Probabilistic Robotics Homework 1: Kalman Filter

Aparna Penmetcha

February 8 2020

1 Kalman Filter: Prediction

In this homework, the task is to design a Kalman Filter for a simple dynamical system. You are the captain of a sailboat in a 1-D ocean where x_t is the position of your boat at time t, while $\dot{x}_t$ and $\ddot{x}_t$ are the velocity and acceleration. For simplicity, assume that $\triangle t = 1$.

Due to random wind fluctuations, at each new time step, your acceleration is set randomly accordingly to the distribution $\mathcal{N}(\mu_{wind}, \sigma^2_{wind})$, where $\mu_{wind} = 0.0$ and $\sigma^2_{wind} = 1.0$.

Question 1.1: What is the minimal state vector for the Kalman filter so that the resulting system is Markovian?

ANS 1.1

The minimal state vector includes both the position and velocity:

$$M_t = \begin{bmatrix} x_t \\ \dot{x}_t \end{bmatrix}$$

With x_t and $\dot{x}_t$ being the position and velocity of the boat respectively

Question 1.2: Design the state transition probability function $p(x_t|u_t, x_{t-1})$. The transition function should contain linear matrices A and B and a noise covariance R.

ANS 1.2

The kinematic equations for position and velocity are as follows:

$$x_{t+1} = x_t + \Delta\mathrm{t} \times \dot{x}_t + \tfrac{1}{2}(\Delta\mathrm{t})^2 \times \ddot{x}_t$$

$$\dot{x}_{t+1} = \dot{x}_t + \Delta\mathrm{t} \times \ddot{x}_t$$

The position and velocity at time $t+1$ is predicted based on the previous position and velocity at time t.

The Δt is considered to be 1 which simplifies the resulting equations as:

$$x_{t+1} = x_t + 1 \times \dot{x}_t + \frac{1}{2} \times \ddot{x}_t$$

$$\dot{x}_{t+1} = \dot{x}_t + 1 \times \ddot{x}_t$$

The next state probability $p(x_t|u_t, x_{t-1})$ has to be a linear function with added Gaussian noise. The equation is as follows:

$$x_t = A_t x_{t-1} + B_t u_t + \epsilon_t$$

Since there is no control factor in this portion of the problem, $u_t = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ which would simplify B $\times$ u_t to 0 as well.

This equation is represented in matrix form,

$$\begin{bmatrix} x_{t+1} \\ \dot{x}_{t+1} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}}_{A} \times \begin{bmatrix} x_t \\ \dot{x}_t \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{1}{2} \times \ddot{x}_t \\ 1 \times \ddot{x}_t \end{bmatrix}}_{\epsilon_t}$$

$$\begin{bmatrix} x_{t+1} \\ \dot{x}_{t+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_t \\ \dot{x}_t \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}}_{G} \times \ddot{x}_t$$

With:

$\epsilon_t \approx \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, R \right)$

R $= \sigma^2 \underbrace{\begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}}_{G} \underbrace{\begin{bmatrix} \frac{1}{2} & 1 \end{bmatrix}}_{G^T}$

Question 1.3: Implement the state prediction step of the Kalman filter, assuming that at time $t = 0$, we start at rest, i.e., $x_t = \dot{x}_t = \ddot{x}_t = 0.0$. Use your code to calculate the state distribution for times $t = 1, 2, \ldots, 5$.

ANS 1.3:
Using the first two lines of the Kalman filter algorithm:

$$\bar{\mu_t} = A_t \ \mu_{t-1} + B_t \ u_t$$

$$\bar{\Sigma_t} = A_t \ \Sigma_{t-1} \ A_t^T + R_t$$

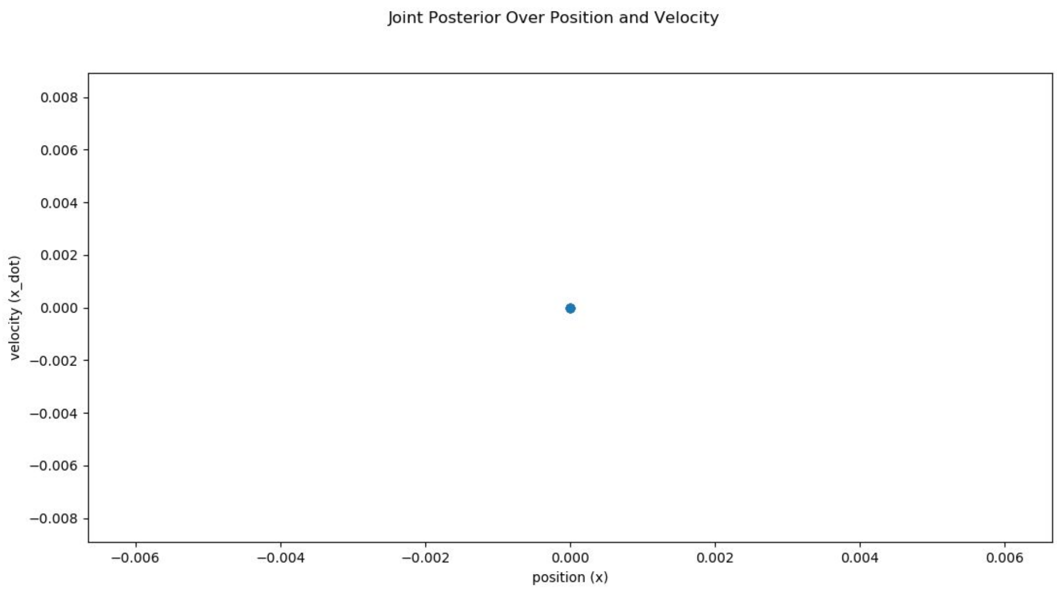
The state distribution for times t = 0,1,2,...,5 is:

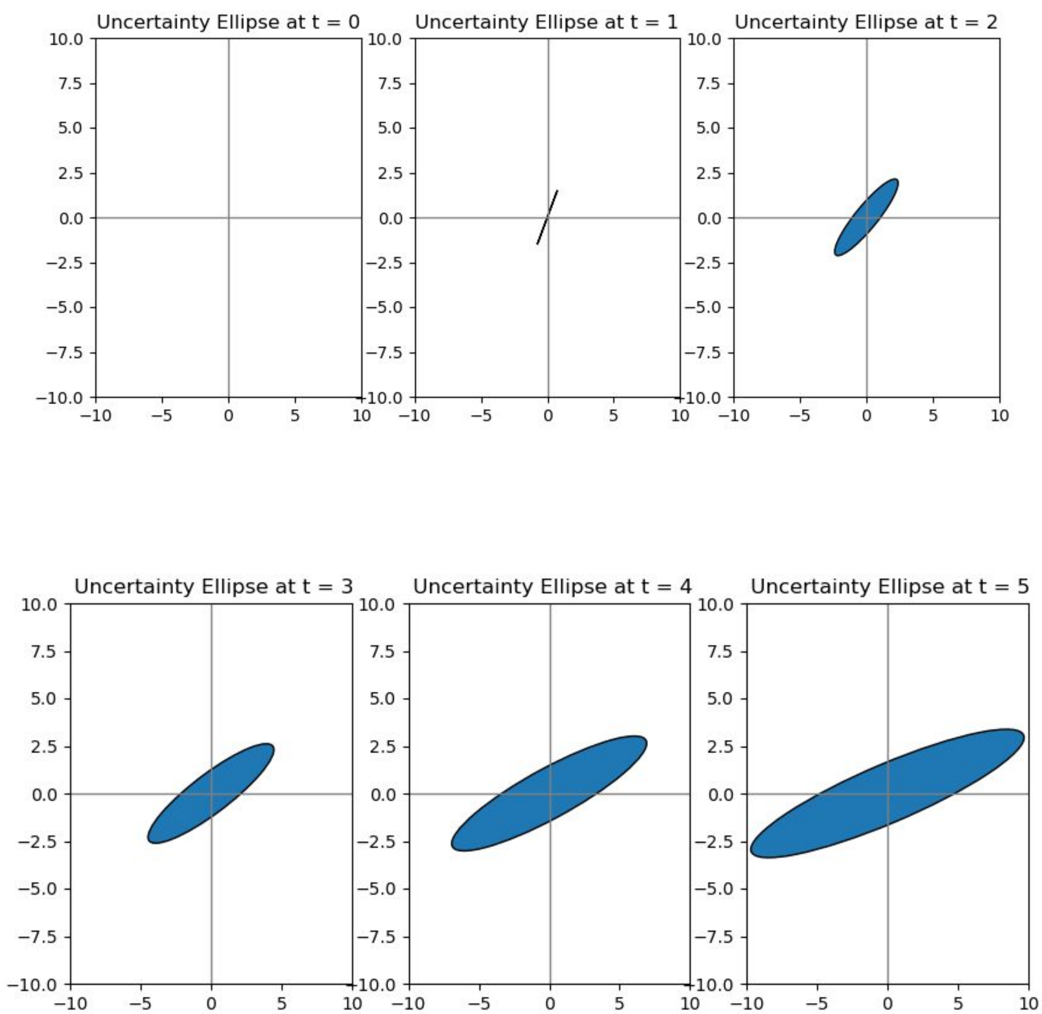
At each time step, the $\bar{\mu_t} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

