Filtering without a model or with a partial model

Tyrus Berry

George Mason University

July 2, 2016

Joint work with John Harlim, PSU and Dimitris Giannakis, NYU

Model Error

- Partially known model $\dot{x} = f(x, \theta)$
- Dynamics $d\theta = a(\theta)\,dt + b(\theta)\,dW_t$ are unknown
- Stochastic parameterizations:

$$
\begin{array}{ccc}
(x^k(t), \theta^k(t)) & \xrightarrow{\quad \dot{x}=f(x,\theta) \quad} & (x^k(t+\tau), \theta^k(t+\tau)) \\
\uparrow \theta^k(t) & & \uparrow \theta^k(t+\tau) \\
p(\theta, t) & \dashrightarrow^{\text{Diffusion Forecast}} & p(\theta, t+\tau)
\end{array}
$$

Goal of the Diffusion Forecast

(Loading Video...)

- **Stochastic Dynamics:** Autonomous, Markov, Ergodic
- **Data:** Generic observable

The Shift Map

- Consider a dynamical system $d\theta = a(\theta)\,dt + b(\theta)\,dW_t$
- Consider an uncertain initial state $\theta(0)$ with density $p(\theta,0)$
- Given data samples $\theta_i = \theta(t_i)$ with $\tau = t_{i+1} - t_i$
- Using the Itô lemma we can show:

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

- Feynman-Kac implies unbiased estimator: $\mathbb{E}[S(f)] = e^{\tau\mathcal{L}}f$
- Project on smooth basis to minimize stochastic integrand $\nabla f^\top b$

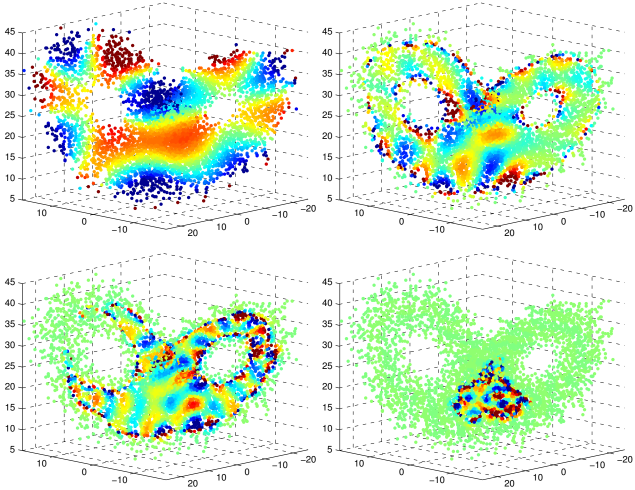
Representing the Shift Map

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

$$c_l(t+\tau) = \langle p(\theta, t+\tau), \varphi_l \rangle = \left\langle e^{\tau \mathcal{L}^*} p(\theta, t), \varphi_l \right\rangle = \left\langle p(\theta, t), e^{\tau \mathcal{L}} \varphi_l \right\rangle$$
$$= \sum_j c_j(t) \left\langle \varphi_j, e^{\tau \mathcal{L}} \varphi_l \right\rangle_{p_{\mathrm{eq}}} = \sum_j A_{lj} c_j(t)$$

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

Diffusion Maps → Custom Fourier Basis



Forecast Operator is Linear in the Diffusion Basis

$$p(\theta,t) \xrightarrow{\text{Diffusion Forecast}} p(\theta,t+\tau)$$

$$p(\theta,t) \xrightarrow{\langle p,\varphi_j\rangle} \vec{c}(t) \qquad\qquad \vec{c}(t+\tau) \xrightarrow{\sum_j c_j\varphi_j p_{\rm eq}} p(\theta,t+\tau)$$

$$\vec{c}(t) \xrightarrow{A_{lj}\equiv\mathbb{E}[\langle\varphi_j,S\varphi_l\rangle_{p_{\rm eq}}]} \vec{c}(t+\tau) = A\vec{c}(t)$$

- Shift Map: $S(\varphi_l)(\theta_i) = \varphi_l(\theta_{i+1})$
- Forecast Operator: $A_{lj} = \overline{\varphi_j(\theta_i)\varphi_l(\theta_{i+1})}$
- Diffusion maps: Eigenfunctions of $\Delta + \frac{\nabla p_{\rm eq}}{p_{\rm eq}} \cdot \nabla$ are optimal basis

