Improving EnKF with machine learning algorithms

John Harlim

Department of Mathematics and Department of Meteorology
The Pennsylvania State University

June 12, 2017

Overview

A supervised learning algorithm

An unsupervised learning algorithm (diffusion maps)

Learning the localization function of EnKF

Learning a likelihood function. Application: To Correct biased observation model error in DA

A supervised learning algorithm

The basic idea of supervised learning algorithm is to train a map

$$\mathcal{H} : X \to Y,$$

from a pair of data set $\{x_i, y_i\}_{i=1,\dots,N}$.

A supervised learning algorithm

The basic idea of supervised learning algorithm is to train a map

$$\mathcal{H} : X \to Y,$$

from a pair of data set $\{x_i, y_i\}_{i=1,\ldots,N}$.

Remarks:

- The objective is to use the estimated map $\hat{\mathcal{H}}$ to predict $y_s = \hat{\mathcal{H}}(x_s)$ given new data x_s.

A supervised learning algorithm

The basic idea of supervised learning algorithm is to train a map

$$\mathcal{H} : X \to Y,$$

from a pair of data set $\{x_i, y_i\}_{i=1,\dots,N}$.

Remarks:

- The objective is to use the estimated map $\hat{\mathcal{H}}$ to predict $y_s = \hat{\mathcal{H}}(x_s)$ given new data x_s.
- Various methods to estimate $\mathcal{H}$ include regression, SVM, KNN, Neural Nets, etc.

A supervised learning algorithm

The basic idea of supervised learning algorithm is to train a map

$$\mathcal{H} : X \to Y,$$

from a pair of data set $\{x_i, y_i\}_{i=1,\ldots,N}$.

Remarks:

- The objective is to use the estimated map $\hat{\mathcal{H}}$ to predict $y_s = \hat{\mathcal{H}}(x_s)$ given new data x_s.
- Various methods to estimate $\mathcal{H}$ include regression, SVM, KNN, Neural Nets, etc.
- For this talk, we will focus on how to use regression in appropriate spaces to improve EnKF.

An unsupervised learning algorithm

Given a data set $\{x_i\}$, the main task is to learn a function $\varphi(x_i)$ that can describe the data.

[1] Coifman & Lafon 2006, Berry & H, 2016.

An unsupervised learning algorithm

Given a data set $\{x_i\}$, the main task is to learn a function $\varphi(x_i)$ that can describe the data.

In this talk, I will focus on a nonlinear manifold learning algorithm, the **diffusion maps**[1]: Given $\{x_i\} \in \mathcal{M} \subset \mathbb{R}^n$ with a sampling measure q, the diffusion maps algorithm is a kernel based method that produces orthonormal basis functions on the manifold, $\varphi_k \in L^2(\mathcal{M}, q)$.

[1] Coifman & Lafon 2006, Berry & H, 2016.

An unsupervised learning algorithm

Given a data set $\{x_i\}$, the main task is to learn a function $\varphi(x_i)$ that can describe the data.

In this talk, I will focus on a nonlinear manifold learning algorithm, the **diffusion maps**[1]: Given $\{x_i\} \in \mathcal{M} \subset \mathbb{R}^n$ with a sampling measure q, the diffusion maps algorithm is a kernel based method that produces orthonormal basis functions on the manifold, $\varphi_k \in L^2(\mathcal{M}, q)$.

These basis functions are solutions of an eigenvalue problem,

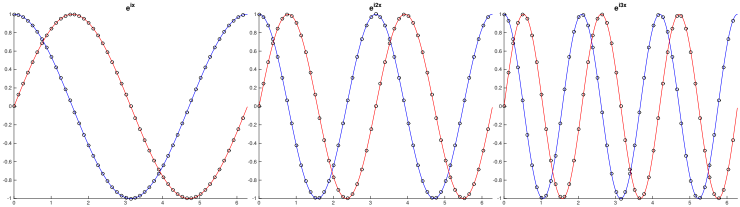
$$q^{-1}\text{div}\Big(q\nabla\varphi_k(x)\Big) = \lambda_k\varphi_k(x),$$

where the weighted Laplacian operator is approximated with an integral operator with appropriate normalization.

