Meta-state-space learning: A Novel Approach for the Identification of Stochastic Dynamic Systems

Gerben I. Beintema, Roland Tóth, Thomas B. Schön, Maarten Schoukens
contact: g.i.beintema@tue.nl

TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

UPPSALA UNIVERSITET

Linear stochastic systems

Representation	Identification
IO stochastic model, e.g. Box-Jenkins $\boldsymbol{y} = \frac{B}{F}\, u + \frac{C}{D}\, \boldsymbol{e}$	*Prediction Error Minimization (PEM) [1]*
State-space models, e.g. $\boldsymbol{x}_+ = A\,\boldsymbol{x} + B\,u + E\,\boldsymbol{v}$ $\boldsymbol{y} = C\,\boldsymbol{x} + D\,u + F\,\boldsymbol{e}$	*PEM-SS [1]* *Subspace (PBSID, SS-ARX) [2,3]*

[1] Ljung, Lennart. "System Identification: Theory for the User." Prentice Hall PTR, 1998.
[2] A. Chiuso. The role of vector auto regressive modeling in predictor based subspace identification. Automatica, 43(6):1034–1048, 2007.
[3] Katayama, Tohru. Subspace methods for system identification. Vol. 1. London: Springer, 2005.

Nonlinear stochastic systems

Representation	Identification
Nonlinear stochastic state-space $\boldsymbol{x}_+ = f(\boldsymbol{x}, u, \boldsymbol{e})$ $\boldsymbol{y} = h(\boldsymbol{x}, u, \boldsymbol{v})$	Particle Smoothers[4] Bayesian Methods[5] Challenging e.g. computationally (Monte Carlo)
Probability distributions $p(\boldsymbol{x}_+) = \int p(\boldsymbol{x}_+ \vert \boldsymbol{x}, u) p(\boldsymbol{x}) d\boldsymbol{x}$ $p(\boldsymbol{y}) = \int p(\boldsymbol{y} \vert \boldsymbol{x}, u) p(\boldsymbol{x}) d\boldsymbol{x}$	

EINDHOVEN UNIVERSITY OF TECHNOLOGY

[4] Schön, Thomas B., Adrian Wills, and Brett Ninness. "System identification of nonlinear state-space models." Automatica (2011)
[5] Ninness, Brett, and Soren Henriksen. "Bayesian system identification via Markov chain Monte Carlo techniques." Automatica 46.1 (2010): 40-51.

Nonlinear stochastic: Meta-state-space

Systems represented as

Nonlinear stochastic state-space

$$\boldsymbol{x}_+ = f(\boldsymbol{x}, u, \boldsymbol{e})$$

$$\boldsymbol{y} = h(\boldsymbol{x}, u, \boldsymbol{v})$$

Probability distributions

$$p(\boldsymbol{x}_+) = \int p(\boldsymbol{x}_+|\boldsymbol{x}, u)p(\boldsymbol{x})d\boldsymbol{x}$$

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{x}, u)p(\boldsymbol{x})d\boldsymbol{x}$$

have a meta-state-space representation

Meta-state-space:

Deterministic NL-SS!

$$z_{t+1} = f_z(z_t, u_t)$$

$$p(\boldsymbol{y}_t|z_t, u_t)$$

$z_t \in \mathbb{R}^{n_z}$ is the meta-state

Exact representation

Efficiently identified

EINDHOVEN UNIVERSITY OF TECHNOLOGY

The route

Derive meta-state for linear systems

Generalize to nonlinear systems

Identification method

Benchmark results

Derivation meta-state for linear system

$$\boldsymbol{x}_{t+1} = A\boldsymbol{x}_t + Bu_t + \boldsymbol{e}_t$$

$\boldsymbol{x}_0$ & $\boldsymbol{e}_t$ Gaussian

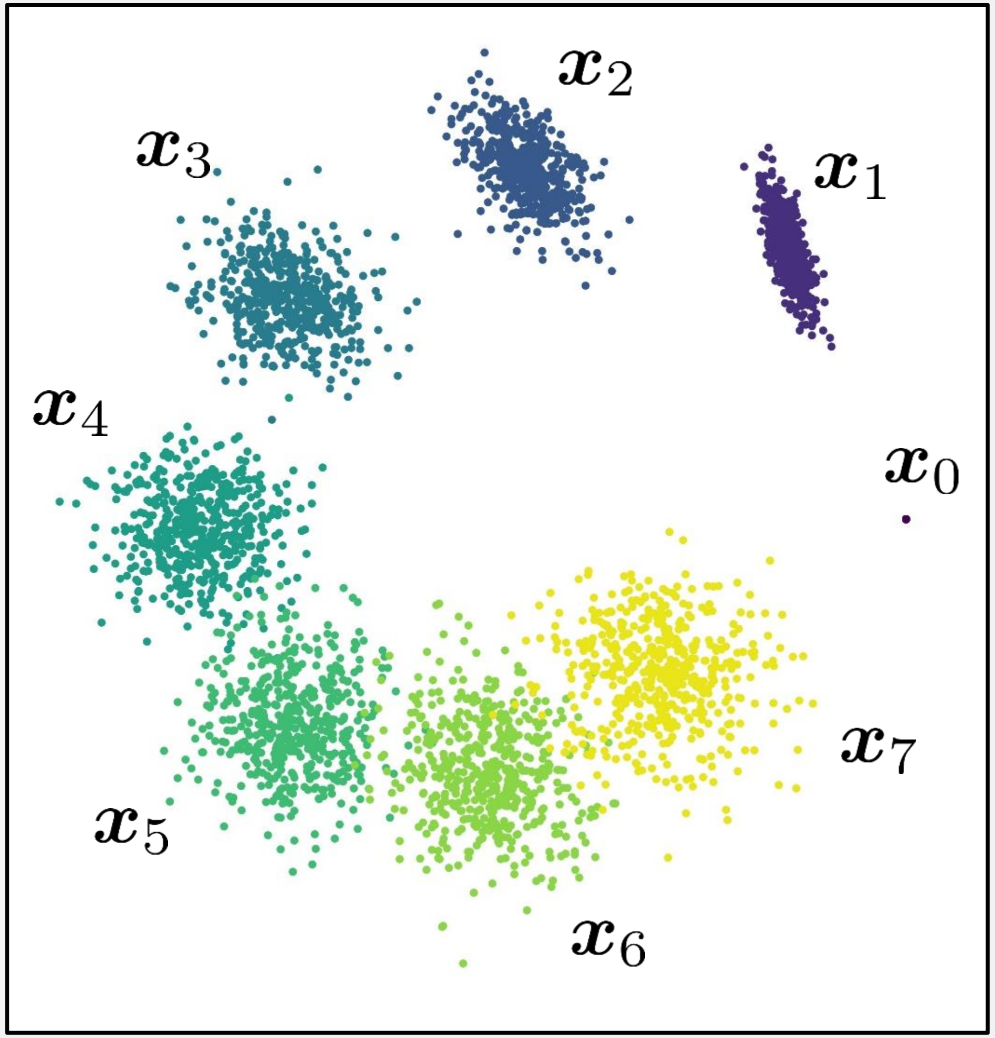
Observe that

$$p(\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{x}_t | \mu_t, \Sigma_t)$$

$$p^F(\boldsymbol{x}_{t+1} | \boldsymbol{x}_t, u_t) = \mathcal{N}(\boldsymbol{x}_{t+1} | A\boldsymbol{x}_t + Bu_t, \Sigma_e)$$

and we know that

$$p(\boldsymbol{x}_{t+1}) = \int p^F(\boldsymbol{x}_{t+1} | \boldsymbol{x}_t, u_t) p(\boldsymbol{x}_t) d\boldsymbol{x}_t$$