The $\bar{\Sigma_t}$ at t = 1,2,...,5 are as follows:

t	Σ_t
0	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$
1	$\begin{bmatrix} 0.25 & 0.5 \\ 0.5 & 1 \end{bmatrix}$
2	$\begin{bmatrix} 2.5 & 2 \\ 2 & 2 \end{bmatrix}$
3	$\begin{bmatrix} 8.75 & 4.5 \\ 4.5 & 3 \end{bmatrix}$
4	$\begin{bmatrix} 21 & 8 \\ 8 & 4 \end{bmatrix}$
5	$\begin{bmatrix} 41.25 & 12.5 \\ 12.5 & 5 \end{bmatrix}$

Question 1.4: For each value of t in the previous question, plot the joint posterior over x and $\dot{x}$ in a diagram where x is the horizontal and $\dot{x}$ is the vertical axis. For each posterior, you are asked to plot the uncertainty ellipse which is the ellipse of points that are one standard deviation away from the mean. Some additional information about uncertainty ellipses and how to calculate them using MATLAB or C++ can be found here: http://www.visiondummy.com/2014/04/draw-error-ellipse-representing-covariance-matrix/.


Joint Posterior Over Position and Velocity
velocity (x_dot)
position (x)


Uncertainty Ellipse at t = 0
Uncertainty Ellipse at t = 1
Uncertainty Ellipse at t = 2
Uncertainty Ellipse at t = 3
Uncertainty Ellipse at t = 4
Uncertainty Ellipse at t = 5

2 Kalman Filter: Measurement

Prediction alone will result in greater and greater uncertainty as time goes on. Fortunately, your sailboat has a GPS sensor, which in expectation, measures the true position. However, the measurement is corrupted by Gaussian noise with covariance $\sigma^2_{gps} = 8.0$.

Question 2.1: Define the measurement model. You will need to define matrices C and Q.

ANS 2.1

The measurement probability $p(z_t|x_t)$ has to also be linear in its arguments, with added Gaussian noise. The equation is as follows:

$$z_t = C_t x_t + \delta_t$$

With:

$\delta_t \approx$ N $(0, Q)$

Q $= \sigma^2_{gps} = 8.0$

C $= \begin{bmatrix} 1 & 0 \end{bmatrix}$

Question 2.2: Implement the measurement update. Suppose at time $t = 5$, we query our sensor for the first time to obtain the measurement $z = 10$. State the parameters of the Gaussian estimate before and after incorporating the measurement. Afterwards, implement the sensor modal to randomly sample the true position, corrupted with noise σ^2_{gps}.

