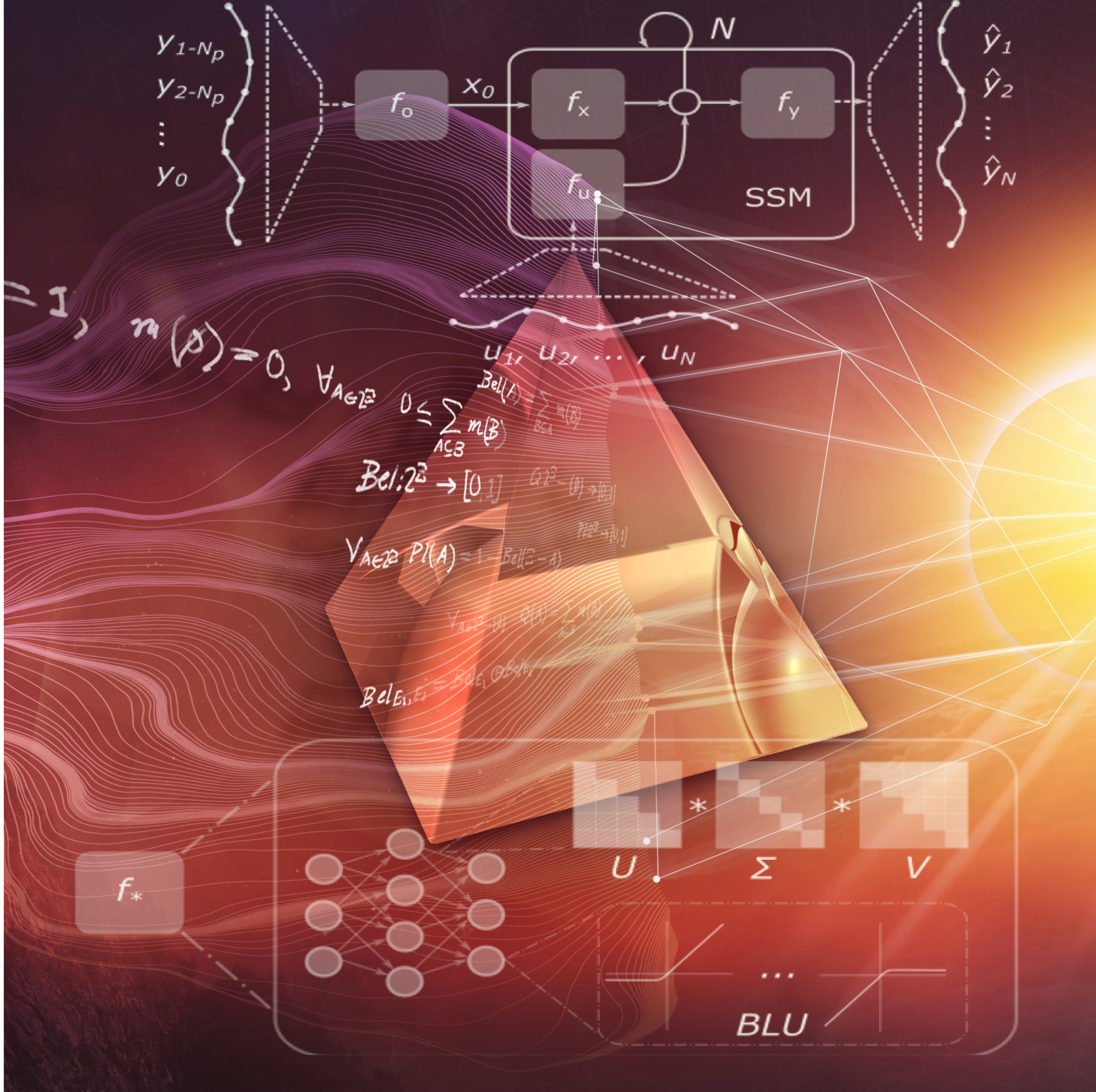
Pacific Northwest NATIONAL LABORATORY

OAK RIDGE National Laboratory

On Stochastic Stability of Deep Markov Models

DLC, April 8, 2021

Jan Drgona (PNNL), Sayak Mukherjee (PNNL), Jiaxin Zhang (ORNL), Frank Liu (ORNL), Mahantesh Halappanavar (PNNL)



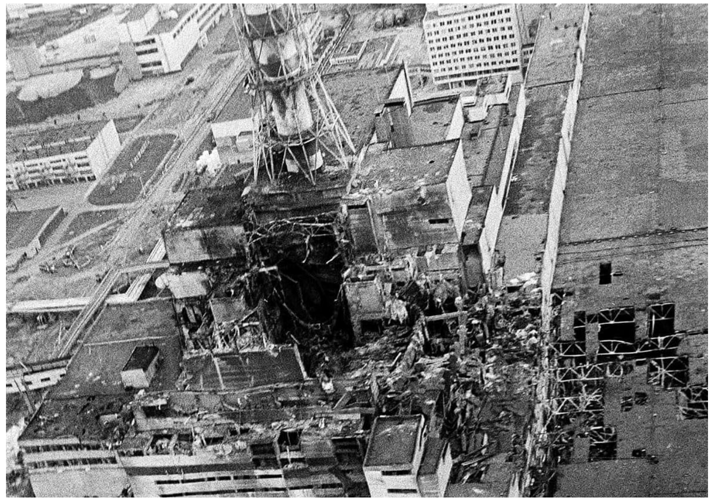
U.S. DEPARTMENT OF ENERGY BATTELLE

PNNL is operated by Battelle for the U.S. Department of Energy

Stability Analysis of Deep Markov Models

- **Motivation**
  - Safety-critical systems call for formal verification methods to ensure safe operation.
  - Properties like stability and robustness are crucial for reliable modeling and control.
- **Objectives**
  - Sufficient conditions for stochastic stability of deep Markov models (DMMs).
- **Approaches**
  - Apply system-theoretic analysis methods on DMMs.

Real-world consequences of unstable systems.



