Bayes filtering: A deeper dive

Sanjiban Choudhury

TAs: Matthew Rockett, Gilwoo Lee, Matt Schmittle
Recap: Key players in a Bayes filter

State
“Hidden stuff we want to know”
(everything needed to predict measurement / effect of action)

Action
“Affects how state evolves”

New state

Measurement
“Some information relevant to state”

State $x_{t-1}$ -> New state $x_t$
Action $u_t$ -> Measurement $z_t$
Today’s objective

1. Work through examples of Bayes filtering

2. Work through derivation

3. Question assumptions along the way
States and beliefs

<table>
<thead>
<tr>
<th>State</th>
<th>Belief</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## States and beliefs

<table>
<thead>
<tr>
<th>State</th>
<th>Belief</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete (Binary)</td>
<td></td>
</tr>
</tbody>
</table>

\[ X = \{X_1, X_2\} \]
States and beliefs

<table>
<thead>
<tr>
<th>State</th>
<th>Belief</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete (Binary)</td>
<td></td>
</tr>
</tbody>
</table>

\[ X = \{X_1, X_2\} \]

![Probability Distribution](image)
## States and beliefs

<table>
<thead>
<tr>
<th>State</th>
<th>Belief</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete (Binary)</td>
<td></td>
</tr>
<tr>
<td>$X = {X_1, X_2}$</td>
<td></td>
</tr>
<tr>
<td>Discrete (More than 2)</td>
<td></td>
</tr>
<tr>
<td>$X = {X_1, X_2, \ldots, X_N}$</td>
<td></td>
</tr>
</tbody>
</table>

![Bar chart showing probabilities of $X_1$ and $X_2$.]
States and beliefs

<table>
<thead>
<tr>
<th>State</th>
<th>Belief</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete (Binary)</td>
<td>$X = {X_1, X_2}$</td>
</tr>
<tr>
<td>Discrete (More than 2)</td>
<td>$X = {X_1, X_2, \ldots, X_N}$</td>
</tr>
</tbody>
</table>
### States and beliefs

<table>
<thead>
<tr>
<th>State</th>
<th>Belief</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Discrete (Binary)</strong></td>
<td></td>
</tr>
<tr>
<td>( X = {X_1, X_2} )</td>
<td></td>
</tr>
<tr>
<td><strong>Discrete (More than 2)</strong></td>
<td></td>
</tr>
<tr>
<td>( X = {X_1, X_2, \ldots, X_N} )</td>
<td></td>
</tr>
<tr>
<td><strong>Continuous</strong></td>
<td></td>
</tr>
<tr>
<td>( X = [a, b] )</td>
<td></td>
</tr>
</tbody>
</table>

\[ P(x) \]

![Bar chart showing discrete and continuous distributions](chart.png)
## States and beliefs

<table>
<thead>
<tr>
<th>State</th>
<th>Belief</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Discrete (Binary)</strong></td>
<td></td>
</tr>
<tr>
<td>$X = {X_1, X_2}$</td>
<td></td>
</tr>
<tr>
<td><strong>Discrete (More than 2)</strong></td>
<td></td>
</tr>
<tr>
<td>$X = {X_1, X_2, \ldots, X_N}$</td>
<td></td>
</tr>
<tr>
<td><strong>Continuous</strong></td>
<td></td>
</tr>
<tr>
<td>$X = [a, b]$</td>
<td></td>
</tr>
</tbody>
</table>

- $P(x)$ is the probability distribution for discrete and continuous states.
- The diagrams illustrate the probability distributions for discrete and continuous states.
API of a general Bayes filter
API of a general Bayes filter

Parameters of the Bayes filter:

Transition model: \( P(x_t | x_{t-1}, u_t) \)

Measurement model: \( P(z_t | x_t) \)
API of a general Bayes filter

Parameters of the Bayes filter:

Transition model: \( P(x_t|x_{t-1}, u_t) \)  
Measurement model: \( P(z_t|x_t) \)

Input to the filter:

