SDEs and filtering

Erik Lindström / Petar Jovanovski

Centre for Mathematical Sciences
Lund University

LU/LTH & DTU



LUND UNIVERSITY

Overview

Stochastic Differential Equations - SDEs

- Ideas
- Technical properties
- Itô formula

Filtering - Partially observed Models

- Optimal filtering
- Kalman filters
- SMC filters

General

Generalize classic ODE

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \mu(t, X_t)$$

to stochastic ODE

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \mu(t, X_t) + \sigma(t, X_t)v_t,$$

where v_t is a formal time derivative of a continuous time random walk.

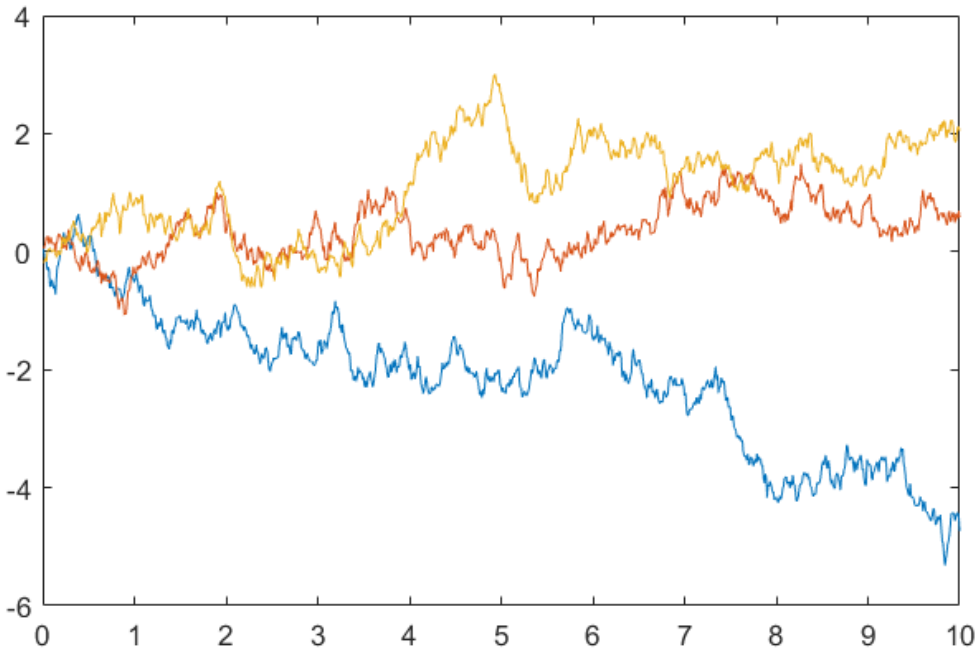
Such object cannot be defined!!!

This is an applied course. We try to preserve the relevant properties.

Derivative of a standard Brownian motion

A **standard Brownian motion** is a stochastic process satisfying

1. $W(0) = 0$ *a.s.*
2. $W(t)$ is continuous w.p.1.
3. $W(t) - W(s)$ is indep. of $W(u) - W(0)$ for $t > s \geq u > 0$
4. $W(t) - W(s) \sim N(0, t - s)$



Other types of continuous time random walks

- Poission process $N(t)$ or compensated Poisson process $N(t) - \lambda t$. It can be defined as $Y(t) = \sum_{n=1}^{N(t)} 1$. (Note the convention $\sum_{n=1}^{0} 1 \triangleq 0$)
- Compound Poisson process $Y(t) = \sum_{n=1}^{N(t)} \xi_n$, where ξ_n are *iid* r.v.s
- Levy processes, $L(\mu, \sigma, \nu)$, generalizing Brownian motion and infinite activity compound Poisson processes, where μ is the drift, σ the diffusion and ν is a jump measure.

Other types of continuous time random walks

- Poission process $N(t)$ or compensated Poisson process $N(t) - \lambda t$. It can be defined as $Y(t) = \sum_{n=1}^{N(t)} 1$. (Note the convention $\sum_{n=1}^{0} 1 \triangleq 0$)
- Compound Poisson process $Y(t) = \sum_{n=1}^{N(t)} \xi_n$, where ξ_n are *iid* r.v.s
- Levy processes, $L(\mu, \sigma, \nu)$, generalizing Brownian motion and infinite activity compound Poisson processes, where μ is the drift, σ the diffusion and ν is a jump measure.

Other types of continuous time random walks

- Poission process $N(t)$ or compensated Poisson process $N(t) - \lambda t$. It can be defined as $Y(t) = \sum_{n=1}^{N(t)} 1$. (Note the convention $\sum_{n=1}^{0} 1 \triangleq 0$)
- Compound Poisson process $Y(t) = \sum_{n=1}^{N(t)} \xi_n$, where ξ_n are *iid* r.v.s
- Levy processes, $L(\mu, \sigma, \nu)$, generalizing Brownian motion and infinite activity compound Poisson processes, where μ is the drift, σ the diffusion and ν is a jump measure.

Other types of continuous time random walks

- Poission process $N(t)$ or compensated Poisson process $N(t) - \lambda t$. It can be defined as $Y(t) = \sum_{n=1}^{N(t)} 1$. (Note the convention $\sum_{n=1}^{0} 1 \triangleq 0$)
- Compound Poisson process $Y(t) = \sum_{n=1}^{N(t)} \xi_n$, where ξ_n are *iid* r.v.s
- Levy processes, $L(\mu, \sigma, \nu)$, generalizing Brownian motion and infinite activity compound Poisson processes, where μ is the drift, σ the diffusion and ν is a jump measure.

Derivative of a Brownian motion

- Try to use the formal definition of a derivative

$$v(t) = \frac{\mathrm{d}W}{\mathrm{d}t}(t) = \lim_{h \to 0} \frac{W(t+h) - W(t)}{h} \tag{1}$$

- Note that $\frac{W(t+h)-W(t)}{h} \in N\left(0, \frac{h}{h^2}\right)$.
- **Thus, the derivative process would have zero mean and infinite variance!**
- We have to define Stochastic Differential Equations carefully to obtain a meaningful mathematical object.

Derivative of a Brownian motion

- Try to use the formal definition of a derivative

$$v(t) = \frac{\mathrm{d}W}{\mathrm{d}t}(t) = \lim_{h \to 0} \frac{W(t+h) - W(t)}{h} \tag{1}$$

- Note that $\frac{W(t+h)-W(t)}{h} \in N\left(0, \frac{h}{h^2}\right)$.
- **Thus, the derivative process would have zero mean and infinite variance!**
- We have to define Stochastic Differential Equations carefully to obtain a meaningful mathematical object.

Derivative of a Brownian motion

- Try to use the formal definition of a derivative

$$v(t) = \frac{\mathrm{d}W}{\mathrm{d}t}(t) = \lim_{h \to 0} \frac{W(t+h) - W(t)}{h} \tag{1}$$

- Note that $\frac{W(t+h)-W(t)}{h} \in N\left(0, \frac{h}{h^2}\right)$.
- **Thus, the derivative process would have zero mean and infinite variance!**
- We have to define Stochastic Differential Equations carefully to obtain a meaningful mathematical object.