Forecasting with the Shift Map

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

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

Filtering with the Shift Map

Introduce an observable $y = h(\theta) + \nu$ with $\nu \sim q$

$$
\begin{array}{ccccc}
p^a(\theta, t-\tau) & \xrightarrow{\text{Diffusion Forecast}} & p^f(\theta, t) & \xrightarrow{p^f(\theta)p(y \,|\, \theta)} & p^a(\theta, t) \\
\Big\downarrow \langle p^a, \varphi_j \rangle & & \Big\uparrow \sum_j c_j^f \varphi_j p_{\mathrm{eq}} & & \langle p^a, \varphi_j \rangle \Big\downarrow \\
\vec{c}^{\,a}(t-\tau) & \xrightarrow{A_{lj} c^a(t-\tau)} & \vec{c}^{\,f}(t) & \xrightarrow{\text{Bayesian Update}} & \vec{c}^{\,a}(t)
\end{array}
$$

- Likelihood is $p(y \,|\, \theta) = q(y - h(\theta))$
- Evaluate on the training data $p^a(\theta_i) = p^f(\theta_i) q(y - h(\theta_i))$
- Continuing work: Learn the conditional density from data

Recovering the Kalman Filter for Linear Systems

(Loading Video...)

- **Linear Dynamics:** $dx = ax\,dt + b\,dW_t$
- **Linear Observation:** $dy = x\,dt + R\,dW_t$

Filtering in a Double Well Potential

(Loading Video...)

- **Double Well Potential:** $dx = x(1 - x^2)\, dt + b\, dW_t$
- **Linear Observation:** $dy = x\, dt + R\, dW_t$

Non-observatibility

(Loading Video...)

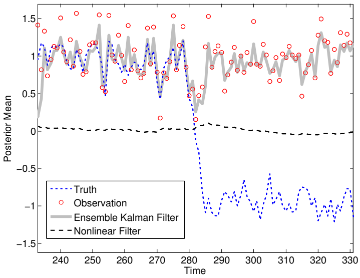
- **Double Well Potential:** $dx = x(1 - x^2)\,dt + b\,dW_t$
- **Absolute Value Observation:** $dy = |x|\,dt + R\,dW_t$

Restricted Observatibility

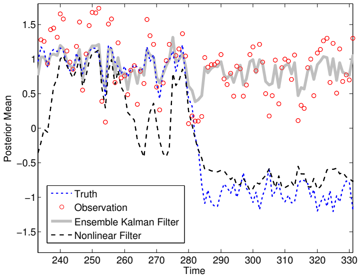
(Loading Video...)

- **Double Well Potential:** $dx = x(1-x^2)\,dt + b\,dW_t$
- **Tough Observation:** $dy = (x-0.05)^2\,dt + R\,dW_t$

Double Well with Observatibility Restrictions



$$dy = |x|\,dt + R\,dW_t$$



$$dy = (x - 0.05)^2\,dt + R\,dW_t$$

Model Error and the Curse of Dimensionality

- Nonparametric model *interpolates* from the training data
- Data required grows exponentially in the dimension of the manifold
- For high-dimensional systems we usually have an approximate model
- High-dimensional models are subject to model error
- Idea: Use the nonparametric model for the model error

Semiparametric Model

- Partially known model $\dot{x} = f(x, \theta)$
- Dynamics $d\theta = a(\theta)\,dt + b(\theta)\,dW_t$ are unknown
- Build a Diffusion Forecast model for $p(\theta, t)$
- Sample $\theta^k(t) \sim p(\theta, t)$ to use with ensemble forecast (x^k, θ^k)