Reactors

Autonomous cars

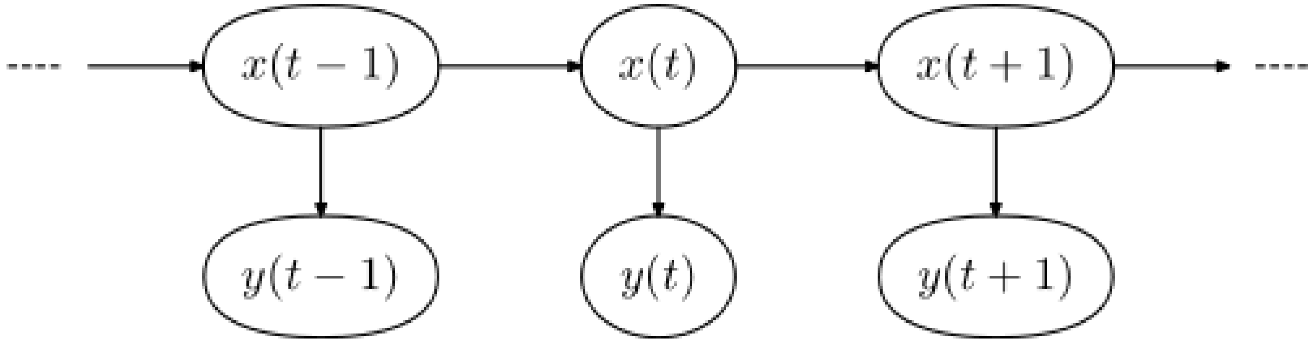
Rocket landing

Deep Markov Models

$$P(\mathbf{x}_{0:T}, \mathbf{y}_{0:T}) = P(\mathbf{x}_0)P(\mathbf{y}_0|\mathbf{x}_0) \prod_{0}^{T-1} P(\mathbf{x}_{t+1}|\mathbf{x}_t)P(\mathbf{y}_t|\mathbf{x}_t).$$

$$\mathbf{x}_{t+1} \sim \mathcal{N}(K_\alpha(\mathbf{x}_t, \Delta t), L_\beta(\mathbf{x}_t, \Delta t))$$
$$\mathbf{y}_t \sim \mathcal{M}(F_\kappa(\mathbf{x}_t))$$

$$K_\alpha(\mathbf{x}_t, \Delta t) = \mathbf{f}_{\theta_\mathbf{f}}(\mathbf{x}_t)$$
$$\text{vec}(L_\beta(\mathbf{x}_t, \Delta t)) = \mathbf{g}_{\theta_\mathbf{g}}(\mathbf{x}_t)$$



https://en.wikipedia.org/wiki/Hidden_Markov_mode

Deep Markov Models:
- Probabilistic graphical model (PGM)
- Generative model of sequential data
- Applications:
  - Economics, finance
  - Pattern recognition
  - Signal processing

Exploring connections between:
- Stability of stochastic systems
- Deep Markov models (DMMs)
- Contraction of DMM transitions
- Operator norms
- Banach fixed point theorem

Deep Neural Networks as Piecewise Affine Maps

DNN $$\psi_{\theta_\psi}(\mathbf{x}) = \mathbf{A}_L^\psi \mathbf{h}_L^\psi + \mathbf{b}_L$$
$$\mathbf{h}_l^\psi = v(\mathbf{A}_{l-1}^\psi \mathbf{h}_{l-1}^\psi + \mathbf{b}_{l-1})$$

PWA map $$\psi_{\theta_\psi}(\mathbf{x}) = \mathbf{A}_\psi(\mathbf{x})\mathbf{x} + \mathbf{b}_\psi(\mathbf{x}).$$

Local linear dynamics of DNN

- At every point $\mathbf{x}$, DNN can be represented as a product of PWA maps:

$$\mathbf{A}_\psi(\mathbf{x})\mathbf{x} = \mathbf{A}_L^\psi \mathbf{\Lambda}_{\mathbf{z}_L}^\psi \mathbf{A}_{L-1}^\psi \ldots \mathbf{\Lambda}_{\mathbf{z}_1}^\psi \mathbf{A}_0^\psi \mathbf{x}$$

$$\mathbf{b}_l^\psi := \mathbf{A}_i^\psi \mathbf{\Lambda}_{\mathbf{z}_{l-1}}^\psi \mathbf{b}_{l-1}^\psi + \mathbf{A}_i^\psi \boldsymbol{\sigma}_{l-1}(\mathbf{0}) + \mathbf{b}_l, \quad l \in \mathbb{N}_1^L$$

PWA activation map

$$\boldsymbol{\sigma}(\mathbf{z}) = \begin{bmatrix} \sigma(z_1) \\ \vdots \\ \sigma(z_n) \end{bmatrix} = \begin{bmatrix} \frac{z_1(\sigma(z_1)-\sigma(0)+\sigma(0))}{z_1} \\ \vdots \\ \frac{z_n(\sigma(z_n)-\sigma(0)+\sigma(0))}{z_n} \end{bmatrix} = \begin{bmatrix} \frac{\sigma(z_1)-\sigma(0)}{z_1} & & \\ & \ddots & \\ & & \frac{\sigma(z_n)-\sigma(0)}{z_n} \end{bmatrix} \mathbf{z} + \begin{bmatrix} \sigma(0) \\ \vdots \\ \sigma(0) \end{bmatrix}$$

Deep Neural Networks as Piecewise Affine Maps

DNN $\psi_{\theta_\psi}(\mathbf{x}) = \mathbf{A}_L^\psi \mathbf{h}_L^\psi + \mathbf{b}_L$

$$\mathbf{h}_l^\psi = v(\mathbf{A}_{l-1}^\psi \mathbf{h}_{l-1}^\psi + \mathbf{b}_{l-1})$$

PWA map $\psi_{\theta_\psi}(\mathbf{x}) = \mathbf{A}_\psi(\mathbf{x})\mathbf{x} + \mathbf{b}_\psi(\mathbf{x}).$

Local linear dynamics of DNN

- At every point $\mathbf{x}$, DNN layer can be represented as:

$$\boldsymbol{\sigma}_l(\mathbf{A}_l^\psi \mathbf{x}_l + \mathbf{b}_l) = \boldsymbol{\Lambda}_{\mathbf{z}_l}^\psi(\mathbf{A}_l^\psi \mathbf{x}_l + \mathbf{b}_l) + \boldsymbol{\sigma}(\mathbf{0}) = \boldsymbol{\Lambda}_{\mathbf{z}_l}^\psi \mathbf{A}_l^\psi \mathbf{x}_l + \boldsymbol{\Lambda}_{\mathbf{z}_l}^\psi \mathbf{b}_l + \boldsymbol{\sigma}_l(\mathbf{0})$$

