CS 4649/7649
Robot Intelligence: Planning

Estimation Basic

Sungmoon Joo

School of Interactive Computing
College of Computing
Georgia Institute of Technology

Slides are partly from http://stanford.edu/class/ee363/lectures/estim.pdf

Estimation, Prediction, Smoothing

Gaussian Random Vectors

random vector $x \in \mathbf{R}^n$ is *Gaussian* if it has density

$$p_x(v) = (2\pi)^{-n/2}(\det \Sigma)^{-1/2} \exp\left(-\frac{1}{2}(v-\bar{x})^T\Sigma^{-1}(v-\bar{x})\right),$$

for some $\Sigma = \Sigma^T > 0$, $\bar{x} \in \mathbf{R}^n$

$$f(x) = (2\pi\sigma^2)^{-\frac{1}{2}} \exp[-\frac{1}{2}(x-\mu)\sigma^{-2}(x-\mu)]$$

- denoted $x \sim \mathcal{N}(\bar{x}, \Sigma)$
- $\bar{x} \in \mathbf{R}^n$ is the *mean* or *expected* value of x, *i.e.*,

$$\bar{x} = \mathbf{E}\, x = \int v p_x(v) dv$$

- $\Sigma = \Sigma^T > 0$ is the *covariance* matrix of x, *i.e.*,

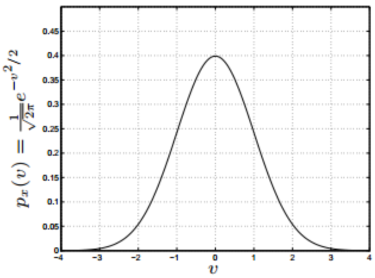
$$\Sigma \quad = \quad \mathbf{E}(x-\bar{x})(x-\bar{x})^T$$

Gaussian Random Vectors

$$= \mathbf{E}\, xx^T - \bar{x}\bar{x}^T \quad \textbf{Exercise!}$$

$$= \int (v-\bar{x})(v-\bar{x})^T p_x(v) dv$$

density for $x \sim \mathcal{N}(0,1)$:



Gaussian Random Vectors

- mean and variance of scalar random variable x_i are

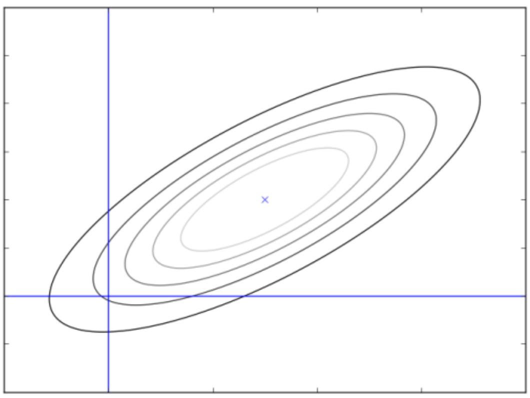
$$\mathbf{E}\, x_i = \bar{x}_i, \quad \mathbf{E}(x_i - \bar{x}_i)^2 = \Sigma_{ii}$$

  hence standard deviation of x_i is $\sqrt{\Sigma_{ii}}$
- covariance between x_i and x_j is $\mathbf{E}(x_i - \bar{x}_i)(x_j - \bar{x}_j) = \Sigma_{ij}$
- correlation coefficient between x_i and x_j is $\rho_{ij} = \dfrac{\Sigma_{ij}}{\sqrt{\Sigma_{ii}\Sigma_{jj}}}$
- mean (norm) square deviation of x from $\bar{x}$ is

$$\mathbf{E}\,\|x - \bar{x}\|^2 = \mathbf{E}\,\mathbf{Tr}(x - \bar{x})(x - \bar{x})^T = \mathbf{Tr}\,\Sigma = \sum_{i=1}^{n} \Sigma_{ii}$$

example: $x \sim \mathcal{N}(0, I)$ means x_i are independent identically distributed (IID) $\mathcal{N}(0,1)$ random variables

Ellipse



Confidence Ellipsoids

- $p_x(v)$ is constant for $(v - \bar{x})^T \Sigma^{-1} (v - \bar{x}) = \alpha$, *i.e.*, on the surface of ellipsoid