Interpretation of μ and σ

We are interested in properties when modelling using SDEs

- $\mu = \lim_{h \to 0} \frac{1}{h} \mathbf{E}[X_{t+h} - X_t | X_t]$.
- $\sigma\sigma^T = \lim_{h \to 0} \frac{1}{h} Var[X_{t+h} - X_t | X_t]$.
- Thus $\frac{\mathrm{d}X}{\mathrm{d}t} = \mu(t, X_t) + "noise"$,
- where $\mathbf{E}[noise] = 0$ *regardless of* $\sigma(\cdot)$. Almost true...

These choices conditions have far-reaching consequences!

Interpretation of μ and σ

We are interested in properties when modelling using SDEs

- $\mu = \lim_{h \to 0} \frac{1}{h} \mathbf{E}[X_{t+h} - X_t | X_t]$.
- $\sigma\sigma^T = \lim_{h \to 0} \frac{1}{h} Var[X_{t+h} - X_t | X_t]$.
- Thus $\frac{\mathrm{d}X}{\mathrm{d}t} = \mu(t, X_t) + \text{"}noise\text{"}$,
- where $\mathbf{E}[noise] = 0$ *regardless of* $\sigma(\cdot)$. Almost true...

These choices conditions have far-reaching consequences!

Interpretation of μ and σ

We are interested in properties when modelling using SDEs

- $\mu = \lim_{h\to 0} \frac{1}{h}\mathbf{E}[X_{t+h} - X_t | X_t]$.
- $\sigma\sigma^T = \lim_{h\to 0} \frac{1}{h} Var[X_{t+h} - X_t | X_t]$.
- Thus $\frac{\mathrm{d}X}{\mathrm{d}t} = \mu(t, X_t) + \text{"}noise\text{"}$,
- where $\mathbf{E}[noise] = 0$ *regardless of* $\sigma(\cdot)$. Almost true...

These choices conditions have far-reaching consequences!

Interpretation of μ and σ

We are interested in properties when modelling using SDEs

- $\mu = \lim_{h\to 0} \frac{1}{h}\mathbf{E}[X_{t+h} - X_t | X_t]$.
- $\sigma\sigma^T = \lim_{h\to 0} \frac{1}{h} Var[X_{t+h} - X_t | X_t]$.
- Thus $\frac{\mathrm{d}X}{\mathrm{d}t} = \mu(t, X_t) + \text{"}noise\text{"}$,
- where $\mathbf{E}[noise] = 0$ *regardless of* $\sigma(\cdot)$. Almost true...

These choices conditions have far-reaching consequences!

Problems (P) and solutions (S)

- P: The time derivative of a BM cannot be defined, $\nexists v_t$
- S: Thus, we define stoch. integral equations instead

$$\int \mathrm{d}X_u = \int \mu(u, X_u)\mathrm{d}u + \int \sigma(u, X_u)\mathrm{d}W_u \qquad (2)$$

- P: The Brownian Motion has unbounded variation - the Riemann integral does not converge.
- S: But the BM has bounded quadratic variation. Thus, define the stoch. integral as limit in $L^2(\mathrm{d}\mathbb{P} \times \mathrm{d}t)$.

Problems (P) and solutions (S)

- P: The time derivative of a BM cannot be defined, $\nexists v_t$
- S: Thus, we define stoch. integral equations instead

$$\int \mathrm{d}X_u = \int \mu(u, X_u)\mathrm{d}u + \int \sigma(u, X_u)\mathrm{d}W_u \qquad (2)$$

- P: The Brownian Motion has unbounded variation - the Riemann integral does not converge.
- S: But the BM has bounded quadratic variation. Thus, define the stoch. integral as limit in $L^2(\mathrm{d}\mathbb{P} \times \mathrm{d}t)$.

Problems (P) and solutions (S)

- P: The time derivative of a BM cannot be defined, $\nexists v_t$
- S: Thus, we define stoch. integral equations instead

$$\int \mathrm{d}X_u = \int \mu(u, X_u)\mathrm{d}u + \int \sigma(u, X_u)\mathrm{d}W_u \qquad (2)$$

- P: The Brownian Motion has unbounded variation - the Riemann integral does not converge.
- S: But the BM has bounded quadratic variation. Thus, define the stoch. integral as limit in $L^2(\mathrm{d}\mathbb{P} \times \mathrm{d}t)$.

Length of Brownian motion

Heuristic argument - consider a partition of time $t = u_0 < u_1 < \ldots < u_N = T$, with $\delta = T/N$.

- We now have N elements, each of length $\sqrt{\delta^2 + (\delta W)^2} = \ldots = \mathcal{O}_p(\sqrt{\delta})$ as $\delta \to 0$
- Total length is $N\mathcal{O}_p(\sqrt{\delta}) = T\mathcal{O}_p(\sqrt{N})$

Length of Brownian motion

Heuristic argument - consider a partition of time $t = u_0 < u_1 < \ldots < u_N = T$, with $\delta = T/N$.

- We now have N elements, each of length $\sqrt{\delta^2 + (\delta W)^2} = \ldots = \mathcal{O}_p(\sqrt{\delta})$ as $\delta \to 0$
- Total length is $N\mathcal{O}_p(\sqrt{\delta}) = T\mathcal{O}_p(\sqrt{N})$

Length of Brownian motion

Heuristic argument - consider a partition of time $t = u_0 < u_1 < \ldots < u_N = T$, with $\delta = T/N$.

- We now have N elements, each of length $\sqrt{\delta^2 + (\delta W)^2} = \ldots = \mathcal{O}_p(\sqrt{\delta})$ as $\delta \to 0$
- Total length is $N\mathcal{O}_p(\sqrt{\delta}) = T\mathcal{O}_p(\sqrt{N})$

Integrals

Define a partition in time $t = u_0 < u_1 < \ldots < u_N = T$ and let $N \to \infty$

- $\int \mu(u, X_u)\mathrm{d}u$ is a standard **Riemann integral**.
- $\int \mu(u, X_u)\mathrm{d}u = \lim_{N\to\infty} \sum \mu(\tau, X_\tau)(u_{n+1} - u_n)$, where $u_n \leq \tau \leq u_{n+1}$.

- $\int \sigma(u, X_u)\mathrm{d}W_u$ is a **Itô integral**.
- $\int \sigma(u, X_u)\mathrm{d}W_u = \lim_{N\to\infty} \sum \sigma(u_n, X_{u_n})(W_{u_{n+1}} - W_{u_n})$.

Integrals

Define a partition in time $t = u_0 < u_1 < \ldots < u_N = T$ and let $N \to \infty$

- $\int \mu(u, X_u)\mathrm{d}u$ is a standard **Riemann integral**.
- $\int \mu(u, X_u)\mathrm{d}u = \lim_{N\to\infty} \sum \mu(\tau, X_\tau)(u_{n+1} - u_n)$, where $u_n \leq \tau \leq u_{n+1}$.

