Recursive State Estimation

2.1 Introduction

⇒ At the core of probabilistic robotics is the idea of estimating state from sensor data.

"estimating quantities from sensor data that are not directly observable, but can be inferred"

⇒ The goal of this chapter is to introduce the basic vocabulary & mathematical tools for estimating state from sensor data.

2.2 Basic Concepts in Probability.

⇒ Let X denote a random variable and x denote a specific event that X might take on.

⇒ $p(X=x)$, denotes probability that the random variable has value x.

⇒ Discrete probabilities sums to one, that is,

$$\boxed{\sum_x p(X=x) = 1}$$

⇒ Probabilities are always non-negative.

⇒ We usually omit explicit mention

$$p(X=x) \iff p(x)$$

⇒ Continuous spaces are characterized by random variables that can take on a continuum of values.

⇒ We assume that all continuous random variables possess probability density function (PDFs).

⇒ A common density function is that of the one-dimensional <u>normal distribution</u> with <u>mean μ</u> & <u>variance σ^2</u>.

↳ This distribution is given by the following <u>Gaussian function</u>:

$$p(x) = (2\pi\sigma^2)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\}$$

$$\mu = \frac{\int_{x_{min}}^{x_{max}} p(x)\,dx}{\int_{x_{min}}^{x_{max}} dx}$$

{Average value}

$$\sigma^2 = \int_{x_{min}}^{x_{max}} (x-\mu)^2 p(x)\,dx$$

{It measures how far a set of random numbers are spread out from their average value}

⇒ Normal distribution will be abbreviated as:

$$\boxed{N(x : \mu, \sigma^2)}$$

random Variable, mean, Variance

⇒ x can be a vector. Normal distributions over vectors are called multivariate.

⇒ Multivariate normal distributions are characterized by density functions of the following form:

$$p(x) = \det(2\pi \Sigma)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right\}$$

{Positive Semidefinite Symmetric matrix called Covariance matrix}

⇒ The joint distribution of two random variables X and Y is given by:

$$p(x,y) = p(X=x, \& \ Y=y)$$

⇒ If X and Y are independent, we have

$$p(x,y) = p(x)\,p(y)$$

⇒ If we already know that Y's value is y and we would like to know probability that X's value is x conditioned on that fact.

↳ Such probability is called Conditional probability and is denoted as:

$$\boxed{p(x|y)}$$

⇒ If $P(y) > 0$,

$$\boxed{P(x|y) = \frac{P(x,y)}{P(y)}}$$

⇒ <u>Theorem of total probability</u>

$$\boxed{P(x) = \int P(x|y)\, P(y)\, dy} \text{ or } \boxed{P(x) = \sum_y P(x|y)\, P(y)}$$

⇒ <u>Bayes rule</u>: It states that if $P(y) > 0$

$$\boxed{P(x|y) = \frac{P(y|x)\, P(x)}{P(y)}}$$

{ Predominant role in 'Probabilistic robotics }

Let x ⇒ quantity that we would like to know
y ⇒ It is the data that we have. (Sensor measurement)

$P(x)$ ⇒ It is called prior probability distribution.
$P(x|y)$ ⇒ It is called posterior probability distribution

⇒ In robotics, this inverse probability is often coined "<u>generative model</u>".

⇒ $P(y)$, does not depend on x.

↳ Thus the factor $P(y)^{-1}$ will be same for any value x in the posterior $p(x|y)$.

↳ For this reason, $P(y)^{-1}$ is often written as a normalizer variable, and generically denoted η:

$$P(x|y) = \eta \, P(y|x)\, p(x)$$

⇒ If X is discrete, equation of this type can be computed as follows:

$$\forall x: \; aux_{x|y} = P(y|x)\, P(x)$$

$$aux_y = \sum_x aux_{x|y}$$

$$\forall x: \; p(x|y) = \frac{aux_{x|y}}{aux_y}$$

⇒ The expectation of a random variable X is given by

$$\boxed{E[X] = \int x \, p(x)\, dx} \qquad E[X] = \sum_x x\, P(x)$$

⇒ For arbitrary numerical value a and b,

$$E[aX + b] = a\,E[X] + b$$

⇒ The covariance of X is obtained as follows:

$$Cov[X] = E[X - E[X]]^2 = E[X^2] - E[X]^2$$

The covariance measures the squared expected deviation from the mean

⇒ Another important characteristic of a random variable is its entropy.

⇒ For discrete random variables, the entropy is given by the following expression.

$$H(P) = E[-\log_2 P(x)] = -\sum_x P(x) \log_2 P(x)$$

→ The concept of entropy originates in information theory.

→ Entropy is the expected information that the value of x carries.

→ $-\log_2 P(x)$ is the number of bits required to encode x using an optimal encoding

2.3) Robot Environment Interaction

2.3.1) State

→ Environments are characterized by state.

State → { Collection of all aspect of the robot and its environment that can impact the future }

⇒ State that changes ⇒ dynamic state
which not ⟶ static state

⇒ State will be denoted as x.
⇒ State at time t will be denoted as x_t.

Examples of State

robot pose (location and Orientation)
↳ Also referred to as Kinematic state.

Configuration of robot's actuators
↳ Also part of Kinematic state of the robot.

robot velocity

The location and features of surrounding objects in the environment.

The location and velocity of moving objects and people.

Scanned by CamScanner

⇒ A state x_t is called Complete if Knowledge of past carry no additional information that would help us to predict the future more accurately.

