High-Dimensional Learning for Conditioning

Youssef Marzouk, joint work with
Ricardo Baptista, Alessio Spantini, & Olivier Zahm

Department of Aeronautics and Astronautics
Center for Computational Science and Engineering
Statistics and Data Science Center
Massachusetts Institute of Technology
http://uqgroup.mit.edu

ANSRE MURI Kickoff Meeting.

Support from the OSD/Air Force under award number FA9550-20-1-0397.

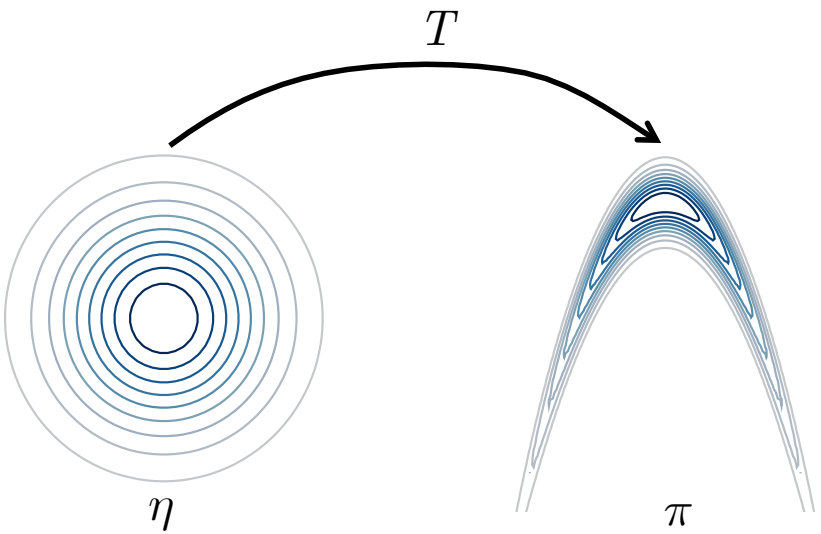
16 February 2021

Problems of interest

High-dimensional learning for *conditioning* is central to:

- Bayesian inference in **stochastic models**
- **Data assimilation** (filtering, smoothing, prediction) in dynamical systems
- **Likelihood-free** ("simulation-based") inference: when closed-form density functions are not available
- Characterizing **rare events** in all of these contexts. . .

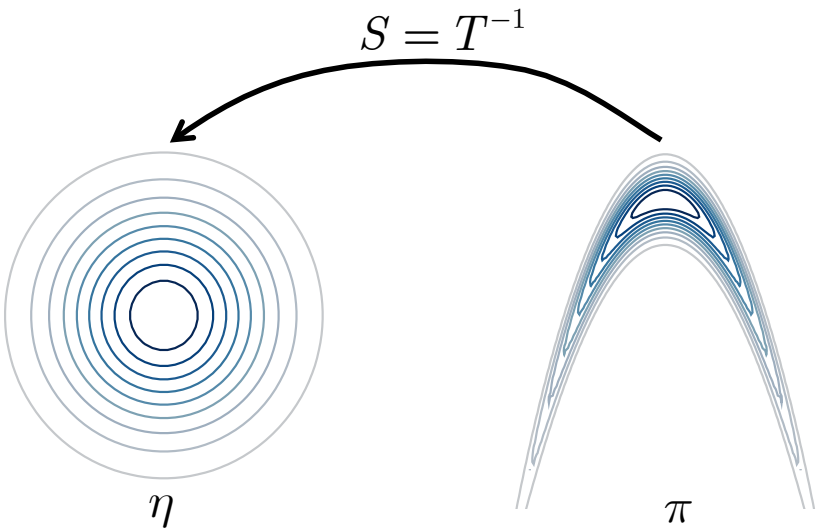
Tool: deterministic couplings of probability measures



Core idea

- Choose a *reference distribution* η (e.g., standard Gaussian)
- Seek a transport map $T : \mathbb{R}^d \to \mathbb{R}^d$ such that $T_\sharp \eta = \pi$

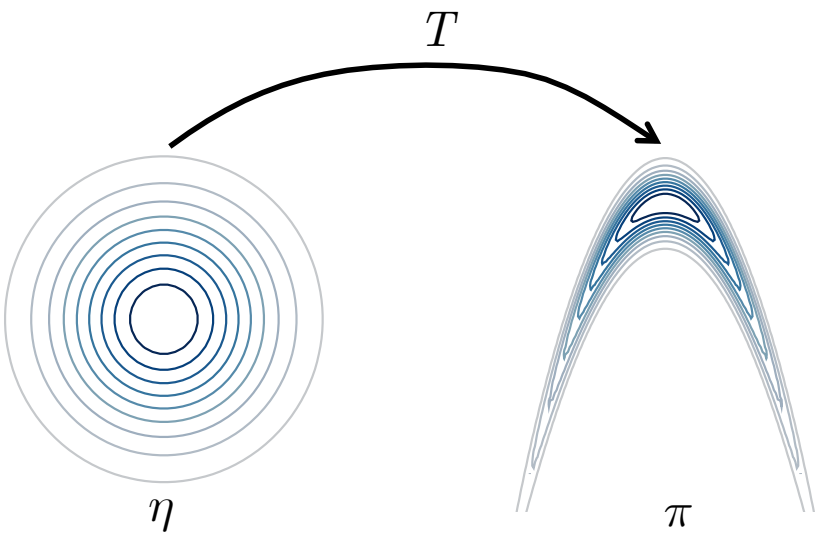
Tool: deterministic couplings of probability measures



Core idea

- Choose a *reference distribution* η (e.g., standard Gaussian)
- Seek a transport map $T : \mathbb{R}^d \to \mathbb{R}^d$ such that $T_\sharp \eta = \pi$
- Equivalently, find $S = T^{-1}$ such that $S_\sharp \pi = \eta$

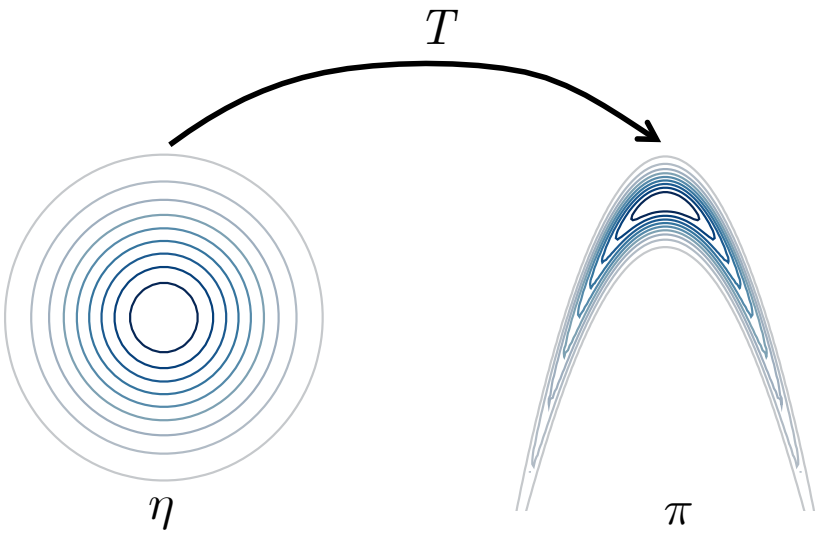
Tool: deterministic couplings of probability measures



Core idea

- Choose a *reference distribution* η (e.g., standard Gaussian)
- Seek a transport map $T : \mathbb{R}^d \to \mathbb{R}^d$ such that $T_\sharp \eta = \pi$
- Equivalently, find $S = T^{-1}$ such that $S_\sharp \pi = \eta$
- In principle, enables *exact* (independent, unweighted) sampling!

Tool: deterministic couplings of probability measures



Core idea

- Choose a *reference distribution* η (e.g., standard Gaussian)
- Seek a transport map $T : \mathbb{R}^d \to \mathbb{R}^d$ such that $T_\sharp \eta = \pi$
- Equivalently, find $S = T^{-1}$ such that $S_\sharp \pi = \eta$
- Satisfying these conditions only **approximately** can still be useful!

Choice of transport map

Consider the triangular **Knothe-Rosenblatt rearrangement** on $\mathbb{R}^d$

$$S(\mathbf{x}) = \begin{bmatrix} S^1(x_1) \\ S^2(x_1, x_2) \\ \vdots \\ S^d(x_1, x_2, \ldots, x_d) \end{bmatrix}$$

1. Unique S s.t. $S_\sharp \pi = \eta$ exists under mild conditions on π and η
2. Map is easily invertible and Jacobian ∇S is simple to evaluate
3. Monotonicity is essentially one-dimensional: $\partial_{x_k} S^k > 0$
4. Each component S^k characterizes one marginal conditional

$$\pi(\mathbf{x}) = \pi(x_1)\pi(x_2|x_1)\cdots\pi(x_d|x_1, \ldots, x_{d-1})$$

From conditional simulation to inference

- Suppose we now have parameters $\mathbf{X} \in \mathbb{R}^n$ and data $\mathbf{Y} \in \mathbb{R}^m$, and joint prior model $\pi_{\mathbf{Y},\mathbf{X}}$. Seek the KR map S that pushes $\pi_{\mathbf{Y},\mathbf{X}}$ to $\mathcal{N}(0, \mathbf{I}_{m+n})$
- The KR map immediately has a block structure

$$S(\mathbf{y}, \mathbf{x}) = \begin{bmatrix} S^{\mathbf{Y}}(\mathbf{y}) \\ S^{\mathbf{X}}(\mathbf{y}, \mathbf{x}) \end{bmatrix},$$

which suggests two properties:

$$S^{\mathbf{X}} \text{ pushes } \pi_{\mathbf{Y},\mathbf{X}} \text{ to } \mathcal{N}(0, \mathbf{I}_n)$$

$$\boldsymbol{\xi} \mapsto S^{\mathbf{X}}(\mathbf{y}^*, \boldsymbol{\xi}) \text{ pushes } \pi_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^*} \text{ to } \mathcal{N}(0, \mathbf{I}_n)$$