ANS 2.2

Before measurement

$$\begin{aligned} \bar{\mu_5} &= \begin{bmatrix} 0 & 0 \end{bmatrix} \\ \bar{\Sigma_5} &= \begin{bmatrix} 41.25 & 12.5 \\ 12.5 & 5 \end{bmatrix} \end{aligned}$$

After measurement

$$\begin{aligned} \mu_5 &= \begin{bmatrix} 8.38 \\ 2.54 \end{bmatrix} \\ \Sigma_5 &= \begin{bmatrix} 6.70 & 2.03 \\ 2.03 & 1.83 \end{bmatrix} \end{aligned}$$

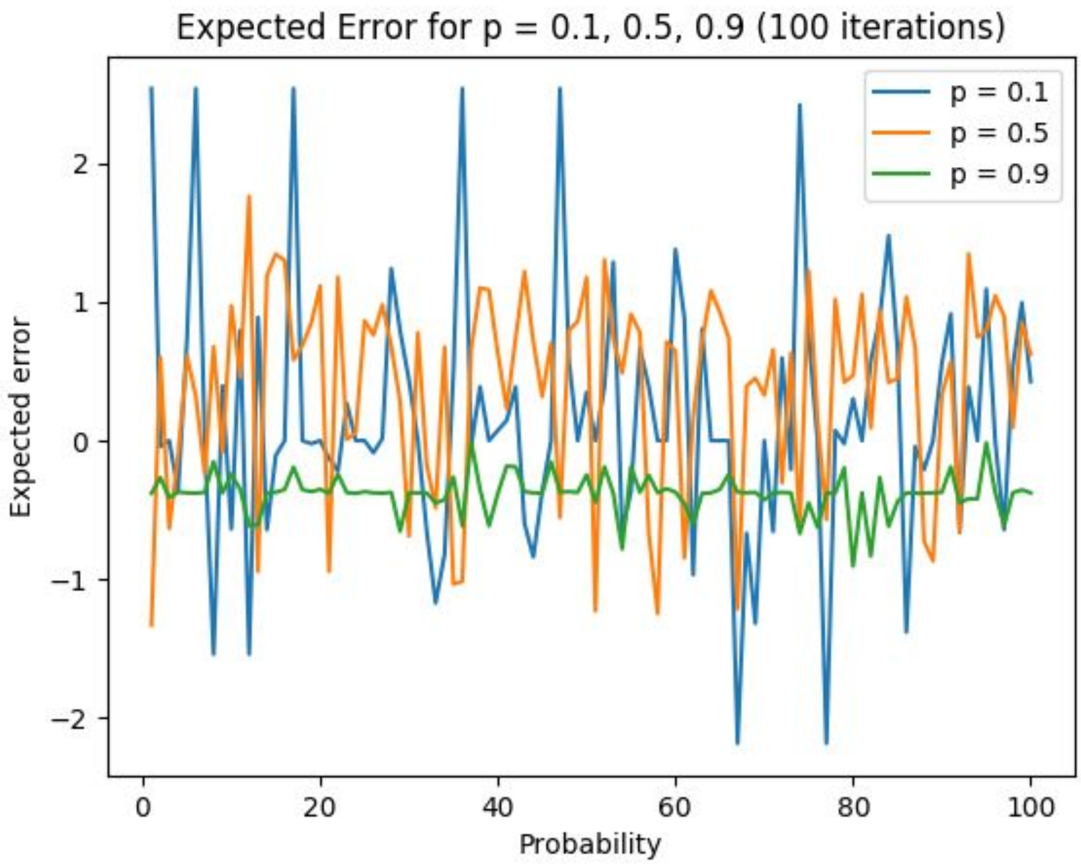
$$K_5 = \begin{bmatrix} 0.83 \\ 0.25 \end{bmatrix}$$

Question 2.3: All of a sudden, the sky gets cloudy which may cause the sensor to fail and not produce a measurement with probability $p_{gps-fail}$. For three different values of this probability (e.g., 0.1, 0.5, and 0.9), compute and plot the expected error from the true position at time $t = 20$. You may do so by running up to N simulations and use the observed errors to obtain the expected error empirically.

3 Extra Credit

Now, formulate both the prediction and measurement steps in the 2-D case. Construct a plot showing the true position and the position tracked by the Kalman filter over the first 30 time steps.

What to turn in: A PDF document with the answers to the questions, along with the code implementation and a README file that describes what to run in order to get the results in your PDF. You can use a language of your choice.


Expected Error for p = 0.1, 0.5, 0.9 (100 iterations)
p = 0.1
p = 0.5
p = 0.9
Expected error
Probability
2
1
0
−1
−2
0
20
40
60
80
100