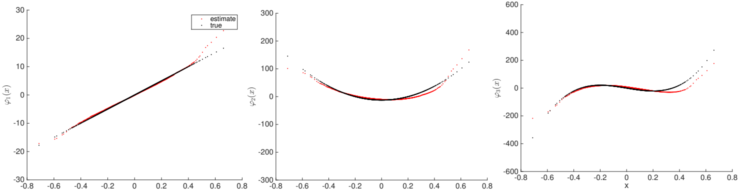
[1] Coifman & Lafon 2006, Berry & H, 2016.

Examples:

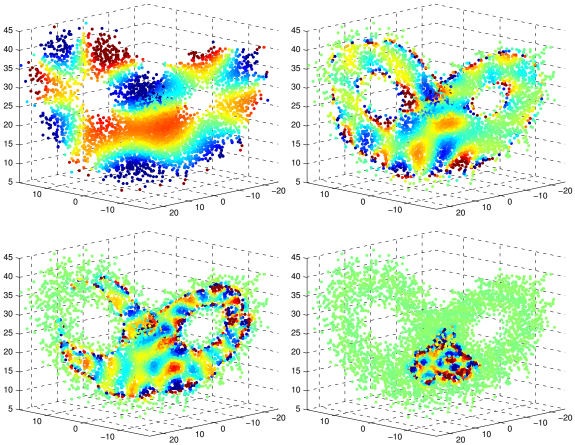
Example: Uniformly distributed data on a circle, we obtain the Fourier basis.



Example: Gaussian distributed data on a real line, we obtain the Hermite polynomials.



Example: Nonparametric basis functions estimated on nontrivial manifold



Remark: Essentially, one can view the DM as a method to learn generalized Fourier basis on the manifold.

Learning the localization function of EnKF

- When EnKF is performed with small ensemble size, one way to alleviate the spurious correlation is to employ a localization function.

Learning the localization function of EnKF

- When EnKF is performed with small ensemble size, one way to alleviate the spurious correlation is to employ a localization function.
- For example, in the serial EnKF, for each scalar observation, y_i, one "localizes" the Kalman gain,

$$K = L_{xy_i} \circ XY_i^\top (Y_i Y_i^\top + R)^{-1},$$

with an empirically chosen localization function L_{xy_i} (Gaspari-Cohn, etc), which requires some tunings.

Learning the localization function of EnKF

- When EnKF is performed with small ensemble size, one way to alleviate the spurious correlation is to employ a localization function.
- For example, in the serial EnKF, for each scalar observation, y_i, one "localizes" the Kalman gain,

$$K = L_{xy_i} \circ XY_i^\top (Y_i Y_i^\top + R)^{-1},$$

  with an empirically chosen localization function L_{xy_i} (Gaspari-Cohn, etc), which requires some tunings.
- Let's use the idea from machine learning to train this localization function. The key idea is to find a map that takes poorly estimated correlations to accurately estimated correlations.

Learning localization map[2]

Given a set of large ensemble EnKF solutions, $\{x_m^{a,k}\}_{\substack{k=1,\ldots,L \\ m=1,\ldots,M}}$ as a training data set, where L is large enough so the correlation, $\rho_{ij}^L \approx \rho(x_i, y_j)$, is accurate.

[2]De La Chevrotière & H, 2017.

Learning localization map[2]

Given a set of large ensemble EnKF solutions, $\{x_m^{a,k}\}_{\substack{k=1,\ldots,L\\ m=1,\ldots,M}}$ as a training data set, where L is large enough so the correlation, $\rho_{ij}^L \approx \rho(x_i, y_j)$, is accurate.

- Operationally, we wish to run EnKF with $K \ll L$ ensemble members. Then our goal is to train a map that transform the subsampled correlation ρ_{ij}^K into the accurate correlation ρ_{ij}^L.

[2] De La Chevrotière & H, 2017.

