EC 702
Spring, 2018
Homework 3

Due Mon, Mar. 19 2018

1. Consider an autoregressive (all-pole) discrete-time sequence $y(t)$ generated by a model of the form:

$$y(t) = -a_1 y(t-1) - a_2 y(t-2) - ... - a_p y(t-p) + w(t)$$

where $w(t)$ is a white, zero-mean process with $E[w(t)w(s)] = \sigma^2 \delta(t-s)$, and where the parameters $a_1, \ldots, a_p$ are unknown. Suppose we observe samples $y(-p), y(-p+1), ..., y(N)$, and wish to use these to estimate the autoregressive parameters. One approach to solving this problem is by least squares:

$$a_1^*, \ldots, a_p^* \leftarrow \min_{a_1,\ldots,a_p} \sum_{t=0}^{N-1} \frac{1}{\sigma^2} |y(t) + a_1 y(t-1) + \ldots + a_p y(t-p)|^2$$

An alternative approach is to solve this as a Kalman filtering problem, by constructing a state space model as follows: Let $\underline{a}(t)$ be the vector of autoregressive parameters that generate the sample $y(t)$ from $y(t-1), \ldots, y(t-p)$. Assume that, a priori, $\underline{a}(0)$ is a Gaussian random vector with zero mean and covariance Σ_0. Then, a state space model is given as

$$\underline{a}(t+1) = \underline{a}(t), \quad y(t) = \begin{pmatrix} y(t-1) & \ldots & y(t-p) \end{pmatrix} \underline{a}(t) + w(t)$$

The parameters $\underline{a}(t)$ behave like the state of this system. However, note that the observation matrix C is no longer deterministic, but depends on $y(t-1), \ldots, y(t-p)$.

(a) Is $y(t)$ a Gaussian random variable? Explain.

(b) Now, consider the problem of obtaining the best MMSE estimator of $\underline{a}$ given $y(-p), \ldots, y(t)$. Construct the innovations sequence $\nu(t) = y(t) - E[y(t)|y(-p), \ldots, y(t-1)]$. Is this innovations sequence uncorrelated over time? Are the innovations independent? Explain.

(c) Write a recursive form for the MMSE estimate of $x(t)$ given $y(-p), \ldots, y(t)$. This will be like a Kalman filter, but the matrix C will depend on past measurements. This is similar to the problem you did in HW 2 where you had a y-dependent input in the state dynamics.

2. This problem is to illustrate the uses of Extended Kalman Filters and their behaviors, depending on the choice of coordinates. Consider the following EKF problem: A sensor is mounted on a platform, moving sideways in discrete time at constant speed. As a function of discrete time index k, the x and y positions of the sensor location are given by:

$$x_s(k) = 4k; y_s(k) = 20$$

An object is moving randomly along the x axis, according to a random position and velocity model described in discrete - time as:

$$x(k+1) = x(k) + s(k) + 1/2w(k)$$
$$s(k+1) = s(k) + w(k)$$

where $w(k)$ is a white, Gaussian process noise with covariance q. For each discrete time index, the sensor provides a measurement of the angle between the object and the sensor, given by:

$$y(k) = a(k) + v(k)$$

where $a(k)$ is the true angle in radians, and $v(k)$ is measurement noise, with covariance $9\pi^2/(180x180)$, measured in radians squared. The true angle $a(k)$ is given by:

$$a(k) = \tan^{-1}\left(\frac{y_s(k)}{x(k) - x_s(k)}\right)$$

(a) Using the above coordinates, compute the equations to be used in an Extended Kalman Filter.

(b) Rewrite the above problem using polar coordinates for the state of the object, in terms of a new state $z(k)$ consisting of the angle $a(k)$ and the angular velocity $b(k)$. Develop the extended Kalman filter equations for this coordinate system. Note that you can't get an exact expression for the dynamics, but derive an approximate expression that has similar behavior by using a state-dependent term modulating the process noise.

(c) Implement in MATLAB the filters in parts a), c) with the initial condition that the state is known to be on the average at position x = 60, with initial average velocity s = 0, and with independent errors in position and velocity with standard deviations 20, 1 respectively.

(d) Test your filters out against simulated data available in file **track.ascii** in the Homework 3 folder.

3. Here is a problem of maneuver or failure detection. Consider the following system

$$\underline{x}(t+1) = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \underline{x}(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} w(t) + \begin{pmatrix} 0 \\ 10 \end{pmatrix} u_{-1}(t-a)$$

$$y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \underline{x}(t) + v(t)$$

where v, w are independent white noises with unit variance, and u_{-1} is the unit step function which is 1 if its argument is non-negative, 0 otherwise, and a is an unknown onset time to be estimated. The states x_1, x_2 of this system can be thought of as position and velocity, and the last term in the equation is an acceleration term to a vehicle maneuver or to a failure such as a leak in a jet in a space vehicle. We want to estimate the time a when this occurs by computing the ML estimate of a, by observing the output $y(t), t \in [0,T]$. Assume that $x(0)$ is Gaussian, zero-mean, with initial covariance P_x of x that satisfies the algebraic Riccati equation

$$P = APA^T + GG^T - APC^T(CPC^T + 1)^{-1}CPA^T$$

where

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, G = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 \end{pmatrix}$$

Note that this P is the steady state solution of the predicted error covariance equation.