PWA activation map

$$\boldsymbol{\sigma}(\mathbf{z}) = \begin{bmatrix} \frac{\sigma(z_1)-\sigma(0)}{z_1} & & \\ & \ddots & \\ & & \frac{\sigma(z_n)-\sigma(0)}{z_n} \end{bmatrix} \mathbf{z} + \begin{bmatrix} \sigma(0) \\ \vdots \\ \sigma(0) \end{bmatrix} = \boldsymbol{\Lambda}_{\mathbf{z}}^\psi \mathbf{z} + \boldsymbol{\sigma}(\mathbf{0})$$

Local Lipschitz Constants of Deep Neural Networks

Neural Network PWA operator norm:

$$||\mathbf{g}_{\theta_{\mathbf{g}}}(\mathbf{x})||_p = ||\mathbf{A}_{\mathbf{g}}(\mathbf{x})\mathbf{x} + \mathbf{b}_{\mathbf{g}}(\mathbf{x})||_p$$

Triangle inequality:

$$||\mathbf{g}_{\theta_{\mathbf{g}}}(\mathbf{x})||_p \leq ||\mathbf{A}_{\mathbf{g}}(\mathbf{x})\mathbf{x}||_p + ||\mathbf{b}_{\mathbf{g}}(\mathbf{x})||_p,$$

$$\frac{||\mathbf{g}_{\theta_{\mathbf{g}}}(\mathbf{x})||_p}{||\mathbf{x}||_p} \leq ||\mathbf{A}_{\mathbf{g}}(\mathbf{x})||_p + \frac{||\mathbf{b}_{\mathbf{g}}(\mathbf{x})||_p}{||\mathbf{x}||_p}.$$

Local Lipschitz constant of DNN:

$$\mathcal{K}^{\mathbf{g}}(\mathbf{x}) = ||\mathbf{A}_{\mathbf{g}}(\mathbf{x})||_p + \frac{||\mathbf{b}_{\mathbf{g}}(\mathbf{x})||_p}{||\mathbf{x}||_p}.$$

Stochastic Stability of Deep Markov Models

Definition 3. *The stochastic process* $\mathbf{x}_t \in \mathbb{R}^n$ *is mean-square stable (MSS) if and only if there exists* $\mu \in \mathbb{R}^n, \boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}$, *such that* $\lim_{t\to\infty} \mathbb{E}(\mathbf{x}_t) = \mu$, *and* $\lim_{t\to\infty} \mathbb{E}(\mathbf{x}_t \mathbf{x}_t^T) = \boldsymbol{\Sigma}$.

Converging mean and bounded variance: $\mu = \mathbf{f}_{\theta_\mathbf{f}}(\mu) = \lim_{t\to\infty} \mathbf{f}_{\theta_\mathbf{f}}(\bar{\mathbf{x}}_t) \quad \|\mathbf{g}_{\theta_\mathbf{g}}(\mathbf{x}_t)\|_p < K,\ K > 0,\ \forall t.$

Sufficient stability conditions of DMM:

$$\|\mathbf{A}_\mathbf{f}(\mathbf{x})\|_p < 1$$

$$||\mathbf{A}_\mathbf{g}(\mathbf{x})||_p + \frac{||\mathbf{b}_\mathbf{g}(\mathbf{x})||_p}{\|\mathbf{x}\|_p} < K, K > 0,$$

$$\forall \mathbf{x} \in Domain(\mathbf{f}_{\theta_\mathbf{f}}(\mathbf{x}), \mathbf{g}_{\theta_\mathbf{g}}(\mathbf{x})).$$

Local Lipschitz constant of DNN:

$$\mathcal{K}^\mathbf{g}(\mathbf{x}) = ||\mathbf{A}_\mathbf{g}(\mathbf{x})||_p + \frac{||\mathbf{b}_\mathbf{g}(\mathbf{x})||_p}{\|\mathbf{x}\|_p}.$$

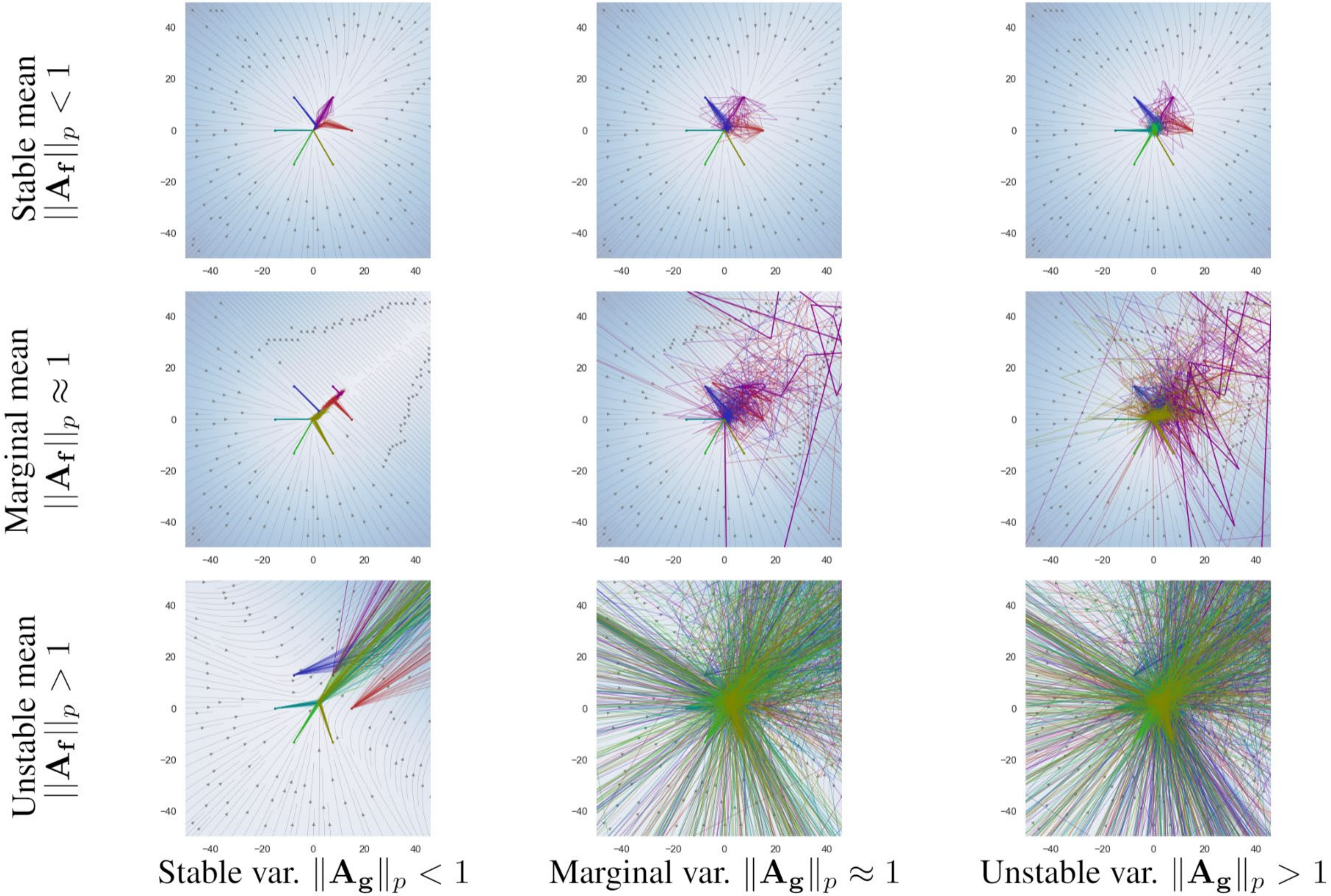
Contractive weights and activations imply DMM stability:

