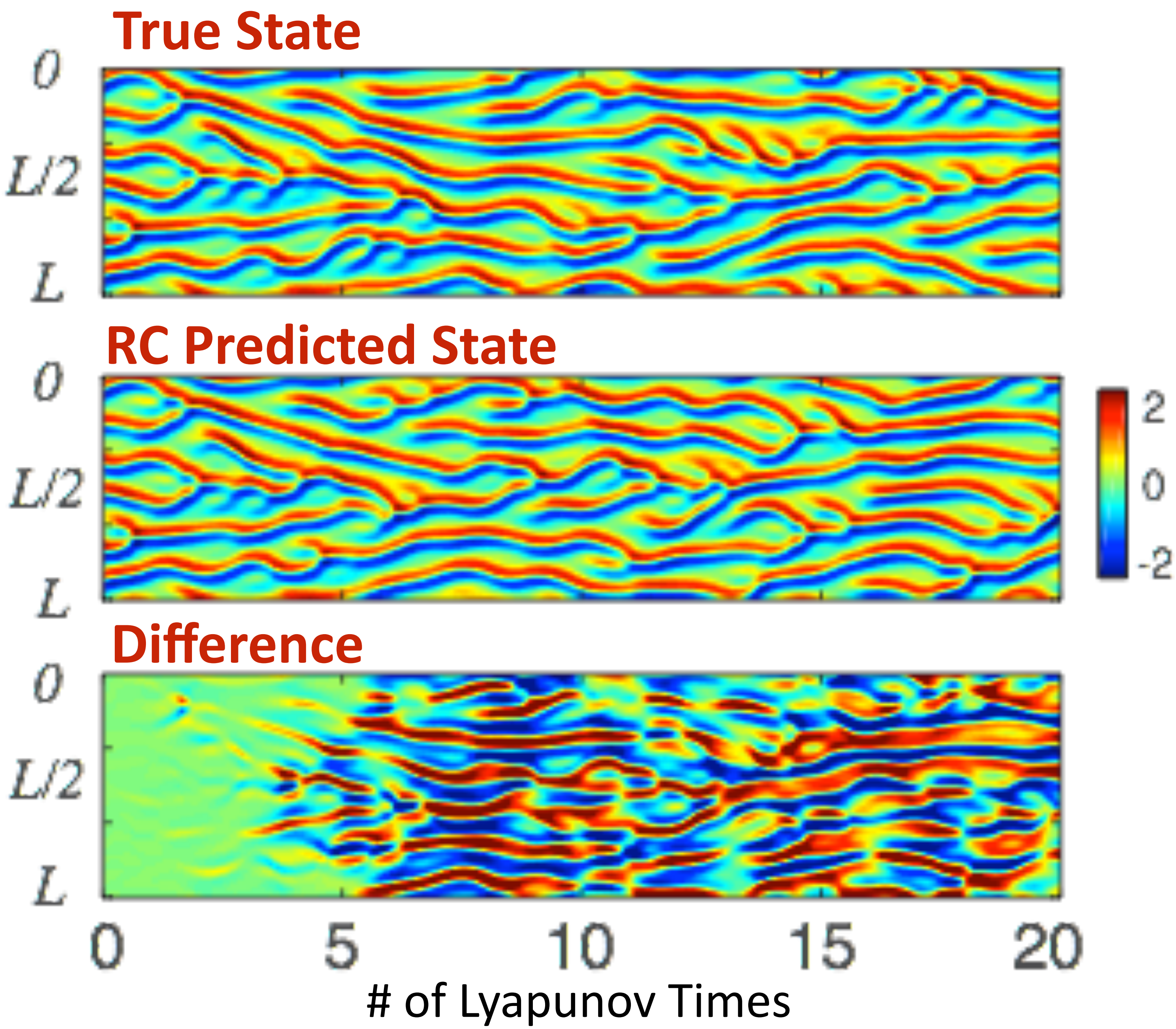
Notice that the predicted state looks very much like the true state even after the predictions become unreliable.

The "climate" is accurately captured even after the "forecast" fails.

See Ed Ott's talk on Wednesday morning
Ref: Pathak, Lu, Hunt, Girvan, and Ott. *Chaos* (2017)

Ref: J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott. *Phys. Rev. Lett.* (2018).