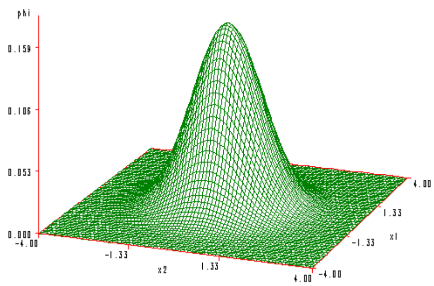
$$\mathcal{E}_\alpha = \{v \mid (v - \bar{x})^T \Sigma^{-1} (v - \bar{x}) \leq \alpha\}$$

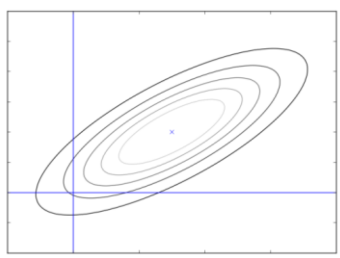
  - thus $\bar{x}$ and Σ determine shape of density

- η*-confidence set* for random variable z is smallest volume set S with $\mathbf{Prob}(z \in S) \geq \eta$
  - in general case confidence set has form $\{v \mid p_z(v) \geq \beta\}$

- $\mathcal{E}_\alpha$ are the η-confidence sets for Gaussian, called *confidence ellipsoids*
  - α determines confidence level η

Confidence Ellipsoids





Confidence Levels

the nonnegative random variable $(x - \bar{x})^T \Sigma^{-1} (x - \bar{x})$ has a χ_n^2 distribution, so $\mathbf{Prob}(x \in \mathcal{E}_\alpha) = F_{\chi_n^2}(\alpha)$ where $F_{\chi_n^2}$ is the CDF

some good approximations:

Cumulative Density Function

$$F_X(x) = P(X \leq x)$$

- $\mathcal{E}_n$ gives about 50% probability
- $\mathcal{E}_{n+2\sqrt{n}}$ gives about 90% probability

geometrically:

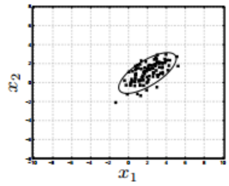
- mean $\bar{x}$ gives center of ellipsoid
- semiaxes are $\sqrt{\alpha\lambda_i}u_i$, where u_i are (orthonormal) eigenvectors of Σ with eigenvalues λ_i

Confidence Levels

example: $x \sim \mathcal{N}(\bar{x}, \Sigma)$ with $\bar{x} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$, $\Sigma = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$

- x_1 has mean 2, std. dev. $\sqrt{2}$
- x_2 has mean 1, std. dev. 1
- correlation coefficient between x_1 and x_2 is $\rho = 1/\sqrt{2}$
- $\mathbf{E}\,\|x - \bar{x}\|^2 = 3$

90% confidence ellipsoid corresponds to $\alpha = 4.6$:



(here, 91 out of 100 fall in $\mathcal{E}_{4.6}$)

Another Example

navigation using range measurements to distant beacons

$$y = Ax + v$$

- $x \in \mathbf{R}^2$ is location
- y_i is range measurement to ith beacon
- v_i is range measurement error, IID $\mathcal{N}(0, 1)$
- ith row of A is unit vector in direction of ith beacon

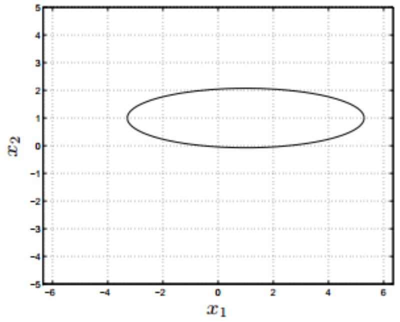
prior distribution:

$$x \sim \mathcal{N}(\bar{x}, \Sigma_x), \quad \bar{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \Sigma_x = \begin{bmatrix} 2^2 & 0 \\ 0 & 0.5^2 \end{bmatrix}$$

x_1 has std. dev. 2; x_2 has std. dev. 0.5

Another Example (cont.)