Learning localization map[2]

Given a set of large ensemble EnKF solutions, $\{x_m^{a,k}\}_{\substack{k=1,\dots,L\\ m=1,\dots,M}}$ as a training data set, where L is large enough so the correlation, $\rho_{ij}^L \approx \rho(x_i, y_j)$, is accurate.

- Operationally, we wish to run EnKF with $K \ll L$ ensemble members. Then our goal is to train a map that transform the subsampled correlation ρ_{ij}^K into the accurate correlation ρ_{ij}^L.
- Basically, we consider the following optimization problem:

$$\min_{L_{x_i y_j}} \int_{[-1,1]} \int_{[-1,1]} \left(L_{x_i y_j} \rho_{ij}^K - \rho_{ij}^L \right)^2 p(\rho_{ij}^K | \rho_{ij}^L) p(\rho_{ij}^L) \, d\rho_{ij}^K \, d\rho_{ij}^L$$

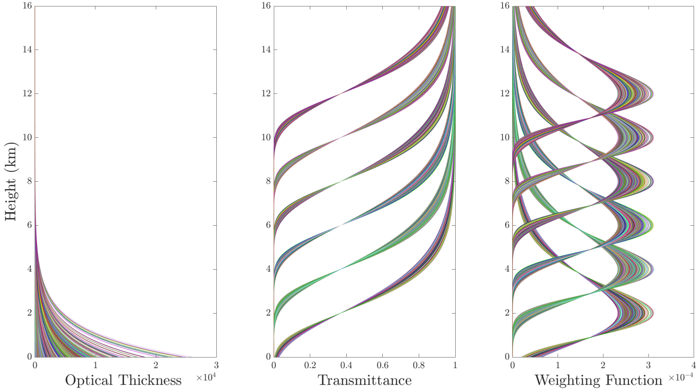
$$\overset{MC}{\approx} \min_{L_{x_i y_j}} \frac{1}{MS} \sum_{m,s=1}^{M,S} (L_{x_i y_j} \rho_{ij,m,s}^K - \rho_{ij,m}^L)^2,$$

where $\rho_{ij,m}^L \sim p(\rho_{ij}^L)$ and $\rho_{ij,m,s}^K \sim p(\rho_{ij}^K | \rho_{ij}^L)$ is an estimated correlation using only K out of L training data.

[2] De La Chevrotière & H, 2017.

Example: On Monsoon-Hadley multicloud model[3]

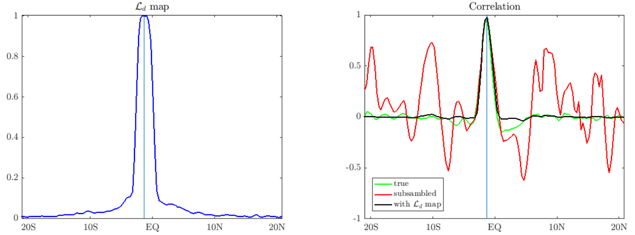
It's a Galerkin projection of zonally symmetric β-plane primitive eqns into the barotropic, and first two baroclinic modes, stochastically driven by a three-cloud model paradigm. Consider observation model $h(x)$ that is similar to a RTM.



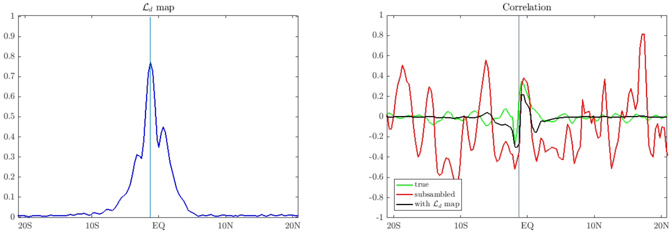
[3] M. De La Chevrotière and B. Khouider 2016.