Results

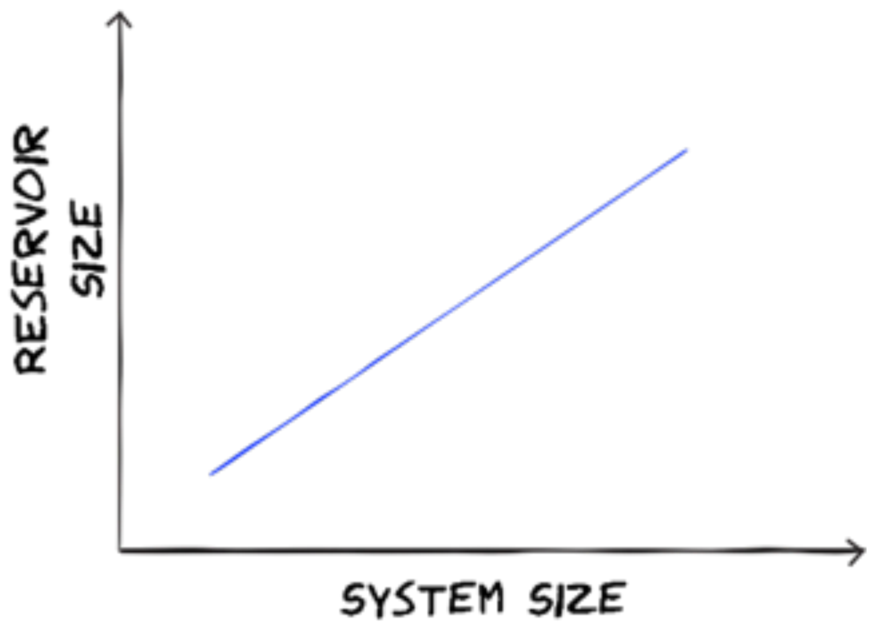


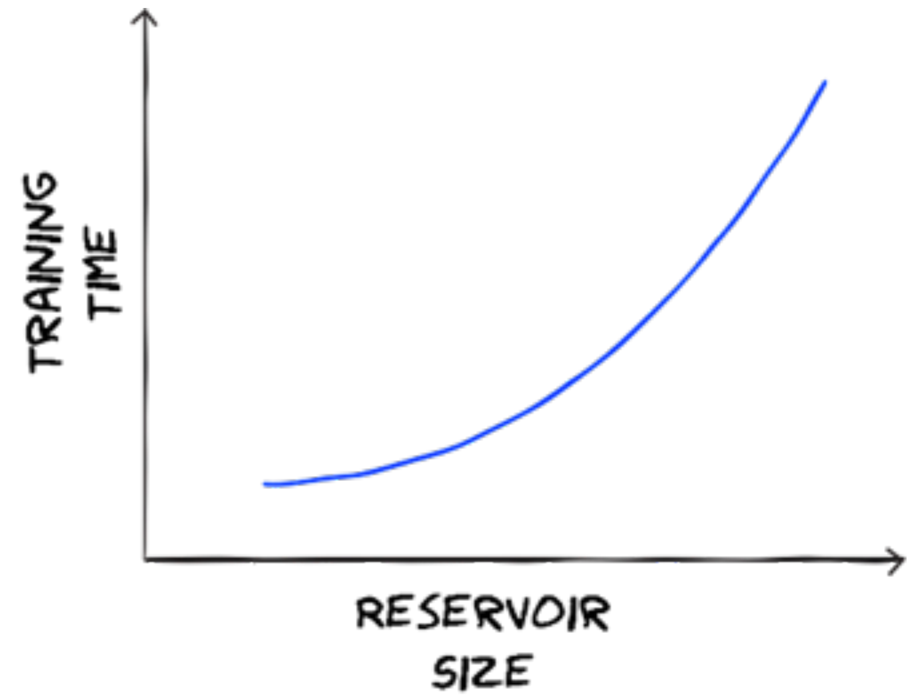
Advantages of the RC

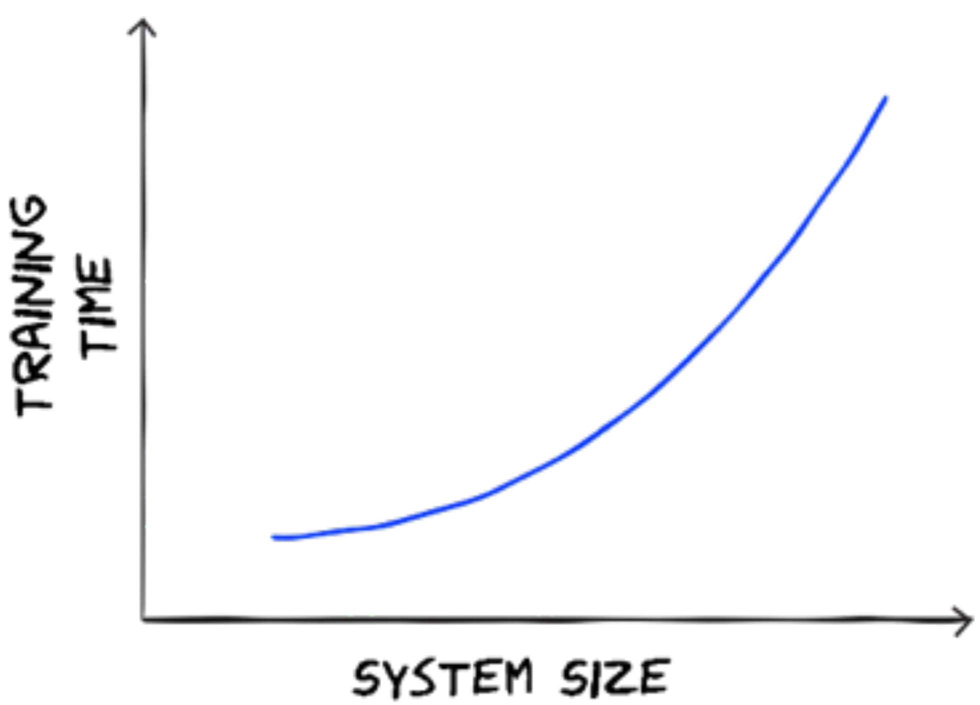
- Seems to capture the climate better than other model-free methods
- Simpler to implement
- Not prone to spurious results

Ref: J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott. *Phys. Rev. Lett.* (2018).

What about predicting really large systems?

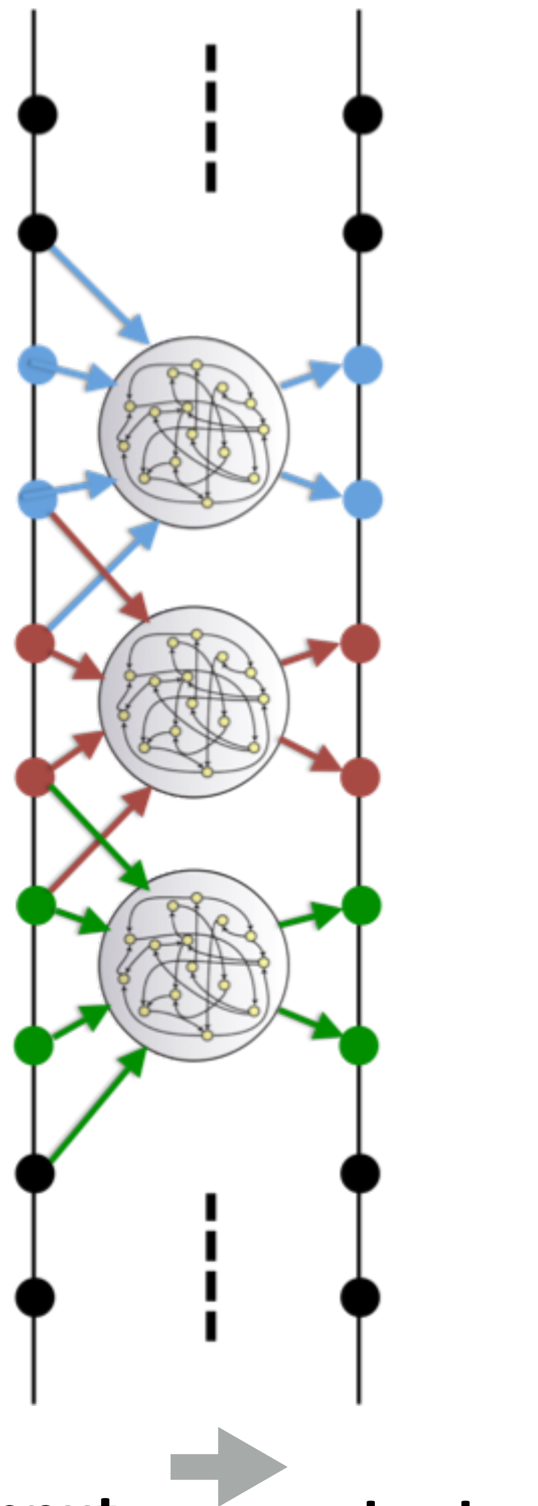






Typical spatiotemporal systems, like the weather, are very large. How do we deal with them?