- $\int \sigma(u, X_u)\mathrm{d}W_u$ is a **Itô integral**.
- $\int \sigma(u, X_u)\mathrm{d}W_u = \lim_{N\to\infty} \sum \sigma(u_n, X_{u_n})(W_{u_{n+1}} - W_{u_n})$.

Integrals

Define a partition in time $t = u_0 < u_1 < \ldots < u_N = T$ and let $N \to \infty$

- $\int \mu(u, X_u)\mathrm{d}u$ is a standard **Riemann integral**.
- $\int \mu(u, X_u)\mathrm{d}u = \lim_{N\to\infty} \sum \mu(\tau, X_\tau)(u_{n+1} - u_n)$, where $u_n \leq \tau \leq u_{n+1}$.

- $\int \sigma(u, X_u)\mathrm{d}W_u$ is a **Itô integral**.
- $\int \sigma(u, X_u)\mathrm{d}W_u = \lim_{N\to\infty} \sum \sigma(u_n, X_{u_n})(W_{u_{n+1}} - W_{u_n})$.

Integrals

Define a partition in time $t = u_0 < u_1 < \ldots < u_N = T$ and let $N \to \infty$

- $\int \mu(u, X_u)\mathrm{d}u$ is a standard **Riemann integral**.
- $\int \mu(u, X_u)\mathrm{d}u = \lim_{N\to\infty} \sum \mu(\tau, X_\tau)(u_{n+1} - u_n)$, where $u_n \leq \tau \leq u_{n+1}$.

- $\int \sigma(u, X_u)\mathrm{d}W_u$ is a **Itô integral**.
- $\int \sigma(u, X_u)\mathrm{d}W_u = \lim_{N\to\infty} \sum \sigma(u_n, X_{u_n})(W_{u_{n+1}} - W_{u_n})$.

Properties of stochastic integrals

The **filtration** $\mathcal{F}_t$ is the sigma algebra "information" generated by the Brownian motion $W_u,\ u \leq t$.

- Martingale property of stochastic integrals
  $\mathbf{E}\left[\int_0^T \sigma(W_u)\mathrm{d}W_u|\mathcal{F}_t\right] = \int_0^t \sigma(W_u)\mathrm{d}W_u + 0.$
- Ito isometry $\mathbf{E}\left[\left(\int_0^T \sigma(W_u)\mathrm{d}W_u\right)^2\right] = \int_0^T \mathbf{E}\left[\sigma^2(W_u)\right]\mathrm{d}u$

Proof of Martingale property

Exercise

- **To do:** Calculate $\int W_u \mathrm{d}W_u$.
- **To do:** Calculate $\mathbf{E}[\int_t^T W_u \mathrm{d}W_u | \mathcal{F}_t]$.

Finding analytical solutions to SDEs

3 well known cases

- ABM $\mathrm{d}X_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t$ Solution: $X_t = X_0 + \mu t + \sigma W_t$
- GBM $\mathrm{d}X_t = \mu X_t \mathrm{d}t + \sigma X_t \mathrm{d}W_t$ Solution: $X_t = X_0 \exp\left((\mu - \frac{\sigma^2}{2})t + \sigma W_t\right)$
- Ornstein-Uhlenbeck $dX_t = \kappa(\theta - X_t)\,\mathrm{d}t + \sigma dW_t$ Gaussian process, AR(1)

Exercise: Solve the ABM using the definitions.

Finding analytical solutions to SDEs

3 well known cases

- ABM $\mathrm{d}X_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t$ Solution: $X_t = X_0 + \mu t + \sigma W_t$
- GBM $\mathrm{d}X_t = \mu X_t \mathrm{d}t + \sigma X_t \mathrm{d}W_t$ Solution: $X_t = X_0 \exp\left((\mu - \frac{\sigma^2}{2})t + \sigma W_t\right)$
- Ornstein-Uhlenbeck $dX_t = \kappa(\theta - X_t)\,\mathrm{d}t + \sigma dW_t$ Gaussian process, AR(1)

Exercise: Solve the ABM using the definitions.

Finding analytical solutions to SDEs

3 well known cases

- ABM $\mathrm{d}X_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t$ Solution: $X_t = X_0 + \mu t + \sigma W_t$
- GBM $\mathrm{d}X_t = \mu X_t \mathrm{d}t + \sigma X_t \mathrm{d}W_t$ Solution: $X_t = X_0 \exp\left((\mu - \frac{\sigma^2}{2})t + \sigma W_t\right)$
- Ornstein-Uhlenbeck $dX_t = \kappa(\theta - X_t)\,\mathrm{d}t + \sigma dW_t$ Gaussian process, AR(1)

Exercise: Solve the ABM using the definitions.

Finding analytical solutions to SDEs

3 well known cases

- ABM $\mathrm{d}X_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t$ Solution: $X_t = X_0 + \mu t + \sigma W_t$
- GBM $\mathrm{d}X_t = \mu X_t \mathrm{d}t + \sigma X_t \mathrm{d}W_t$ Solution: $X_t = X_0 \exp\left((\mu - \frac{\sigma^2}{2})t + \sigma W_t\right)$
- Ornstein-Uhlenbeck $dX_t = \kappa(\theta - X_t)\,\mathrm{d}t + \sigma dW_t$ Gaussian process, AR(1)

Exercise: Solve the ABM using the definitions.

Finding analytical solutions to SDEs

3 well known cases

- ABM $\mathrm{d}X_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t$ Solution: $X_t = X_0 + \mu t + \sigma W_t$
- GBM $\mathrm{d}X_t = \mu X_t \mathrm{d}t + \sigma X_t \mathrm{d}W_t$ Solution: $X_t = X_0 \exp\left((\mu - \frac{\sigma^2}{2})t + \sigma W_t\right)$
- Ornstein-Uhlenbeck $dX_t = \kappa\left(\theta - X_t\right)\mathrm{d}t + \sigma dW_t$ Gaussian process, AR(1)

Exercise: Solve the ABM using the definitions.

Finding analytical solutions to SDEs

3 well known cases

- ABM $\mathrm{d}X_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t$ Solution: $X_t = X_0 + \mu t + \sigma W_t$
- GBM $\mathrm{d}X_t = \mu X_t \mathrm{d}t + \sigma X_t \mathrm{d}W_t$ Solution: $X_t = X_0 \exp\left((\mu - \frac{\sigma^2}{2})t + \sigma W_t\right)$
- Ornstein-Uhlenbeck $dX_t = \kappa\left(\theta - X_t\right)\mathrm{d}t + \sigma dW_t$ Gaussian process, AR(1)

Exercise: Solve the ABM using the definitions.

Finding analytical solutions to SDEs

3 well known cases

- ABM $\mathrm{d}X_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t$ Solution: $X_t = X_0 + \mu t + \sigma W_t$
- GBM $\mathrm{d}X_t = \mu X_t \mathrm{d}t + \sigma X_t \mathrm{d}W_t$ Solution: $X_t = X_0 \exp\left((\mu - \frac{\sigma^2}{2})t + \sigma W_t\right)$
- Ornstein-Uhlenbeck $dX_t = \kappa\left(\theta - X_t\right)\mathrm{d}t + \sigma dW_t$ Gaussian process, AR(1)

Exercise: Solve the ABM using the definitions.