1. Approximate the conditional **density**:

$$\boxed{\widehat{\pi}_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^*} = \widehat{S}^{\mathbf{X}}(\mathbf{y}^*, \cdot)^\sharp \mathcal{N}(0, \mathbf{I}_n)}$$

From conditional simulation to inference

- Suppose we now have parameters $\mathbf{X} \in \mathbb{R}^n$ and data $\mathbf{Y} \in \mathbb{R}^m$, and joint prior model $\pi_{\mathbf{Y},\mathbf{X}}$. Seek the KR map S that pushes $\pi_{\mathbf{Y},\mathbf{X}}$ to $\mathcal{N}(0, \mathbf{I}_{m+n})$
- The KR map immediately has a block structure

$$S(\mathbf{y}, \mathbf{x}) = \begin{bmatrix} S^{\mathbf{Y}}(\mathbf{y}) \\ S^{\mathbf{X}}(\mathbf{y}, \mathbf{x}) \end{bmatrix},$$

which suggests two properties:

$$S^{\mathbf{X}} \text{ pushes } \pi_{\mathbf{Y},\mathbf{X}} \text{ to } \mathcal{N}(0, \mathbf{I}_n)$$

$$\boldsymbol{\xi} \mapsto S^{\mathbf{X}}(\mathbf{y}^*, \boldsymbol{\xi}) \text{ pushes } \pi_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^*} \text{ to } \mathcal{N}(0, \mathbf{I}_n)$$

2. **Sample** the conditional distribution $\widehat{\pi}_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^*}$:

$$\boxed{\text{invert } \widehat{S}^{\mathbf{X}}(\mathbf{y}^*, \mathbf{x}^i) = \boldsymbol{\xi}^i \text{ for } \mathbf{x}^i \text{ given } \boldsymbol{\xi}^i \sim \mathcal{N}(0, \mathbf{I}_n)}$$

From conditional simulation to inference

- Suppose we now have parameters $\mathbf{X} \in \mathbb{R}^n$ and data $\mathbf{Y} \in \mathbb{R}^m$, and joint prior model $\pi_{\mathbf{Y},\mathbf{X}}$. Seek the KR map S that pushes $\pi_{\mathbf{Y},\mathbf{X}}$ to $\mathcal{N}(0, \mathbf{I}_{m+n})$
- The KR map immediately has a block structure

$$S(\mathbf{y}, \mathbf{x}) = \begin{bmatrix} S^{\mathbf{Y}}(\mathbf{y}) \\ S^{\mathbf{X}}(\mathbf{y}, \mathbf{x}) \end{bmatrix},$$

which suggests two properties:

$$S^{\mathbf{X}} \text{ pushes } \pi_{\mathbf{Y},\mathbf{X}} \text{ to } \mathcal{N}(0, \mathbf{I}_n)$$

$$\boldsymbol{\xi} \mapsto S^{\mathbf{X}}(\mathbf{y}^*, \boldsymbol{\xi}) \text{ pushes } \pi_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^*} \text{ to } \mathcal{N}(0, \mathbf{I}_n)$$

3. **Sample** the conditional via a *composed map* T that pushes forward $\pi_{\mathbf{Y},\mathbf{X}}$ to $\pi_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^*}$:

$$\boxed{T(\mathbf{y}, \mathbf{x}) = S^{\mathbf{X}}(\mathbf{y}^*, \cdot)^{-1} \circ S^{\mathbf{X}}(\mathbf{y}, \mathbf{x})}$$

How to construct triangular maps?

Data-driven formulation: learning **"maps from samples"**

- **Given a sample** $(\mathbf{x}^i)_{i=1}^M \sim \pi$**:** find each component function via convex (wrt S^k) constrained minimization (here, for standard Gaussian η):

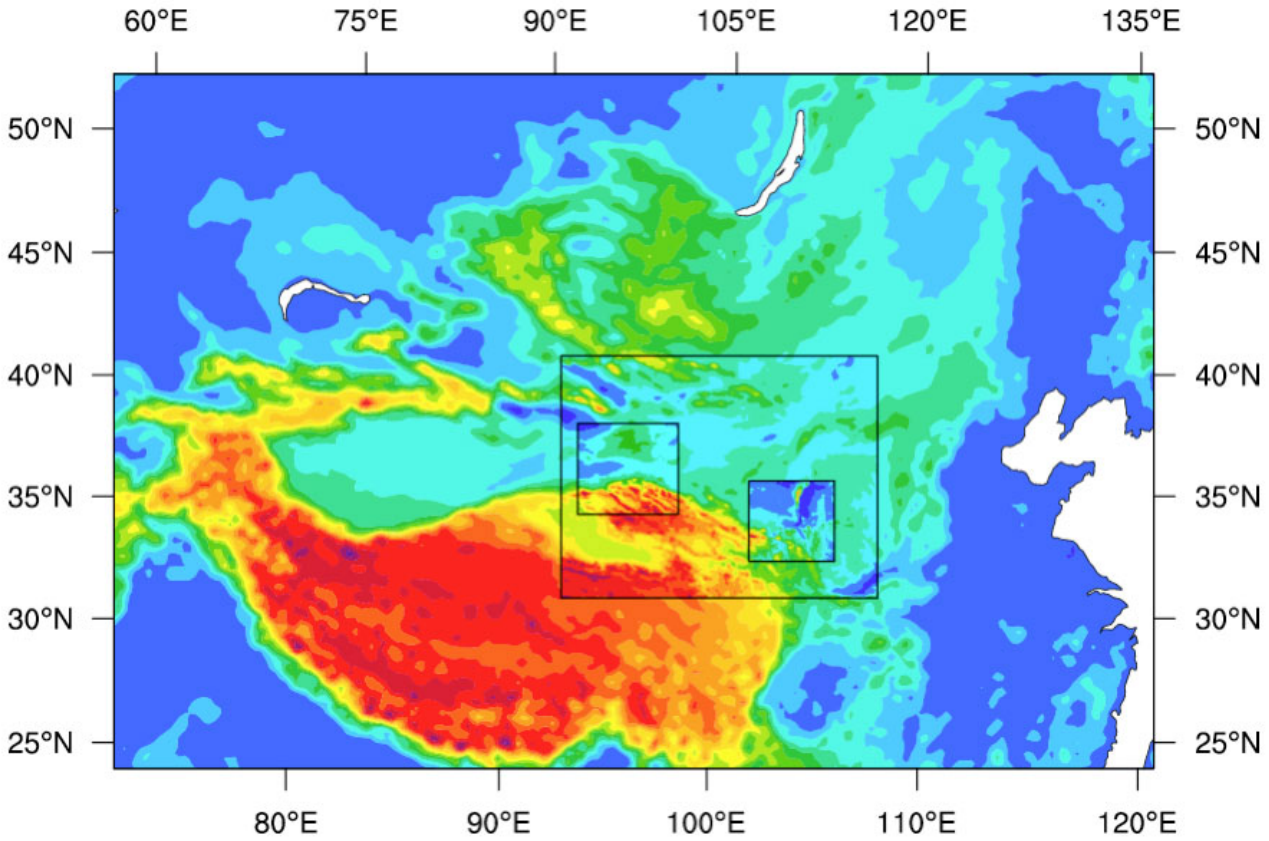
$$\min_S D_{KL}(\pi||S^\sharp\eta) \quad \Leftrightarrow \quad \min_{S^k\,:\,\partial_k S^k>0} \mathbb{E}_\pi\left[\frac{1}{2}S^k(\mathbf{x}_{1:k})^2 - \log\partial_k S^k(\mathbf{x}_{1:k})\right] \;\forall k$$

- Approximate $\mathbb{E}_\pi$ given i.i.d. samples from π: KL minimization equivalent to maximum likelihood estimation

$$\boxed{\widehat{S}^k \in \arg\min_{S^k\in\mathcal{S}^h_{\triangle,k}} \frac{1}{M}\sum_{i=1}^M\left(\frac{1}{2}S^k(\mathbf{x}^i_{1:k})^2 - \log\partial_k S^k(\mathbf{x}^i_{1:k})\right)}$$

Example: data assimilation

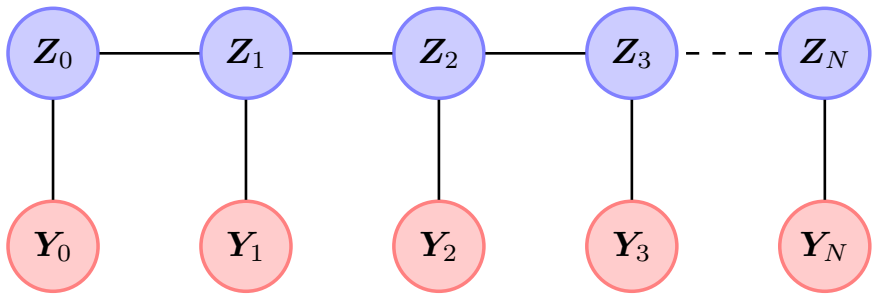
How to solve **sequential inference** problems in dynamical systems, where key density functions cannot be evaluated?



