

EXPLORING THE ABILITY OF RESERVOIR COMPUTERS TO LEARN CHAOTIC SYSTEMS

Joseph Wilkes
Brigham Young University

Thank You Team

Joseph Wilkes

joseph.benson.wilkes@gmail.com

BYU



DJ Passey

dipassey@unc.edu

UNC



Benjamin Webb

bwebb@mathematics.byu.edu

BYU



Joseph Jamieson

joebobjamieson@gmail.com

BYU



BYU

Feel free to Email Us

Joseph Wilkes

joseph.benson.wilkes@gmail.com

BYU



DJ Passey

dipassey@unc.edu

UNC



Benjamin Webb

bwebb@mathematics.byu.edu

BYU



Joseph Jamieson

joebobjamieson@gmail.com

BYU



BYU

Thank you Sponsors



National Science Foundation
WHERE DISCOVERIES BEGIN

Overview

- Lorenz Attractor
- Reservoir Computer Model
- Network Thinning
- Spectral Radius
- Future Work
- Questions

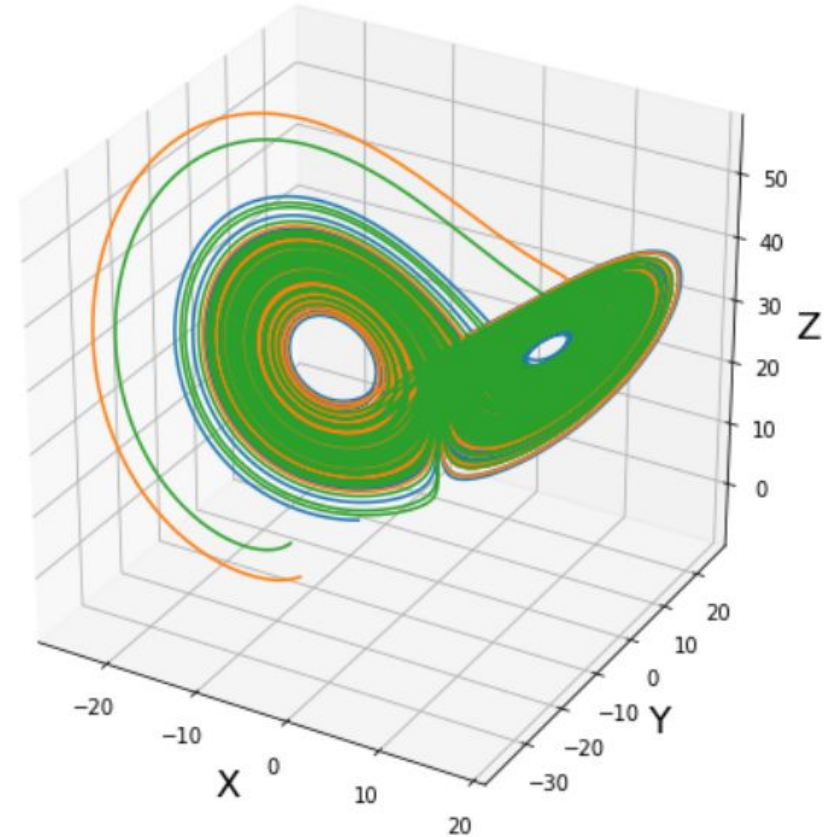
PREDICTING THE UNPREDICTABLE

Lorenz Attractor - Equations

$$dx/dt = 10(y - x)$$

$$dy/dt = x(28 - z) - y$$

$$dz/dt = xy - 8z/3$$



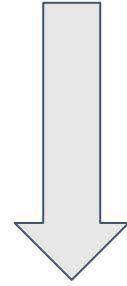
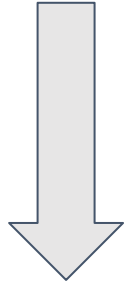
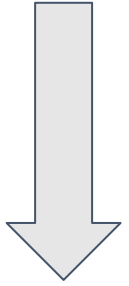
Reservoir Computer Model

$$\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{u}(t))]$$

$$\mathbf{W}_{\text{out}} = \operatorname{argmin}_{\mathbf{W} \in \mathbb{R}^{m \times n}} \|\mathbf{W}\mathbf{r}(t) - \mathbf{u}(t)\| + \alpha \|\mathbf{W}\|^2$$

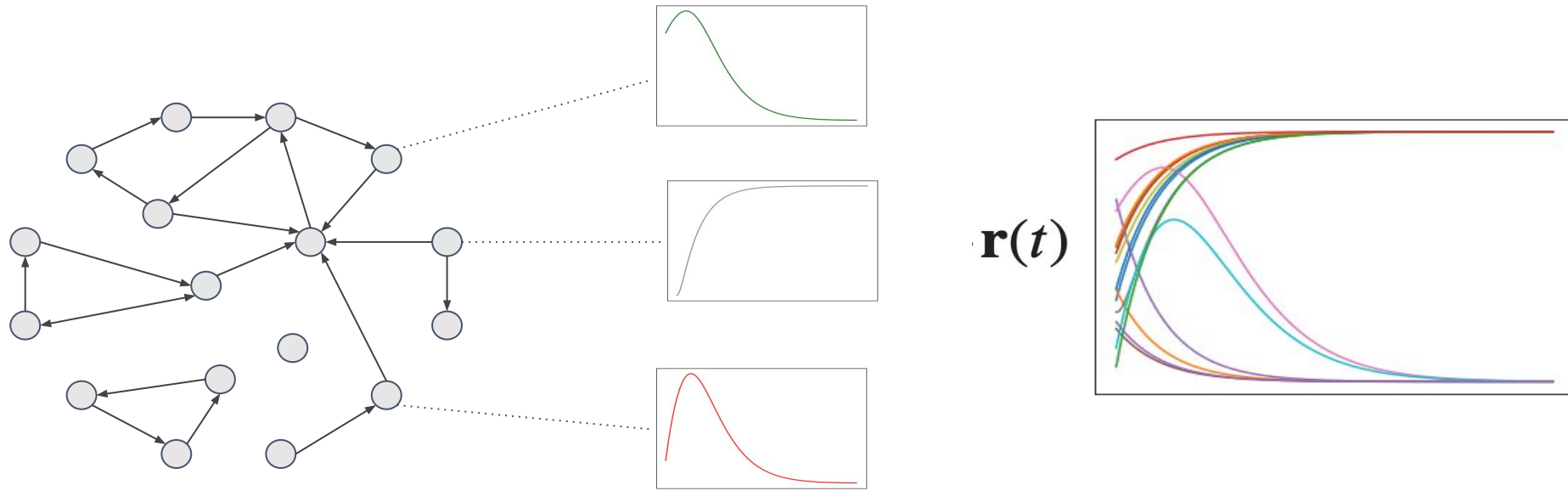
$$\frac{d}{dt}\hat{\mathbf{r}}(t) = \gamma[-\hat{\mathbf{r}}(t) + \tanh(\mathbf{A}\hat{\mathbf{r}}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{W}_{\text{out}}\hat{\mathbf{r}}(t))]$$

Reservoir Computer Model



$$\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t))]$$

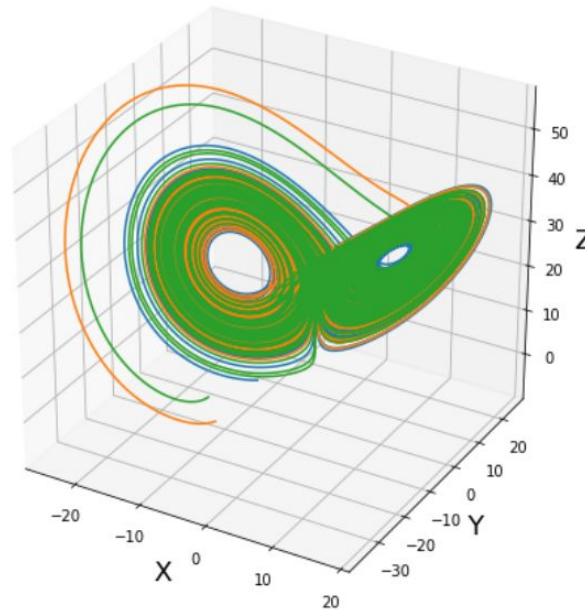
Reservoir Computer Model



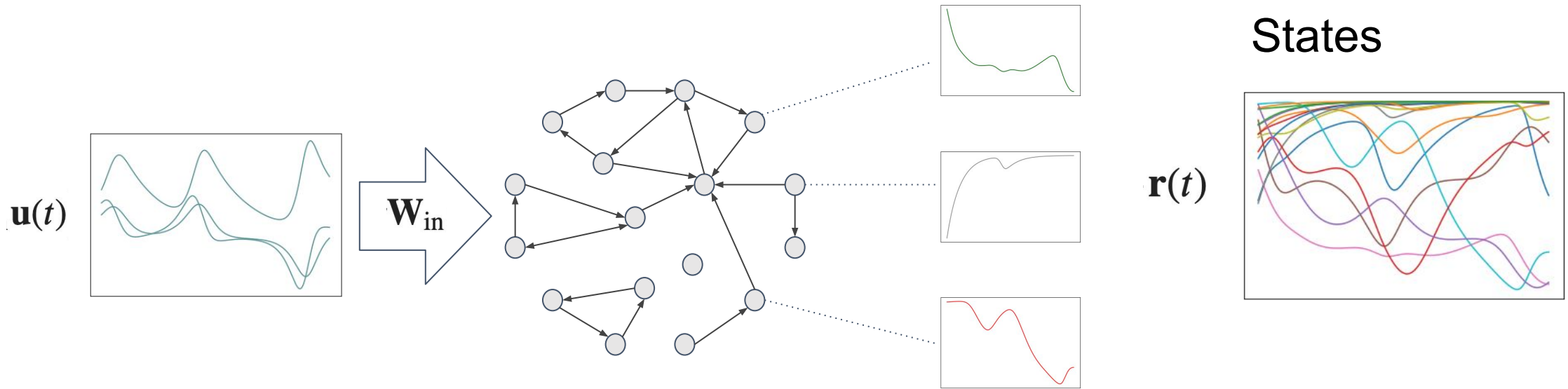
$$\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t))]$$

Reservoir Computer Model

$$\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{u}(t))]$$



Reservoir Computer Model



$$\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t) + \sigma\mathbf{W}_{in}\mathbf{u}(t))]$$

