

Lecture #6: *Bayesian and Kalman Filters*

Scribes: *Eric Ewing, Connie Zhang, Kegan Strawn, Pushpreet Singh Hanspal*
CSCI 545 Introduction to Robotics

Notes Taken September 16, 2019

Contents

1 Introduction	1
2 Bayes Theorem	1
3 Discrete Bayes Filter	4
4 Kalman Filter	8
5 References, Resources, and Further Readings	10

1 Introduction

The previous lecture (5) covered Bayesian networks, the Markov assumption, linear dynamical systems, and control strategies. In this lecture we use these previous topics and some probability theory from lecture three to introduce and provide an example of the Bayes filtering algorithm. At the end, another filtering algorithm called the Kalman filter is introduced, however, the algorithm is covered in a following lecture. The inspiration behind filtering algorithms is to model our belief of the state based on only our previous state, some observation, and an input leading to a transition while accounting for some error in our model.

2 Bayes Theorem

For a more comprehensive review of probability theory and Bayesian networks, see the scribe notes for the third and fifth lectures. The following is a quick review to prepare for Bayesian filters.

2.1 Bayes Theorem Understanding

Bayes theorem provides a mathematical formula for a conditional probability problem. The sort of probability we are working with here could also be thought of as a belief. If we are flipping a coin, we might hold the belief that a coin landing face up as heads or tails as 50% each. First, let's quickly review what we will need to work with Bayes Theorem by working through an example.

Let's say you are working in the lab where half of the researchers work on aerial vehicles and the other half of researchers work on ground vehicles, and a box arrives. This box has a company logo on it for an aerospace company. You might reasonably guess that this box contains aerial vehicle supplies. However, if you then notice that the box has been placed on one of the ground vehicle researcher's desks this would likely change your guess. Bayes theorem gives us a formula to calculate these guesses given previous knowledge.

2.1.1 Basic Probability

So, if 50% of the researchers work on aerial vehicles and 50% work on ground vehicles, it is reasonable to say that 50% of boxes arriving at this lab contain supplies for aerial researchers and 50% contain supplies for ground researchers:

$$P(\text{Aerial Supplies}) = \frac{\text{aerial}}{\text{total researchers}} \\ = 0.50$$

$$P(\text{Ground Supplies}) = \frac{\text{ground}}{\text{total researchers}} \\ = 0.50$$

Now, let's say the probability that a box on a ground vehicle researcher's desk contains each type of supply as being different from the general probability of a box in the lab:

$$P(\text{Aerial Supplies}) = \frac{\text{aerial}}{\text{total desk supplies}}$$

$$P(\text{Aerial Supplies}) = 0.02$$

$$P(\text{Ground Supplies}) = \frac{\text{ground}}{\text{total desk supplies}}$$

$$P(\text{Ground Supplies}) = 0.98$$

Here is where the conditional probability begins. We need to take into account what the probability of having a package from the aerospace company is for both ground and aerial researchers.

$$P(\text{Aerospace Company Logo} \mid \text{Aerial Supplies}) = \frac{\text{aerial supplies with company logo}}{\text{aerial supplies}}$$

$$P(\text{Aerospace Company Logo} \mid \text{Aerial Supplies}) = 0.50$$

$$P(\text{Aerospace Company Logo} \mid \text{Ground Supplies}) = \frac{\text{ground supplies with company logo}}{\text{ground supplies}}$$

$$P(\text{Aerospace Company Logo} \mid \text{Ground Supplies}) = 0.04$$

Also, we can deal with joint probabilities. For example, what is the probability that the box contains aerial supplies and has the aerospace company logo? Using our previous conditional

probability we can see that it is 25%.