[image: NCAR]

Formalize as Bayesian filtering

- **Nonlinear/non-Gaussian** state-space model:
  - Transition density $\pi_{\mathbf{Z}_k|\mathbf{Z}_{k-1}}$
  - Observation density (likelihood) $\pi_{\mathbf{Y}_k|\mathbf{Z}_k}$



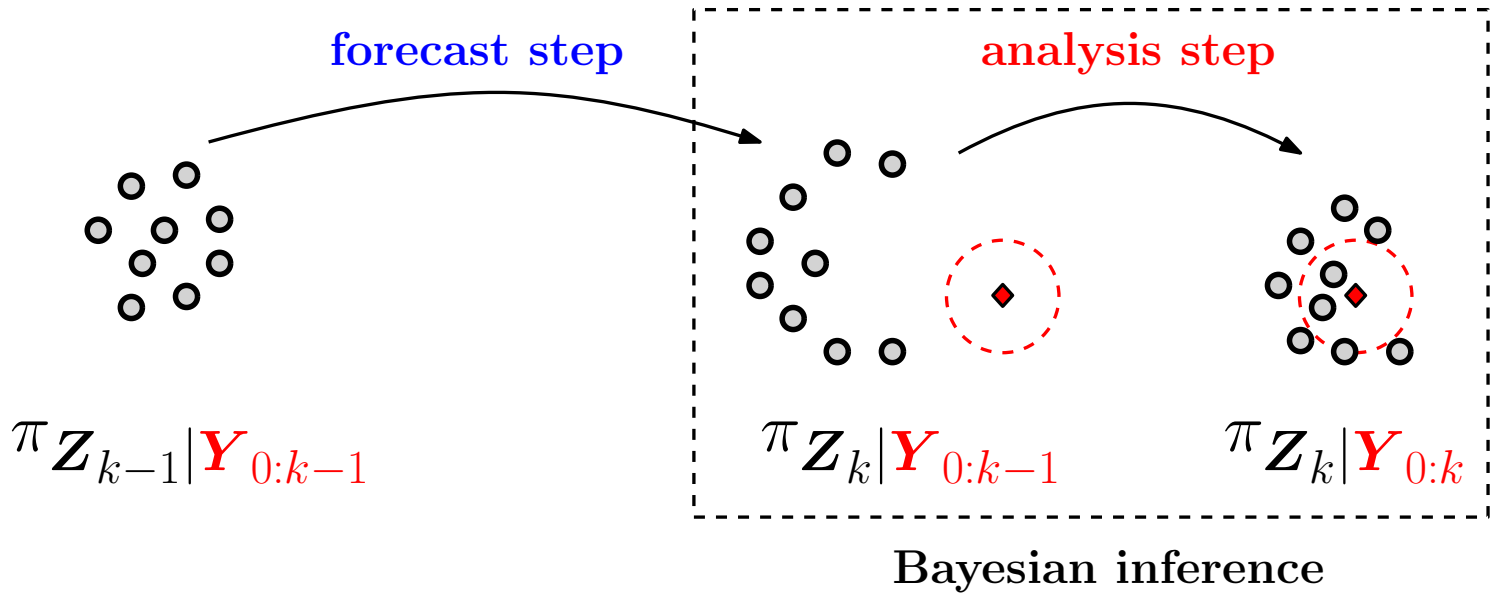
- Focus on recursively approximating the filtering distribution:
  $\pi_{\mathbf{Z}_k \,|\, \mathbf{y}_{0:k}} \rightarrow \pi_{\mathbf{Z}_{k+1} \,|\, \mathbf{y}_{0:k+1}}$ (marginals of the full Bayesian solution)

Problem setting

- Consider the filtering of state-space models with:
  1. High-dimensional states
  2. Challenging nonlinear dynamics
  3. Intractable transition kernels: can only obtain *forecast* samples, i.e., *draws* from $\pi_{\mathbf{z}_{k+1} \mid \mathbf{z}_k}$
  4. Limited model evaluations, e.g., small ensemble sizes
  5. Sparse and local observations

Ensemble Kalman filter

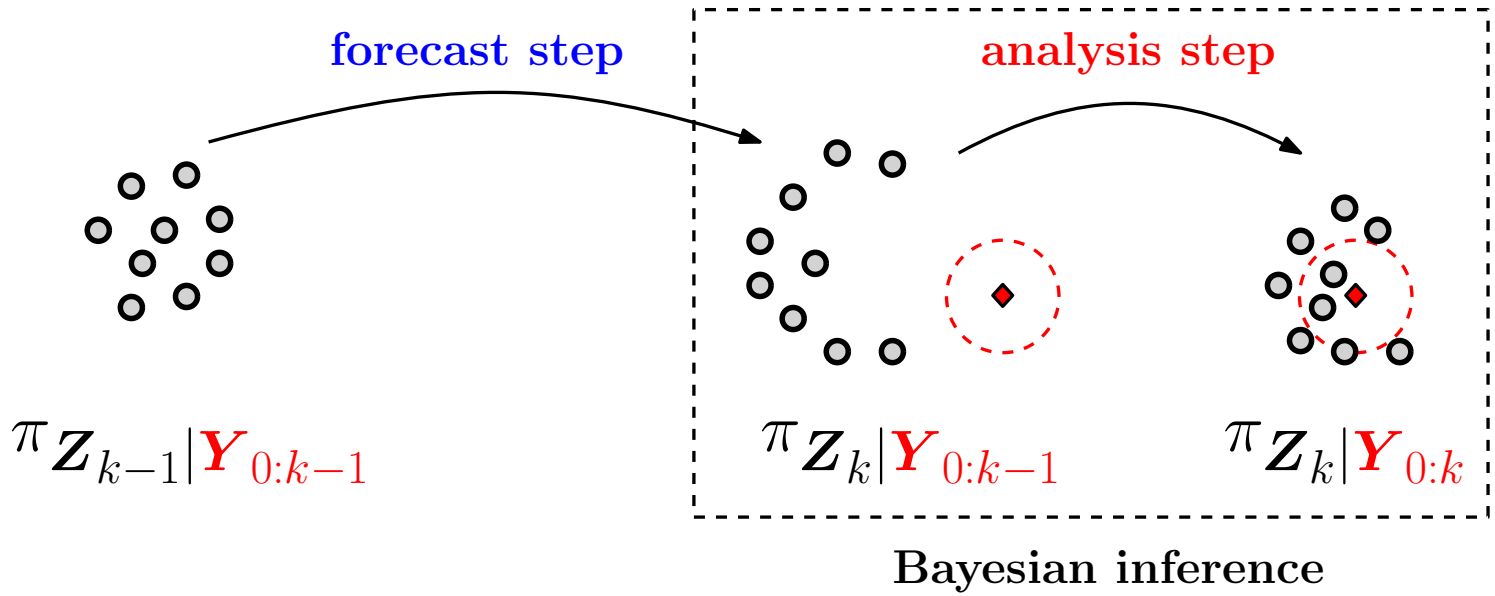
- State-of-the-art results (in terms of tracking) are often obtained with the ensemble Kalman filter (EnKF)



- Move samples via an **affine** transformation; no weights or resampling!
- Yet ultimately **inconsistent**: does not converge to the true posterior

Ensemble Kalman filter

- State-of-the-art results (in terms of tracking) are often obtained with the ensemble Kalman filter (EnKF)

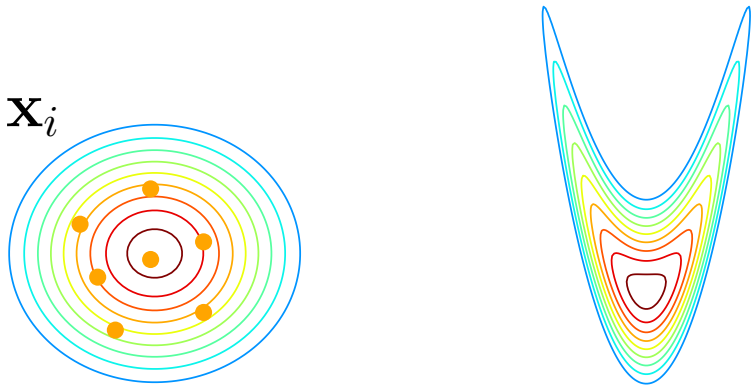


- Move samples via an **affine** transformation; no weights or resampling!
- Yet ultimately **inconsistent**: does not converge to the true posterior

Can we *improve* and *generalize* the EnKF while preserving scalability?

Assimilation step

At any assimilation time k, we have a Bayesian inference problem:



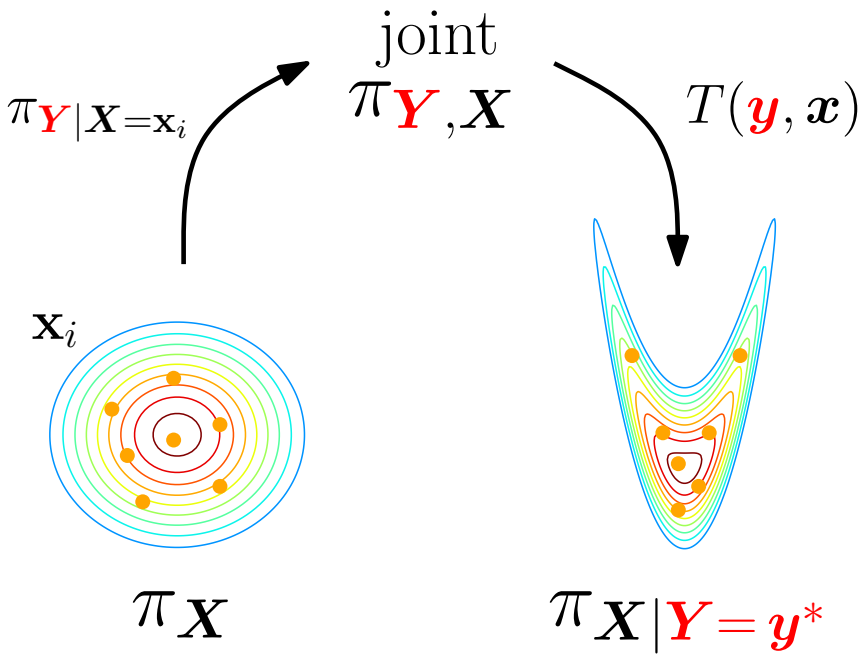
$\pi_{\boldsymbol{X}}$ prior

$\pi_{\boldsymbol{X}|\boldsymbol{Y}=\boldsymbol{y}^*}$ posterior

- $\pi_{\mathbf{X}}$ is the forecast distribution on $\mathbb{R}^n$
- $\pi_{\mathbf{Y}|\mathbf{X}}$ is the likelihood of the observations $\mathbf{Y} \in \mathbb{R}^d$
- $\pi_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^*}$ is the filtering distribution for a realization $\mathbf{y}^*$ of the data

Goal: sample the posterior given only *(few)* prior samples $\mathbf{x}_1, \ldots, \mathbf{x}_M$ and the ability to simulate data $\mathbf{y}_i|\mathbf{x}_i$

A likelihood-free inference algorithm with maps



Transport map ensemble filter

1. Compute forecast ensemble $\mathbf{x}_1, \ldots, \mathbf{x}_M$
2. Generate samples $(\mathbf{y}_i, \mathbf{x}_i)$ from $\pi_{\mathbf{Y},\mathbf{X}}$ with $\mathbf{y}_i \sim \pi_{\mathbf{Y}|\mathbf{X}=\mathbf{x}_i}$
3. Build an estimator $\widehat{T}$ of T
4. Compute analysis ensemble as $\mathbf{x}_i^{\mathrm{a}} = \widehat{T}(\mathbf{y}_i, \mathbf{x}_i)$ for $i = 1, \ldots, M$

Estimator for the analysis map

- Recall the form of S:

$$S(\mathbf{y}, \mathbf{x}) = \begin{bmatrix} S^{\mathbf{Y}}(\mathbf{y}) \\ S^{\mathbf{X}}(\mathbf{y}, \mathbf{x}) \end{bmatrix}, \qquad S_\sharp \, \pi_{\mathbf{Y},\mathbf{X}} = \mathcal{N}(0, \mathbf{I}_{d+n}).$$

- We propose a simple estimator $\widehat{T}$ of T:

$$\boxed{\widehat{T}(\mathbf{y}, \mathbf{x}) = \widehat{S}^{\mathbf{X}}(\mathbf{y}^*, \cdot)^{-1} \circ \widehat{S}^{\mathbf{X}}(\mathbf{y}, \mathbf{x}),}$$

  where $\widehat{S}$ is a **maximum likelihood estimator** of S
- This is simply the "maps from samples" approach!

