EE5506 Advanced Statistical Signal Processing
Cheat Sheet

Oğul Can Yurdakul

07/07/2022

Contents

0. Preliminaries

0.a Notation and Basic Definitions

Observations: $\mathbf{x} \in \mathbf{X}$ ($x \in X$ if scalar)

True parameter: $\theta \in \Theta$ ($\theta \in \Theta$ if scalar)

Estimate: $\hat{\theta}$ ($\hat{\theta}$ if scalar)

Likelihood function: $p(\mathbf{x}; \theta)$ is the probability distribution of the observations, parametrized by θ. It is taken as a function of the parameters θ and NOT of the observations $\mathbf{x}$.

Log Likelihood Function (LLF): $\ln p(\mathbf{x}; \theta)$, similarly **Negative Log Likelihood Function (NLLF)**

0.b Useful Linear Algebra

Matrix Inversion Lemma:

$$(\mathbf{A}+\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}=\mathbf{A}^{-1}-\mathbf{A}^{-1}\mathbf{B}(\mathbf{D}+\mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1}$$

0.c Useful Gaussians

Suppose $\mathbf{x}$ and $\mathbf{y}$ are jointly Gaussian random variables with joint pdf

$$\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}\sim\mathcal{N}\left(\begin{bmatrix}\mu_{\mathbf{x}}\\ \mu_{\mathbf{y}}\end{bmatrix},\begin{bmatrix}\mathbf{C}_{\mathbf{x}} & \mathbf{C}_{\mathbf{xy}}\\ \mathbf{C}_{\mathbf{yx}} & \mathbf{C}_{\mathbf{y}}\end{bmatrix}\right)$$

where

$$\mathbf{C}_{\mathbf{x}}=\mathbb{E}\left\{(\mathbf{x}-\mu_{\mathbf{x}})(\mathbf{x}-\mu_{\mathbf{x}})^T\right\}\qquad \mathbf{C}_{\mathbf{y}}=\mathbb{E}\left\{(\mathbf{y}-\mu_{\mathbf{y}})(\mathbf{y}-\mu_{\mathbf{y}})^T\right\}$$
$$\mathbf{C}_{\mathbf{xy}}=\mathbb{E}\left\{(\mathbf{x}-\mu_{\mathbf{x}})(\mathbf{y}-\mu_{\mathbf{y}})^T\right\}=\mathbb{E}\left\{(\mathbf{y}-\mu_{\mathbf{y}})(\mathbf{x}-\mu_{\mathbf{x}})^T\right\}^T=\mathbf{C}_{\mathbf{yx}}^T$$

Marginalization: The marginal pdf's of $\mathbf{x}$ and $\mathbf{y}$ are

$$\mathbf{x}\sim\mathcal{N}(\mu_{\mathbf{x}},\mathbf{C}_{\mathbf{x}})\qquad\text{and}\qquad\mathbf{y}\sim\mathcal{N}(\mu_{\mathbf{y}},\mathbf{C}_{\mathbf{y}})$$

Conditioning: The conditional pdf of $\mathbf{x}\,|\,\mathbf{y}$ is

$$\mathbf{x}\,|\,\mathbf{y}\sim\mathcal{N}(\mu_{\mathbf{x}\,|\,\mathbf{y}},\mathbf{C}_{\mathbf{x}\,|\,\mathbf{y}})$$

where

$$\mu_{\mathbf{x}\,|\,\mathbf{y}}=\mu_{\mathbf{x}}+\mathbf{C}_{\mathbf{xy}}\mathbf{C}_{\mathbf{y}}^{-1}(\mathbf{y}-\mu_{\mathbf{y}})\qquad\text{and}\qquad\mathbf{C}_{\mathbf{x}\,|\,\mathbf{y}}=\mathbf{C}_{\mathbf{x}}-\mathbf{C}_{\mathbf{xy}}\mathbf{C}_{\mathbf{y}}^{-1}\mathbf{C}_{\mathbf{yx}}$$

1. Classical Estimation: Deterministic θ

1.a Minimum Variance Unbiased Estimate (MVUE)

Bias: The estimate $\hat{\theta}$ is a function of random variables $\mathbf{x}$, and so is random itself. The bias is the expected deviation from the *true* parameter θ.

$$b(\theta) := \mathbb{E}\left\{\hat{\theta} - \theta\right\} = \mathbb{E}\left\{\hat{\theta}\right\} - \theta$$

MSE and Bias: The bias $b(\theta)$ of an estimator $\hat{\theta}$ and its mean squared error $\mathrm{MSE}(\hat{\theta}, \theta)$ are related as

$$\mathrm{MSE}(\hat{\theta}, \theta) = b(\theta)^2 + \mathrm{var}(\hat{\theta})$$

This reliance on the true knowledge of θ is what motivates the unbiasedness constraint.