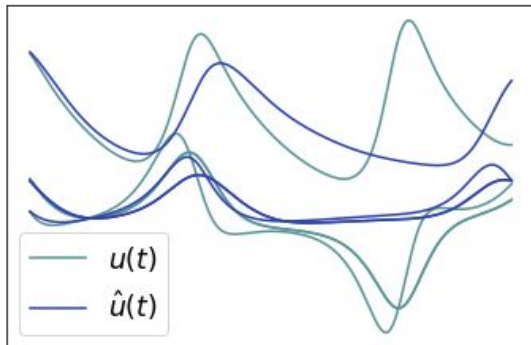


Reservoir Computer Model

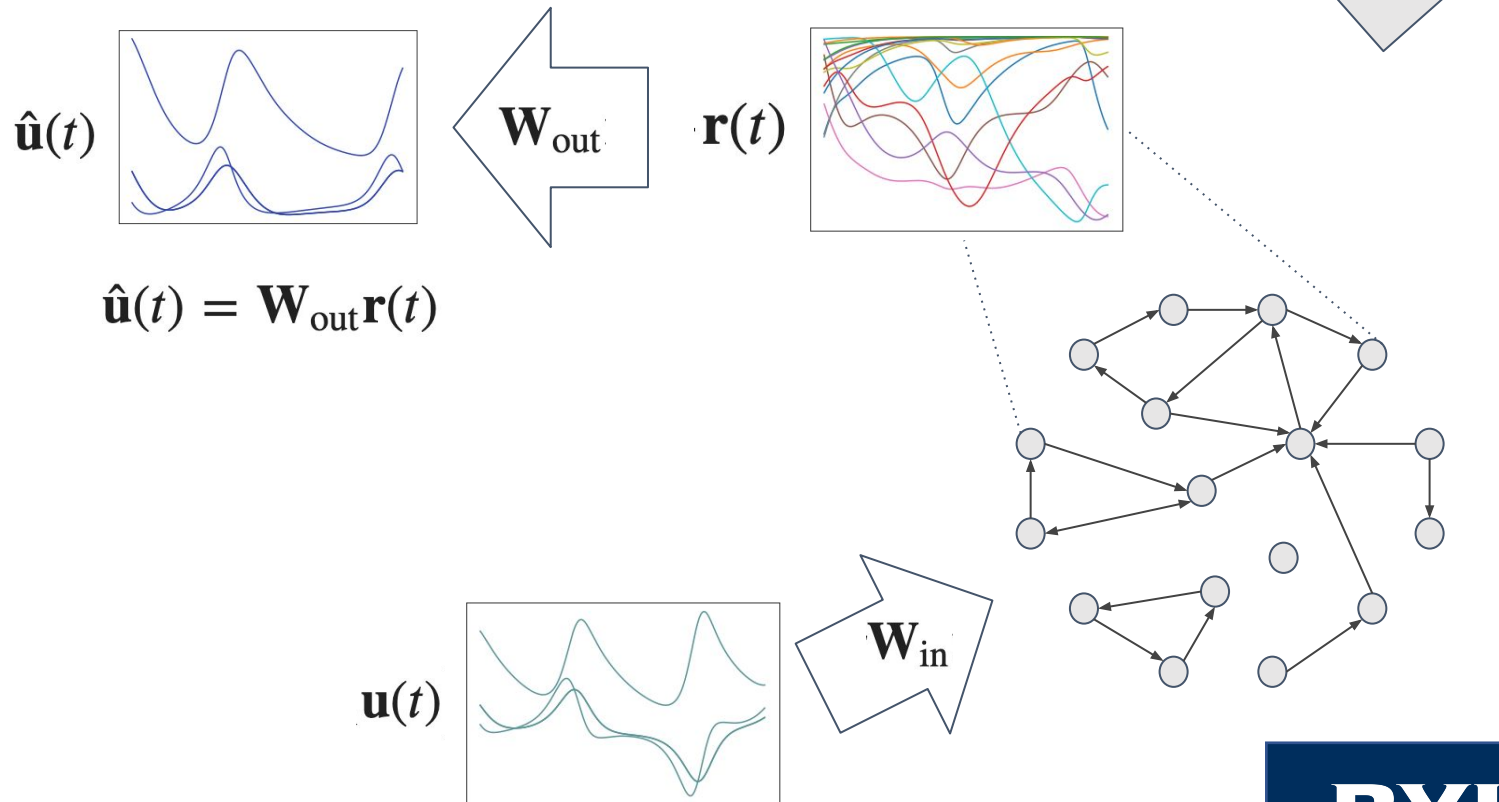
$$\mathbf{W}_{\text{out}} = \operatorname{argmin}_{\mathbf{W} \in \mathbb{R}^{m \times n}} \|\mathbf{W}\mathbf{r}(t) - \mathbf{u}(t)\| + \alpha \|\mathbf{W}\|^2$$

$$\frac{d}{dt}\hat{\mathbf{r}}(t) = \gamma[-\hat{\mathbf{r}}(t) + \tanh(\mathbf{A}\hat{\mathbf{r}}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{W}_{\text{out}}\hat{\mathbf{r}}(t))]$$

Learning Step

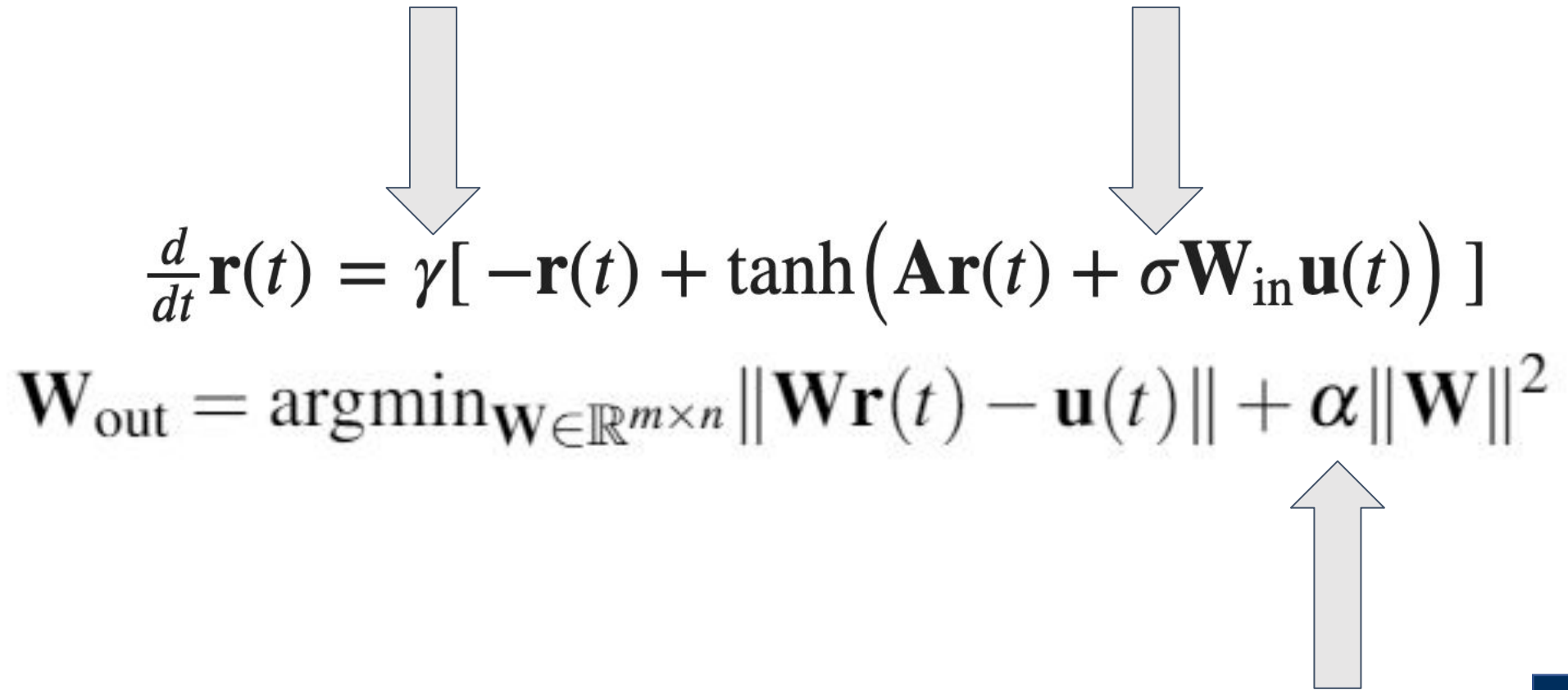


We solve for $\mathbf{W}_{\text{out}}$ so
 $\hat{u}(t)$ approximates $u(t)$



MAIN HYPOTHESIS

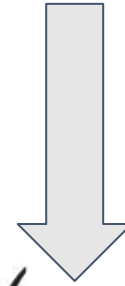
Reservoir Computer Model



The diagram illustrates the Reservoir Computer Model with two equations. The first equation, $\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{u}(t))]$, is preceded by a large downward arrow. The second equation, $\mathbf{W}_{\text{out}} = \operatorname{argmin}_{\mathbf{W} \in \mathbb{R}^{m \times n}} \|\mathbf{W}\mathbf{r}(t) - \mathbf{u}(t)\| + \alpha \|\mathbf{W}\|^2$, is preceded by a large upward arrow. The arrows indicate that the reservoir state $\mathbf{r}(t)$ is used to compute the output $\mathbf{W}_{\text{out}}$.

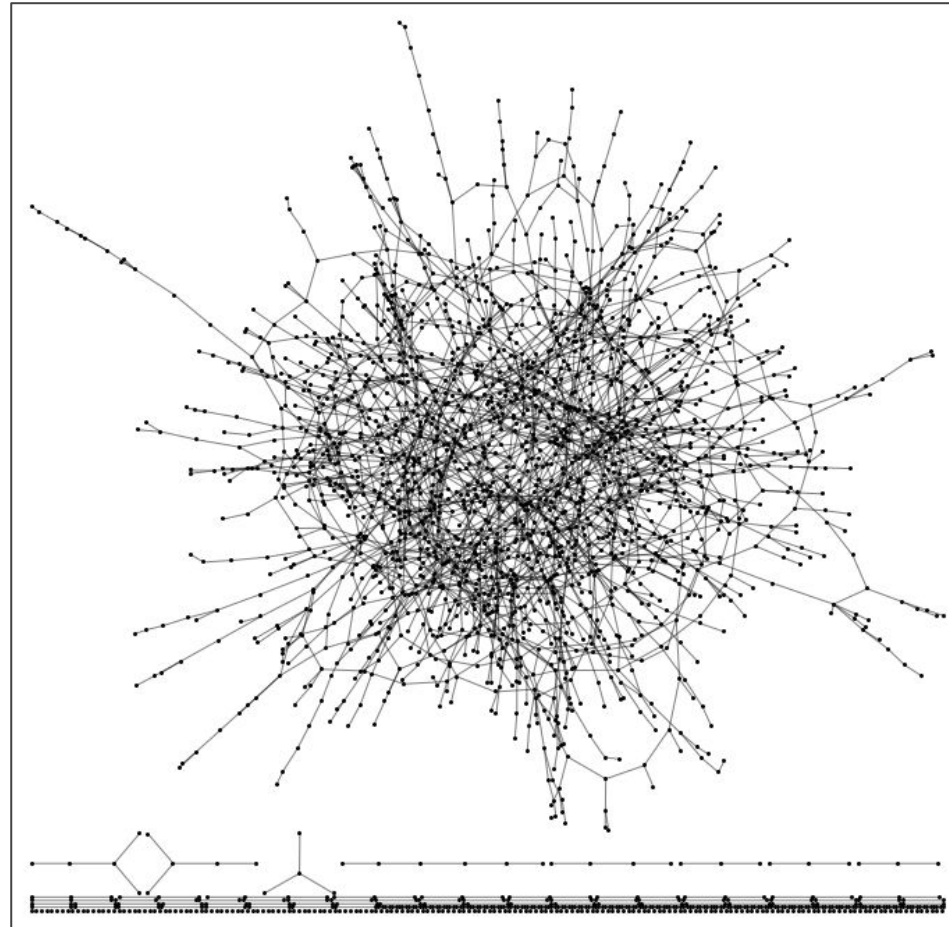
$$\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{u}(t))]$$
$$\mathbf{W}_{\text{out}} = \operatorname{argmin}_{\mathbf{W} \in \mathbb{R}^{m \times n}} \|\mathbf{W}\mathbf{r}(t) - \mathbf{u}(t)\| + \alpha \|\mathbf{W}\|^2$$

