Statistical Data Analysis

Jana de Wiljes

wiljes@uni-potsdam.de

www.dewiljes-lab.com

8. November 2022

Universität Potsdam

Sequential Learning Algorithm

New challenge: Need to learn Θ sequentially or online, i.e.,

$$\Theta^j = \arg\min_{\Theta \in \Omega} R_j(h_j(\cdot, \Theta^j | \Theta^{j-1}))$$

But why?

- sequential decision involved, i.e., $h(\cdot, \Theta, a_t)$
- data can only be collected individually in time and we already want to start predicting
- nonstationary Θ_t

Sequential update of Linear Regression Parameter

Choose: $h(x, \Theta) = x\Theta$ and we assume $y = h(x, \Theta) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, 1)$

Linear regression with batch data:

$$R_N(h) = \frac{1}{N}\sum_{i=1}^{N}(y_i - x_i\Theta)^2 \quad \text{where } \Theta_N^* = (\sum_{i=1}^{N} x_i^2)^{-1}(\sum_{i=1}^{N} x_i y_i)$$

Sherman-Morrison formula

$$\left(A + uv^\mathsf{T}\right)^{-1} = A^{-1} - \frac{A^{-1}uv^\mathsf{T}A^{-1}}{1 + v^\mathsf{T}A^{-1}u}. \tag{1}$$

New data: (x_{N+1}, y_{N+1}) yield a new parameter

$$\Theta^* = (\sum_{i=1}^{N} x_i^2 + x_{N+1}x_{N+1})^{-1}(\sum_{i=1}^{N} x_i y_i + x_{N+1}y_{N+1}) \tag{2}$$

Using the Sherman-Morrison formula we can update recursively.

New challenge

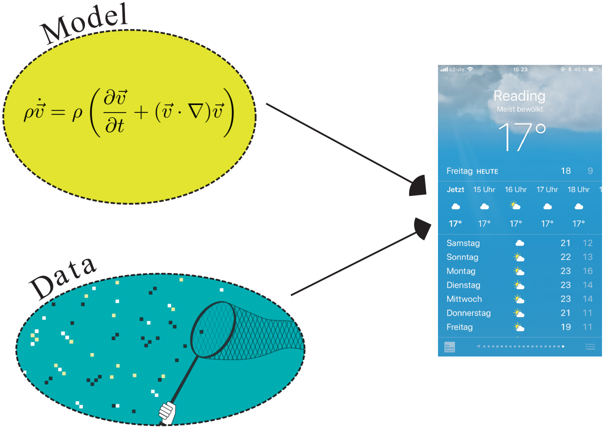
Setting:

- h is known and links noisy and partial observations y to x
- x is not given directly as a sample but we have a model f that we can use as a surrogate to generate samples

Goal: estimating the associate density conditioned on the data and using the prior information given via f

What is Data Assimilation?

Data Assimilation

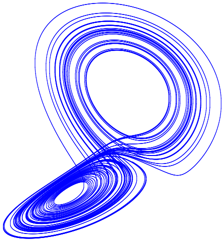


Uncertainty?

A toy atmospheric model

Lorenz equations:

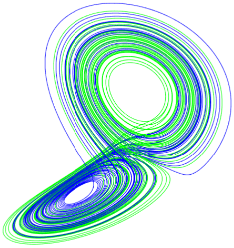
$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = x(\rho - z) - y$$
$$\dot{z} = xy - \beta z$$



Uncertainty in initial conditions

Lorenz equations:

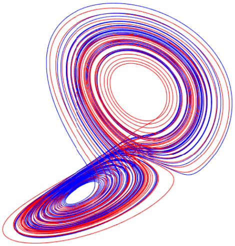
$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = x(\rho - z) - y$$
$$\dot{z} = xy - \beta z$$



Uncertainty in parameters

Lorenz equations:

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = x(\rho - z) - y$$
$$\dot{z} = xy - \beta z$$