Map parameterizations

$$\widehat{S}^k \in \arg \min_{S^k \in \mathcal{S}^h_{\triangle,k}} \frac{1}{M} \sum_{i=1}^{M} \left(\frac{1}{2} S^k(\mathbf{x}_i)^2 - \log \partial_k S^k(\mathbf{x}_i) \right)$$

- Optimization is not needed for nonlinear separable parameterizations of the form $\widehat{S}^k(x_{1:k}) = g(x_{1:k-1}) + \alpha x_k$ (just *linear regression*)
- **Connection to EnKF:** a linear parameterization of $\widehat{S}^k$ **recovers** a particular form of EnKF with "perturbed observations"
- Choice of approximation space allows control of the bias and variance of $\widehat{S}$
  - Richer parameterizations yield less bias, but potentially higher variance

Example: Lorenz-63

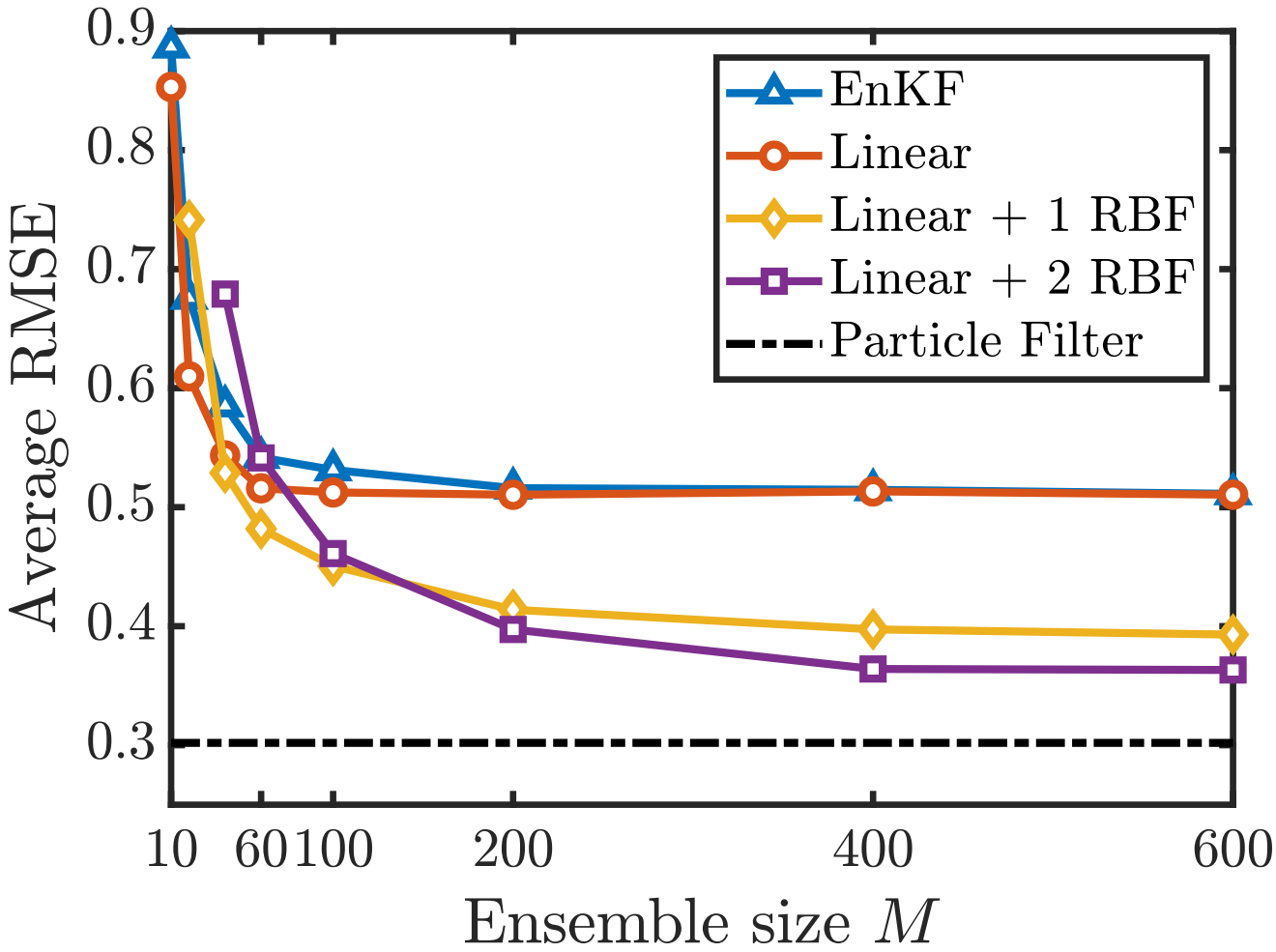
Simple example: three-dimensional Lorenz-63 system

$$\begin{aligned} \frac{\mathrm{d}X_1}{\mathrm{d}t} &= \sigma(X_2 - X_1), \\ \frac{\mathrm{d}X_2}{\mathrm{d}t} &= X_1(\rho - X_3) - X_2 \\ \frac{\mathrm{d}X_3}{\mathrm{d}t} &= X_1 X_2 - \beta X_3 \end{aligned}$$

- Chaotic setting: $\rho = 28$, $\sigma = 10$, $\beta = 8/3$
- Fully observed, with additive Gaussian observation noise $\mathcal{E}_j \sim \mathcal{N}(0, 2^2)$
- Assimilation interval $\Delta t = 0.1$
- Results computed over 2000 assimilation cycles, following spin-up
- **Map parameterizations:** $S^k(x_{1:k}) = \sum_{i \leq k} \Psi_i(x_i)$, with Ψ_i = linear + {RBFs or sigmoids }

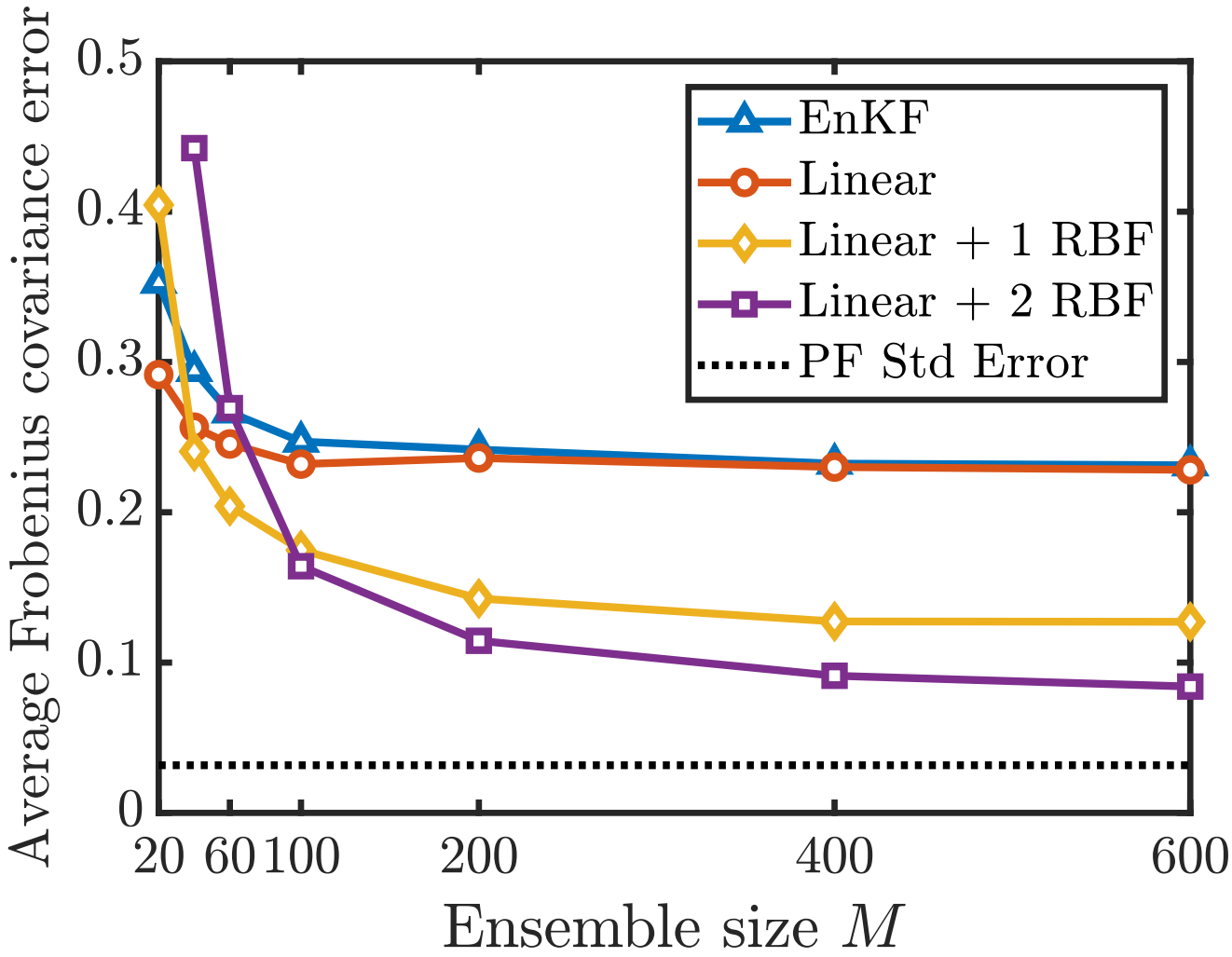
Example: Lorenz-63

Mean "tracking" error vs. ensemble size and choice of map



Example: Lorenz-63

What about comparison to the *true Bayesian solution*?