Reservoir Computer Model



$$\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{u}(t))]$$

Internal Processing Network

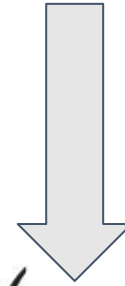


Reservoir Computer Model

Size of Network (Number of Nodes)

Topology (Network Structure)

Spectral Radius

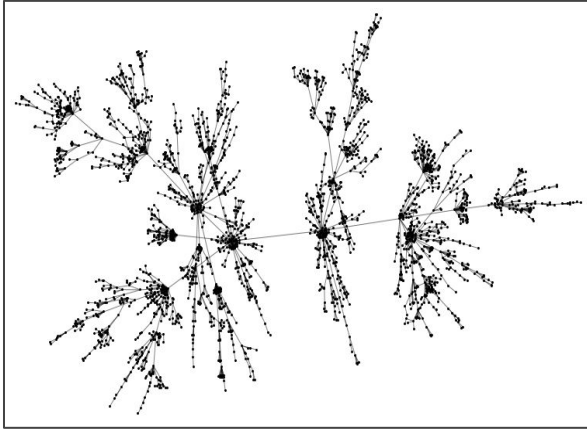


$$\frac{d}{dt}\mathbf{r}(t) = \gamma[-\mathbf{r}(t) + \tanh(\mathbf{A}\mathbf{r}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{u}(t))]$$

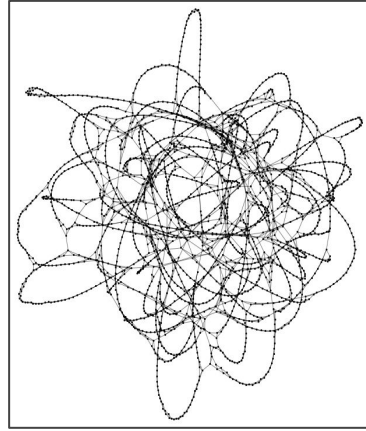
Initial Mean Degree (connectivity)

Sparsity

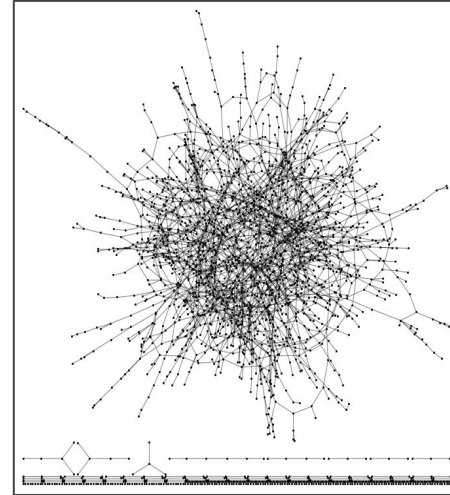
Network Topologies



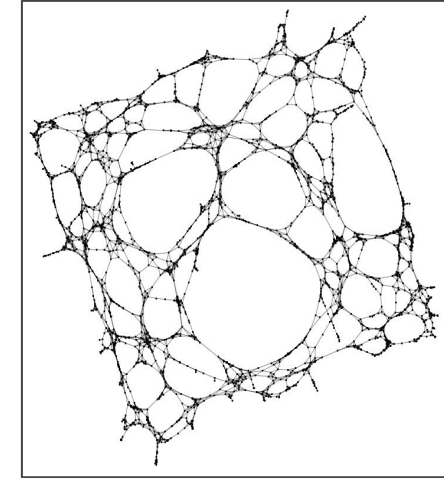
Albert-Barabási



Watts-Strougal



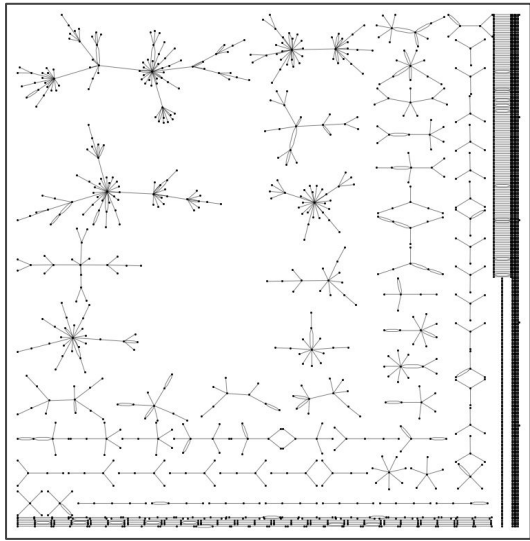
Erdős-Rényi



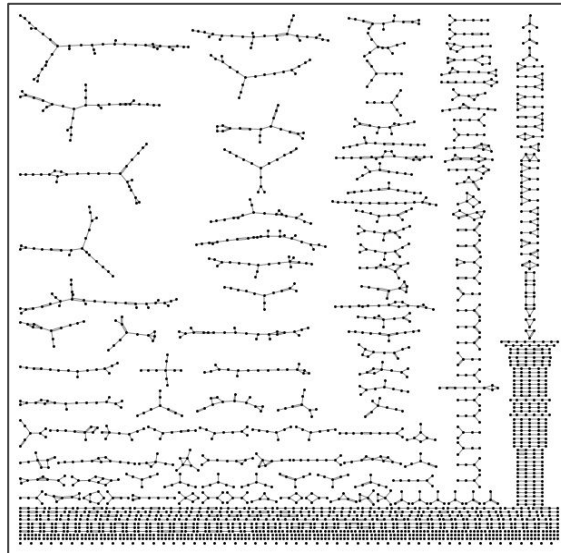
Random
Geometric
Graph

MAIN IDEA

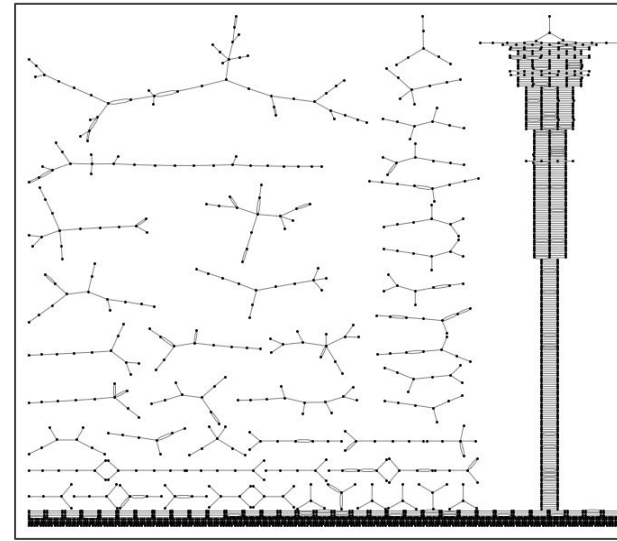
Randomly Remove Edges (a lot of them)