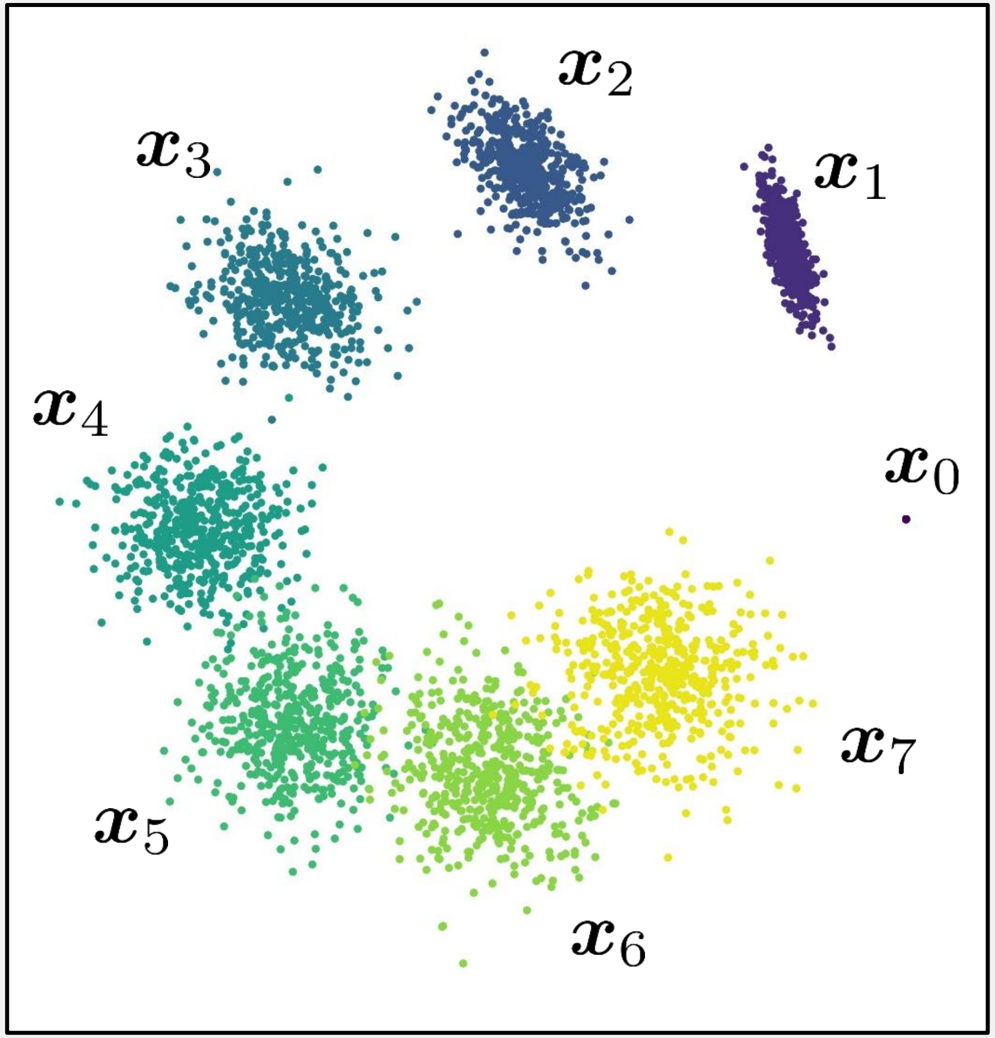
Derivation meta-state for linear system

$$\mathcal{N}(\boldsymbol{x}_{t+1}|\mu_{t+1}, \Sigma_{t+1}) =$$

$$\int \mathcal{N}(\boldsymbol{x}_{t+1}|A\boldsymbol{x}_t + Bu_t, \Sigma_e)\mathcal{N}(\boldsymbol{x}_t|\mu_t, \Sigma_t)d\boldsymbol{x}_t$$

Thus:

$$\mu_{t+1} = f_\mu(\mu_t, \Sigma_t, u_t)$$
$$\Sigma_{t+1} = f_\Sigma(\mu_t, \Sigma_t, u_t)$$

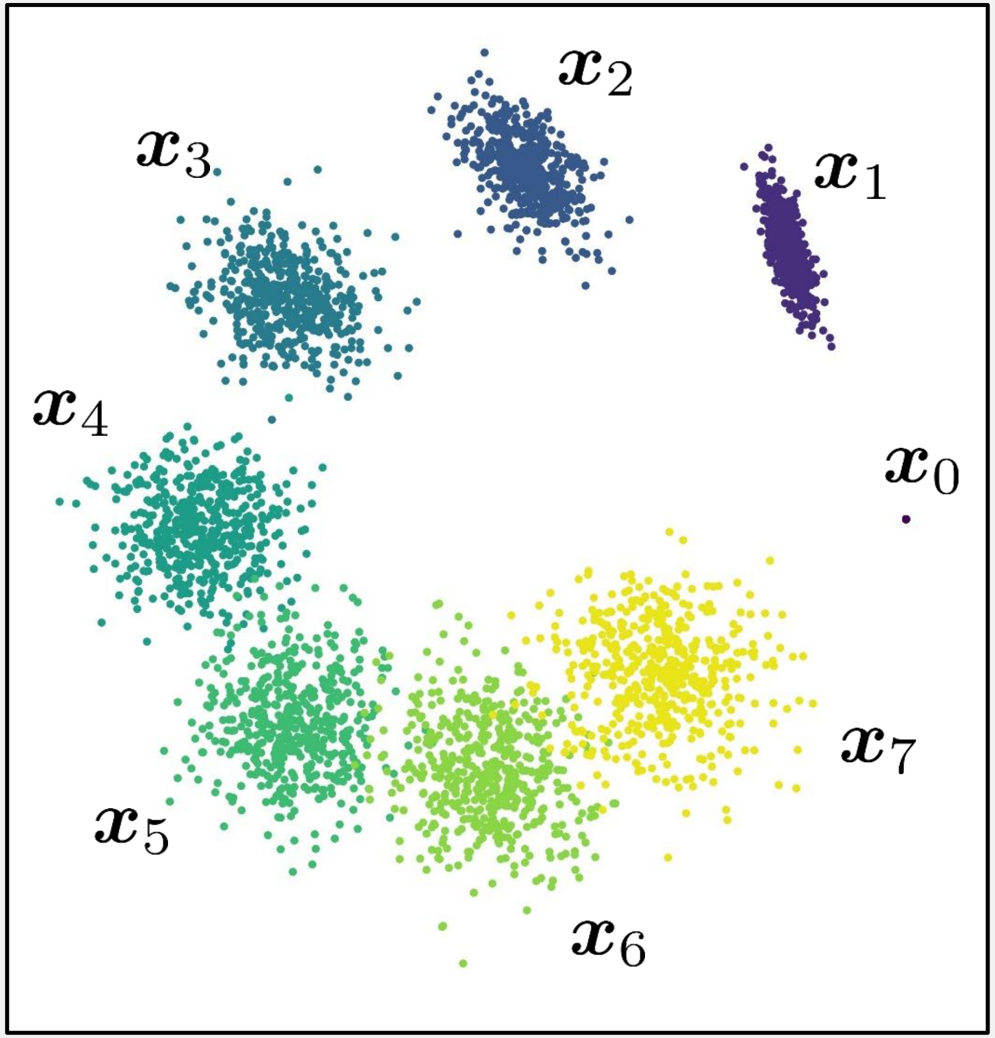


Derivation meta-state for linear system

$$\mu_{t+1} = f_\mu(\mu_t, \Sigma_t, u_t)$$
$$\Sigma_{t+1} = f_\Sigma(\mu_t, \Sigma_t, u_t)$$