Solution: Parallel Reservoir Computing



input:
time (t)

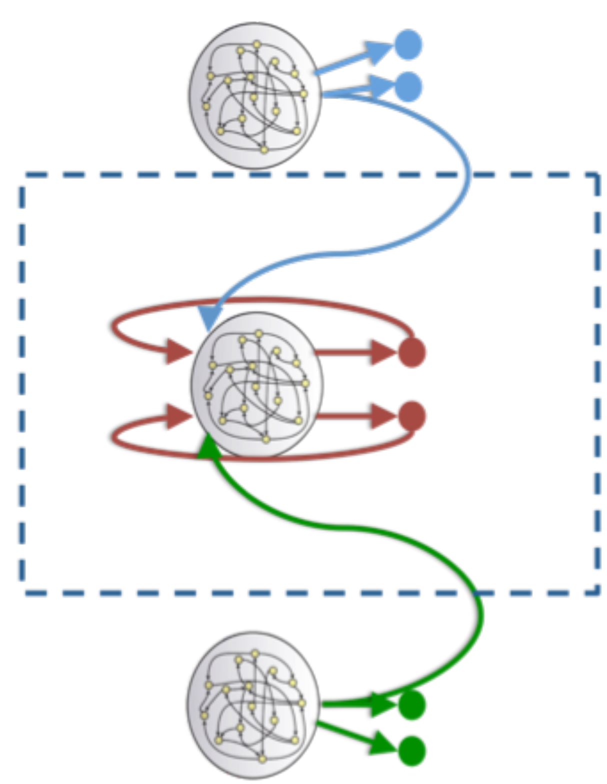
output:
(t+Δt)
next step

Training Mode

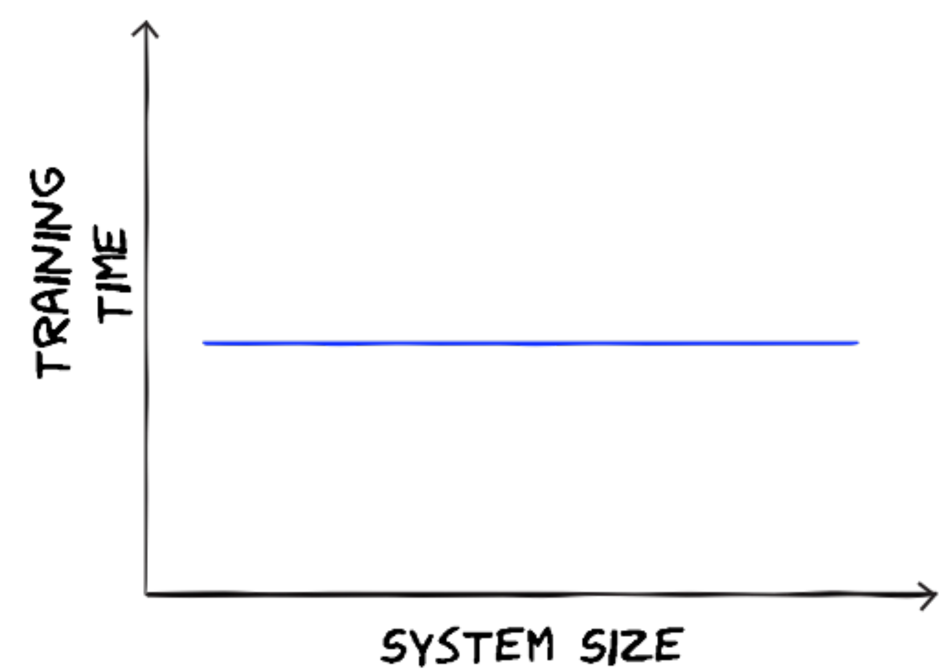
Architecture: Each reservoir takes ***inputs from a spatially local neighborhood*** and predicts a subset of its neighborhood.

Performance: Using model systems, we have shown this scheme is highly effective for very large systems.

Next challenge: Real weather forecasting



Prediction Mode



What about predicting ***stochastic*** systems?

A simple test case: Predicting user activity on Twitter

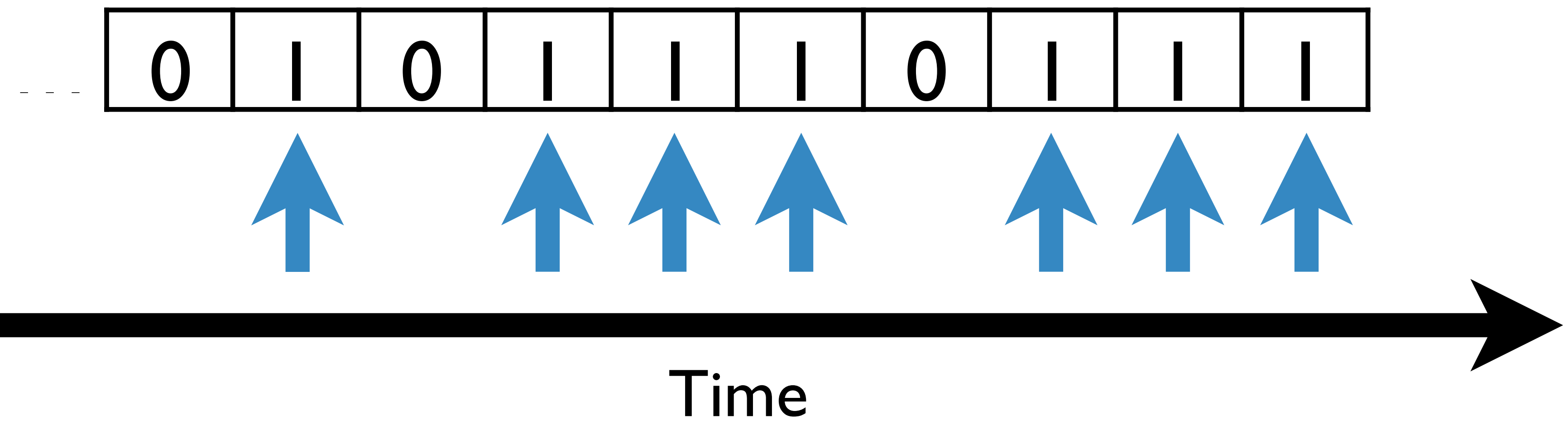
Ref: D. Darmon, J. Sylvester, M. Girvan, and W. Rand, "Predictability of User Behavior in Social Media: Bottom-Up v. Top-Down Modeling," International Conference on Social Computing (2013).

