Data-driven forecasting for projections of complex systems

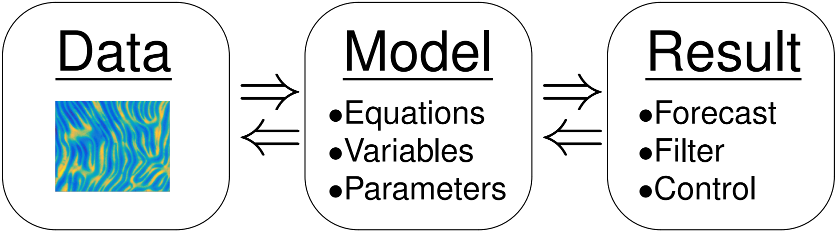
Tyrus Berry

George Mason University

July 16, 2019

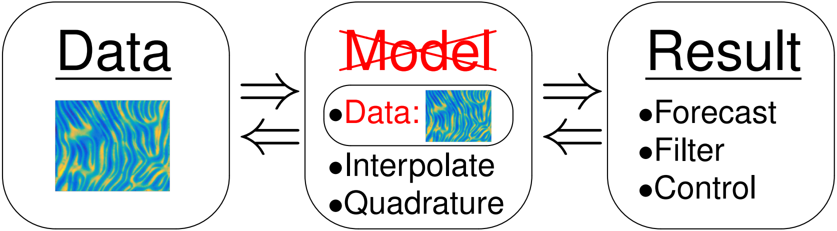
Joint work with John Harlim, PSU and Dimitris Giannakis, NYU
Supported by NSF-DMS

PARAMETRIC MODELING



- **Design Model:** Limited resolution and complexity
- **Assimilate Data:** Fit Parameters/Variables
  - Observability and noise
  - Model error
- **Study/Apply:** Ensemble Forecast

NONPARAMETRIC MODELING



- **Data IS the model:**
  - Assume a model exists
    - Data lies on/near an unknown sub-manifold
    - Data obeys an unknown dynamical system
  - Represent the model using training data

WHAT IS MANIFOLD LEARNING?

- **Manifold learning** $\Leftrightarrow$ **Estimating Laplace Operator**
- Euclidean space:
  - Eigenfunctions of Δ are $e^{i\vec{\omega}\cdot\vec{x}}$
  - Basis for Fourier transform
- Unit circle:
  - Eigenfunctions of Δ are $e^{ik\theta}$
  - Basis for Fourier series
- **Key Fact**: Eigenfunctions of Δ give the smoothest basis for square integrable functions on any manifold.

WHY THE LAPLACIAN?

- **Manifold learning $\Leftrightarrow$ Estimating Laplace-Beltrami**
- Eigenfunctions $\Delta\varphi_i = \lambda_i\varphi_i$ orthonormal basis for $L^2(\mathcal{M})$
- Smoothest functions: φ_i minimizes the functional

$$\lambda_i = \min_{\substack{f \perp \varphi_k \\ k=1,\ldots,i-1}} \left\{ \frac{\int_{\mathcal{M}} ||\nabla f||^2 \, dV}{\int_{\mathcal{M}} |f|^2 \, dV} \right\}$$

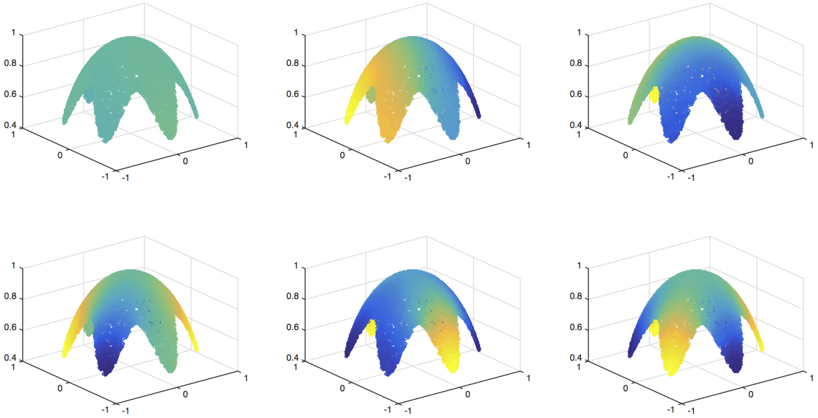
- Eigenfunctions of Δ are custom Fourier basis
  - Smoothest orthonormal basis for $L^2(\mathcal{M})$
  - Can be used to define wavelets
  - Define the Hilbert/Sobolev spaces on $\mathcal{M}$

