# Miniset 2

Saturday, September 17, 2022 8:17 PM

BF-Q1. In robotics, we commonly want to keep track of the possible values (or distribution of values) our current state could take on given all previous actions and measurements we have taken or observed up until the current time. We call this our **posterior** belief and often denote it as follows:

$$Bel(x_t) = p(x_t|u_1, z_1, \cdots, u_t, z_t). \tag{1}$$

The Bayes filter provides a method for recursively updating our estimate of our state given only the most recent measurement(s), the most recent action, and our belief at the previous time step:

$$Bel(x_t) = \overbrace{\eta p(z_t|x_t)}^{\text{Correction Step}} \overbrace{\int p(x_t|u_t, x_{t-1}) \underbrace{Bel(x_{t-1})}_{\text{prior belief}} dx_{t-1}}^{\text{Update step}} \tag{2}$$

The bayes filter is often broken down into the correction step, the update step, and the prior term. Label the parts of equation 2 that refer to these three main components.

Show how to derive equation 2 from equation 1. Label each step of the derivation with a description of what rule/law/assumption is being used to make it.

$Bel(x_t) = p(x_t|u_1, z_1, \cdots, u_t, z_t)$

Bayes $= \eta p(z_t|x_t, u_1, z_1, \ldots u_t) p(x_t|u_1, z_1 \ldots u_t)$

Markov $= \eta p(z_t|x_t) p(x_t|u_1, z_1 \ldots u_t)$

Total Prob. $= \eta p(z_t|x_t) \int p(x_t|u_1, z_1 \ldots u_t, x_{t-1}) p(x_{t-1}|u_1, z_1 \ldots u_t) dx_{t-1}$

Markov $= \eta p(z_t|x_t) \int p(x_t|u_t, x_{t-1}) p(x_{t-1}|u_1, z_1 \ldots u_t) dx_{t-1}$

Markov $= \eta p(z_t|x_t) \int p(x_t|u_t, x_{t-1}) p(x_{t-1}|u_1, z_1 \ldots z_{t-1}) dx_{t-1}$

$$\boxed{= \eta p(z_t|x_t) \int p(x_t|u_t, x_{t-1}) Bel(x_{t-1}) dx_{t-1}}$$

BF-Q2. Describe the Markov Assumption. In what situations can we make the Markov assumption and when can we not.

Essentially, the Markov Assumption says that the state at the next time step only depends on the current state (not past states) and current input/commands.

Underlying Assumptions
- Perfect model structure w/o approximation errors
- Independent measurement noise
- Random controls

BF-Q3. You recently purchased a robotic vacuum cleaner for your apartment. The vacuum cleaner is able to navigate throughout your home automatically and is supposed to return to its charging station when its battery is low. The robotic system comes pre-programmed with a script that should return it to the docking station, however, you notice that the script only functions part of the time. After running some tests, you determine that the likelihood of of the action "return-to-home" working is $p(x_{t+1} = \text{'home'}|u_{t+1} = \text{'return-to-home'}, x_t = \text{'not-home'}) = 0.7$, where $x_{t+1}$ is the state of the robot after the action is taken, $u_{t+1}$ is the action the robot takes, and $x_t$ is the state of the robot before taking the action. Once returning home, if the return action is taken again, the robot stays at the charging station.

Assuming the robot has a $\overbrace{\text{70\% chance of being 'not-home'}}^{p(x_0=\text{"not home"})}$ when it first detects a low battery. What is the likelihood that its state is 'home' after executing the 'return-to-home' action once time? What is the likelihood if the robot were to execute the 'return-to-home' action once more after executing it the first time?

Write out your steps and label the equations you use.