Predicting User Activity on Twitter

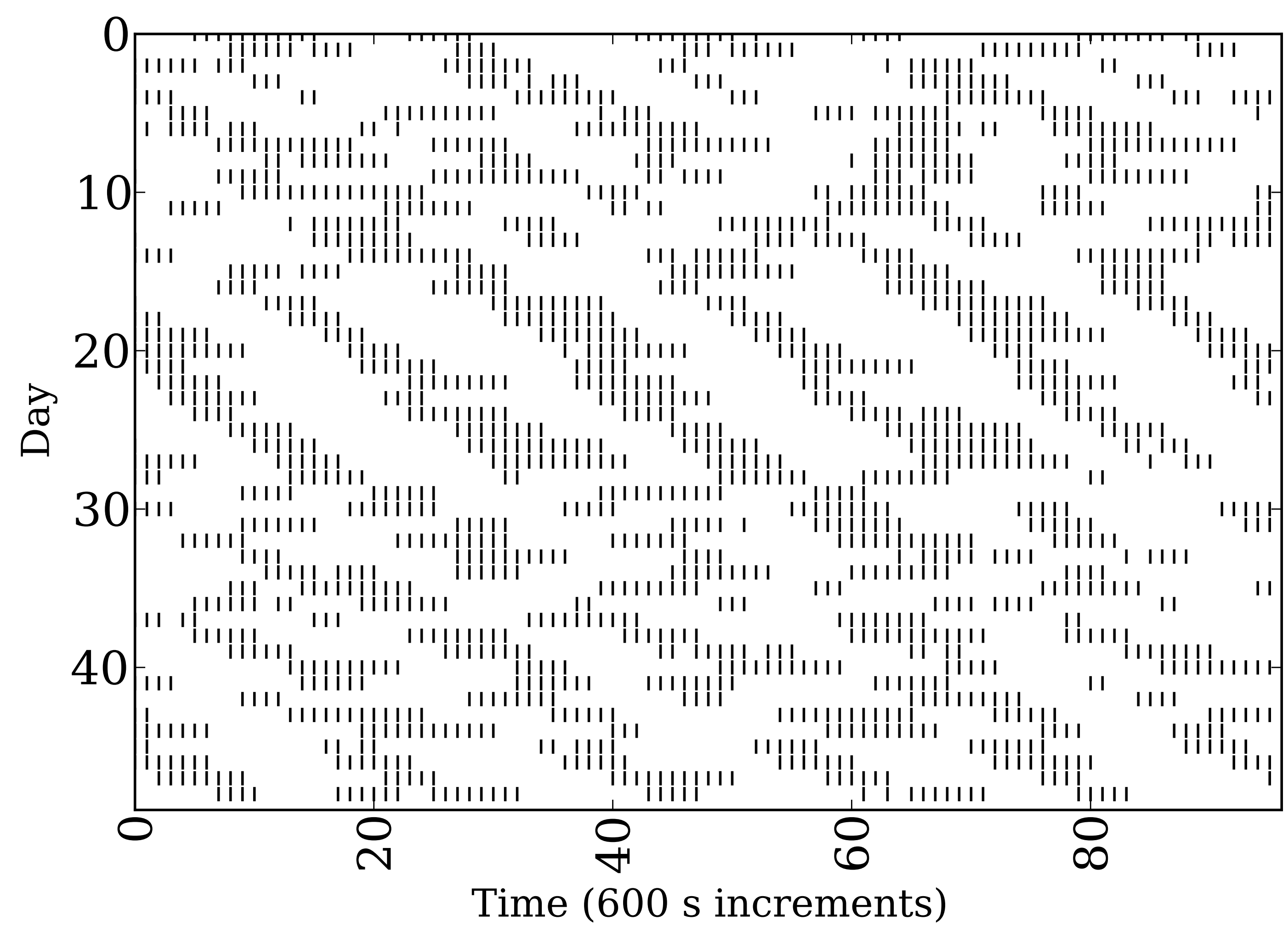
Sample Data for One User

Timestamp	Tweet Text
2013-08-22 12:54:06	Is Your Gmail Social? How to Use Gmail Daily to Build an Engaged Com
2013-08-22 13:11:22	Facebook's Embedded Posts Now Available to Everyone http://t.co/Cc0
2013-08-22 13:14:06	The Credible Hulk http://t.co/q17VrcSdBs
2013-08-22 13:29:02	25 Things You Didn't Know About Ninjas http://t.co/CcJz92sRyy
2013-08-22 13:32:59	Twitter Users: Revoke and Reestablish Third Party App Access Now http
2013-08-22 13:48:46	10 Brilliant Facebook Marketing Tactics to Increase Reach and Engageme
2013-08-22 14:17:11	Google Now Adds Cards for NCAA Football Scores, Concert Tickets, C
2013-08-22 15:18:03	What is the NSA Really Up To? [COMIC] http://t.co/hakoq1mFNX
2013-08-22 15:39:04	6 Things Every Good Business Blog MUST Have http://t.co/KRGzZRCrb

Predicting User Activity on Twitter



- Divide time into short intervals (600s = 10 minutes)
- Discretize activity: 1 if user tweeted in the interval, 0 if not
- Can we predict the user's activity?
- First approach: Start by trying to predict the user's activity based only on his/her past activity