MSE: Notice how the unbiasedness constraint $b(\theta) = \mathbb{E}\left\{\hat{\theta}\right\} - \theta = 0$ results in

$$\mathrm{MSE}(\hat{\theta}, \theta) = \mathrm{var}(\hat{\theta})$$

Existence of MVUE: MVUE may not exist if

1. if an unbiased estimator doesn't exist (losing on the U side), or
2. none of the existing unbiased estimators has a *uniformly* minimum variance (losing on the MV side).

Finding MVUE: There are three methods to find MVUE:

1. Determine the Cramér-Rao Lower Bound (CRLB) and see if an estimator satisfies it.
2. Use the Rao-Blackwell-Lechman-Scheffe Theorem (skipped).
3. Restrict in form, for example restrict to linear estimators only. Notice that this will give true MVUE is the problem is linear by nature.

1.b Cramér-Rao Lower Bound (CRLB)

Meaning: The Cramér-Rao lower bound sets the best performance criterion for an *unbiased* estimator.

$$\sigma^2_{\hat{\theta}}(\theta) \geq CRLB(\theta)$$

Inserting the definitions from above, I see that it sets a lower bound on the MSE:

$$\mathrm{MSE}(\hat{\theta}) \geq CRLB(\theta)$$

Regularity Condition: CRLB is defined under the following condition:

$$\mathbb{E}_{\mathbf{x}}\left\{\frac{\partial \ln p(\mathbf{x};\theta)}{\partial \theta}\right\} = \frac{\partial}{\partial \theta}\mathbb{E}_{\mathbf{x}}\{\ln p(\mathbf{x};\theta)\} = 0 \qquad \forall \theta \in \Theta$$

This can be interpreted as requiring the expected log likelihood to be independent of θ.

Alternative: The support of $p(\mathbf{x};\theta)$ as the pdf of $\mathbf{x}$ must not depend on θ, and the derivative $\frac{\partial}{\partial \theta}\ln p(\mathbf{x};\theta)$ must exist and be finite for all $\mathbf{x} \in X$ and $\theta \in \Theta$.

Non-example: The following uniform pdf parametrized by $\theta \in \mathbb{R}^+$ violates this regularity condition:

$$p(x;\theta) = \frac{1}{\theta} \qquad \text{for} \qquad 0 \leq x \leq \theta$$

Formula: Under the above regularity condition, CRLB is given by the following expression:

$$CRLB(\theta) = \frac{-1}{\mathbb{E}_{\mathbf{x}}\left\{ \frac{\partial^2}{\partial \theta^2} \ln p(\mathbf{x};\theta) \right\} \Big|_{\theta=\text{true value}}}$$

Notice how this is still dependent on the true value θ. Depending on the problem CRLB may vary as the true parameter value changes, which effectively means some parameter values are better estimated compared to the others.

Alternative: The alternative form of CRLB, which requires proof, is given as follows:

$$CRLB(\theta) = \frac{1}{\mathbb{E}_{\mathbf{x}}\left\{ \left[\frac{\partial}{\partial \theta} \ln p(\mathbf{x};\theta) \right]^2 \right\}}$$

Theorem: There exists an unbiased estimator that achieves the CRLB if and only if

$$\frac{\partial}{\partial \theta} \ln p(\mathbf{x};\theta) = I(\theta)\left(g(\mathbf{x}) - \theta\right)$$

for some functions $I(\theta)$ and $g(\mathbf{x})$. In this case, the estimate $\hat{\theta}$ and the CRLB become

$$\hat{\theta} = g(\mathbf{x}) \qquad \text{and} \qquad CRLB(\theta) = \frac{1}{I(\theta)}$$

because

$$I(\theta) = -\mathbb{E}_{\mathbf{x}}\left\{ \frac{\partial^2}{\partial \theta^2} \ln p(\mathbf{x};\theta) \right\}$$

Fisher Information: The function $I(\theta)$ is called as the Fisher information.

$$I(\theta) = -\mathbb{E}_{\mathbf{x}}\left\{ \frac{\partial^2}{\partial \theta^2} \ln p(\mathbf{x};\theta) \right\} = \mathbb{E}_{\mathbf{x}}\left\{ \left[\frac{\partial}{\partial \theta} \ln p(\mathbf{x};\theta) \right]^2 \right\}$$

It requires the following desired properties of information, just like Shannon information does:

1. $I(\theta) \geq 0$ (see the alternative form), and
2. $I(\theta)$ is additive over independent observations (follows easily from the pdf of independent variables, properties of log and linearity of differentiation).

Efficiency: An estimator is said to be efficient if

- it is unbiased and
- it attains CRLB.

