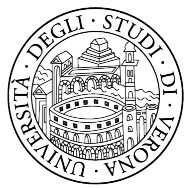
Statistical Filtering and Control for AI and Robotics

Part II. Linear methods for regression & Kalman filtering

Riccardo Muradore

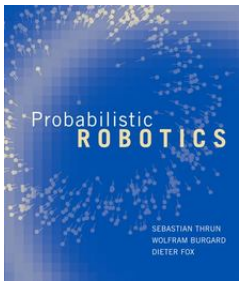


UNIVERSITÀ
di VERONA
Dipartimento
di INFORMATICA

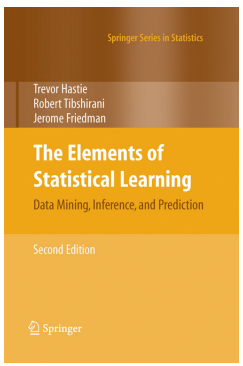
Outline

References

These lectures are based on the following books



Sebastian Thrun, Wolfram Burgard and Dieter Fox, "ProbabilisticRobotics", MIT Press, 2005



Trevor Hastie, Robert Tibshirani and Jerome Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Springer, 2009

Several pictures from those books have been copied and pasted here

Linear Methods for Regression

Supervised learning: use the inputs (i.e. predictors, independent variables, features) to predict the values of the outputs (i.e. responses, dependent variables)

This distinction in output type has led to a naming convention for the prediction tasks: regression when we predict quantitative outputs, and classification when we predict qualitative outputs.

Notation:

- $\mathbf{x} \in \mathbb{R}^m$ random variable ($\mathbf{x}_i \in \mathbb{R}$ is its i-th component)
- $x \in \mathbb{R}^m$ an observation of the random variable $\mathbf{x} \in \mathbb{R}^m$ ($x_i \in \mathbb{R}$ is its i-th component)
- $X \in \mathbb{R}^{m \times N}$ a collection of N observations ($X_i^T \in \mathbb{R}^m$ is its i-th row)

We will focus on the regression problem: this means that input and output vectors consist of qualitative measurements

Input: $\mathbf{x} \in \mathbb{R}^m, x \in \mathbb{R}^m, X \in \mathbb{R}^{N \times m}$
Output: $\mathbf{y} \in \mathbb{R}^p, y \in \mathbb{R}^p, Y \in \mathbb{R}^{N \times p}$
Prediction: $\hat{\mathbf{y}} \in \mathbb{R}^p, \hat{y} \in \mathbb{R}^p, \hat{Y} \in \mathbb{R}^{p \times N}$

Linear Model: (from now on $p = 1$)

$$y = f(x) = x^T \beta$$

where $\beta \in \mathbb{R}^m$

Prediction

$$\hat{y} = x^T \hat{\beta}$$

where $\hat{\beta} \in \mathbb{R}^m$ is the matrix of coefficients that we have to determine

Remark. If $p = 1$, the gradient $f'(x) = \nabla_x f(x) = \beta$ is a vector pointing in the steepest uphill direction

Let $X \in \mathbb{R}^{N \times m}$ and $Y \in \mathbb{R}^N$ a training set of data (collection of N pairs (x, y))
How to choice β?
First of all we have to introduce an index as a function of β.

Let $RSS(\beta)$ be the residual sum of squares

$$RSS(\beta) := \sum_{i=1}^{N} (Y_i - X_i\beta)^T (Y_i - X_i\beta) = (Y - X\beta)^T (Y - X\beta)$$

We search for

$$\hat{\beta} := \arg\min_{\beta} RSS(\beta)$$

Computing the first and second derivative we get the normal equations

$$\begin{aligned} \nabla_\beta RSS(\beta) &= -2X^T(Y - X\beta) \\ \nabla^2_{\beta\beta} RSS(\beta) &= 2X^T X \end{aligned}$$

If $X^T X$ is nonsingular (i.e. X has full column rank), the unique solution is given by the normal equations

$$\nabla_\beta RSS(\beta) = 0 \quad \Leftrightarrow \quad X^T(Y - X\beta) = 0$$

i.e.

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

and the prediction of y given a new value x is

$$\hat{y} = x^T \hat{\beta}$$

Observations:

- We assume that the underlying model is linear
- Statistics of x and y do not play any role (it seems ...)

Least Squares $p > 1$

Linear model

$$Y = XB + E$$

where $X \in \mathbb{R}^{N \times m}$, $Y \in \mathbb{R}^{N \times p}$, $E \in \mathbb{R}^{N \times p}$ and $B \in \mathbb{R}^{m \times p}$

The RSS takes the form

$$RSS(B) := \text{trace}\{(Y - XB)^T (Y - XB)\}$$

and the least square estimation of B is written in the same way

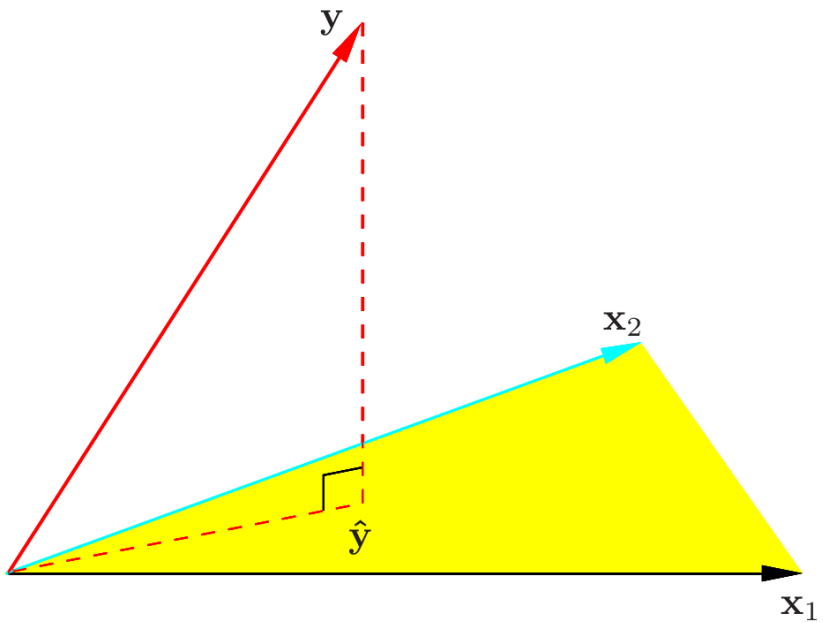
$$\hat{B} = (X^T X)^{-1} X^T Y$$

Multiple outputs do not affect one another's least squares estimates

If the component of the vector r.v $\mathbf{e}$ are correlated, i.e. $\mathbf{e} \sim \mathcal{N}(0, \Sigma)$, then we can define a weighted RSS

$$RSS(B, \Sigma) := \sum_{i=1}^{N} (Y_i - X_i B)^T \Sigma^{-1} (Y_i - X_i B)$$

Geometric interpretation



The normal equations

$$X^T(Y - X\beta) = 0$$

means the estimation $\hat{Y} = X\hat{\beta} = X(X^TX)^{-1}X^TY$ is the orthogonal projection of Y into the subspace X

Statistical interpretation

We now consider the r.v. $\mathbf{x}$ and $\mathbf{y}$ as input and output, respectively, and we seek a function $f(\mathbf{x})$ for predicting $\mathbf{y}$.