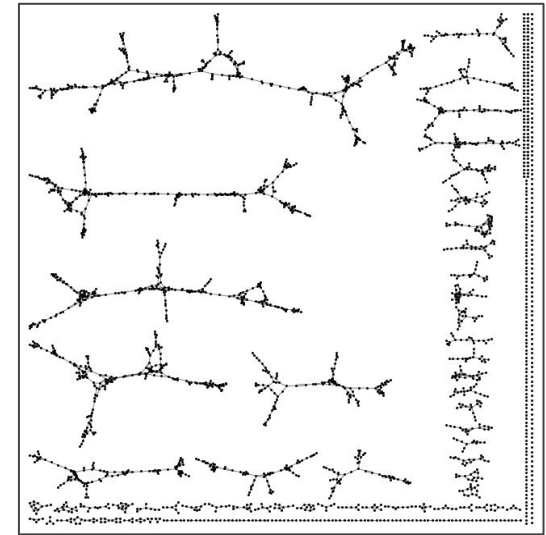
Albert-Barabási



Watts–Strogatz

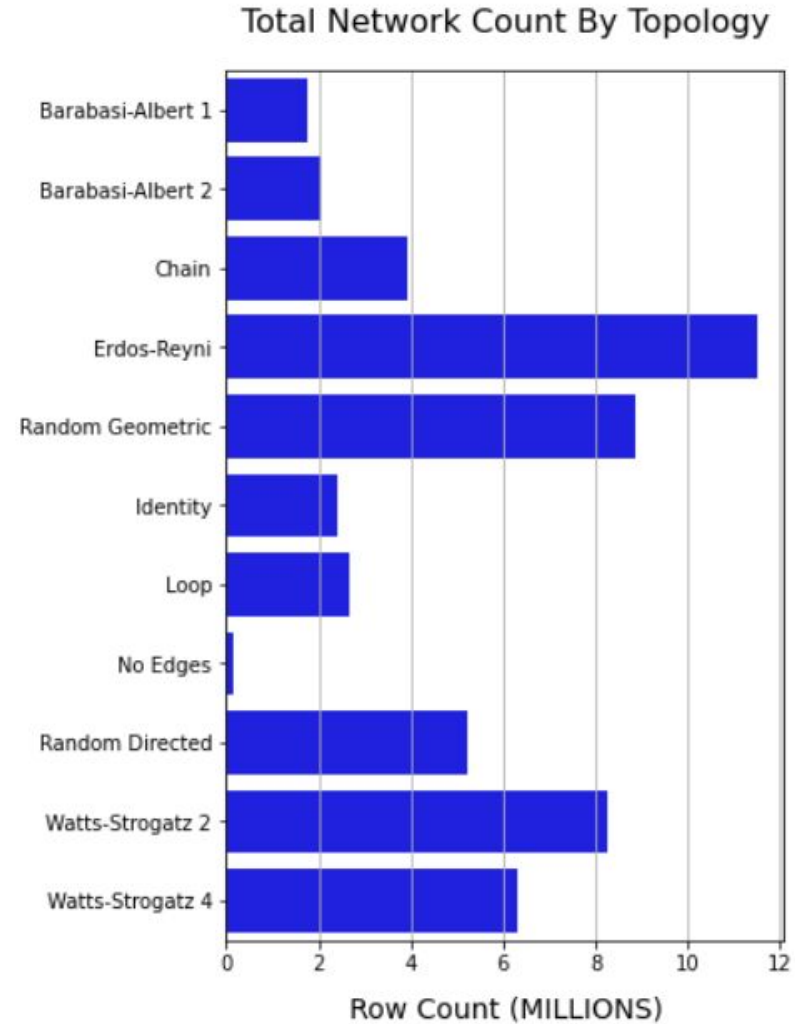


Erdős–Rényi



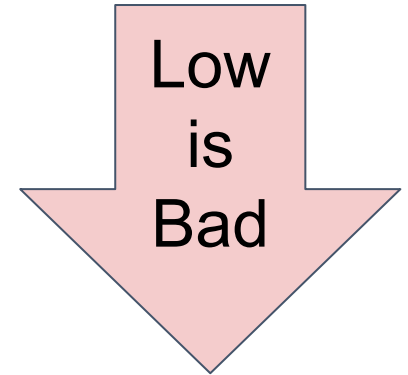
Random Geometric Graph

Data Generation

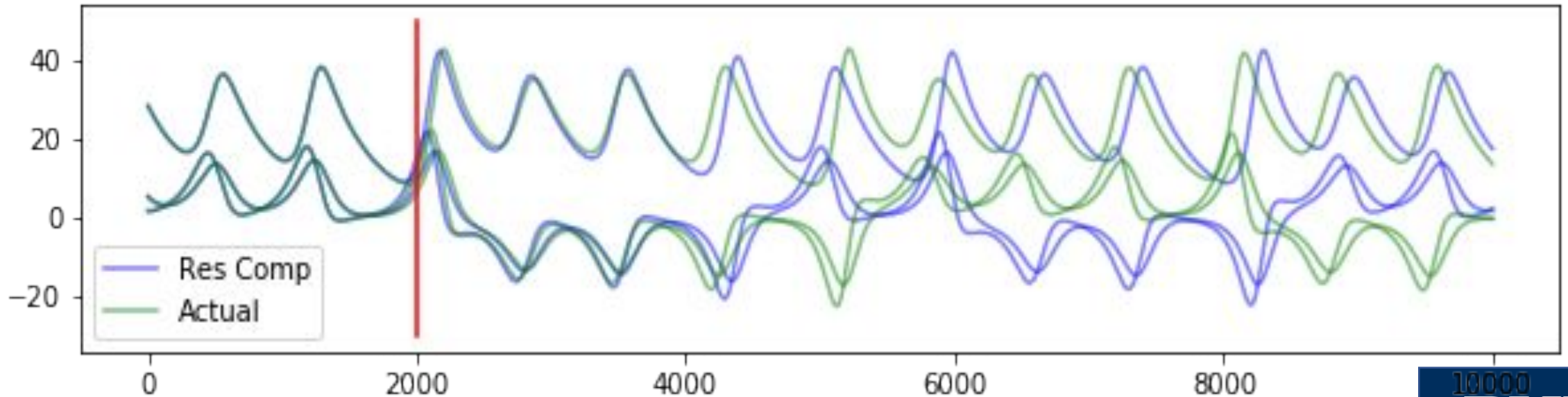


Performance Metric

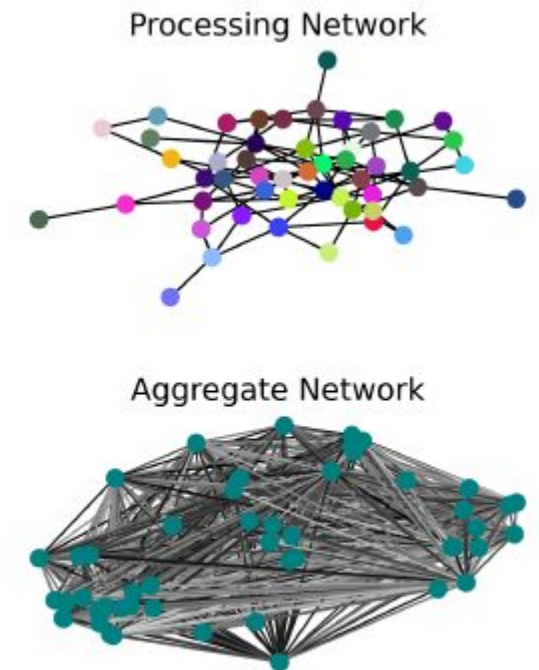
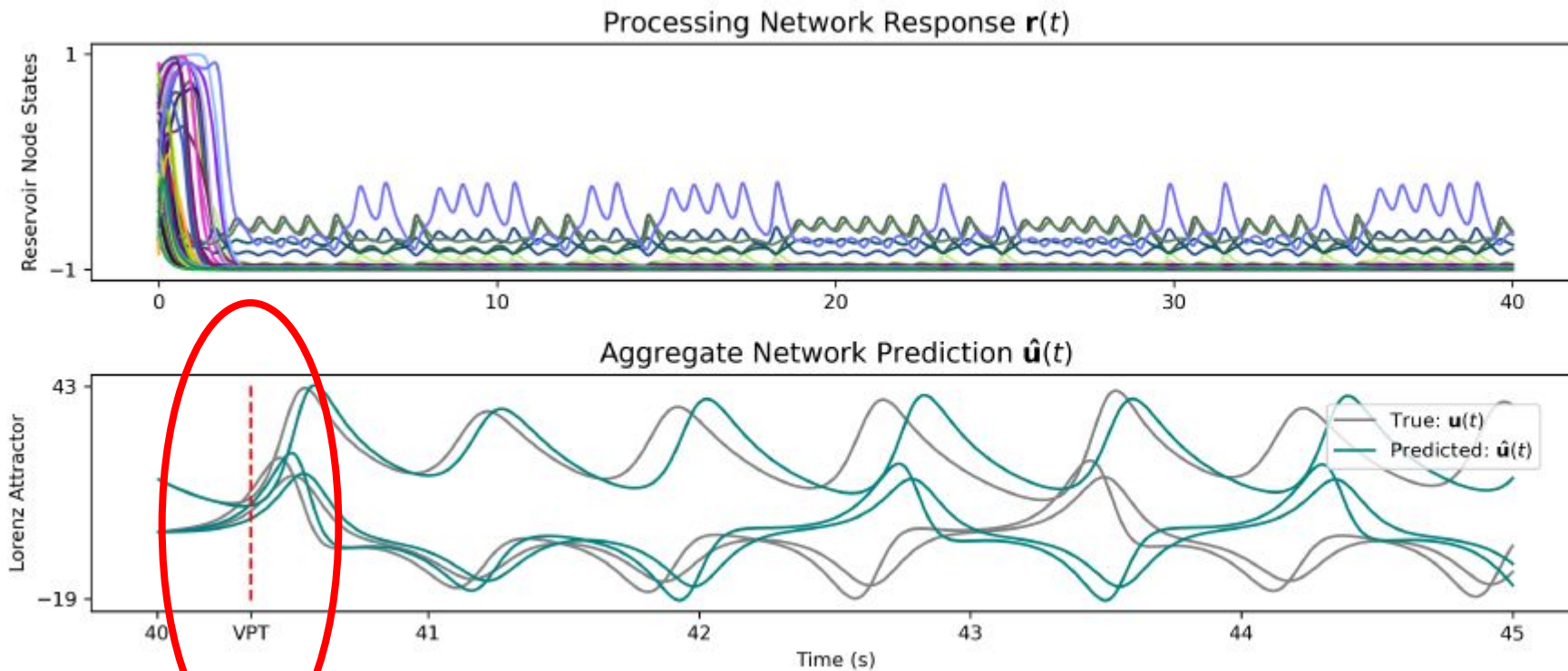
$$\max_{t^* \in [0, T]} \{t^* : \|\tilde{\mathbf{u}}(t) - \mathbf{u}(t)\|_2 < 5, \forall t \in [0, t^*]\}$$



Model Fit to Test Data

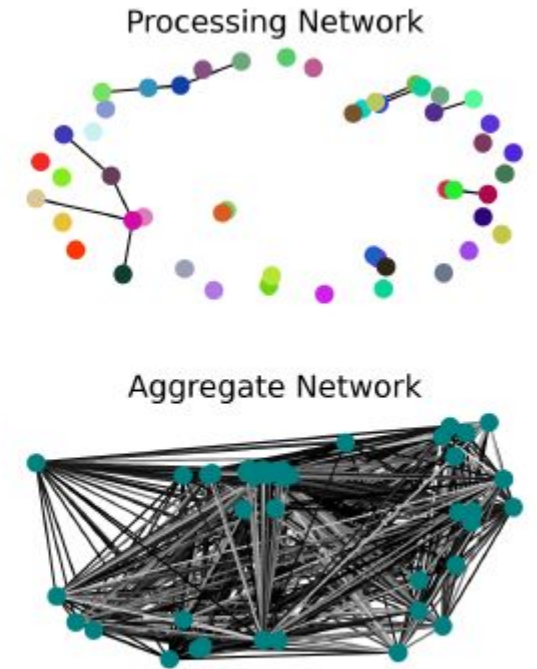
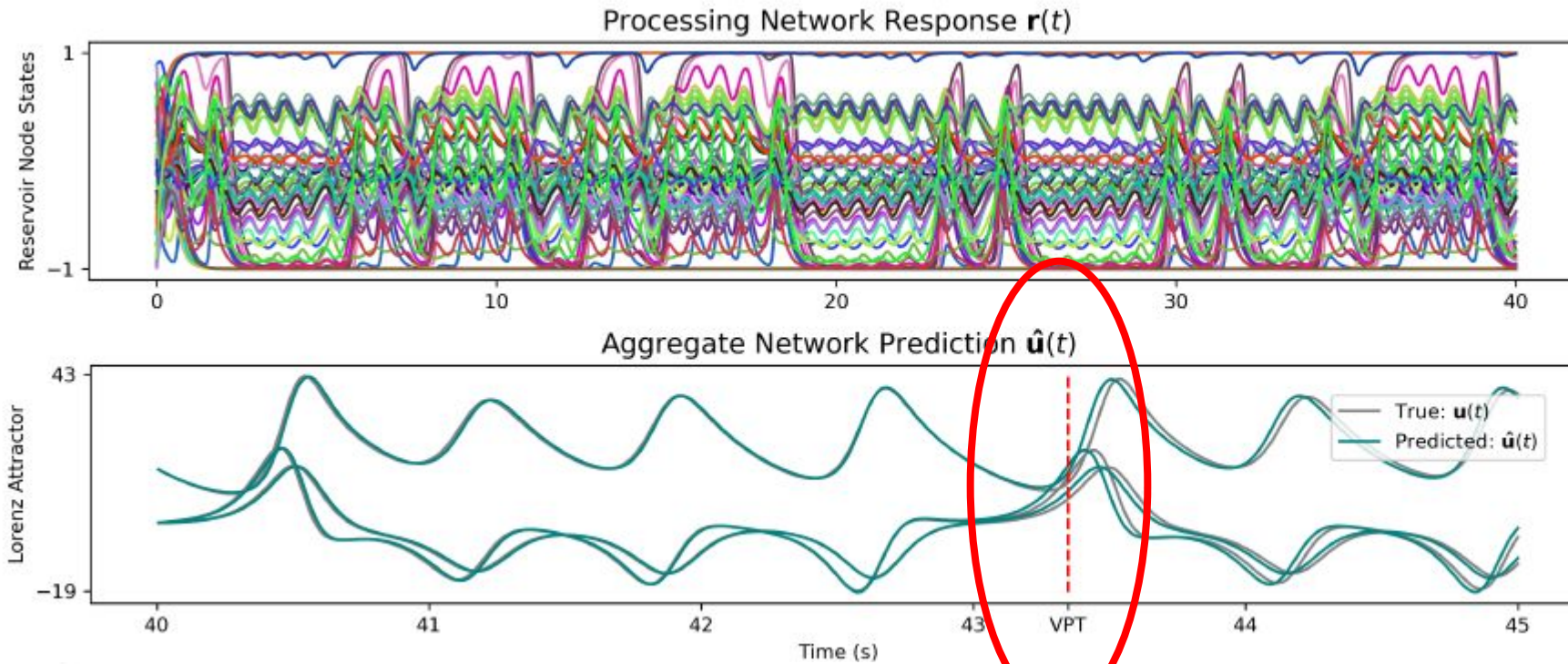