Example of trained localization map

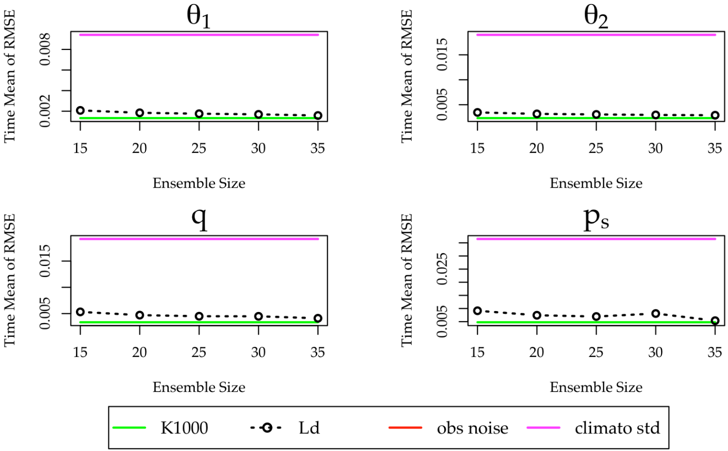
Channel 3 and θ_1



Channel 6 and θ_{eb}



DA results



θ1
θ2
q
ps
Time Mean of RMSE
0.002
0.008
0.005
0.015
0.025
15
20
25
30
35
Ensemble Size
K1000
Ld
obs noise
climato std

Correcting biased observation model error[4]

All the Kalman based DA method assumes unbiased observation model error, e.g.,

$$y_i = h(x_i) + \eta_i, \quad \eta_i \sim \mathcal{N}(0, R).$$

Suppose the operator h is un known. Instead, we are only given $\tilde{h}$, then

$$y_i = \tilde{h}(x_i) + b_i$$

where we introduce a biased model error, $b_i = h(x_i) - \tilde{h}(x_i) + \eta_i$.

[4]Berry & H, 2017.

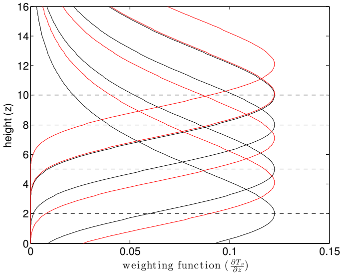
Example: Basic radiative transfer model

Consider solutions of the stochastic cloud model[5], $\{T(z), \theta_{eb}, q, f_d, f_s, f_c\}$. Based on this solutions, define a basic radiative transfer model as follows,

$$h_\nu(x) = \theta_{eb} T_\nu(0) + \int_0^\infty T(z) \frac{\partial T_\nu}{\partial z}(z)\, dz,$$

where T_ν is the transmission between heights z to ∞ that is defined to depend on q.

The weighting function, $\frac{\partial T_\nu}{\partial z}$ are defined as follows:



[5]Khouider, Biello, Majda 2010

Example: Basic radiative transfer model

Suppose the deep and stratiform cloud top height is $z_d = 12$km, while the cumulus cloud top height is $z_c = 3$km. Define $f = \{f_d, f_c, f_s\}$ and $x = \{T(z), \theta_{eb}, q\}$. Then the cloudy RTM is given by,

$$
\begin{aligned}
h_\nu(x, f) \quad = \quad & (1 - f_d - f_s)\left[\theta_{eb} T_\nu(0) + \int_0^{z_d} T(z) \frac{\partial T_\nu}{\partial z}(z)\, dz\right] \\
& + (f_d + f_s) T(z_t) T_\nu(z_d) + \int_{z_d}^{\infty} T(z) \frac{\partial T_\nu}{\partial z}(z)\, dz
\end{aligned}
$$

Example: Basic radiative transfer model

Suppose the deep and stratiform cloud top height is $z_d = 12\text{km}$, while the cumulus cloud top height is $z_c = 3\text{km}$. Define $f = \{f_d, f_c, f_s\}$ and $x = \{T(z), \theta_{eb}, q\}$. Then the cloudy RTM is given by,

$$\begin{aligned}
h_\nu(x,f) &= (1-f_d-f_s)\Big[\theta_{eb}T_\nu(0) + \int_0^{z_d} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz\Big] \\
&\quad +(f_d+f_s)T(z_t)T_\nu(z_d) + \int_{z_d}^{\infty} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz \\
&= (1-f_d-f_s)\Big[(1-f_c)\Big(\theta_{eb}T_\nu(0) + \int_0^{z_c} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz\Big) \\
&\quad +f_c T(z_c)T_\nu(z_c) + \int_{z_c}^{z_d} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz\Big] \\
&\quad +(f_d+f_s)T(z_d)T_\nu(z_t) + \int_{z_d}^{\infty} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz
\end{aligned}$$