$$\|\mathbf{A}_i^\mathbf{f}\|_p < 1,\ ||\boldsymbol{\Lambda}_{\mathbf{z}_i}^\mathbf{f}||_p \leq 1\ i \in \mathbb{N}_1^{L_\mathbf{f}},$$

$$\|\mathbf{A}_j^\mathbf{g}\|_p < c^\mathbf{\Lambda},\ ||\boldsymbol{\Lambda}_{\mathbf{z}_j}^\mathbf{g}||_p \leq c^\mathbf{\Lambda},\ j \in \mathbb{N}_1^{L_\mathbf{g}},$$

$$\forall \mathbf{x} \in Domain(\mathbf{f}_{\theta_\mathbf{f}}(\mathbf{x}), \mathbf{g}_{\theta_\mathbf{g}}(\mathbf{x})).$$

Effect of Mean and Variance on Stochastic Stability of Deep Markov Models



Effect of Biases and Depth on the Stability of Deep Markov Models

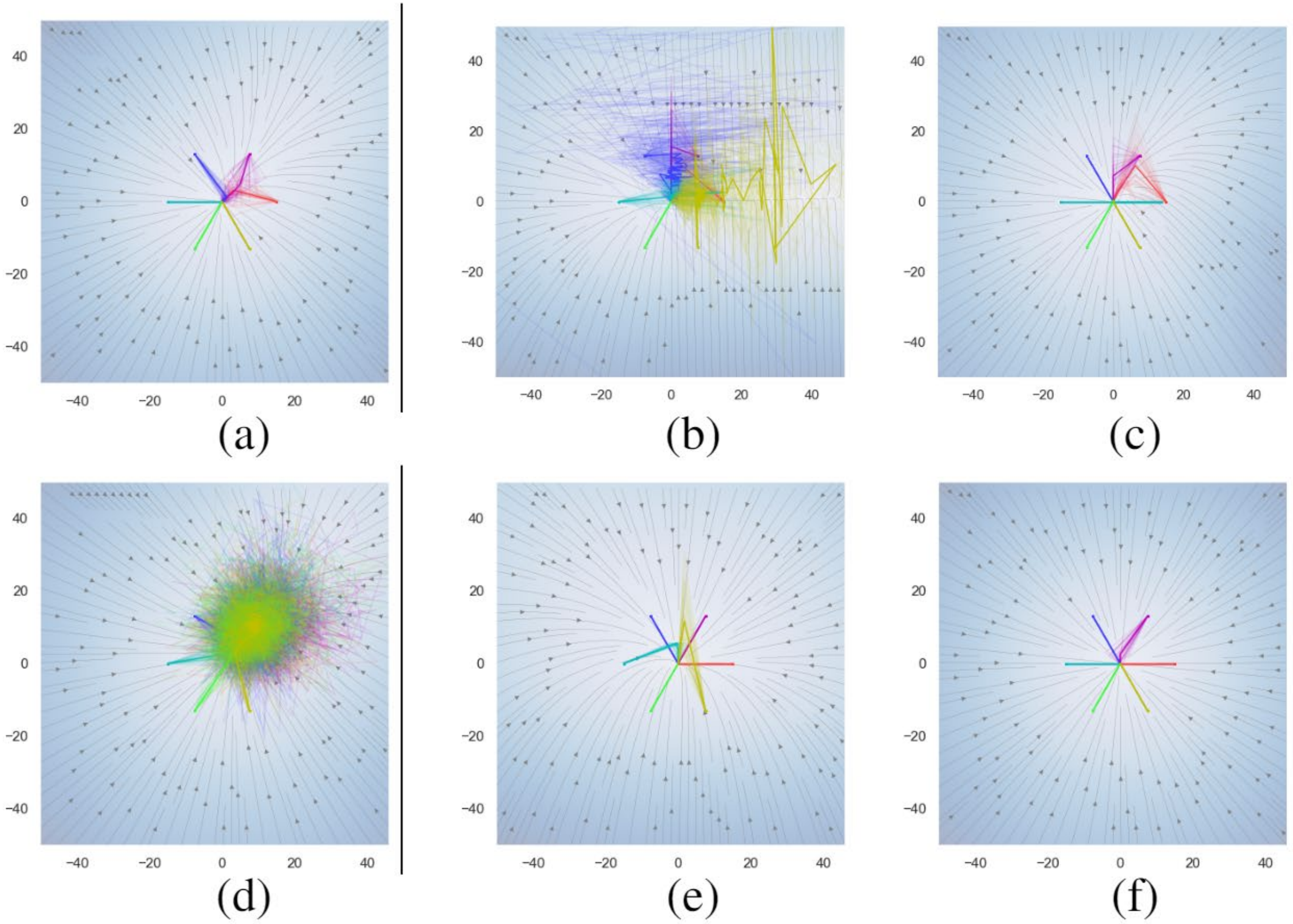


Figure 2: Left panels show the effect of biases using PF regularization and `ReLU` *activation ((a) w/o bias, (d) w bias). Right panels show the effect of network* $\mathbf{f}$ *depths with SVD regularization and* `ReLU` *: (b) 1 layer, (c) 2 layers, (e) 4 layers, (f) 8 layers.*