Before thinning the Processing Network



$$\frac{d}{dt}\hat{\mathbf{r}}(t) = \gamma[-\hat{\mathbf{r}}(t) + \tanh(\underbrace{\mathbf{A}\hat{\mathbf{r}}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{W}_{\text{out}}\hat{\mathbf{r}}(t)}_{\text{}})]$$

After removing 90% of the connections

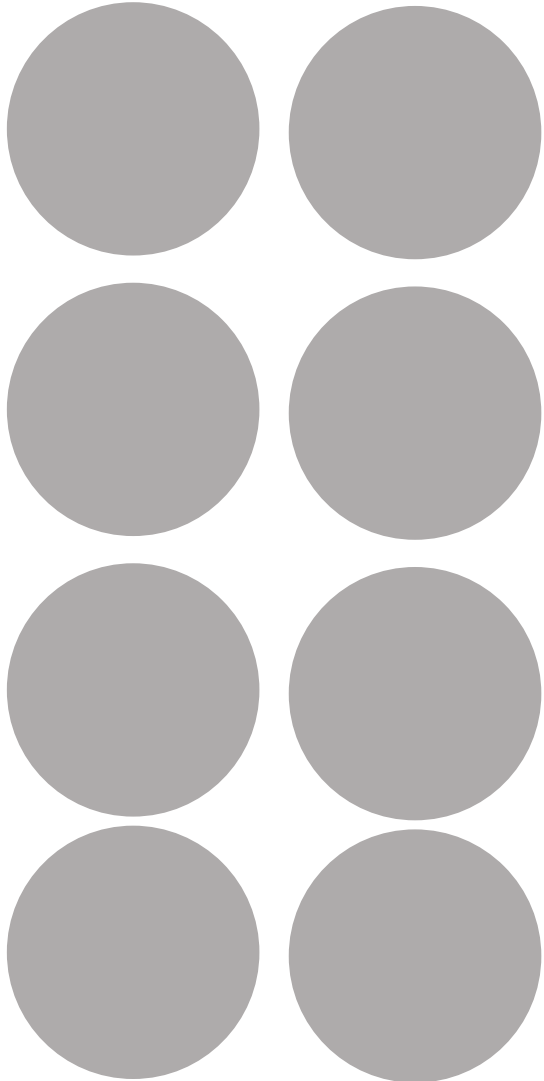


$$\frac{d}{dt}\hat{\mathbf{r}}(t) = \gamma[-\hat{\mathbf{r}}(t) + \tanh(\underbrace{\mathbf{A}\hat{\mathbf{r}}(t) + \sigma\mathbf{W}_{\text{in}}\mathbf{W}_{\text{out}}\hat{\mathbf{r}}(t)}_{\text{}})]$$

Extreme Thinning of Network Summary

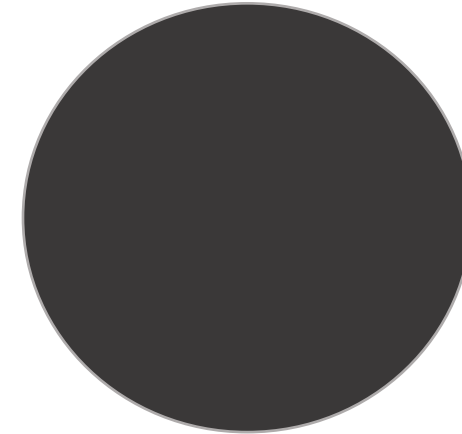
The data points suggest that there exist parameter sets where thinning the internal network of the Reservoir Computer can improve the valid prediction time (or accuracy duration) on the Lorenz Attractor.

Crowd Sourcing



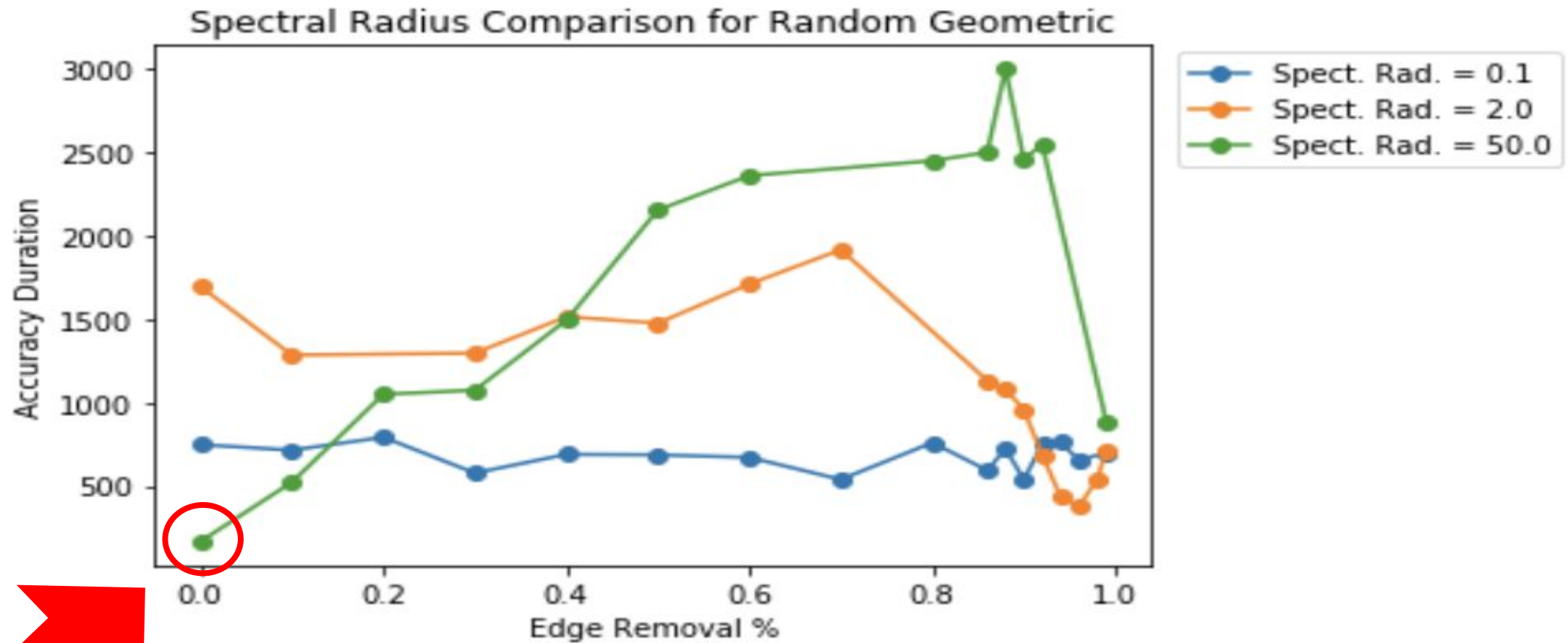
Many amateurs can be greater than One Professional / Expert