Old belief: \( bel(x_{t-1}) \)  
Action: \( u_t \)  
Measurement: \( z_t \)
API of a general Bayes filter

Parameters of the Bayes filter:

Transition model: \( P(x_t|x_{t-1}, u_t) \)

Measurement model: \( P(z_t|x_t) \)

Input to the filter:

Old belief: \( bel(x_{t-1}) \)

Action: \( u_t \)

Measurement: \( z_t \)

Output of the filter:

Updated belief: \( bel(x_t) \)
2 simple steps:

1. Predict belief after \textit{action}

2. Correct belief after \textit{measurement}
Discrete (Binary)
Example 1: Robot opening door
Example 1: Robot opening door
Example 1: Robot opening door
Example 1: Robot opening door

There are two states that we are tracking:

\[ X = \{ \text{Open, Closed} \} \]
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

Our robot can do two actions

\[ A = \{ \text{Pull, Leave} \} \]
Example 1: Robot opening door

There are two states that we are tracking

$$X = \{ \text{Open, Closed} \}$$

Our robot can do two actions

$$A = \{ \text{Pull, Leave} \}$$

We define a transition model (note: our robot is clumsy)

$$P(x_t | x_{t-1}, u_t)$$
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

Our robot can do two actions

A = \{ \text{Pull, Leave} \}

We define a transition model (note: our robot is clumsy)

\[ P(x_t|x_{t-1}, u_t) \]

\[ P(O \mid C, P) = 0.7 \quad P(C \mid C, P) = 0.3 \]

........... and so on
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open}, \text{Closed} \} \]

Our robot can do two actions

\[ A = \{ \text{Pull}, \text{Leave} \} \]
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open}, \text{Closed} \} \]

Our robot can do two actions

\[ A = \{ \text{Pull}, \text{Leave} \} \]

Rewrite the transition model as a matrix

\[
\begin{bmatrix}
P(x_t = \text{O}|x_{t-1} = \text{O}, u_t) & P(x_t = \text{O}|x_{t-1} = \text{C}, u_t) \\
P(x_t = \text{C}|x_{t-1} = \text{O}, u_t) & P(x_t = \text{C}|x_{t-1} = \text{C}, u_t)
\end{bmatrix}
\]
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

Our robot can do two actions

\[ A = \{ \text{Pull, Leave} \} \]

Rewrite the transition model as a matrix

\[
\begin{bmatrix}
P(x_t = O | x_{t-1} = O, u_t) & P(x_t = O | x_{t-1} = C, u_t) \\
P(x_t = C | x_{t-1} = O, u_t) & P(x_t = C | x_{t-1} = C, u_t)
\end{bmatrix}
\]

\[
P(., ., P) = \begin{bmatrix} 0.8 & 0.7 \\ 0.2 & 0.3 \end{bmatrix} \quad P(., ., L) = \begin{bmatrix} 0.5 & 0 \\ 0.5 & 1 \end{bmatrix}
\]
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

Our robot can do two actions

\[ A = \{ \text{Pull, Leave} \} \]

We have a door detector sensor. The sensor is kinda buggy!

\[ Z = \{ \text{Open, Closed} \} \]

\[ P(z_t \mid x_t) \]

.... let’s use our matrix format
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

\[ A = \{ \text{Pull, Leave} \} \]

\[ Z = \{ \text{Open, Closed} \} \]

Rewrite the measurement model as a vector

\[
\begin{bmatrix}
P(z_t | O) \\
P(z_t | C)
\end{bmatrix}
\]

\[ P(O | .) = \begin{bmatrix} 0.6 \\ 0.2 \end{bmatrix} \]

\[ P(C | .) = \begin{bmatrix} 0.4 \\ 0.8 \end{bmatrix} \]
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

\[ A = \{ \text{Pull, Leave} \} \]

\[ Z = \{ \text{Open, Closed} \} \]

Let’s get ready to Bayes filter!
Example 1: Robot opening door

There are two states that we are tracking:

\[ X = \{ \text{Open, Closed}\} \]
\[ A = \{ \text{Pull, Leave}\} \]
\[ Z = \{ \text{Open, Closed}\} \]

