Data Assimilation: theory and practice

Amit Apte

Department of Data Science
Indian Institute of Science Education and Research (IISER) Pune, India

Data Science: Probabilistic and Optimization Methods II
ICTS-TIFR, Bengaluru, 08 August 2025

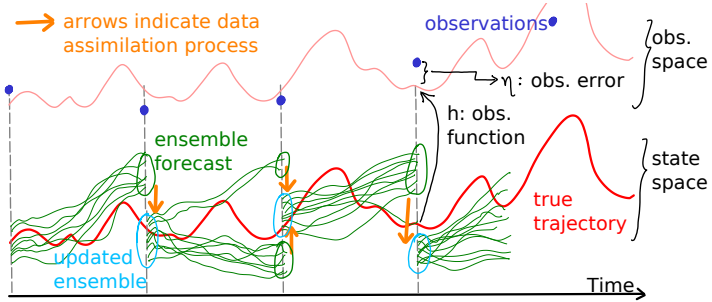
Outline

jupyter notebooks:
Gaussian conditional distributions: `https://tinyurl.com/37c46z6f`
Chaos: `https://tinyurl.com/47djm3x7`
Kalman filter: `https://tinyurl.com/2aj7wz2y`
Ensemble Kalman filter: `https://tinyurl.com/yeykzvbp`

What is data assimilation?

The art of optimally incorporating

- partial and noisy observational data of a
- chaotic, nonlinear, complex dynamical system with an
- imperfect model (of the data and the system) to get an
- estimate and the associated uncertainty for the system state



- Main application: weather and climate predictions
- Also being used biology, industrial applications, etc.
- Main challenges: the complexity of the system; the uncertainties in modelling processes such as the clouds or aerosols; computational challenges

Main ingredients - dynamics and observation

and imperfect models for both! (Hidden Markov Model for state estimation)

- A dynamical model: given the state $x_t \in \mathbb{R}^d$ at any time t, obtain the state x_s at any later time $s > t$ (could be probabilistic model, a Markov process):

  $$x_t = m(x_{t-1}) \qquad \text{that is always imperfect}^1$$

  E.g. x_t = velocity, temperature of the atmosphere (on a grid in real or Fourier space, for a PDE solver)

- Observations: $y_t \in \mathbb{R}^p$ over a certain time period, $t = 1, \ldots, N$

  $$y_t = h(x_t) + \eta_t \qquad \text{that is } \textbf{also} \text{ imperfect}$$

  E.g. y_t = temperature and rainfall measurements at a few locations

- Observation operator $h : \mathbb{R}^d \to \mathbb{R}^p$ to relate the model variables at time t to observations at the same time: if the state were x_t, the observations without noise would be $h(x_t)$

  E.g. How do {velocity, temperature} at a given time relate to {temperature, rainfall} at the same time

- Observational uncertainty: η_t accounts for how the real system is represented in the model (representativeness error) and the instrumental uncertainty

  E.g. lack of knowledge of exact conditions for cloud formation and rain (and rain-guage 'errors')

[1] 'Perfect model' is an oxymoron anyway!

Main ingredients - dynamics and observation

and imperfect models for both! (Hidden Markov Model for state estimation)

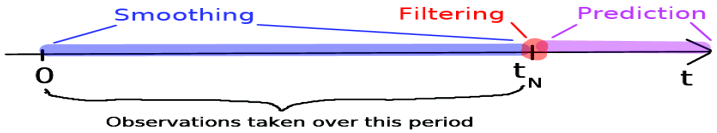
- A dynamical model: given the state $x_t \in \mathbb{R}^d$ at any time t, obtain the state x_s at any later time $s > t$ (could be probabilistic model, a Markov process):

$$x_t = m(x_{t-1}) \qquad \text{Markov with transition density} \qquad p^m(x_t|x_{t-1})$$

- Observations $y_t \in \mathbb{R}^p$ at time t, for $t = 1, \ldots, N$

$$y_t = h(x_t) + \eta_t \qquad \text{with likelihood} \qquad p_\eta(y_t|x_t)$$

- We will consider the problem of "estimating" the state x_t at some time t given observations at times $1, 2, \ldots, N$.