Numerical discretization and differentiation

Lorenz equations:

$$
\begin{aligned}
x_n &= x_{n-1} + [\sigma(y_{n-1} - x_{n-1})]\mathrm{dt} \\
y_n &= y_{n-1} + [x_{n-1}(\rho - z_{n-1}) - x_{n-1}]\mathrm{dt} \\
z_n &= z_{n-1} + [x_{n-1}y_{n-1} - \beta z_{n-1}]\mathrm{dt}
\end{aligned}
$$

The Model

Model (deterministic)

Evolution equation

$$z_n = \Psi(z_{n-1}, \lambda)$$

where

$$z_0 \sim \mathcal{N}(m_0, C_0)$$

Model

Evolution equation

$$z_n = \Psi(z_{n-1}, \lambda) + \xi_{n-1}$$

where

$$z_0 \sim \mathcal{N}(m_0, C_0)$$
$$\xi_n \sim \mathcal{N}(0, B) \quad \text{i.i.d.} \quad \forall\ n$$

Parameter estimation

Augmented state space

$$\begin{aligned} z_n &= \Psi(z_{n-1}, \lambda_{n-1}) + \xi_{n-1} \\ \lambda_n &= \lambda_{n-1} \end{aligned}$$

where

$$\begin{aligned} [z_0, \lambda_0]^\top &\sim \mathcal{N}(m_0, C_0) \\ \xi_n &\sim \mathcal{N}(0, B) \quad \text{i.i.d.} \quad \forall\ n \end{aligned}$$

Observations

Observations

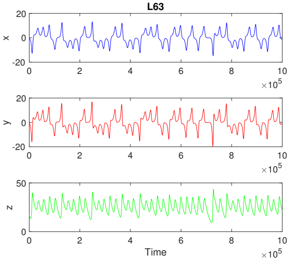
Partial and noisy data:

$$y_n = h(z_n) + \eta_n$$

where

$$\eta_n \sim \mathcal{N}(0, R) \quad \text{i.i.d.} \quad \forall\ n$$

Example



Lorenz equations:

$$x_n = x_{n-1} + [\sigma(y_{n-1} - x_{n-1})]\mathrm{dt}$$

$$y_n = y_{n-1} + [x_{n-1}(\rho - z_{n-1}) - x_{n-1}]\mathrm{dt}$$

$$z_n = z_{n-1} + [x_{n-1}y_{n-1} - \beta z_{n-1}]\mathrm{dt}$$

The Math behind it...

Conditional probability

Definition (Conditional probability)

For a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and events $A, B \in \mathcal{F}$ the conditional probability of B given A is

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A, B)}{\mathbb{P}(B)}.$$

Bayes theorem

Theorem (Bayes)

For a given probability space $(\Omega, \mathcal{F}, \mathbb{P})$ *the following holds for two events* A *and* B *in* $\mathcal{F}$

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(B|A)\mathbb{P}(A)}{\mathbb{P}(B)}.$$

Bayesian data assimilation ansatz

$$\mathbb{P}(\text{Model}|\text{Obs}) = \frac{\mathbb{P}(\text{Obs}|\text{Model})\mathbb{P}(\text{Model})}{\mathbb{P}(\text{Obs})}$$

Bayesian data assimilation ansatz

$$\mathbb{P}(\mathsf{Model}|\mathsf{Obs}) \propto \mathbb{P}(\mathsf{Obs}|\mathsf{Model})\mathbb{P}(\mathsf{Model})$$

Bayesian data assimilation for densities

$$
\begin{aligned}
\pi(z_{n+1}|y_{1:n+1}) &= \pi(z_{n+1}|y_{1:n}, y_{n+1}) \\
&= \frac{\pi(y_{n+1}|y_{1:n}, z_{n+1})\pi(z_{n+1}|y_{1:n})}{\pi(y_{n+1}|y_{1:n})} \\
&= \frac{\pi(y_{n+1}|z_{n+1})\pi(z_{n+1}|y_{1:n})}{\pi(y_{n+1}|y_{1:n})}
\end{aligned}
$$

$$
\implies \pi(z_{n+1}|y_{1:n+1}) \propto \pi(y_{n+1}|z_{n+1})\pi(z_{n+1}|y_{1:n}) \tag{3}
$$