DIFFUSION MAPS: GRAPH LAPLACIAN → MANIFOLD LAPLACIAN

- For data points $\{x_i\}_{i=1}^N \subset \mathcal{M} \subset \mathbb{R}^n$
- Define $J_{ij} = J(x_i, x_j) = \exp\left(-\frac{||x_i - x_j||^2}{\delta^2}\right)$
- Define $D_i = \sum_j J_{ij}$
- Right normalization: $K_{ij} = D_j^{-1/2} J_{ij}$ and $\hat{D}_i = \sum_j \hat{J}_{ij}$
- Left normalization: $\hat{K}_{ij} = D_i^{-1} K_{ij}$ and finally $L = \frac{I - \hat{K}}{\delta^2}$
- **Theorem:** $L\vec{f} = \Delta_{p_{\mathrm{eq}}} + \mathcal{O}\left(\delta^2, N^{-1/2}\delta^{-1-d/2}\right)$

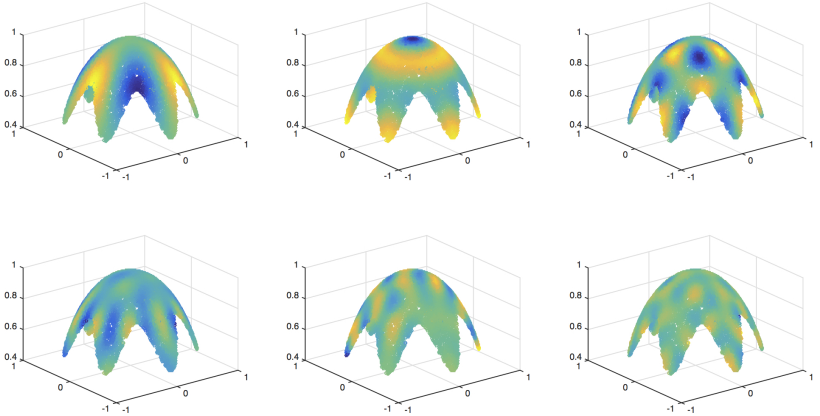
Harmonic Analysis on Manifolds/Data Sets

- Unit circle: $\Delta = \frac{d^2}{d\theta^2}$ eigenfunctions are Fourier basis
- General manifold or data set $\Rightarrow$ Custom Fourier basis



Harmonic Analysis on Manifolds/Data Sets

- Unit circle: $\Delta = \frac{d^2}{d\theta^2}$ eigenfunctions are Fourier basis
- General manifold or data set $\Rightarrow$ Custom Fourier basis



Forecasting with the Fokker-Plank PDE

- Dynamical system: $dx = a(x)\,dt + b(x)\,dW_t$
- Uncertain initial state $x(0)$ with density $p(x,0)$
- Density solves Fokker-Planck PDE, $p_t = \mathcal{L}^* p$ where

$$\mathcal{L}^* f = -\nabla \circ (fa) + \frac{1}{2}\sum_{i,j}\frac{\partial^2}{\partial x_i \partial x_j}\left(f\sum_k b_{ik}b_{jk}\right)$$

- Semigroup solution, $p(x,t) = e^{t\mathcal{L}^*}p(x,0)$

The Shift Map

- Given data samples $x_i = x(t_i)$ with $\tau = t_{i+1} - t_i$
- Define the *shift map* of a function by $Sf(x_i) = f(x_{i+1})$
- Using the Itô lemma we can show:

$$Sf(x_i) = f(x_{i+1}) = e^{\tau\mathcal{L}}f(x_i) + \int_{t_i}^{t_{i+1}} \nabla f^\top b \, dW_s + \int_{t_i}^{t_{i+1}} Bf \, ds$$

- Notice: $\mathbb{E}[S(f)] = e^{\tau\mathcal{L}}f$
- Need to minimize the stochastic integrand $\nabla f^\top b$