Asymptotic Efficiency: An estimator is said to be asymptotically efficient if it tends to efficiency as the number of observations tends to infinity.

Transformation of Parameters: Say that I have an estimate $\hat{\theta}$, and another variable of interest given by $\alpha = f(\theta)$. Then the CRLB of the parameter α can be expressed as

$$CRLB(\alpha) = \left(\frac{\partial}{\partial \theta} f(\theta) \right)^2 CRLB(\theta)$$

The factor $\left(\frac{\partial}{\partial\theta}f(\theta)\right)^2$ captures the sensitivity of α to θ.

Affine Transformation: Suppose $g(\mathbf{x}) = \theta$ is an efficient, and $\alpha = a\theta + b$ is a parameter affinely related to θ. Then the estimator of α given by $\hat{\alpha} = ag(\mathbf{x}) + b = a\hat{\theta} + b$ is an efficient estimator.

Vector Estimation: If I have N parameters of interest expressed as a vector

$$\theta = \begin{bmatrix}\theta_1 & \dots & \theta_N\end{bmatrix}$$

then I have

Fisher Information Matrix (FIM) given by

$$[\mathbf{I}(\theta)]_{mn} = -\mathbb{E}_{\mathbf{x}}\left\{\frac{\partial^2}{\partial\theta_n\partial\theta_m}\ln p(\mathbf{x};\theta)\right\}$$

Notice that it is the expected Hessian matrix of the NLLF.

CRLB matrix which is given by

$$\mathbf{CRLB}(\theta) = \mathbf{I}^{-1}(\theta)$$

The diagonal elements of the CRLB matrix are the lower bounds on the MSE values of individual estimate θ_n's. Further more, I have

$$\mathbf{C}_\theta - \mathbf{I}(\theta) \geq \mathbf{0} \qquad \text{Positive Semi-Definiteness}$$

Tranformation of Parameters: If $\alpha = g(\theta)$ is another estimated vector, its CRLB matrix is obtained by

$$\mathbf{CLRB}(\alpha) = \left[\frac{\partial g(\theta)}{\partial\theta}\right]\mathbf{CRLB}(\theta)\left[\frac{\partial g(\theta)}{\partial\theta}\right]^T = \left[\frac{\partial g(\theta)}{\partial\theta}\right]\mathbf{I}^{-1}(\theta)\left[\frac{\partial g(\theta)}{\partial\theta}\right]^T$$

General Gaussian Case: Let's say that the observations $\mathbf{x}$ are jointly Gaussian, parametrized by θ in the most general setting:

$$\mathbf{x} \sim \mathcal{N}(\mu(\theta),\, \mathbf{C}(\theta))$$

Then the Fisher information matrix's entries are computed as follows:

$$[\mathbf{I}(\theta)]_{ij} = \left[\frac{\partial\mu(\theta)}{\partial\theta_i}\right]^T\mathbf{C}^{-1}(\theta)\left[\frac{\partial\mu(\theta)}{\partial\theta_j}\right] + \frac{1}{2}\,\mathrm{Tr}\left[\mathbf{C}^{-1}(\theta)\frac{\partial\mathbf{C}(\theta)}{\partial\theta_i}\mathbf{C}^{-1}(\theta)\frac{\partial\mathbf{C}(\theta)}{\partial\theta_j}\right]$$

1.c Linear Models

Linear Models: A linear model is defined as

$$\mathbf{x} = \mathbf{H}\theta + \mathbf{b} + \mathbf{w}$$

where the observations $\mathbf{x}$ are taken to be an affine function of θ, corrupted by the zero-mean Gaussian noise vector $\mathbf{w} \sim \mathcal{N}(\mathbf{0},\, \mathbf{C})$. I assume the matrix $\mathbf{H}$ is full rank.

Importance: Linear models are important for the following reasons:

1. Some applications admit this model
2. Nonlinear systems can be handled through linearization
3. Optimal estimator is easy to find

Estimator: The MVUE and its covariance matrix are given as follows:

$$\hat{\theta}_{MVU} = \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{C}^{-1}(\mathbf{x}-\mathbf{b})$$
$$\mathbf{C}_{\hat{\theta}} = \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1} \iff \mathbf{I}(\theta) = \mathbf{H}^T\mathbf{C}^{-1}\mathbf{H} \qquad \text{(Achieves CRLB)}$$

1.d Best Linear Unbiased Estimate (BLUE)