$$\begin{aligned}
 P(\text{Aerial Supplies With Company Logo}) &= P(\text{Aerial Supplies} \cap \text{Company Logo}) \\
 P(\text{Aerial Supplies With Company Logo}) &= P(\text{Aerial Supplies}) \\
 &\quad * P(\text{Aerospace Company Logo} \mid \text{Aerial Supplies}) \\
 P(\text{Aerial Supplies With Company Logo}) &= (0.5)(0.5) = 0.25
 \end{aligned}$$

$$\begin{aligned}
 P(\text{Ground Supplies} \cap \text{Company Logo}) &= P(\text{Ground Supplies}) \\
 &\quad * P(\text{Aerospace Company Logo} \mid \text{Ground Supplies}) \\
 P(\text{Ground Supplies} \cap \text{Company Logo}) &= (0.5)(0.04) = 0.02
 \end{aligned}$$

Finally, to solve our example question. We can calculate the probability of ground supplies with the company logo and separately the probability of the box having the company logo and containing ground supplies, which we can then work with algebraically to get:

$$\begin{aligned}
 P(\text{Ground Supplies} \mid \text{Company Logo}) &= \frac{P(\text{Ground Supplies})P(\text{Company Logo} \mid \text{Ground Supplies})}{P(\text{Company Logo})} \\
 &= \frac{(0.50)(0.04)}{(0.50)(0.04) + P(\text{Aerial} \cap \text{Company Logo})} \\
 P(\text{Ground Supplies} \mid \text{Company Logo}) &= \frac{(0.50)(0.04)}{(0.02) + (0.25)} = \frac{0.02}{0.27} = 0.07
 \end{aligned}$$

So, this shows us that the probability of a box containing ground supplies given that it has an aerospace company logo on it is estimated to be 7%. However, we want to know the probability it has ground supplies given it has an aerospace company logo on it, but is on a ground vehicle researcher's desk. Note that now instead of using the probabilities of ground and aerial supplies in the lab as a whole, we use the probabilities of ground and aerial supplies on a ground vehicle researcher's desk:

$$P(\text{Ground Supplies} \mid \text{Company Logo}) = \frac{(0.98)(0.04)}{(0.01) + (0.04)} = \frac{0.04}{0.05} = 0.80$$

2.2 Bayes Theorem Formula

As we can see Bayes Theorem here has given us the ability to calculate this conditional probability and to see how our belief in the probability of something changes given some new piece of information.

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

The way in which we use this formula might often be thought of more as updating our belief $P(A|B)$ by using the prior belief $P(A)$ and the likelihood of the new information $P(B|A)$.

3 Discrete Bayes Filter

The Bayes filter is a framework for recursive state estimation that utilizes the Bayes theorem, Markov assumption, probability theory, and Bayesian networks to do so. A Bayes filter allows you to estimate a probability density function of states over time using observations.

3.1 State Estimation

We will generally use Bayes Filtering for robotics state estimation. State estimation is attempting to determine the current state given the history of observations and inputs. We are trying to update our belief of the current state based on our previous beliefs and observations. We will define our system with the following variables:

- x_t : the state at a time t
- z_t : the observation at a time t
- u_t : the input at a time t

Recall from the Bayesian networks section that the order of these events is: an input u_t is made, the state updates to x_t , and finally an observation z_t is made.

3.1.1 Markov Assumption

For any event y_t , which depends on previous events y_1, \dots, y_{t-1} , the Markov property is as follows:

$$p(y_t | y_1, \dots, y_{t-1}) = p(y_t | y_{t-1})$$

The Markov Assumption provides a way of simplifying our equations and computations by assuming that the current state depends only on the previous state and the previous input, given as:

$$p(x_t | x_0, u_1, \dots, x_{t-1}, u_t) = p(x_t | x_{t-1}, u_t)$$

The observation function under the Markov assumption states that the probability of getting an observation at a given time depends only on the previous state, given as

$$p(z_t | x_1, \dots, x_t) = p(z_t | x_t)$$

Note that the Markov property assumes:

1. A static world
2. Independent noise
3. A perfect model

3.1.2 Recursive Belief Equation

Referring to the previous lectures and specifically the third lecture notes, we can see that the belief function can be transformed using Bayes rule for three variables and the Markov assumption into a recursive equation for our new belief:

$$\begin{aligned} \text{bel}(x_t) &= p(x_t | u_{1:t}, z_{1:t}) \\ &= \eta p(z_t | x_t) \int_{-\infty}^{\infty} p(x_t | x_{t-1}, u_t) \text{bel}(x_{t-1}) dx_{t-1} \end{aligned}$$

3.2 Bayes Filter Algorithm

Using the recursive equation there are two general steps for a Bayes filter, the prediction and correction steps.