Representing the Shift Map

- Choose a basis $\{\varphi_j\}$ orthonormal with respect to $\langle\cdot,\cdot\rangle_{p_{\rm eq}}$
- The coefficients $c_l(t) = \langle p(x,t), \varphi_l\rangle$ have evolution:

$$\begin{aligned} c_l(t+\tau) &= \langle p(x,t+\tau), \varphi_l\rangle \\ &= \left\langle e^{\tau\mathcal{L}^*} p(x,t), \varphi_l \right\rangle = \left\langle p(x,t), e^{\tau\mathcal{L}}\varphi_l\right\rangle \\ &= \sum_j c_j(t) \left\langle \varphi_j, e^{\tau\mathcal{L}}\varphi_l\right\rangle_{p_{\rm eq}} = \sum_j A_{lj} c_j(t) \end{aligned}$$

- So $\vec{c}(t+\tau) = A\vec{c}(t)$
- Where $A_{lj} = \left\langle \varphi_j, e^{\tau\mathcal{L}}\varphi_l\right\rangle_{p_{\rm eq}} \approx \frac{1}{N}\sum_{i=1}^N \varphi_j(x_i)\varphi_l(x_{i+1})$

FORECASTING WITH THE SHIFT MAP

$$
\begin{array}{ccc}
p(x,t) & \xrightarrow{\text{Diffusion Forecast}} & p(x,t+\tau) \\
\Big\downarrow \langle p,\varphi_j \rangle & & \Big\uparrow \sum_j c_j \varphi_j p_{\mathrm{eq}} \\
\vec{c}(t) & \xrightarrow{A_{lj} \equiv \mathbb{E}[\langle \varphi_j, S\varphi_l \rangle_{p_{\mathrm{eq}}}]} & \vec{c}(t+\tau) = A\vec{c}(t).
\end{array}
$$

- We estimate $c_l(t) \approx \frac{1}{N}\sum_{i=1}^{N} \varphi_l(x_i) p(x_i,t)/p_{\mathrm{eq}}(x_i)$
- We estimate A_{lj} with $\hat{A}_{lj} = \frac{1}{N}\sum_{i=1}^{N} \varphi_j(x_i)\varphi_l(x_{i+1})$
- $\mathbb{E}[\hat{A}_{lj}] = A_{lj}$ with error $\mathcal{O}(||\nabla\varphi_l||_{p_{\mathrm{eq}}}\sqrt{\tau/N})$

Choosing a basis

- Need to minimize the error term $\mathcal{O}(||\nabla \varphi_l||_{p_{\mathrm{eq}}} \sqrt{\tau/N})$
- The minimizers of $||\nabla \varphi_l||_{p_{\mathrm{eq}}}$ are a generalized Fourier basis
- Let $\Delta_{p_{\mathrm{eq}}} = \Delta + \frac{\nabla p_{\mathrm{eq}}}{p_{\mathrm{eq}}} \cdot \nabla$ be the Laplacian weighted by p_{eq}
- The eigenfunctions $\Delta_{p_{\mathrm{eq}}} \varphi_j = \lambda_j \varphi_j$ minimize $||\nabla \varphi_j||_{p_{\mathrm{eq}}} = \lambda_j$
- How do we find φ_j? Manifold Learning: Diffusion Maps

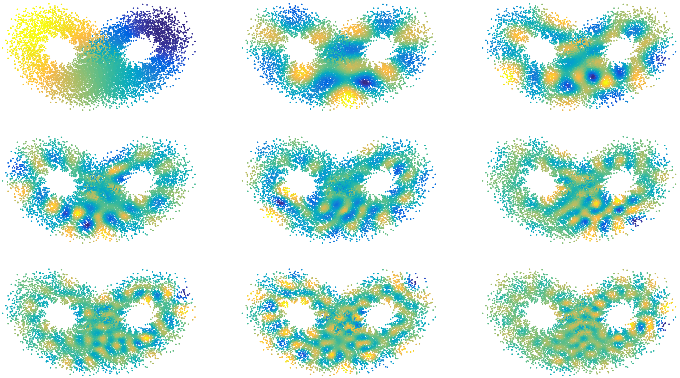
Diffusion Forecast

- Autonomous SDE: $dx = a(x)\,dt + b(x)\,dW_t$
- Density solves Fokker-Planck PDE: $\frac{\partial}{\partial t} p = \mathcal{L}^* p$
- Shift map: $S(p)(x_i) = p(x_{i+1})$ estimates: $\mathbb{E}[S(p)] = e^{\tau \mathcal{L}} p$
- Project onto custom Fourier basis (spectral method)

$$
\begin{array}{ccc}
p(x,t) & \xrightarrow{\text{Diffusion Forecast}} & p(x,t+\tau) = e^{\tau\mathcal{L}^*} p(x,t) \\
\Big\downarrow \langle p, \varphi_j \rangle & & \Big\uparrow \sum_j c_j \varphi_j q \\
\vec{c}(t) & \xrightarrow{A_{lj} \equiv \mathbb{E}[\langle \varphi_j, S\varphi_l \rangle_q]} & \vec{c}(t+\tau) = A\vec{c}(t).
\end{array}
$$