Special case

Linear model: Ψ is linear, e.g.,

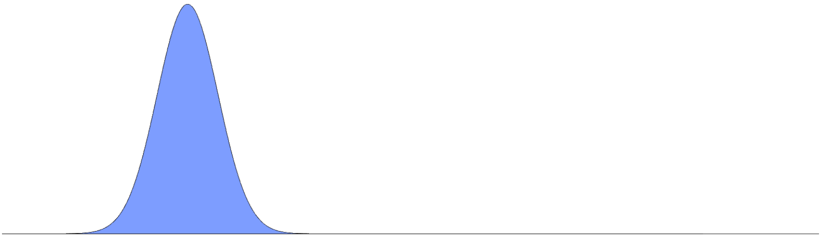
$$z_n = Az_{n-1} + \xi_{n-1} \tag{4}$$

with $A \in \mathbb{R}^{N_z} \times \mathbb{R}^{N_z}$

Linear observation operator

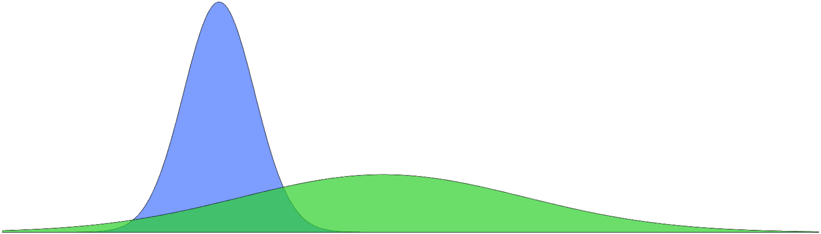
$$h = H \quad \text{with} \quad H \in \mathbb{R}^{N_y} \times \mathbb{R}^{N_y}$$

Linear model



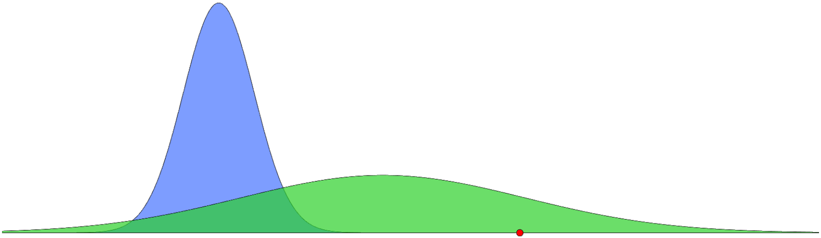
Initial distribution: $z_0 \sim \mathcal{N}(m_0, C_0)$

Linear model



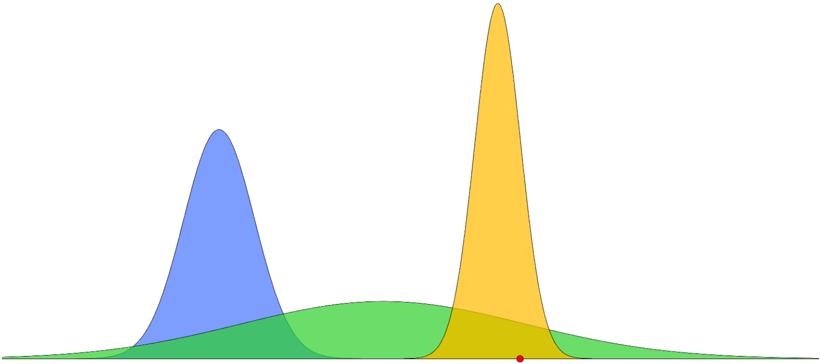
Prior distribution: $\mathcal{N}(\hat{m}_1, \hat{C}_1)$