Main ingredients - dynamics and observation

and imperfect models for both! (Hidden Markov Model for state estimation)

- A dynamical model: given the state $x_t \in \mathbb{R}^d$ at any time t, obtain the state x_s at any later time $s > t$ (could be probabilistic model, a Markov process):

$$x_t = m(x_{t-1}) \qquad \text{Markov with transition kernel} \qquad p^m(x_t|x_{t-1})$$

- Observations $y_t \in \mathbb{R}^p$ at time t, for $t = 1, \ldots, N$

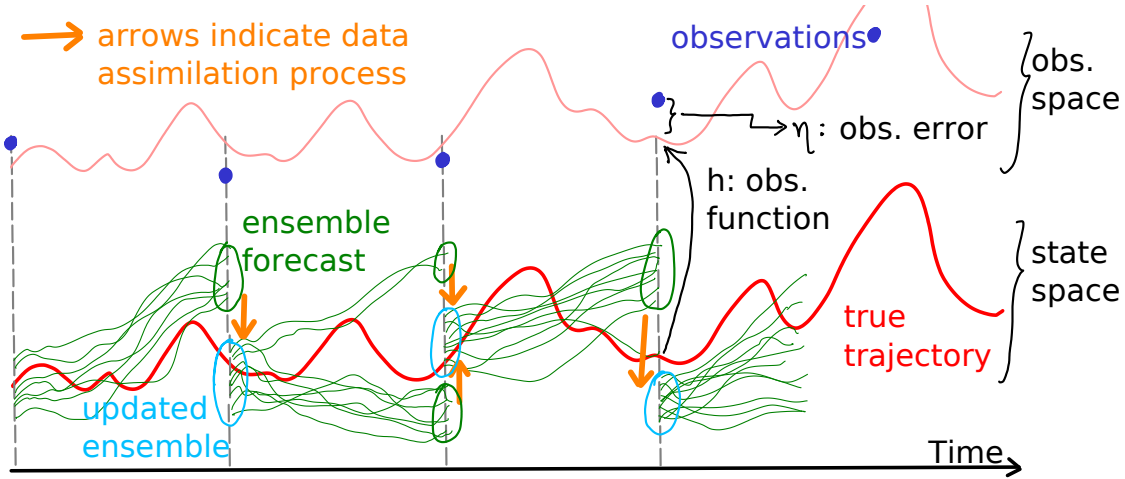
$$y_t = h(x_t) + \eta_t \qquad \text{with likelihood} \qquad p_\eta(y_t|x_t)$$

- **Main object of interest in data assimilation** (basically filtering theory in discrete time, in the context of the above framework):

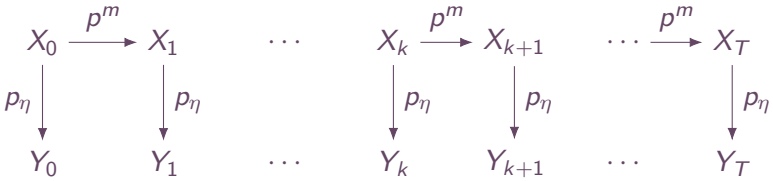
$p^a(x_t|y_{1:t})$ = posterior distribution of the state x_t at time t given observations $y_{1:t} \equiv (y_1, \ldots, y_t)$ up to time t,

- Other problems: Smoothing: $p(x_t|y_1, y_2, \ldots, y_N)$ for $t < N$; Prediction: $p(x_t|y_1, y_2, \ldots, y_N)$ for $t > N$
- 'Deterministic' (not probabilistic) formulation: inverse problem of finding a function g_t such that $x_t = g_t(y_{1:t})$