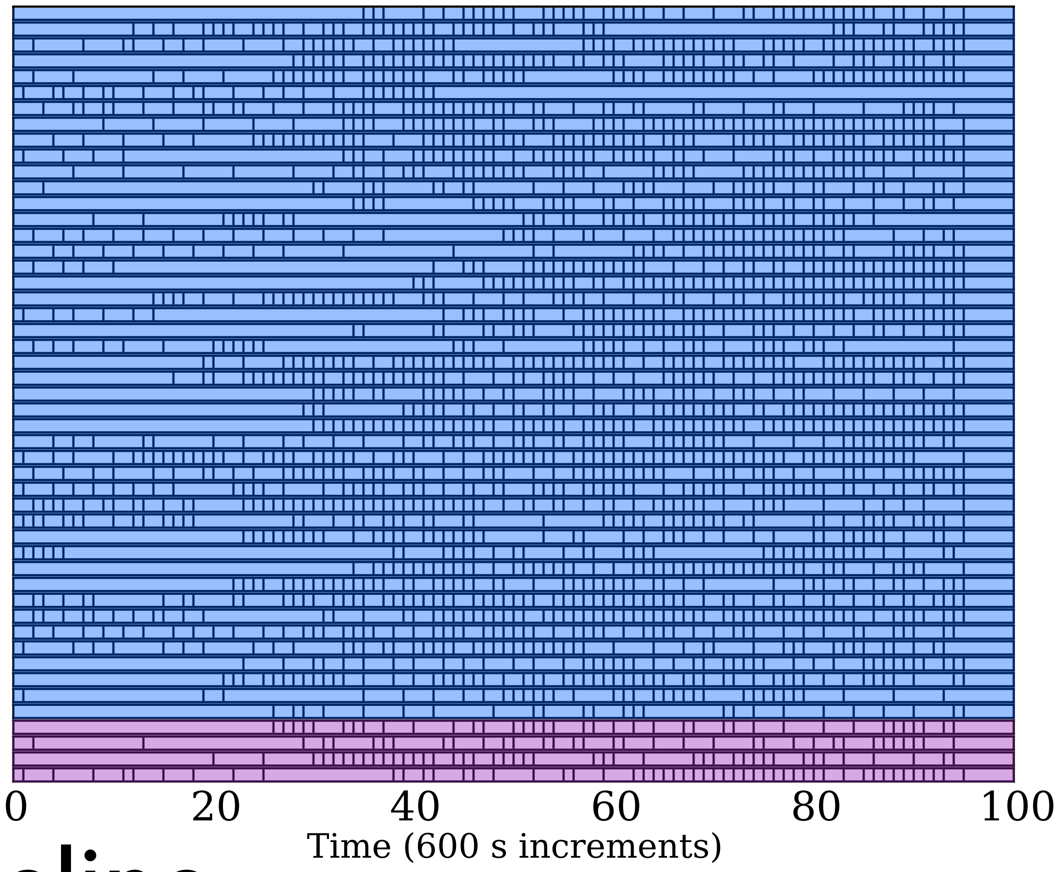


Compare Approaches

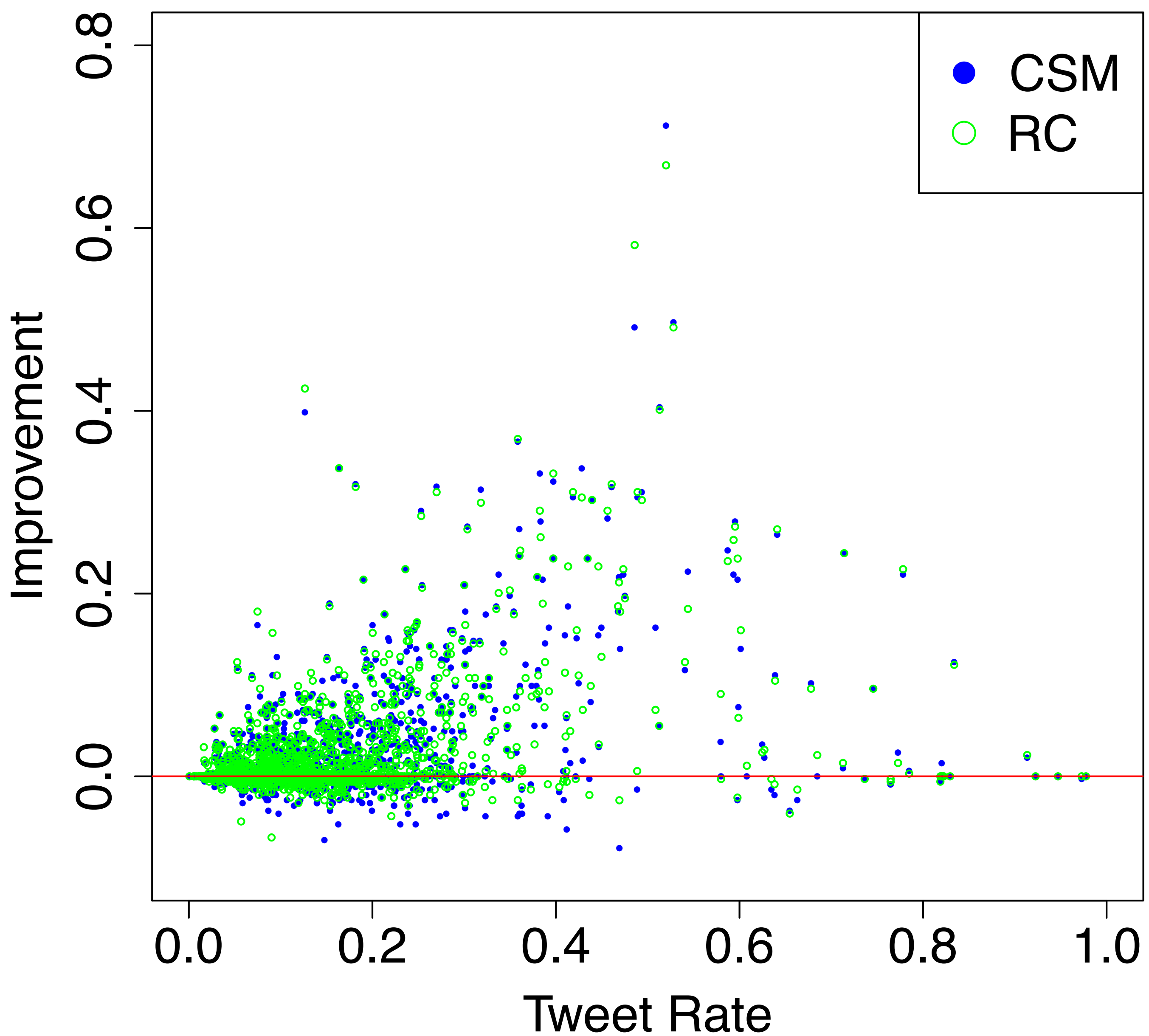
- **Reservoir Computing (RC)**
  - **“Top-down”**: Relaxed from very complicated dynamics until maximum predictive capability is reached
  - Black-box, hard to interpret
- **Causal State Modeling (CSM)** (similar to hidden markov modeling):
  - Looks at the user’s past N steps and determines probability of tweeting in the next time step, based on previous behavior
  - **“Bottom-up”**: Aims to construct the simplest predictive model
  - Easy to interpret

Predicting User Activity on Twitter

- Build model for each user separately
- Training: 45 days
- Testing: 4 days
- Look back 10 steps
- Predict ahead 1 step
- Compare to "majority vote" baseline



Comparing "**top-down**" (RC) and "**bottom-up**" (CSM) approaches



- Both methods showed strong improvements in prediction for a subset of users based on limited info (the user's own previous history)
- Average improvement of the two methods was about the same
- For some users, CSM did better, for others RC did better
- **Extensions:** we're working on prediction schemes that also take into account the activity patterns in the user's network
- **Challenge:** how can we combine approaches?