The criterion should be now deal with stochastic quantities: we introduce the expected squared prediction error EPE (strictly related with the mean squared error MSE)

$$\begin{aligned} EPE(f) &:= \mathbb{E}\left[(\mathbf{y} - f(\mathbf{x}))^T(\mathbf{y} - f(\mathbf{x}))\right] \\ &= \int_{S_x, S_y} (y - f(x))^T(y - f(x))p(x, y)dxdy \end{aligned}$$

where we implicitly assumed that $\mathbf{x}$ and $\mathbf{y}$ have a joint PDF. $EPE(f)$ is a $\mathcal{L}_2$ loss function

Conditioning on $\mathbf{x}$ we can re-write $EPE(f)$ as

$$EPE(f) := \mathbb{E}_x\left[\ \mathbb{E}_{y|x}\left[(\mathbf{y} - f(\mathbf{x}))^T(\mathbf{y} - f(\mathbf{x}))|\mathbf{x}\right]\ \right]$$

We can determine $f(\cdot)$ pointwise

$$f(x) = \arg\min_c \mathbb{E}_{y|x}\left[(\mathbf{y}-c)^T(\mathbf{y}-c)|\mathbf{x}=x\right]$$

which means that

$$f(x) = \mathbb{E}\left[\mathbf{y}|\mathbf{x}=x\right]$$

i.e. the best $f(x)$ is the conditional mean (according to the *EPE* criterion).

Beautiful but, given the data X, Y how can we compute the conditional expectation?!?

Statistical interpretation

Let us assume again

$$f(\mathbf{x}) = \mathbf{x}^T \beta$$

then

$$EPE(f) \quad := \quad \mathbb{E}\left[(\mathbf{y} - \mathbf{x}^T\beta)^T(\mathbf{y} - \mathbf{x}^T\beta)\right]$$

Differentiating w.r.t. β we end up with

$$\beta = \left(\mathbb{E}[\mathbf{x}\mathbf{x}^T]\right)^{-1}\mathbb{E}[\mathbf{x}^T\mathbf{y}]$$

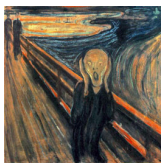
Computing the auto- and cross-correlation (i.e. using real numbers!)

$$\mathbb{E}[\mathbf{x}\mathbf{x}^T] \overset{N\to\infty}{\longrightarrow} S_{xx} := \frac{1}{N}\sum_{i=1}^{N} X_i^T X_i = \frac{1}{N} X^T X$$

$$\mathbb{E}[\mathbf{x}^T\mathbf{y}] \overset{N\to\infty}{\longrightarrow} S_{xy} := \frac{1}{N}\sum_{i=1}^{N} X_i Y_i^T = \frac{1}{N} X Y^T$$

Then we get

$$\begin{aligned}\hat{\beta} &= \left(\frac{1}{N}X^T X\right)^{-1}\frac{1}{N}XY^T \\ &= (X^T X)^{-1} XY^T\end{aligned}$$



Again the normal equations !!!

But now we can provide a statistical interpretation of $\hat{\beta}$. Let $\mathbf{y} = \mathbf{x}^T\beta + \mathbf{e}$, $\mathbf{e} \sim \mathcal{N}(0, \sigma^2)$ be our model ($p = 1$), then $\hat{\beta}$ is a Gaussian variable

$$\hat{\beta} \sim \mathcal{N}(\beta, (X^T X)^{-1}\sigma^2)$$

In fact, since $\hat{\beta} = (X^T X)^{-1} X\mathbf{y} - (X^T X)^{-1} X\mathbf{e}$

$$\hat{\mathbf{y}} = \mathbf{x}^T\hat{\beta} + \mathbf{e}$$

Gauss-Markov theorem

Given the linear model

$$y = x^T \beta, \qquad Y = X\beta$$

the least squares estimator $\hat{\phi}(x_0) = x_0^T \hat{\beta}$ of $\phi(x_0) = x_0^T \beta$ is unbiased because

$$\mathbb{E}[x_0^T \hat{\beta}] = x_0^T \beta$$

Theorem

If $\bar{\phi}(x_0)$ is any other unbiased estimation ($\mathbb{E}[\bar{\phi}(x_0)] = x_0^T \beta$) then

$$\mathrm{Var}(\hat{\phi}(x_0)) \leq \mathrm{Var}(\bar{\phi}(x_0))$$

Remark. Mean square error of a generic estimator $\bar{\phi}$ ($p = 1$)

$$MSE(\bar{\phi}) = \mathbb{E}[(\bar{\phi} - \phi)^2] \overset{(*)}{=} \underbrace{\mathrm{Var}(\bar{\phi})}_{\text{variance}} + \underbrace{(\mathbb{E}[\bar{\phi}] - \phi)^2}_{\text{bias}}$$

(*) = sum and subtract $\mathbb{E}[\bar{\phi}]$.

Given the stochastic linear model

$$\mathbf{y} = \mathbf{x}^T\beta + \mathbf{e}, \qquad \mathbf{e} \sim \mathcal{N}(0, \sigma^2)$$

