Advanced Machine Learning
State-space models
Lecture 3

Mattias Villani

Division of Statistics and Machine Learning
Department of Computer and Information Science
Linköping University

LECTURE OVERVIEW

- **Non-linear state space models**
- **The extended Kalman filter**
- **Non-Gaussian models**
- **Particle filters**

NONLINEAR STATE SPACE MODELS

- The **linear Gaussian state-space (LGSS) model**

$$\text{Measurement eq:} \quad \mathbf{y}_t = \mathbf{C}\mathbf{x}_t + \varepsilon_t \qquad \varepsilon_t \overset{iid}{\sim} N(0, \Omega_\varepsilon)$$

$$\text{State eq:} \quad \mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_t + \nu_t \qquad \nu_t \overset{iid}{\sim} N(0, \Omega_\nu)$$

- The **non-linear Gaussian state-space model**

$$\text{Measurement eq:} \quad \mathbf{y}_t = h(\mathbf{x}_t) + \varepsilon_t \qquad \varepsilon_t \overset{iid}{\sim} N(0, \Omega_\varepsilon)$$

$$\text{State eq:} \quad \mathbf{x}_t = g(\mathbf{x}_{t-1}, \mathbf{u}_t) + \nu_t \qquad \nu_t \overset{iid}{\sim} N(0, \Omega_\nu)$$

  where $h()$ and $g()$ are functions that maps vectors to vectors.
- **Kalman filter** relied on **linearity**. Not applicable here.
- The extended Kalman filter (EKF):
  1. linearize $h()$ and $g()$.
  2. Apply the usual Kalman filter on the linearized model.
- EKF works well when $h()$ and $g()$ are not too nonlinear and when the state uncertainty is not too large.

Linearization of the state equation

- State equation
$$x_t = g(x_{t-1}, u_t) + \nu_t$$
- **Linearization of the state equation** around the point $x_{t-1} = \tilde{x}_{t-1}$ (no need linearize wrt u_t):
$$g(x_{t-1}, u_t) \approx g(\tilde{x}_{t-1}, u_t) + G_t(\tilde{x}_{t-1}, u_t)\,(x_{t-1} - \tilde{x}_{t-1})$$
$$G_t \equiv g'(\tilde{x}_t) \equiv \frac{\partial g(x_{t-1}, u_t)}{\partial x_{t-1}}$$
- But what is a good point $\tilde{x}_{t-1}$?
- Use the value most likely **at the time of linearization**.
- Kalman filter uses the state equation at the **prediction step**. Most likely value for x_{t-1} at that time is μ_{t-1}.
- When $\mathbf{x}_t$ is a vector: $\mathbf{G}_t$ is a matrix of derivatives (with elements $\frac{\partial g_i}{\partial x_j}$) like in multi-dimensional calculus.

Linearization of the measurement equation

- Measurement equation

$$y_t = h(x_t) + \varepsilon_t$$

- **Linearization of the measurement equation** around the point $x_t = \tilde{x}_t$:

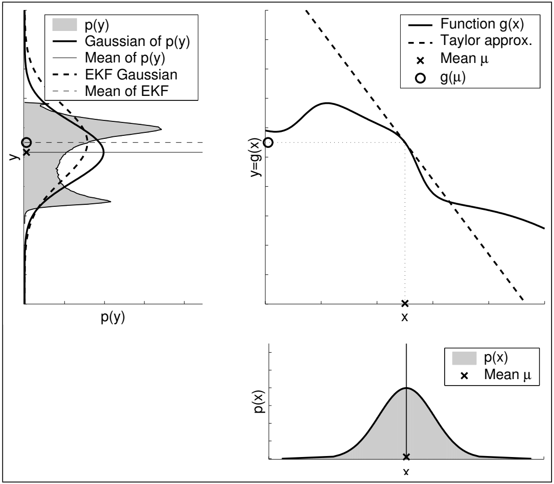
$$h(x_t) \approx h(\tilde{x}_t) + H_t\,(x_t - \tilde{x}_t)$$

  where

$$H_t \equiv h'(\tilde{x}_t) \equiv \frac{\partial h(x_t)}{\partial x_t}$$

- But what is a good point $\tilde{x}_t$?
- Kalman filter uses measurement equation in the **measurement update step**. Most likely value for x_t at that time is $\bar{\mu}_t$.
- When $\mathbf{x}_t$ and $\mathbf{y}_t$ are vectors: $\mathbf{H}_t$ is a matrix of derivatives (with elements $\frac{\partial h_i}{\partial x_j}$) like in multi-dimensional calculus.