Example: Basic radiative transfer model

Suppose the deep and stratiform cloud top height is $z_d = 12$km, while the cumulus cloud top height is $z_c = 3$km. Define $f = \{f_d, f_c, f_s\}$ and $x = \{T(z), \theta_{eb}, q\}$. Then the cloudy RTM is given by,

$$
\begin{aligned}
h_\nu(x,f) &= (1-f_d-f_s)\left[\theta_{eb}T_\nu(0) + \int_0^{z_d} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz\right] \\
&\quad +(f_d+f_s)T(z_t)T_\nu(z_d) + \int_{z_d}^{\infty} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz \\
&= (1-f_d-f_s)\Big[(1-f_c)\Big(\theta_{eb}T_\nu(0) + \int_0^{z_c} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz\Big) \\
&\quad +f_c T(z_c)T_\nu(z_c) + \int_{z_c}^{z_d} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz\Big] \\
&\quad +(f_d+f_s)T(z_d)T_\nu(z_t) + \int_{z_d}^{\infty} T(z)\frac{\partial T_\nu}{\partial z}(z)\,dz
\end{aligned}
$$

One can check that $h_\nu(x,0)$ corresponds to cloud-free RTM.

Systematic model error in data assimilation

Suppose the observation is generated with

$$y_\nu = h_\nu(x, f) + \eta, \qquad \eta \sim \mathcal{N}(0, R)$$

The difficulty in estimating the cloud fractions, cloud top heights and (in reality we don't know precisely how many clouds under a column) induces model error.

Systematic model error in data assimilation

Suppose the observation is generated with

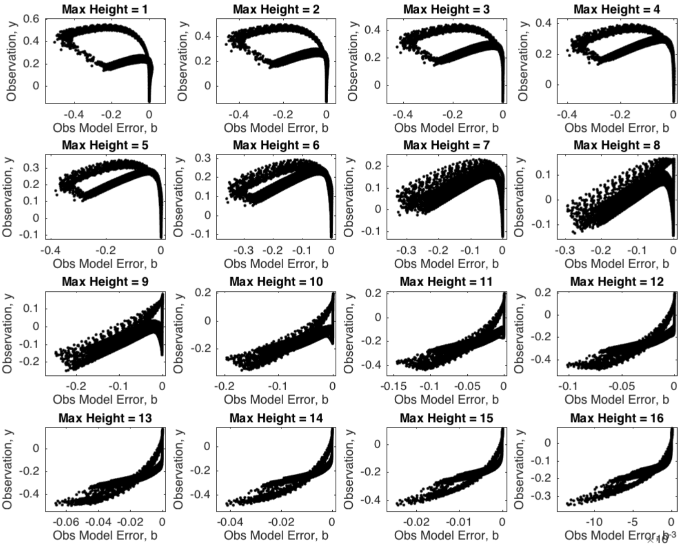
$$y_\nu = h_\nu(x, f) + \eta, \qquad \eta \sim \mathcal{N}(0, R)$$

The difficulty in estimating the cloud fractions, cloud top heights and (in reality we don't know precisely how many clouds under a column) induces model error.
In an extreme case, we consider filtering with a cloud-free RTM:

$$y_\nu = h_\nu(x, 0) + b_\nu$$

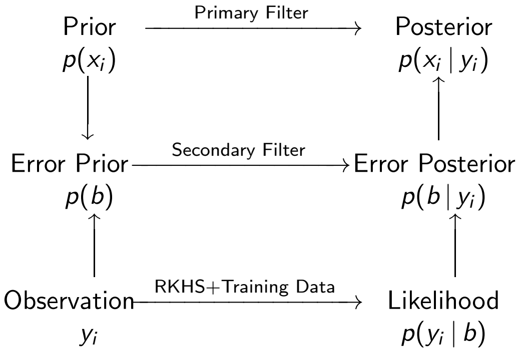
where $b_\nu = h_\nu(x, f) - h_\nu(x, 0) + \eta$ is model error with bias.

