Bayesian Filtering

Václav Šmídl

April 19, 2022

Recapitulation

Joint distribution

$$p(x_1, x_2) = \mathcal{N}\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \right)$$

has marginals:

$$p(x_1) = \mathcal{N}(\mu_1, \Sigma_{11}), \qquad p(x_2) = \mathcal{N}(\mu_2, \Sigma_{22}),$$

and conditional (Bayes rule)

$$p(x_2 | x_1 = a) = \mathcal{N}(\overline{\mu}, \overline{\Sigma}) \qquad \begin{aligned} \overline{\mu} &= \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(a - \mu_1) \\ \overline{\Sigma} &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} \end{aligned}$$

The covariance is known as Schur complement.

Least Squares Revisited (notation)

Model of linear regression with unknown parameters $\boldsymbol{x}$:

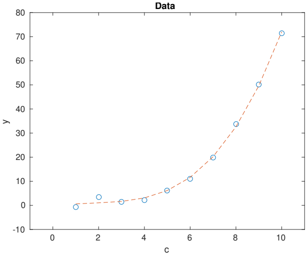
$$y_i = \boldsymbol{c}_i \boldsymbol{x} + e_i,$$

with sum of squares

$$\sigma = \sum_{i=1}^{n} e_i = \boldsymbol{e}^T \boldsymbol{e}.$$

In polynomial regression

$$y = x_1 + x_2 c + x_3 c^2 = \underbrace{[1, c, c^2]}_{\boldsymbol{c}} \underbrace{[x_1, x_2, x_3]^\top}_{\boldsymbol{x}}$$



Least Squares Revisited (notation)

Model of linear regression with unknown parameters $\boldsymbol{x}$:

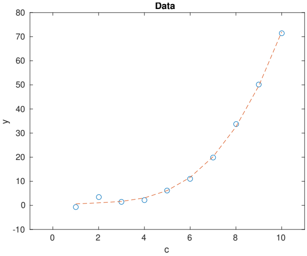
$$y_i = \boldsymbol{c}_i \boldsymbol{x} + e_i,$$

with sum of squares

$$\sigma = \sum_{i=1}^{n} e_i = \boldsymbol{e}^T \boldsymbol{e}.$$

In polynomial regression

$$y = x_1 + x_2 c + x_3 c^2 = \underbrace{[1, c, c^2]}_{\boldsymbol{c}} \underbrace{[x_1, x_2, x_3]^\top}_{\boldsymbol{x}}$$



Least Squares Revisited (notation)

Model of linear regression with unknown parameters $\boldsymbol{x}$:

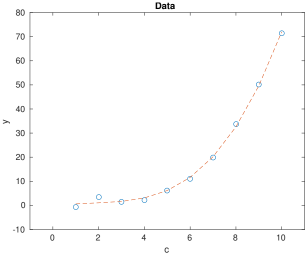
$$y_i = \boldsymbol{c}_i \boldsymbol{x} + e_i,$$

with sum of squares

$$\sigma = \sum_{i=1}^{n} e_i = \boldsymbol{e}^T \boldsymbol{e}.$$

In polynomial regression

$$y = x_1 + x_2 c + x_3 c^2 = \underbrace{[1, c, c^2]}_{\boldsymbol{c}} \underbrace{[x_1, x_2, x_3]^\top}_{\boldsymbol{x}}$$



Solution:

$$\hat{\boldsymbol{x}} = (C^T C)^{-1} C^T \boldsymbol{y}.$$
$$= (\sum_i \boldsymbol{c}_i^\top \boldsymbol{c}_i)^{-1} \sum_i \boldsymbol{c}_i^\top y_i$$

Sufficient statistics

Model:

$$y_i = c_i^T \boldsymbol{x} + e_i,$$

For $i = 1 : n$

$$\hat{\boldsymbol{x}} = \left(\sum_{i=1}^{n} \boldsymbol{c}_i^\top \boldsymbol{c}_i \right)^{-1} \sum_{i=1}^{n} \boldsymbol{c}_i^\top y_i.$$

Sufficient statistics

Model:

$$y_i = c_i^T \boldsymbol{x} + e_i,$$

For $i = 1 : n$

$$\hat{\boldsymbol{x}} = \left(\sum_{i=1}^{n} \boldsymbol{c}_i^\top \boldsymbol{c}_i \right)^{-1} \sum_{i=1}^{n} \boldsymbol{c}_i^\top y_i.$$

Observing $i = n + 1$:

Sufficient statistics

Model:

$$y_i = c_i^T \boldsymbol{x} + e_i,$$

For $i = 1 : n$

$$\hat{\boldsymbol{x}} = \left(\sum_{i=1}^{n} \boldsymbol{c}_i^\top \boldsymbol{c}_i \right)^{-1} \sum_{i=1}^{n} \boldsymbol{c}_i^\top y_i.$$

Observing $i = n + 1$:

$$\hat{\boldsymbol{x}} = \left(\sum_{i=1}^{n} \boldsymbol{c}_i^\top \boldsymbol{c}_i + \boldsymbol{c}_{n+1}^\top \boldsymbol{c}_{n+1} \right)^{-1} \left(\sum_{i=1}^{n} \boldsymbol{c}_i^\top y_i + \boldsymbol{c}_{n+1}^\top y_{n+1} \right).$$

The notion of sufficient statistics:

$$\hat{\boldsymbol{x}} = V^{-1} v,$$

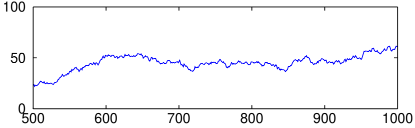
$$V = \sum_{i=1}^{n} \boldsymbol{c}_i^\top \boldsymbol{c}_i,$$

$$v = \sum_{i=1}^{n} \boldsymbol{c}_i^\top y_i,$$

Least Squares for time series

Least squares :

$$V = \sum_{i=1}^{n} \mathbf{c}_i^\top \mathbf{c}_i,$$
$$v = \sum_{i=1}^{n} \mathbf{c}_i^\top y_i,$$



Least Squares for time series

Least squares :

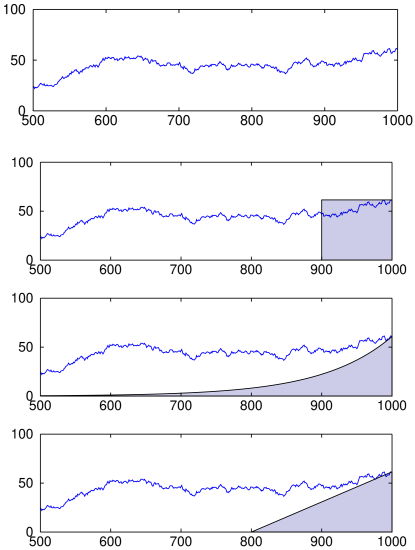
$$V = \sum_{i=1}^{n} \boldsymbol{c}_i^\top \boldsymbol{c}_i,$$
$$v = \sum_{i=1}^{n} \boldsymbol{c}_i^\top y_i,$$

Weighted least squares :

$$V = \sum_{i=1}^{n} w_i \boldsymbol{c}_i^\top \boldsymbol{c}_i,$$
$$v = \sum_{i=1}^{n} w_i \boldsymbol{c}_i^\top y_i,$$

Weight profile - window:

- square window
- exponential window
- linear window



Recursive computation of moving windows of length L

Recursive least squares with scalar x:

$$\sum_{i=1}^{L} \boldsymbol{c}_i^\top \boldsymbol{c}_i = \boldsymbol{c}_1^\top \boldsymbol{c}_1 + \boldsymbol{c}_2^\top \boldsymbol{c}_2 + \ldots + \boldsymbol{c}_L^\top \boldsymbol{c}_L \qquad = \sum_{i=1}^{L-1} \boldsymbol{c}_i^\top \boldsymbol{c}_i + \boldsymbol{c}_L^\top \boldsymbol{c}_L$$

Recursive computation of moving windows of length L

Recursive least squares with scalar x:

$$\sum_{i=1}^{L} \boldsymbol{c}_i^\top \boldsymbol{c}_i = \boldsymbol{c}_1^\top \boldsymbol{c}_1 + \boldsymbol{c}_2^\top \boldsymbol{c}_2 + \ldots + \boldsymbol{c}_L^\top \boldsymbol{c}_L \qquad = \sum_{i=1}^{L-1} \boldsymbol{c}_i^\top \boldsymbol{c}_i + \boldsymbol{c}_L^\top \boldsymbol{c}_L$$

Square window:

$$\sum_{i=n-L}^{n} \boldsymbol{c}_i^\top \boldsymbol{c}_i = \boldsymbol{c}_{n-L}^\top \boldsymbol{c}_{n-L} + \ldots + \boldsymbol{c}_n^\top \boldsymbol{c}_n = \underbrace{\sum_{i=n-L-1}^{n-1} \boldsymbol{c}_i^\top \boldsymbol{c}_i}_{\textit{last results}} + \boldsymbol{c}_n^\top \boldsymbol{c}_n - \boldsymbol{c}_{n-L-1}^\top \boldsymbol{c}_{n-L-1},$$

Recursive computation of moving windows of length L

Recursive least squares with scalar x:

$$\sum_{i=1}^{L} \boldsymbol{c}_i^\top \boldsymbol{c}_i = \boldsymbol{c}_1^\top \boldsymbol{c}_1 + \boldsymbol{c}_2^\top \boldsymbol{c}_2 + \ldots + \boldsymbol{c}_L^\top \boldsymbol{c}_L \qquad = \sum_{i=1}^{L-1} \boldsymbol{c}_i^\top \boldsymbol{c}_i + \boldsymbol{c}_L^\top \boldsymbol{c}_L$$

Square window:

$$\sum_{i=n-L}^{n} \boldsymbol{c}_i^\top \boldsymbol{c}_i = \boldsymbol{c}_{n-L}^\top \boldsymbol{c}_{n-L} + \ldots + \boldsymbol{c}_n^\top \boldsymbol{c}_n = \underbrace{\sum_{i=n-L-1}^{n-1} \boldsymbol{c}_i^\top \boldsymbol{c}_i}_{\text{last results}} + \boldsymbol{c}_n^\top \boldsymbol{c}_n - \boldsymbol{c}_{n-L-1}^\top \boldsymbol{c}_{n-L-1},$$

Exponential window

$$\sum_{i=1}^{n} \phi^{n-i} \boldsymbol{c}_i^\top \boldsymbol{c}_i = \phi^n \boldsymbol{c}_1^\top \boldsymbol{c}_1 + \ldots \quad + \boldsymbol{c}_n^\top \boldsymbol{c}_n = \phi \underbrace{\sum \phi^{n-i-1} \boldsymbol{c}_i^\top \boldsymbol{c}_i}_{\text{last}} + \boldsymbol{c}_n^\top \boldsymbol{c}_n,$$

How to choose window shape and length?

Recursive least squares RLS

If we already collected n measurements, we do not collect statistics but its inverse:

$$U_n = V_n^{-1} = \left(\sum_{i=1}^{n} \phi_i^{n-i} \boldsymbol{c}_i^\top \boldsymbol{c}_i \right)^{-1}$$

incorporation the $n+1$ data record is

$$\hat{\boldsymbol{x}}_{n+1} = U_{n+1} v_{n+1},$$

$$U_{n+1} = \left(\sum_{i=1}^{n} \phi_i^{n-i} \boldsymbol{c}_i^\top \boldsymbol{c}_i + \boldsymbol{c}_{n+1}^\top \boldsymbol{c}_{n+1} \right)^{-1}$$

Recursive least squares RLS

If we already collected n measurements, we do not collect statistics but its inverse:

$$U_n = V_n^{-1} = \left(\sum_{i=1}^{n} \phi_i^{n-i} \boldsymbol{c}_i^\top \boldsymbol{c}_i \right)^{-1}$$

incorporation the $n+1$ data record is

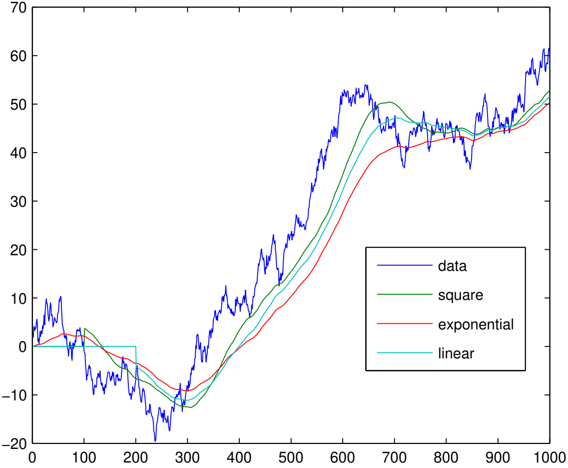
$$\hat{\boldsymbol{x}}_{n+1} = U_{n+1} v_{n+1},$$
$$U_{n+1} = \left(\sum_{i=1}^{n} \phi_i^{n-i} \boldsymbol{c}_i^\top \boldsymbol{c}_i + \boldsymbol{c}_{n+1}^\top \boldsymbol{c}_{n+1} \right)^{-1}$$

No need for inversion due to Matrix inversion lemma:

$$\begin{aligned} U_{n+1} &= \left(\phi U_n + \boldsymbol{x}_{n+1} \boldsymbol{x}_{n+1}^T \right)^{-1} \\ &= \phi^{-1} U_n - \frac{U_n \boldsymbol{x}_{n+1} \boldsymbol{x}_{n+1}^T U_n}{\phi^2 (1 + \boldsymbol{x}_{n+1}^T \phi^{-1} U_n \boldsymbol{x}_{n+1})}. \end{aligned}$$

Fast and numerically stable (square root form) algorithm.

Moving average – comparison



Probabilistic model of RLS

Bayesian formulation of least squares

$$
\begin{aligned}
&p(y_i|\boldsymbol{c}_i, \boldsymbol{x}) = \mathcal{N}(\boldsymbol{c}_i\boldsymbol{x}, r), \\
&p(\boldsymbol{x}|X, Y) \propto \mathcal{N}(\boldsymbol{c}_1^T\boldsymbol{x}, r)\mathcal{N}(\boldsymbol{c}_2^T\boldsymbol{x}, r)\ldots\mathcal{N}(\boldsymbol{c}_3^T\boldsymbol{x}, r)p(\boldsymbol{x}),
\end{aligned}
$$

Probabilistic model of RLS

Bayesian formulation of least squares

$$
\begin{aligned}
p(y_i|\boldsymbol{c}_i, \boldsymbol{x}) &= \mathcal{N}(\boldsymbol{c}_i \boldsymbol{x}, r),\\
p(\boldsymbol{x}|X, Y) &\propto \mathcal{N}(\boldsymbol{c}_1^T \boldsymbol{x}, r)\mathcal{N}(\boldsymbol{c}_2^T \boldsymbol{x}, r)\ldots\mathcal{N}(\boldsymbol{c}_3^T \boldsymbol{x}, r)p(\boldsymbol{x}),
\end{aligned}
$$

Bayesian formulation (discounting) of weighted least squares

$$
\begin{aligned}
p(y_i|\boldsymbol{c}_i, \boldsymbol{x}) &= \mathcal{N}(\boldsymbol{c}_i^T \boldsymbol{x}, r),\\
p(\boldsymbol{x}|X, Y) &\propto \mathcal{N}(\boldsymbol{c}_1 \boldsymbol{x}, r)^{\phi^{n-1}}\mathcal{N}(\boldsymbol{c}_2 \boldsymbol{x}, r)^{\phi^{n-2}}\ldots\mathcal{N}(\boldsymbol{c}_n \boldsymbol{x}, r)^1\ldots,
\end{aligned}
$$

Not a proper probability rule.
Solution:

- admit that $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$ are **different** but correlated variables

Probabilistic model of RLS

Bayesian formulation of least squares

$$p(y_i|\boldsymbol{c}_i, \boldsymbol{x}) = \mathcal{N}(\boldsymbol{c}_i\boldsymbol{x}, r),$$
$$p(\boldsymbol{x}|X, Y) \propto \mathcal{N}(\boldsymbol{c}_1^T\boldsymbol{x}, r)\mathcal{N}(\boldsymbol{c}_2^T\boldsymbol{x}, r)\ldots\mathcal{N}(\boldsymbol{c}_3^T\boldsymbol{x}, r)p(\boldsymbol{x}),$$

Bayesian formulation (discounting) of weighted least squares

$$p(y_i|\boldsymbol{c}_i, \boldsymbol{x}) = \mathcal{N}(\boldsymbol{c}_i^T\boldsymbol{x}, r),$$
$$p(\boldsymbol{x}|X, Y) \propto \mathcal{N}(\boldsymbol{c}_1\boldsymbol{x}, r)^{\phi^{n-1}}\mathcal{N}(\boldsymbol{c}_2\boldsymbol{x}, r)^{\phi^{n-2}}\ldots\mathcal{N}(\boldsymbol{c}_n\boldsymbol{x}, r)^1\ldots,$$

Not a proper probability rule.
Solution:

- admit that $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$ are **different** but correlated variables
- random walk model

$$\boldsymbol{x}_i = \boldsymbol{x}_{i-1} + \boldsymbol{e}_i,\ \ \boldsymbol{e}_i \sim \mathcal{N}(0, \epsilon I),$$

which is a discretization of stochastic process (Brownian motion).

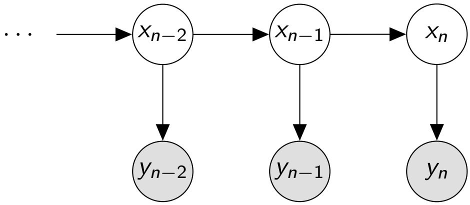
State space model

Bayesian formulation, example

$$p(y_n|\boldsymbol{c}_n, \boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{c}_n\boldsymbol{x}_n, r),$$
$$p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1}) = \mathcal{N}(\boldsymbol{x}_{n-1}, Q),$$

We no longer distinguish running index i and number of observations n. We will keep n as index of the last observation and assume that it keep increasing.

Conditional independence structure:



State-space model: example

Object of interest is a fighter jet with state variable $\boldsymbol{x}_n$ containing

- position,
- speed
- direction (yaw, pitch, roll)

Observation y_n available from radar:

- distance
- direction (bearing)

Inference in state space models

From observations $y_{1:n} = [y_1, \dots y_n]$ we can determine:
Path estimation, we seek trajectory $\boldsymbol{x}_{1:n} = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_n]$

$$p(\boldsymbol{x}_{1:n}|y_{1:n}) \propto p(y_n|\boldsymbol{x}_n)p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1}) \dots p(y_1|\boldsymbol{x}_1)p(\boldsymbol{x}_1)$$

Filtering:

$$p(\boldsymbol{x}_n|y_{1:n}) \propto \int p(\boldsymbol{x}_{1:n}|y_{1:n}) d\boldsymbol{x}_{1:n-1}$$

Smoothing (fixed lag L):

$$p(\boldsymbol{x}_{n-L}|y_{1:n}) \propto \int p(\boldsymbol{x}_{1:n}|y_{1:n}) d\boldsymbol{x}_{1:n-L-1} d\boldsymbol{x}_{n-L+1:n}$$

Prediction (h-step ahead):

$$p(\boldsymbol{x}_{n+h}|y_{1:n}) \propto p(\boldsymbol{x}_{n+h}|\boldsymbol{x}_{n+h-1}) \dots p(\boldsymbol{x}_n|y_{1:n}).$$

Bayesian filtering

Assume that we have previous estimate, model, and measurement:

$$p(\boldsymbol{x}_{n-1}|y_{1:n-1}) \quad p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1}) \quad p(y_n|\boldsymbol{x}_n) \quad y_n$$

we want new estimate

$$p(\boldsymbol{x}_n|y_{1:n}) = p(\boldsymbol{x}_n|y_{1:n-1}, y_n).$$

We can build

$$p(\boldsymbol{x}_{n-1}, \boldsymbol{x}_n, y_n|y_{1:n-1}) = p(\boldsymbol{x}_{n-1}|y_{1:n-1})p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1})p(y_n|\boldsymbol{x}_n).$$

Bayesian filtering

Assume that we have previous estimate, model, and measurement:

$$p(\boldsymbol{x}_{n-1}|y_{1:n-1}) \quad p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1}) \quad p(y_n|\boldsymbol{x}_n) \quad y_n$$

we want new estimate

$$p(\boldsymbol{x}_n|y_{1:n}) = p(\boldsymbol{x}_n|y_{1:n-1}, y_n).$$

We can build

$$p(\boldsymbol{x}_{n-1}, \boldsymbol{x}_n, y_n|y_{1:n-1}) = p(\boldsymbol{x}_{n-1}|y_{1:n-1})p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1})p(y_n|\boldsymbol{x}_n).$$

We apply marginalization on $\boldsymbol{x}_{n-1}$ and Bayes rule for y_n.

Bayesian filtering

Assume that we have previous estimate, model, and measurement:

$$p(\boldsymbol{x}_{n-1}|y_{1:n-1}) \quad p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1}) \quad p(y_n|\boldsymbol{x}_n) \quad y_n$$

we want new estimate

$$p(\boldsymbol{x}_n|y_{1:n}) = p(\boldsymbol{x}_n|y_{1:n-1}, y_n).$$

We can build

$$p(\boldsymbol{x}_{n-1}, \boldsymbol{x}_n, y_n|y_{1:n-1}) = p(\boldsymbol{x}_{n-1}|y_{1:n-1})p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1})p(y_n|\boldsymbol{x}_n).$$

We apply marginalization on $\boldsymbol{x}_{n-1}$ and Bayes rule for y_n.

$$\begin{aligned} p(\boldsymbol{x}_n|y_{1:n}) &= \int p(\boldsymbol{x}_n, \boldsymbol{x}_{n-1}|y_{1:n})d\boldsymbol{x}_{n-1} \\ &= \frac{p(y_n|\boldsymbol{x}_n) \int p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1})p(\boldsymbol{x}_{n-1}|y_{n-1})d\boldsymbol{x}_n}{p(y_n|y_{1:n-1})} \end{aligned}$$

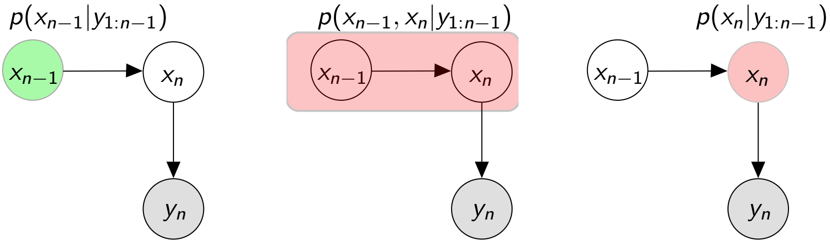
Bayesian filtering:

$$p(\boldsymbol{x}_n|y_{1:n-1}) = \int p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1})p(\boldsymbol{x}_{n-1}|y_{1:n-1})d\boldsymbol{x}_{n-1} \qquad \text{prediction}$$

$$p(\boldsymbol{x}_n|y_{1:n}) = \frac{p(y_n|\boldsymbol{x}_n)p(\boldsymbol{x}_n|y_{1:n-1})}{p(y_n|y_{1:n-1})}. \qquad \text{update}$$

Bayesian filtering: prediction

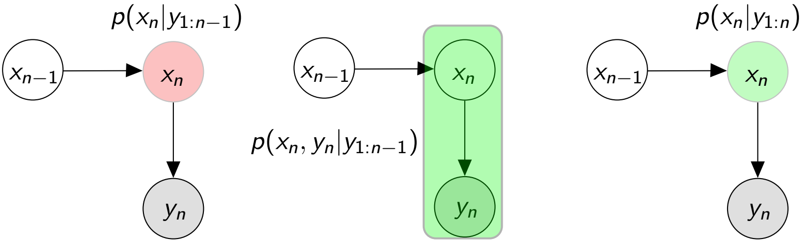
$$p(\boldsymbol{x}_n|y_{1:n-1}) = \int p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1})p(\boldsymbol{x}_{n-1}|y_{1:n-1})d\boldsymbol{x}_{n-1} \qquad \text{prediction}$$



State space model

$$p(\boldsymbol{x}_n|y_{1:n}) = \frac{p(y_n|\boldsymbol{x}_n)p(\boldsymbol{x}_n|y_{1:n-1})}{p(y_n|y_{1:n-1})}.$$

update



Special case of Linear Gaussian

General linear state-space model

$$p(\boldsymbol{x}_{n+1}|\boldsymbol{x}_n) = \mathcal{N}(A\boldsymbol{x}_n + B\boldsymbol{u}_n, Q)$$
$$p(\boldsymbol{y}_n|\boldsymbol{x}_n) = \mathcal{N}(C\boldsymbol{x}_n + D\boldsymbol{u}_n, R),$$

With initial condition $p(\boldsymbol{x}_0|y_0) = \mathcal{N}(\hat{x}_0, P_0), y_0 = \{\}$ goes:

$$p(\boldsymbol{x}_n, \boldsymbol{x}_{n-1}|y_{1:n-1}) = p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1})p(\boldsymbol{x}_{n-1}|y_{1:n-1}) \qquad \text{Gaussian}$$
$$p(\boldsymbol{x}_n|y_{1:n-1}) = \mathcal{N}(\hat{\boldsymbol{x}}_{n|n-1}, P_{n|n-1}) \qquad \text{marginal}$$
$$p(\boldsymbol{x}_n, y_n|y_{1:n-1}) = p(\boldsymbol{x}_n|y_{1:n-1})p(y_n|x_n) \qquad \text{Gaussian}$$
$$p(\boldsymbol{x}_n|y_{1:n}) = \mathcal{N}(\hat{\boldsymbol{x}}_{n|n}, P_{n|n}) \qquad \text{conditional}$$

using analytical marginal and conditionals of Gaussian distribution.

Decomposition of multivariate Gaussian

Joint distribution

$$p(x_1, x_2) = \mathcal{N}\left(\begin{bmatrix}\mu_1 \\ \mu_2\end{bmatrix}, \begin{bmatrix}\Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}\right)$$

has marginals:

$$p(x_1) = \mathcal{N}(\mu_1, \Sigma_{11}), \qquad p(x_2) = \mathcal{N}(\mu_2, \Sigma_{22}),$$

and conditional (Bayes rule)

$$p(x_2|x_1 = a) = \mathcal{N}(\overline{\mu}, \overline{\Sigma}) \qquad \begin{aligned} \overline{\mu} &= \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(a - \mu_1) \\ \overline{\Sigma} &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} \end{aligned}$$

The covariance is known as Schur complement.

State space model – prediction

Bayesian formulation, example

$$p(y_i|\boldsymbol{c}_i, \boldsymbol{x}_i) = \mathcal{N}(\boldsymbol{c}_i\boldsymbol{x}_i, r),$$
$$p(\boldsymbol{x}_i|\boldsymbol{x}_{i-1}) = \mathcal{N}(\boldsymbol{x}_{i-1}, Q),$$
$$p(\boldsymbol{x}_0) = \mathcal{N}(0, P_0),$$

Prediction

$$p(\boldsymbol{x}_1) = \int p(\boldsymbol{x}_1|\boldsymbol{x}_0)p(\boldsymbol{x}_0)d\boldsymbol{x}_0$$
$$\propto \int \exp\left(-\frac{1}{2}(\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top Q^{-1}(\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top - \frac{1}{2}\boldsymbol{x}_0^\top P_0^{-1}\boldsymbol{x}_0\right) d\boldsymbol{x}_0$$
$$\propto \int \exp\left(-\frac{1}{2}[\boldsymbol{x}_1, \boldsymbol{x}_0]^\top \begin{bmatrix} Q^{-1} & -Q^{-1} \\ -Q^{-1} & Q^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_0 \end{bmatrix} - \frac{1}{2}[\boldsymbol{x}_1, \boldsymbol{x}_0]^\top \begin{bmatrix} 0 & 0 \\ 0 & P_0^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_0 \end{bmatrix}\right.$$
$$\left(\begin{bmatrix} Q^{-1} & -Q^{-1} \\ -Q^{-1} & Q^{-1} + P_0^{-1} \end{bmatrix}\right)^{-1} = \begin{bmatrix} Q + P_0 & P_0 \\ P_0 & P_0 \end{bmatrix} \text{ using Schur complemen}$$
$$\propto \int \exp\left(-\frac{1}{2}[\boldsymbol{x}_1, \boldsymbol{x}_0]^\top \begin{bmatrix} Q + P_0 & P_0 \\ P_0 & P_0 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_0 \end{bmatrix}^\top\right) d\boldsymbol{x}_0$$

State space model – prediction

Bayesian formulation, example

$$\begin{aligned} p(y_i|\boldsymbol{c}_i, \boldsymbol{x}_i) &= \mathcal{N}(\boldsymbol{c}_i \boldsymbol{x}_i, r), \\ p(\boldsymbol{x}_i|\boldsymbol{x}_{i-1}) &= \mathcal{N}(\boldsymbol{x}_{i-1}, Q), \\ p(\boldsymbol{x}_0) &= \mathcal{N}(0, P_0), \end{aligned}$$

Prediction

$$\begin{aligned} p(\boldsymbol{x}_1) &= \int p(\boldsymbol{x}_1|\boldsymbol{x}_0) p(\boldsymbol{x}_0) d\boldsymbol{x}_0 \\ &\propto \int \exp\left(-\frac{1}{2}(\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top Q^{-1} (\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top - \frac{1}{2}\boldsymbol{x}_0^\top P_0^{-1} \boldsymbol{x}_0\right) d\boldsymbol{x}_0 \end{aligned}$$

$$\left(\begin{bmatrix} Q^{-1} & -Q^{-1} \\ -Q^{-1} & Q^{-1} + P_0^{-1} \end{bmatrix}\right)^{-1} = \begin{bmatrix} Q + P_0 & P_0 \\ P_0 & P_0 \end{bmatrix} \quad \text{using Schur complemen}$$

$$\propto \int \exp\left(-\frac{1}{2}[\boldsymbol{x}_1, \boldsymbol{x}_0]^\top \begin{bmatrix} Q + P_0 & P_0 \\ P_0 & P_0 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_0 \end{bmatrix}^\top\right) d\boldsymbol{x}_0$$

State space model – prediction

Bayesian formulation, example

$$\begin{aligned} p(y_i|\boldsymbol{c}_i, \boldsymbol{x}_i) &= \mathcal{N}(\boldsymbol{c}_i\boldsymbol{x}_i, r), \\ p(\boldsymbol{x}_i|\boldsymbol{x}_{i-1}) &= \mathcal{N}(\boldsymbol{x}_{i-1}, Q), \\ p(\boldsymbol{x}_0) &= \mathcal{N}(0, P_0), \end{aligned}$$

Prediction

$$\begin{aligned} p(\boldsymbol{x}_1) &= \int p(\boldsymbol{x}_1|\boldsymbol{x}_0)p(\boldsymbol{x}_0)d\boldsymbol{x}_0 \\ &\propto \int \exp\left(-\frac{1}{2}(\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top Q^{-1}(\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top - \frac{1}{2}\boldsymbol{x}_0^\top P_0^{-1}\boldsymbol{x}_0\right) d\boldsymbol{x}_0 \end{aligned}$$

$$\propto \int \exp\left(-\frac{1}{2}[\boldsymbol{x}_1, \boldsymbol{x}_0]^\top \begin{bmatrix} Q + P_0 & P_0 \\ P_0 & P_0 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_0 \end{bmatrix}^\top\right) d\boldsymbol{x}_0$$

State space model – prediction

Bayesian formulation, example

$$p(y_i|\boldsymbol{c}_i, \boldsymbol{x}_i) = \mathcal{N}(\boldsymbol{c}_i\boldsymbol{x}_i, r),$$
$$p(\boldsymbol{x}_i|\boldsymbol{x}_{i-1}) = \mathcal{N}(\boldsymbol{x}_{i-1}, Q),$$
$$p(\boldsymbol{x}_0) = \mathcal{N}(0, P_0),$$

Prediction

$$\begin{aligned}
p(\boldsymbol{x}_1) &= \int p(\boldsymbol{x}_1|\boldsymbol{x}_0)p(\boldsymbol{x}_0)d\boldsymbol{x}_0 \\
&\propto \int \exp\left(-\frac{1}{2}(\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top Q^{-1}(\boldsymbol{x}_1 - \boldsymbol{x}_0)^\top - \frac{1}{2}\boldsymbol{x}_0^\top P_0^{-1}\boldsymbol{x}_0\right) d\boldsymbol{x}_0 \\
&\propto \int \exp\left(-\frac{1}{2}[\boldsymbol{x}_1, \boldsymbol{x}_0]^\top \begin{bmatrix} Q^{-1} & -Q^{-1} \\ -Q^{-1} & Q^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_0 \end{bmatrix} - \frac{1}{2}[\boldsymbol{x}_1, \boldsymbol{x}_0]^\top \begin{bmatrix} 0 & 0 \\ 0 & P_0^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_0 \end{bmatrix}\right) \\
&\left(\begin{bmatrix} Q^{-1} & -Q^{-1} \\ -Q^{-1} & Q^{-1} + P_0^{-1} \end{bmatrix}\right)^{-1} = \begin{bmatrix} Q + P_0 & P_0 \\ P_0 & P_0 \end{bmatrix} \quad \text{using Schur complemen} \\
&\propto \int \exp\left(-\frac{1}{2}[\boldsymbol{x}_1, \boldsymbol{x}_0]^\top \begin{bmatrix} Q + P_0 & P_0 \\ P_0 & P_0 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_0 \end{bmatrix}^\top\right) d\boldsymbol{x}_0
\end{aligned}$$

The Kalman filter

General linear state-space model

$$\boldsymbol{x}_{n+1} = \mathcal{N}(A\boldsymbol{x}_n + B\boldsymbol{u}_n, Q)$$
$$\boldsymbol{y}_n = \mathcal{N}(C\boldsymbol{x}_n + D\boldsymbol{u}_n, R),$$

The solution is:

$$\begin{aligned}
p(\boldsymbol{x}_n|\boldsymbol{y}_{1:n}) &= \mathcal{N}(\hat{\boldsymbol{x}}_{n|n}, P_{n|n}), \\
\hat{\boldsymbol{x}}_{n|n} &= \hat{\boldsymbol{x}}_{n|n-1} + K(\boldsymbol{y}_n - \hat{\boldsymbol{y}}_n), \\
P_{n|n} &= (I - KC)P_{n|n-1}, \\
\hat{\boldsymbol{y}}_n &= C\hat{\boldsymbol{x}}_{n|n-1} + D\boldsymbol{u}_n, \\
K &= P_{n|n-1}C^T R_y^{-1}, \\
R_y &= C^T P_{n|n-1} C + R,
\end{aligned}$$

$$\begin{aligned}
p(\boldsymbol{x}_n|\boldsymbol{y}_{1:n-1}) &= \mathcal{N}(\hat{\boldsymbol{x}}_{n|n-1}, P_{n|n-1}), \\
\hat{\boldsymbol{x}}_{n|n-1} &= A\boldsymbol{x}_{n-1} + B\boldsymbol{u}_{n-1} \\
P_{n|n-1} &= AP_{n-1|n-1}A^T + Q
\end{aligned}$$

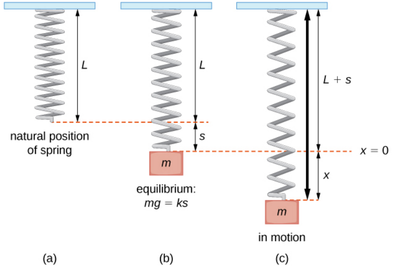
Representing state by a Gaussian distribution.

Example: Spring-mass system

Consider system spring mass system

$$ma = m\frac{d^2x}{dt} = -kx = F$$

where we want to track the position x from noisy measurements.

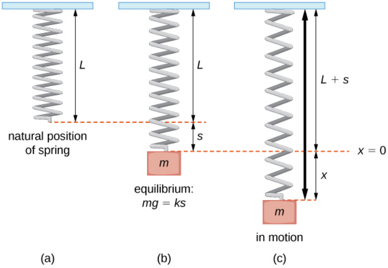


Example: Spring-mass system

Consider system spring mass system

$$ma = m\frac{d^2x}{dt} = -kx = F$$

where we want to track the position x from noisy measurements.



1. First order system

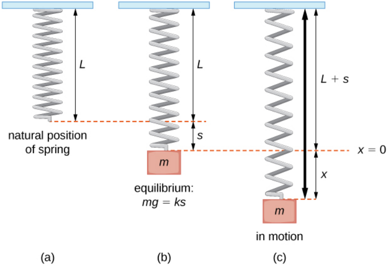
$$\frac{d}{dt}\begin{bmatrix} x \\ x' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k/m & 0 \end{bmatrix}\begin{bmatrix} x \\ x' \end{bmatrix}. \quad \Leftrightarrow \quad \frac{d}{dt}\boldsymbol{x} = A_c\boldsymbol{x}, \boldsymbol{x} = \begin{bmatrix} x \\ x' \end{bmatrix},$$

Example: Spring-mass system

Consider system spring mass system

$$ma = m\frac{d^2x}{dt} = -kx = F$$

where we want to track the position x from noisy measurements.



1. First order system

$$\frac{d}{dt}\begin{bmatrix} x \\ x' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k/m & 0 \end{bmatrix}\begin{bmatrix} x \\ x' \end{bmatrix}. \quad \Leftrightarrow \quad \frac{d}{dt}\boldsymbol{x} = A_c\boldsymbol{x}, \boldsymbol{x} = \begin{bmatrix} x \\ x' \end{bmatrix},$$

2. Discretization (Euler) $\boldsymbol{x}_n = \boldsymbol{x}(t), x_{n+1} = \boldsymbol{x}(t + \Delta t)$,

$$\frac{\boldsymbol{x}_{n+1} - \boldsymbol{x}_n}{\Delta t} = A_c\boldsymbol{x}, \implies \qquad \boldsymbol{x}_{n+1} = \underbrace{(I + \Delta t A_c)}_{A}\boldsymbol{x}_n,$$

Spring mass system

Exact solution of linear ODE:

$$\boldsymbol{x}(t+\Delta t) = e^{A_c \Delta t}\boldsymbol{x}(t),$$
$$\boldsymbol{x}_{n+1} = A\boldsymbol{x}_n,$$

Spring mass system

Exact solution of linear ODE:

$$\boldsymbol{x}(t+\Delta t) = e^{A_c \Delta t}\boldsymbol{x}(t),$$
$$\boldsymbol{x}_{n+1} = A\boldsymbol{x}_n,$$

Properties:

$$\text{eig}(A_c) = \pm\sqrt{-\frac{k}{m}}, \quad \omega = \sqrt{-\frac{k}{m}},$$
$$\boldsymbol{x}(t) = \begin{bmatrix} \cos(\omega t) & \frac{1}{\omega}\sin(\omega t) \\ -\omega\sin(\omega t) & \cos(\omega t) \end{bmatrix} \boldsymbol{x}(0)$$

Spring mass system

Exact solution of linear ODE:

$$\boldsymbol{x}(t + \Delta t) = e^{A_c \Delta t}\boldsymbol{x}(t),$$
$$\boldsymbol{x}_{n+1} = A\boldsymbol{x}_n,$$

Properties:

$$\mathrm{eig}(A_c) = \pm\sqrt{-\frac{k}{m}}, \quad \omega = \sqrt{-\frac{k}{m}},$$
$$\boldsymbol{x}(t) = \begin{bmatrix} \cos(\omega t) & \frac{1}{\omega}\sin(\omega t) \\ -\omega\sin(\omega t) & \cos(\omega t) \end{bmatrix} \boldsymbol{x}(0)$$

Observation (with noise):

$$\boldsymbol{y}_n = [x, x']^\top = \boldsymbol{I}\boldsymbol{x}_n + \epsilon \qquad \text{complete,}$$
$$\boldsymbol{y}_n = [x] = [1, 0]\boldsymbol{x}_n + \epsilon \qquad \text{position only,}$$
$$\boldsymbol{y}_n = [x'] = [0, 1]\boldsymbol{x}_n + \epsilon \qquad \text{velocity only}$$

Linear Gaussian State space

General linear Gaussian state-space model

$$\begin{aligned} \boldsymbol{x}_{n+1} &= A\boldsymbol{x}_n + v_n, & v_n &\sim \mathcal{N}(0, Q), \\ \boldsymbol{y}_n &= C\boldsymbol{x}_n + w_n, & w_n &\sim \mathcal{N}(0, R). \end{aligned}$$

where $\boldsymbol{x}_n$ is the state variable, and $\boldsymbol{y}_n$ is the observation at time n.

Linear Gaussian State space

General linear Gaussian state-space model

$$\begin{aligned} \boldsymbol{x}_{n+1} &= A\boldsymbol{x}_n + v_n, & v_n &\sim \mathcal{N}(0, Q), \\ \boldsymbol{y}_n &= C\boldsymbol{x}_n + w_n, & w_n &\sim \mathcal{N}(0, R). \end{aligned}$$

where $\boldsymbol{x}_n$ is the state variable, and $\boldsymbol{y}_n$ is the observation at time n.

- for $\omega = 10$, $\Delta t = 0.01$

$$A_c = \begin{bmatrix} 0 & 1 \\ -10^2 & 0 \end{bmatrix}, \quad A = e^{A_c \Delta t} = \begin{bmatrix} 0.9950040 & .00998334 \\ -0.998334 & 0.9950040 \end{bmatrix},$$

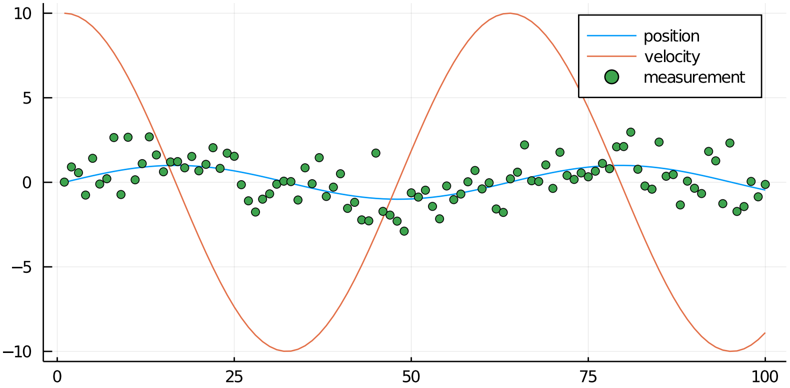
- Simulate with $Q = 0I_2$, $x_1 = [0, 10]$, $R = I_2$
- Estimate 100 steps with KF, for position only measurement $C = [1, 0]$

Simulation #1

System

$$\begin{aligned} \boldsymbol{x}_n &= A\boldsymbol{x}_{n-1} + 0v_n \\ y_n &= [1,0]\boldsymbol{x} + 1w_n \\ \boldsymbol{x}_0 &= [10,0] \end{aligned}$$

estimation, KF $p(\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{0}, I_2)$, $Q = 0I$, (test $Q = qI_2$)



Homework

Assignment	points
simulation + KF of Simulation #1	8
#2: add step change at $n = 200$: $x_{1,200} = x_{1,200} + 10$	
- perform sensitivity study to $Q = \text{diag}([q_1, q_2])$	15