Chain rule for SDEs

- Analytical solutions to ODEs are solved by transforming, and thereafter integrating the system.
- The same holds for SDEs.
- Let $\mathrm{d}X_t = \mu(t, X_t)\mathrm{d}t + \sigma(t, X_t)\mathrm{d}W_t$
- Introduce $Y_t = F(t, X_t)$. What about $\mathrm{d}Y_t$?
- It can be shown that for $F \in C^{1,2}$

$$\mathrm{d}Y_t = \left(F_t + \mu F_x + \frac{1}{2}\sigma^2 F_{xx}\right)\mathrm{d}t + \sigma F_x \mathrm{d}W_t \tag{3}$$

$$= (F_t + \mathcal{A}F)\,\mathrm{d}t + \sigma F_x \mathrm{d}W_t. \tag{4}$$

Look at handouts for more theoretical background.

Chain rule for SDEs

- Analytical solutions to ODEs are solved by transforming, and thereafter integrating the system.
- The same holds for SDEs.
- Let $\mathrm{d}X_t = \mu(t, X_t)\mathrm{d}t + \sigma(t, X_t)\mathrm{d}W_t$
- Introduce $Y_t = F(t, X_t)$. What about $\mathrm{d}Y_t$?
- It can be shown that for $F \in C^{1,2}$

$$\mathrm{d}Y_t = \left(F_t + \mu F_x + \frac{1}{2}\sigma^2 F_{xx}\right)\mathrm{d}t + \sigma F_x \mathrm{d}W_t \tag{3}$$

$$= (F_t + \mathcal{A}F)\,\mathrm{d}t + \sigma F_x \mathrm{d}W_t. \tag{4}$$

Look at handouts for more theoretical background.

Chain rule for SDEs

- Analytical solutions to ODEs are solved by transforming, and thereafter integrating the system.
- The same holds for SDEs.
- Let $\mathrm{d}X_t = \mu(t, X_t)\mathrm{d}t + \sigma(t, X_t)\mathrm{d}W_t$
- Introduce $Y_t = F(t, X_t)$. What about $\mathrm{d}Y_t$?
- It can be shown that for $F \in C^{1,2}$

$$\mathrm{d}Y_t = \left(F_t + \mu F_x + \frac{1}{2}\sigma^2 F_{xx}\right)\mathrm{d}t + \sigma F_x \mathrm{d}W_t \tag{3}$$

$$= (F_t + \mathcal{A}F)\,\mathrm{d}t + \sigma F_x \mathrm{d}W_t. \tag{4}$$

Look at handouts for more theoretical background.

Monte Carlo methods

We can easily simulate from an SDE.

$$\mathrm{d}X_t = \mu(t, X_t)\mathrm{d}t + \sigma(t, X_t)\mathrm{d}W_t \tag{5}$$

Convergence is defined as

- **Weak convergence** of $\mathcal{O}(h^\beta)$ is defined for $0 < h < h_0$ for some small h_0

$$\left|\mathbf{E}[g(X_T)] - \mathbf{E}[g(X_T^h)]\right| \leq Dh^\beta \tag{6}$$

- **Strong convergence** of $\mathcal{O}(h^\alpha)$ is defined as

$$\mathbf{E}\left[\sup\left|X_T - X_T^h\right|\right] \leq Ch^\alpha \tag{7}$$

Popular schemes

- The *Euler-Maruyama* scheme is given by

$$X_{t+h} = X_t + \mu(t, X_t)h + \sigma(t, X_t)\delta W_t^h \tag{8}$$

  where $\delta W_t^h = W_{t+h} - W_t$.
- Weak convergence $\mathcal{O}(h)$, strong convergence $\mathcal{O}(\sqrt{h})$
- The *Milstein* scheme is given by

$$\begin{aligned} X_{t+h} = X_t + \mu(t, X_t)h + \sigma(t, X_t)\delta W_t^h + \\ \frac{1}{2}\sigma(t, X_t)\sigma'(t, X_t)\left((\delta W_t^h)^2 - h\right) \end{aligned} \tag{9}$$

- Weak convergence $\mathcal{O}(h)$, strong convergence $\mathcal{O}(h)$. Note - difficult to generalize to higher dimensions!

Fokker-Planck - Likelihood estimation

The transition density $p(x_t|x_s)$ is derived from the Fokker-Planck equation

$$\frac{\partial}{\partial t}p(x_t|x_s) = \mathcal{A}^* p(x_t|x_s) \tag{10}$$

where $\mathcal{A}^*$ is the Hermitian adjoint operator for the Ito generator $\mathcal{A}$

$$\mathcal{A}^* p = -\left(\frac{\partial}{\partial x}\mu(\cdot)p\right) + \frac{1}{2}\frac{\partial^2}{\partial x^2}\left(\sigma^2(\cdot)p\right) \tag{11}$$

Thus, the transition density is found by solving this PDE with initial condition $p(x_s) = \delta(x - x_s)$.

Fokker-Planck - Likelihood estimation

The transition density $p(x_t|x_s)$ is derived from the Fokker-Planck equation

$$\frac{\partial}{\partial t}p(x_t|x_s) = \mathcal{A}^* p(x_t|x_s) \tag{10}$$

where $\mathcal{A}^*$ is the Hermitian adjoint operator for the Ito generator $\mathcal{A}$

$$\mathcal{A}^* p = -\left(\frac{\partial}{\partial x}\mu(\cdot)p\right) + \frac{1}{2}\frac{\partial^2}{\partial x^2}\left(\sigma^2(\cdot)p\right) \tag{11}$$

Thus, the transition density is found by solving this PDE with initial condition $p(x_s) = \delta(x - x_s)$.

Monte Carlo - Likelihood estimation

A simple method is to use kernel density estimation A more refined version is the *Pedersen estimator*

$$p(x_t|x_0) = \int p(x_t, x_s|x_0)\mathrm{d}x_s, \ 0 < s < t \tag{12}$$

$$= \int p(x_t|x_s)p(x_s|x_0)\mathrm{d}x_s \tag{13}$$

$$= \mathbf{E}\left[p(x_t|x_s)|\mathcal{F}_0\right] \tag{14}$$

This is easy to compute using Monte Carlo methods and is not biased, contrary to kernel density estimation.

Monte Carlo - Likelihood estimation

A simple method is to use kernel density estimation A more refined version is the *Pedersen estimator*

$$p(x_t|x_0) = \int p(x_t, x_s|x_0)\mathrm{d}x_s, \ 0 < s < t \tag{12}$$

$$= \int p(x_t|x_s)p(x_s|x_0)\mathrm{d}x_s \tag{13}$$

$$= \mathbf{E}\left[p(x_t|x_s)|\mathcal{F}_0\right] \tag{14}$$

This is easy to compute using Monte Carlo methods and is not biased, contrary to kernel density estimation.

Evolution of mobile phones



The phone on the left is the Motorola DanTac 8000X, the first "mobile" phone (weight about 2 kg) and costing about 4 000 USD when launched in 1984.