Here is a "picture" of data assimilation



And here is a more "standard" one for HMM

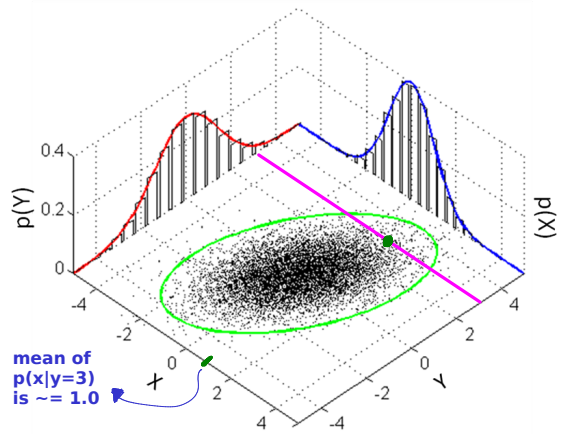


Main ingredients - some details about the "HMM" we confront

- A dynamical model: given the state $x(t) \in \mathbb{R}^d$ at any time t, gives the state $x(s)$ at any later time $s > t$: Lorenz-63, Lorenz-96, etc. (for synthetic data studies, $d = 3$ or $d = 40$ etc.) or general circulation models (for ocean / atmosphere / coupled $d = 10^7$ or $d = 10^4$)
- Observations $y_1 \in \mathbb{R}^p$ at time t_i, for $i = 1, \ldots, T$ (typically $p \ll d$)
- Observations are partial (with gaps), noisy, discrete in time
- Observation operator $h : \mathbb{R}^d \to \mathbb{R}^p$ to relate the model variables at time t with observations at the same time: if the state were $x(t)$, the observations without noise would be $h(x(t))$
- Observational "errors": need to account for the difference between how the real system is represented in the model (representativeness error) and the instrumental uncertainty (noise)

How do we represent uncertainty? Using probabilities!

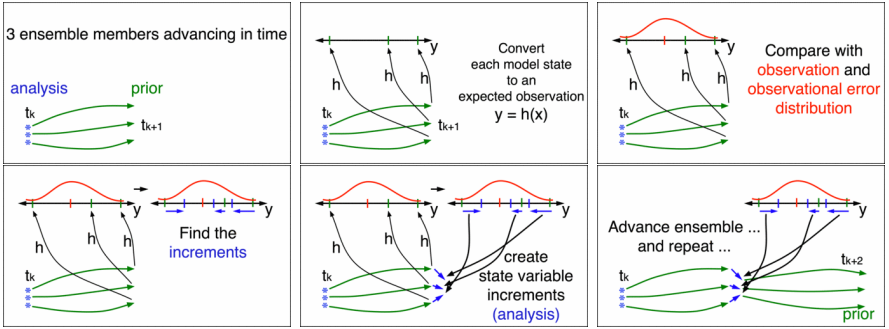
If and only if two random variables are correlated, information about one gives some information about the other



That's it: that is data assimilation! jupyter notebook: `https://tinyurl.com/37c46z6f`

So what is the big deal!? Time dependence....

we should really be watching a movie of the probability densities of the Markov process (x_t, y_t)



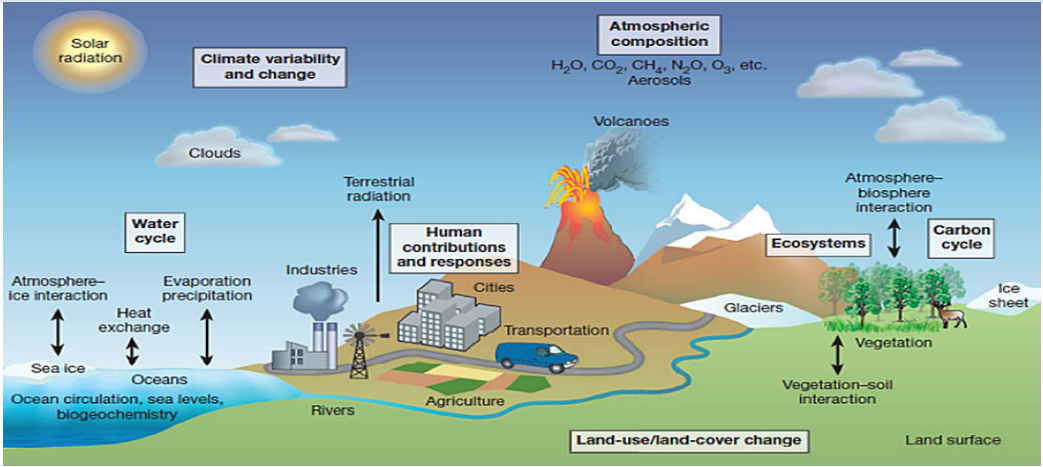
images from DART `http://www.image.ucar.edu/DAReS/images/AssimAnim.gif`