VS



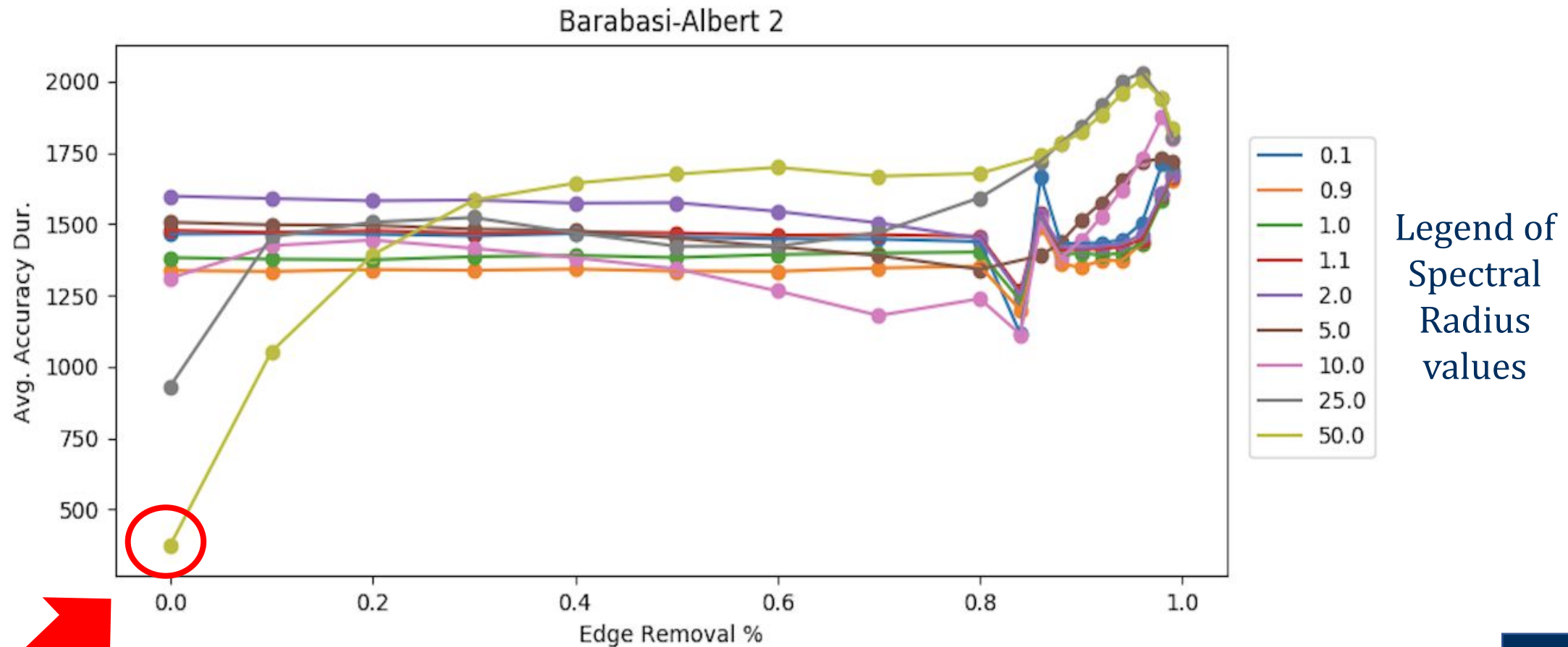
SPECTRAL RADIUS

Thinning & Spectral Radius



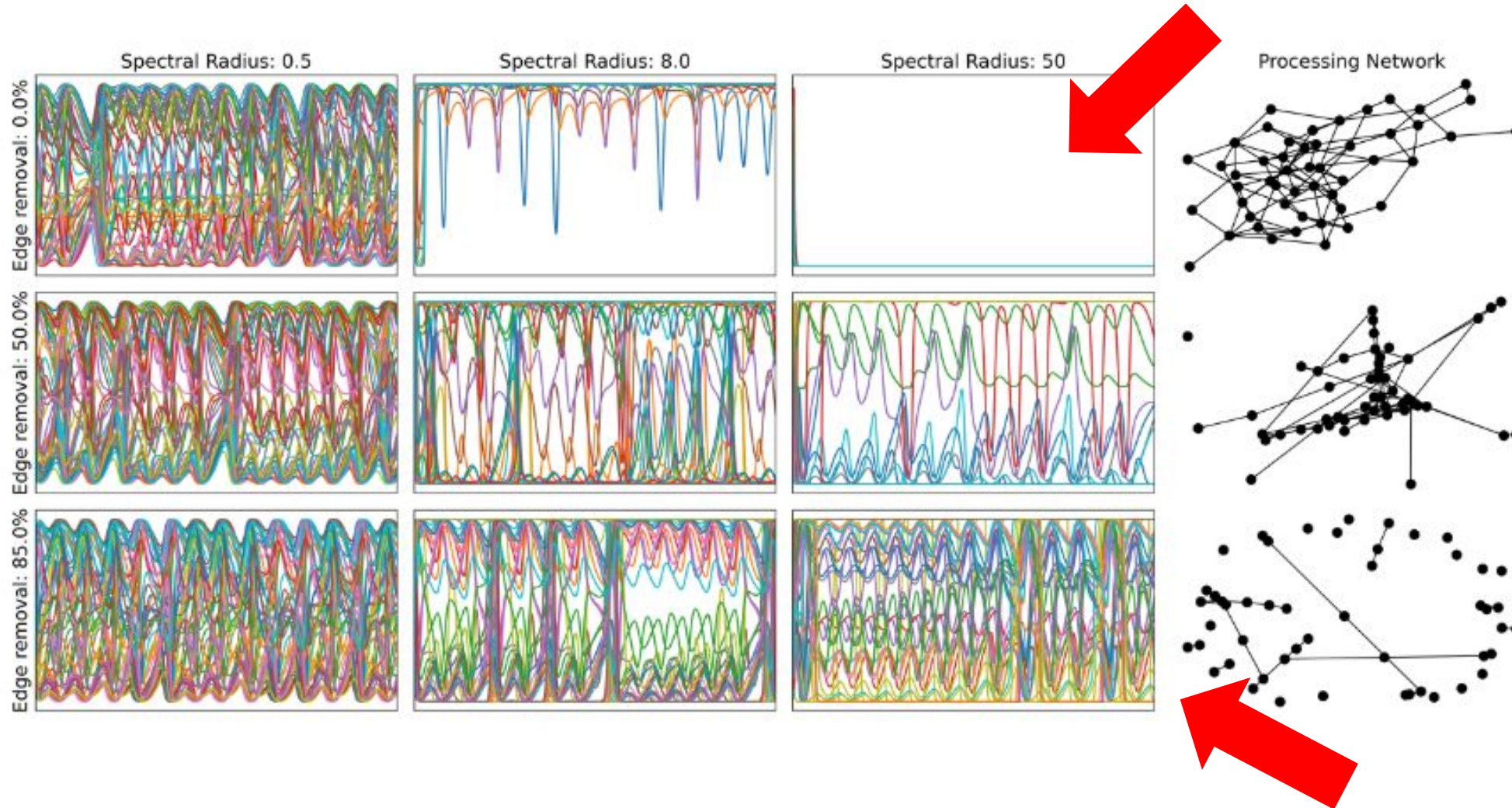
* Individual Parameter sets are shown
(no aggregation across other parameters like gamma, sigma, etc)

Thinning & Spectral Radius



* Aggregating tens of thousands of simulations across many parameter sets

Node State Responsiveness according to Thinning & Spectral Radius



Thinning & Spectral Radius Summary

The data points suggest that Reservoir computers with dense internal networks and a high spectral radius can be really bad predictors whereas extremely sparse networks with high spectral radius can be the best of predictors.

Future Research

*Processing
Network*

$$\frac{d}{dt}\hat{\mathbf{r}}(t) = \gamma[-\hat{\mathbf{r}}(t) + \tanh(\underbrace{\mathbf{A}\hat{\mathbf{r}}(t) + \boldsymbol{\sigma}\mathbf{W}_{\text{in}}\mathbf{W}_{\text{out}}\hat{\mathbf{r}}(t)}_{\text{Aggregate Network or Effective Adjacency Matrix}})]$$

*Aggregate Network or
Effective Adjacency Matrix*

Optional Questions

- How do our results compare to other researchers?
- What sources of bias could there be in our results?
- How does topology affect predictability?
- Pro's & Con's of Reservoir Computing

APPENDIX

See also



Benjamin Webb
bwebb@mathematics.byu.edu
BYU

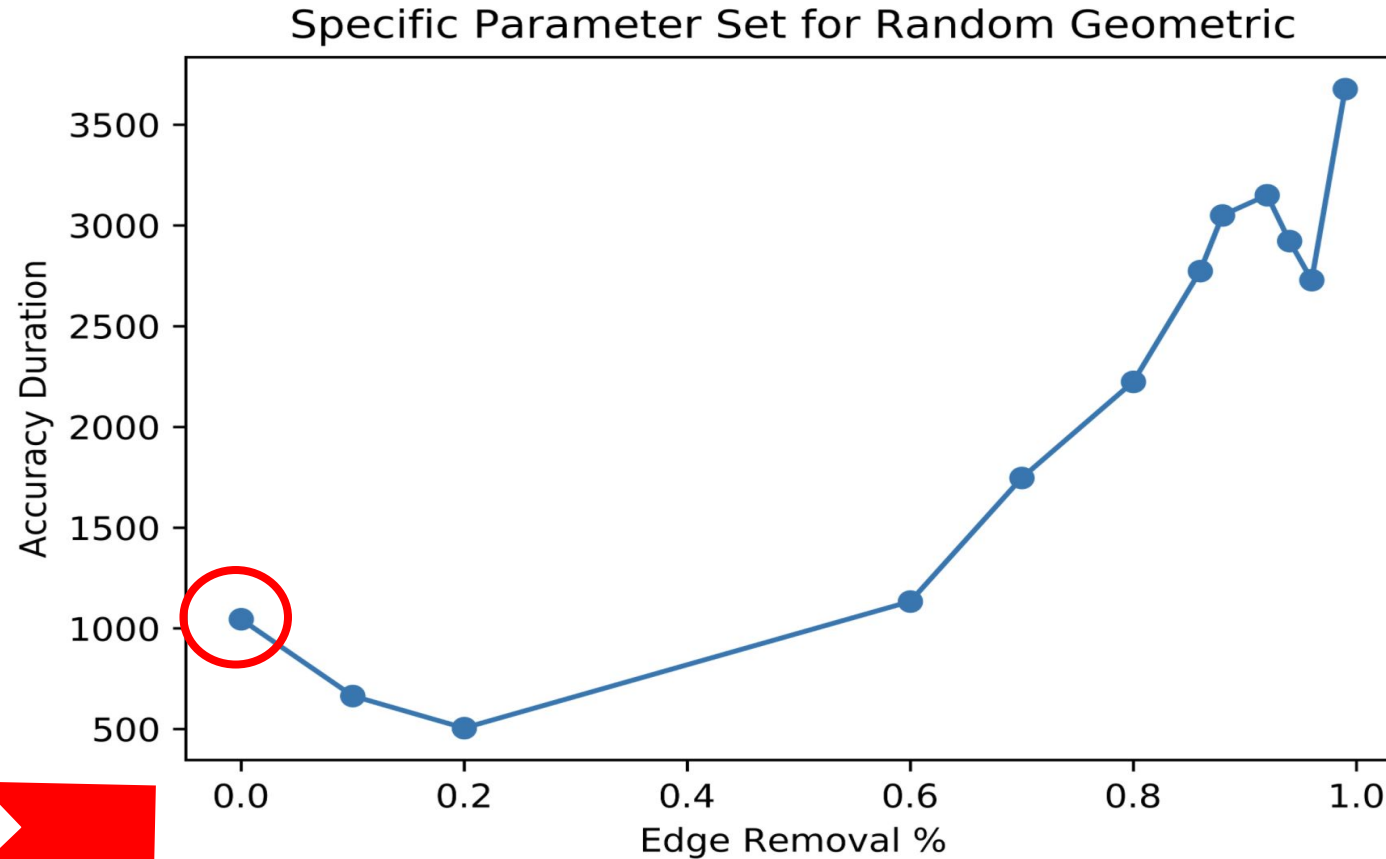
MS13

Ideal Reservoir Networks for Replicating Chaos

Reservoir computers are a machine learning model that, unlike most standard models, make use of an internal complex network. Often these networks are chosen to have a real-world like topology, e.g. Barabasi-Albert, Watts-Strogatz, Erdos-Renyi, etc. To understand the effect of this network structure we trained a large number of reservoir computers to replicate the chaotic dynamics of the Lorenz attractor. Over a wide range of network topologies we found that sparse or "thinned" networks are better suited to learn the dynamics of the Lorenz attractor. To help interpret these findings we analyzed how the stability of the reservoir's fixed points vary in response to the training signal. Here we found that high spectral radius and high connectivity of the network reduce the sensitivity of reservoir computers to input data, making them poorly suited for learning. This analysis offers some insight into better design principles for reservoir computers and a better understanding of the role of network topology in machine learning.

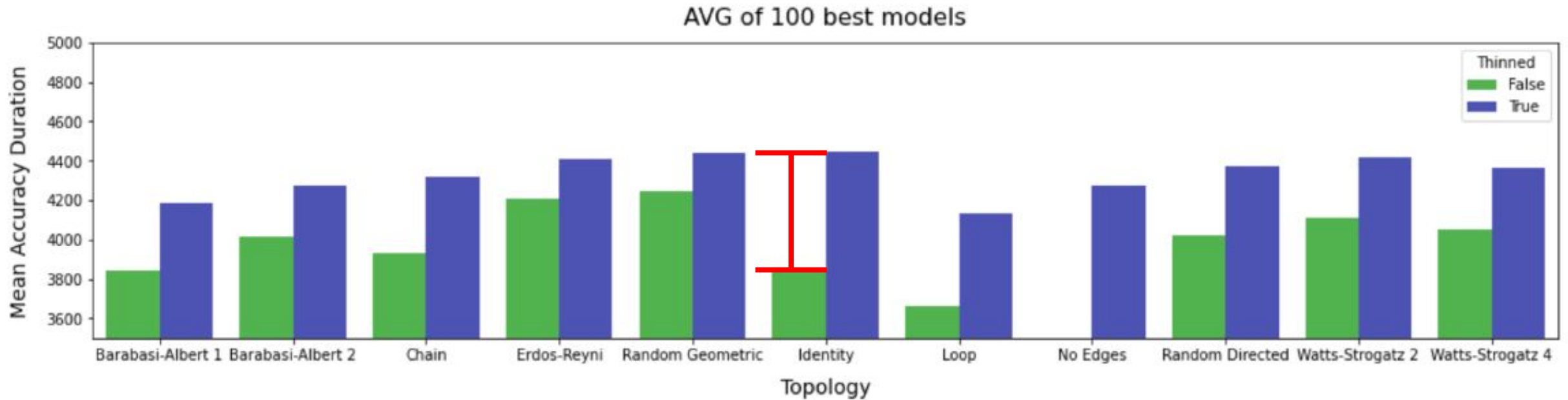
Ben Z. Webb
Brigham Young University
bwebb@mathematics.byu.edu