Observation: we can create a new state-space equation by collecting into vector $z_t = Vec(\mu_t, \Sigma_t)$

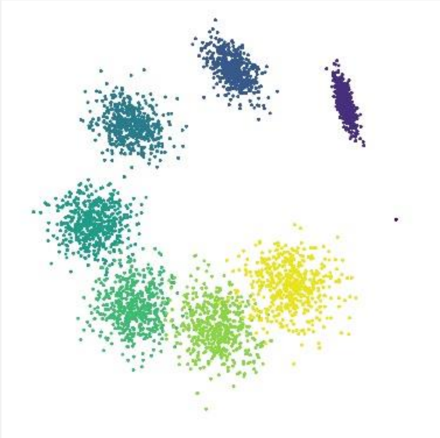
$$z_{t+1} = f_z(z_t, u_t)$$



Comparison state-space and new state-space

State-space

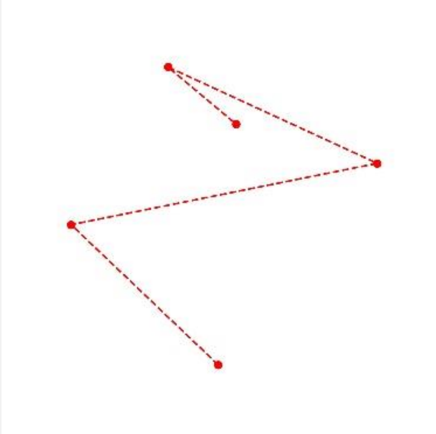
$$\boldsymbol{x}_{t+1} = A\boldsymbol{x}_t + Bu_t + \boldsymbol{e}_t$$