(a) Use the variation of constants formula (or any other technique) to derive a decomposition for the output $y(t)$ as

$$y(t) = h(t-a)u_{-1}(t-a) + n(t)$$

where $n(t)$ is the zero-mean, colored noise and $h(t-a)$ is a deterministic function that depends on the time of the input $u_{-1}(t-a)$. This decomposition will follow from the linearity of the system, as there are two types of inputs: the noises $w(t), v(t)$ and the input $u_{-1}(t-a)$.

(b) Assume that $a > t$, so that the failure time occurs after time t. Consider the model for $y(s), 0 \le s \le t$. This is a standard Kalman filter model, and from the previous part, $y(t) = n(t)$. Find a causal, and causally invertible, whitening filter, to convert $n(t)$ to white noise, for $0 \le s \le t$. (Hint: Kalman filter...). Write the equations for this filter.

(c) Now, assume that the input may occur at $a < t$. From the first part, $y(t) = h(t-a)u_{-1}(t-a) + n(t)$. If you put this input into the whitening filter in the above part, you get an output (by superposition), a new signal $m(t)$ such that

$$m(t) = s(t-a)u_{-1}(t-a) + m_0(t)$$

where $m_0(t)$ is white, zero-mean and known variance and $s(t-a)$ is a known time signal. This is now a problem in detecting a known time signal in white noise. For a given $a < t$, compute the likelihood that the input disturbance occurred at time a given measurements $y(s), s \in [0,t]$. We can use this likelihood to estimate a if $a \le t$.

(d) Sometimes, it is easier to work with likelihood ratios. Compute the likelihood of the measurements $y(s), s \in [0,t]$ given hypothesis that a is the correct delay, divided by the same likelihood given the hypothesis that $a > t$ (so that $y(t) = n(t)$). Write an expression for the negative log likelihood ratio. (Hint: use innovations of the whitening filter...)

(e) Find an algorithm for computing the max likelihood estimate of a if $a < t$ using the above likelihood ratio.

4. This problem is to illustrate the uses of Extended Kalman Filters. Consider the following problem: There are two radar-like sensors, one deployed at (0,0), and the other one deployed nominally at (50,0). However, the second sensor has a relative location error in its position (bias in its x and y coordinates) and a pointing angle error (angle bias, so that the zero angle direction is different than that of sensor 1 by this bias).

   Both of the sensors can see a common moving object, and each sensor measures noisily estimates of range and angle to the object as it moves over time. These measurements are sent to a fusion station, that has to run an algorithm to estimate the position and velocity (in 2-D) of the object, as well as the biases (x, y and angle) of sensor 2.

   (a) Define a state vector consisting of the quantities of interest to estimate:

$$\underline{x} = \begin{pmatrix} p_x \\ v_x \\ p_y \\ v_y \\ bias_x \\ bias_y \\ bias_a \end{pmatrix}$$

Assume that measurements are collected every time unit ($dt = 1$). Model the dynamics of such a system in sampled time as

$$\underline{x}(t+1) = A\underline{x}(t) + \underline{w}(t)$$

where

$$A = \begin{pmatrix} 1 & dt & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & dt & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and $\underline{w}$ is a zero-mean, white noise matrix with covariance

$$Q = q * \begin{pmatrix} dt^3/3 & dt^2/2 & 0 & 0 & 0 & 0 & 0 \\ dt^2/2 & dt & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & dt^3/3 & dt^2/2 & 0 & 0 & 0 \\ 0 & 0 & dt^2/2 & dt & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.0009 \end{pmatrix}$$

The measurement equations for this estimation problem are nonlinear, and summarized as follows: at each time, a four-dimensional vector is measured, corresponding to the range and angle measurements by sensor 1 and sensor 2. In terms of the states defined above, the measurement equation is:

$$\underline{y} = \begin{pmatrix} R_1 \\ A_1 \\ R_2 \\ A_2 \end{pmatrix}$$

where angles are measured in radians, and

$$R_1 = \sqrt{(px)^2 + (py)^2} + v_{r1}$$

$$R_2 = \sqrt{(px - 50 - bias_x)^2 + (py - bias_y)^2} + v_{r2}$$

$$A_1 = \tan^{-1}\frac{py}{px} + v_{a1}$$

$$A_2 = \tan^{-1}\left(\frac{py - bias_y}{px - 50 - bias_x}\right) + bias_a + v_{a2}$$

and $v_{r1}, v_{r2}, v_{a1}, v_{a2}$ are white, Gaussian noises with zero mean and variances $R_{r1} = R_{r2} = 4$ and $R_{a1} = R_{a2} = .01$.

(b) In the assignments page in the web site, you will find a Matlab file containing a shell script (**hw3p4script.m**) setting up this problem, and generating data for this problem. Download this file to your workspace and read it carefully.

(c) Implement an extended Kalman filter for this problem in this file where indicated, using the state space and measurement equations described above. Note the initial conditions required for this filter in the Matlab file you downloaded.

(d) Run the script as indicated, evaluating the filter against 40 instances of noise realizations for this problem. Generate plots of the absolute error for the filter, averaged across the 40 instances (see script file). Save this work, as you will use it in the next HW.