3.3 Bayes Filter Algorithm Steps

1. The Prediction Step

$$\bar{bel}(x_t) = \sum p(x_t|u_t, x_{t-1}) bel(x_{t-1})$$

2. The Correction Step

$$bel(x_t) = \eta p(z_t|x_t) \bar{bel}(x_t)$$

The prediction step contains our transition (or motion) model:

$$p(x_t|u_t, x_{t-1})$$

This prediction model states a probability that an action brings the state from the previous state to the current state. The correction step contains our observation (or sensor) model:

$$p(z_t|x_t)$$

This observation model states a probability of an observation given the current state independently.

3.4 The Bayes Filter Algorithm

```

function BAYESFILTER( $b(x_{t-1}), u_t, z_t$ )
  for all  $x_t$  do
     $\bar{b}(x_t) = \sum p(x_t|u_t, x_{t-1})b(x_{t-1})$ 
     $b(x_t) = \eta p(z_t|x_t)\bar{b}(x_t)$ 
  end for
end function

```

3.5 Bayes Filter Robot Example

Let's explore the Bayes filter algorithm by working through an example. Let's imagine there is a robot with a camera attempting to go through a door.

3.5.1 Initial Setup

We could define our variable states as:

x: {door.open, door.closed}
u: {do.nothing, do.push}
z: {sense.open, sense.closed}

And let's assume a uniform distribution with probabilities at the initial time step of:

$$p(x_0 = \text{door.open}) = 0.50$$

$$p(x_0 = \text{door.closed}) = 0.50$$

3.5.2 Example Behavior

For our example let's compute how the belief changes over two time steps. To do so we can arbitrarily select our inputs to send to the robot and what observations were observed for the sake of this example.

$$u_1 = do.nothing$$

$$u_2 = do.push$$

$$z_1 = sense.open$$

$$z_2 = sense.open$$

So, our robot will do nothing but observe an open door on the first time step and then on the second time step it will take in an input/action of pushing and observe, again, an open door.

3.5.3 Finding Our Models

First, we need to find our models to later work with in our Bayes Algorithm. This is given information we are obtaining through testing, other resources, or in our case making up.

3.5.4 Sensor Model

Let's say the probability of sensing an open door given that the door is open is 60%.

$$p(z_t = sense.open \mid x_t = door.open) = 0.60$$

Thus, the probability for sensing a closed door given it is open should be 40%. This makes sense as the sensor could have a range of error, detecting a closed door even when the door is open.

$$p(z_t = sense.closed \mid x_t = door.open) = 0.40$$

However, when the door is closed we should have a lower error rate. In other words, it is more likely to detect a closed door. Also, detecting a closed door as being open is probably less likely than detecting an open door as being closed. This could be represented as such:

$$p(z_t = sense.open \mid x_t = door.closed) = 0.20$$

$$p(z_t = sense.closed \mid x_t = door.closed) = 0.80$$

If the sensor was perfect these values might be one and zero, or at least closer to one and zero, however, we are working with some uncertainty in our sensor model.