Step 0. Start with the belief at time step t-1

\[ \text{bel}(x_{t-1}) = \begin{bmatrix} 0.4 \\ 0.6 \end{bmatrix} \]

Robot thinks the door is open with 0.4 probability.
Example 1: Robot opening door

There are two states that we are tracking:

\[ X = \{ \text{Open}, \text{Closed}\} \]

\[ A = \{ \text{Pull}, \text{Leave}\} \]

\[ Z = \{ \text{Open}, \text{Closed}\} \]

Robot executes action Pull
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]
\[ A = \{ \text{Pull, Leave} \} \]
\[ Z = \{ \text{Open, Closed} \} \]

Step 1: Prediction - push belief through dynamics given action

\[
\overline{bel}(x_t) = \sum_{x_{t-1}} P(x_t | x_{t-1}, u_t) \overline{bel}(x_{t-1})
\]
**Example 1: Robot opening door**

There are two states that we are tracking:

\[
X = \{ \text{Open, Closed} \}
\]

\[
A = \{ \text{Pull, Leave} \}
\]

\[
Z = \{ \text{Open, Closed} \}
\]

**Step 1: Prediction - push belief through dynamics given action**

\[
\begin{bmatrix}
P(x_t = O) \\
P(x_t = C)
\end{bmatrix}
= \begin{bmatrix}
P(x_t = O | x_{t-1} = O, u_t) \\
P(x_t = C | x_{t-1} = O, u_t)
\end{bmatrix}
\begin{bmatrix}
P(x_{t-1} = O) \\
P(x_{t-1} = C)
\end{bmatrix}
\]

\[
\overline{bel}(x_t)
\]

\[
bel(x_{t-1})
\]
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open}, \text{Closed} \} \]

\[ A = \{ \text{Pull}, \text{Leave} \} \]

\[ Z = \{ \text{Open}, \text{Closed} \} \]

Step 1: Prediction - push belief through dynamics given action

\[
\begin{bmatrix}
0.74 \\
0.26
\end{bmatrix} =
\begin{bmatrix}
0.8 & 0.7 \\
0.2 & 0.3
\end{bmatrix}
\begin{bmatrix}
0.4 \\
0.6
\end{bmatrix}
\]

\[ \overline{bel}(x_t) = P(.|., P) \overline{bel}(x_{t-1}) \]

Robot thinks the door is open with 0.74 probability
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

\[ A = \{ \text{Pull, Leave} \} \]

\[ Z = \{ \text{Open, Closed} \} \]

Robot receives measurement

Closed
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

\[ A = \{ \text{Pull, Leave} \} \]

\[ Z = \{ \text{Open, Closed} \} \]

Step 2: Correction - apply Bayes rule given measurement

\[ \text{bel}(x_t) = \eta P(z_t | x_t) \text{bel}(x_t) \]

(normalize)
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

\[ A = \{ \text{Pull, Leave} \} \]

\[ Z = \{ \text{Open, Closed} \} \]

Step 2: Correction - apply Bayes rule given measurement

\[
\begin{bmatrix}
P(x_t = \text{O}) \\
P(x_t = \text{C})
\end{bmatrix}
= \eta
\begin{bmatrix}
P(z_t | \text{O}) \\
P(z_t | \text{C})
\end{bmatrix}
* 
\begin{bmatrix}
P(x_t = \text{O}) \\
P(x_t = \text{C})
\end{bmatrix}
\]

\[ \text{bel}(x_t) \]

\[ P(C \mid .) \]

\[ \overline{\text{bel}}(x_t) \]

element wise
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]
\[ A = \{ \text{Pull, Leave} \} \]
\[ Z = \{ \text{Open, Closed} \} \]