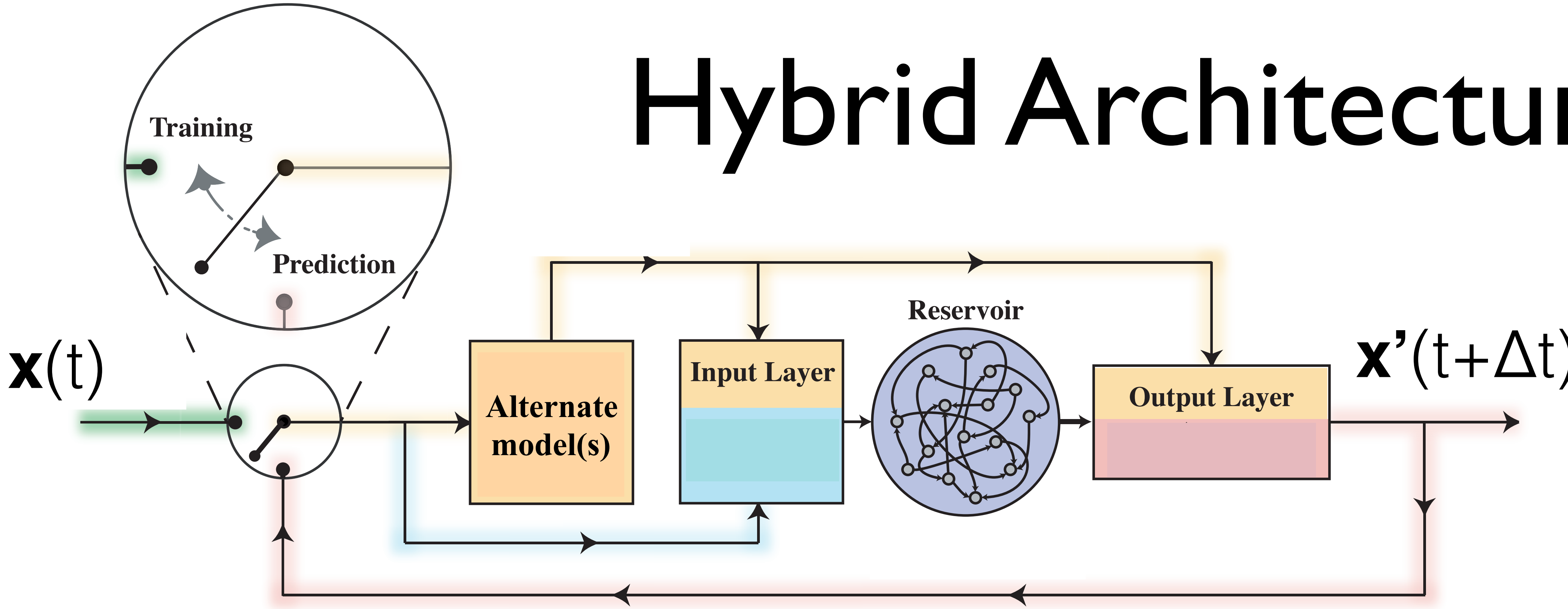
Ref: D. Darmon, J. Sylvester, M. Girvan, and W. Rand, "Predictability of User Behavior in Social Media: Bottom-Up v. Top-Down Modeling," International Conference on Social Computing (2013).

How do we effectively *combine* approaches?

- Sometimes we want to combine 2 or more data-driven approaches, like the "***top-down***" and"***bottom-up***" approaches

- Other times we have knowledge of the dynamics (e.g., in weather) and we want to combine ***knowledge-based*** models with ***knowledge-free*** (data-driven) models