3.5.5 Transition Model

Let's define the probability of the door being open given that the door was previously open and the input was to do nothing as being 100%. Conversely, the probability of the door being closed given that it was previously believed to be open and nothing was done as 0%.

$$p(x_t = door.open \mid x_{t-1} = door.open, u_t = do.nothing) = 1.00$$

$$p(x_t = door.closed \mid x_{t-1} = door.open, u_t = do.nothing) = 0.00$$

We can do the same for when the previous belief is that the door was closed.

$$\begin{aligned} p(x_t = \text{door.open} \mid x_{t-1} = \text{door.closed}, u_t = \text{do.nothing}) &= 0.00 \\ p(x_t = \text{door.closed} \mid x_{t-1} = \text{door.closed}, u_t = \text{do.nothing}) &= 1.00 \end{aligned}$$

And now we need to explore what happens when the robot pushes or does not push. If the door is open on the previous state and we push, we will assume 100% probability in the door continuing to be open. However, if the door was closed on the previous state and we push, we will assume that it is likely the push succeeded to now open the door, but that there is a chance it might have failed. For this, let's assume 80% and 20% respectively.

$$\begin{aligned} p(x_t = \text{door.open} \mid x_{t-1} = \text{door.open}, u_t = \text{do.push}) &= 1.00 \\ p(x_t = \text{door.closed} \mid x_{t-1} = \text{door.open}, u_t = \text{do.push}) &= 0.00 \\ p(x_t = \text{door.open} \mid x_{t-1} = \text{door.closed}, u_t = \text{do.push}) &= 0.80 \\ p(x_t = \text{door.closed} \mid x_{t-1} = \text{door.closed}, u_t = \text{do.push}) &= 0.20 \end{aligned}$$

The probabilities are the same regardless of the time step. They represent the observations and transitions that the state might take which we will need to use in our Bayes filter.

3.5.6 Running the Bayesian Filter Algorithm

Now we can use the above probabilities and the two steps in the Bayes filter algorithm to find our beliefs at the two time steps for the robot trying to go through a door.

$$\begin{aligned} t = 1 \\ \overline{bel}(x_1) &= \sum p(x_1 \mid u_1, x_0) bel(x_0) \\ &= p(x_1 \mid u_1 = \text{do.nothing}, x_0 = \text{door.open}) bel(x_0 = \text{door.open}) \\ &\quad + p(x_1 \mid u_1 = \text{do.nothing}, x_0 = \text{door.closed}) bel(x_0 = \text{door.closed}) \end{aligned}$$

Above we took the summation of all possible previous states for which we have two of (the initial possible states). The door could potentially be open or closed and for our input we had specified earlier that the first action would be to do nothing on time step 1. We can now calculate this using our probabilities from our transition and observation models:

$$\begin{aligned} t = 1 \\ \overline{bel}(x_1 = \text{door.open}) &= (1)(0.50) + (0.00)(0.50) = 0.50 \\ \overline{bel}(x_1 = \text{door.closed}) &= (0)(0.50) + (1.00)(0.50) = 0.50 \end{aligned}$$

That completes the prediction step, so now we can compute the correction step:

$$\begin{aligned}
 t &= 1 \\
 \text{bel}(x_1 = \text{door.open}) &= \eta p(z_1 = \text{sense.open} | x_1 = \text{door.open}) \overline{\text{bel}}(x_1 = \text{door.open}) \\
 &= \eta (0.60)(0.50) \\
 &= (0.3) \eta
 \end{aligned}$$

$$\begin{aligned}
 \text{bel}(x_1 = \text{door.closed}) &= \eta p(z_1 = \text{sense.open} | x_1 = \text{door.closed}) \overline{\text{bel}}(x_1 = \text{door.closed}) \\
 &= \eta (0.20)(0.50) \\
 &= (0.1) \eta
 \end{aligned}$$

Before we finish we need the probabilities to sum to one to uphold the correct probability properties:

$$\begin{aligned}
 \text{bel}(x_1 = \text{door.closed}) + \text{bel}(x_1 = \text{door.open}) &= 1 \\
 0.30\eta + 0.10\eta &= 1 \\
 \eta &= 2.5
 \end{aligned}$$

Using this we can now find that our beliefs are:

$$\begin{aligned}
 t &= 1 \\
 \text{bel}(x_1 = \text{door.open}) &= 0.75 \\
 \text{bel}(x_1 = \text{door.closed}) &= 0.25
 \end{aligned}$$