- Stochastic state transition
- Linear

New state-space

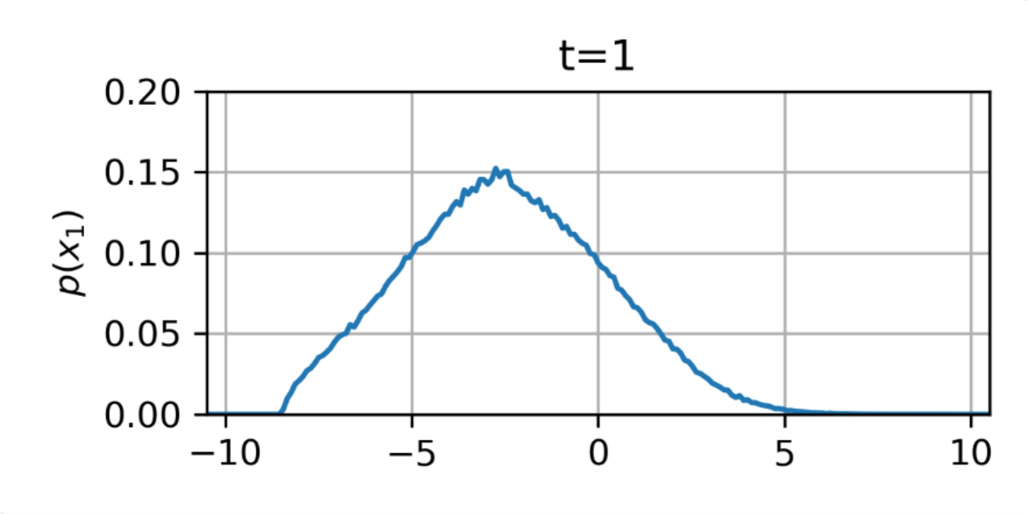
$$z_{t+1} = f_z(z_t, u_t)$$

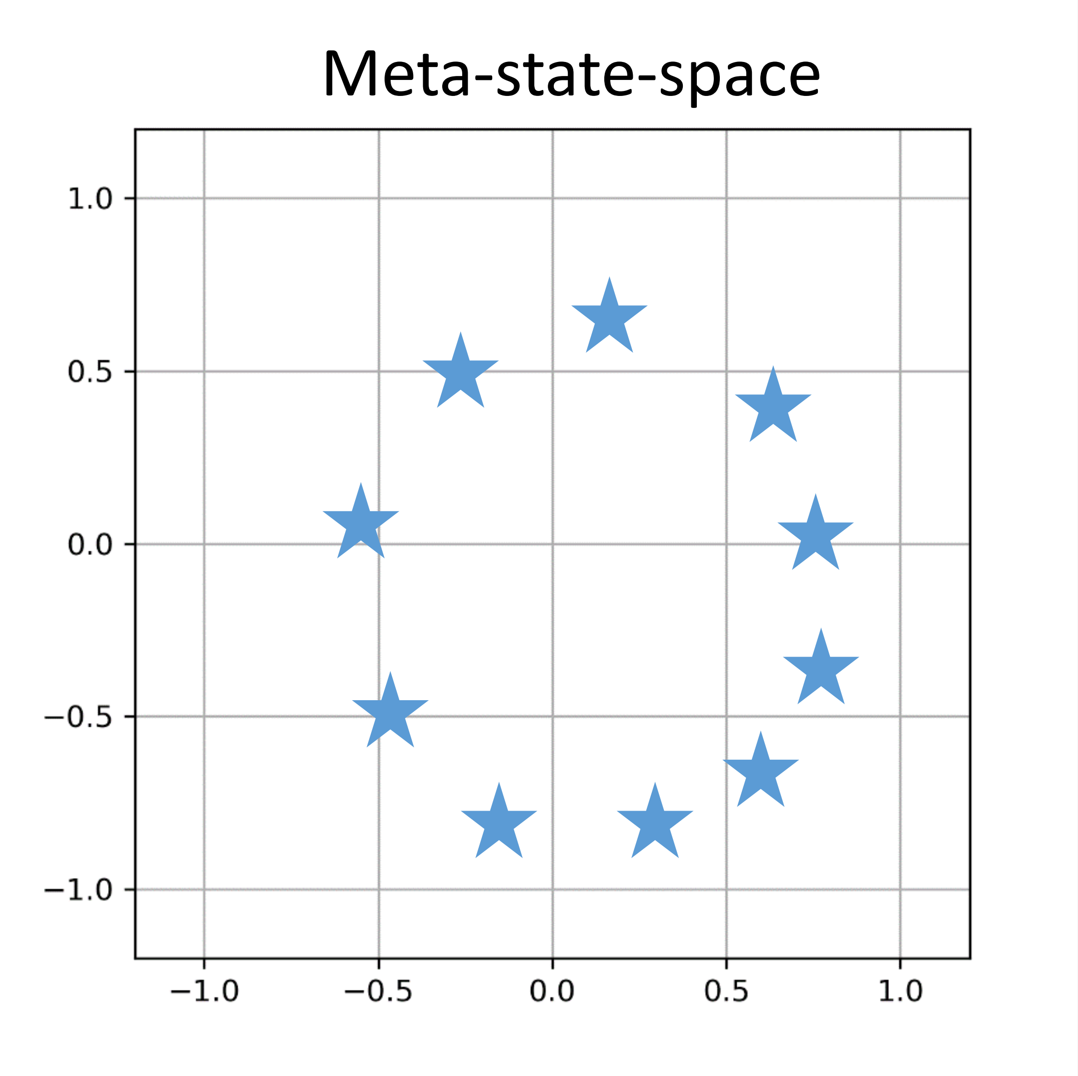


- Deterministic state transition
- Nonlinear

Same IO behaviour $p(\boldsymbol{y}_t)$

The parameterization is a mapping to a new space



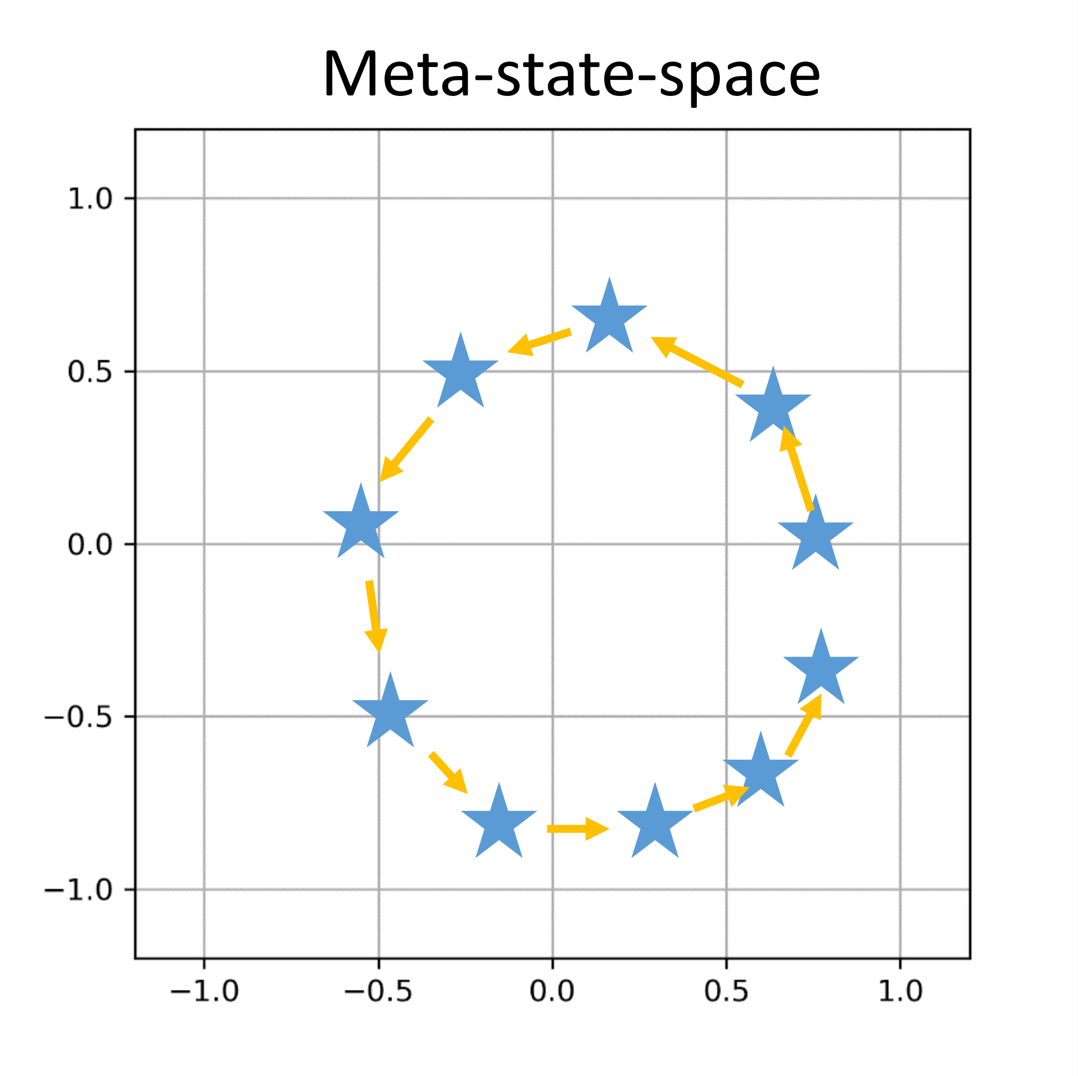


Connecting the dots…

$$z_{t+1} = f_z(z_t, u_t)$$

z_t: meta-state-space

Next a mathematical derivation



$$p(\boldsymbol{x}_{t+1}) = \int p^F(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, u_t)p(\boldsymbol{x}_t)d\boldsymbol{x}_t$$

$$p_t(\boldsymbol{x}) = p(\boldsymbol{x}_t)$$

$$p_{t+1}(\boldsymbol{x}) = \int p^F(\boldsymbol{x}|\boldsymbol{x}', u_t)p_t(\boldsymbol{x}')d\boldsymbol{x}'$$

Functional notation

$$p_{t+1} = F(p_t, u_t)$$

$$p_0 \overset{F}{\underset{u_0}{\longrightarrow}} p_1 \overset{F}{\underset{u_1}{\longrightarrow}} p_2 \overset{F}{\underset{u_2}{\longrightarrow}} p_3 \overset{F}{\underset{u_3}{\longrightarrow}} \dots \qquad p_{t+1} = F(p_t, u_t)$$

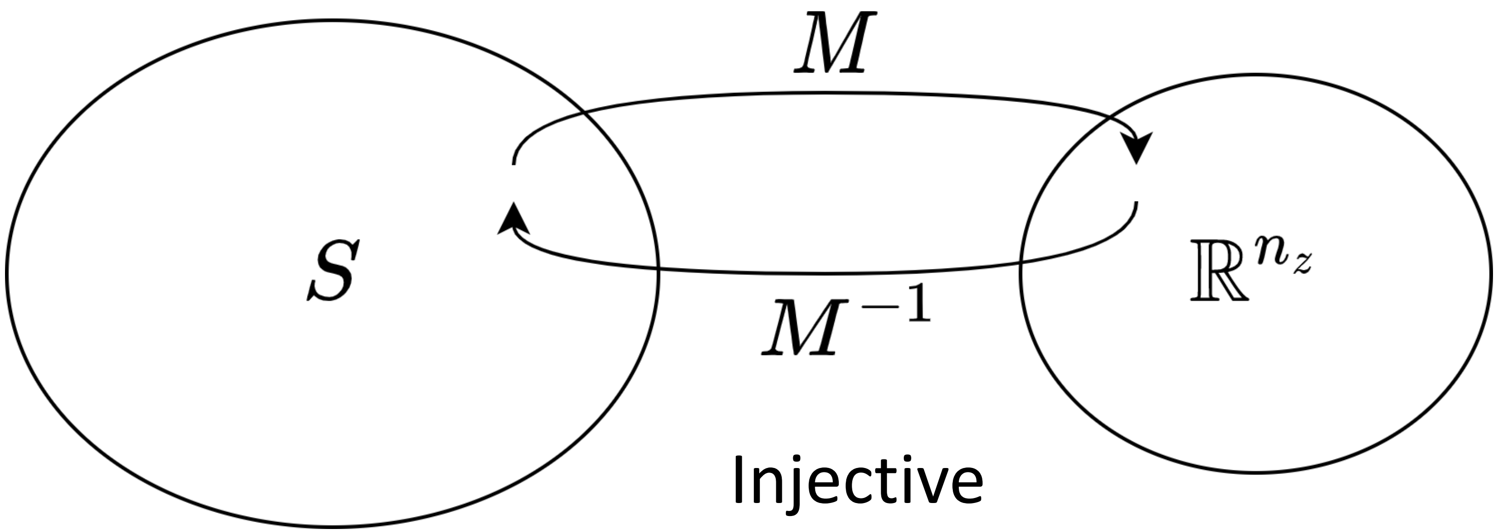
$$\text{Collect: } S = \{p_0, p_1, p_2, \dots\}$$