Best Linear Unbiased Estimator: When the MVUE does not exist or difficult/impossible to find, I resort to restricting the estimator to be a linear one, and find the best (i.e. minimum variance) estimator among the linear ones. So, the *constraint* optimization problem of finding BLUE is formulated as follows:

$$\begin{aligned} \text{minimize} \quad & \mathbb{E}_{\mathbf{x}}\left\{(\mathbf{A}\mathbf{x}-\theta)^2\right\} && \text{(MSE minimization)} \\ \text{subject to} \quad & \mathbb{E}_{\mathbf{x}}\{\mathbf{A}\mathbf{x}\} = \hat{\theta} = \theta && \text{(Linearity \& Unbiasedness Constraint)} \end{aligned}$$

Thanks to the unbiasedness constraint, I know that the MSE is equal to the variance of the estimate, and so, it can be reformulated as

$$\begin{aligned} \text{minimize} \quad & \mathbb{E}_{\mathbf{x}}\left\{\left(\mathbf{A}\mathbf{x}-\hat{\theta}\right)^2\right\} = \text{var}(\hat{\theta}) && \text{(Variance minimization)} \\ \text{subject to} \quad & \mathbb{E}_{\mathbf{x}}\{\mathbf{A}\mathbf{x}\} = \hat{\theta} = \theta && \text{(Linearity \& Unbiasedness Constraint)} \end{aligned}$$

Linear Observations: If the observations $\mathbf{x}$ are generated linearly as

$$\mathbf{x} = \mathbf{H}\theta + \mathbf{w}$$

where $\mathbf{H}$ and the mean and covariance of $\mathbf{w}$ known, the BLUE and its covariance are given by

$$\hat{\theta}_{BLU} = \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{C}^{-1}\mathbf{x}$$
$$\mathbf{C}_{\hat{\theta}} = \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1}$$

If the noise $\mathbf{w}$ is Gaussian, then this is MVUE.

1.e Maximum Likelihood Estimation

Motivation: MVUE may not exist, and BLUE may not be applicable (over-restriction in modelling/insufficiently large MSE)

Advantages: It is a turn-the-crank method (algorithmic) and optimal for large data size.

Disadvantages: It is not optimal for small data sizes and can be computationally complex.

Rationale: Find the parameters that make the observed data the most likely data to have been observed.

Definition: Estimation problem becomes an optimization problem. Notice that passing the function to be optimized through a monotonically increasing function

$$\begin{aligned} \hat{\theta}_{ML} &= \underset{\theta\in\Theta}{\text{argmax}}\, p(\mathbf{x};\theta) && \text{(Maximize LF)} \\ &= \underset{\theta\in\Theta}{\text{argmax}}\, \ln p(\mathbf{x};\theta) && \text{(Maximize LLF)} \end{aligned}$$

$$= \underset{\theta\in\Theta}{\operatorname{argmin}} - \ln p(\mathbf{x};\theta) \qquad \text{(Minimize NLLF)}$$

Properties: The MLE is *asymptotically*

1. unbiased,
2. efficient (i.e. achieves CRLB),
3. normally distributed.

Efficiency: If a truly efficient estimator finds it, MLE finds it.

Asymptotic Gaussianity: Under the regularity conditions of

1. Existence of the first and second derivatives of the LLF
2. The usual
$$\mathbb{E}\left\{\frac{\partial}{\partial\theta}\ln p(\mathbf{x};\theta)\right\} = 0$$

the MLE is asymptotically Gaussian, i.e.

$$\hat{\theta}_{ML} \overset{a}{\sim} \mathcal{N}\big(\theta,\, I(\theta)^{-1}\big)$$

The N required to achieve this asymptotic behaviour is determined through Monte Carlo simulations.

Transformed Parameters: Let's assume I have a second quantity $\alpha = f(\theta)$ to be estimated. How to I find the MLE for α? The solution depends on the injectivity of the map f:

- If f is injective, then I can define f^{-1} and thus
$$\hat{\alpha}_{ML} = \underset{\alpha\in\mathrm{A}}{\operatorname{argmax}}\, p(\mathbf{x}; f^{-1}(\alpha))$$
- If f is *not* injective, then I need the modified likelihood function:
$$\overline{p}(\mathbf{x};\alpha) = \max_{\{\theta|\alpha=f(\theta)\}} p(\mathbf{x};\theta)$$
And so, the MLE of α maximizes this modified likelihood function.
$$\hat{\alpha}_{ML} = \underset{\alpha\in\mathrm{A}}{\operatorname{argmax}}\, \overline{p}(\mathbf{x};\alpha)$$

Invariance Property: The MLE of $\alpha = f(\theta)$ is given by $\hat{\alpha}_{ML} = f(\hat{\theta}_{ML})$. The function maximized by the MLE changes depending on the injectivity of f.

Efficiency: Even though I have θ efficiently estimated, its image under a nonlinear function $f(\theta)$ cannot be efficiently estimated.

Numerical Determination of MLE: I hope to have a closed form solution to the MLE determination problem, but In many cases, the derivative of the LLF may not allow for a closed form solution, which is why I need numerical methods.