90% confidence ellipsoid for prior distribution
$\{ \; x \mid (x - \bar{x})^T \Sigma_x^{-1} (x - \bar{x}) \leq 4.6 \; \}$:



Affine Transformation

suppose $x \sim \mathcal{N}(\bar{x}, \Sigma_x)$

consider affine transformation of x:

$$z = Ax + b,$$

where $A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$

then z is Gaussian, with mean

$$\mathbf{E}\, z = \mathbf{E}(Ax + b) = A\,\mathbf{E}\, x + b = A\bar{x} + b$$

and covariance

$$\begin{aligned} \Sigma_z &= \mathbf{E}(z - \bar{z})(z - \bar{z})^T \\ &= \mathbf{E}\, A(x - \bar{x})(x - \bar{x})^T A^T \\ &= A\Sigma_x A^T \end{aligned}$$

Affine Transformation

examples:

- if $w \sim \mathcal{N}(0, I)$ then $x = \Sigma^{1/2} w + \bar{x}$ is $\mathcal{N}(\bar{x}, \Sigma)$

  useful for simulating vectors with given mean and covariance

- conversely, if $x \sim \mathcal{N}(\bar{x}, \Sigma)$ then $z = \Sigma^{-1/2}(x - \bar{x})$ is $\mathcal{N}(0, I)$

  (normalizes & decorrelates; called *whitening* or *normalizing*)

Linear Measurements

linear measurements with noise:

$$y = Ax + v$$

- $x \in \mathbf{R}^n$ is what we want to measure or estimate
- $y \in \mathbf{R}^m$ is measurement
- $A \in \mathbf{R}^{m \times n}$ characterizes sensors or measurements
- v is sensor noise

Linear Measurements

common assumptions:

- $x \sim \mathcal{N}(\bar{x}, \Sigma_x)$
- $v \sim \mathcal{N}(\bar{v}, \Sigma_v)$
- x and v are independent

- $\mathcal{N}(\bar{x}, \Sigma_x)$ is the *prior distribution* of x (describes initial uncertainty about x)
- $\bar{v}$ is noise *bias* or *offset* (and is usually 0)
- Σ_v is noise *covariance*

Linear Measurements

thus

$$\begin{bmatrix} x \\ v \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \bar{x} \\ \bar{v} \end{bmatrix}, \begin{bmatrix} \Sigma_x & 0 \\ 0 & \Sigma_v \end{bmatrix}\right)$$

using

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} I & 0 \\ A & I \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix}$$

we can write

$$\mathbf{E} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \bar{x} \\ A\bar{x} + \bar{v} \end{bmatrix}$$

and

$$\begin{aligned} \mathbf{E} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix}^T &= \begin{bmatrix} I & 0 \\ A & I \end{bmatrix} \begin{bmatrix} \Sigma_x & 0 \\ 0 & \Sigma_v \end{bmatrix} \begin{bmatrix} I & 0 \\ A & I \end{bmatrix}^T \\ &= \begin{bmatrix} \Sigma_x & \Sigma_x A^T \\ A\Sigma_x & A\Sigma_x A^T + \Sigma_v \end{bmatrix} \end{aligned}$$

Linear Measurements

covariance of measurement y is $A\Sigma_x A^T + \Sigma_v$

- $A\Sigma_x A^T$ is 'signal covariance'
- Σ_v is 'noise covariance'

State Estimator

- When we cannot measure the true state, we usually implement an estimator (or observer).

- Idea: Construct a new dynamical system that takes the available measurement as inputs, and provides as output an estimate of the state, $\hat{x}$

- Suppose we have a system $\dot{x} = A(t)x(t),\ y(t) = Hx(t)$
- Most common structure for a state estimator, called Luenberger form:

$$\dot{\hat{x}} = A(t)\hat{x}(t) + L(t)[y(t) - H\hat{x}(t)]$$

$$L(t)\ :\ \text{estimator gain}$$

- Estimation error

$$e(t) = x(t) - \hat{x}(t) \qquad \dot{e}(t) = (\mathbf{A} - \mathbf{LH})e(t)$$

Choose L s.t. the error dynamics are asymptotically stable → How?