"Localize" the map in high dimensions

- Regularize the estimator $\widehat{S}$ of S by imposing **sparsity**, e.g.,

$$\widehat{S}(x_1,\dots,x_4) = \begin{bmatrix} \widehat{S}^1(x_1) & & & \\ \widehat{S}^2(x_1,x_2) & & & \\ \widehat{S}^3(& x_2,x_3) & & \\ \widehat{S}^4(& & x_3,x_4) \end{bmatrix}$$

- The sparsity of the kth component of S depends on the sparsity of the marginal conditional function $\pi_{\mathbf{X}_k|\mathbf{X}_{1:k-1}}(x_k|\mathbf{x}_{1:k-1})$
- **Localization heuristic:** let each $\widehat{S}^k$ depend on variables $(x_j)_{j<k}$ that are within a distance ℓ from x_k in state space.
- Explicit link between sparsity of S and **conditional independence** in non-Gaussian graphical models described in

  [Inference via low-dimensional couplings, Spantini/Bigoni/M JMLR 2018]

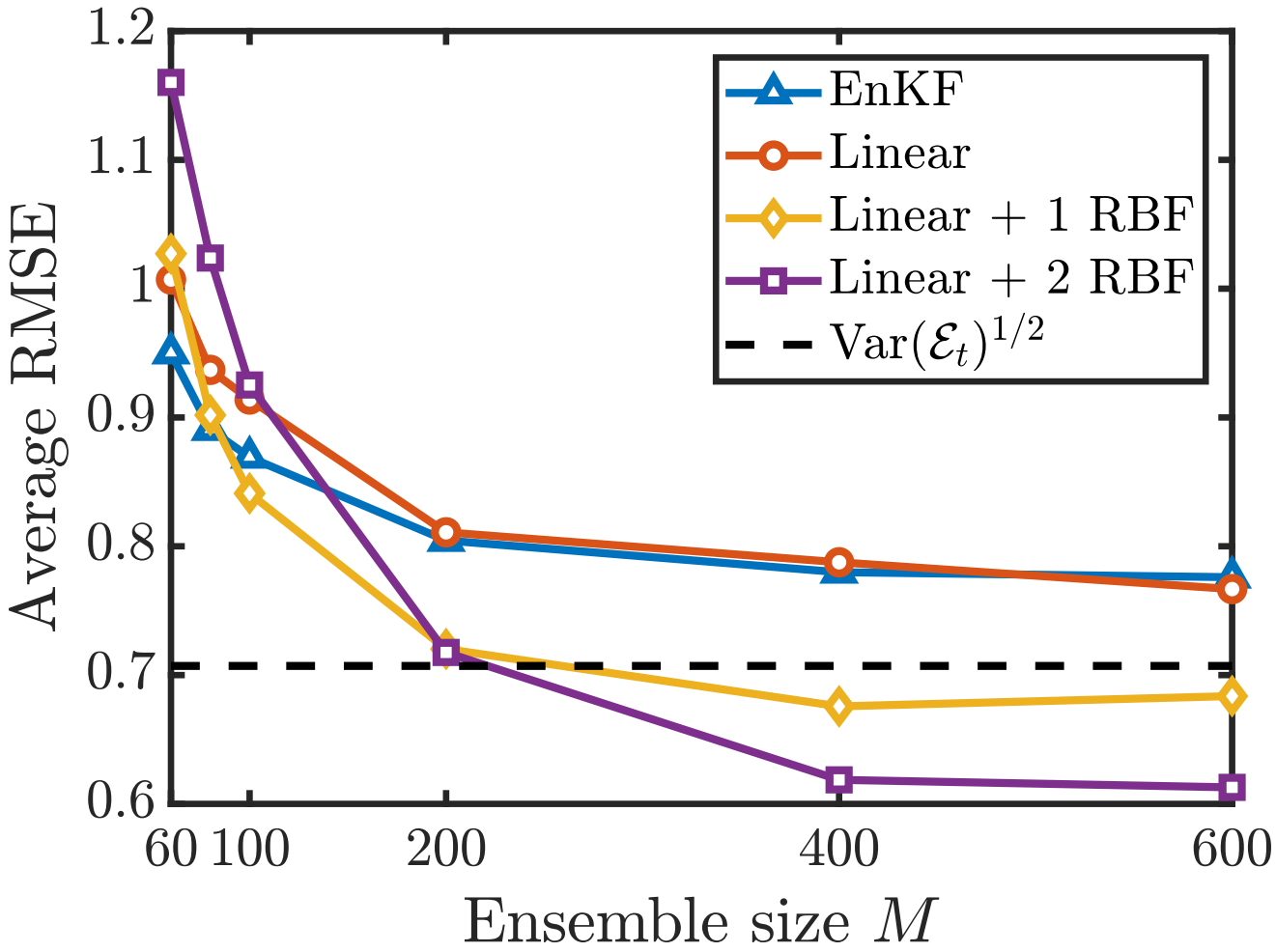
Lorenz-96 in chaotic regime (40-dimensional state)

- A **hard** test-case configuration [Bengtsson et al. 2003]:

$$\begin{aligned} \frac{\mathrm{d}\mathbf{X}_j}{\mathrm{d}t} &= (\mathbf{X}_{j+1} - \mathbf{X}_{j-2})\mathbf{X}_{j-1} - \mathbf{X}_j + F, \qquad j = 1, \ldots, 40 \\ \mathbf{Y}_j &= \mathbf{X}_j + \mathcal{E}_j, \qquad j = 1, 3, 5 \ldots, 39 \end{aligned}$$

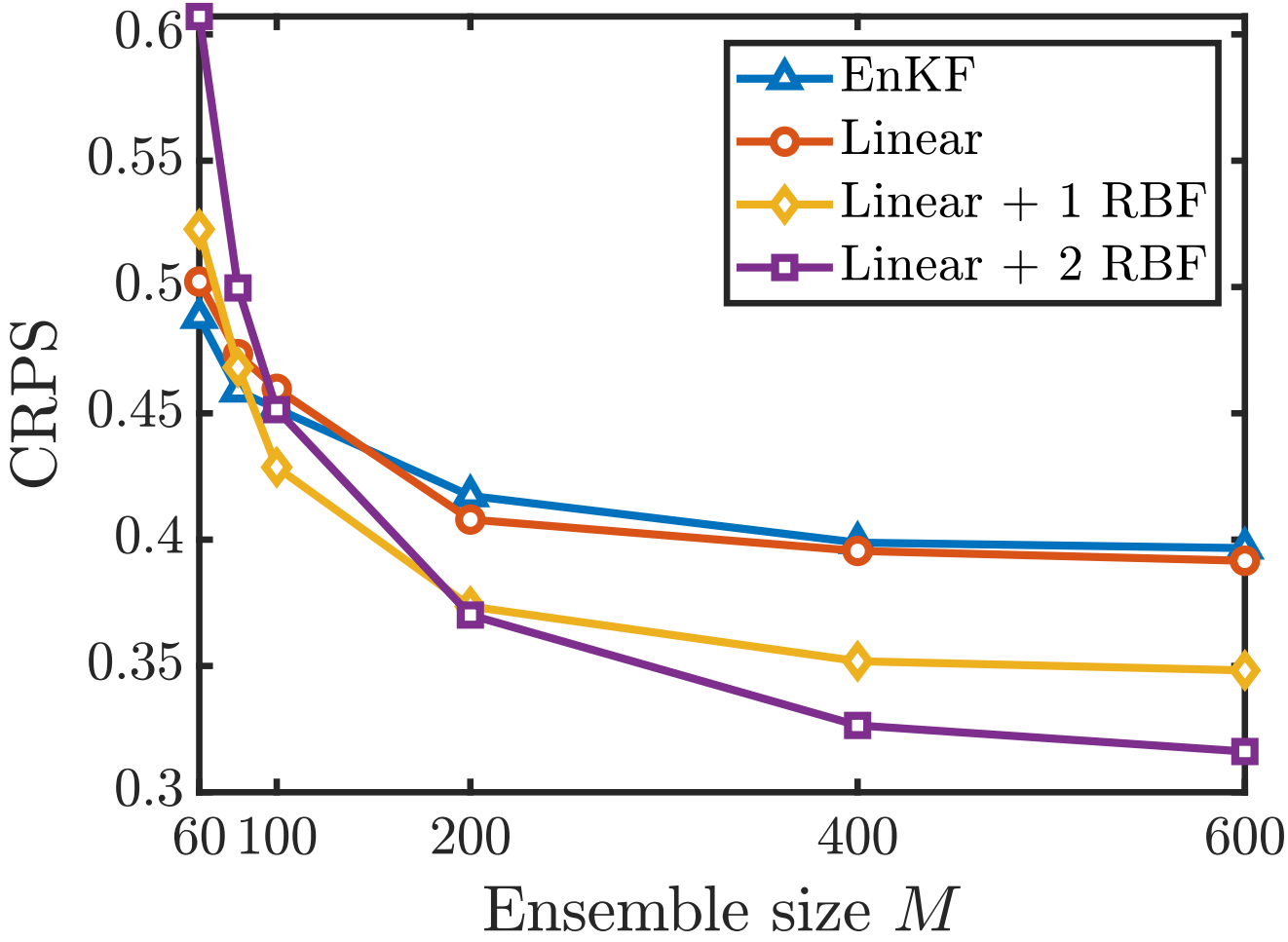
  - $F = 8$ (chaotic) and $\mathcal{E}_j \sim \mathcal{N}(0, 0.5)$ (**small noise for PF**)
  - Time between observations: $\Delta_{\text{obs}} = 0.4$ (**large**)
  - Results computed over 2000 assimilation cycles, following spin-up