MANIFOLD LEARNING ⇒ CUSTOM 'FOURIER' BASIS

- **Optimal basis:** Minimum variance $A_{lj} \equiv \mathbb{E}[\langle \varphi_j, S\varphi_l \rangle_q]$



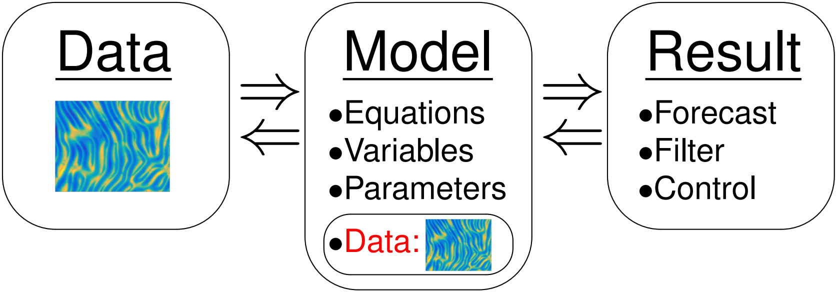
Diffusion Forecast Example

(Loading Video...)

PROBLEM: CURSE OF DIMENSIONALITY

- Nonparametric methods $\rightarrow$ Data required grows like a^{dim}
- Maybe we shouldn't throw out the model...
- Use diffusion forecast to fix model error!

SEMIPARAMETRIC MODELING



- **Data becomes part of the model:**
  - Start with imperfect parametric model
  - Fit training data with time-varying parameters
  - Query data as part of running model
- **Compensate for model error:**
  - Truncated resolution and complexity
  - Non-analytic expressions
  - Non-stationarity/Inhomogeneity

Semiparametric Forecast Model

- Partially known model $\dot{x} = f(x,\theta)$
- Unknown: $d\theta = a(\theta)\,dt + b(\theta)\,dW_t$
- Apply the Diffusion Forecast to $p(\theta, t)$
- Sample $\theta^k(t) \sim p(\theta, t)$ and pair with ensemble $x^k(t)$

$$(x^k(t), \theta^k(t)) \xrightarrow{\dot{x}=f(x,\theta)} (x^k(t+\tau), \theta^k(t+\tau))$$

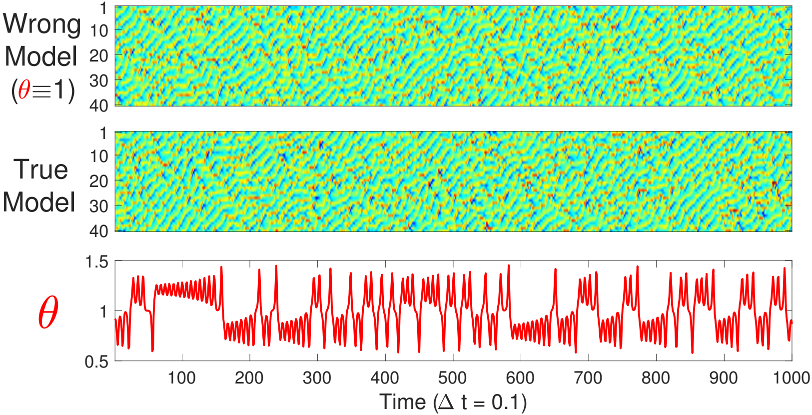
$$\uparrow \theta^k(t) \qquad\qquad \uparrow \theta^k(t+\tau)$$