Estimation vs Filtering

- In the presence of noise, the estimator must pick up information related to the state while rejecting noise.
- In that sense, the stochastic state estimation problem is a filtering problem

Least-squares Estimation

$$y_i = x + \epsilon_i, \;\; i = 1, 2, ..., N.$$

$$J(y, \hat{x}) = \sum_{i=1}^{N} (y_i - \hat{x})^2$$

$$\frac{\partial J(y, \hat{x})}{\partial \hat{x}} = \sum_{i=1}^{N} (y_i - \hat{x})^2 = -2 \sum_{i=1}^{N} (y_i - \hat{x}).$$

How?

$$\hat{x} = \frac{\sum_{i=1}^{N} y_i}{N}.$$

Least-squares Estimation

$$y_i = x + \epsilon_i, \;\; i = 1, 2, ..., N.$$

where ϵ_i is a random variable with $\mathbf{E}\epsilon_i = 0, \;\; \mathbf{E}\epsilon_i^2 = \sigma_i^2$

Unbiased Estimator $\mathbf{E}\hat{x} = x$

Minimum Variance Estimator (MVE) $\hat{x} = \arg\min_{\hat{x}} \mathbf{E}(\hat{x} - \mathbf{E}\hat{x})^2$

Gauss-Markov Theorem

Minimum variance, unbiased estimation is the one that minimizes:

$$J = \frac{1}{N} \sum_{i=1}^{N} \frac{(y_i - \hat{x})^2}{\sigma_i^2}$$

Least-squares Estimation

$y_i = \phi^T x + \epsilon_i, \quad i = 1, 2, 3, \ldots N.$ $\quad Y = \Phi x + e \quad E[ee^T] = \Sigma_e$

MVE $\hat{x} = \arg\min_x e^T \Sigma_e^{-1} e$

$= (\Phi^T \Sigma_e^{-1} \Phi)^{-1} \Phi^T \Sigma_e^{-1} Y.$

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \Phi = \begin{bmatrix} \phi_1^T \\ \phi_2^T \\ \vdots \\ \phi_N^T \end{bmatrix} e = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_N \end{bmatrix}$$

Minimum Mean-Square Estimation

suppose $x \in \mathbf{R}^n$ and $y \in \mathbf{R}^m$ are random vectors (not necessarily Gaussian)

we seek to estimate x given y

thus we seek a function $\phi : \mathbf{R}^m \to \mathbf{R}^n$ such that $\hat{x} = \phi(y)$ is near x

one common measure of nearness: mean-square error,

$$\mathbf{E}\,\|\phi(y) - x\|^2$$

minimum mean-square estimator (MMSE) ϕ_{mmse} minimizes this quantity

general solution: $\phi_{\text{mmse}}(y) = \mathbf{E}(x|y)$, *i.e.*, the conditional expectation of x given y

MMSE for Gaussian Vectors

now suppose $x \in \mathbf{R}^n$ and $y \in \mathbf{R}^m$ are jointly Gaussian:

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}, \begin{bmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{xy}^T & \Sigma_y \end{bmatrix} \right)$$

(after a lot of algebra) the conditional density is

$$p_{x|y}(v|y) = (2\pi)^{-n/2}(\det \Lambda)^{-1/2} \exp\left(-\frac{1}{2}(v-w)^T \Lambda^{-1}(v-w) \right),$$

where

$$\Lambda = \Sigma_x - \Sigma_{xy}\Sigma_y^{-1}\Sigma_{xy}^T, \quad w = \bar{x} + \Sigma_{xy}\Sigma_y^{-1}(y - \bar{y})$$

hence MMSE estimator (*i.e.*, conditional expectation) is

$$\hat{x} = \phi_{\text{mmse}}(y) = \mathbf{E}(x|y) = \bar{x} + \Sigma_{xy}\Sigma_y^{-1}(y - \bar{y})$$

Look familiar?

MMSE for Gaussian Vectors

ϕ_{mmse} is an affine function

MMSE estimation error, $\hat{x} - x$, is a Gaussian random vector

$$\hat{x} - x \sim \mathcal{N}(0, \Sigma_x - \Sigma_{xy}\Sigma_y^{-1}\Sigma_{xy}^T)$$

note that

$$\Sigma_x - \Sigma_{xy}\Sigma_y^{-1}\Sigma_{xy}^T \leq \Sigma_x$$

i.e., covariance of estimation error is always less than prior covariance of x

Q. How do we compare two covariance matrices?

Best Linear Unbiased Estimator

estimator

$$\hat{x} = \phi_{\text{blu}}(y) = \bar{x} + \Sigma_{xy}\Sigma_y^{-1}(y - \bar{y})$$