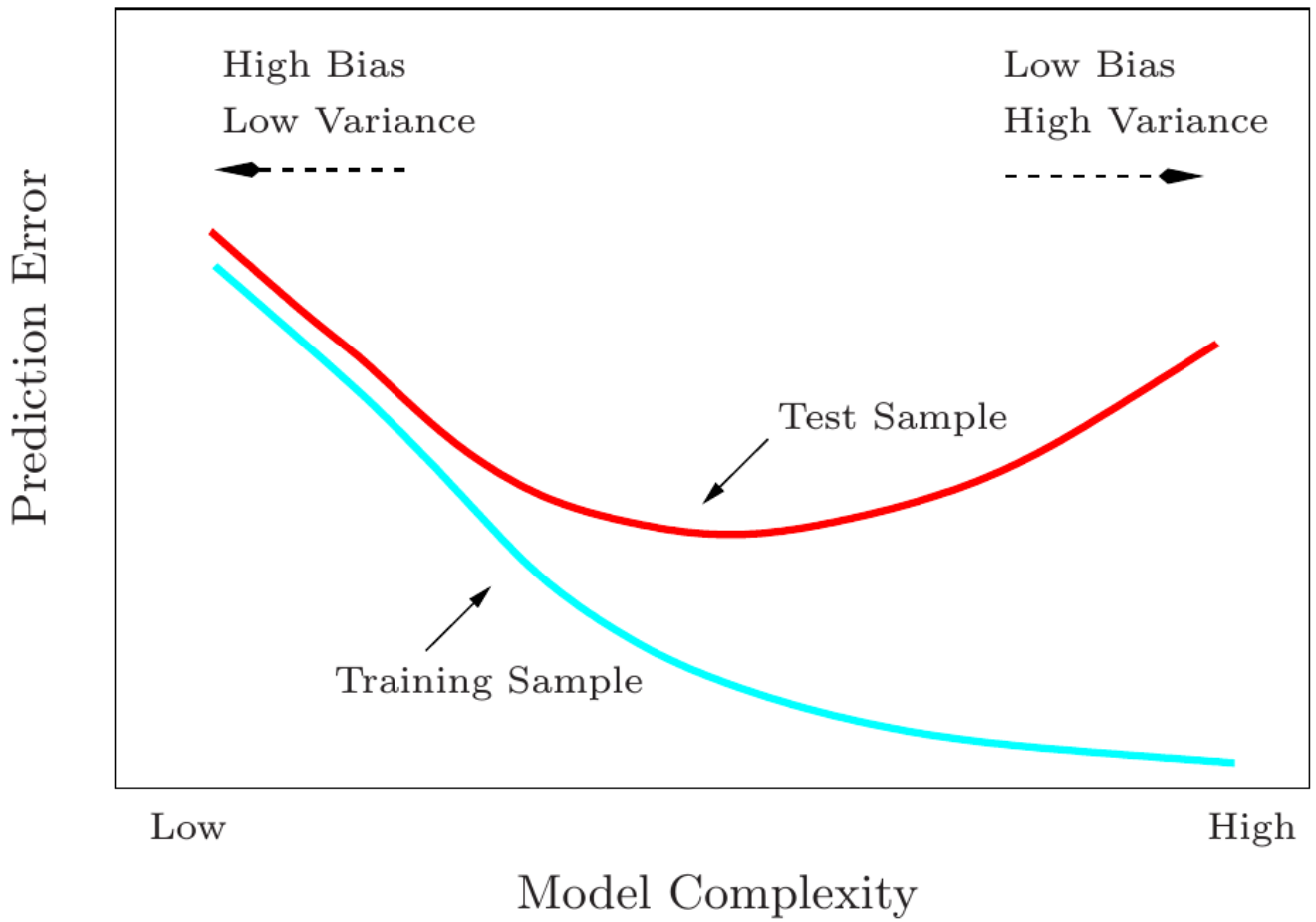
and let $\bar{\phi}(x_0)$ be the estimator for $y_0 = \phi(x_0) + e_0$, $\phi(x_0) = x_0^T\beta$.

The expected prediction error (EPE) of $\bar{\phi}(x_0)$ is

$$\begin{aligned} EPE(\bar{\phi}(x_0)) &= \mathbb{E}[(y_0 - \bar{\phi}(x_0))^2] \\ &= \sigma^2 + \mathbb{E}[(x_0^T\beta - \bar{\phi}(x_0))^2] \\ &= \sigma^2 + \underbrace{\mathrm{Var}(\bar{\phi}) + (\mathbb{E}[\bar{\phi}] - \phi)^2}_{MSE} \end{aligned}$$

Bias-variance trade-off

underfitting VS overfitting



Statistical model:

$$\mathbf{y} = f(\mathbf{x}) + \mathbf{e}$$

where $\mathbf{y}$ is a random error with zero mean ($\mathbb{E}[\mathbf{e}] = 0$) and is independent of $\mathbf{x}$.

This means that the relationship between $\mathbf{y}$ and $\mathbf{x}$ is not deterministic ($f(\cdot)$)

The additive r.v. $\mathbf{e}$ takes care of measurement noise, model uncertainty and non measured variables correlated with $\mathbf{y}$ as well

We often assume that the random variables $\mathbf{e}$ are independent and identically distributed (i.i.d.)

Assuming a linear basis expansion for $f_\theta(x)$ parametrized by the unknowns collected within the vector θ

$$f_\theta(x) = \sum_1^K h_k(x)\theta_k$$

where examples of $h_k(x)$ can be

$$\begin{aligned} h_k(x) &= x_k \\ h_k(x) &= (x_k)^2 \\ h_k(x) &= \sin(x_k) \\ h_k(x) &= \frac{1}{1+e^{-x^T\beta_k}} \end{aligned}$$

The optimization problem to solve is

$$\hat{\theta} = \arg\min_{\theta\in\Theta} RSS(\theta) = \sum_1^N (y_i - f_\theta(x_i))^2$$

where RSS stands for Residual Sum of Squares

Are there other kinds of criterion besides RSS, EPE?

YES, A more general principle for estimation is maximum likelihood estimation

Let $p_\theta(y)$ be the PDF of the samples $y_1, \ldots, y_N$

The log-probability (or log-likelihood) of the observed samples is

$$L(\theta) = \sum_{1}^{N} \log p_\theta(y_i)$$

Principle of maximum likelihood: the most reasonable values for θ are those for which the probability of the observed samples is largest

If the error $\mathbf{e}$ in the following statistical model

$$\mathbf{y} = f_\theta(\mathbf{x}) + \mathbf{e}$$

is Gaussian, $\mathbf{e} \sim \mathcal{N}(0, \sigma^2)$, then the conditional probability is

$$p(y|x, \theta) \sim \mathcal{N}(f_\theta(x), \sigma^2)$$

Then log-likelihood of the data is

$$\begin{aligned} L(\theta) &= \sum_1^N \log p(y_i|f_\theta(x_i), \theta) \\ &= -\frac{N}{2}\log(2\pi) - N\log\sigma - \frac{1}{2\sigma^2}\sum_{i=1}^N (y_i - f_\theta(x_i))^2 \end{aligned}$$

Least squares for the additive error model is equivalent to maximum likelihood using the conditional probability (The yellow is the $RSS(\theta)$)

Penalty function, or regularization methods, introduces our knowledge about the type of functions $f(x)$ we are looking for

$$PRSS(f, \lambda) := RSS(f) + \lambda g(f)$$

where the functional $g(f)$ will force our knowledge (or desiderata) on f

Example. One-dimension cubic smoothing spline is the solution of

$$PRSS(f, \lambda) := \sum_{i=1}^{N} (y_i - f(x_i))^2 + \lambda \int [f''(s)]^2 dx$$

Remark. Penalty function methods have a Bayesian interpretation:

- $g(f)$ is the log-prior distribution
- $PRSS(f, \lambda)$ is the log-posterior distribution
- the solution of $\arg\min_f PRSS(f, \lambda)$ is the posterior mode

If we want a local regression estimation of $f(x_0)$, we have to solve the problem

$$\hat{\theta} = \arg\min_{\theta} RSS(f_\theta, x_0) = \sum_{i=1}^{N} K_\lambda(x_0, x_i)(y_i - f_\theta(x_i))^2$$

where the kernel function $K_\lambda(x_0, x)$ weights the point x around x_0. The optimal estimation is $f_{\hat{\theta}}(x_0)$