Lorenz-96: "hard" case

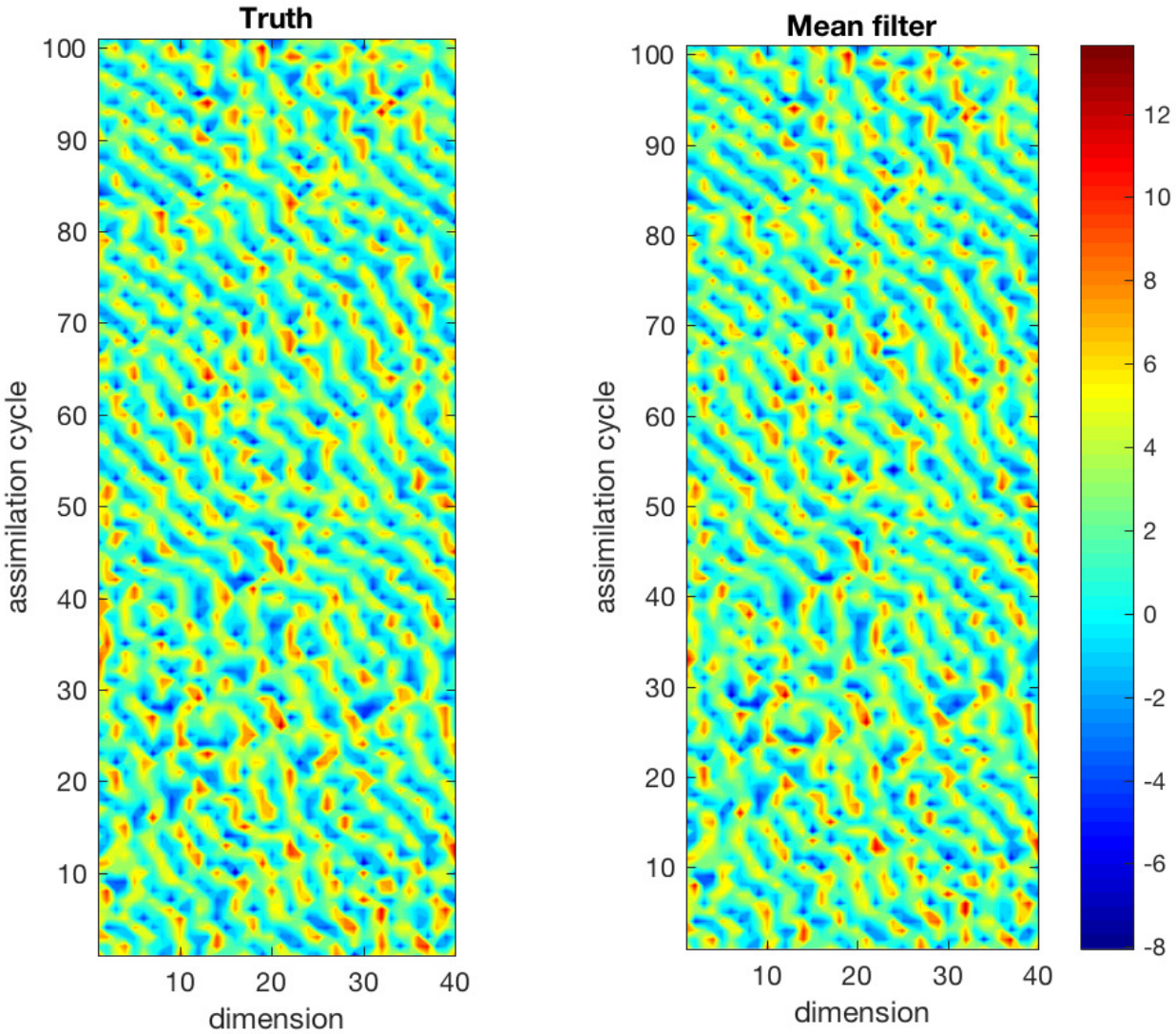


- The nonlinear filter is $\approx 25\%$ more accurate in RMSE than EnKF

Lorenz-96: "hard" case



Lorenz-96: tracking performance of the filter



- Simple and and localized nonlinearities have significant impact

Remarks and issues

- **Nonlinear generalization of the EnKF:** *move* the ensemble members via local nonlinear transport maps, *no weights or degeneracy*
- Learn non-Gaussian features via nonlinear continuous transport and *convex optimization*
- Choice of map basis and **sparsity** provide regularization

Remarks and issues

- **Nonlinear generalization of the EnKF:** *move* the ensemble members via local nonlinear transport maps, *no weights or degeneracy*
- Learn non-Gaussian features via nonlinear continuous transport and *convex optimization*
- Choice of map basis and **sparsity** provide regularization

- In principle, inference is consistent as $\mathcal{S}^h_\triangle$ is enriched and $M \to \infty$. But what is a good choice of $\mathcal{S}^h_\triangle$ for any fixed ensemble size M?
- How to relate map structure/parameterization to the underlying dynamics, observation operators, and data?
- How well can these maps capture **tails** and **extremes?**

Underlying question: how to learn maps?

The "ML way:" many normalizing/autoregressive "flows" are built from special cases of triangular maps, and their compositions:

Underlying question: how to learn maps?

The "ML way:" many normalizing/autoregressive "flows" are built from special cases of triangular maps, and their compositions:

- NICE: Nonlinear independent component estimation [Dinh et al. 2015]
$$S^k(x_1, \dots, x_k) = \mu_k(\mathbf{x}_{i<k}) + x_k$$
- Inverse autoregressive flow [Dinh et al. 2017]
$$S^k(x_1, \dots, x_k) = (1 - \sigma_k(\mathbf{x}_{i<k}))\mu_k(\mathbf{x}_{i<k}) + x_k\sigma_k(\mathbf{x}_{i<k})$$
- Masked autoregressive flow [Papamakarios et al. 2017]
$$S^k(x_1, \dots, x_k) = \mu_k(\mathbf{x}_{i<k}) + x_k \exp(\alpha_k(\mathbf{x}_{i<k}))$$
- Sum-of-squares polynomial flow [Jaini et al. 2019]
$$S^k(x_1, \dots, x_k) = a_k(\mathbf{x}_{i<k}) + \int_0^{x_k} \sum_{\kappa=1}^{p} (\text{poly}(t; \mathbf{a}_{\kappa,k}(\mathbf{x}_{i<k}))^2 dt$$

Underlying question: how to learn maps?

The "ML way:" many normalizing/autoregressive "flows" are built from special cases of triangular maps, and their compositions:

- NICE: Nonlinear independent component estimation [Dinh et al. 2015]
$$S^k(x_1, \dots, x_k) = \mu_k(\mathbf{x}_{i<k}) + x_k$$
- Inverse autoregressive flow [Dinh et al. 2017]
$$S^k(x_1, \dots, x_k) = (1 - \sigma_k(\mathbf{x}_{i<k}))\mu_k(\mathbf{x}_{i<k}) + x_k\sigma_k(\mathbf{x}_{i<k})$$
- Masked autoregressive flow [Papamakarios et al. 2017]
$$S^k(x_1, \dots, x_k) = \mu_k(\mathbf{x}_{i<k}) + x_k \exp(\alpha_k(\mathbf{x}_{i<k}))$$
- Sum-of-squares polynomial flow [Jaini et al. 2019]
$$S^k(x_1, \dots, x_k) = a_k(\mathbf{x}_{i<k}) + \int_0^{x_k} \sum_{\kappa=1}^{p} (\text{poly}(t; \mathbf{a}_{\kappa,k}(\mathbf{x}_{i<k}))^2 dt$$
- Many **ad hoc choices** and **challenging optimization problems . . .**

Parameterizing monotone maps

Structure: Satisfy monotonicity constraint $\partial_k S^k(\mathbf{x}_{1:k}) > 0 \; \forall \mathbf{x}_{1:k}$

Existing methods

- Enforce at finite training samples $\partial_k S^k(\mathbf{x}^i_{1:k}) > 0$ for $i = 1, \ldots, n$
- Enforce by construction: e.g., SOS polynomial flows

$$S^k(\mathbf{x}_{1:k}) = a_k(\mathbf{x}_{<k}) + \int_0^{x_k} b_k(\mathbf{x}_{<k}, t)^2 dt$$

Improved idea: Represent S^k via an **invertible** "rectifier"

$$S^k(\mathbf{x}_{1:k}) = \mathcal{R}_k(f)(\mathbf{x}_{1:k}) := f(\mathbf{x}_{<k}, 0) + \int_0^{x_k} g(\partial_k f(\mathbf{x}_{<k}, t)) dt,$$