Dynamical model of phone market

Carry capacity

$$\mathrm{d}C = \mu C \mathrm{d}t + \sigma C \mathrm{d}W_t^C \tag{15}$$

Simulation using four competing brands (subject to carry capacity)
Each brand is modeled using standard growth model, with parameters capturing the growth and adaptivity properties of specific companies.

Dynamical model of phone market

Carry capacity

$$\mathrm{d}C = \mu C \mathrm{d}t + \sigma C \mathrm{d}W_t^C \tag{15}$$

Simulation using four competing brands (subject to carry capacity)

Each brand is modeled using standard growth model, with parameters capturing the growth and adaptivity properties of specific companies.

Dynamical model of phone market

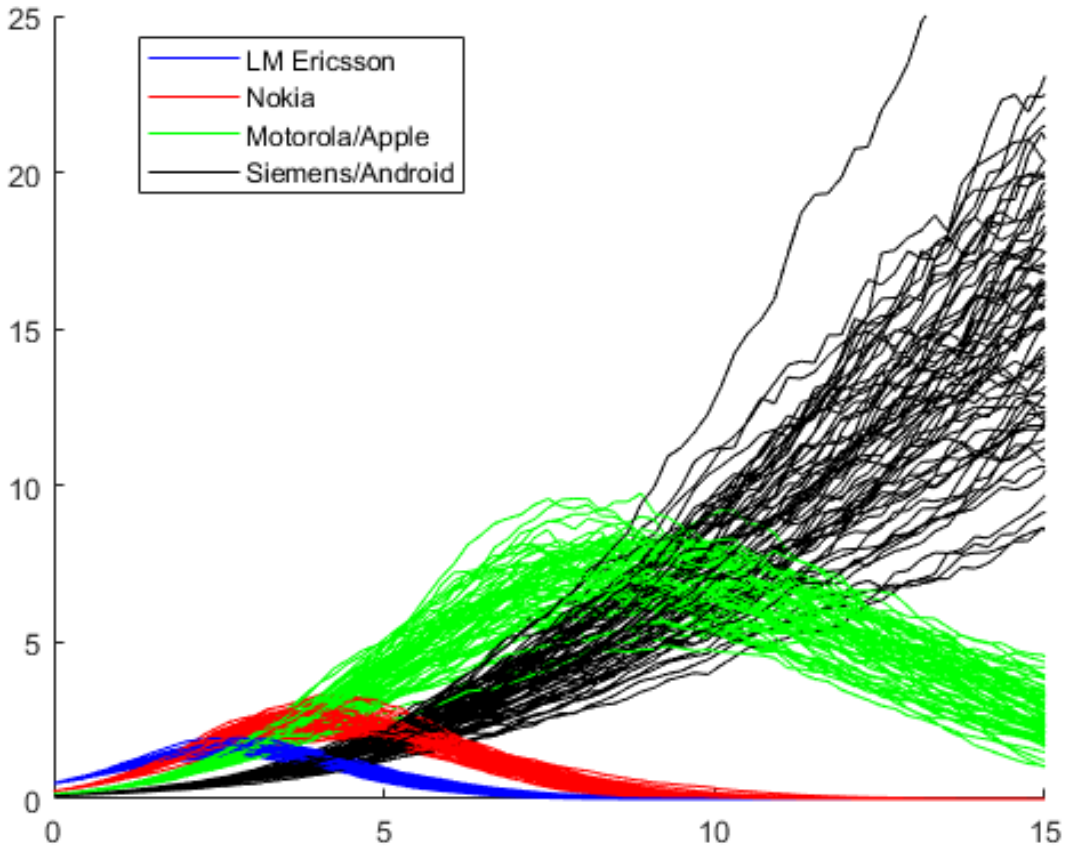
Carry capacity

$$\mathrm{d}C = \mu C \mathrm{d}t + \sigma C \mathrm{d}W_t^C \tag{15}$$

Simulation using four competing brands (subject to carry capacity)
Each brand is modeled using standard growth model, with parameters capturing the growth and adaptivity properties of specific companies.