An example of kernel function is the Gaussian kernel

$$K_\lambda(x_0, x) = \frac{1}{\lambda} \exp\left[-\frac{\|x - x_0\|^2}{2\lambda}\right]$$

Examples of $f_\theta(x)$ are

- $f_\theta(x) = \theta_0$, constant function
- $f_\theta(x) = \theta_0 + \theta_1 x$, linear regression

Basis functions

The function f can be approximated using a set of M basis functions h_m

$$f_\theta(x) = \sum_{m=1}^{M} \theta_m h_m(x)$$

where $\theta = [\theta_1 \quad \cdots \quad \theta_M]$

Examples of basis functions:

- Radial basis functions:

$$f_\theta(x) = \sum_{m=1}^{M} \theta_m K_{\lambda_m}(\mu_m, x), \qquad K_\lambda(\mu, x) = e^{-\|x-\mu\|^2/2\lambda}$$

- Single-layer feed-forward neural network

$$f_\theta(x) = \sum_{m=1}^{M} \theta_m \sigma(\alpha_m^T x + b_m), \qquad \sigma(x) = \frac{1}{1+e^{-x}}$$

Remark. Linear methods can then be used with nonlinear input-output transformation because the model is linear in the parameters θ

Subset selection

"The least squares estimates often have low bias but large variance. Prediction accuracy can sometimes be improved by shrinking or setting some coefficients to zero. By doing so we sacrifice a little bit of bias to reduce the variance of the predicted values, and hence may improve the overall prediction accuracy."

Ridge Regression

Ridge regression shrinks the regression coefficients by imposing a penalty on their size.
The coefficients $\hat{\beta}^{ridge}$ are obtained solving the minimization problem

$$\hat{\beta}^{ridge} = \arg\min_{\beta}\{\underbrace{\sum_{i=1}^{N}(Y_i - X_i\beta)^T(Y_i - X_i\beta)}_{RSS(\beta)} + \lambda \underbrace{\sum_{i=1}^{m}\beta_i^2}_{g(\beta)=\beta^T\beta}\}$$

with $\lambda \geq 0$, or the equivalent constrained problem

$$\hat{\beta}^{ridge} = \arg\min_{\beta} \sum_{i=1}^{N}(Y_i - X_i\beta)^T(Y_i - X_i\beta)$$

$$\text{s. to} \quad \sum_{i=1}^{m}\beta_i^2 \leq t$$

The solution is

$$\hat{\beta}^{ridge} = (X^TX + \lambda I)^{-1}X^TY$$

The coefficients $\hat{\beta}^{lasso}$ are obtained solving the minimization problem

$$\hat{\beta}^{lasso} = \arg\min_{\beta}\{\underbrace{\sum_{i=1}^{N}(Y_i - X_i\beta)^T(Y_i - X_i\beta)}_{RSS(\beta)} + \lambda \underbrace{\sum_{i=1}^{m}|\beta_i|}_{g(\beta)}\}$$

with $\lambda \geq 0$, or the equivalent constrained problem

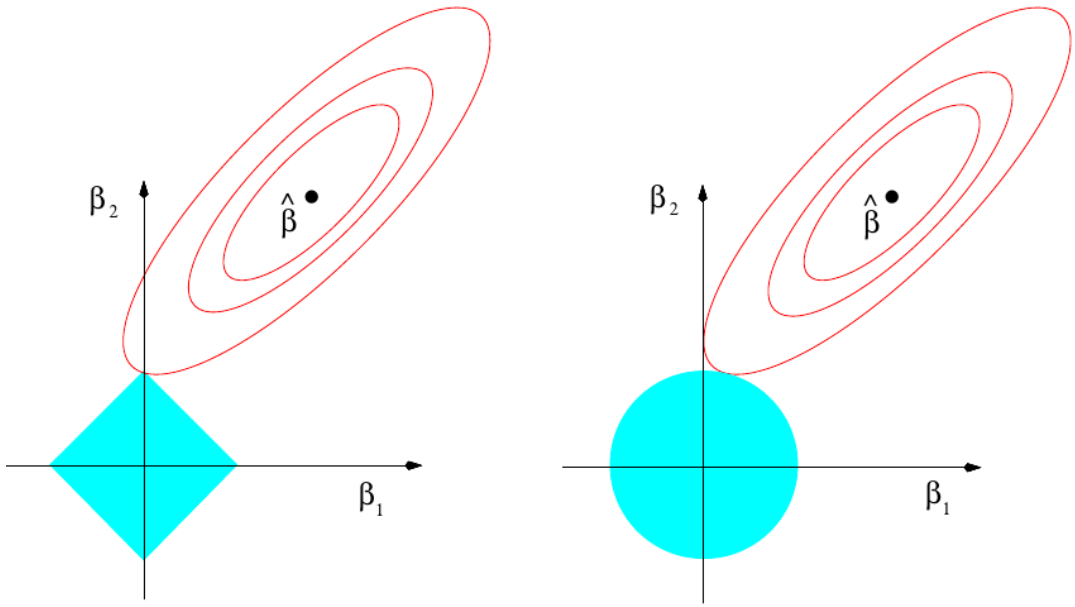
$$\begin{aligned}\hat{\beta}^{lasso} = \quad & \arg\min_{\beta} \quad \sum_{i=1}^{N}(Y_i - X_i\beta)^T(Y_i - X_i\beta) \\ & \text{s. to} \quad \sum_{i=1}^{m}|\beta_i| \leq t\end{aligned}$$

The are no closed form expression for $\hat{\beta}^{lasso}$

Remark 1. The Ridge Regression uses a $\mathcal{L}_2$ norm on β, whereas Lasso the $\mathcal{L}_1$ norm. This means that the solution is nonlinear in the data.

Remark 1. Decreasing t forces some of the coefficients to be set to zero (exactly).

Ridge regression VS Lasso



Lasso

$$|\beta_1| + |\beta_2| \leq t$$

Ridge

$$\beta_1^2 + \beta_2^2 \leq t^2$$

The red ellipses are the contours of the least squares error function

Definition (random variable)

A random variable $\mathbf{x}: \Omega \to E$ is a measurable function from the set of possible outcomes Ω to some set E. Ω is a probability space and E is a measurable space.

Roughly speaking: A random variable $\mathbf{x}$ is a rule for assigning to every outcome ω of an experiments a number $\mathbf{x}(\omega)$

Definition (stochastic process)

Given a probability space $(\Omega, \mathcal{F}, P)$ and a measurable space (S, Σ), an S-valued stochastic process is a collection of S-valued random variables on Ω, indexed by a totally ordered set T ("time"). That is, a stochastic process is a collection $\{\mathbf{x}_t : t \in T\}$ where each $\mathbf{x}_t$ is an S-valued random variable on Ω. The space S is then called the state space of the process.