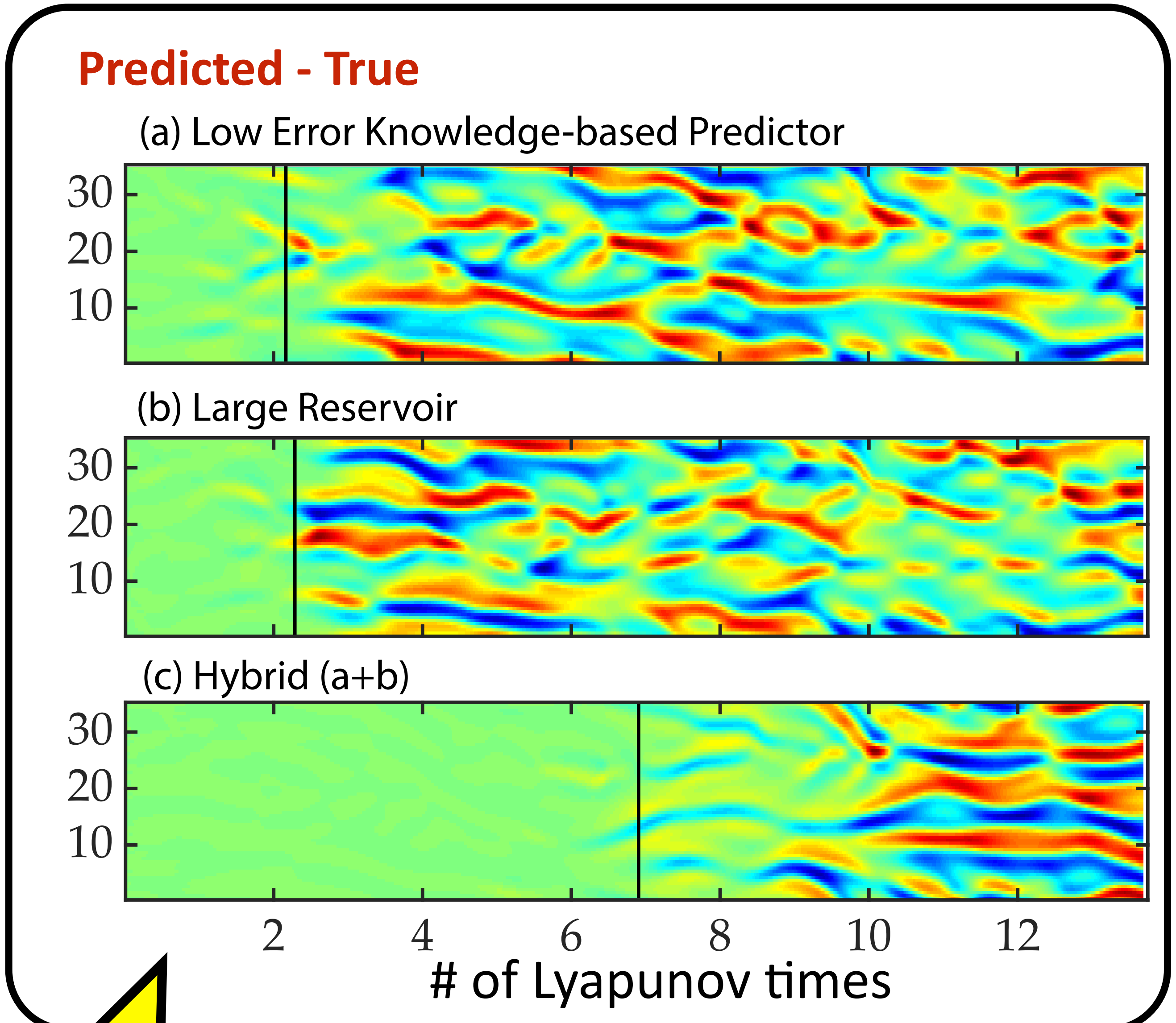
Hybrid Architecture



- Input is fed into both the alternate model(s) and the reservoir (through input layer)
- Output from the alternate model(s) is fed into the reservoir (through input layer) and the output layer
- Weights in the output layer are trained to maximize prediction in the training data.

Ref: J. Pathak, A. Wikner, R. Fussell, S. Chandra, B.R. Hunt, M. Girvan, and E. Ott, "Hybrid forecasting of chaotic processes: using machine learning in conjunction with a knowledge-based model." *Chaos,* 28, 041101 (2018).

Performance of the Hybrid Model for a Simple Test Case



KS System

$$y_t = -yy_x - y_{xx} - y_{xxxx}$$

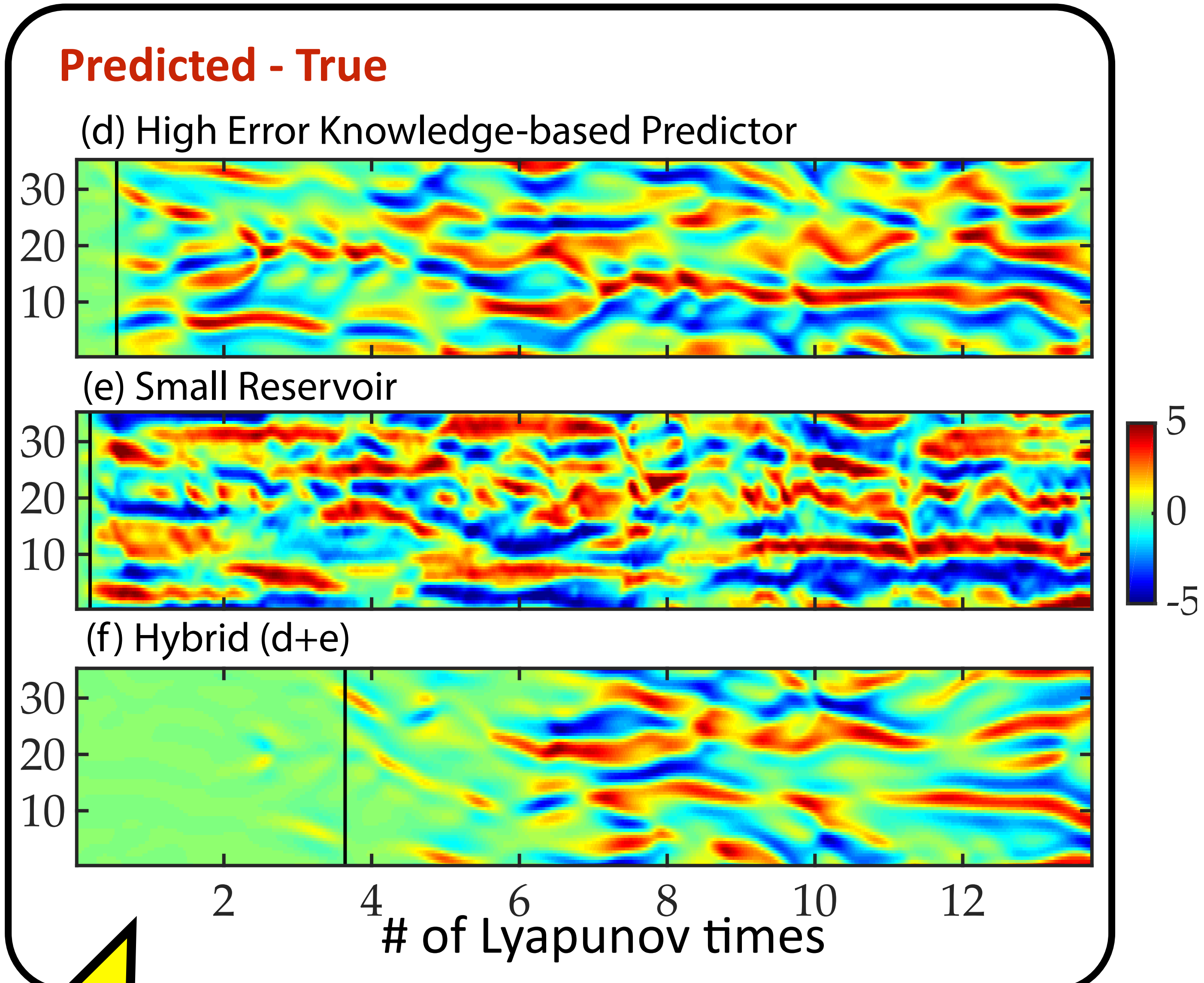
Imperfect "model"

$$y_t = -yy_x - (1+\epsilon)y_{xx} - y_{xxxx}$$

<u>Good model</u> + <u>large reservoir</u> ***combine*** to give <u>impressive predictions</u>

Ref: J. Pathak, A. Wikner, R. Fussell, S. Chandra, B.R. Hunt, M. Girvan, and E. Ott, "Hybrid forecasting of chaotic processes: using machine learning in conjunction with a knowledge-based model." *Chaos,* 28, 041101 (2018).