- A more general mathematical description with transition kernels e.g
  `https://web.math.princeton.edu/~rvan/orf557/hmm080728.pdf`
- For Monte Carlo approximations, Feynman–Kac models and McKean approaches to data assimilation e.g. books by del Moral "Interacting particle systems" and Reich, Cotter "probabilistic forecasting..."

Outline

The earth is a complex, high-dimensional, chaotic, dynamical system

Atmosphere, oceans, solar radiation, volcanoes, marine biology, ice and snow, clouds, precipitation, evaporation, land, rivers, lakes, CO_2, CH_4, vegetation, agriculture, ecosystems, human activities



https://www.gfdl.noaa.gov/earth-system-model/

And there is just one earth!

A few random(!) questions

- When is the first total solar eclipse in India after 2100?
- When will be the next two (or 20) perihelia of Halley's comet? (2061, 2134, ...)
- How many times in the next hour will a double pendulum reach the apogee? What will be the angle of a double pendulum after 5 min., 10 min., ...? (... video ...)
- Breaking waves – which wave will reach you?
- What will be the min/max temperatures in five largest cities in India, tomorrow, day-after, over the next month?
- What will be the major stock exchange indices tomorrow?
- What will be the number of cars that will enter the golden gate bridge in next 30 minutes?
- Who will be the prime minister of India in 2030?
- How many nuclei from a given piece of U^{235} will decay in next 10 minutes? ...

How do we predict weather? (Or forecast floods?)#

Three main ingredients (for most complex systems):

- The main, relevant **scientific theories** are fluid dynamics, electrodynamics, thermodynamics, biogeochemistry, etc. for the fluid flow, phase transitions, etc.
- **Models** are either derived from the scientific theories (e.g. quasi-geostrophic equations, radiative balance models), or phenomenological (e.g. ice, groundwater).
- Since the models, and arguably the system, are chaotic (unlike, say, the solar system on $\sim$1000-year scale), we need accurate **initial conditions and uncertainty estimates**.
- These require use of **observational data** of the earth system, and increasingly huge quantities of model **simulation data**: this is the **data assimilation problem**.

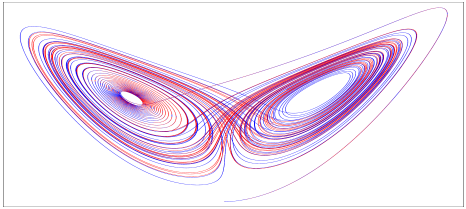
$$\textbf{Theories} \overset{*}{\longleftrightarrow} \textbf{Models} \xleftrightarrow{\text{Data Assimilation}} \textbf{Data}$$

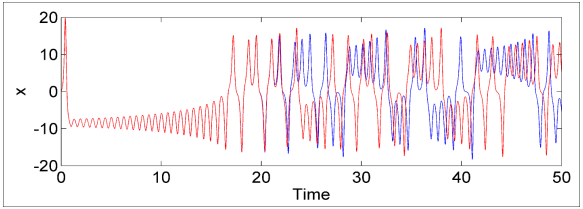
* For detailed discussion: "How the laws of physics lie?" by Nancy Cartwright

"The quiet revolution of numerical weather prediction," by Peter Bauer, Alan Thorpe, Gilbert Brunet; doi:10.1038/nature14956

Chaos necessitates probability and data!