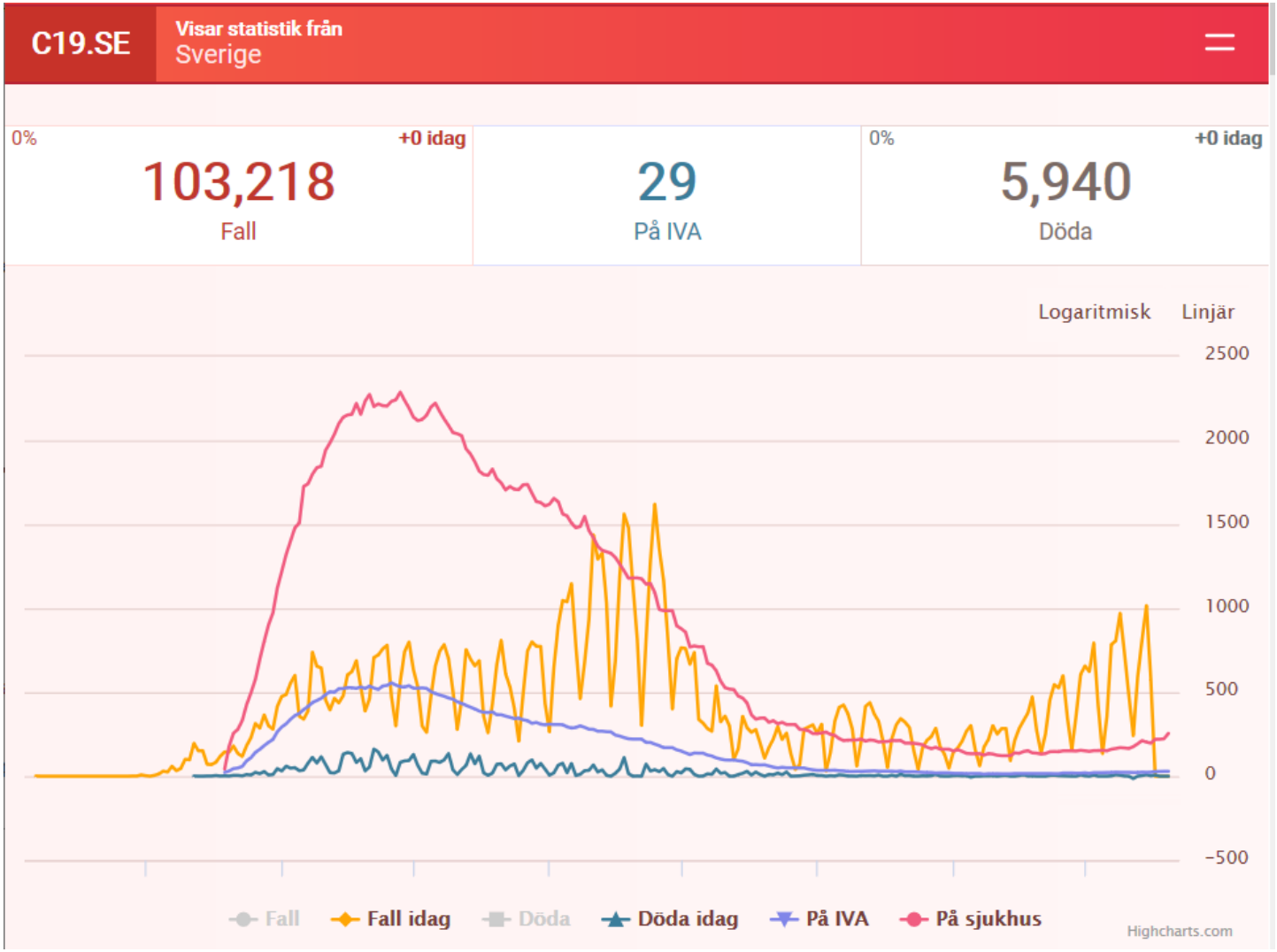
Evolution

Growth over time, 20 different realizations



`Matlab: GenLotkaVolterra.m`

Covid-19 cases in Sweden as of October 19th, 2020

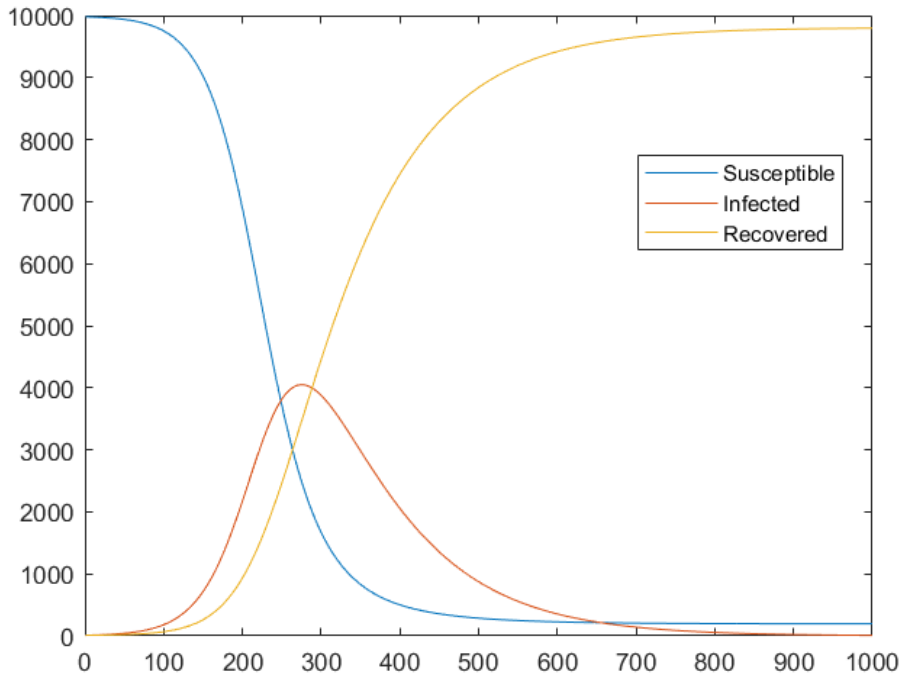


The standard model is the S-I-R models

Three components, **S**usceptibles, **I**nfected and **R**ecoved.

$$\begin{pmatrix} \mathrm{d}S_t \\ \mathrm{d}I_t \\ \mathrm{d}R_t \end{pmatrix} = \begin{pmatrix} -\beta \frac{S_t I_t}{N_t} \\ \beta \frac{S_t I_t}{N_t} - \gamma I_t \\ \gamma I_t \end{pmatrix} \mathrm{d}t \tag{16}$$

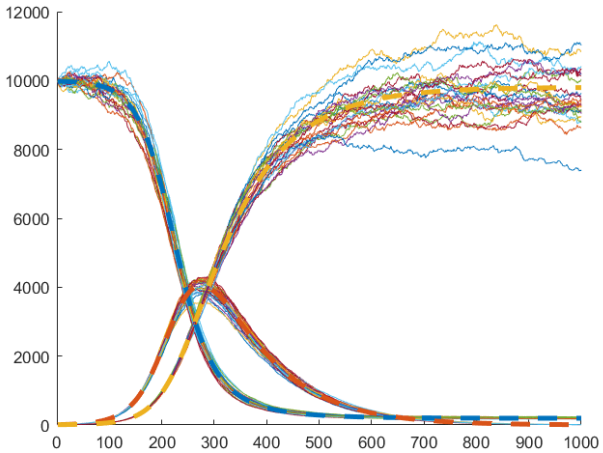
with $\beta = 0.4$ and $\gamma = 0.1$.

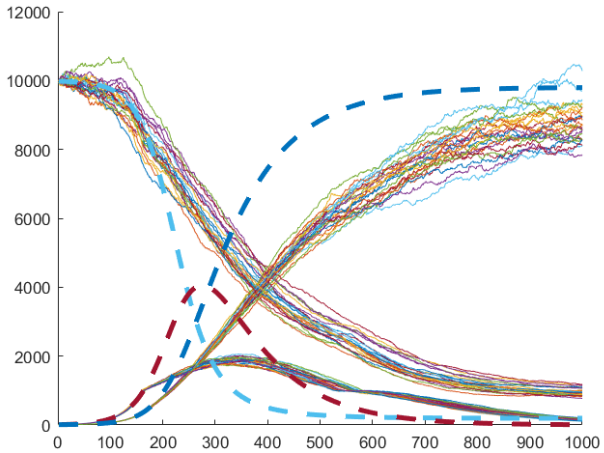


Stochastic version of that model

$$\begin{pmatrix} \mathrm{d}S_t \\ \mathrm{d}I_t \\ \mathrm{d}R_t \end{pmatrix} = \begin{pmatrix} -\beta \frac{S_t I_t}{N_t} \\ \beta \frac{S_t I_t}{N_t} - \gamma I_t \\ \gamma I_t \end{pmatrix} \mathrm{d}t + \sigma \begin{pmatrix} S_t & 0 & 0 \\ 0 & I_t & 0 \\ 0 & 0 & R_t \end{pmatrix} \begin{pmatrix} \mathrm{d}W_t^S \\ \mathrm{d}W_t^I \\ dW_t^R \end{pmatrix} \quad (17)$$

with $\sigma = 0.01$.





Feel free to play with the `Matlab` scripts, including flattening the curve interventions ($\beta = 0.4$ left, $\beta = 0.2$ when $I_t > 1000$ right).

How would you model Covid-19?

Same plot as before, except in log-scale - the number of cases is increasing steadily



Filtering

- Assume that we have a latent (hidden) process X, in discrete or continuous time.
- X is assumed to be Markov, with known initial distribution.
- Observations Y are generated from X
- and perturbated by noise.

$$
\begin{aligned}
x_0 &\sim p(x_0) \\
x_{n+1} &= f(t_n, x_n, e_{n+1}), && \Leftrightarrow p(x_{n+1}|x_n) \text{ known} \\
y_{n+1} &= h(t_{n+1}, x_{n+1}, \eta_{n+1}), && \Leftrightarrow p(y_n|x_n) \text{ known}
\end{aligned}
$$

Dependence structure

Consider the model

$$x_{n+1} = ax_n + e_{n+1} \tag{18}$$

$$y_{n+1} = x_{n+1} + \eta_{n+1} \tag{19}$$

Exercise: Show that the correlation structure for $\{y\}$ is that of an ARMA(1,1) model.