makes sense when x, y aren't jointly Gaussian

this estimator

- is *unbiased*, *i.e.*, $\mathbf{E}\,\hat{x} = \mathbf{E}\,x$
- often works well
- is widely used
- has minimum mean square error among all *affine* estimators

sometimes called *best linear unbiased* estimator

Minimum Variance

MMSE with Linear Measurements

consider specific case

$$y = Ax + v, \quad x \sim \mathcal{N}(\bar{x}, \Sigma_x), \quad v \sim \mathcal{N}(\bar{v}, \Sigma_v),$$

x, v independent

MMSE of x given y is affine function

$$\hat{x} = \bar{x} + B(y - \bar{y})$$

where $B = \Sigma_x A^T (A\Sigma_x A^T + \Sigma_v)^{-1}$, $\bar{y} = A\bar{x} + \bar{v}$

intepretation:

- $\bar{x}$ is our best prior guess of x (before measurement)
- $y - \bar{y}$ is the discrepancy between what we actually measure (y) and the expected value of what we measure ($\bar{y}$)

MMSE with Linear Measurements

- estimator modifies prior guess by B times this discrepancy
- estimator blends prior information with measurement
- B gives *gain* from *observed discrepancy* to *estimate*
- B is small if noise term Σ_v in 'denominator' is large
  (don't' believe sensor measurement, if the sensor is not good)

MMSE Error with Linear Measurements

MMSE estimation error, $\tilde{x} = \hat{x} - x$, is Gaussian with zero mean and covariance

$$\Sigma_{\text{est}} = \Sigma_x - \Sigma_x A^T (A\Sigma_x A^T + \Sigma_v)^{-1} A\Sigma_x$$

- $\Sigma_{\text{est}} \leq \Sigma_x$, *i.e.*, measurement always decreases uncertainty about x
- difference $\Sigma_x - \Sigma_{\text{est}}$ (or some other comparison) gives *value* of measurement y in estimating x
  - $(\Sigma_{\text{est}\ ii}/\Sigma_{x\ ii})^{1/2}$ gives fractional decrease in uncertainty of x_i due to measurement
  - $(\mathbf{Tr}\,\Sigma_{\text{est}}/\,\mathbf{Tr}\,\Sigma)^{1/2}$ gives fractional decrease in uncertainty in x, measured by mean-square error

Estimation Error Covariance

- error covariance Σ_{est} can be determined *before* measurement y is made!
- to evaluate Σ_{est}, only need to know
  - A (which characterizes sensors)
  - prior covariance of x (*i.e.*, Σ_x)
  - noise covariance (*i.e.*, Σ_v)
- you *do not* need to know the measurement y (or the means $\bar{x}$, $\bar{v}$)
- useful for *experiment design* or *sensor selection*

Information Matrix Formulas

we can write estimator gain matrix as

$$\begin{aligned} B &= \Sigma_x A^T (A \Sigma_x A^T + \Sigma_v)^{-1} \\ &= \left(A^T \Sigma_v^{-1} A + \Sigma_x^{-1}\right)^{-1} A^T \Sigma_v^{-1} \end{aligned}$$

How?

- $n \times n$ inverse instead of $m \times m$
- Σ_x^{-1}, Σ_v^{-1} sometimes called *information matrices*

corresponding formula for estimator error covariance:

$$\begin{aligned} \Sigma_{\text{est}} &= \Sigma_x - \Sigma_x A^T (A \Sigma_x A^T + \Sigma_v)^{-1} A \Sigma_x \\ &= \left(A^T \Sigma_v^{-1} A + \Sigma_x^{-1}\right)^{-1} \end{aligned}$$

Information Matrix Formulas

can interpret $\Sigma_{\text{est}}^{-1} = \Sigma_x^{-1} + A^T \Sigma_v^{-1} A$ as:

$$\begin{aligned} &\text{posterior information matrix } (\Sigma_{\text{est}}^{-1}) \\ &= \text{prior information matrix } (\Sigma_x^{-1}) \\ &+ \text{information added by measurement } (A^T \Sigma_v^{-1} A) \end{aligned}$$