Roughly speaking: A stochastic process $\mathbf{x}_t$ is a rule for assigning to every outcome ω a function $\mathbf{x}(t, \omega)$

$\{\mathbf{x}_t\}$ has the following interpretations:

- It is a family of functions $\mathbf{x}_t(\omega)$ when t and ω are variables.
- It is a single time function (or a realization of the given process) $\mathbf{x}_t(\bar{\omega})$ when t is a variable and $\omega = \bar{\omega}$ is fixed.
- It is a random variable if $t = \bar{t}$ is fixed and ω is variable, i.e. $\mathbf{x}_{\bar{t}}(\omega)$ state of the process at time t.
- It is a number if t and ω are fixed

If $T = \mathbb{R}$, $\{\mathbf{x}_t\}$ is a continuous-time process

If $T = \mathbb{Z}$, $\{\mathbf{x}_k\}$ is a discrete-time process

Even though the dynamics of the system is described by ODE, in the following we will consider discrete-time processes because the sensing system provides measurements at discrete moments.

Remark The r.v. $\mathbf{x}_{\bar{k}}(\omega)$ can be continuous even if $k \in T = \mathbb{Z}$

Gaussian filter

Given the measurement y_k, $k = 0, 1, \ldots$ related to an unknown state variable x_k, we are interested in the following estimation problems

- **Filtering**

$$y_0, y_1, \ldots, y_k \longrightarrow \hat{x}_{k|k}$$

- ***h*-step ahead Prediction**

$$y_0, y_1, \ldots, y_k \longrightarrow \hat{x}_{k+h|k}$$

- ***h*-step backward Smoothing**

$$y_0, y_1, \ldots, y_k \longrightarrow \hat{x}_{k-h|k}$$

- **Smoothing**

$$y_0, y_1, \ldots, y_N \longrightarrow \hat{x}_{k|N}$$

Gaussian filters

Gaussian filters assume that the undergoing phenomena can be modeled by Gaussian distributions.

This assumption allows to solve in recursive way the general Bayes filters' formulation

Why are Gaussian distributions so good?

- Gaussians are unimodal: they have a single maximum
- The statistics (mean, variance and higher order moments) of a Gaussian are described by two parameters: its mean and variance
- The linear combination of Gaussians is still Gaussian

Definition (Gaussian r.v.)

An n-dimensional random variable X is Gaussian with mean $\mu \in \mathbb{R}^n$ and variance $\Sigma \in \mathbb{R}^{n \times n}, \Sigma = \Sigma^T > 0$, $X \sim \mathcal{N}(\mu, \Sigma)$, if its probability density function (PDF) is given by

$$p(x) = \frac{1}{\sqrt{(2\pi)^n \det \Sigma}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}.$$

This means that

$$\begin{aligned} \mu &= \mathbb{E}[X] \\ \Sigma &= \mathrm{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])(X - \mathbb{E}[X])^T]. \end{aligned}$$

Theorem (Joint Gaussian r.v.)

Let $\mathbf{x} \in \mathbb{R}^n$ *and* $\mathbf{y} \in \mathbb{R}^m$ *be joint Gaussian*

$$p(x, y) \sim \mathcal{N}\left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \right)$$

Then

- *the r.v.* $\mathbf{z} = A\mathbf{x} + B\mathbf{y}$ *is still Gaussian, i.e.* $\mathbf{z} \sim \mathcal{N}(\mu_z, \Sigma_z)$, *where*

$$\begin{aligned}
\mu_z &= \mathbb{E}[Ax + By] = A\mu_x + B\mu_y \\
\Sigma_z &= \mathbb{E}\left[(Ax + By - A\mu_x - B\mu_y)(Ax + By - A\mu_x - B\mu_y)^T \right] \\
&= \mathbb{E}\left[\left([A \;\; B] \begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix} \right) \left([A \;\; B] \begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix} \right)^T \right] \\
&= [A \;\; B] \mathbb{E}\left[\begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix} \begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix}^T \right] [A \;\; B]^T \\
&= [A \;\; B] \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} A^T \\ B^T \end{bmatrix}
\end{aligned}$$

Theorem (...)

- *the Gaussian random variable* **x** *conditioned on the Gaussian random variable* **y** *is still a Gaussian random variable. The PDF of* **x** *given* **y** *is*

$$p(x|y) = \mathcal{N}(\mu_{x|y}, \Sigma_{x|y}) \tag{1}$$

where

$$\mu_{x|y} = \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y) \tag{2}$$

$$\Sigma_{x|y} = \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx} \tag{3}$$

if $\Sigma_{yy} > 0$.

Theorem (Minimum Variance Estimator)

Let $\mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^m$ *be two r.v. (non necessariamente Gaussiane), and* $g : \mathbb{R}^m \to \mathbb{R}^n$ *a measurable function.*

We define $\hat{\mathbf{x}}_g = g(\mathbf{y})$ *as the estimator of* $\mathbf{x}$ *given* $\mathbf{y}$ *through the function* g*, and* $\mathbf{e}_g = \mathbf{x} - g(\mathbf{y}) = \mathbf{x} - \hat{\mathbf{x}}_g$ *the corresponding estimation error.*

The estimator $\hat{\mathbf{x}} = \mathbb{E}[\mathbf{x}|\mathbf{y}] = \hat{g}(\mathbf{y})$ *is optimal because it minimizes the error variance, i.e.*

$$Var(\mathbf{e}) = \mathbb{E}[(\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^T] \leq \mathbb{E}[(\mathbf{x} - \hat{\mathbf{x}}_g)(\mathbf{x} - \hat{\mathbf{x}}_g)^T] = Var(\mathbf{e}_g), \quad \forall g(\cdot)$$

where $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$ *is the error of the optimal estimator.*
The error of the optimal estimator and the estimation are uncorrelated

$$\mathbb{E}[e\hat{g}(\mathbf{y})^T] = 0.$$

Stochastic model

LTI Stochastic model

We focus now on the state-space representation of a generic Linear Time-Invariant (LTI) stochastic model:

$$\begin{cases} x_{k+1} &= Ax_k + w_k \\ y_k &= Cx_k + v_k \end{cases}$$

where:

$$\begin{cases} v_k &\sim \mathcal{N}(0, R), & \mathbb{E}[v_k v_h^T] = R\delta(k-h) \\ w_k &\sim \mathcal{N}(0, Q), & \mathbb{E}[w_k w_h^T] = Q\delta(k-h) \\ x_0 &\sim \mathcal{N}(\bar{x}_0, P_0) \end{cases}$$

and v_k, w_k, x_0 are uncorrelated zero-mean Gaussian r.v.

$$\begin{aligned} \mathbb{E}[v_k w_h^T] &= 0 \\ \mathbb{E}[x_0 v_k^T] &= 0 \\ \mathbb{E}[x_0 w_k^T] &= 0 \end{aligned}$$

The state-space model is a way to describe the dynamical evolution of a stochastic process

From the evolution of the state and of the output at time t

$$\begin{aligned} x_k &= A^{k-k_0}x_0 + \sum_{i=k_0}^{k-1} A^{k-i-1}w_i \\ y_k &= CA^{k-k_0}x_0 + \sum_{i=k_0}^{k-1} CA^{k-i-1}w_i + v_k \end{aligned}$$

we also have

$$\begin{aligned} \mathbb{E}[x_k w_h^T] &= 0, \quad \forall h \geq k \\ \mathbb{E}[x_k v_h^T] &= 0 \\ \mathbb{E}[y_k v_h^T] &= Q\delta(k-h) \end{aligned}$$

Kalman filtering

The Kalman filter (or minimum variance filter) is defined as:

$$\hat{x}_{k+1|k+1} = \mathbb{E}\left[x_{k+1}|y_0, \ldots, y_{k+1}\right] = \mathbb{E}\left[x_{k+1}|y_{k+1}, Y^k\right] \tag{4}$$

where $Y^k = (y_k, \ldots, y_1, y_0)$.