Lorenz ODE: $\dot{x} = \sigma(y - x); \quad \dot{y} = x(\rho - z) - y; \quad \dot{z} = xy - \beta z$ (or even the solar system)





- Almost every trajectory "looks" the same and has the same statistics (ergodicity)
- But trajectories diverge away from each other exponentially (positive Lyapunov exponent)

Consequences:

- Deterministic predictions (beyond $t \sim O(1/\lambda)$, where λ is the largest Lyapunov exponent) are impossible[2]
- Probabilistic predictions (with uncertainty smaller than the long-term statistics) require "frequent" observations

jupyter notebook: `https://tinyurl.com/47djm3x7`

[2] For the solar system, uncertainties multiply by a factor of 10 every 10 My. doi:10.1073/pnas.1813901116 (Fascinating history from Poincaré, ..., to Laskar)

Drowning in the ocean of observational and simulation data

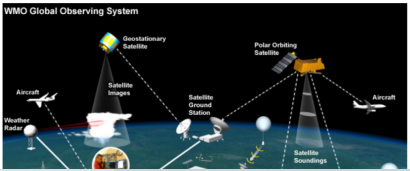
Simulation data: e.g. reanalysis of the past century climate / weather, or climate predictions of the future (usually lower resolution), is limited by

- computing and storage one is willing to utilize,
- 'fidelity' of the models, and
- data assimilation methodology used to generate the reanalysis.

Several such datasets are available:

- E.g. `climatedataguide.ucar.edu/climate-data/era5-atmospheric-reanalysis`: '*hourly data, 31 km resolution on 137 levels*': this translates to around $18M$ grid points per hour, so $\sim 1GB$ per day, for 40 years, so $\sim 20 - 100TB$ of data from one 'model'
- There are around ~ 20 (or more?) such datasets
- Similar quantities of data from other models such as those in IPCC reports

Observational data: largest quantity is satellite data, which is necessarily only for the atmosphere and ocean surface; Deep ocean observations is a major challenge



Data assimilation is a estimation problem.

Estimation of state, in time, repetitively.

- Breaking waves – which wave will reach you? (insurance)
- What will be the min/max temperatures in five largest cities in India, tomorrow, day-after, over the next month? (planning)
- What will be the average temperature in Bangalore, month by month, in 2050, or up to 2050? (design)

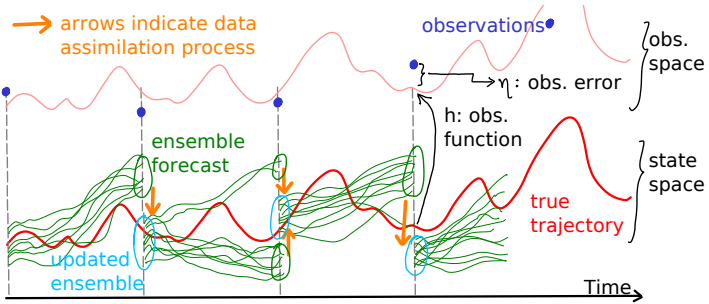
A few characteristics of data assimilation problems:

- Good physical theories, but not necessarily good models
- Systems are nonlinear and chaotic (usually deterministic)
- Multiscale – temporal and spatial – dynamics
- Observations of the system are
  - noisy
  - partial (sparse)
  - discrete in time

What is data assimilation? "Data science for chaotic dynamical systems"

The art of optimally incorporating

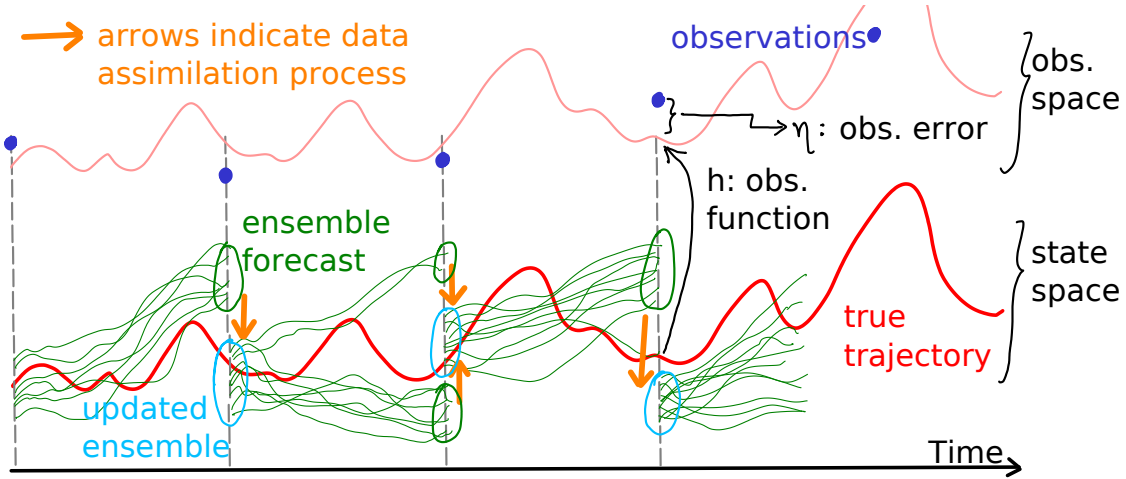
- partial and noisy observational data of a
- chaotic, nonlinear, complex dynamical system with an
- imperfect model (of the data and the system) to get an
- estimate and the associated uncertainty for the system state



- Main application: weather and climate predictions
- Also being used biology, industrial applications, etc.
- Main challenges: the complexity of the system; the uncertainties in modelling processes such as the clouds or aerosols; computational challenges

Outline

Recall the "picture" of data assimilation



Conditional density satisfies a recurrence relation

Dynamics on the space of probability distributions

$$p^a(x_{t-1}|y_{1:t-1}) \xrightarrow[\text{or prediction}]{\text{forecast}} p^f(x_t|y_{1:t-1}) \xrightarrow[\text{or analysis}]{\text{update}} p^a(x_t|y_{1:t})$$

- "prediction" uses Markov transition density (i.e. dynamical model) $p^m(x_t|x_{t-1})$

$$p^f(x_t|y_{1:t-1}) = \int p^a(x_{t-1}|y_{1:t-1}) \cdot p^m(x_t|x_{t-1}) dx_{t-1}$$

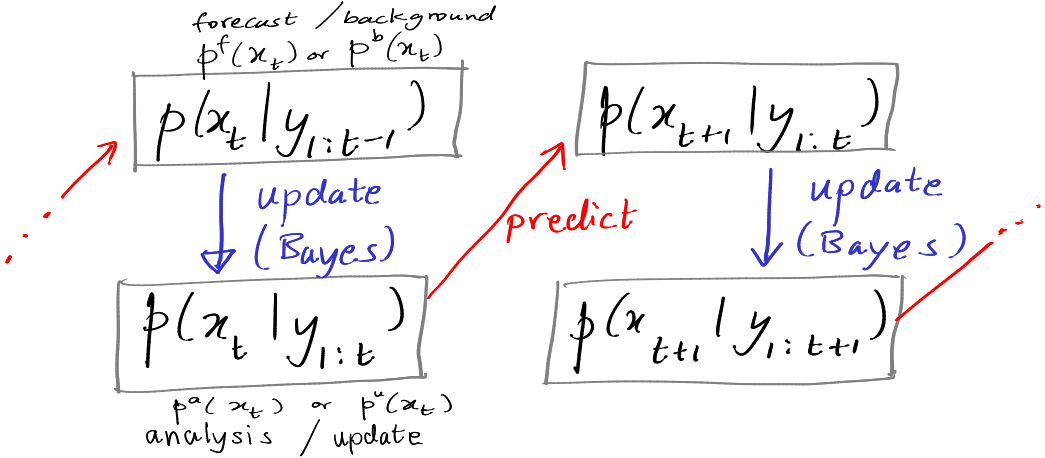
- "update" uses the likelihood $p_\eta(y_t|x_t)$ and Bayes theorem

$$p^a(x_t|y_{1:t-1}, y_t) \propto p^f(x_t|y_{1:t-1}) \cdot p_\eta(y_t|x_t)$$

Thus we obtain the following recurrence relation for the posterior distribution

$$p^a(x_t|y_{1:t}) \propto \left[\int p^a(x_{t-1}|y_{1:t-1}) \cdot p^m(x_t|x_{t-1}) dx_{t-1}\right] \cdot p_\eta(y_t|x_t)$$

Two-step process for obtaining the filtering density



Kalman filter: a "two moment" representation

The 'simple harmonic oscillator' of filtering theory: We only need mean and covariance for linear, Gaussian case

- Suppose 'everything' is linear, Gaussian: the Markov model $p^m(x_t|x_{t-1})$; The observation operator $h(x) = Hx$; The initial distribution for x_0 and the observation noise η_t
- Kalman filter gives a recursion relation for the mean and covariance: $p^a(x_t|y_{1:t}) \sim \mathcal{N}(x_t^a, C_t^a)$ and $p^f(x_{t+1}|y_{1:t}) \sim \mathcal{N}(x_{t+1}^f, C_{t+1}^f)$:
  - "Update step" given by

$$x_t^a = x_t^f + K(y_t - Hx_t^f) \qquad \text{and} \qquad C_t^a = (I - KH)C_t^f$$

  - Here $K = P_t^f H^T (HP_t^f H^T + R)^{-1}$ is the Kalman gain matrix
  - "Prediction step" given by

$$x_{t+1}^f = Mx_t^a \qquad \text{and} \qquad C_{t+1}^f = MC_t^a M^T$$

  jupyter notebook: `https://tinyurl.com/2aj7wz2y`
- But what do we do when C_t is $10^6 \times 10^6$ dimensional? Ensemble Kalman filter is the Monte Carlo version of this filter.

Ensemble (Monte Carlo) approximation of the recursion

Recall

- "prediction" $p^f(x_t|y_{1:t-1}) = \int p^a(x_{t-1}|y_{1:t-1}) \cdot p^m(x_t|x_{t-1}) dx_{t-1}$
- "update" $p^a(x_t|y_{1:t-1}, y_t) \propto p^f(x_t|y_{1:t-1}) \cdot p_\eta(y_t|x_t)$

Monte Carlo approximation (think of 'weighted histogram'!):

$p^{a/f}(x_t|...) \approx \sum_{i=1}^N w_{t,i}^{a/f} \delta(x_t - x_{t,i}^{a/f})$ using weighted sample $\left\{(w_{t,i}^{a/f}, x_{t,i}^{a/f})\right\}_{i=1}^N$

Two important classes of methods:

- Ensemble Kalman filter (EnKF): is all about the states 'x' (with $w_{t,i}^{a/f} = 1/N$)! Key step:

$$x_{t,i}^a = x_{t,i}^f + K(y_t^i - Hx_{t,i}^f)$$

- Particle filter (PF): is all about the weights 'w'! Key step:

$$w_{t,i}^a \propto w_{t,i}^f p_\eta(y_t|x_{t,i}^f)$$

Particle filter: a "weighted sample" representation

Recall

- "prediction" $p^f(x_t|y_{1:t-1}) = \int p^a(x_{t-1}|y_{1:t-1}) \cdot p^m(x_t|x_{t-1}) dx_{t-1}$
- "update" $p^a(x_t|y_{1:t-1}, y_t) \propto p^f(x_t|y_{1:t-1}) \cdot p_\eta(y_t|x_t)$

- PF summary: $(w^a_{t-1,i}, x^a_{t-1,i}) \xrightarrow[\text{unchanged } w]{\text{forecast } x} (w^f_{t,i} = w^a_{t-1,i}, x^f_{t,i}) \xrightarrow[\text{reweighted } w]{\text{unchanged } x} (w^a_{t,i}, x^f_{t,i} = x^f_{t,i})$
- if $x^f_{t,i}$ is a sample from an "importance sampling density" $q(x_t|x^a_{t-1,i}, \dots)$:

$$x^f_{t,i} \sim q(x_t|x^a_{t-1,i}, \dots) \qquad \text{e.g. could be } \; p^m(x_t|x^a_{t-1,i})$$

- then the weighted sample $\{w^a_{t,i}, x^f_{t,i}\}_{i=1}^N$ approximates the posterior at time t if we choose

$$w^a_{t,i} \propto w^a_{t-1,i} \cdot \frac{p^m(x^f_{t,i}|x^a_{t-1,i}) \cdot p_\eta(y_t|x^f_{t,i})}{q(x^f_{t,i}|x^a_{t-1,i}, \dots)}$$

