Assignment-2: Kalman Filter

(CSL 7580 Advanced AI)

Report

Assignment	**02**
Roll No.	**B20CS018**
Name	**Harshita Gupta**

PART 1

Introduction:

In this scenario, we have an Autonomous Surface-to-Air Missile-launcher (ASAM) that needs to track an enemy aircraft(target) using a RADAR that scans a large area and takes time T to make a full rotation while scanning. The ASAM takes input from the RADAR and is designed to lock on and fire missiles at a given location as the target enters a specific range. To do so, it needs to continuously monitor the target's position using the Kalman filter.

Model:

The enemy aircraft(target) can be modeled as a discrete dynamic system with state S_t=[v; x] where v is the 3-D velocity vector, and x is the 3-D position vector. The acceleration control u_t=a_t acting on the target is known, which is a 3-D acceleration vector at that given time instant. The equations to deduce the next corresponding values of the state are given:

$v_{t+1} - v_t = a_t \times \Delta t$
$x_{t+1} - x_t = v_t \times \Delta t$

Assumptions:

1. The time taken between each time tick for the ASAM's processor is the same Δt, which is 1.
2. All RADAR readings are subject to a Gaussian noise η, which is known.
3. The measurements of position, velocity, and acceleration are independent, and there is no co-variance in error for the observed readings.

Kalman Filter Equations:

The Kalman filter is an algorithm that uses a series of measurements observed over time, containing statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone. The Kalman filter equations for this scenario are:

Prediction Step

In the prediction step, the Kalman filter predicts the state of the target at time t+1 based on the state and control input at time t. This is done by propagating the state forward using the equations of motion. The predicted state is represented as $\hat{S}_{t+1}=[\hat{v}; \hat{x}]$, and the state transition matrix is F.

$\hat{S}_{t+1} = F * \hat{S}_t + B * u_t$
where $F = [1\ \Delta t; 0\ 1]$ for position and velocity, and $B = [\Delta t^2/2; \Delta t]$ for acceleration.

The state covariance matrix P_t is also propagated forward using the error dynamics equation.

$\hat{P}_{t+1} = F * P_t * F^T + Q$
where Q is the process noise covariance matrix. In this case, we assume that the acceleration is constant, so Q is given by:

$Q = G * G^T * \sigma^2$
where $G = [\Delta t^2/2; \Delta t]$, and σ is the standard deviation of the acceleration.

Update Step

In the update step, the Kalman filter corrects the predicted state based on the measurements from the radar. The measured state is represented as $z_t=[v_m; x_m]$, and the measurement matrix H is given by:

$H = [1\ 0; 0\ 1]$

The measurement noise covariance matrix R is assumed to be known.

The Kalman gain K_t is computed using the predicted state covariance matrix $\hat{P}_{t+1}$, the measurement noise covariance matrix R, and the measurement matrix H.

$K_t = \hat{P}_{t+1} * H^T * (H * \hat{P}_{t+1} * H^T + R)^{-1}$

The corrected state is given by:

$S_{t+1} = \hat{S}_{t+1} + K_t * (z_t - H * \hat{S}_{t+1})$

The state covariance matrix is updated as:

$P_{t+1} = (I - K_t * H) * \hat{P}_{t+1}$

Example

Assume the initial state of the target is [v0, x0] = [10, 50]. Let the acceleration control a_t be [2, 1] at each time step, and let the Gaussian noise η for all measurements be 0.5. Also, let k = 2.

Here are the steps that would be followed in each iteration:

Prediction Step:

Predict the next state of the target using the dynamic model equations:

$\hat{v}_{t+1} = \hat{v}_t + a_t$

$\hat{x}_{t+1} = \hat{x}_t + \hat{v}_t$

Predict the error covariance matrix using the dynamic model:

$\hat{P}_{t+1} = P_t + Q$

where Q is the process noise covariance matrix, assumed to be diagonal and defined as follows:

Q = [1 0; 0 1]

Update Step:

Obtain the measurement of the target's position and velocity from the RADAR. Let's say the measured values are [v_meas, x_meas] = [12, 52].

Calculate the Kalman gain using the error covariance matrix and the measurement noise covariance matrix R, which is also assumed to be diagonal:

$K = \hat{P}_{t+1} / (\hat{P}_{t+1} + R)$

where R = [0.5 0; 0 0.5]

Update the estimated state and error covariance matrix using the Kalman gain and the measurement:

$\hat{v}_{t+1} = \hat{v}_{t+1} + K * (v\_meas - \hat{v}_{t+1})$

$\hat{x}_{t+1} = \hat{x}_{t+1} + K * (x\_meas - \hat{x}_{t+1})$

$\hat{P}_{t+1} = (I - K) * \hat{P}_{t+1}$

where I is the identity matrix.

Repeat the Prediction and Update steps for k-1 more times before the next RADAR reading is obtained.