Goal: we need a recursive expression for $\hat{x}_{k+1|k+1}$ without using

$$\mu_{x|y} \quad = \quad \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y) \tag{5}$$

at any time instant k, i.e. when a new measurement is available.

The explicit expression for $\mathbb{E}[X|Y]$ is easy to derive from (5) if X e Y are joint Gaussian with means μ_X, μ_Y and variances $\begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{bmatrix}$.

To rewrite

$$\hat{x}_{k+1|k+1} = \mathbb{E}\left[x_{k+1}|y_0, \ldots, y_{k+1}\right] = \mathbb{E}\left[x_{k+1}|y_{k+1}, Y^k\right]$$

in the form $\mathbb{E}[X|Y]$ we introduce the following conditional random variables

$$\begin{aligned} X &= x_{k+1}|Y^k \\ Y &= y_{k+1}|Y^k \end{aligned}$$

and compute the following means, variances and covariances:

$$\begin{aligned} \mu_X &= \mathbb{E}\left[x_{k+1}|Y^k\right] \\ \mu_Y &= \mathbb{E}\left[y_{k+1}|Y^k\right] \\ P_{k+1|k} = \Sigma_{XX} &= Var\left[x_{k+1}|Y^k\right] \\ \Sigma_{YY} &= Var\left[y_{k+1}|Y^k\right] \\ \Sigma_{XY} = \Sigma_{YX}^T &= Cov\left[x_{k+1}, y_{k+1}|Y^k\right]. \end{aligned}$$

The optimal estimator is given by:

$$\mathbb{E}[X|Y] = \hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + \Sigma_{XY}\Sigma_{YY}^{-1}\left(y_{k+1} - \hat{y}_{k+1|k}\right) \tag{6}$$

and the variance of the estimation error is

$$\Sigma_{X|Y} = P_{k+1|k+1} = \Sigma_{XX} - \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX} \tag{7}$$

Conditional means

Mean μ_X

$$\begin{aligned}
\mu_X &= \mathbb{E}\left[x_{k+1}|Y^k\right] \\
&= \mathbb{E}\left[Ax_k + w_k|Y^k\right] \\
&= A\mathbb{E}\left[x_k|Y^k\right] + \mathbb{E}\left[w_k|Y^k\right] \\
&= A\hat{x}_{k|k} \\
&= \hat{x}_{k+1|k}
\end{aligned}$$

Mean μ_Y

$$\begin{aligned}
\mu_Y &= \mathbb{E}\left[y_{k+1}|Y^k\right] \\
&= \mathbb{E}\left[Cx_{k+1} + v_{k+1}|Y^k\right] \\
&= C\mathbb{E}\left[x_{k+1}|Y^k\right] + \mathbb{E}\left[v_{k+1}|Y^k\right] \\
&= C\hat{x}_{k+1|k} \\
&= CA\hat{x}_{k|k}
\end{aligned}$$

Variance Σ_{XX}

$$
\begin{aligned}
\Sigma_{XX} &= Var\left[x_{k+1}|Y^k\right] \\
&= \mathbb{E}\left[\left(x_{k+1}-\hat{x}_{k+1|k}\right)\left(x_{k+1}-\hat{x}_{k+1|k}\right)^T|Y^k\right] \\
&= \mathbb{E}\left[\left(Ax_k+w_k-A\hat{x}_{k|k}\right)\left(Ax_k+w_k-A\hat{x}_{k|k}\right)^T|Y^k\right] \\
&= A\mathbb{E}\left[\left(x_k-\hat{x}_{k|k}\right)\left(x_k-\hat{x}_{k|k}\right)^T|Y^k\right]A^T + \\
&\quad +A\mathbb{E}\left[\left(x_k-\hat{x}_{k|k}\right)w_k^T|Y^k\right] + \\
&\quad +\mathbb{E}\left[w_k\left(x_k-\hat{x}_{k|k}\right)^T|Y^k\right]A^T+\mathbb{E}\left[w_kw_k^T|Y^k\right] \\
&= AP_{k|k}A^T+Q \\
&= P_{k+1|k}
\end{aligned}
$$

Variance Σ_{YY}

$$
\begin{aligned}
\Sigma_{YY} &= Var\left[y_{k+1}|Y^k\right] \\
&= \mathbb{E}\left[\left(y_{k+1}-\hat{y}_{k+1|k}\right)\left(y_{k+1}-\hat{y}_{k+1|k}\right)^T|Y^k\right] \\
&= \mathbb{E}\left[\left(Cx_{k+1}+v_{k+1}-C\hat{x}_{k+1|k}\right)\left(Cx_{k+1}+v_{k+1}-C\hat{x}_{k+1|k}\right)^T|Y^k\right] \\
&= C\mathbb{E}\left[\left(x_{k+1}-\hat{x}_{k+1|k}\right)\left(x_{k+1}-\hat{x}_{k+1|k}\right)^T|Y^k\right]C^T + \\
&\quad + C\mathbb{E}\left[\left(x_{k+1}-\hat{x}_{k+1|k}\right)v_{k+1}^T|Y^k\right] + \\
&\quad + \mathbb{E}\left[v_{k+1}\left(x_{k+1}-\hat{x}_{k+1|k}\right)^T|Y^k\right]C^T + \mathbb{E}\left[v_{k+1}v_{k+1}^T|Y^k\right] \\
&= CP_{k+1|k}C^T + R
\end{aligned}
$$

Conditional variances

Covariance $\Sigma_{XY} = \Sigma_{YX}^T$

$$
\begin{aligned}
\Sigma_{XY} &= Cov\left[x_{k+1}, y_{k+1} | Y^k\right] \\
&= \mathbb{E}\left[\left(x_{k+1} - \hat{x}_{k+1|k}\right)\left(y_{k+1} - \hat{y}_{k+1|k}\right)^T | Y^k\right] \\
&= \mathbb{E}\left[\left(Ax_k - A\hat{x}_{k|k} + w_k\right)\left(CAx_k - CA\hat{x}_{k|k} + v_{k+1} + Cw_k\right)^T | Y^k\right] \\
&= A\mathbb{E}\left[\left(x_k - \hat{x}_{k|k}\right)\left(x_k - \hat{x}_{k|k}\right)^T | Y^k\right] A^T C^T + \mathbb{E}\left[w_k w_k^T | Y^k\right] C^T \\
&= AP_{k|k}A^T C^T + QC^T \\
&= P_{k+1|k}C^T
\end{aligned}
$$