"Resample if necessary" (because 'many' $w^a_{t,i}$ may become very small)

Ensemble Kalman filter: a "two moment" representation

Recall

- "prediction" $p^f(x_t|y_{1:t-1}) = \int p^a(x_{t-1}|y_{1:t-1}) \cdot p^m(x_t|x_{t-1}) dx_{t-1}$
- "update" $p^a(x_t|y_{1:t-1}, y_t) \propto p^f(x_t|y_{1:t-1}) \cdot p_\eta(y_t|x_t)$

- EnKF summary ($w_i = 1/N$): $(x^a_{t-1,i}) \xrightarrow[\text{unchanged } w]{\text{forecast } x} (x^f_{t,i}) \xrightarrow[\text{unchanged } w]{\text{'shift' } x \text{ using KF}} (x^a_{t,i})$
- $x^f_{t,i}$ is a sample from a Markov transition density $p^m(x_t|x^a_{t-1,i})$

$$x^f_{t,i} \sim p^m(x_t|x^a_{t-1,i})$$

- Update step is like a 'weighted linear combination of forecast and observations'

$$x^a_{t,i} = x^f_{t,i} + K(y^i_t - Hx^f_{t,i}) = (I - KH)x^f_{t,i} + Ky^i_t$$

- $K = \hat{P}^f_t H^T (H\hat{P}^f_t H^T + R)^{-1}$ uses sample covariance $\hat{P}^f_t$.
- **Commonly used version: square-root filters** with localization and inflation jupyter notebook: `https://tinyurl.com/yeykzvbp`

Contrasting the properties of particle and Kalman filters

On one hand,

- Particle filter has sampling errors ($\sim C/\sqrt{N}$), but the errors (C) grow exponentially with the dimension of the state space ("curse of dimensionality")[3],
- but it does not have any restrictions about the dynamics being linear.

On the other hand,

- Ensemble Kalman filter is obviously designed for linear systems - for sufficiently nonlinear systems, it fails to represent the true posterior distribution, (examples below)[4]
- but it seems to work well[5] even in high dimensional systems with very small ensembles ($N \sim 100$ for systems with $d \sim 10^6$!)

---

[3] e.g. Rebeschini and van Handel, *Can local particle filters beat the curse of dimensionality?*, Ann.Appl.Probab., V.25 (2015), p.2809

[4] e.g. Apte, Jones, *The impact of nonlinearity in Lagrangian data as- similation*, Nonlin.Proc.Geo. v.20 (2013) p.329

[5] "seems to capture the truth" but how about the true posterior?

Curse of dimensionality for particle filters

- If π_N^f is the particle filter approximation with N particles of the exact filtering distribution π for a D-dimensional problem, the error is

$$\|\pi_N^f - \pi\| \sim \frac{e^{\alpha D}}{\sqrt{N}}$$

- If we divide the system into blocks such that the dynamics is approximately local within each block, the above result can be improved[6] to

$$\|\pi_N^f - \pi\| \sim \frac{e^{\alpha d}}{\sqrt{N}} + e^{-\beta r}$$

where r is the radius of the "local block" and d is its dimension. (e.g. discretization of n-dimensional PDE will give $d \sim r^n$.)

[6] P. Rebeschini, R. van Handel, "Can local particle filters beat the curse of dimensionality?"

Data assimilation is the art of optimally incorporating

- partial and noisy observational data of a
- chaotic, nonlinear, complex dynamical system with an
- imperfect model (of data noise and system dynamics) to get an
- estimate and the associated uncertainty for the system state

---

Main challenges:

- Curse of dimensionality: sampling methods need exponentially large number of samples (computational challenges)
- complex dynamical system: require innovative approaches, such as localization and inflation (conceptual challenges), and theoretical guarantees (mathematical challenges)
- state estimation and uncertainty quantification: several new approaches, such as multi-level Monte Carlo, interacting particle systems