Stability of Activation Functions

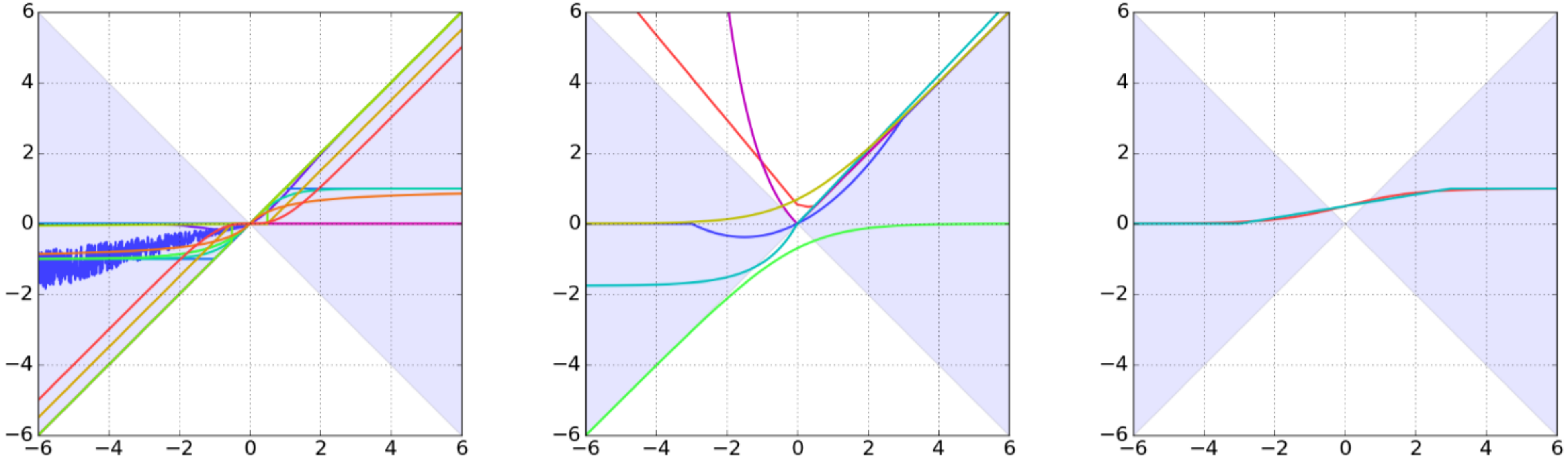
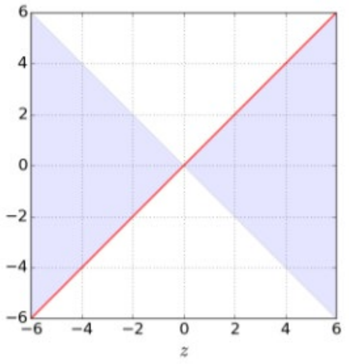


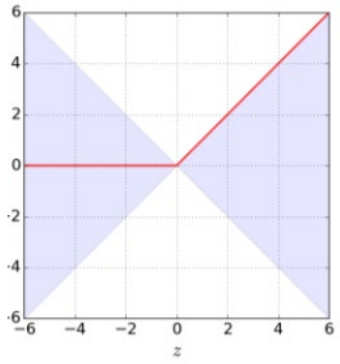
Figure 1: Activation functions with asymptotic stability (left), with unstable regions (middle), and with BIBO stability (right), respectively. Blue areas represent stable regions covering functions with trivial null space and Lipschitz constant $K \leq 1$.

1. Globally stable activations: `SoftExponential`, `BLU`, `PReLU`, `ReLU`, `GELU`, `RReLU`, `Hardtanh`, `ReLU6`, `Tanh`, `ELU`, `CELU`, `Hardshrink`, `LeakyReLU`, `Softshrink`, `Softsign`, `Tanhshrink`
2. Activations with unstable regions: `APLU`, `PELU`, `Sigmoid`, `Hardsigmoid`, `Hardswish`, `SELU`, `LogSigmoid`, `Softplus`, `Hardswish`, `Sigmoid`, `Hardsigmoid`

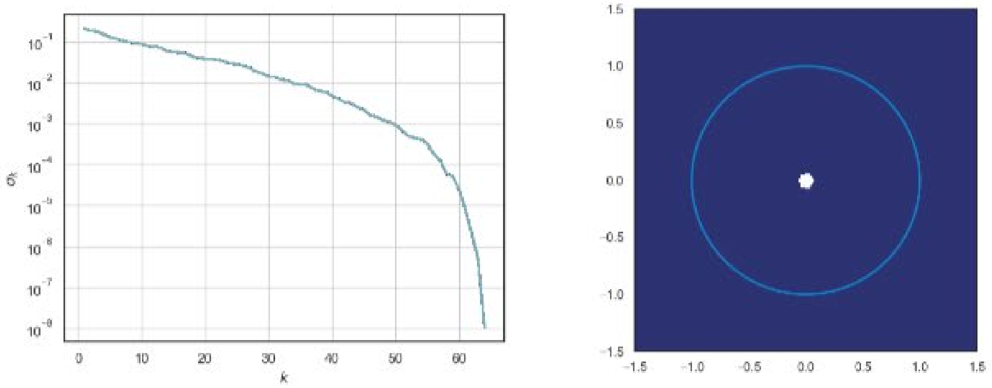
Nonlinearity of Deep Neural Networks with Different Activation Functions