Extreme Thinning of Network



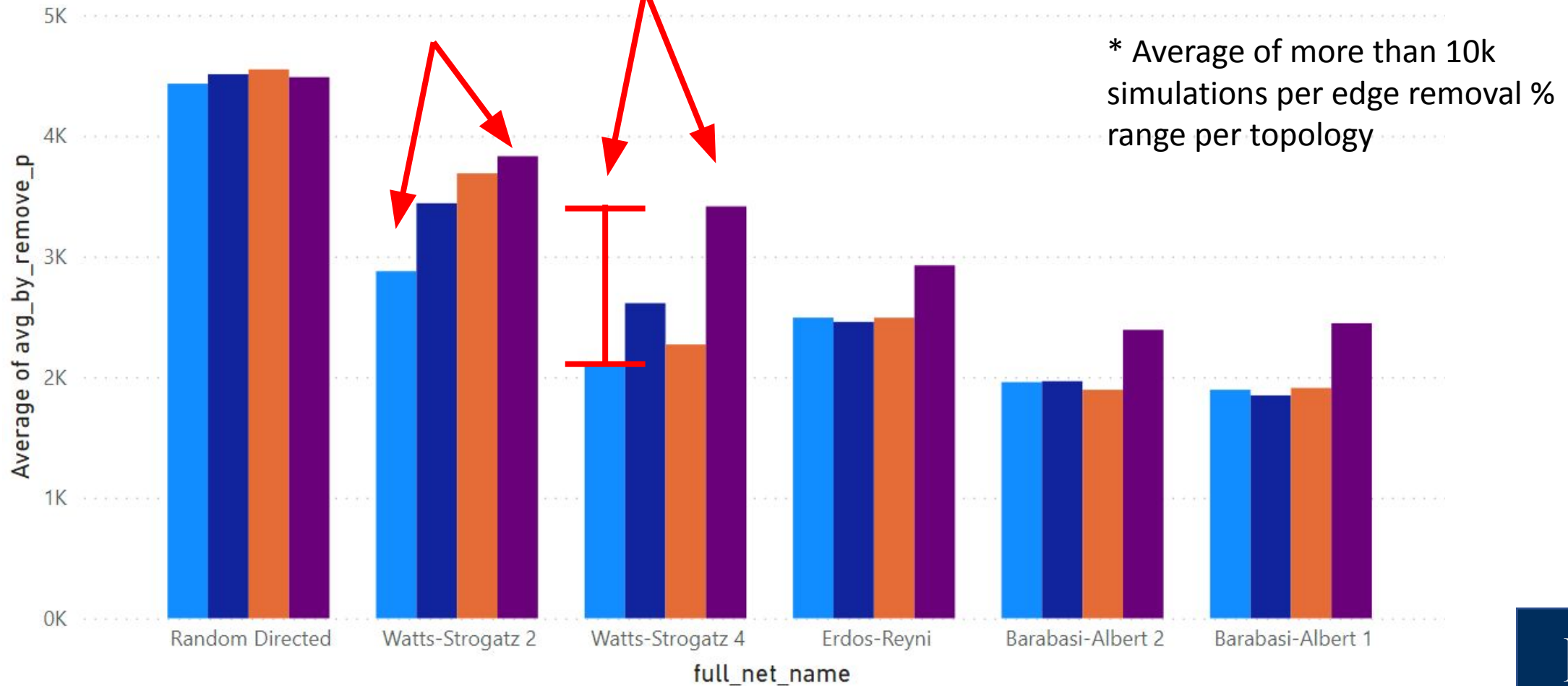
* Individual Parameter sets are shown
(no aggregation across other parameters like gamma, sigma, etc)

Extreme Thinning of Network



Extreme Thinning of Network

range_name ● 0.0 to 0.25 ● 0.25 to 0.5 ● 0.5 to 0.75 ● 0.75 to 1.0



Optional Questions

How do our results compare to other researchers?

[Return](#)

BYU

How do common network structures (topology) affect predictability beyond thinning?

[Return](#)

BYU

Why does spectral radius interact with thinning?

[Return](#)

BYU

How do our results compare to other researchers?

Hybrid Forecasting of Chaotic Processes: Using Machine Learning in Conjunction with a Knowledge-Based Model

Jaideep Pathak,¹ Alexander Wikner,² Rebeckah Fussell,³ Sarthak Chandra,¹ Brian R. Hunt,¹ Michelle Girvan,¹ and Edward Ott¹

¹⁾ *University of Maryland, College Park, Maryland 20742, USA.*

²⁾ *Rice University, Houston, Texas 77005, USA.*

³⁾ *Haverford College, Haverford, Pennsylvania 19041, USA.*

(Dated: 14 March 2018)

[Return](#)

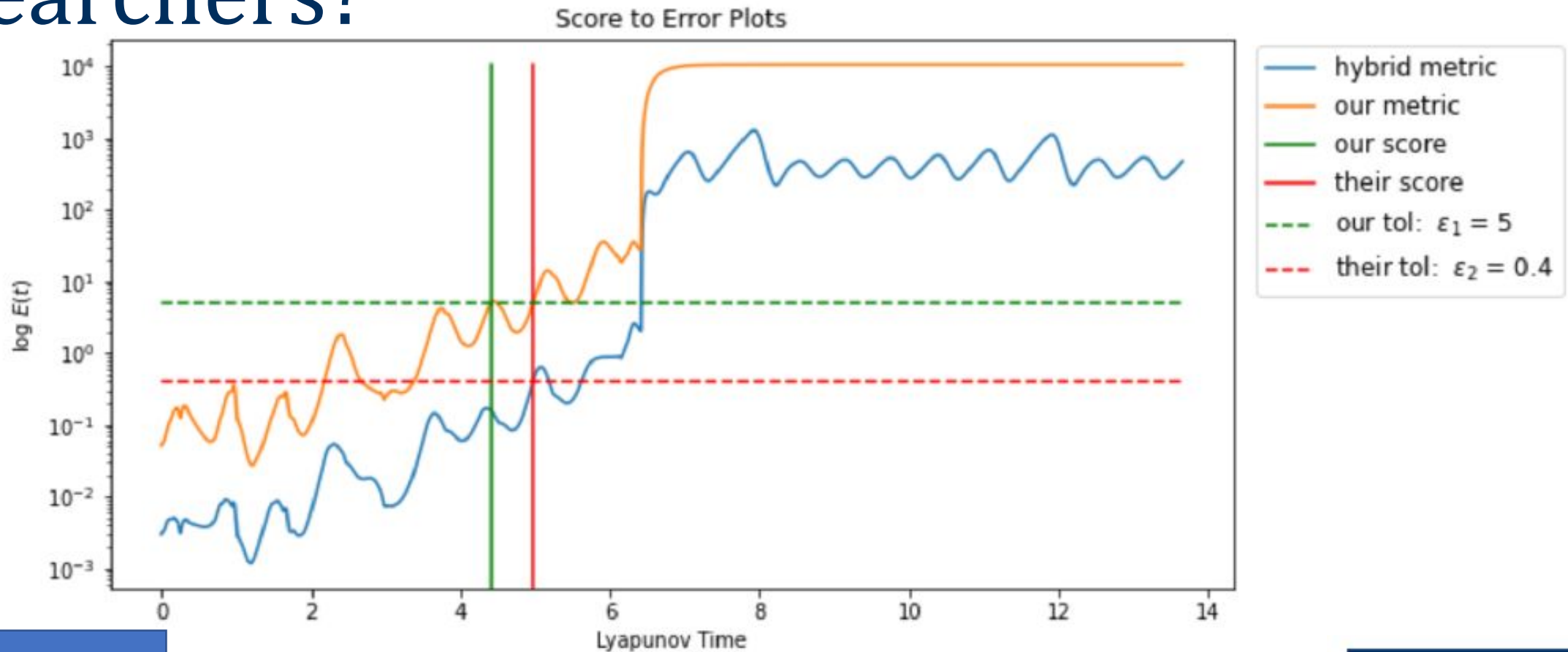
BYU

Metric Comparison

$$\max_{t^* \in [0, T]} \{t^* : \|\tilde{\mathbf{u}}(t) - \mathbf{u}(t)\|_2 < 5, \forall t \in [0, t^*]\}$$

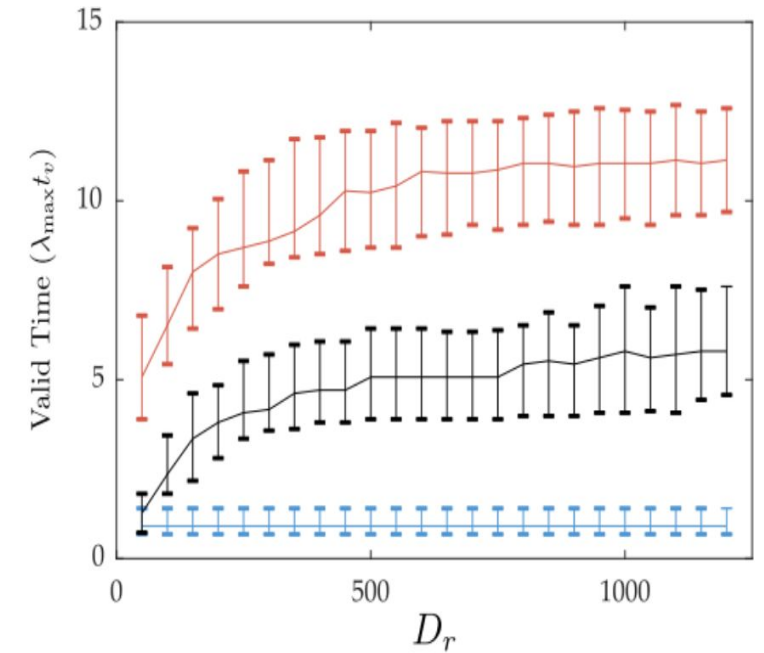
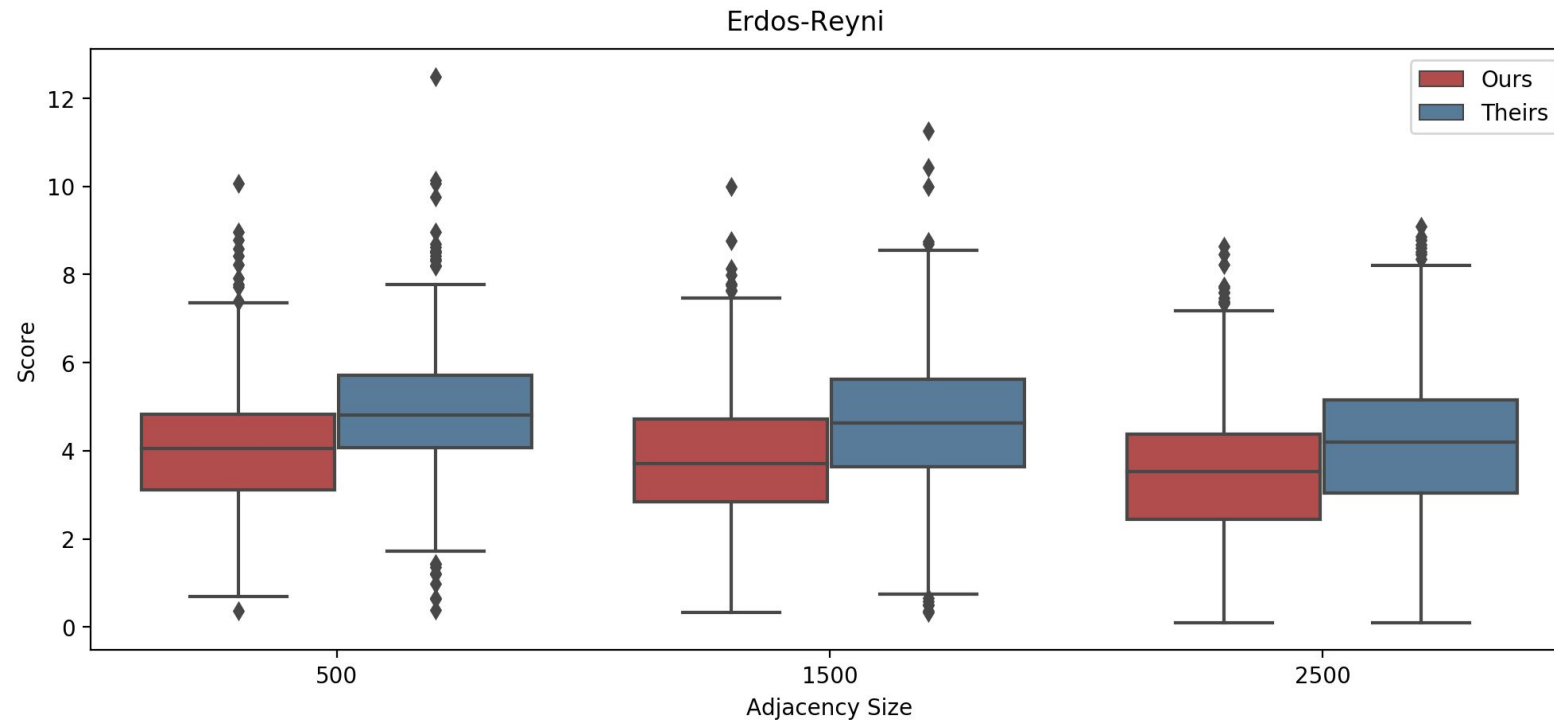
$$\max_{t^* \in [0, T]} \{t^* : \frac{\|\tilde{\mathbf{u}}(t) - \mathbf{u}(t)\|_2}{\|\mathbf{u}(t)\|_2} < 0.4, \forall t \in [0, t^*]\}$$