$$p(\theta, t) \; \dashrightarrow \; p(\theta, t+\tau)$$

Diffusion Forecast

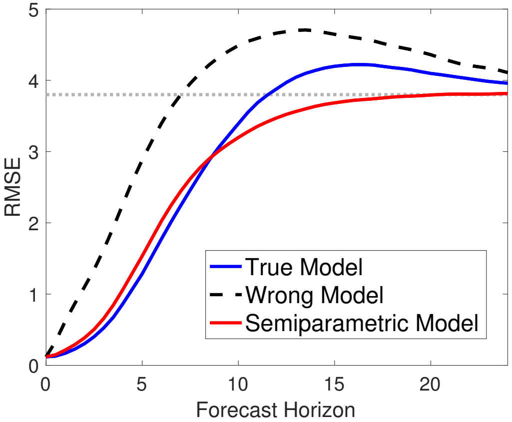
Example: 40-dimensional Lorenz-96 system

$$\dot{x}_i = \theta x_{i-1}x_{i+1} - x_{i-1}x_{i-2} - x_i + 8$$



EXAMPLE: 40-DIMENSIONAL LORENZ-96 SYSTEM

$$\dot{x}_i = \theta x_{i-1}x_{i+1} - x_{i-1}x_{i-2} - x_i + 8$$



PROJECTIONS OF HIGH DIMENSIONAL DYNAMICS

- Consider the 40-dimensional Lorenz-96 system:

$$\dot{x}_i = x_{i-1}x_{i+1} - x_{i-1}x_{i-2} - x_i + 8$$

- Assume we only observe a projection of this system

$$y = h(x_1, ..., x_{40})$$

- Example: Spatial Fourier mode $y = \hat{x}_\omega = \sum_{k=1}^{40} x_i e^{-k\omega}$
- Evolution of y is not closed, sometimes modeled by SDEs

ATTRACTOR RECONSTRUCTION

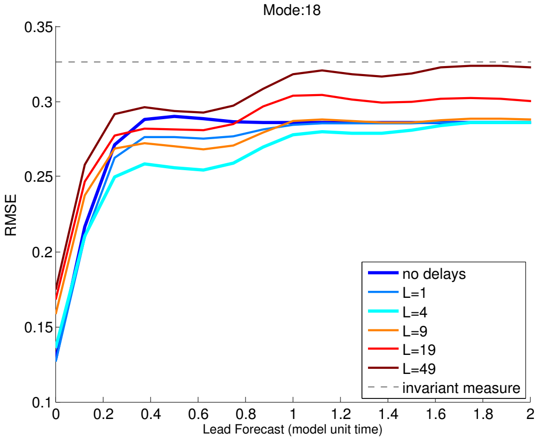
- Evolution of $y = h(x)$ is not closed (missing information)
- Idea: Use delay-embedding to recover the missing info
- Problem 1: Delay embeddings are biased towards stable directions

$$\tilde{y}_t \equiv (y_t, y_{t-\tau}, ..., y_{t-L\tau}) = (h(x_t), h(F_{-\tau}(x_t), ..., h(F_{-L\tau}(x_t))$$

- Problem 2: Curse-of-dimensionality prevents learning the full attractor
- Adding some delays helps, but adding too many hurts

ATTRACTOR RECONSTRUCTION

- Evolution of $y = h(x)$ is not closed
- Adding some delays helps, but adding too many hurts



NEXT STEPS: MORI-ZWANZIG FORMALISM

- Evolution of $y = h(x)$ is not closed
- Delay-embedding, $\tilde{y}_t$ only yeilds partial reconstruction
- Projections of dynamical systems can be closed as

Mori-Zwanzig formalism: $$\frac{d}{dt}\tilde{y} = V + K + R$$

- Diffusion Forecast includes: V (Markovian), R (stochastic)
- Missing the memory term: $K = \int_{-\infty}^{t} K(s, \tilde{y}_t, \tilde{y}_s)\tilde{y}_s \, ds$

For more information: http://math.gmu.edu/~berry/

Building the basis

- Coifman and Lafon, *Diffusion maps.*
- B. and Harlim, *Variable Bandwidth Diffusion Kernels.*
- B. and Sauer, *Local Kernels and Geometric Structure of Data.*

Nonparametric forecast

- B., Giannakis, and Harlim, *Nonparametric forecasting of low-dimensional dynamical systems.*
- B. and Harlim, *Forecasting Turbulent Modes with Nonparametric Diffusion Models.*

Semiparametric forecast

- B. and Harlim, *Semiparametric forecasting and filtering: correcting low-dimensional model error in parametric models.*