Linearization - Thrun et al illustration



The extended Kalman filter

- The **non-linear Gaussian state-space model**

$$\text{Measurement eq:} \quad \mathbf{y}_t = h(\mathbf{x}_t) + \varepsilon_t \qquad \varepsilon_t \overset{iid}{\sim} N(0, \Omega_\varepsilon)$$

$$\text{State eq:} \quad \mathbf{x}_t = g(\mathbf{x}_{t-1}, \mathbf{u}_t) + \nu_t \qquad \nu_t \overset{iid}{\sim} N(0, \Omega_\nu)$$

- **Algorithm ExtendedKalmanFilter**$(\mu_{t-1}, \Sigma_{t-1}, \mathbf{u}_t, \mathbf{y}_t)$
  - Prediction update: $\begin{cases} \bar{\mu}_t = g(\mathbf{x}_{t-1}, \mathbf{u}_t) \\ \bar{\Sigma}_t = \mathbf{G}_t \Sigma_{t-1} \mathbf{G}_t^T + \Omega_\nu \end{cases}$
  - Measurement update : $\begin{cases} \mathbf{K}_t = \bar{\Sigma}_t \mathbf{H}_t^T \left(\mathbf{H}_t \bar{\Sigma}_t \mathbf{H}_t^T + \Omega_\varepsilon \right)^{-1} \\ \mu_t = \bar{\mu}_t + \mathbf{K}_t \left(\mathbf{y}_t - h\left(\bar{\mu}_t\right)\right) \\ \Sigma_t = \left(\mathbf{I} - \mathbf{K}_t \mathbf{H}_t \right) \bar{\Sigma}_t \end{cases}$
  - Return μ_t, Σ_t

Non-Gaussian state space models

- **Linear non-Gaussian state-space model**

$$
\begin{aligned}
\text{Measurement eq:} \quad & \mathbf{y}_t = \mathbf{C}\mathbf{x}_t + \varepsilon_t & \varepsilon_t \overset{iid}{\sim} Non - Gaussian \\
\text{State eq:} \quad & \mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_t + \nu_t & \nu_t \overset{iid}{\sim} Non - Gaussian
\end{aligned}
$$

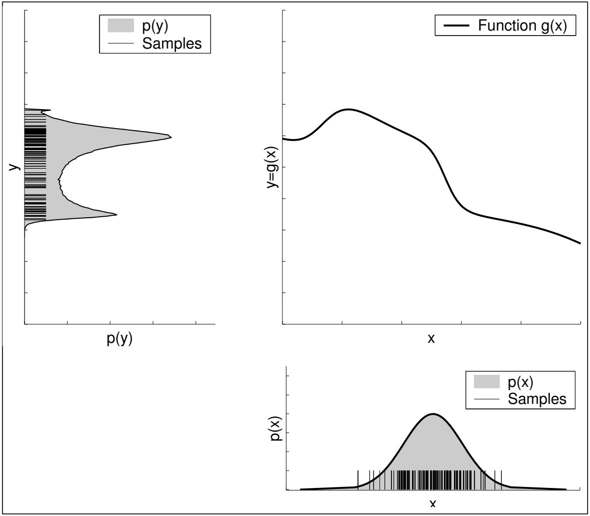
- Student-t errors can be turned (conditionally) Gaussian by data-augmentation.
- The **non-linear non-Gaussian state-space (LGSS) mode**

$$
\begin{aligned}
\text{Measurement eq:} \quad & p(\mathbf{y}_t | \mathbf{x}_t) \\
\text{State eq:} \quad & p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t)
\end{aligned}
$$

- Example: **Poisson with time-varying intensity**

$$
\begin{aligned}
y_t | x_t &\sim \text{Pois}(x_t) \\
x_t | x_{t-1} &\sim N\left(x_{t-1}, \omega_\nu^2\right)
\end{aligned}
$$

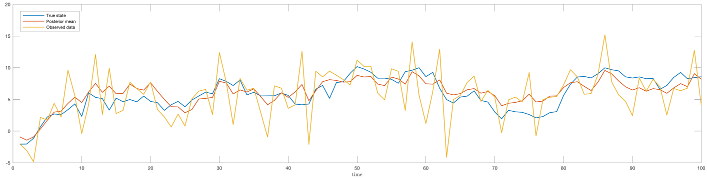
THE PARTICLE FILTER - THRUN ET AL ILLUSTRATION

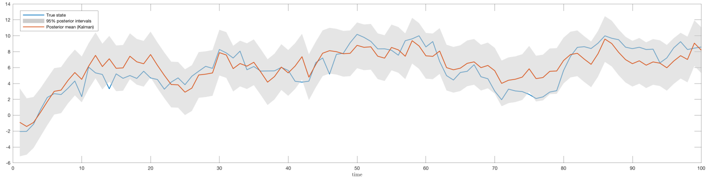


The particle filter

- **Algorithm ParticleFilter**$(\mathcal{X}_{t-1},\mathbf{u}_t,\mathbf{y}_t)$
  - $$\begin{cases} \bar{\mathcal{X}}_t = \mathcal{X}_t = \emptyset \\ \text{for } m = 1 \text{ to } M \text{ do} \\ \quad \text{sample } x_t^{(m)} \sim p\left(x_t | x_{t-1}^{(m)}, u_t\right) \\ \quad w_t^{(m)} = p\left(y_t | x_t^{(m)}\right) \\ \quad \text{append } (x_t^{(m)}, w_t^{(m)}) \text{ to } \bar{\mathcal{X}}_t \\ \text{end} \end{cases}$$ **Prediction/Propagation update**
  - $$\begin{cases} \text{for } m = 1 \text{ to } M \text{ do} \\ \quad \text{draw } i \text{ with probability } \propto w_t^{(i)} \\ \quad \text{append } x_t^{(i)} \text{ to } \bar{\mathcal{X}}_t \\ \text{end} \end{cases}$$ **Measurement/resampling update**
  - Return $\mathcal{X}_t$

Kalman filter - simulated LGSS data





Particle filter (M=1000) - simulated LGSS data