Step 2: Correction - apply Bayes rule given measurement

\[
\begin{bmatrix}
P(x_t = \text{O}) \\
P(x_t = \text{C})
\end{bmatrix}
= \eta
\begin{bmatrix}
0.4 \\
0.8
\end{bmatrix}
\begin{bmatrix}
0.74 \\
0.26
\end{bmatrix}
\]

\[ \text{bel}(x_t) \]
\[ \overline{\text{bel}}(x_t) \]
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]

\[ A = \{ \text{Pull, Leave} \} \]

\[ Z = \{ \text{Open, Closed} \} \]

Step 2: Correction - apply Bayes rule given measurement

\[
\begin{bmatrix}
P(x_t = O) \\
P(x_t = C)
\end{bmatrix}
\mathrel{\sim}
\eta
\begin{bmatrix}
0.4 & 0.74 \\
0.8 & 0.26
\end{bmatrix}
\mathrel{\sim}
\eta
\begin{bmatrix}
0.296 \\
0.208
\end{bmatrix}
\mathrel{\sim}
\begin{bmatrix}
0.58 \\
0.42
\end{bmatrix}
\]

\[ \overline{\text{bel}}(x_t) \]

\[ \text{bel}(x_t) \]
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed} \} \]
\[ A = \{ \text{Pull, Leave} \} \]
\[ Z = \{ \text{Open, Closed} \} \]

Step 2: Correction - apply Bayes rule given measurement

\[ bel(x_t) = \begin{bmatrix} 0.58 \\ 0.42 \end{bmatrix} \]

Robot thinks the door is open with 0.58 probability
Example 1: Robot opening door

There are two states that we are tracking

\[ X = \{ \text{Open, Closed}\} \]
\[ A = \{ \text{Pull, Leave}\} \]
\[ Z = \{ \text{Open, Closed}\} \]

Let’s summarize

Robot thought the door is open with 0.4 probability

Robot executed \text{Pull} action.
Robot thinks the door is open with 0.74 probability

Robot got \text{Closed} measurement.
Robot thinks the door is open with 0.58 probability
Continuous (Non-parametric)
Bayes filter in a nutshell

Step 0. Start with the belief at time step t-1
\[
\text{bel}(x_{t-1})
\]

Step 1: Prediction - push belief through dynamics given action
\[
\overline{\text{bel}}(x_t) = \int P(x_t|u_t, x_{t-1})\text{bel}(x_{t-1})dx_{t-1}
\]

Step 2: Correction - apply Bayes rule given measurement
\[
\text{bel}(x_t) = \eta P(z_t|x_t)\overline{\text{bel}}(x_t)
\]
Robot lost in a 1-D hallway

$bel(x_{t-1})$
Action at time $t$: NOP

$u_t = \text{NOP}$

$P(x_t|u_t, x_{t-1}) = \delta(x_t = x_{t-1})$

$\overline{\text{bel}}(x_t) = \int P(x_t|u_t, x_{t-1})\text{bel}(x_{t-1})dx_{t-1} = \text{bel}(x_t)$

NOP action implies belief remains the same!
Measurement at time $t$: “Door”

$z_t = \text{Door}$

$P(z_t|x_t) = \mathcal{N}(\text{door centre, } 0.75m)$
Measurement at time $t$: “Door”

$z_t = \text{Door}$

$P(z_t | x_t) = \mathcal{N}(\text{door centre}, 0.75m)$

$P(z_t | x_t)$

$\text{bel}(x_t) = \eta P(z_t | x_t) \overline{\text{bel}}(x_t)$
Action at time $t+1$: Move 3m right

$u_{t+1} = 3\text{m right}$

$P(x_{t+1}|u_{t+1}, x_t) = \mathcal{N}(x_t + u_{t+1}, 0.25\text{m})$

$\overline{bel}(x_{t+1}) = \int P(x_{t+1}|u_{t+1}, x_t)\overline{bel}(x_t) dx_t$
Measurement at time $t+1$: “Door”

$z_{t+1} = \text{Door}$

$P(z_{t+1} | x_{t+1}) = \mathcal{N} \text{(door centre, 0.75m)}$
Measurement at time $t+1$: “Door”