Likelihood function

The log-likelihood is given by

$$\ell(\theta) = \log p_\theta(y_1, \ldots, y_N) \tag{20}$$

$$= \sum_{n=1}^{N} \log p_\theta(y_n | y_1 \ldots, y_{n-1}) = \sum_{n=1}^{N} \log p_\theta(y_n | y_{1:n-1}) \tag{21}$$

This can be expanded as

$$p_\theta(y_n | y_1 \ldots, y_{n-1}) = \int p_\theta(y_n, x_n | y_{1:n-1}) \mathrm{d}x_n \tag{22}$$

$$\int \underbrace{p_\theta(y_n | x_n)}_{\textit{Known}} p_\theta(x_n | y_{1:n-1}) \mathrm{d}x_n \tag{23}$$

Filter equations

Recursive formulas can be derived for *the predictive density* $p_\theta\left(x_n|y_{1:n-1}\right)$ is given by

$$p_\theta\left(x_n|y_{1:n-1}\right) = \int \underbrace{p_\theta\left(x_n|x_{n-1}\right)}_{\textit{Known}} p_\theta\left(x_{n-1}|y_{1:n-1}\right) \mathrm{d}x_{n-1}. \tag{24}$$

and for *the filter density*

$$p_\theta(x_n|y_{1:n}) = \frac{p_\theta(y_n|x_n)p_\theta(x_n|y_{1:n-1})}{\int p_\theta(y_n|x_n)p_\theta(x_n|y_{1:n-1})\mathrm{d}x_n}, \tag{25}$$

Filter algorithms

Exact algorithms

- **HMM filter** "Trivial", all integrals are sums
- **Kalman filter** Linear, Gaussian transformations.

Approximate filters

- (Iterated) Extended Kalman filter
- Unscented Kalman filter
- Gaussian sum filters
- Ensemble Kalman filter
- Seq. Monte Carlo filters

Filter algorithms

Exact algorithms

- **HMM filter** "Trivial", all integrals are sums
- **Kalman filter** Linear, Gaussian transformations.

Approximate filters

- (Iterated) Extended Kalman filter
- Unscented Kalman filter
- Gaussian sum filters
- Ensemble Kalman filter
- Seq. Monte Carlo filters

Filter algorithms

Exact algorithms

- **HMM filter** "Trivial", all integrals are sums
- **Kalman filter** Linear, Gaussian transformations.

Approximate filters

- (Iterated) Extended Kalman filter
- Unscented Kalman filter
- Gaussian sum filters
- Ensemble Kalman filter
- Seq. Monte Carlo filters

Filter algorithms

Exact algorithms

- **HMM filter** "Trivial", all integrals are sums
- **Kalman filter** Linear, Gaussian transformations.

Approximate filters

- (Iterated) Extended Kalman filter
- Unscented Kalman filter
- Gaussian sum filters
- Ensemble Kalman filter
- Seq. Monte Carlo filters

Filter algorithms

Exact algorithms

- **HMM filter** "Trivial", all integrals are sums
- **Kalman filter** Linear, Gaussian transformations.

Approximate filters

- (Iterated) Extended Kalman filter
- Unscented Kalman filter
- Gaussian sum filters
- Ensemble Kalman filter
- Seq. Monte Carlo filters

Kalman filter

Main reference for derivations of this type: The Matrix Cookbook[1]

$$x_{n+1} = Ax_n + e_n, \;\; e_n \sim N(0, Q) \tag{26}$$
$$y_n = Cx_n + \epsilon_n, \;\; \epsilon \sim N(0, R) \tag{27}$$

Initial conditions

- $p(x_0) = \phi(x_0; m_0, P_0)$

Predictive density, $p(x_n|y_{1:n-1}) = \phi(x_n; m_{n|n-1}, P_{n|n-1})$

- $m_{n|n-1} = Am_{n-1|n-1}$
- $P_{n|n-1} = AP_{n-1|n-1}A^T + Q$

[1] https://www2.imm.dtu.dk/pubdb/pubs/3274-full.html

Kalman filter

Main reference for derivations of this type: The Matrix Cookbook[1]

$$x_{n+1} = Ax_n + e_n, \;\; e_n \sim N(0, Q) \tag{26}$$
$$y_n = Cx_n + \epsilon_n, \;\; \epsilon \sim N(0, R) \tag{27}$$

Initial conditions

- $p(x_0) = \phi(x_0; m_0, P_0)$

Predictive density, $p(x_n|y_{1:n-1}) = \phi(x_n; m_{n|n-1}, P_{n|n-1})$

- $m_{n|n-1} = Am_{n-1|n-1}$
- $P_{n|n-1} = AP_{n-1|n-1}A^T + Q$

[1] https://www2.imm.dtu.dk/pubdb/pubs/3274-full.html

Kalman filter

Main reference for derivations of this type: The Matrix Cookbook[1]

$$x_{n+1} = Ax_n + e_n, \;\; e_n \sim N(0, Q) \tag{26}$$
$$y_n = Cx_n + \epsilon_n, \;\; \epsilon \sim N(0, R) \tag{27}$$

Initial conditions

- $p(x_0) = \phi(x_0; m_0, P_0)$

Predictive density, $p(x_n|y_{1:n-1}) = \phi(x_n; m_{n|n-1}, P_{n|n-1})$

- $m_{n|n-1} = Am_{n-1|n-1}$
- $P_{n|n-1} = AP_{n-1|n-1}A^T + Q$

[1] https://www2.imm.dtu.dk/pubdb/pubs/3274-full.html

Kalman filter

Main reference for derivations of this type: The Matrix Cookbook[1]

$$x_{n+1} = Ax_n + e_n, \;\; e_n \sim N(0, Q) \tag{26}$$
$$y_n = Cx_n + \epsilon_n, \;\; \epsilon \sim N(0, R) \tag{27}$$

Initial conditions

- $p(x_0) = \phi(x_0; m_0, P_0)$

Predictive density, $p(x_n|y_{1:n-1}) = \phi(x_n; m_{n|n-1}, P_{n|n-1})$

- $m_{n|n-1} = Am_{n-1|n-1}$
- $P_{n|n-1} = AP_{n-1|n-1}A^T + Q$

[1] https://www2.imm.dtu.dk/pubdb/pubs/3274-full.html

Kalman filter

Likelihood term $p(y_n|y_{1:n-1}) = \phi(y_n; \hat{y}_{n|n-1}, \Omega_n)$

- $\hat{y}_{n|n-1} = Cm_{n|n-1}$
- $\Omega_n = CP_{n|n-1}C^T + R$

Filter density

- $K_n = P_{n|n-1}C^T\Omega_n^{-1}$
- $m_{n|n} = m_{n|n-1} + K_n\left(y_n - \hat{y}_{n|n-1}\right)$
- $P_{n|n} = (I - KC)\,P_{n|n-1}$