Brute Force Method: Only if you are rich. It is sure to find the global maximum over a large and fine enough grid, but it demands a ton of computation power.

Iterative Methods: The way of the smart. The general methodology is to

1. Pick some initial estimate $\hat{\theta}_0$, and

2. Iteratively improve it using

$$\hat{\theta}_{k+1} = f(\hat{\theta}_k, \mathbf{x}) \qquad \text{such that} \qquad \lim_{k\to\infty} p(\mathbf{x}; \hat{\theta}_k) = \max_{\theta} p(\mathbf{x}; \theta)$$

The issues with this approach are the following

- It may not converge, or
- Even if it converges, it might converge to a local maximum instead of the global.

Newton-Raphson: This is a general method to find *the zero of a differentiable function*, let's say $f(x)$. The idea is to approximate f with its first-order Taylor series expansion, and equate that approximation to zero, which always has a solution, and use that as the next point of Taylor approximation:

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0) = 0 \quad \Rightarrow \quad x = x_0 - \frac{f(x_0)}{f'(x_0)} \qquad \text{(Scalar case)}$$

The function I am trying to equate to zero is the *derivative* of LLF, so that I can maximize it. Therefore the Newton-Raphson MLE iteration rule becomes the following (For the vector case):

$$\hat{\theta}_{k+1} = \hat{\theta}_k - \left[\frac{\partial^2}{\partial \theta^2} \ln p(\mathbf{x}; \hat{\theta}_k)\right]^{-1} \frac{\partial}{\partial \theta} \ln p(\mathbf{x}; \hat{\theta}_k) \qquad \text{(Repeat until convergence)}$$

Vector MLE: Vector parameter case has the analogue properties of the scalar case.

Asymptotic Gaussianity: If the LF satisfies similar regularity conditions, then MLE is asymptotically distributed with a Gaussian distribution:

$$\hat{\theta}_{ML} \overset{a}{\sim} \mathcal{N}\big(\theta, \mathbf{I}(\theta)^{-1}\big)$$

where $\mathbf{I}(\theta)$ is the Fisher Information Matrix.

Invariance: Invariance property also holds for the vector case: If $\alpha = f(\theta)$, then $\hat{\alpha}_{ML} = f(\hat{\theta}_{ML})$.

Gaussian MLE: If I know that the data vector $\mathbf{x}$ is Gaussian, the LF becomes

$$p(\mathbf{x}; \theta) = \mathcal{N}(\mathbf{x}; \mu(\theta), \mathbf{C}(\theta))$$

This information, however, is not guaranteed to give me a closed form solution to the LF maximization problem. If I further know that the data is *linearly generated*, i.e.

$$\mathbf{x} = \mathbf{H}\theta + \mathbf{w}, \quad \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}) \qquad \text{or alternatively} \qquad \mathbf{w} \sim \mathcal{N}(\mathbf{H}\theta, \mathbf{C})$$

then the MLE has the familiar closed form solution

$$\hat{\theta}_{ML} = \left(\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H}\right)^{-1} \mathbf{H}\mathbf{C}^{-1}\mathbf{x}$$

Then, I notice that the MLE is MVUE, and I can say that

$$\hat{\theta}_{ML} \sim \mathcal{N}\Big(\theta, \left(\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H}\right)^{-1}\Big) \qquad \text{(Exact distribution)}$$

2. Bayesian Philosophy: Random θ

Bayesian Approach: Assumes θ is random with pdf $p(\theta)$.

Bayesian MSE: The cost function in the Bayesian estimation framework is

$$\text{BMSE}(\hat{\theta}) = \mathbb{E}_{\mathbf{x},\theta}\Big\{(\hat{\theta}(\mathbf{x}) - \theta)^2\Big\} = \iint (\hat{\theta}(\mathbf{x}) - \theta)^2 p(\mathbf{x}, \theta) d\mathbf{x} d\theta$$

Bayesian Estimator and Minimum BMSE: The general estimator in the Bayesian framework that minimizes the above BMSE is given as

$$\hat{\theta}_{BMMSE} = \mathbb{E}\left\{\theta \,|\, \mathbf{x}\right\}$$

This estimator always exists, but not necessarily in closed form. Its minimum BMSE value is the following:

$$\text{BMSE}(\hat{\theta}_{BMMSE}) = \mathbb{E}_{\mathbf{x}}\{\text{var}\left\{\theta \,|\, \mathbf{x}\right\}\}$$