Observations (y_ν) v Model error (b_ν)



State estimation of the model error

We propose a secondary filter to estimate the statistics for b_i as follows:

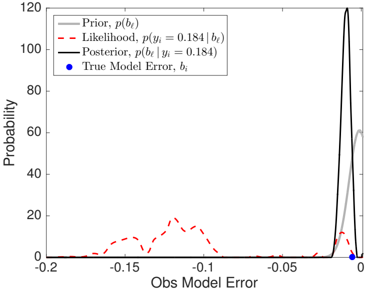


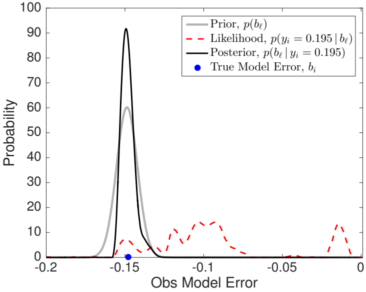
A machine learning technique, kernel embedding of conditional distribution[6], is employed to train a nonparametric likelihood function.

[6] Song, Fukumizu, Gretton, 2013.

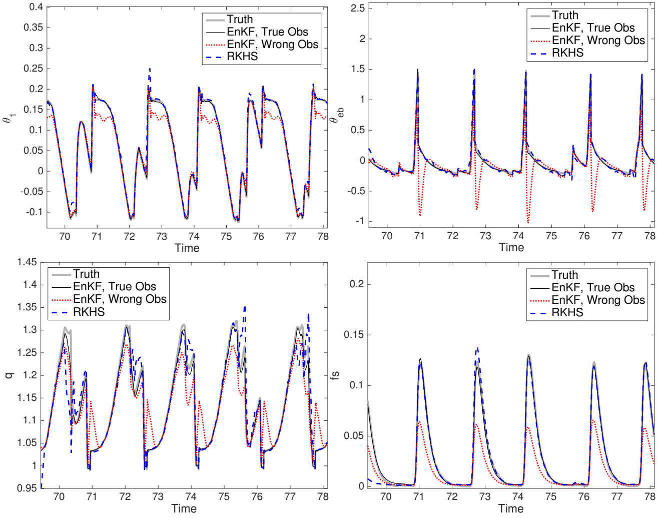
Secondary Bayesian filter

$$p(b|y_i) \propto p(b)p(y_i|b)$$



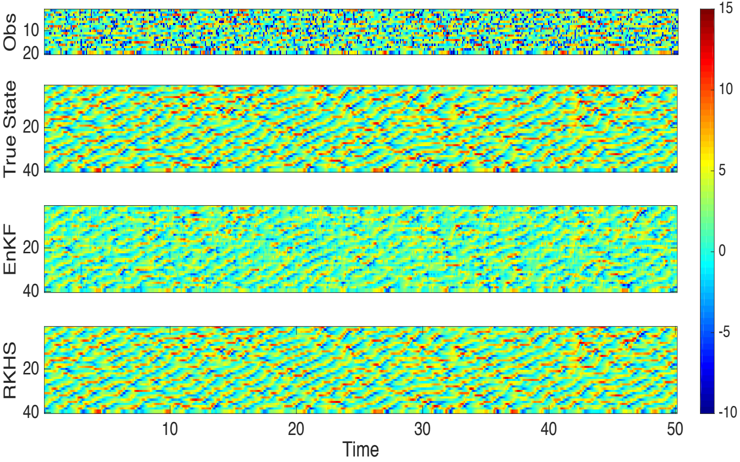


Filter estimates (with adaptive tuning of R and Q).



Example: Lorenz-96

Biased occurs random in space and times.



Nonparametric likelihood function

We will use the kernel embedding of conditional distribution.[7]

Recall: Let X be a r.v on $\mathcal{M}$ and distribution $P(X)$. Given a kernel $K : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$, the Moore-Aronszajn theorem states that there exists a Reproducing Kernel Hilbert Space (RKHS) $L^2(\mathcal{M}, q)$. This means that that $f(x) = \langle f, K(x, \cdot) \rangle_q$.