Kalman filter

Likelihood term $p(y_n|y_{1:n-1}) = \phi(y_n; \hat{y}_{n|n-1}, \Omega_n)$

- $\hat{y}_{n|n-1} = Cm_{n|n-1}$
- $\Omega_n = CP_{n|n-1}C^T + R$

Filter density

- $K_n = P_{n|n-1}C^T\Omega_n^{-1}$
- $m_{n|n} = m_{n|n-1} + K_n\left(y_n - \hat{y}_{n|n-1}\right)$
- $P_{n|n} = (I - KC)\,P_{n|n-1}$

Approx. Gaussian Filters

Treat all distributions as Gaussians, $p(x|\cdot) = \phi(x, \cdot, \cdot)$

- Approximate Kalman filters (think QML)
- Easy to implement
- Nearly optimal (projections in Hilbert space)
- Historical reasons
- EKF is the simplest option (?), Iterated EKF is often more accurate with UKF often being even better. *Thus, you should know the EKF, be familiar with the IEKF and use the UKF*

`Matlab code: UKFERK1d.m`

Approx. Gaussian Filters

Treat all distributions as Gaussians, $p(x|\cdot) = \phi(x, \cdot, \cdot)$

- Approximate Kalman filters (think QML)
- Easy to implement
- Nearly optimal (projections in Hilbert space)
- Historical reasons
- EKF is the simplest option (?), Iterated EKF is often more accurate with UKF often being even better. *Thus, you should know the EKF, be familiar with the IEKF and use the UKF*

`Matlab code: UKFERK1d.m`

Sequential Monte Carlo Filters

Treat all distributions as countable sums of Dirac measures $p_J(x|\cdot) = \sum_j \frac{\omega_j}{\sum \omega_l} \delta(x - x_j)$

- Use when the drift is highly non-linear
- Non-Gaussian innovations
- Restricted state spaces,
- Performance is critical, and computational resources are abundant.
- E.g. Stochastic volatility

Proofs in handouts.

Sequential Monte Carlo Filters

Treat all distributions as countable sums of Dirac measures $p_J(x|\cdot) = \sum_j \frac{\omega_j}{\sum \omega_l} \delta(x - x_j)$

- Use when the drift is highly non-linear
- Non-Gaussian innovations
- Restricted state spaces,
- Performance is critical, and computational resources are abundant.
- E.g. Stochastic volatility

Proofs in handouts.

- Computing moments

$$\mathbf{E}\left[\phi(x_n)|y_{1:n}\right] = \int \phi(x_n)p(x_n|y_{1:n})\mathrm{d}x_n \tag{28}$$

$$\approx \int \phi(x_n)p_J(x_n|y_{1:n})\mathrm{d}x_n \tag{29}$$

$$= \sum_j \frac{\omega_j}{\sum \omega_l}\phi(x_{n,j}) \tag{30}$$

- Moreover, prediction is easy as

$$p_J(x_{n+1}|y_{1:n}) = \sum_j \frac{\omega_j}{\sum \omega_l}\delta(x - x_{n+1,j}) \tag{31}$$

with simulated as

$$x_{n+1,j} \sim p(x_{n+1}|x_{n,j}) \tag{32}$$

- Computing moments

$$\mathbf{E}\left[\phi(x_n)|y_{1:n}\right] = \int \phi(x_n)p(x_n|y_{1:n})\mathrm{d}x_n \tag{28}$$

$$\approx \int \phi(x_n)p_J(x_n|y_{1:n})\mathrm{d}x_n \tag{29}$$

$$= \sum_j \frac{\omega_j}{\sum \omega_l}\phi(x_{n,j}) \tag{30}$$

- Moreover, prediction is easy as

$$p_J(x_{n+1}|y_{1:n}) = \sum_j \frac{\omega_j}{\sum \omega_l}\delta(x - x_{n+1,j}) \tag{31}$$

with simulated as

$$x_{n+1,j} \sim p(x_{n+1}|x_{n,j}) \tag{32}$$

Bootstrap filter

Initial conditions

- Sample $x_{0,j} \sim p(x_0)$

Prediction

- Sample $x_{n,j} \sim p(x_n|x_{n-1,j})$

Bootstrap filter

Initial conditions

- Sample $x_{0,j} \sim p(x_0)$

Prediction

- Sample $x_{n,j} \sim p(x_n|x_{n-1,j})$

Bootstrap filter

Likelihood computation

- Assume $p_J(x_{n+1}|y_{1:n}) = \sum_j \frac{\omega_j}{\sum \omega_l} \delta(x - x_{n+1,j})$
- Then

$$p(y_{n+1}|y_{1:n}) = \int p(y_{n+1}|x_{n+1}) p_J(x_{n+1}|y_{1:n}) \mathrm{d}x_n \quad (33)$$

$$= \sum_j \frac{\omega_j}{\sum \omega_l} p(y_{n+1}|x_{n+1,j}) \quad (34)$$

Empirical Filter density

-

$$p_J(x_{n+1}|y_{1:n+1}) = \frac{p(y_{n+1}|x_{n+1}) p_J(x_{n+1}|y_{1:n})}{p(y_{n+1}|y_{1:n})} \quad (35)$$

$$\propto \sum_j \frac{\omega_j}{\sum \omega_l} p(y_{n+1}|x_{n+1,j}) \delta(x - x_{n+1,j}) \quad (36)$$

- **Important:** Resample the weights.

Bootstrap filter

Likelihood computation

- Assume $p_J(x_{n+1}|y_{1:n}) = \sum_j \frac{\omega_j}{\sum \omega_l} \delta(x - x_{n+1,j})$
- Then

$$p(y_{n+1}|y_{1:n}) = \int p(y_{n+1}|x_{n+1}) p_J(x_{n+1}|y_{1:n}) \mathrm{d}x_n \tag{33}$$

$$= \sum_j \frac{\omega_j}{\sum \omega_l} p(y_{n+1}|x_{n+1,j}) \tag{34}$$

Empirical Filter density

-

$$p_J(x_{n+1}|y_{1:n+1}) = \frac{p(y_{n+1}|x_{n+1}) p_J(x_{n+1}|y_{1:n})}{p(y_{n+1}|y_{1:n})} \tag{35}$$

$$\propto \sum_j \frac{\omega_j}{\sum \omega_l} p(y_{n+1}|x_{n+1,j}) \delta(x - x_{n+1,j}) \tag{36}$$

- **Important:** Resample the weights.

Parameter estimation

Still an open question

- Direct maximization is difficult for a variety of reasons
- EM
- Iterated filtering
- PMMH

Boat navigation

One possible project on this course was navigation in a lake, based on only a sonar and a chart. This film is by a former student on this course, Jonas Hallgren

`https://www.youtube.com/watch?v=bNDcXDilSHA`

Information: Red star = origin, Red triangle = true pos. of the boat, Blue star = particle, Green star-ring = current guess of location.

Feedback

Send feedback, questions and information about typos to `erik.lindstrom@matstat.lu.se`.