How do our results compare to other researchers?



[Return](#)

Metric & Performance Comparison



Pathak
et. al.
2018

[Return](#)

BYU

THANK You

[Return](#)

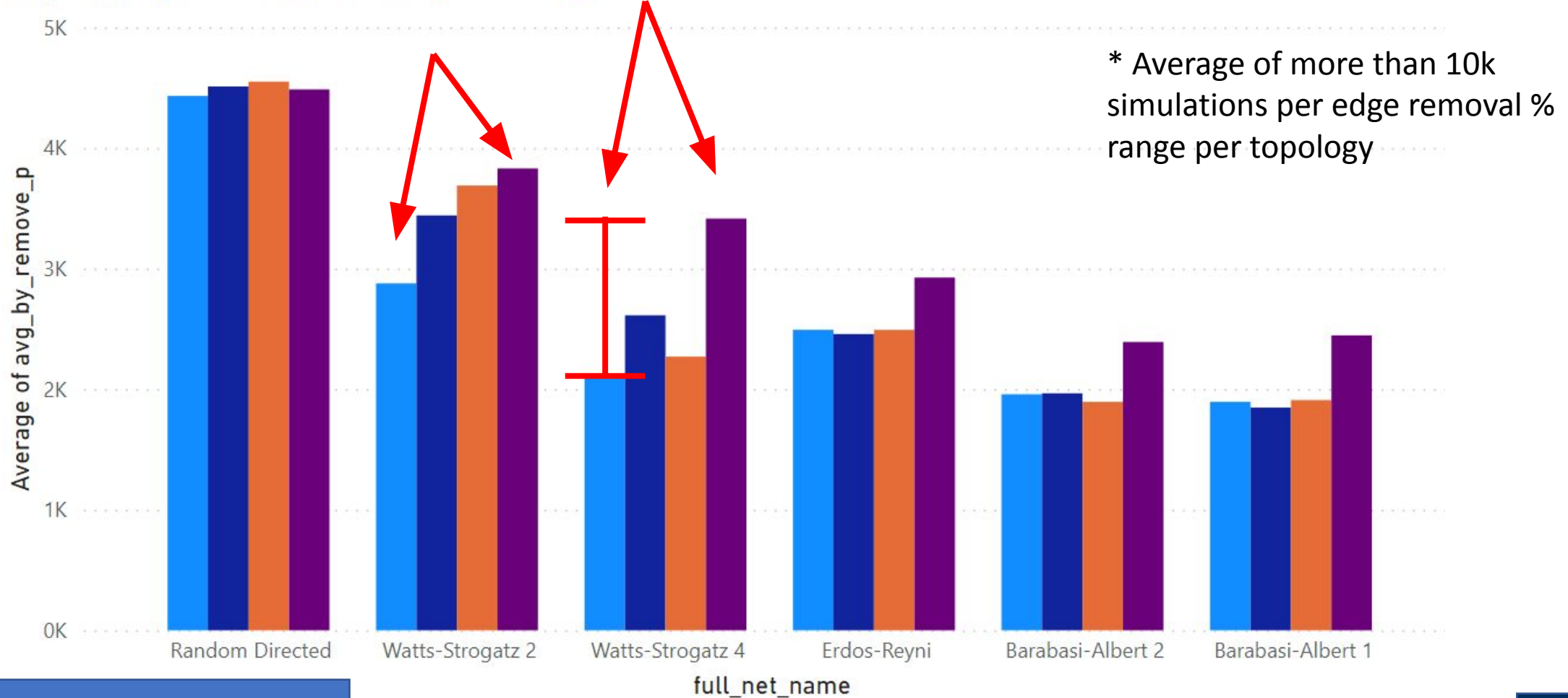
BYU

Potential Sources of Bias

- Potential Bias: 18x more thinned simulations than not
 - Prevalent when aggregating thinned & not thinned
 - Like in the Avg 100 best models comparison
 - Comparing Topology's performance
- Potential Solution:
 - awareness
 - aggregate first by edge removal percentage and then aggregate thereafter (if necessary)
 - Edge removal % range by topology barchart

Overcoming Potential Bias

range_name ● 0.0 to 0.25 ● 0.25 to 0.5 ● 0.5 to 0.75 ● 0.75 to 1.0



[Return](#)

BYU

Potential Sources of Bias

- Potential Bias: Confirmation Bias
 - Prevalent when showing examples of individual parameter sets
 - This system is susceptible to high variance in terms of performance
- Potential Solution:
 - Aggregate as to identify general trends
 - Specify whether we seek to show that THERE EXIST cases where sparsity improves performance
 - Perform theoretical analyses on dynamics to discover design principles

[Return](#)

BYU

Potential Sources of Bias

- Potential Bias: Comparison Incompatibilities
 - Disparities between number of simulations for various parameter sets, like topology for example.
- Potential Solution:
 - The topologies become nearly indistinguishable when they've been thinned to nearly 100%.
 - Consider claims made by analysis within one topology
 - Science: keep variables constant like Topology – and vary a different variable (like just spectral radius, or just edge removal %)

THANK You

[Return](#)

BYU

Pro's & Con's of a Reservoir Computer (or Liquid State Machine = LSM), Wikipedia states

LSMs have been put forward as a way to explain the operation of [brains](#). LSMs are argued to be an improvement over the theory of artificial neural networks because:

1. Circuits are not hard coded to perform a specific task.
2. Continuous time inputs are handled "naturally".
3. Computations on various time scales can be done using the same network.
4. The same network can perform multiple computations.

Criticisms of LSMs as used in [computational neuroscience](#) are that

1. LSMs don't actually explain how the brain functions. At best they can replicate some parts of brain functionality.
2. There is no guaranteed way to dissect a working network and figure out how or what computations are being performed.
3. Very little control over the process.

[Return](#)

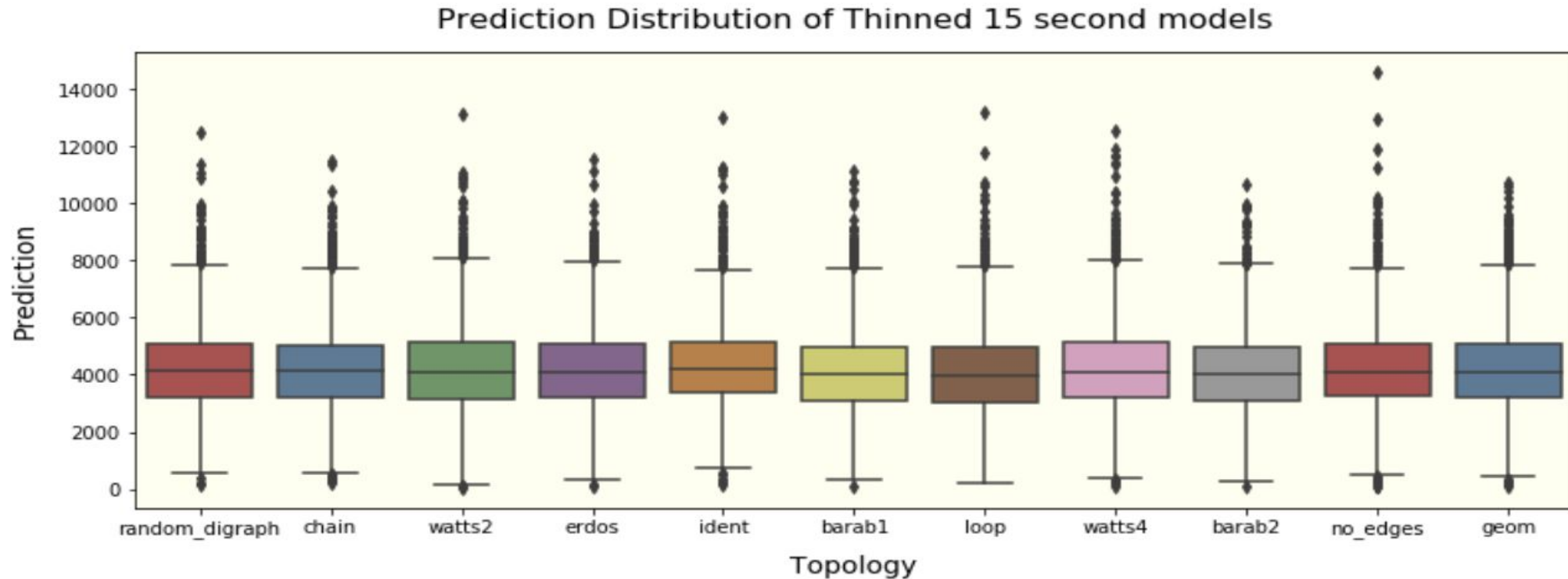
BYU

THANK You

[Return](#)

BYU

How does topology affect predictability?



[Return](#)

The networks of the different topologies become nearly indistinguishable as thinned completely which could explain why there is minimal difference in predictability.

THANK You

[Return](#)

BYU