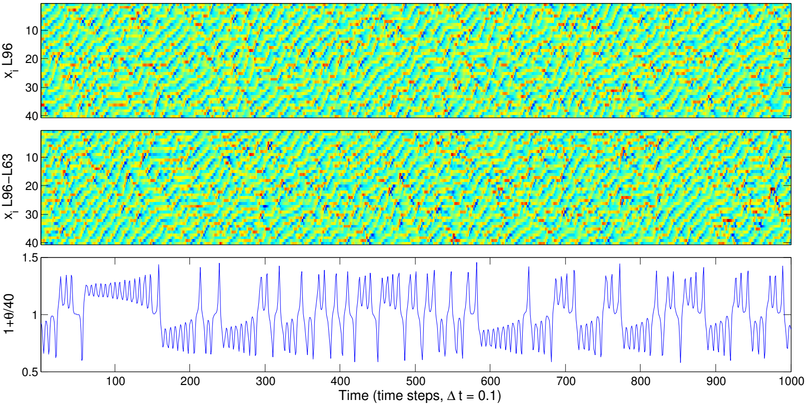
$$\begin{array}{ccc}
(x^k(t), \theta^k(t)) & \xrightarrow{\dot{x}=f(x,\theta)} & (x^k(t+\tau), \theta^k(t+\tau)) \\
\Big\uparrow \theta^k(t) & & \Big\uparrow \theta^k(t+\tau) \\
p(\theta, t) & \overset{\text{Diffusion Forecast}}{- - - - - - \longrightarrow} & p(\theta, t+\tau)
\end{array}$$

Semiparametric Filter: It's a bit complicated...

$$
\begin{array}{ccccc}
\begin{pmatrix} x^{k,a}(t-\tau) \\ \theta^{k,a}(t-\tau) \end{pmatrix} & \xrightarrow{\dot{x}=f(x,\theta)} & \begin{pmatrix} x^{k,f}(t) \\ \theta^{k,f}(t) \end{pmatrix} & \xrightarrow{\text{EnKF } y^{o}(t)} & \begin{pmatrix} x^{k,a}(t) \\ \theta^{k,a}(t) \end{pmatrix} \\
\Big\downarrow \theta^{a} & & \Big\uparrow \theta^{k,f}(t) & & p(\theta^{a}(t) \,|\, \theta(t)) \Big\downarrow \\
p^{a}(\theta, t-\tau) & \xrightarrow{\text{Diffusion Forecast}} & p^{f}(\theta, t) & \xrightarrow{p^{f}(\theta) p(y \,|\, \theta)} & p^{a}(\theta, t) \\
\Big\downarrow \langle p^{a}, \varphi_j \rangle & & \Big\uparrow \sum_j c_j^{f} \varphi_j p_{\mathrm{eq}} & & \langle p^{a}, \varphi_j \rangle \Big\downarrow \\
\vec{c}^{\,a}(t-\tau) & \xrightarrow{A_{lj} c^{a}(t-\tau)} & \vec{c}^{\,f}(t) & \xrightarrow{\text{Bayesian Update}} & \vec{c}^{\,a}(t)
\end{array}
$$

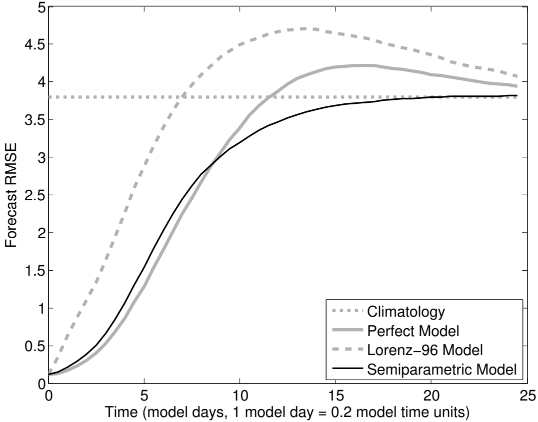
Example: 40-dimensional Lorenz-96 system

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



Example: 40-dimensional Lorenz-96 system

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



Additional challenges of semiparametric modeling

- Need a training data set for θ
- Need initial condition $p(\theta, t)$ for nonparametric forecast
- We developed semiparametric filtering to address these
- Still require that evolution of θ does not depend on x

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

Building the basis

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

Nonparametric forecast

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

Semiparametric forecast

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

Advertisement: Kalman-Takens Filter

- Franz Hamilton (North Carolina State University), et al.
  Ensemble Kalman Filtering without a Model in PRX
- No model, but observation function is known
- Only care about the mean (non-probabilistic)
- Takens embedding + local linear forecast $\Rightarrow$ Short term forecast
- Apply EnKF using the local linear forecast for the ensemble
- High-dimensional systems: assimilate one variable at a time

Advertisement: Kalman-Takens Filter

Filtering Lorenz-96 with no model (one variable at a time):