Mean and Variance:

$$\begin{aligned}
\mathbb{E}_{\mathbf{x}}\Big\{\hat{\theta}_{BMMSE}\Big\} &= \mathbb{E}_{\mathbf{x}}\{\mathbb{E}\left\{\theta \,|\, \mathbf{x}\right\}\} = \mathbb{E}\left\{\theta\right\} \\
\text{var}_{\mathbf{x}}\Big\{\hat{\theta}_{BMMSE}\Big\} &= \text{var}_{\mathbf{x}}\{\mathbb{E}\left\{\theta \,|\, \mathbf{x}\right\}\} \;\; \text{Use } \text{var}\left\{X\right\} = \mathbb{E}\left\{\text{var}\left\{X \,|\, Y\right\}\right\} + \text{var}\left\{\mathbb{E}\left\{X \,|\, Y\right\}\right\} \\
&= \text{var}\left\{\theta\right\} - \mathbb{E}_{\mathbf{x}}\{\text{var}\left\{\theta \,|\, \mathbf{x}\right\}\} \\
&= \text{var}\left\{\theta\right\} - \text{BMSE}(\hat{\theta}_{BMMSE})
\end{aligned}$$

Bayesian Linear Model: Consider the linear data generation model

$$\mathbf{x} = \mathbf{H}\theta + \mathbf{w}$$

where $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\mathbf{w}})$ and $\theta \sim \mathcal{N}(\mu_\theta, \mathbf{C}_\theta)$. Then the observations $\mathbf{w}$ and θ are jointly Gaussian because

$$\begin{bmatrix} \theta \\ \mathbf{x} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{H} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \theta \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{w} \end{bmatrix}$$

with

$$\begin{bmatrix} \theta \\ \mathbf{0} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_\theta \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{C}_\theta & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\right) \qquad \text{and} \qquad \begin{bmatrix} \mathbf{0} \\ \mathbf{w} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_\theta \end{bmatrix}\right)$$

Therefore I can write

$$\begin{bmatrix} \theta \\ \mathbf{x} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_\theta \\ \mathbf{H}\mu_\theta \end{bmatrix}, \begin{bmatrix} \mathbf{C}_\theta & \mathbf{C}_\theta\mathbf{H}^T \\ \mathbf{H}\mathbf{C}_\theta & \mathbf{H}\mathbf{C}_\theta\mathbf{H}^T + \mathbf{C}_{\mathbf{w}} \end{bmatrix}\right)$$

Using the conditioning rule for jointly Gaussian random vectors, I can obtain the conditional density as

$$\begin{aligned}
\theta \,|\, \mathbf{x} &\sim \mathcal{N}(\mu_{\theta\,|\,\mathbf{x}}, \mathbf{C}_{\theta\,|\,\mathbf{x}}) \\
\mu_{\theta\,|\,\mathbf{x}} &= \mu_\theta + \mathbf{C}_\theta\mathbf{H}^T\left(\mathbf{H}\mathbf{C}_\theta\mathbf{H}^T + \mathbf{C}_{\mathbf{w}}\right)^{-1}(\mathbf{x} - \mathbf{H}\mu_\theta) \\
\mathbf{C}_{\theta\,|\,\mathbf{x}} &= \mathbf{C}_\theta - \mathbf{C}_\theta\mathbf{H}^T\left(\mathbf{H}\mathbf{C}_\theta\mathbf{H}^T + \mathbf{C}_{\mathbf{w}}\right)^{-1}\mathbf{H}\mathbf{C}_\theta \\
&= \left(\mathbf{C}_\theta^{-1} + \mathbf{H}^T\mathbf{C}_{\mathbf{w}}^{-1}\mathbf{H}\right)^{-1} \qquad \text{(Matrix Inversion Lemma)}
\end{aligned}$$

2.a General Bayesian Estimators

Scalar Cost Functions: The above mean squared cost function is only one possible cost function to optimize. In general, if $\mathcal{C}(\theta-\hat{\theta})$ is a cost function, the Bayesian estimation problem is minimizing the expectation of this cost function:

$$\mathbb{E}_{\mathbf{x},\theta}\Big\{\mathcal{C}(\theta-\hat{\theta}(\mathbf{x}))\Big\} = \iint \mathcal{C}(\theta-\hat{\theta}(\mathbf{x}))p(\mathbf{x},\theta)d\theta d\mathbf{x}$$

Different cost functions yield different estimates, and a few possible ones are as follows for the scalar θ:

Quadratic:

$$\mathcal{C}(\theta-\hat{\theta}(\mathbf{x})) = (\theta-\hat{\theta}(\mathbf{x}))^2$$

The optimal estimator is then the posterior mean:

$$\hat{\theta}(\mathbf{x}) = \mathbb{E}\left\{\theta \,|\, \mathbf{x}\right\} = \int \theta p(\theta \,|\, \mathbf{x})d\theta \qquad \text{(MMSE Estimation)}$$

Absolute:

$$\mathcal{C}(\theta-\hat{\theta}(\mathbf{x})) = |\theta-\hat{\theta}(\mathbf{x})|$$