[7] Song, Fukumizu, Gretton, 2013.

Nonparametric likelihood function

The kernel embedding of conditional distribution $P(Y|B)$ is defined as,

$$\mu_{Y|b} = \mathbb{E}_{Y|b}[\tilde{K}(Y,\cdot)] = \int_{\mathcal{N}} \tilde{K}(y,\cdot)dP(y|b).$$

Given $g \in L^2(\mathcal{N}, \tilde{q})$,

$$\begin{aligned}\mathbb{E}_{Y|b}[g(Y)] &= \int_{\mathcal{N}} g(y)dP(y|b) = \int_{\mathcal{N}} \langle g, \tilde{K}(y,\cdot)\rangle_{\tilde{q}} dP(y|b) \\ &= \left\langle g, \int_{\mathcal{N}} \tilde{K}(y,\cdot)dP(y|b)\right\rangle_{\tilde{q}} = \langle g, \mu_{Y|b}\rangle_{\tilde{q}}.\end{aligned}$$

One can verify that

$$\mu_{Y|b} = q\mathcal{C}_{YB}\mathcal{C}_{BB}^{-1}K(b,\cdot),$$

where

$$\mathcal{C}_{BY} = \int_{\mathcal{M}\times\mathcal{N}} K(b,\cdot)\otimes\tilde{K}(y,\cdot)\, dP(b,y)$$

is the kernel embedding of $P(B,Y)$ on appropriate Hilbert spaces.

Nonparametric likelihood function $p(y|b)$

Given $\{b_i\}_{i=1}^N$ and $\{y_i\}_{i=1}^N$ Apply diffusion maps to learn the data-driven orthonormal basis functions $\varphi_j(b) \in L^2(\mathcal{M}, q)$ and $\tilde{\varphi}_k(y) \in L^2(\mathcal{M}, \tilde{q})$. Let

$$p(y|b) = \sum_k \mu_{Y|b,k} \tilde{\varphi}_k(y) \tilde{q}(y)$$

where

$$\begin{aligned}
\mu_{Y|b,k} &= \langle p(\cdot|b), \tilde{\varphi}_k \rangle = \mathbb{E}_{Y|b}[\tilde{\varphi}_k] = \langle \mu_{Y|b}, \tilde{\varphi}_k \rangle_{\tilde{q}} \\
&= \langle q \mathcal{C}_{YB} \mathcal{C}_{BB}^{-1} K(b, \cdot), \tilde{\varphi}_k \rangle_{\tilde{q}} \\
&= \ldots \\
&= \sum_j \varphi_j(x) [C_{YB} C_{BB}^{-1}]_{kj}
\end{aligned}$$

where

$$[C_{YB}]_{jk} = \langle \mathcal{C}_{YB}, \tilde{\varphi}_j \otimes \varphi_k \rangle_{\tilde{q} \otimes q} \approx \frac{1}{N} \sum_{i=1}^N \tilde{\varphi}_j(y_i) \varphi_k(b_i),$$

$$[C_{BB}]_{jk} = \langle \mathcal{C}_{BB}, \varphi_j \otimes \varphi_k \rangle_q \approx \frac{1}{N} \sum_{i=1}^N \varphi_j(b_i) \varphi_k(b_i),$$

References:

1. M. De La Chevrotière & H, "A data-driven method for improving the correlation estimation in serial ensemble Kalman filters.", Mon. Wea. Rev. 145(3), 985-1001, 2017.
2. M. De La Chevrotière & H, "Localization mappings for filtering a monsoon-Hadley multicloud model" (in progress).
3. T. Berry & H, "Correcting biased observation model error in data assimilation", Mon. Wea. Rev. (in press).
4. T. Berry & H, "Variable bandwidth diffusion kernels", Appl. Comput. Harmon. Anal. 40, 68-96, 2016.
5. H, "An introduction to data-driven methods for stochastic modeling of dynamical systems", Springer (to appear).