$bel(x_0 = \text{'home'}) = 0.7$
$bel(x_0 = \text{'not-home'}) = 0.3$ } when first detect low-battery

Determined by law of total probability

$p(x_{t+1} = \text{'home'}|u_t = \text{'return-to-home'}, x_t = \text{'home'}) = 1$
$p(x_{t+1} = \text{'not-home'}|u_t = \text{'return-to-home'}, x_t = \text{'home'}) = 0$

$p(x_{t+1} = \text{'home'}|u_{t+1} = \text{'return-to-home'}, x_t = \text{'not-home'}) = 0.7$
$p(x_{t+1} = \text{'not-home'}|u_{t+1} = \text{'return-to-home'}, x_t = \text{'not-home'}) = 0.3$

Determined by law of total probability

Bayes Filter

$\overline{bel}(x_1 = \text{'home'}) = \sum_{x_0} p(x_1|u_1, x_0) bel(x_0)$

Substitute

$= p(x_1 = \text{'home'}|u_1 = \text{'return-to-home'}, x_0 = \text{'not-home'}) bel(x_0 = \text{'not-home'}) +$
$p(\ldots|u_1 = \text{'return-to-home'}, x_0 = \text{'home'}) bel(x_0 = \text{'home'})$

$x_0$

$= p(x_1 = \text{'home'} \mid u_1 = \text{'return-to-home'}, x_0 = \text{'not-home'})\, bel(x_0 = \text{'not-home'}) +$ Substitute

$p(x_1 = \text{'home'} \mid u_1 = \text{'return-to-home'}, x_0 = \text{'home'})\, bel(x_0 = \text{'home'})$

$= (0.7)(0.7) + (1)(0.3) = 0.79$

$bel(x_1 = \text{'home'}) = \eta\, p(z_1 = \text{'home'} \mid x_1 = \text{'home'})\, \overline{bel}(x_1 = \text{'home'}) = \frac{1}{1}(1)(.79) = \boxed{0.79}$

$\overline{bel}(x_2 = \text{'home'}) = p(x_2 = \text{'home'} \mid u_2 = \text{'return-to-home'}, x_1 = \text{'not-home'})\, bel(x_1 = \text{'not-home'}) +$

$p(x_2 = \text{'home'} \mid u_2 = \text{'return-to-home'}, x_1 = \text{'home'})\, bel(x_1 = \text{'home'})$

via law of Total Probability

$(.7)(1 - .79) + 1(0.79) = \boxed{0.937}$ After 2nd timestep

BF-Q4. A[1] robot is equipped with a manipulator to paint an object. Furthermore, the robot has a sensor to detect whether the object is colored or blank. Neither the manipulation unit nor the sensor are perfect.

From previous experience you know that the robot succeeds in painting a blank object with a probability of

$$p(x_{t+1} = \text{colored} \mid x_t = \text{blank}, u_{t+1} = \text{paint}) = 0.9,$$

where $x_{t+1}$ is the state of the object after executing a painting action, $u_{t+1}$ is the control command, and $x_t$ is the state of the object before performing the action.

The probability that the sensor indicates that the object is colored although it is blank is given by $p(z = \text{colored} \mid x = \text{blank}) = 0.2$, and the probability that the sensor correctly detects a colored object is given by $p(z = \text{colored} \mid x = \text{colored}) = 0.7$.

Unfortunately, you have no knowledge about the current state of the object. However, after the robot performed a painting action the sensor of the robot indicates that the object is colored.

Compute the probability that the object is blank after the robot has performed an action to paint it. Use an appropriate prior distribution and justify your choice.

$p(x_{t+1} = \text{Colored} \mid x_t = \text{blank}, u_{t+1} = \text{paint}) = 0.9$

$p(z = \text{colored} \mid x = \text{blank}) = 0.2$

$p(z = \text{colored} \mid x = \text{colored}) = 0.7$

$bel(x_0 = \text{colored}) = bel(x_0 = \text{blank}) = 0.5$ → Prior distribution that evenly weights all possible states

$\overline{bel}(x_1 = \text{blank}) = p(x_1 = \text{blank} \mid u_1 = \text{paint}, x_0 = \text{blank})\, bel(x_0 = \text{blank}) +$

$p(x_1 = \text{blank} \mid u_1 = \text{paint}, x_0 = \text{colored})\, bel(x_0 = \text{colored})$

$= (0.1)(0.5) + (0)(0.5) = 0.05$

not possible for canvas to go back to blank after being colored

$\overline{bel}(x_1 = \text{colored}) = (.9)(.5) + 1(.5) = .95$

told in problem description

$bel(x_1 = \text{blank}) = \eta\, p(z_1 = \text{colored} \mid x_1 = \text{blank})\, \overline{bel}(x_1 = \text{blank})$

$= \frac{1}{.05 + .95}(.2)(0.05) = \boxed{0.01}$

OGM-Q1. What are the major assumptions of the classic occupancy grid mapping algorithm?