We can repeat this for the second time step:

$$\begin{aligned}
 \overline{\text{bel}}(x_2 = \text{door.open}) &= (1.00)(0.75) + (0.80)(0.25) = 0.95 \\
 \overline{\text{bel}}(x_2 = \text{door.closed}) &= 0.05 \\
 \text{bel}(x_2 = \text{door.open}) &= 0.983 \\
 \text{bel}(x_2 = \text{door.closed}) &= 0.017
 \end{aligned}$$

As we can see after one time step it is likely that the door is open. We reserve some probability to the door being closed and potentially our sensors are off. However, after a second time step with a pushing input we are even more certain in our belief of the state that the door is open by using this discrete Bayesian filter algorithm.

4 Kalman Filter

A Kalman filter allows us to model any linear system accurately by linearizing the predictions and observations about their mean. For nonlinear systems we can use the extended Kalman filter, but for now let's focus on the basic Kalman filter.

4.1 Kalman Filter Background

A Kalman filter is a special case of the Bayes filter where the dynamics and sensory model is linear Gaussian. Kalman filters are used where there is uncertain information about a dynamic system and you need to make a guess or form a belief about what the system will do next. They do not store more than the previous state, and are defined for continuously changing systems only.

4.2 Kalman Filter Model

A Kalman filter can be modeled with a state transition probability $p(x_t|u_t, x_{t-1})$ and a measurement probability $p(z_t|x_t)$, both must be linear¹ for every iteration to be Gaussian.

$$\begin{aligned}x_t &= Ax_{t-1} + Bu_t + \varepsilon \\z_t &= cx_t + \delta\end{aligned}$$

Where A is a $n \times n$ linear transformation prediction matrix, x is a $n \times 1$ vector describing the state, B is a $n \times m$ control matrix, u is a $m \times 1$ matrix of the input, and C is a transformation matrix for the observation. ε and δ are random normal Gaussian distributions with zero means with R_t and Q covariance matrices.

$$\begin{aligned}\varepsilon &\sim N(0, R_t) \\ \delta &\sim N(0, Q)\end{aligned}$$

Recall that the definition of a normal multivariate Gaussian distribution gives us:

$$p(x) = 2\pi^{-\frac{k}{2}} |\Sigma|^{-\frac{1}{2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}$$

So, for our Kalman filter the probability equations can be written as such:

$$\begin{aligned}p(x_t|x_{t-1}, u_t) &= 2\pi^{-\frac{k}{2}} |R|^{-\frac{k}{2}} e^{-\frac{1}{2}(x_t - Ax_{t-1} - Bu_t)^T R^{-1}(x_t - Ax_{t-1} - Bu_t)} \\ p(z_t|x_t) &= 2\pi^{-\frac{k}{2}} |Q|^{-\frac{1}{2}} e^{-\frac{1}{2}(z_t - x_t)^T Q^{-1}(z_t - x_t)}\end{aligned}$$

This is as far into the Kalman Filter as we got in this lecture. For more info on the algorithm and a numerical example, see the following day's scribe.

¹All functions of the form $f(x) = Ax$ with A as a $m \times n$ matrix and x as a $n \times 1$ vector are linear with respect to x . Functions like $f(x) = x^T Ax$ are not linear in x .

5 References, Resources, and Further Readings

- Conditional probability notes: <http://www.stat.yale.edu/Courses/1997-98/101/condprob.htm>
- Lecture slides on Bayesian Filtering: <https://people.eecs.berkeley.edu/~pabbeel/cs287-fa13/slides/bayes-filters.pdf>
- More lecture slides on Bayesian Filtering: <http://ais.informatik.uni-freiburg.de/teaching/ws12/mapping/pdf/slam02-bayes-filter-short.pdf>
- Lecture slides on Kalman Filtering http://biorobotics.ri.cmu.edu/papers/sbp_papers/integrated3/kleeman_kalman_basics.pdf
- More lecture slides on Kalman Filtering <https://stanford.edu/class/ee363/lectures/kf.pdf>
- Article on Kalman Filtering <https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/>