The optimal estimator is then the posterior median θ_{med}, given by

$$\int_{-\infty}^{\theta_{med}} p(\theta \,|\, \mathbf{x})d\theta = \int_{\theta_{med}}^{\infty} p(\theta \,|\, \mathbf{x})d\theta$$

Hit-or-Miss:

$$\mathcal{C}(\theta-\hat{\theta}(\mathbf{x})) = \begin{cases} 0, & |\theta-\hat{\theta}(\mathbf{x})| < \delta \\ 1, & |\theta-\hat{\theta}(\mathbf{x})| \geq \delta \end{cases} \qquad \text{(Always thought as } \delta \to 0\text{)}$$

The optimal estimator is then the posterior mode θ_{mode}, given by

$$\theta_{mode} = \underset{\theta\in\Theta}{\operatorname{argmax}}\, p(\theta \,|\, \mathbf{x}) \qquad \text{(MAP Estimation)}$$

Vector Cost Functions: The cost functions for the vector case are similar, though a bit harder to handle.

Vector MMSE (Quadratic): If the quadratic cost function is used for each θ_i in the parameter vector $\boldsymbol{\theta} = \begin{bmatrix}\theta_1 & \cdots & \theta_p\end{bmatrix}^T$, then the vector MMSE estimator is immediately the vector extension of the scalar case:

$$\hat{\boldsymbol{\theta}}_{MMSE} = \mathbb{E}\left\{\boldsymbol{\theta} \,|\, \mathbf{x}\right\}$$

The BMSE for each parameter θ_i is similarly expressed as

$$\text{BMSE}(\hat{\theta}_i) = \int \left[\mathbf{C}_{\boldsymbol{\theta}\,|\,\mathbf{x}}\right]_{ii} p(\mathbf{x})d\mathbf{x} \qquad \text{where} \qquad \mathbf{C}_{\boldsymbol{\theta}\,|\,\mathbf{x}} = \mathbb{E}\left\{\left[\boldsymbol{\theta}-\mathbb{E}\left\{\boldsymbol{\theta} \,|\, \mathbf{x}\right\}\right]\left[\boldsymbol{\theta}-\mathbb{E}\left\{\boldsymbol{\theta} \,|\, \mathbf{x}\right\}\right]^T\right\}$$

Properties:

1. MMSE estimation commutes over affine mappings.

$$\boldsymbol{\alpha} = \mathbf{A}\boldsymbol{\theta}+\mathbf{b} \qquad \text{then} \qquad \hat{\boldsymbol{\alpha}}_{MMSE} = \mathbf{A}\hat{\boldsymbol{\theta}}_{MMSE}+\mathbf{b}$$

2. Independent Gaussian datasets contribute additively to the estimate. If $\boldsymbol{\theta}$, $\mathbf{x}_1$ and $\mathbf{x}_2$

are jointly Gaussian with $\mathbf{x}_1$ and $\mathbf{x}_2$ independent, then

$$\hat{\theta} = \mathbb{E}\{\theta\} + \mathbf{C}_{\theta\mathbf{x}_1}\mathbf{C}_{\mathbf{x}_1}^{-1}(\mathbf{x}_1 - \mathbb{E}\{\mathbf{x}_1\}) + \mathbf{C}_{\theta\mathbf{x}_2}\mathbf{C}_{\mathbf{x}_2}^{-1}(\mathbf{x}_2 - \mathbb{E}\{\mathbf{x}_2\})$$

3. Jointly Gaussian case results in an affine estimator.

$$\hat{\theta} = \mathbf{P}\mathbf{x} + \mathbf{m}$$

Vector MAP (Hit-or-Miss): The extension here is not as straight-forward as the MMSE case, as the MAP estimates for all parameters θ_i may not be the same. For vector MAP, the vector cost function is defined as

$$\mathcal{C}(\theta - \hat{\theta}) = \begin{cases} 0, & \|\theta - \hat{\theta}\| < \delta \\ 1, & \|\theta - \hat{\theta}\| \geq \delta \end{cases}$$

The norm here is commonly selected as the ℓ_2 (inner product) norm. In this case, the vector MAP estimator becomes

$$\hat{\theta}_{MAP} = \underset{\theta\in\Theta}{\operatorname{argmax}}\, p(\theta \,|\, \mathbf{x})$$

This whole-vector MAP estimate is usually *not* the same as element-wise scalar MAP estimate vector.