Parameter vector z_t

$$z_t = M(p_t)$$

$$p_t = M^{-1}(z_t)$$

Assume existence mapping/parametrization



Injective

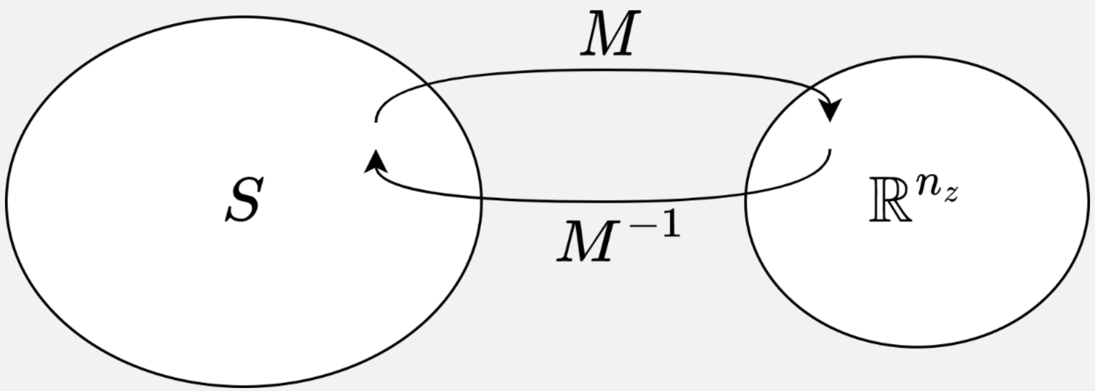
$$p_{t+1} = F(p_t, u_t)$$

$\downarrow$ $p_t = M^{-1}(z_t)$

$$M^{-1}(z_{t+1}) = F(M^{-1}(z_t), u_t)$$

M $\downarrow$ Both sides

$$z_{t+1} = M(F(M^{-1}(z_t), u_t)) = f_z(z_t, u_t)$$



$$p_t^y(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{x}, u_t)p_t(\boldsymbol{x})d\boldsymbol{x}$$

$$p_t^y = H(p_t, u_t) = H(M^{-1}(z_t), u_t) = h(z_t, u_t)$$

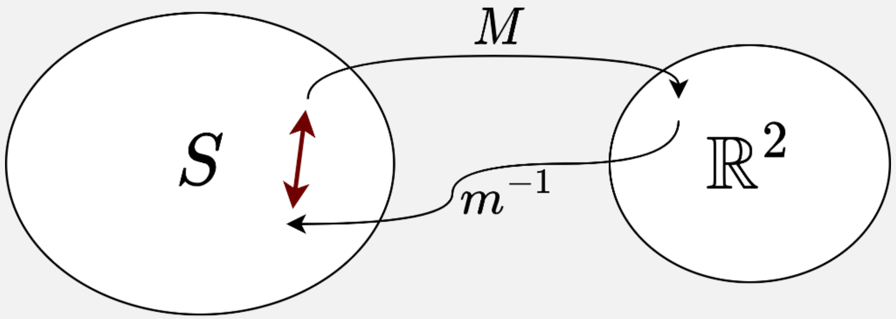
$$z_{t+1} = f_z(z_t, u_t)$$

$$p(\boldsymbol{y}_t|z_t, u_t)$$

State-space	Meta-state-space
$\boldsymbol{x}_{t+1} = f_x(\boldsymbol{x}_t, u_t, \boldsymbol{e}_t)$ $\boldsymbol{y}_t = h(\boldsymbol{x}_t, u_t, \boldsymbol{v}_t)$	$z_{t+1} = f_z(z_t, u_t)$ $p(\boldsymbol{y}_t \mid z_t, u_t)$
Same IO behaviour	
Stochastic dynamics	Deterministic dynamics
Order n_x	Order n_z (often $> n_x$)

How harsh of an assumption is it?

Increasing n_z



Choose M to be any universal approximator

- More moments
- More particles
- More basis functions

Differences get arbitrarily small

Meta-state-space for identification

Meta-state-space is interesting in a mathematical sense

→

How can we make use of meta-state-space for system identification?

Meta-state-space for identification

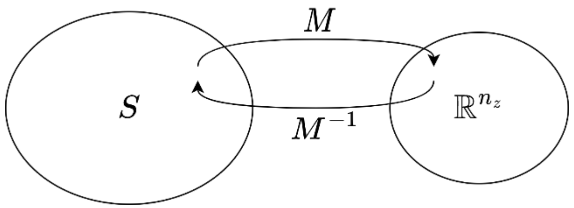
Parameterize meta-state-space

$$z_{t+1} = f_\theta(\hat{z}_t, u_t)$$

$$p_\theta(\hat{\boldsymbol{y}}_t | z_t, u_t)$$

E.g. θ is the network parameters
z_t and u_t is the input to that network.

We never define!



Dataset

$$Y_N^* = \{y_1^*, y_2^*, \ldots, y_N^*\},$$

$$U_N = \{u_1, u_2, \ldots, u_N\},$$

Note: no joint output probabilities $p(y_1, y_2)$
Maximum a posteriori (MAP) estimation

$$\min_{\theta, \hat{z}_1} -\frac{1}{N}\sum_{t=1}^{N} \log(p_\theta(y_t^* | U_t, \hat{z}_1))$$

$$\text{s.t} \quad \hat{z}_{t+1} = f_\theta(\hat{z}_t, u_t)$$

Simulation example

$$\boldsymbol{y}_t = \boldsymbol{x}_t$$
$$\boldsymbol{x}_{t+1} = \alpha(\boldsymbol{x}_t, \boldsymbol{e}_t)\boldsymbol{x}_t + u_t$$
$$\alpha(\boldsymbol{x}_t, \boldsymbol{e}_t) = 0.7\exp(-(\boldsymbol{x}_t + \boldsymbol{e}_t)^2) + 0.3$$
$$|\alpha(\boldsymbol{x}_t, \boldsymbol{e}_t)| \leq 1$$