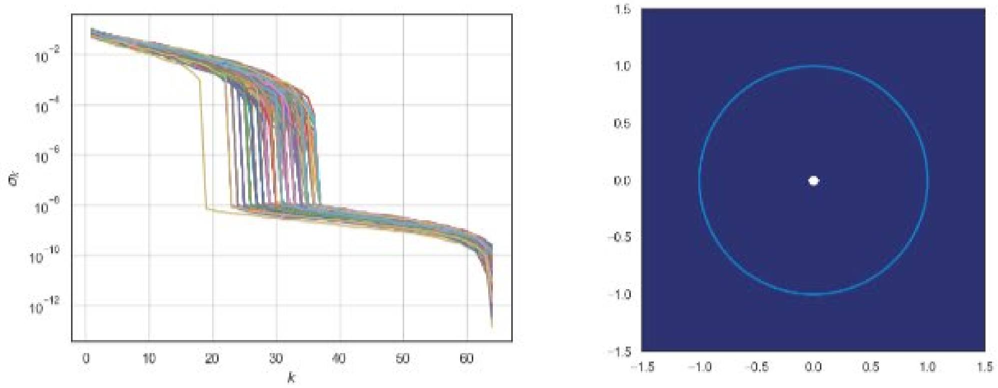
Identity



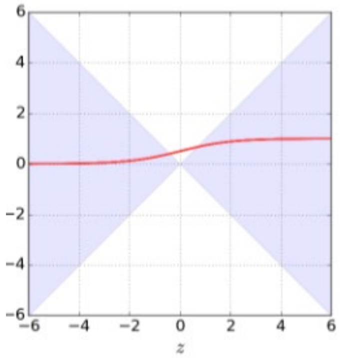
ReLU



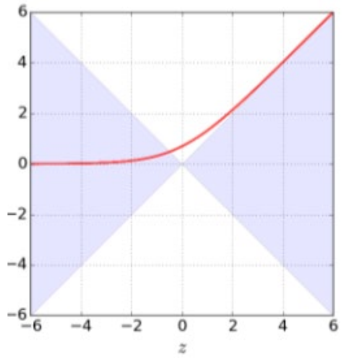
Identity



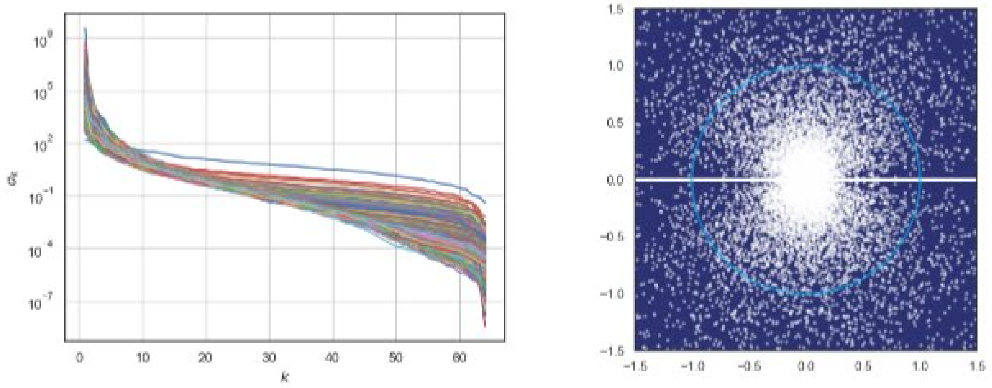
ReLU



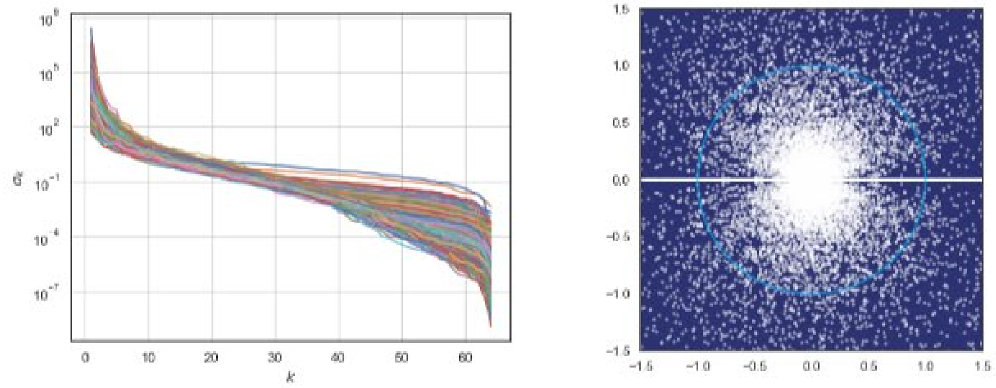
Sigmoid



SoftExponential



Sigmoid



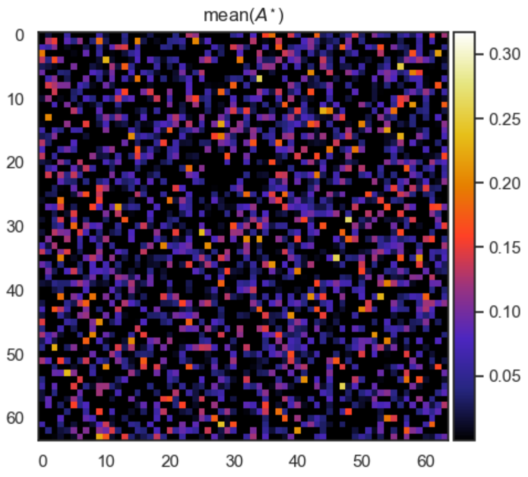
SoftExponential

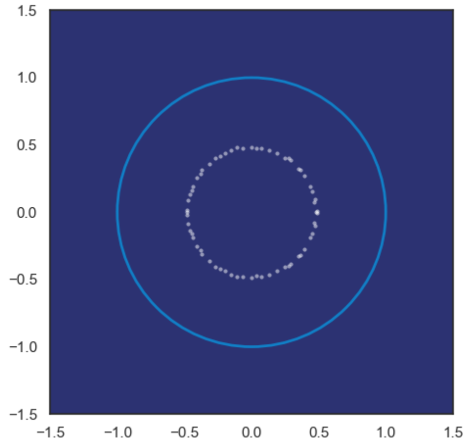
Practical Stability Constraints for DNN and DMM

SVD factorization

$$\tilde{\Sigma} = \mathrm{diag}(\lambda_{\max} - (\lambda_{\max} - \lambda_{\min}) \cdot \sigma(\Sigma))$$

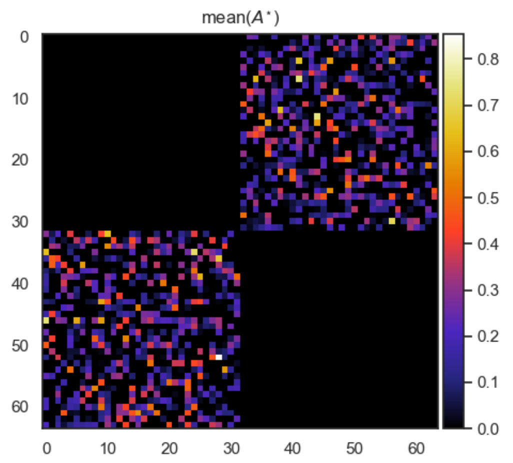
$$\tilde{\mathbf{A}} = \mathbf{U}\tilde{\Sigma}\mathbf{V}$$