$z_{t+1} = \text{Door}$

$P(z_{t+1}|x_{t+1}) = \mathcal{N}(\text{door centre, } 0.75m)$

$$P(z_{t+1}|x_{t+1})$$

$$bel(x_{t+1}) = \eta P(z_{t+1}|x_{t+1})\overline{bel}(x_{t+1})$$
Questions
Questions

Do actions always increase uncertainty?
Questions

Do actions always increase uncertainty?

\[ u_t \]

\[ \text{bel}(x_{t-1}) \]

\[ \text{bel}(x_t) \]
Questions

Do actions always increase uncertainty?

Do measurements always reduce uncertainty?
Questions

Do actions always increase uncertainty?

\[ \text{Do measurements always reduce uncertainty?} \]

(What happens when you reach into your bag and don’t find your keys?
Example of a negative measurement)
Bayes derivation

Past data

\[ x_{t-1} \rightarrow x_t \]

\[ u_t \]

\[ z_t \]
Bayes derivation

\[ bel(x_t) \]
Bayes derivation

\[ \text{bel}(x_t) = P(x_t | z_{1:t-1}, u_{1:t-1}, z_t, u_t) \]
$bel(x_t) = P(x_t | z_{1:t-1}, u_{1:t-1}, z_t, u_t)$

(Bayes) $= \eta P(z_t | x_t, z_{1:t-1}, u_{1:t-1}, u_t) P(x_t | z_{1:t-1}, u_{1:t-1}, u_t)$
Bayes derivation

\[ bel(x_t) = P(x_t | z_{1:t-1}, u_{1:t-1}, z_t, u_t) \]

(Bayes) \quad = \eta P(z_t | x_t, z_{1:t-1}, u_{1:t-1}, u_t) \ P(x_t | z_{1:t-1}, u_{1:t-1}, u_t) 

(Markov) \quad = \eta P(z_t | x_t) \ P(x_t | z_{1:t-1}, u_{1:t-1}, u_t) \]
Bayes derivation

\[ \text{bel}(x_t) = P(x_t | z_{1:t-1}, u_{1:t-1}, z_t, u_t) \]

(Bayes) \[ = \eta P(z_t | x_t, z_{1:t-1}, u_{1:t-1}, u_t) P(x_t | z_{1:t-1}, u_{1:t-1}, u_t) \]

(Markov) \[ = \eta P(z_t | x_t) P(x_t | z_{1:t-1}, u_{1:t-1}, u_t) \]

\[ = \eta P(z_t | x_t) \text{bel}(x_t) \]
Bayes derivation
Bayes derivation

\[ \text{bel}(x_t) = P(x_t | z_{1:t-1}, u_{1:t-1}, u_t) \]
Bayes derivation

$$\text{bel}(x_t) = P(x_t|z_{1:t-1}, u_{1:t-1}, u_t)$$

(Total prob.) $$= \int P(x_t|x_{t-1}, z_{1:t-1}, u_{1:t-1}, u_t) \, P(x_{t-1}|z_{1:t-1}, u_{1:t-1}, u_t) \, dx_{t-1}$$
Bayes derivation

\[ \text{bel}(x_t) = P(x_t|z_{1:t-1}, u_{1:t-1}, u_t) \]

(Total prob.) \[ = \int P(x_t|x_{t-1}, z_{1:t-1}, u_{1:t-1}, u_t) P(x_{t-1}|z_{1:t-1}, u_{1:t-1}, u_t) dx_{t-1} \]

(Markov) \[ = \int P(x_t|x_{t-1}, u_t) P(x_{t-1}|z_{1:t-1}, u_{1:t-1}, u_t) dx_{t-1} \]
Bayes derivation

\[ \text{bel}(x_t) = P(x_t | z_{1:t-1}, u_{1:t-1}, u_t) \]

(Total prob.) \[ = \int P(x_t | x_{t-1}, z_{1:t-1}, u_{1:t-1}, u_t) P(x_{t-1} | z_{1:t-1}, u_{1:t-1}, u_t) dx_{t-1} \]