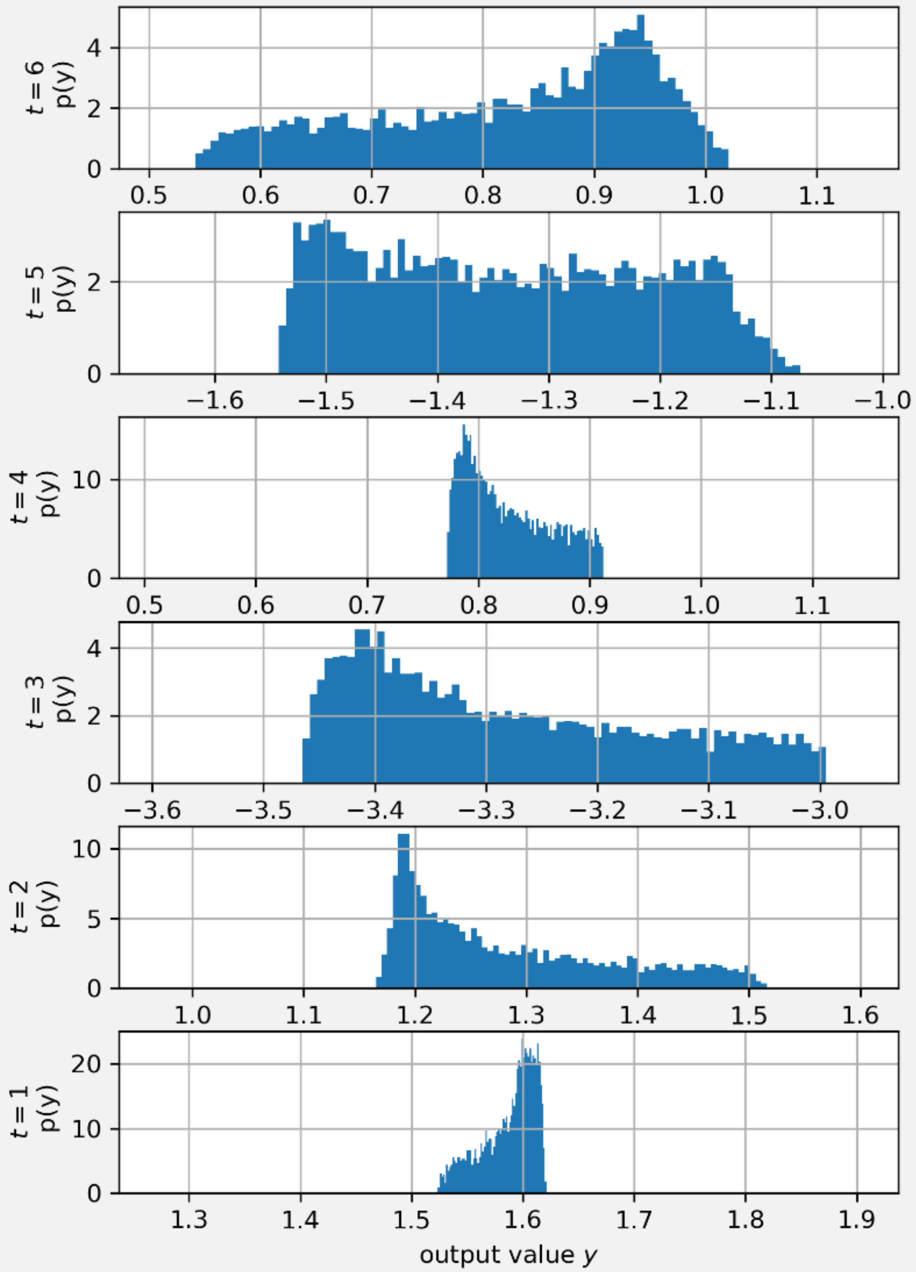
e_t white uniform from -0.5 to 0.5

u_t white normal $\sigma_u = 2$

Data:

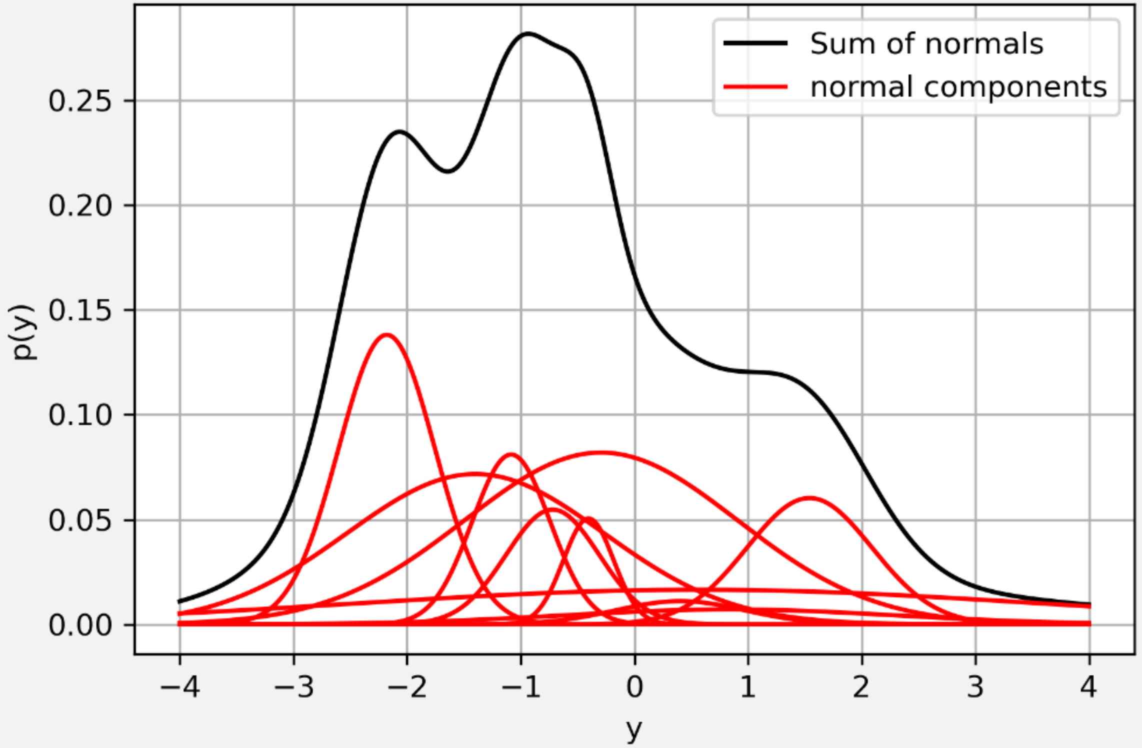
Training: 1 sequence of 300K

Test: 5000 trajectories of 4K



Output distribution parameterization

Sum of normal distributions



Where the weight, mean and std are neural networks with the appropriate constraints.

$$p_\theta(\boldsymbol{y}|z) = \sum_{i=1}^{n_{\text{normals}}} w_{i,\theta}(z)\mathcal{N}(\boldsymbol{y}|\mu_{i,\theta}(z), \sigma_{i,\theta}(z)),$$

$$\text{s.t.} \sum_{i=1}^{n_{\text{normals}}} w_j(z) = 1,\ w_i \geq 0,\ \sigma_j \geq 0,\ \forall z, j$$