1. Each cell is a <u>binary</u> random variable that can be either free or occupied

OGM-Q1. What are the major assumptions of the classic occupancy grid mapping algorithm?

1. Each cell is a binary random variable that can be either free or occupied
2. The world is static
3. Cells are independent of each other

OGM-Q2. Why do we use log odds notation with occupancy grid mapping.

- Updates become additive instead of multiplicative.
- Also, it never allows a probability to be exactly 0 or 1 avoiding truncation issues

OGM-Q3. Write out a short outline of the steps needed to update an occupancy grid map when a new laser range measurement is received.

Measurement is received
for all cells $m_i$ do
if $m_i$ is in the range of sensor $z_t$
new cell value is equal to previous value + initial belief + new sensor data
else leave cell as is

OGM-Q4. When can the algorithm fail?

Combining two maps from different robots/mapping runs can create conflicts if one map shows an occupied cell while the other shows the opposite. Therefore, a lot of data/attempts are needed.

EC-Q1. State the definition of $E[x+y]$ in terms of the underlying probability distribution, then prove that $E[x+y] = E[x]+E[y]$, using the definition of expectation. Your proof should be succinctly and clearly written. **Do NOT assume that $x$ and $y$ are independent.**

$$E[x+y] = \int_{-\infty}^{\infty} (x+y)\, p(x+y)\, dx\, dy$$

$$= \int_{-\infty}^{\infty} (x+y)[p(x)+p(y)]\, dx\, dy$$

$$= \int_{-\infty}^{\infty} x\, p(x)\, dx + \int_{-\infty}^{\infty} y\, p(y)\, dy$$

$$= E[x] + E[y]$$

EC-Q2. For each of the covariance matrices below, indicate whether it is Valid or Invalid. If it is invalid, give a reason.

Needs to be symmetric + positive semi-definite

$C = C^T$

$x^T C x \geq 0$

$\begin{bmatrix} a \\ b \end{bmatrix} C \begin{bmatrix} a & b \end{bmatrix}$

(a) $\begin{bmatrix} 5 & 3 \\ 3 & 9 \end{bmatrix}$ $(5-\lambda)(9-\lambda)-9 \Rightarrow \lambda^2 - 14\lambda + 38 = 0$, $\lambda \geq 0$ VALID

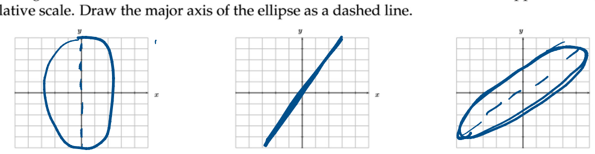
(b) $\begin{bmatrix} 4 & -8 \\ -8 & 9 \end{bmatrix}$ $(4-\lambda)(9-\lambda)-64$, $\lambda^2 - 13\lambda + 28$, $\lambda \geq 0$ VALID

(c) $\begin{bmatrix} 2 & 0 \\ 0 & -8 \end{bmatrix}$ INVALID not positive, semi-definite (has negative values)

(d) $\begin{bmatrix} 3 & 2 \\ 2 & 10 \end{bmatrix}$ $(3-\lambda)(10-\lambda)-4$, $\lambda^2 - 13\lambda + 26$ VALID

(e) $\begin{bmatrix} 6 & 1 \\ 2 & 10 \end{bmatrix}$ Invalid, not symmetric

EC-Q3. Sketch the 1-sigma iso-cost contours for the covariance matrices below. Draw them in approximately correct relative scale. Draw the major axis of the ellipse as a dashed line.



$\begin{bmatrix} 4 & 0 \\ 0 & 8 \end{bmatrix}$ $\begin{bmatrix} 4 & -2 \\ -2 & 8 \end{bmatrix}$ $\begin{bmatrix} 4 & 3.9 \\ 3.9 & 4 \end{bmatrix}$