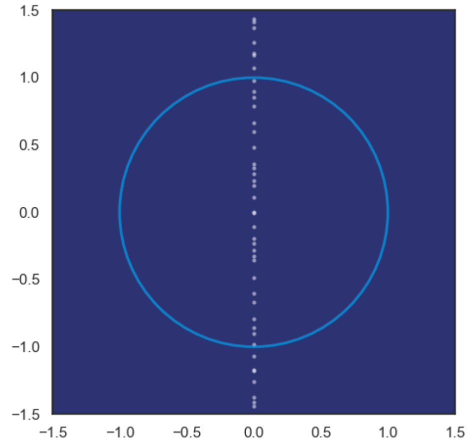




Hamiltonian weight

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{0} & \mathbf{A} \\ -\mathbf{A}^\top & \mathbf{0} \end{bmatrix}$$





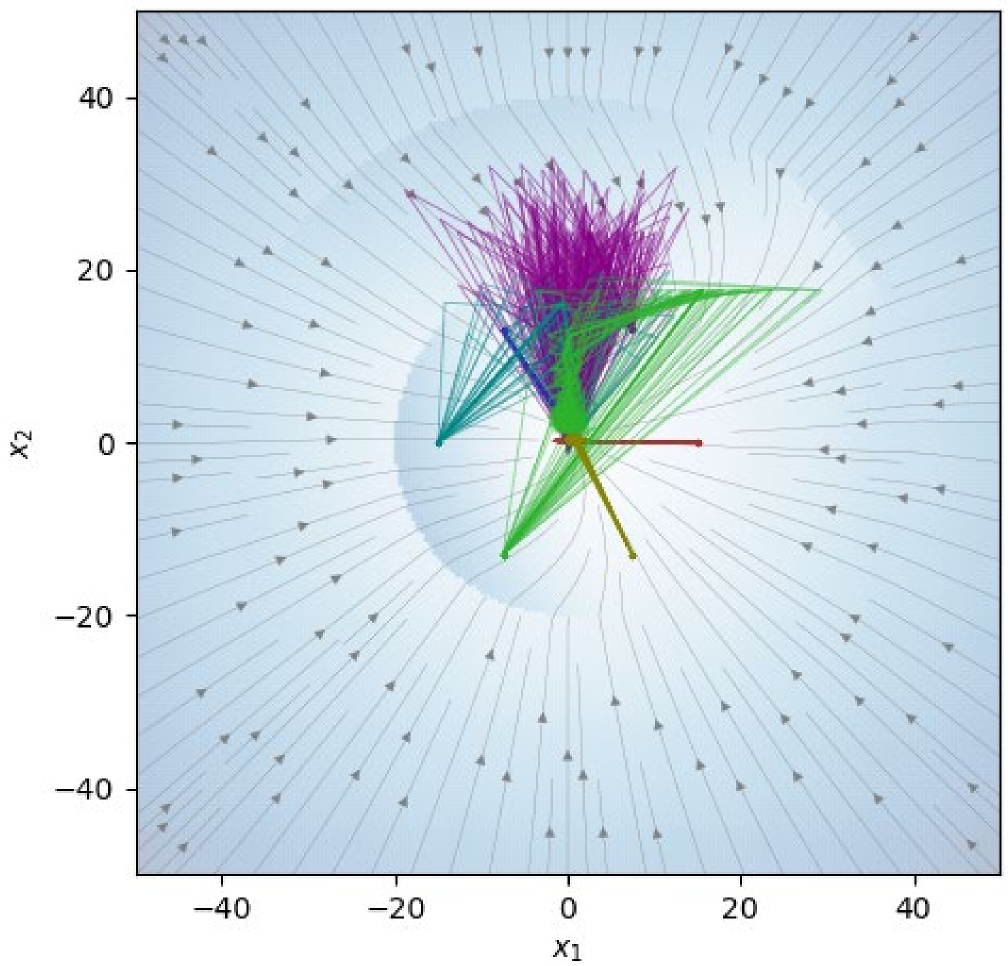
Pytorch implementation: https://github.com/pnnl/slim

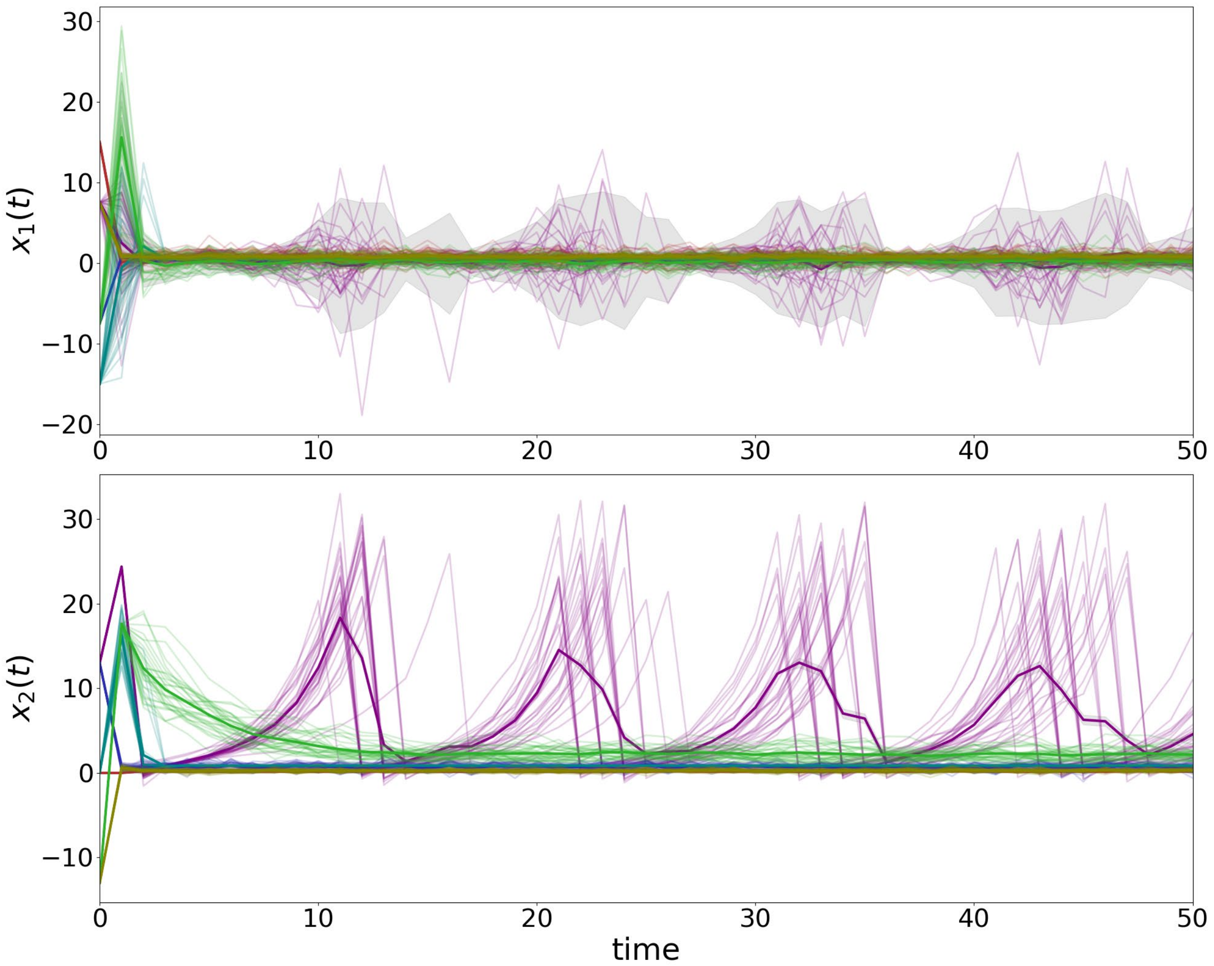
SLIM: Drop-in replacements for PyTorch nn.Linear for stable learning and inductive priors.

