Exercise 3

A computational introduction to stochastic differential equations
FTN0332 TN22H006

Zheng Zhao

How to pass this exercise

This exercise round is concerned with Lectures 6 - 8, and 10. To pass this exercise, score $\geq$ **10** points and finish the assignment(s) marked with $\star$. Please submit your assignments in an email sent to `zheng.zhao@it.uu.se` before 13:15, 18 Nov, 2022.

Assignment 1 (6 point)

Consider a linear Gaussian model

$$
\begin{aligned}
\mathrm{d}X(t) &= \begin{bmatrix} 0 & 1 \\ -3\,/\,\ell^2 & -2\sqrt{3}\,/\,\ell \end{bmatrix} X(t)\,\mathrm{d}t + \begin{bmatrix} 0 \\ 2\,(\frac{\sqrt{3}}{\ell})^{3\,/\,2}\,\sigma \end{bmatrix} \mathrm{d}W(t), \\
X(0) &\sim \mathrm{N}(0, V_0), \\
Y_k &= H\,X(t) + \xi_k, \quad \xi_k \sim \mathrm{N}(0, \Xi),
\end{aligned}
$$

where $X(t) \in \mathbb{R}^2, W(t) \in \mathbb{R}, Y_k \in \mathbb{R}, \ell = 2$, $\sigma = 1$, $\Xi = 1$, $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$, and V_0 is the stationary initial covariance (see, Exercise 2, Assignments 8 and 9).

- Simulate a pair of trajectory and measurements $(X_{1:T}, Y_{1:T})$ at times $t_1 = 0.1, t_2 = 0.2, \ldots, t_{1000} = 100$.

- Do Kalman filtering based on the simulated measurements $Y_{1:T}$. Plot the filtering mean $\mathbb{E}[H\,X(t_k) \mid Y_{1:k}]$ and the 0.95 quantile based on $\mathrm{Var}[H\,X(t_k) \mid Y_{1:k}]$ (e.g., use `plt.fill_between`) for $k = 1, 2, \ldots, 1000$. Compare the estimates to the true trajectory $X_{1:T}$.

- Do RTS smoothing at the times. Plot the smoothing mean $\mathbb{E}[H\,X(t_k) \mid Y_{1:T}]$ and the 0.95 quantile based on $\mathrm{Var}[H\,X(t_k) \mid Y_{1:T}]$ for $k = 1, 2, \ldots, 1000$. Compare the estimates to the true trajectory $X_{1:T}$ and the filtering estimates.

- (**Bonus +2 points**). Numerically verify that the filtering covariances are the same as $\mathbb{E}\big[(\mathbb{E}[X(t_k) \mid Y_{1:k}] - X_{1:k})\,(\mathbb{E}[X(t_k) \mid Y_{1:k}] - X_{1:k})^{\mathsf{T}}\big]$ for all k's.

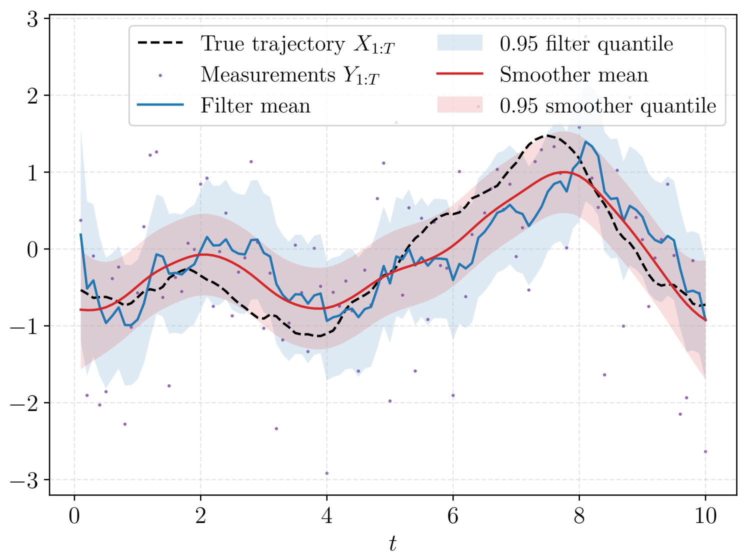

Figure 1: Example result of Assignment 1.

This shows that the Kalman filter is optimal in the least square sense. To do so, you can independently simulate multiple pairs of trajectory and measurements then do the filtering and approximate the expectation by Monte Carlo. Use less T if you encounter the out-of-memory error.

Assignment 2 (2 points)

Let us keep using the model in Assignment 1, and also the measurements you got from the first bullet point question. Now assume that its parameters ℓ and σ are unknown. Estimate the parameters by maximum likelihood estimation, and compare to the true values $\ell = 2$, $\sigma = 1$ which you used to generate the measurements. (Hint: recall the lecture note how to compute the negative log likelihood from Kalman filter, then use, e.g., `scipy.minimize`.)

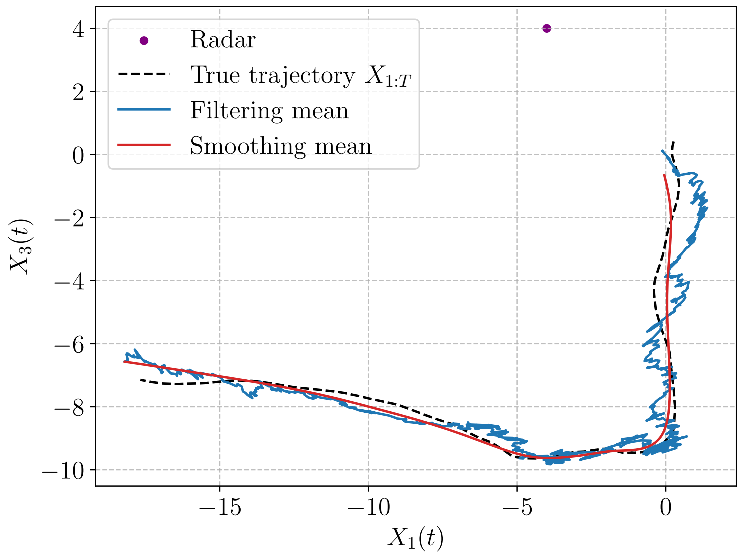

Figure 2: Example result of Assignment 3.

Assignment 3 (4 points)

Consider the motion tracking model

$$\begin{aligned}
\mathrm{d}\begin{bmatrix} X_1(t) \\ X_2(t) \\ X_3(t) \\ X_4(t) \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_1(t) \\ X_2(t) \\ X_3(t) \\ X_4(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \mathrm{d}W(t), \\
X(0) &\sim \mathrm{N}(0, I_4), \\
Y_k &= \begin{bmatrix} \sqrt{(X_1(t_k) - u)^2 + (X_3(t_k) - v)^2} \\ \arctan\Big((X_3(t_k) - v) \,/\, (X_1(t_k) - u) \Big) \end{bmatrix} + \xi_k, \\
\xi_k &\sim \mathrm{N}(0, 0.1\, I_2).
\end{aligned}$$

In the model above, X_1 and X_2 represent the object's position and velocity, respectively, in the abscissa. X_3 and X_4 represent the motion in the ordinate. The Brownian motion $W(t) \in \mathbb{R}^2$. We suppose that there is a radar at location (u, v) detecting the relative distance and bearing of the object.

- Set $u = -4$ and $v = 4$. Simulate a pair of trajectory and measurements $(X_{1:T}, Y_{1:T})$ at times $t_1 = 0.1, t_2 = 0.2, \ldots, t_{100} = 10$. (Note: `arctan2` may be preferred over `arctan`)
- Do the extended Kalman filter and smoother based on the simulated measurements $Y_{1:T}$. Compare the estimates to the true trajectory $X_{1:T}$.

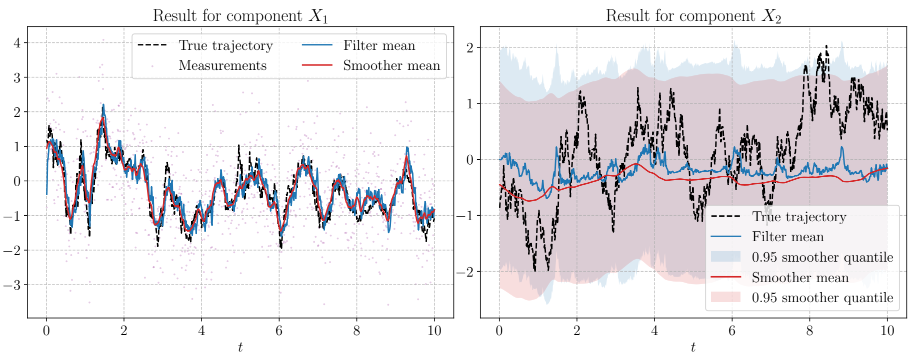

Figure 3: Example result of Assignment 4.

- (**Bonus +2 points**) Let us compare to a baseline approach for tracking the object. Remove the SDE for the process X, instead, consider a conventional non-linear least square problem ($f\colon \mathbb{R}^2 \to \mathbb{R}^2$):

$$
\begin{aligned}
f(z) &\coloneqq \begin{bmatrix} \sqrt{(z_1 - u)^2 + (z_2 - v)^2} \\ \arctan\Big((z_2 - v) \,/\, (z_1 - u)\Big) \end{bmatrix}, \quad z = \begin{bmatrix} z_1 & z_2 \end{bmatrix}, \\
Y_k &= f(z(k)) + \xi_k
\end{aligned}
$$

  Estimate the variables $\{z(1), z(2), \ldots, z(T)\}$ from $Y_{1:T}$ (the same measurements you have generated in the first bullet point question) by using any non-linear least square approach (e.g., Gauss–Newton) and compare to the results of the smoother, and explain the results.

Assignment 4 (4 points)

Consider a cascaded Ornstein–Uhlenbeck model

$$
\begin{aligned}
\mathrm{d}\begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} &= \begin{bmatrix} -\frac{1}{g(X_2(t))} & 0 \\ 0 & -\frac{1}{\ell_2} \end{bmatrix} \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} + \sqrt{2} \begin{bmatrix} \frac{\sigma_1}{\sqrt{g(X_2(t))}} & 0 \\ 0 & \frac{\sigma_2}{\sqrt{\ell_2}} \end{bmatrix} \mathrm{d}W(t), \\
X(0) &\sim \mathrm{N}(0, \operatorname{diag}(\sigma_1^2, \sigma_2^2)), \\
Y_k &= H\, X(t_k) + \xi_k, \quad \xi_k \sim \mathrm{N}(0, 1),
\end{aligned}
$$

where $g(\cdot) = \exp(\cdot)$, $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $\sigma_1 = 1$, $\ell_2 = 1$, and $\sigma_2 = 1$. Simulate a pair of trajectory and measurements $(X_{1:T}, Y_{1:T})$ at times $t_1 = 0.01, t_2 = 0.02, \ldots, t_{1000} = 10$, then apply the extended Kalman filter and smoother to solve the filtering and smoothing problem at the times based on the measurements. Compared the filtering and smoothing estimates to the simulated trajectory. You should get a similar result as in Figure 3.

(**Bonus +2 points**) Explain why the filtering/smoothing covariances for the component X_2 are so large, and we barely have a good estimate of X_2.