(Markov) \[ = \int P(x_t | x_{t-1}, u_t) P(x_{t-1} | z_{1:t-1}, u_{1:t-1}, u_t) dx_{t-1} \]

(Cond. indep.) \[ = \int P(x_t | x_{t-1}, u_t) P(x_{t-1} | z_{1:t-1}, u_{1:t-1}) dx_{t-1} \]
Bayes derivation

\[
\overline{\text{bel}}(x_t) = P(x_t | z_{1:t-1}, u_{1:t-1}, u_t)
\]

(Total prob.) \[= \int P(x_t | x_{t-1}, z_{1:t-1}, u_{1:t-1}, u_t) \, P(x_{t-1} | z_{1:t-1}, u_{1:t-1}, u_t) \, dx_{t-1}\]

(Markov) \[= \int P(x_t | x_{t-1}, u_t) \, P(x_{t-1} | z_{1:t-1}, u_{1:t-1}, u_t) \, dx_{t-1}\]

(Cond. indep) \[= \int P(x_t | x_{t-1}, u_t) \, P(x_{t-1} | z_{1:t-1}, u_{1:t-1}) \, dx_{t-1}\]

\[= \int P(x_t | x_{t-1}, u_t) \, \overline{\text{bel}}(x_{t-1}) \, dx_{t-1}\]
After thoughts ...
Question: When is cond. independence not true?

i.e. when can you tell something about the past based on future data?

E.g. Motion capture data of a human. Human knows the true state and generate control actions accordingly.
Question: When is cond. independence not true?

\[ = \int P(x_t|x_{t-1}, u_t) P(x_{t-1}|z_{1:t-1}, u_{1:t-1}, u_t) dx_{t-1} \]

i.e. when can you tell something about the past based on future data?

E.g. Motion capture data of a human. Human knows the true state and generate control actions accordingly.
Question: When is cond. independence not true?

\[
= \int P(x_t|x_{t-1}, u_t) \, P(x_{t-1}|z_{1:t-1}, u_{1:t-1}, u_t) \, dx_{t-1}
\]

(Cond. ind) \[
= \int P(x_t|x_{t-1}, u_t) \, P(x_{t-1}|z_{1:t-1}, u_{1:t-1}) \, dx_{t-1}
\]

i.e. when can you tell something about the past based on future data?

E.g. Motion capture data of a human.
Human knows the true state and generate control actions accordingly.
Bayes filter in a single line

\[ P(x_t|x_{t-1}, u_t) \quad \text{Motion model} \]

\[ P(z_t|x_t) \quad \text{Measurement model} \]

\[ \text{bel}(x_t) = \eta P(z_t|x_t) \int P(x_t|x_{t-1}, u_t) \text{bel}(x_{t-1}) dx_{t-1} \]

Note that order does not really matter - we can flip measurement and control.
Asynchronous streaming version of Bayes

Input: Datapoint $d$, Current belief $bel(x)$

Output: Updated belief $bel^+(x)$

Process:

If $d$ is measurement $z$ then
for all $x$

$$bel^+(x) = P(z|x)bel(x)$$
$$bel^+(x) = \text{Normalize}(bel^+(x))$$

Else if $d$ is control $u$ then
for all $x$

$$bel^+(x) = \sum_{x_{old}} P(x|x_{old}, u)bel(x_{old})$$

Return $bel^+(x)$
Things to keep in mind...
Things to keep in mind...

1. Bayes filter can be overconfident

Once belief collapses to 0/1 only motion model can shake it loose
Things to keep in mind...

1. Bayes filter can be overconfident

Once belief collapses to 0/1 only motion model can shake it loose

2. Too many measurements will collapse belief
Things to keep in mind…

1. Bayes filter can be overconfident

Once belief collapses to 0/1 only motion model can shake it loose

2. Too many measurements will collapse belief

3. Correlated incorrect measurements are dangerous
Bayes filter is a powerful tool

Localization
Mapping
SLAM
POMDP