⇒ Process where no variable prior to x_t influence the evolution of future states is commonly Known as <u>Markov chains</u>.

2.3.2> Environment Interaction

→ Robot can influence the state of its environment through its actuators.

→ Gather information about the state through its sensors.

{ Typically, Sensor measurement arrive with some delay. Hence they provide information about the state a few moments ago. }

⇒ In accordance with the two types of environment interaction, the robot has access to two different data streams.

① Measurement data

⇒ Provides information about a momentary state of the environment.

⇒ The measurement data at time t will be denoted.

$$z_t$$

⇒ The notation,

$$z_{t_1:t_2} = z_{t_1}, z_{t_1+1} \cdots z_{t_2}$$

{denotes the Set of all measurements acquired from time t_1 to t_2}

② Control data

↳ Carry information about the Change of State

Example: Velocity, odometer etc.

⇒ Control data will be denoted u_t.

⇒ The variable u_t will always correspond to the change of state in the time interval $(t-1; t]$

2.3.3> Probabilistic Generative Laws

⇒ Probabilistic law characterizing the evolution of state might be given by a probability distribution of the following form:

$$p(x_t \mid x_{0:t-1}, z_{1:t-1}, u_{1:t}) = p(x_t \mid x_{t-1}, u_t)$$

⇓

{If state x is Complete}

Scanned by CamScanner

⇒ If x_t is complete, we have an important conditional independence.

$$p(z_t \mid x_{0:t}, z_{1:t-1}, u_{1:t}) = p(z_t \mid x_t)$$

$p(x_t \mid x_{t-1}, u_t)$ ⇒ State transition probability

→ It specifies how environment state evolves over time as a function of robot controls u_t.

$p(z_t \mid x_t)$ ⇒ Measurement probability

⇒ The state transition probability and the measurement probability together describe the dynamical stochastic of the robot and its environment.

⇒ This generative model is known as hidden Markov model (HMM) or dynamic Bayes network (DBN).

2.3.4> Belief Distributions

⇒ A belief reflects the robot's internal knowledge about the state of the environment.

Scanned by CamScanner

⇒ We will denote belief over a state variable x_t by $bel(x_t)$, which is an abbreviation for the posterior

$$bel(x_t) = p(x_t \mid z_{1:t}, u_{1:t})$$

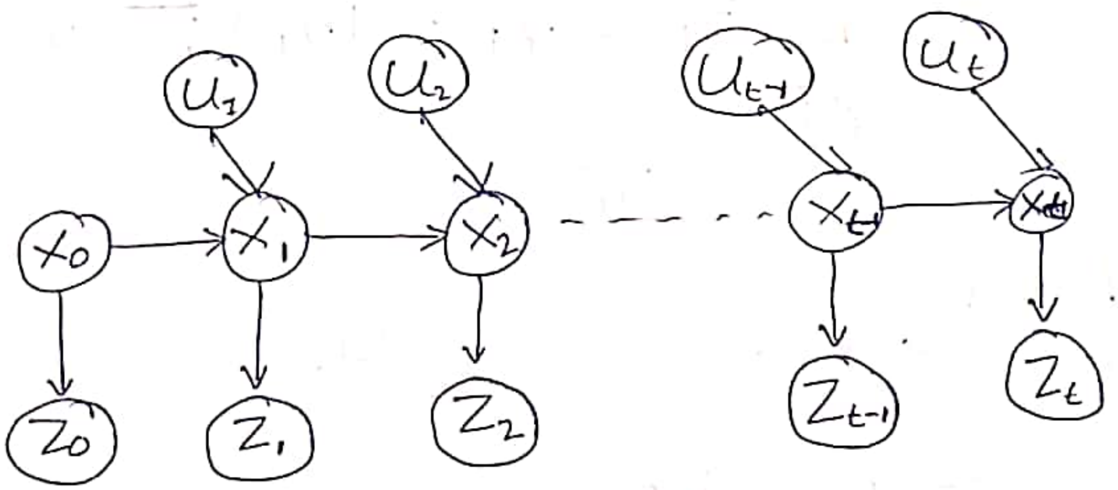
⇒ Posterior before incorporating z_t just after executing the control u_t

$$\overline{bel}(x_t) = p(x_t \mid z_{1:t-1}, u_{1:t})$$

This probability distribution is often referred to as prediction

⇒ Calculating $bel(x_t)$ from $\overline{bel}(x_t)$ is called correction or measurement update

---



2.4) Bayes Filters

2.4.1) The Bayes Filter Algorithm

⇒ The most general algorithm for calculating beliefs is given by the Bayes filter algorithm.

↳ Calculates belief distribution bel from measurement and control data.

Algorithm Bayes filter $(bel(x_{t-1}), u_t, z_t)$:

for all x_t do

$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1})\, bel(x_{t-1})\, dx_{t-1}$

$bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t)$

endfor

return $bel(x_t)$

→ Measurement update.

→ Prediction.

⇒ To compute the posterior belief recursively, the algorithm requires an initial belief $bel(x_0)$ at time $t=0$ as boundary condition.

→ Sensor reliability data and actuator reliability data is needed to perform bayes filter algorithm.

2.4.4> The Markov Assumption

"Markov assumption postulates that past and future data are independent if one knows the current state x_t"

2.5> Representation and Computation

⇒ When choosing an approximation, one has to trade off a range of properties:

1) Computational Efficiency

2) Accuracy of the approximation

3) Ease of implementation.