Also a ANN: $\hat{z}_{t+1} = f_\theta(z_t, u_t)$

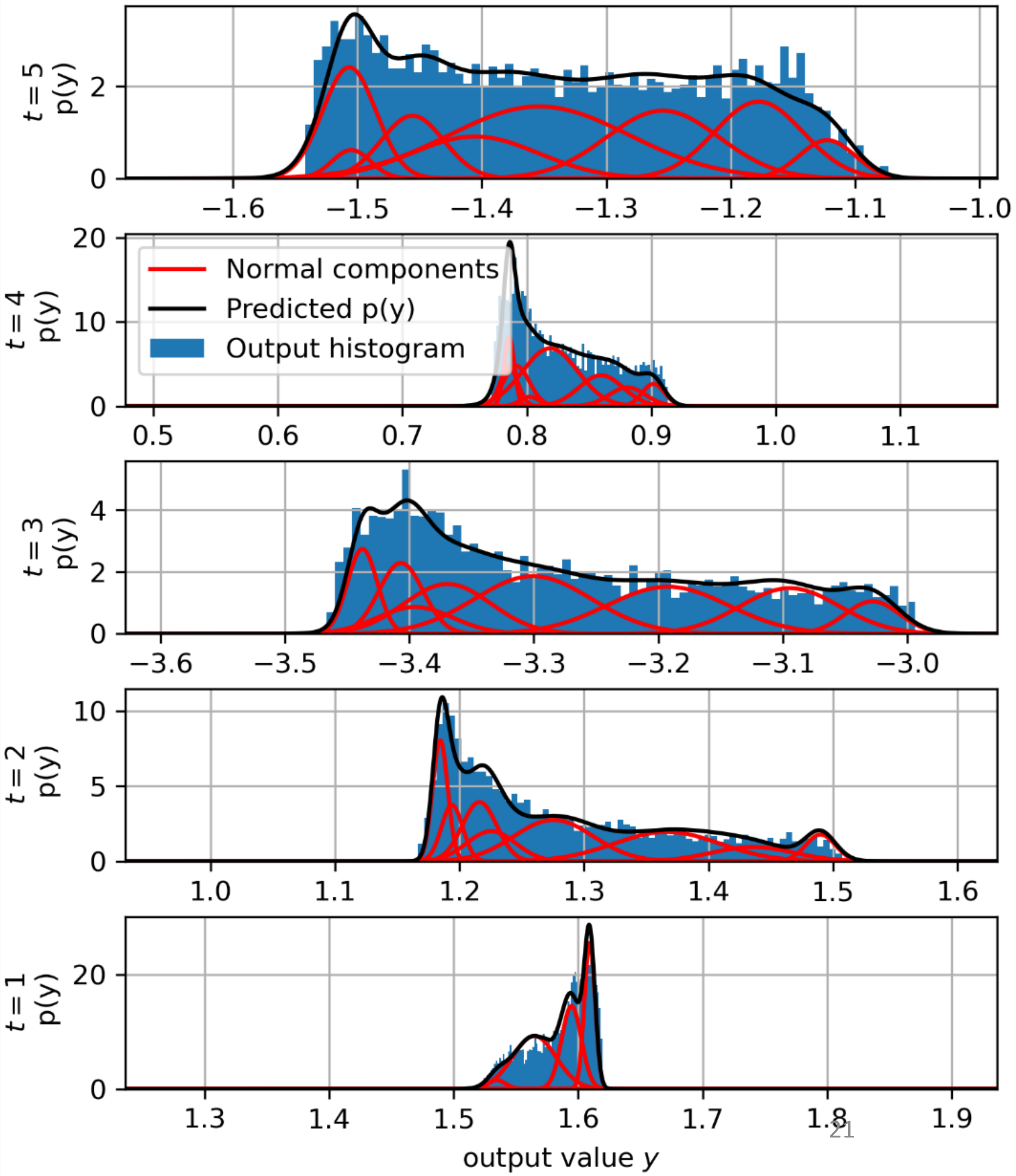
$n_z = 3$

Results: Visual inspection

Free-run simulation results

Meta-state-space model close to the system

Many small details captured!



TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Results: performance

Measure:

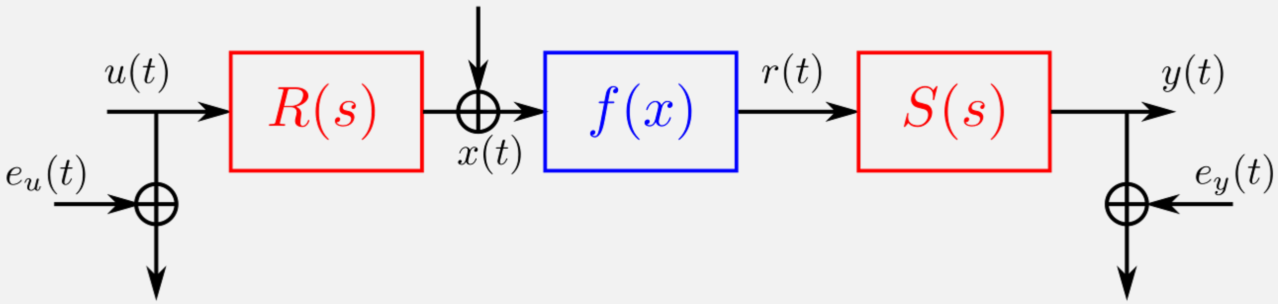
Mean log-likelihood (larger is better)

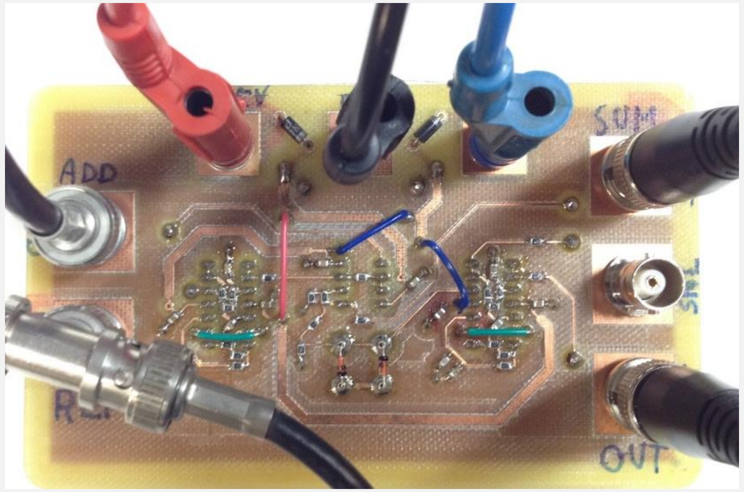
$$\frac{1}{N}\sum_{t=1}^{N}\log(p_\theta(y_t^*|U_t,\hat{z}_1))$$

Meta-state-space model is close to the theoretical limit!