Linear model



Likelihood: $\mathcal{N}(H\hat{m}_1, R)$

Linear model



Posterior: $\mathcal{N}(m_1, C_1) \propto \mathcal{N}(H\hat{m}_1, R)\mathcal{N}(\hat{m}_1, \hat{C}_1)$

Algorithms

Kalman filter

Two steps:

- Forecast: $(m_n, C_n) \mapsto (\hat{m}_{n+1}, \hat{C}_{n+1})$
- Analysis: $(\hat{m}_{n+1}, \hat{C}_{n+1}) \mapsto (m_{n+1}, C_{n+1})$

Forecast formulas

$$\begin{aligned} \hat{m}_{n+1} &= Am_n \\ \hat{C}_{n+1} &= AC_nA^\top + B \end{aligned}$$

Analysis formulas

$$\begin{aligned} m_{n+1} &= \hat{m}_{n+1} - K_{n+1}(H\hat{m}_{n+1} - y_{n+1}) \\ C_{n+1} &= \hat{C}_{n+1} - K_{n+1}H\hat{C}_{n+1} \end{aligned}$$

Kalman gain

$$K_{n+1} = \hat{C}_{n+1}H^\top(R + H\hat{C}_{n+1}H^\top)^{-1}$$

Kalman filter

Two steps:

- Forecast: $(m_n, C_n) \mapsto (\hat{m}_{n+1}, \hat{C}_{n+1})$
- Analysis: $(\hat{m}_{n+1}, \hat{C}_{n+1}) \mapsto (m_{n+1}, C_{n+1})$

Forecast formulas

$$\hat{m}_{n+1} = A m_n$$
$$\hat{C}_{n+1} = A C_n A^\top + B$$

Analysis formulas

$$m_{n+1} = \hat{m}_{n+1} - K_{n+1}(H\hat{m}_{n+1} - y_{n+1})$$
$$C_{n+1} = \hat{C}_{n+1} - K_{n+1} H \hat{C}_{n+1}$$

Kalman gain

$$K_{n+1} = \hat{C}_{n+1} H^\top (R + H\hat{C}_{n+1} H^\top)^{-1}$$

Kalman filter

Two steps:

- Forecast: $(m_n, C_n) \mapsto (\hat{m}_{n+1}, \hat{C}_{n+1})$
- Analysis: $(\hat{m}_{n+1}, \hat{C}_{n+1}) \mapsto (m_{n+1}, C_{n+1})$

Forecast formulas

$$\hat{m}_{n+1} = A m_n$$
$$\hat{C}_{n+1} = A C_n A^\top + B$$

Analysis formulas

$$m_{n+1} = \hat{m}_{n+1} - K_{n+1}(H\hat{m}_{n+1} - y_{n+1})$$
$$C_{n+1} = \hat{C}_{n+1} - K_{n+1} H \hat{C}_{n+1}$$

Kalman gain

$$K_{n+1} = \hat{C}_{n+1} H^\top (R + H\hat{C}_{n+1} H^\top)^{-1}$$

Kalman filter

Two steps:

- Forecast: $(m_n, C_n) \mapsto (\hat{m}_{n+1}, \hat{C}_{n+1})$
- Analysis: $(\hat{m}_{n+1}, \hat{C}_{n+1}) \mapsto (m_{n+1}, C_{n+1})$

Forecast formulas

$$\begin{aligned}\hat{m}_{n+1} &= A m_n \\ \hat{C}_{n+1} &= A C_n A^\top + B\end{aligned}$$

Analysis formulas

$$\begin{aligned}m_{n+1} &= \hat{m}_{n+1} - K_{n+1}(H\hat{m}_{n+1} - y_{n+1}) \\ C_{n+1} &= \hat{C}_{n+1} - K_{n+1} H \hat{C}_{n+1}\end{aligned}$$

Kalman gain

$$K_{n+1} = \hat{C}_{n+1} H^\top (R + H\hat{C}_{n+1}H^\top)^{-1}$$

Kalman filter

Two steps:

- Forecast: $(m_n, C_n) \mapsto (\hat{m}_{n+1}, \hat{C}_{n+1})$
- Analysis: $(\hat{m}_{n+1}, \hat{C}_{n+1}) \mapsto (m_{n+1}, C_{n+1})$

Forecast formulas

$$
\begin{aligned}
\hat{m}_{n+1} &= A m_n \\
\hat{C}_{n+1} &= A C_n A^\top + B
\end{aligned}
$$

Analysis formulas

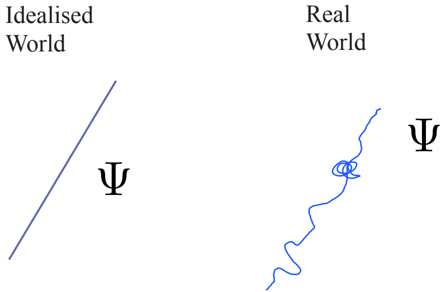
$$
\begin{aligned}
m_{n+1} &= \hat{m}_{n+1} - K_{n+1}(H\hat{m}_{n+1} - y_{n+1}) \\
C_{n+1} &= \hat{C}_{n+1} - K_{n+1} H \hat{C}_{n+1}
\end{aligned}
$$

Kalman gain

$$
K_{n+1} = \hat{C}_{n+1} H^\top (R + H\hat{C}_{n+1} H^\top)^{-1}
$$

Kalman filter

Reality check

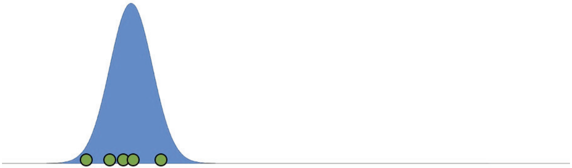


Problem: Kalman Filter is not applicable anymore

Approach: use approximative Algorithms, e.g.,

- Extended Kalman Filter: linearize model function

Monte Carlo approximation



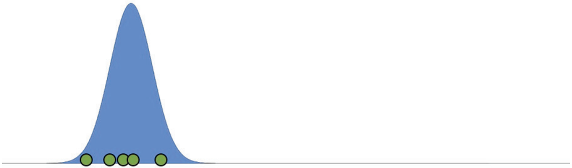
Approach: approximative via empirical measure

$$\pi(z_n|y_{1:n}) = \frac{1}{M}\sum_{i=1}^{M}\delta(z - z_n^i)$$

where

$$z_n^i \sim \pi(z_n|y_n)$$

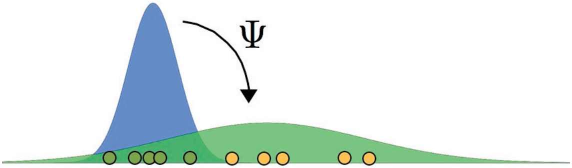
Ensemble Kalman filter



$$\mathcal{N}(m_0, C_0) \text{ with } m_0 \approx \frac{1}{M}\sum_{i=1}^{M} z_0^i$$

$$C_0 \approx \frac{1}{M}\sum_{i=1}^{M}(z_0^i - m_0)(z_0^i - m_0)^\top$$

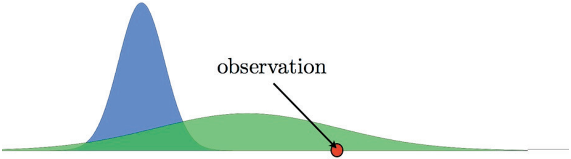
Ensemble Kalman filter



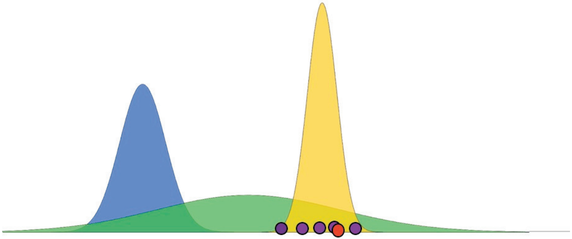
$$\mathcal{N}(\hat{m}_1, \hat{C}_1) \text{ with } \hat{m}_1 \approx \frac{1}{M}\sum_{i=1}^{M} \hat{z}_1^i = \frac{1}{M}\sum_{i=1}^{M} \Psi(z_0^i)$$

$$\hat{C}_1 \approx \frac{1}{M}\sum_{i=1}^{M} (\hat{z}_1^i - \hat{m}_1)(\hat{z}_1^i - \hat{m}_1)^\top$$

Ensemble Kalman filter



Ensemble Kalman filter



$$\mathcal{N}(m_1, C_1) \text{ with } m_1 \approx \frac{1}{M}\sum_{i=1}^{M} z_1^i$$

$$C_1 \approx \frac{1}{M}\sum_{i=1}^{M}(z_1^i - m_1)(z_1^i - m_1)^\top$$

Ensemble Kalman filter

Goal: approximate $\pi(z_n|y_{1:n})$

Approach: propagate samples $\hat{z}^i_{n+1}$ with Kalman formula

$$z^i_{n+1} = \hat{z}^i_{n+1} - K_{n+1}(H\hat{z}^i_{n+1} - \tilde{y}^i_{n+1})$$

Need: perturbed observations

$$\tilde{y}^i_{n+1} = y_{n+1} + \epsilon^i_{n+1}$$

with $\epsilon^i_{n+1} \sim \mathcal{N}(0, R)$ i.i.d. to get the correct mean and covariance in the linear case for $M \to \infty$

Ensemble Kalman filter

Goal: approximate $\pi(z_n | y_{1:n})$

Approach: propagate samples $\hat{z}^i_{n+1}$ with Kalman formula

$$z^i_{n+1} = \hat{z}^i_{n+1} - K_{n+1}(H\hat{z}^i_{n+1} - \tilde{y}^i_{n+1})$$

Need: perturbed observations

$$\tilde{y}^i_{n+1} = y_{n+1} + \epsilon^i_{n+1}$$

with $\epsilon^i_{n+1} \sim \mathcal{N}(0, R)$ i.i.d. to get the correct mean and covariance in the linear case for $M \to \infty$

Ensemble Kalman filter

Goal: approximate $\pi(z_n|y_{1:n})$

Approach: propagate samples $\hat{z}^i_{n+1}$ with Kalman formula

$$z^i_{n+1} = \hat{z}^i_{n+1} - K_{n+1}(H\hat{z}^i_{n+1} - \tilde{y}^i_{n+1})$$

Need: perturbed observations

$$\tilde{y}^i_{n+1} = y_{n+1} + \epsilon^i_{n+1}$$

with $\epsilon^i_{n+1} \sim \mathcal{N}(0, R)$ i.i.d. to get the correct mean and covariance in the linear case for $M \to \infty$

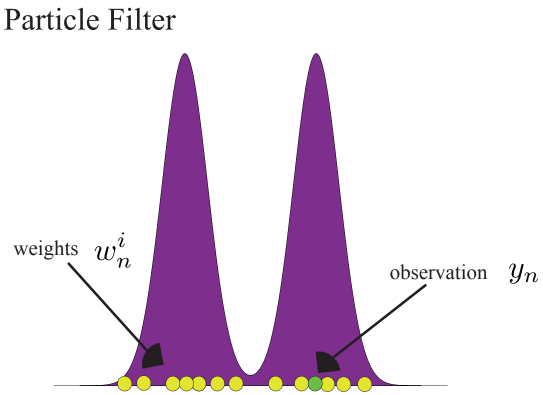
Ensemble Kalman filter

Works well in practice: e.g., EnKF is used for operational NWP for z_n^i with dimension 10^8 only using $M = 100$

Yet: mathematical foundation largely missing

Recent study: accuracy results for EnKF for idealized setting: $H = Id$ and observational error small

Particle filter



Particle filter

Problem: sampling from $\pi(z_n|y_{1:n})$ to approximate posterior via

$$\pi(z_n|y_{1:n}) = \frac{1}{M}\sum_{i=1}^{M}\delta(z - z_n^i)$$

Idea: sampling from $\pi(z_n|y_{1:n-1})$ instead i.e.,

$$\pi(z_n|y_{1:n}) = \sum_{i=1}^{M} w_n^i \delta(z - \hat{z}_n^i)$$

Bayes:

$$\pi(z_{n+1}|y_{1:n}) \propto \pi(y_n|z_n)\pi(z_n|y_{1:n-1}) \tag{5}$$

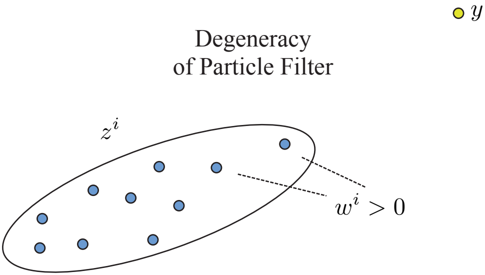
Weighting: unnormalized weights

$$\tilde{w}_n^i = \pi(y_n|z_n^i)w_{n-1}^i \text{ with } w_0^i = \frac{1}{M}$$

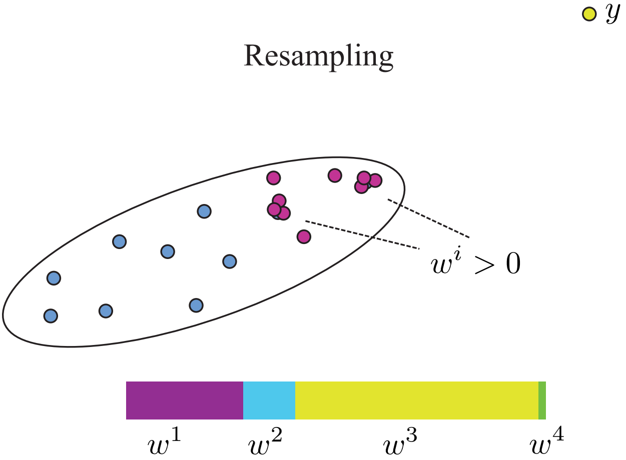
and normalized weights

$$w_n^i = \frac{\tilde{w}_n^i}{\sum_{j=1}^{M}\tilde{w}_n^j}$$

Particle collapse



Resampling



Resampling

Problem: weights w_n^i become very small

Ansatz: resampling

Input: w_n^i

For$(k = 1 : M)$

1. Draw a number $u \in [0, 1]$ from the uniform distribution $U[0, 1]$
2. Compute $i^* \in \{1, \dots, M\}$ which satisfies

$$i^* = \arg\min_{i \geq 1} \sum_{j=1}^{i} w_j \geq u \tag{6}$$

3. Set $\xi_{i^*} = \xi_{i^*} + 1$

Return ξ_i

Still a lot of challenges....

Ansatz: approximative via empirical measure

$$\pi(z_n|y_{1:n}) = \frac{1}{M}\sum_{i=1}^{M}\delta(z - z_n^i)$$

where

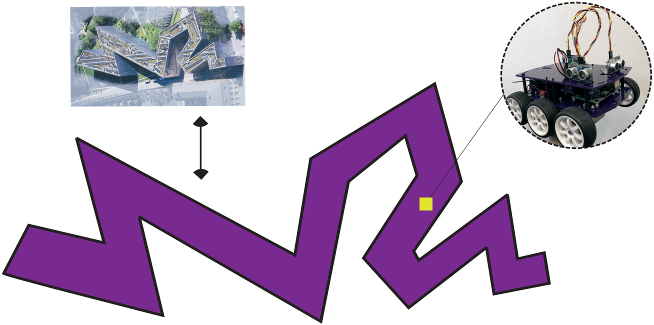
$$z_n^i \sim \pi(z_n|y_n)$$

Monte Carlo approximation leads to a variety of filters e.g.,

- Particle filters (curse of dimensionality)
- Ensemble Kalman filter (underlying Gaussian assumption)

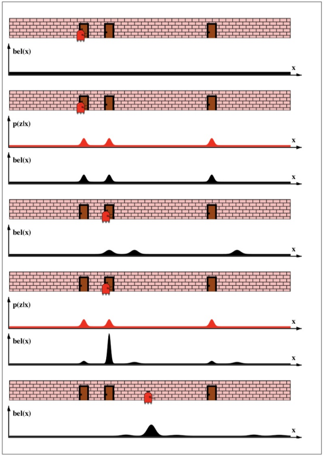
Same Math different Applications

Robotics

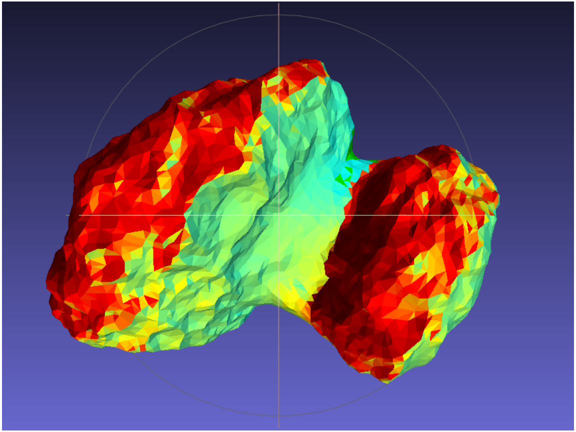


Simultaneous state and parameter estimation

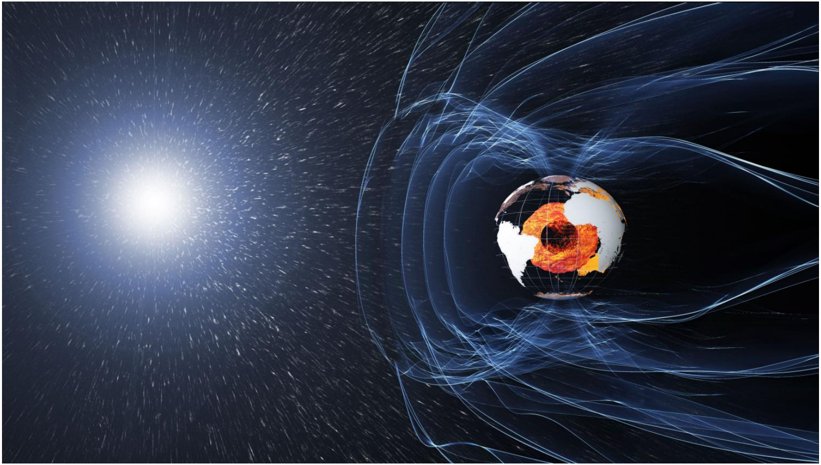
Robotics



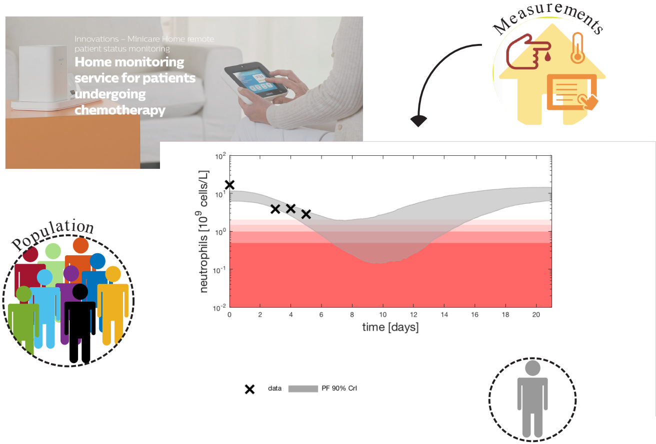
Thermophysical modeling



Space weather



Pharmacology



Personalised medicine