Performance of the Hybrid Model for a Simple Test Case

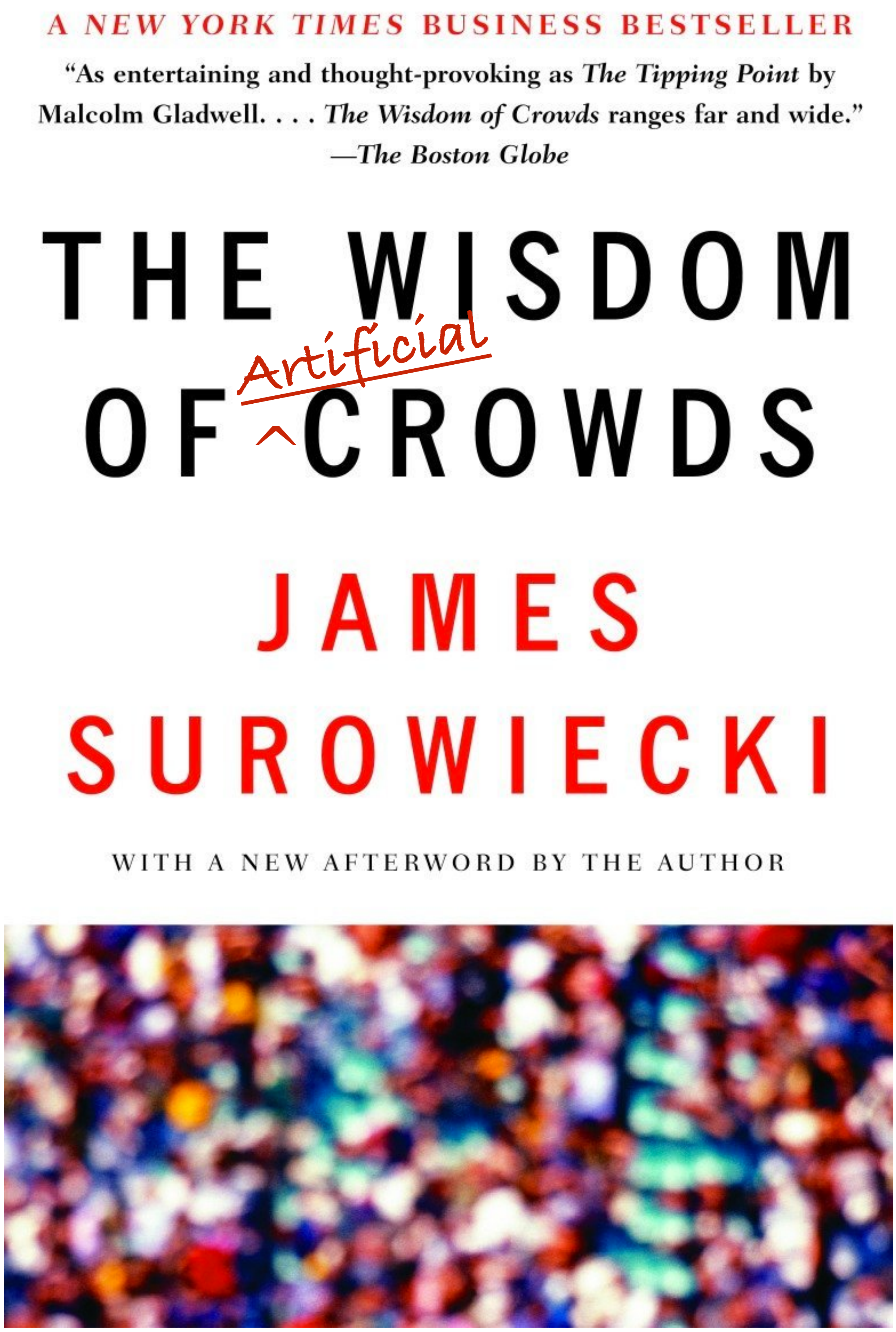


<u>Poor model</u> + <u>small reservoir</u> ***combine*** for <u>surprisingly good predictions</u>

Ref: J. Pathak, A. Wikner, R. Fussell, S. Chandra, B.R. Hunt, M. Girvan, and E. Ott, "Hybrid forecasting of chaotic processes: using machine learning in conjunction with a knowledge-based model." *Chaos,* 28, 041101 (2018).

Can we use the hybrid scheme to combine many ML approaches?

- **“Wisdom of crowds”**: notion that large groups of individuals make high quality collective judgements
- Requires individuals to reach decisions independently for quality judgements
- How does this notion apply to ensembles of machines?



Conclusions

- Reservoir Computing (RC) provides a promising method for predicting complex time series, harnessing the chaos and complexity within the reservoir
- A parallel RC scheme makes prediction of very large systems feasible
- A hybrid scheme that couple alternate forecasting with the reservoir offers further, often dramatic, improvements
- Hybrid method may help reveal the strengths and weaknesses of ML and alternate models, leading to improvements in both
- Open and exciting questions remain about how to scale this to ensembles and layers of coupled machines

Acknowledgements

Students

- Jaideep Pathak
- Sarthak Chandra
- Sanjukta Krisnagopal
- Alex Wikner
- Rebeckah Fussell
- Zhixin Lu (now a postdoc at Penn)
- David Darmon (now on the faculty at Monmouth U)

Faculty

- Ed Ott, UMD Physics and Electrical Engineering
- Brian Hunt, UMD Mathematics
- Jim Reggia, UMD Computer Science
- Yiannis Aloimonos, UMD Computer Science
- Garrett Katz, Syracuse Computer Science
- Istvan Szunyogh, Texas A&M Atmospheric Sciences
- Bill Rand, NC State Business School

References

- Pathak, Wikner, Fussell, Chandra, Hunt, Girvan, and Ott. "Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model." *Chaos* (2018)

- Pathak, Hunt, Girvan, Lu, and Ott. "Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach." *PRL* (2018).

- Pathak, Lu, Hunt, Girvan, and Ott. "Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data." *Chaos* (2017)

- Krishnagopal, Aloimonos, and Girvan. "Similarity Learning and Generalization with Limited Data: A Reservoir Computing Approach." C*omplexity* (2018).

- Darmon, Girvan, and Rand. "Predictability of user behavior in social media: Bottom-up v. top-down modeling." *IEEE International Conference on Social Computing* (2013).

- Darmon, Rand, and Girvan. "Computational landscape of user behavior on social media." *PRE* (2018).