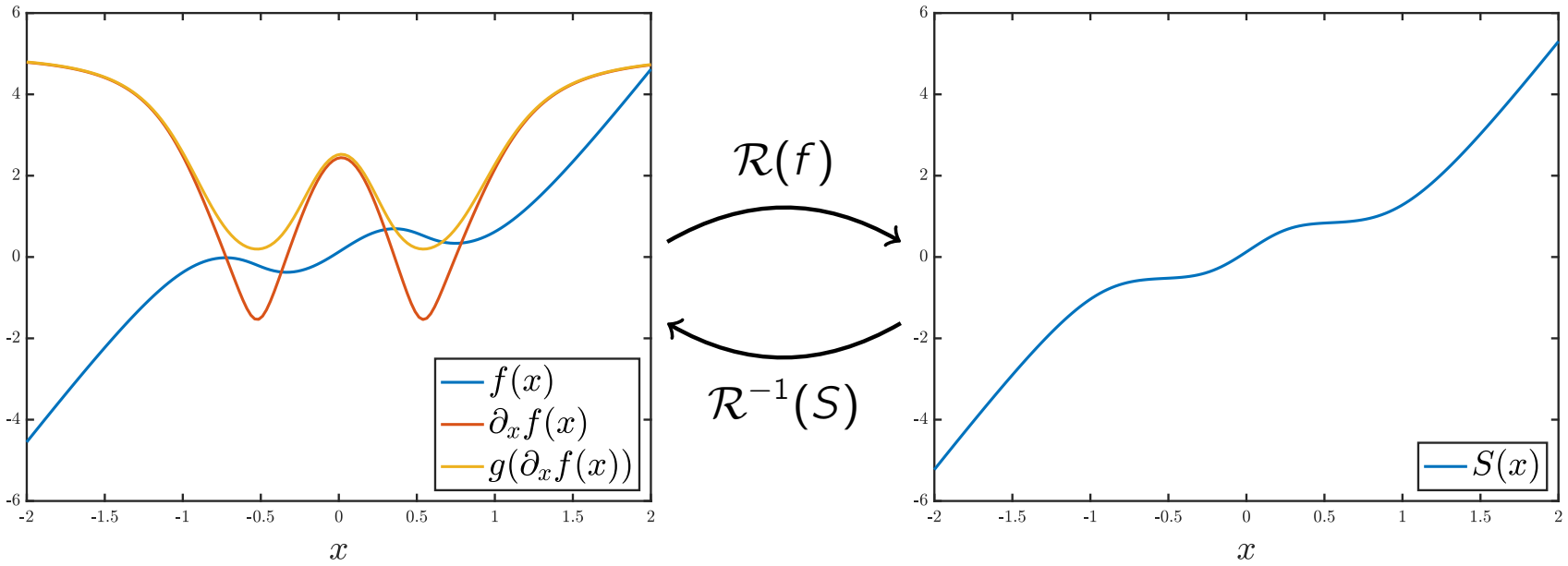
where $g : \mathbb{R} \to \mathbb{R}_{>0}$ is bijective & smooth and $f : \mathbb{R}^k \to \mathbb{R}$ is unconstrained

Parameterizing monotone maps

Rectification of f (1-D example)

For smooth f and bijective $g\colon \mathbb{R} \to \mathbb{R}_{>0}$ (e.g., $g(x) = \log(1 + e^x)$)

$$S(x) = \mathcal{R}(f)(x) := f(0) + \int_0^x g(\partial_x f(t))dt,$$



Approximating monotone maps

Convert the constrained minimization to an unconstrained problem:

$$\min_{\{S:\partial_k S>0\}} \underbrace{\mathbb{E}_\pi\left[\frac{1}{2}S(\mathbf{x}_{1:k})^2 - \log|\partial_k S(\mathbf{x}_{1:k})|\right]}_{\mathcal{J}_k(S)} \quad \Leftrightarrow \quad \min_f \underbrace{\mathcal{J}_k \circ \mathcal{R}_k(f)}_{\mathcal{L}_k(f)}$$

Drawback: With this reparameterization, we lose convexity

Question: When will the objective still have "nice" properties?

Consider the space of functions

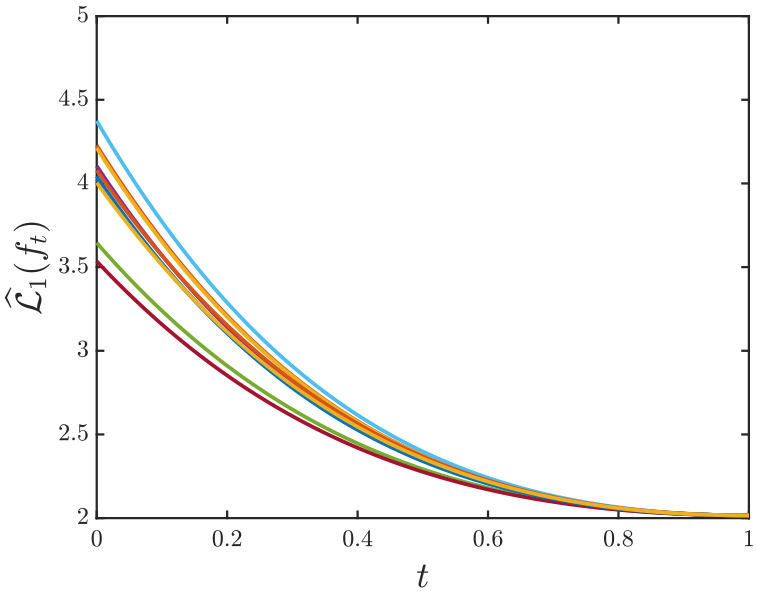
$$H^{1,k}(\mathbb{R}^k) := \left\{f : \mathbb{R}^k \to \mathbb{R} \text{ such that } \int |f(\mathbf{x})|^2 + |\partial_k f(\mathbf{x})|^2 d\mathbf{x} < \infty\right\}$$

Theorem [BZM]

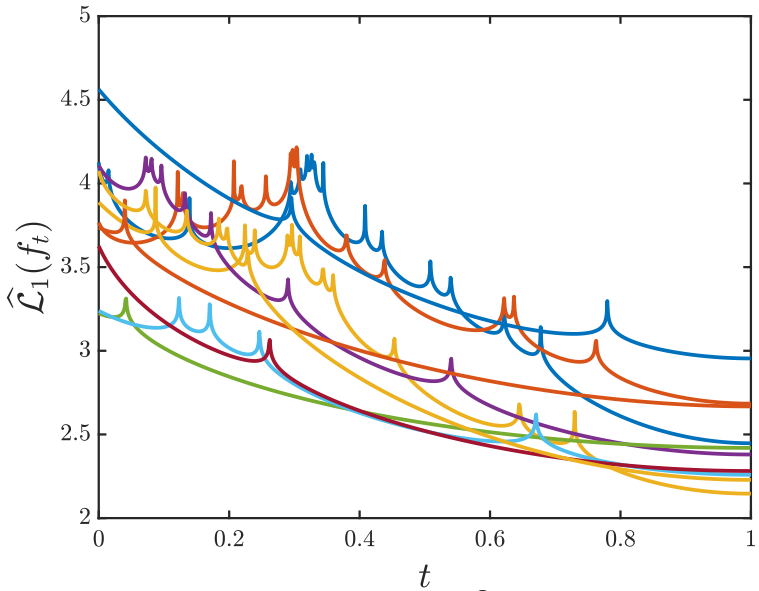
Let $\pi(\mathbf{x}) \leq C_\pi \eta(\mathbf{x})$ for some $C_\pi < \infty$ and η standard Gaussian. Then, for $g(x) = \log(1 + \exp(x))$, $\mathcal{L}_k : H^{1,k}(\mathbb{R}^k) \to \mathbb{R}$ is continuous, bounded, and has a *unique global minimizer*.

Approximating monotone maps

- Mixture of Gaussians target density $\pi(x)$
- Approximate objective as $\widehat{\mathcal{L}}_k$ using $n = 50$ samples
- Evaluate $\widehat{\mathcal{L}}_k$ along segments connecting random initial maps ($t = 0$) to critical points of gradient-based optimizer ($t = 1$)



$g(x) = \log(1 + \exp(x))$



$g(x) = x^2$

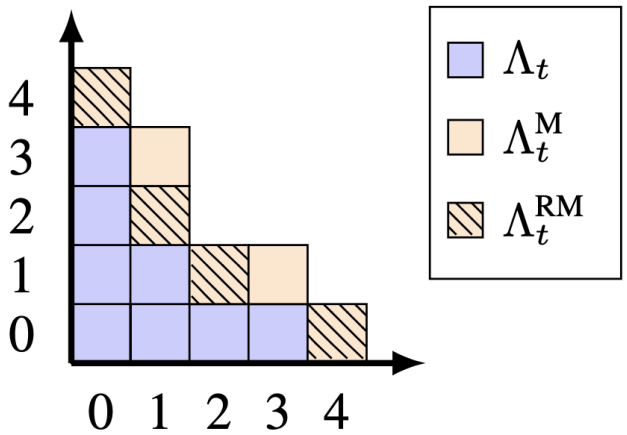
Takeaway: Smooth objective with a single minimizer = reliable training!

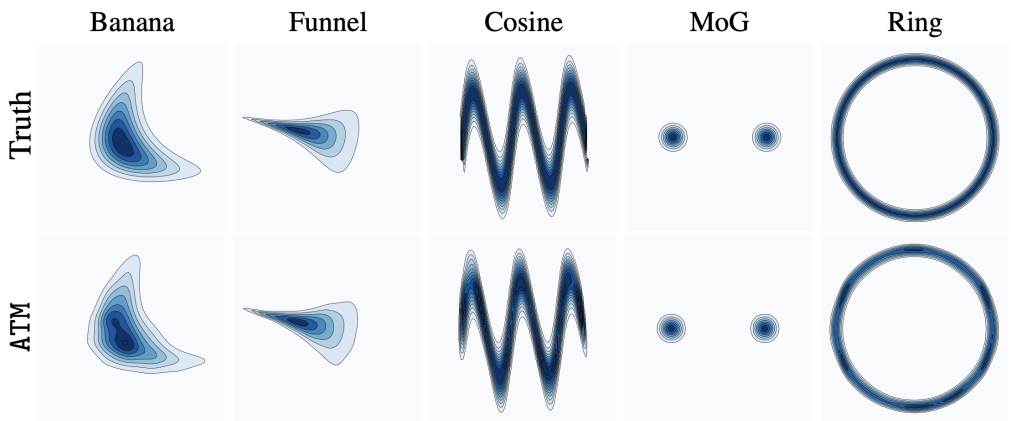
Adaptive transport map (ATM) algorithm

Goal: Approximate $f(\mathbf{x})$ given n i.i.d. samples from π

Greedy enrichment procedure

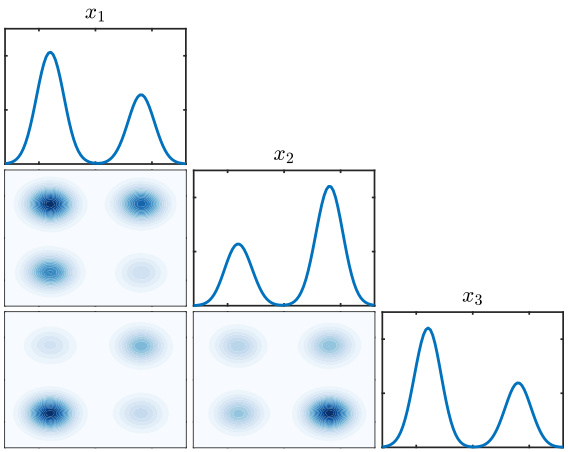
- Look for sparse expansion $f(\mathbf{x}) = \sum_{\boldsymbol{\alpha}\in\Lambda} c_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\mathbf{x})$
- Use tensor-product Hermite functions $\psi_{\boldsymbol{\alpha}}(\mathbf{x}) = P_{\alpha_j}(\mathbf{x})\exp(-\|\mathbf{x}\|^2/2)$
- Add one element to set of active multi-indices Λ_t at a time
- Restrict Λ_t to be downward closed
- Search for new features in the reduced margin of Λ_t