Model/Baseline	Mean log-likelihood
Fixed mean, Fixed output noise Gaussian	-2.18
Modelled mean, Fixed output noise Gaussian	1.04
Modelled mean, Modelled noise Gaussian	1.56
Meta-state-space model	**1.67**
Est. theoretical limit	1.73

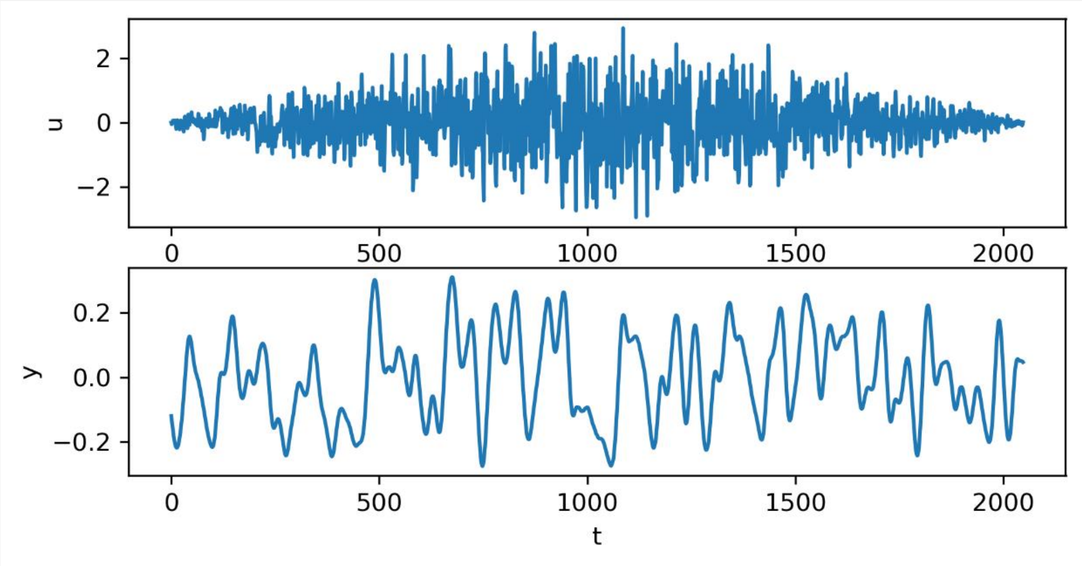
Benchmark Results





Wiener-Hammerstein Process Noise[1]

Only use triangular-shaped multisine data

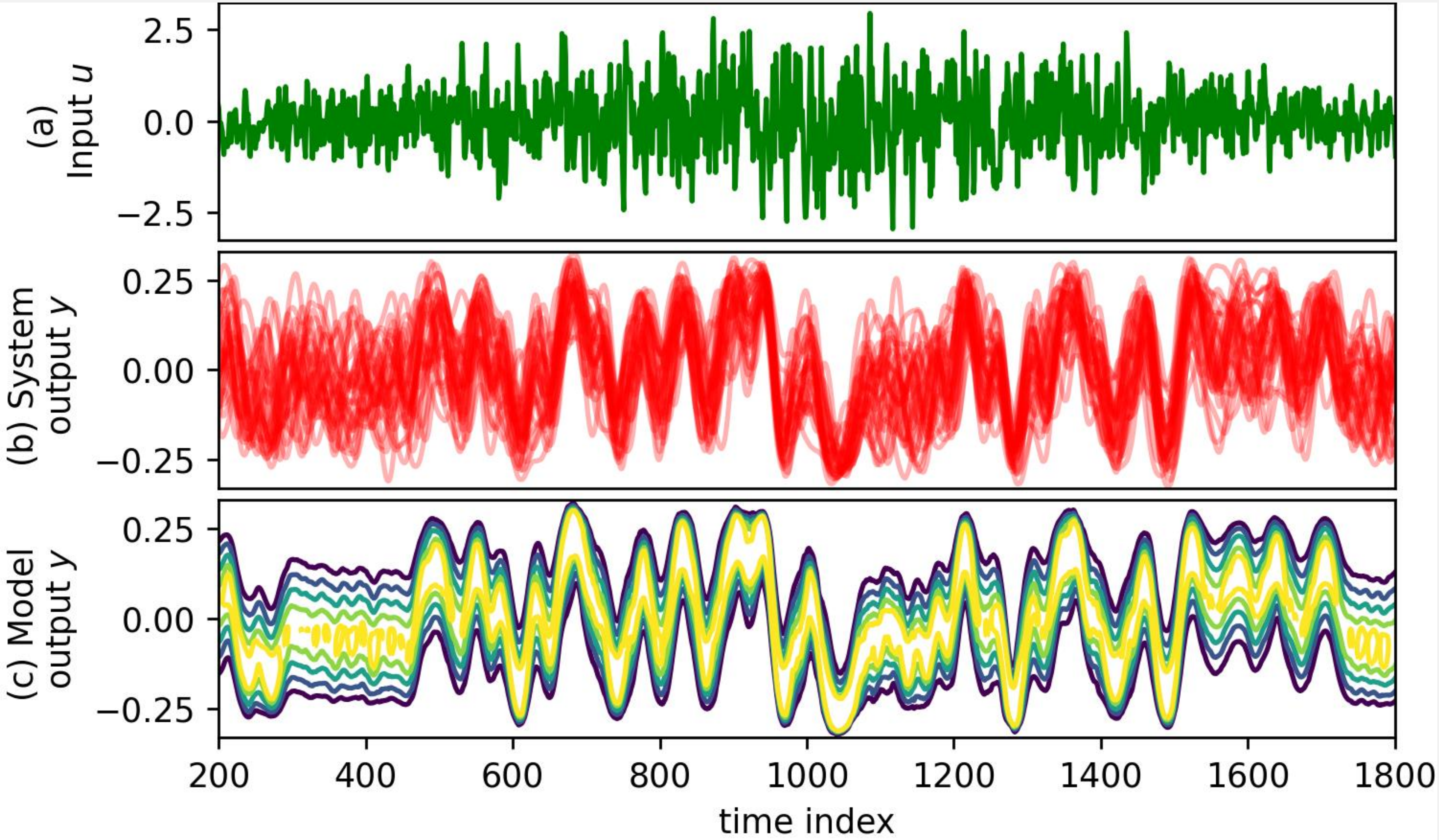


76 sequences with total samples 498K

Similar setup as before, $n_z = 9$

[1] M. Schoukens and J.P. Noël, Three Benchmarks Addressing Open Challenges in Nonlinear System Identification, *20th World Congress of the International Federation of Automatic Control*, pp.448-453, Toulouse, France, July 9-14, 2017, doi: 10.1016/j.ifacol.2017.08.071.

Benchmark Results



Model/Baseline	Mean log-likelihood
Fixed mean, Fixed output noise Gaussian	0.64
Meta-state-space model	**0.99**
Limit	<1.1

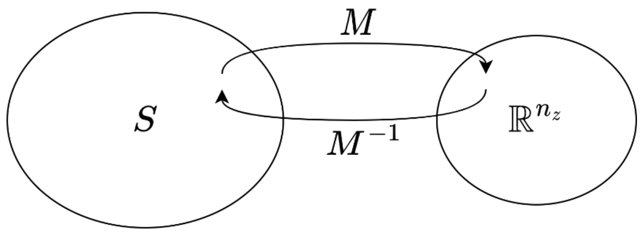
Take-aways form this presentation

Meta-state-space representation

$$z_{t+1} = f_z(z_t, u_t)$$
$$p(\boldsymbol{y}_t | z_t, u_t)$$

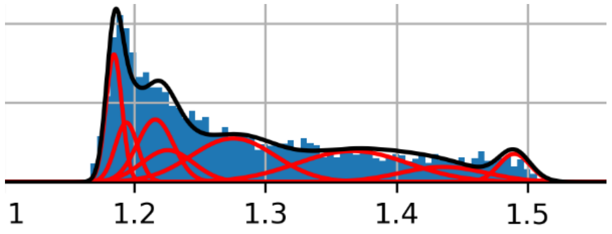
- Exact representation
- Determenistic

Derived by mapping state function space



z_t as parameterization of the state distribution function space

Effective identification exists



ANN Model with MAP close to theoretical limit

Pre-print: https://arxiv.org/abs/2307.06675
Toolbox: https://github.com/GerbenBeintema/metaSI

TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY