Homework exercise I

Filtering and Identification (SC42025)

Hand in pictures / scans of your hand-written solutions as a pdf for Exercises 1 and 2. For the MATLAB exercise, please export your live script as a pdf (instructions in template). Then, **merge all files as a single pdf and upload them through Brightspace** on Friday **25-11-2021** before **18:00**. You are allowed and encouraged to discuss the exercises together but you need to hand in individual solutions.

Please highlight your final answer!

Exercise 1 - State estimation

In this exercise, please consider the system

$$y_k = F_k x + v_k, v_k \sim \mathcal{N}(0, R_k), \tag{1}$$

where $x \in \mathbb{R}^{d_x}$ are states, $y_k \in \mathbb{R}^{d_y}$ are measurements, and $v_k \in \mathbb{R}^{d_y}$ is a measurement noise, and $R_k \in \mathbb{R}^{d_y \times d_y}$ is a positive definite measurement noise covariance matrix. In addition to the measurements y_k, we have prior information available at each timestep about the states x according to

$$x \sim \mathcal{N}(\tilde{x}, Q_x) \tag{2}$$

We use the following notation:

- $y_{1:N} = [y_1^\top, \cdots, y_N^\top]^\top$ indicates the vector stacking all measurements y_k.
- $R = \begin{bmatrix} R_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & R_N \end{bmatrix}$ is the covariance matrix of the vector $v_{1:N} = [v_1^\top, \cdots, v_N^\top]^\top$ of all measurement noises.
- $F_{1:N} = \begin{bmatrix} F_1 \\ \vdots \\ F_N \end{bmatrix}$

We assume that there is no correlation between the prior on the state and the measurement noise. In other words, the joint distribution of $v_{1:N}$ and $\tilde{x} - x$ is given by

$$\begin{bmatrix} v_{1:N} \\ \tilde{x} - x \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} R & 0 \\ 0 & Q_x \end{bmatrix} \right) \tag{3}$$

a) Please find the stochastic least squares expression for $\hat{x}$ and P_x such that $p(x|y_{1:N}) = \mathcal{N}(x\,;\,\hat{x}, P_x)$. Feel free to use formulas from the slides stating the stochastic least squares estimate.

b) Stochastic least squares regression is simply combining different sources of information. An alternative way of viewing that, is that the prior can be expressed as an additional measurement

$$\tilde{x} = x + w, \qquad w \sim \mathcal{N}(0, Q_x) \tag{4}$$

Write down all available information as an augmented measurement model

$$y^{\mathrm{A}} = F^{\mathrm{A}} x + v^{\mathrm{A}}, \qquad v^{\mathrm{A}} \sim \mathcal{N}(0, R^{\mathrm{A}}) \tag{5}$$

such that $y_A = [y_{1:N}^\top, \tilde{x}^\top]^\top$. (In other words, you have to give the expressions for F^{A} and R^{A} so that this holds). Then, prove that the least squares estimate from this model

$$((F^{\mathrm{A}})^\top (R^{\mathrm{A}})^{-1} F^{\mathrm{A}})^{-1} (F^{\mathrm{A}})^\top (R^{\mathrm{A}})^{-1} y^{\mathrm{A}} = \hat{x}, \tag{6}$$

where $\hat{x}$ is the expression you found in Exercise 1a).

c) Assume that you want to have access to an online recursive least-squares estimate $\hat{x}_k$ of the state given all measurements up until and including the one for time k, but that you only want to perform computations that involves matrices that do not increase in size with the amount of timesteps k. (Note that x is not time-varying, $\hat{x}_k$ denotes the estimate of x using measurements up until and including time k). Give an expression for the estimate $\hat{x}_0$ and covariance $P_{x,0}$ for $k = 0$, such that $p(x) = \mathcal{N}(\hat{x}_0, P_{x,0})$. Give closed-form expressions for the online estimate $\hat{x}_k$ and covariance $P_{x,k}$ using the previous estimate $\hat{x}_{k-1}$ and covariance $P_{x,k-1}$, and the current measurement y_k, such that $p(x|y_{1:k}) = \mathcal{N}(x; \hat{x}_k, P_{x,k})$. Comment on the relation between the estimate $\hat{x}_N$ and the estimate $\hat{x}$.

Exercise 2 - Parameter estimation

In this exercise, please consider the system

$$y_k = \sin(k\theta) + v_k, \qquad v_k \sim \mathcal{N}(0, R_k), \tag{7}$$

where $\theta \in \mathbb{R}$ is a parameter, $y_k \in \mathbb{R}$ are measurements, and $v_k \in \mathbb{R}$ is a measurement noise, and $R_k \in \mathbb{R}$ is a positive measurement noise covariance.
We use the following notation:

- $y_{1:N} = [y_1, \cdots, y_N]^\top$ indicates the vector stacking all measurements y_k.
- $R = \begin{bmatrix} R_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & R_N \end{bmatrix}$ is the covariance matrix of the vector $v_{1:N} = [v_1, \cdots, v_N]^\top$ of all measurement noises.

a) The stochastic least squares algorithm can be summarised as follows:

   - Choose an initial estimate $\hat{\theta}^{(i)}$
   - While not converged or maximum iterations not reached
     Linearise the model around the current estimate $F^{(i)} = \frac{df(\theta)}{d\theta}|_{\theta=\hat{\theta}^{(i)}}$
     Compute $e^{(i)} = y - f(\hat{\theta}^{(i)})$
     Solve a least squares problem with $e^{(i)} = F^{(i)}\Delta\theta + \epsilon$, $\epsilon \sim \mathcal{N}(0, R_\epsilon)$ to compute $\Delta\hat{\theta}^{(i)}$
     Update the estimate $\hat{\theta}^{(i+1)} = \hat{\theta}^{(i)} + \Delta\hat{\theta}^{(i)}$
     set i=i+1

   Please specify y, $F^{(i)}$, R_ϵ and $f(\hat{\theta}^{(i)})$ such that this algorithm gives the nonlinear least squares estimate $\hat{\theta}$ and its covariance P_θ such that $p(\theta|y_{1:N}) = \mathcal{N}(\theta\,;\,\hat{\theta}, P_\theta)$. You don't have to choose an initial estimate, and you can assume the algorithm will converge to a global minimum.

b) For this part of the assignment, we assume that in addition to the measurements y_k, we have prior information available at each timestep about the parameters θ according to

$$\theta \sim \mathcal{N}(\tilde{\theta}, Q_\theta). \tag{8}$$

   Please specify y, $F^{(i)}$ and $f(\hat{\theta}^{(i)})$ such that this algorithm gives the nonlinear stochastic least squares estimate $\hat{\theta}$ and its covariance P_θ such that $p(\theta|y_{1:N}) = \mathcal{N}(\theta\,;\,\hat{\theta}, P_\theta)$, including the prior information. Again, you don't have to choose an initial estimate, and you can assume the algorithm will converge to a global minimum.

MATLAB exercise

See the MATLAB live script `Matlab_1_template.mlx`.