Summing up

The random variable $z = \begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix}$ conditioned on Y^k, has the following PDF

$$p\left(z|Y^k\right) \sim \mathcal{N}\left(\begin{bmatrix} \hat{x}_{k+1|k} \\ C\hat{x}_{k+1|k} \end{bmatrix}, \begin{bmatrix} P_{k+1|k} & P_{k+1|k}C^T \\ CP_{k+1|k} & CP_{k+1|k}C^T + R \end{bmatrix}\right)$$

with:

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k}$$

$$P_{k+1|k} = AP_{k|k}A^T + Q$$

The last step is to compute

$$p\left(x_{k+1}|Y^{k+1}\right) \sim \mathcal{N}\left(\hat{x}_{k+1|k+1}, P_{k+1|k+1}\right)$$

where the mean $\hat{x}_{k+1|k+1}$ is the optimal estimation we are looking for and $P_{k+1|k+1}$ the variance of the corresponding estimation error.

Substituting the previous expression we end up with

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + P_{k+1|k}C^T\left(CP_{k+1|k}C^T + R\right)^{-1}\left(y_{k+1} - C\hat{x}_{k+1|k}\right)$$

The **Kalman gain** is the matrix

$$K_{k+1} = P_{k+1|k}C^T\left(CP_{k+1|k}C^T + R\right)^{-1}$$

mapping the output estimation error into the correction of the prediction state

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\left(y_{k+1} - C\hat{x}_{k+1|k}\right)$$

The variance of the estimation error is

$$P_{k+1|k+1} = P_{k+1|k} - P_{k+1|k}C^T\left(CP_{k+1|k}C^T + R\right)^{-1}CP_{k+1|k}$$

Recursive formulation

Prediction step / A priori estimation

$$\begin{aligned} \hat{x}_{k+1|k} &= A\hat{x}_{k|k} \\ P_{k+1|k} &= AP_{k|k}A^T + Q \end{aligned}$$

Estimation step / A posteriori estimation

$$\begin{aligned} \hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + K_{k+1}(y_{k+1} - C\hat{x}_{k+1|k}) \\ P_{k+1|k+1} &= P_{k+1|k} - P_{k+1|k}C^T(CP_{k+1|k}C^T + R)^{-1}CP_{k+1|k} \end{aligned}$$

Initial conditions

$$\begin{aligned} \hat{x}_{0|-1} &= \bar{x}_0 \\ P_{0|-1} &= P_0 \end{aligned}$$

Kalman filter

$$\begin{aligned}\hat{x}_{k+1|k+1} &= A\hat{x}_{k|k} + K_{k+1}(y_{k+1} - CA\hat{x}_{k|k}) \\ P_{k+1|k} &= AP_{k|k-1}A^T - AP_{k|k-1}C^T(CP_{k|k-1}C^T + R)^{-1}CP_{k|k-1}A^T + Q\end{aligned}$$

with

$$K_{k+1} = P_{k+1|k}C^T\left(CP_{k+1|k}C^T + R\right)^{-1}$$

The matrix recursive equation $P_{k+1|k} = \ldots$ is called Riccati equation.

Kalman predictor

$$\begin{aligned}\hat{x}_{k+1|k} &= A\hat{x}_{k|k-1} + K_k(y_k - C\hat{x}_{k|k-1}) \\ P_{k+1|k} &= AP_{k|k-1}A^T - AP_{k|k-1}C^T(CP_{k|k-1}C^T + R)^{-1}CP_{k|k-1}A^T + Q\end{aligned}$$

with

$$K_k = AP_{k|k-1}C^T\left(CP_{k|k-1}C^T + R\right)^{-1}$$

Observations

1. All the information in $y(i)$, $i \in [0, k-1]$ is "contained" in the estimation state $\hat{x}_{k-1|k-1}$: the following conditional expectations are equal

$$\mathbb{E}[x_k | y_k, y_{k-1}, \dots, y_0] = \mathbb{E}[x_k | \hat{x}_{k-1|k-1}, y_k]$$

2. The optimal gain K_k is time-varying even if the stochastic model is LTI.
3. There is a more general formulation of the Kalman filter where w_k and v_k are correlated.
4. The same recursive equation for the Kalman filter can be used with linear time-varying stochastic systems.

What's happen when $k \to \infty$?
Does the estimation error converge to zero with minimal variance?

Theorem

Given the stochastic LTI model

$$\begin{cases} x_{k+1} &= Ax_k + w_k \\ y_k &= Cx_k + v_k \end{cases} \tag{8}$$

with

$$\begin{cases} v_k &\sim \mathcal{N}(0, R), & \mathbb{E}[v_k v_h^T] = R\delta(k-h) \\ w_k &\sim \mathcal{N}(0, Q), & \mathbb{E}[w_k w_h^T] = Q\delta(k-h) \\ x_0 &\sim \mathcal{N}(\bar{x}_0, P_0) \end{cases} \tag{9}$$

where v_k, w_k, x_0 are zero mean uncorrelated Gaussian random variables.

Theorem (...)

Then

1. *The Algebraic Riccati Equation (ARE):*

$$P_\infty = AP_\infty A^T - AP_\infty C^T(CP_\infty C^T + R)^{-1}CP_\infty A^T + Q$$

   has a unique positive definite symmetric matrix solution $P_\infty = P_\infty^T > 0$

2. P_∞ *is stabilizable, i.e.* $(A - K_\infty C)$ *is asymptotically stable with*

$$K_\infty = P_\infty C^T \left(CP_\infty C^T + R\right)^{-1}.$$

3. $\lim_{k\to\infty} P(k|k-1) = P_\infty$ *holds for all initial conditions* $P(0|-1) = P_0 = P_0^T \geq 0$,

if and only if

1. (A, C) *is detectable,*
2. $(A, Q^{1/2})$ *is stabilizable.*

Kalman filter (LTI)

$$\begin{aligned} \hat{x}_{k+1|k+1} &= A\hat{x}_{k|k} + K_\infty(y_{k+1} - CA\hat{x}_{k|k}) \\ P &= APA^T - APC^T(CPC^T + R)^{-1}CPA^T + Q \end{aligned}$$

with

$$K_\infty = PC^T\left(CPC^T + R\right)^{-1}$$

Kalman predictor (LTI)

$$\begin{aligned} \hat{x}_{k+1|k} &= A\hat{x}_{k|k-1} + \bar{K}_\infty(y_k - C\hat{x}_{k|k-1}) \\ P &= APA^T - APC^T(CPC^T + R)^{-1}CPA^T + Q \end{aligned}$$

with

$$\bar{K}_\infty = APC^T\left(CPC^T + R\right)^{-1}.$$

Kalman smoother

Model: $\{A, C, Q, R\}$

$$\begin{aligned} x_{k+1} &= Ax_k + w_k \\ y_k &= CX_k + v_k \end{aligned}$$

Data: sequence of N samples of the output

$$y_0, y_1, \ldots, y_N$$

STEP 1

"Standard" Kalman filtering (**forward step**)

$$\begin{aligned} \hat{x}^f_{k+1|k+1} &= A\hat{x}^f_{k|k} + K_{k+1}(y_{k+1} - CA\hat{x}^f_{k|k}) \\ \hat{x}^f_{0|0} &= \bar{x}_0 \\ P^f_{k|k} &= \ldots \\ P^f_{0|0} &= P_0 \end{aligned}$$

STEP 2

Smoothing (**backward step**)

$$\hat{x}^s_{k|N} = \hat{x}^f_{k|k} + \bar{K}_k \left[\hat{x}^s_{k+1|N} - \hat{x}^f_{k+1|t}\right]$$
$$\hat{x}^s_{N|N} = \hat{x}^f_{N|N}$$

where the conditional covariance matrix $P(t|N)$ satisfies the time-backward matrix equation

$$P_{k|N} = P^f_{k|k} + \bar{K}_k \left[P_{k+1|N} - P^f_{k+1|k}\right]$$
$$P_{N|N} = P^f_{N|N}.$$

with

$$\bar{K}_k = P^f_{k|k} A^T \left(P^f_{k+1|k}\right)^{-1}$$

Problem: speed estimation

Given the measurement equation

$$y(t) = s(t) + v(t)$$

where

- $s(t)$ is the signal we are interesting in (e.g. angular position),
- $y(t)$ is the measurement given by a sensor (e.g. an encoder)
- $v(t)$ is the addictive measurement noise

We can face different kinds of estimation problems:

- **Filtering**: determine the best estimation $\hat{s}(t)$ of $s(t)$ based on the measurements $y(\cdot)$ till time t (i.e. $y(0), y(1), \ldots, y(t)$)
- **Prediction**: determine the best estimation $\hat{s}(t+h)$ of $s(t+h)$ with $h \geq 1$ based on the measurements $y(\cdot)$ till time t (i.e. $y(0), y(1), \ldots, y(t)$)
- **Smoothing**: determine the best estimation $\hat{s}(t-h)$ of $s(t-h)$ with $h \geq 1$ based on the measurements $y(\cdot)$ till time t (i.e. $y(0), y(1), \ldots, y(t)$)

Problem: speed estimation

Let θ and ω be the angular position and velocity, respectively. Knowing nothing about the physical model that produces the signal $s(t)$ we set the derivative of the velocity equal to a white noise.

A stochastic process n is called white noise if its values $n(t_i)$ and $n(t_j)$ are uncorrelated $\forall i \neq j$, i.e.

$$\mathrm{Corr}\{n(t_i), n(t_j)\} = Q(t_i)\delta(t_i - t_j)$$

We also assume the $n(t)$ is Gaussian with zero-mean and constant variance $Q \in \mathbb{R}$ for all t

Kinematic model

$$\begin{aligned} \dot{\theta}(t) &= \omega(t) \\ \dot{\omega}(t) &= n(t) \end{aligned}$$

Measurement equation

$$y(t) = \theta(t) + v(t)$$

where v is another white noise and $v(t)$ is Gaussian with zero-mean and constant variance $R \in \mathbb{R}$.

Problem: speed estimation

Kinematic model

$$\begin{aligned} \dot{\theta}(t) &= \omega(t) \\ \dot{\omega}(t) &= n(t) \end{aligned}$$

Measurement equation

$$y(t) = \theta(t) + v(t)$$

Remark. The process n takes into account the uncertainty on the model of the system, whereas v models the measurement noise superimposed to the "real" value θ

Let $x(t)$ be the vector state

$$x(t) := \begin{bmatrix} \theta(t) \\ \omega(t) \end{bmatrix}$$

Problem: speed estimation

The continuous-time state space model of our basic system is

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} n(t) \\ y(t) &= \begin{bmatrix} 1 & 0 \end{bmatrix} x(t) + v(t) \end{aligned}$$

Its discrete-time approximation with sample time T_s is given by

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} \frac{T_s^2}{2} \\ T_s \end{bmatrix} n_k \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k \end{aligned}$$

Problem: speed estimation

The relationship between the two state space models

$$\begin{cases} x_{k+1} &= \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} \frac{T_s^2}{2} \\ T_s \end{bmatrix} n_k \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k \end{cases}, \qquad \begin{cases} x_{k+1} &= Ax_k + w_k \\ y_k &= Cx_k + v_k \end{cases}$$

is

$$\begin{aligned} A &:= \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \\ w_k &:= \begin{bmatrix} \frac{T_s^2}{2} \\ T_s \end{bmatrix} n_k \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{T_s^2}{2} \\ T_s \end{bmatrix} \begin{bmatrix} \frac{T_s^2}{2} \\ T_s \end{bmatrix}^T Q \right) \\ C &:= \begin{bmatrix} 1 & 0 \end{bmatrix} \\ v_k &:= v_k \end{aligned}$$

Tuning of the filter

The variance R depends on the encoder resolution (we can read it on the datasheet) whereas the matrix Q is chosen by the designer to try to "explain" the measurements in the best way.

Model with inputs

If an input command u_k enters within the stochastic model

$$\begin{cases} x_{k+1} &= Ax_k + Bu_k + w_k \\ y_k &= Cx_k + v_k \end{cases},$$

how do the filter equations change?

Fortunately if u_k is a function of past measurements (e.g. $u_k = f(y_{0:k})$) than we can simple add the term Bu_k in the recursive equations:

$$\begin{aligned} \hat{x}_{k+1|k} &= A\hat{x}_{k|k-1} + Bu_k + K_k(y_k - C\hat{x}_{k|k-1}) \\ P_{k+1|k} &= AP_{k|k-1}A^T - AP_{k|k-1}C^T(CP_{k|k-1}C^T + R)^{-1}CP_{k|k-1}A^T + Q \end{aligned}$$

or

$$\begin{aligned} \hat{x}_{k+1|k} &= A\hat{x}_{k|k-1} + Bu_k + \bar{K}_\infty(y_k - C\hat{x}_{k|k-1}) \\ P &= APA^T - APC^T(CPC^T + R)^{-1}CPA^T + Q \end{aligned}$$

Important remark

In the book "Probabilistic Robotics" the Kalman equations are obtained following a different approach (using first and second derivatives of opportune quadratic functions).

Another difference is that they start with the linear Gaussian system

$$x_k = Ax_{k-1} + Bu_k + w_k$$

which is the same of

$$x_{k+1} = Ax_k + Bu_{k+1} + w_{k+1}$$

Observations:

- using w_{k+1} instead of w_k does not change anything because w is white noise
- on the other hand, it would make a big difference using Bu_{k+1} instead of Bu_k as we did: for this reason the authors of the book introduce the assumption that the control input u is a random process independent of the state and the measurement