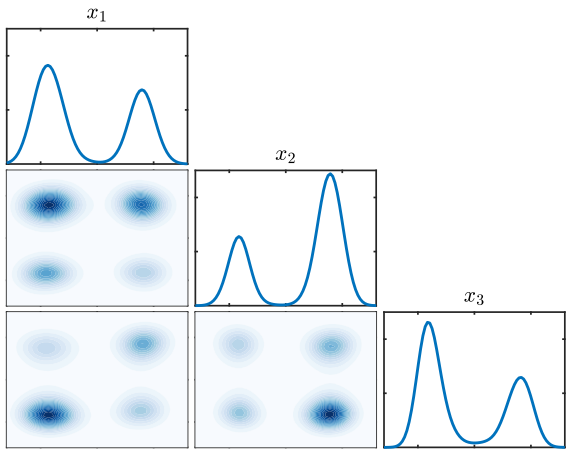


Numerical example: mixture of Gaussians

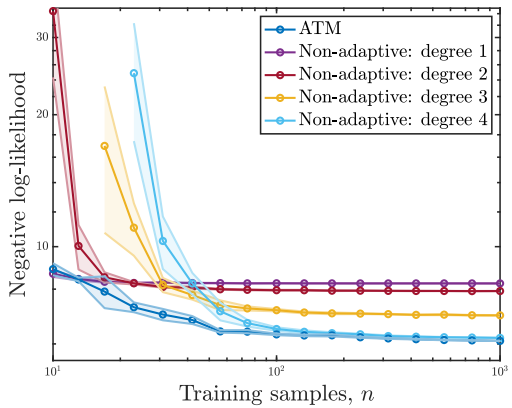
- 3-dimensional mixture of 8 Gaussians with random weights
- Learn map $S = (S^1, S^2, S^3)$ using $n = 100$ training samples
- Compare `ATM` to non-adaptive procedure using total-order expansions



True PDF



Approximate PDF

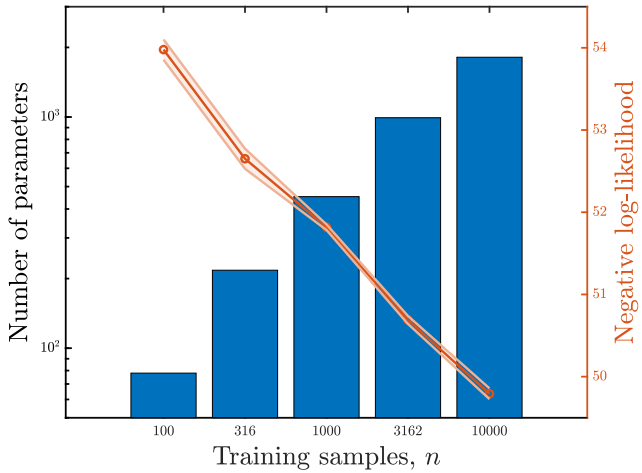


Log-like on test set

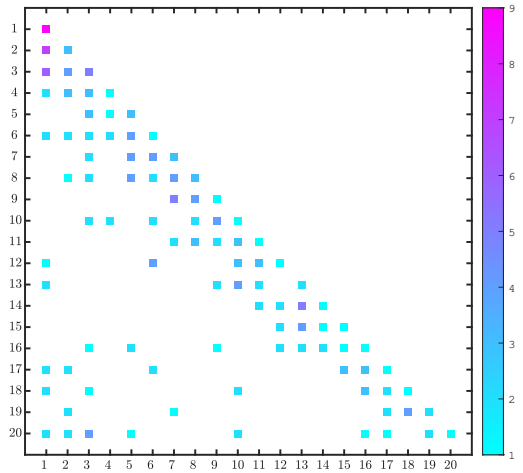
Takeaways: `ATM` finds estimators with number of features m to balance bias and variance for each sample size n

Numerical example: Lorenz-96 data

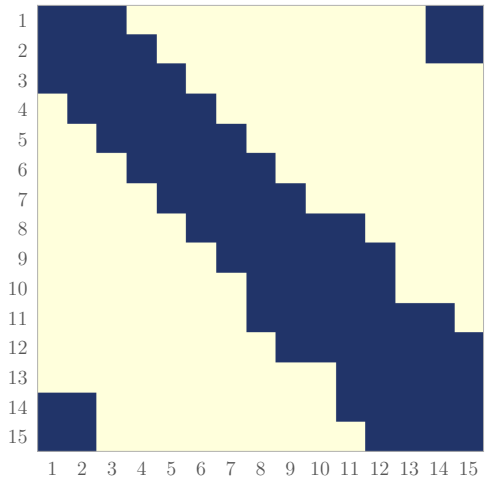
- 20-dimensional distribution for the state of the ODE at a fixed time starting from a Gaussian initial condition



Log-like and m vs n



Sparsity of S: $n = 316$



Conditional independence

Takeaways:

- `ATM` implicitly discovers conditional independence structure in π
- Natural semi-parametric method that gradually increases m with n

Research plans

Broad area #1: high-dimensional conditional generative modeling and likelihood-free inference using *transport maps*

Research plans

Broad area #1: high-dimensional conditional generative modeling and likelihood-free inference using *transport maps*

Will enable **data-driven** inference and prediction in chemical and material systems, using both experimental data and simulation models.

Research plans

Broad area #1: high-dimensional conditional generative modeling and likelihood-free inference using *transport maps*

Research thrusts:

- Developing suitable map parameterizations and characterizing their expressiveness. *Adaptive* representations/algorithms.
- Understanding implications of map parameterization on the associated *estimation* (optimization) problems
- Sample complexity results
- Taking advantage of map **structure** implied by particular problems
  - Sparsity, low rank (Brennan et al. 2020), multiscale behavior...
  - How to parameterize maps to capture *tail behavior* and *extreme events*? What loss/objective should be used to identify such maps?
  - What about data that come from certain ODE or PDE systems?

Research plans

Broad area #1: high-dimensional conditional generative modeling and likelihood-free inference using *transport maps*

Research thrusts (cont.):

- *Direct* transformations of Gaussian (or, e.g., elliptical) reference distributions versus *joint-to-conditional* transformations
- Some previous work on tails of triangular maps [Jaini et al. 2020]. Develop links to extreme value theory.
- Also, learning block-triangular maps in an adversarial framework [Kovachki et al. 2021]

Collaborative efforts

With Evan Reed and team:

- **Chemical kinetic** network models
  - Consider *joint distribution* of chemical species concentrations or *atomic features*, learned from molecular dynamics simulation + physical constraints
  - Predict unobserved species given limited observations
  - Characterize and extrapolate temperature-dependent evolution: reactions become *rare* as T decreases

With Vahid Tarokh and team:

- Extreme values in **PDE** systems
  - Tails of triangular maps and links to (spatial) extreme value theory
  - Generative stochastic models for PDEs with uncertain coefficients and initial/boundary conditions
  - *Conditional* sampling in these models

Research plans

Broad area #2: model misspecification in generative modeling and inference *(collaboration with Jose Blanchet and team)*

- In the misspecified Bayesian setting, posteriors can concentrate in undesirable ways. Can we devise Bayesian procedures that are *robust* to certain kinds of model misspecification?

Research plans

Broad area #2: model misspecification in generative modeling and inference *(collaboration with Jose Blanchet and team)*

- In the misspecified Bayesian setting, posteriors can concentrate in undesirable ways. Can we devise Bayesian procedures that are *robust* to certain kinds of model misspecification?

- Key results from Jose: links between classical regularized estimators and distributionally robust optimization
  - Apply these results to transport-based density estimation, e.g., with ℓ_1 penalties.
  - How can approximation theoretic analysis of transport maps help characterize misspecification of generative models?
  - What are the implications for likelihood-free Bayesian inference? *Consider transport maps trained from synthetic/simulation data and then applied to real/experimental data.*
  - How can we design nonparametric uncertainty sets appropriate for conditional prediction?

Conclusions

Thanks for your attention!

References

- R. Baptista, Y. Marzouk, R. Morrison, O. Zahm. "Learning non-Gaussian graphical models via Hessian scores and triangular transport." arXiv:2101:03093. (And earlier conference version in *NeurIPS 2017*.)
- M. Brennan, D. Bigoni, O. Zahm, A. Spantini, Y. Marzouk. "Greedy inference with structure-exploiting lazy maps." *NeurIPS 2020*, arXiv:1906.00031.
- R. Baptista, O. Zahm, Y. Marzouk. " An adaptive transport framework for joint and conditional density estimation." arXiv:2009.10303, 2020.
- J. Zech, Y. Marzouk. "Sparse approximation of triangular transports on bounded domains." arXiv:2006.06994, 2020.
- N. Kovachki, R. Baptista, B. Hosseini and Y. Marzouk, "Conditional sampling with monotone GANs," arXiv:2006.06755, 2020.
- A. Spantini, R. Baptista, Y. Marzouk. "Coupling techniques for nonlinear ensemble filtering." arXiv:1907.00389, 2020.
- O. Zahm, T. Cui, K. Law, A. Spantini, Y. Marzouk. "Certified dimension reduction in nonlinear Bayesian inverse problems." arXiv:1807.03712, 2019.
- A. Spantini, D. Bigoni, Y. Marzouk. "Inference via low-dimensional couplings." *JMLR* 19(66): 1–71, 2018.
- M. Parno, Y. Marzouk, "Transport map accelerated Markov chain Monte Carlo." *SIAM JUQ* 6: 645–682, 2018.