Now let's apply these steps for 5 iterations:

Iteration 1:

Initial state: [10, 50]

Dynamic model prediction: [12, 62]

Error covariance prediction: [2, 0; 0, 2]
RADAR measurement: [12.5, 52.5]
Kalman gain: [0.8, 0; 0, 0.8]
Estimated state update: [12, 52]
Error covariance update: [0.4, 0; 0, 0.4]

Iteration 2:

Dynamic model prediction: [14, 64]
Error covariance prediction: [1.6, 0; 0, 1.6]
Dynamic model prediction: [16, 66]
Error covariance prediction: [2, 0; 0, 2]
RADAR measurement: [16.5, 67.5]
Kalman gain: [0.8, 0; 0, 0.8]
Estimated state update: [16, 66]
Error covariance update: [0.4, 0; 0, 0.4]

Iteration 3:

Dynamic model prediction: [18, 68]
Error covariance prediction: [1.6, 0; 0, 1.6]
Dynamic model prediction: [20, 70]
Error covariance prediction: [2, 0; 0, 2]
RADAR measurement: [20.5, 71.5]
Kalman gain: [0.8, 0; 0, 0.8]
Estimated state update: [20, 70]
Error covariance update: [0.4, 0; 0, 0.4]

Iteration 4:

Dynamic model prediction: [22, 72]
Error covariance prediction: [1.6, 0; 0, 1.6]
Dynamic model prediction: [24, 74]
Error covariance prediction: [2, 0; 0, 2]
RADAR measurement: [24.5, 75.5]
Kalman gain: [0.8, 0; 0, 0.8]
Estimated state update: [24, 74]
Error covariance update: [0.4, 0; 0, 0.4]

Iteration 5:

Dynamic model prediction: [26, 76]
Error covariance prediction: [1.6, 0; 0, 1.6]
Dynamic model prediction: [28, 78]
Error covariance prediction: [2, 0; 0, 2]

RADAR measurement: [28.5, 79.5]
Kalman gain: [0.8, 0; 0, 0.8]
Estimated state update: [28, 78]
Error covariance update: [0.4, 0; 0, 0.4]

As we can see, the estimated state of the target is updated after every two RADAR measurements. This is because k=2, so the ASAM is updated twice before every two RADAR readings. The estimated state is predicted using the dynamic model and then updated using the Kalman gain and the RADAR measurement. The error covariance matrix is also updated after each measurement.

This process of prediction and update is repeated in every iteration to track the target's position and velocity over time, and the Kalman filter provides the best estimate of the target's state by minimizing the error between the predicted and measured values.

PART 2

The `__init__` function of the `KalmanFilter` class initializes the filter's state matrices `Q` (process noise covariance), `R` (measurement noise covariance), `P` (estimate error covariance), `H` (observation matrix), and `K` (Kalman gain).

The `input` function is called multiple times during the simulation. When `justUpdated` is True, it means that an observation has just been taken, so the state estimate is updated based on the observation. When `justUpdated` is False, it means that the filter has not received a new observation yet, so it predicts the next state estimate based on the current estimate and the control input.

The `update` function updates the current state estimate of the target based on the predicted estimate and the actual observation.

The `RunEnvironment` function runs the simulation for a given number of trials and computes the average error between the estimated state and the true state of the target. The function takes as input the number of trials, the time between updates, the acceleration function, the measurement noise, and the type of estimator to use.

OUTPUT -

```
harshita-gupta@harshitagupta-Inspiron-15-5501:~/Desktop/Semester-6/Advanced AI/A
ssignment-2$ python3 tests.py
Error after test run: 83.94249759402742
Error after test run: 1017821.4168227058
Error after test run: 9131096.954197286
Error after test run: 4.365413854672446e+148
Error after test run: 2.6150769640118927e+148
```

PART 3

In the case of a falling projectile, the state of the object can be defined as its position x and its velocity v. The acceleration a of the object is given by the force of gravity minus the air resistance, which reaches 0 when the object reaches its terminal velocity. Therefore, the state space equation for this system can be rewritten as follows:

State Space Equation:

x(k) = F(k-1)*x(k-1)* + *B*u(k-1) + w(k-1) where x(k) = [x(k), v(k)]^T
y(k) = H(k)*x(k) + v(k) where y(k) is the observed position from RADAR

where F(k-1) is the state transition matrix, B is the input control matrix, u(k-1) is the input control vector, w(k-1) is the process noise vector, H(k) is the observation matrix, and v(k) is the measurement noise.

The state transition matrix for this system is:

F(k-1) = [1 delta_t; 0 1] where delta_t is the time step

The input control matrix and vector are:

B = [0.5*delta_t^2; delta_t]
u(k-1) = a(k-1) = g - (c/m)*v(k-1) where g is the gravitational acceleration, c is the air resistance coefficient, and m is the mass of the object

The process noise vector is assumed to be Gaussian white noise with zero mean and covariance Q(k-1):

w(k-1) ~ N(0, Q(k-1)) where Q(k-1) = [q1*delta_t^3/3 q1*delta_t^2/2; q1*delta_t^2/2 q1*delta_t]

The observation matrix is:

H(k) = [1 0]

The measurement noise is assumed to be Gaussian white noise with zero mean and covariance R(k):

v(k) ~ N(0, R(k)) where R(k) = [r1 0; 0 r2]

The Kalman filter equations for this system can be written as follows:

Prediction:

x_hat(k|k-1) = F(k-1)*x_hat(k-1|k-1)* + *B*u(k-1)
P(k|k-1) = F(k-1)*P(k-1|k-1)*F(k-1)^T + Q(k-1)

Update:

K(k) = P(k|k-1)*H(k)^T*(H(k)*P(k|k-1)*H(k)^T + R(k))^-1*
x_hat(k|k) = x_hat(k|k-1) + *K(k)*(y(k) - H(k)*x_hat(k|k-1))
P(k|k) = (I - K(k)*H(k))*P(k|k-1)

where x_hat(k|k-1) and P(k|k-1) are the predicted state and covariance, x_hat(k|k) and P(k|k) is the updated state and covariance, K(k) is the Kalman gain, and I is the identity matrix.

The main difference between these equations and the previous state equations is the inclusion of the air resistance and the terminal velocity in the system model. This changes the state transition matrix, input control matrix and vector, process noise vector, and observation matrix. Additionally, the measurement noise covariance matrix may need to be adjusted to account for the accuracy and precision of the RADAR system.