Jiong Zhang, et al., Stabilizing Gradients for Deep Neural Networks via Efficient SVD Parameterization, 2018.
Eldad Haber and Lars Ruthotto, Stable Architectures for Deep Neural Networks 2017

Parametric Stability Constraints for DMMs

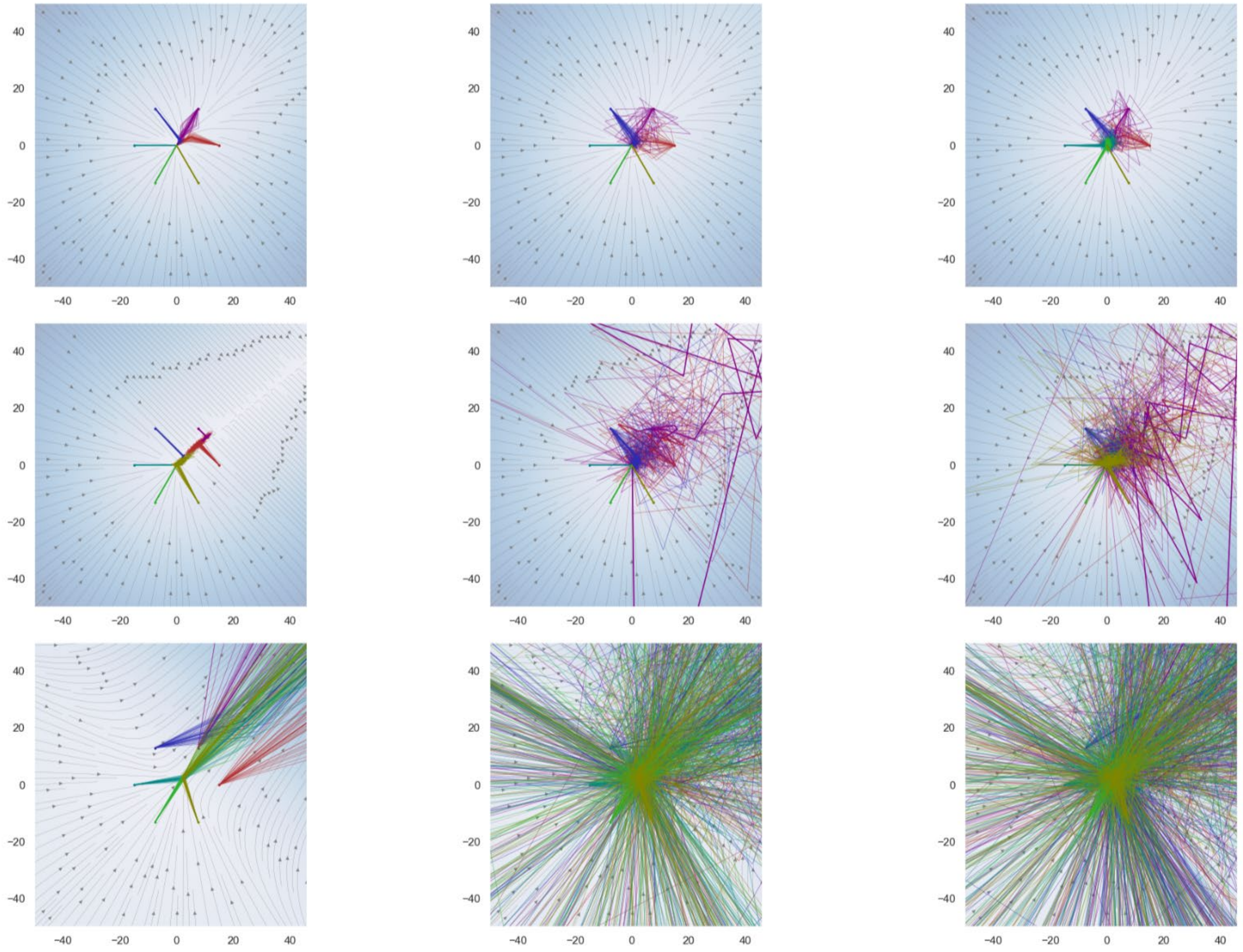
Constrained operator norms of DMMs: $\underline{\mathbf{p}}(\mathbf{x}) < \|\mathbf{A_f}(\mathbf{x})\|_p < \overline{\mathbf{p}}(\mathbf{x})$





Conclusion

- **Stability of Deep Markov Models**
  - Neural Networks as PWA maps
  - Contraction of PWA maps
  - Banach fixed point theorem
  - Operator norm constraints
  - Contraction of DMMs
- **Stable Weights**
  - Structured linear maps in Pytorch
  - https://pnnl.github.io/slim/
- **Contact**
  - jan.drgona@pnnl.gov
  - https://www.linkedin.com/in/drgona/
  - https://twitter.com/jan_drgona



Jan Drgona, Sayak Mukherjee, Jiaxin Zhang, Frank Liu, Mahantesh Halappanavar, On the Stochastic Stability of Deep Markov Models, NeurIPS 2021