Performance of MMSE Estimation: Call the vector estimation error (which is a general definition, not just for MMSE) as $\epsilon := \theta - \hat{\theta}$. If $\hat{\theta} = \hat{\theta}_{MMSE} = \mathbb{E}\{\theta \,|\, \mathbf{x}\}$, then

$$\mathbb{E}\{\epsilon\} = \mathbf{0} \qquad \text{and} \qquad \mathbf{C}_\epsilon = \mathbb{E}_{\mathbf{x}}\{\mathbf{C}_{\theta\,|\,\mathbf{x}}\}$$

$\mathbf{C}_\epsilon$ is also called the Bayesian MSE Matrix, as the diagonal contains the BMSE's of each estimate.

Jointly Gaussian case: If the data vector $\mathbf{x}$ and the parameter vector θ are jointly Gaussian, then the covariance matrix of the error matrix does not depend on $\mathbf{x}$ but only on its covariance matrix $\mathbf{C}_\mathbf{x}$, meaning

$$\mathbf{C}_\epsilon = \mathbb{E}_{\mathbf{x}}\{\mathbf{C}_{\theta\,|\,\mathbf{x}}\} = \mathbf{C}_{\theta\,|\,\mathbf{x}} = \mathbf{C}_\theta - \mathbf{C}_{\theta\mathbf{x}}\mathbf{C}_\mathbf{x}^{-1}\mathbf{C}_{\mathbf{x}\theta}$$

2.b Linear Bayesian Estimators

Core Idea: Restrict the form of the estimator to an affine one

$$\hat{\theta}_{LMMSE} = \mathbf{K}\mathbf{x}$$

and find the optimal such mapping in the Bayesian MSE sense. While I may lose from generality and restrict to a (possibly narrow) set of estimators, LMMSE estimation only requires the first- and second-order moments to work, not whole pdf's.

Orthogonality Principle: The most commonly used fact in the derivation of the coming results is the orthogonality principle, which states that the estimation error vector is orthogonal to the observations:

$$(\theta - \hat{\theta}_{LMMSE}) \perp \mathbf{x} \Rightarrow \mathbb{E}\left\{(\theta - \hat{\theta}_{LMMSE})^T\mathbf{x}\right\} = 0$$

Vector LMMSE Solution: The solution to the LMMSE estimator in the vector case is as follows:

$$\hat{\theta}_{LMMSE} = \mathbb{E}\{\theta\} + \mathbf{C}_{\theta\mathbf{x}}\mathbf{C}_\mathbf{x}^{-1}(\mathbf{x} - \mathbb{E}\{\mathbf{x}\})$$

The BMSE matrix for this estimate is

$$\mathbf{M}_{\hat{\theta}} = \mathbb{E}\left\{(\theta - \hat{\theta})(\theta - \hat{\theta})^T\right\} = \mathbf{C}_{\theta\,|\,\mathbf{x}} = \mathbf{C}_\theta - \mathbf{C}_{\theta\mathbf{x}}\mathbf{C}_{\mathbf{x}}^{-1}\mathbf{C}_{\mathbf{x}\theta}$$

Notice how it matches with the joint Gaussian MMSE solution.

Bayesian Gauss-Markov Theorem: If the data is modelled as

$$\mathbf{x} = \mathbf{H}\theta + \mathbf{w}$$

where

- θ has mean μ_θ and covariance $\mathbf{C}_\theta$,
- $\mathbf{w}$ has mean $\mathbf{0}$ and covariance $\mathbf{C}_\mathbf{w}$
- θ and $\mathbf{w}$ are uncorrelated,

then the LMMSE estimator of θ is

$$\begin{aligned}\hat{\theta}_{LMMSE} &= \mu_\theta + \mathbf{C}_\theta\mathbf{H}^T(\mathbf{H}\mathbf{C}_\theta\mathbf{H}^T + \mathbf{C}_\mathbf{w})^{-1}(\mathbf{x} - \mathbf{H}\mu_\theta)\\ &= \mu_\theta + (\mathbf{C}_\theta^{-1} + \mathbf{H}^T\mathbf{C}_\mathbf{w}^{-1}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{C}_\mathbf{w}^{-1}(\mathbf{x} - \mathbf{H}\mu_\theta)\end{aligned}$$

The error vector $\epsilon = \theta - \hat{\theta}$ is then zero-mean and has covariance matrix $\mathbf{C}_\epsilon$ expressed as

$$\begin{aligned}\mathbf{C}_\epsilon &= \mathbb{E}_{\mathbf{x},\theta}\{\epsilon\epsilon^T\}\\ &= \mathbf{C}_\theta - \mathbf{C}_\theta\mathbf{H}^T(\mathbf{H}\mathbf{C}_\theta\mathbf{H}^T + \mathbf{C}_\mathbf{w})^{-1}\mathbf{H}\mathbf{C}_\theta\\ &= (\mathbf{C}_\theta^{-1} + \mathbf{H}^T\mathbf{C}_\mathbf{w}^{-1}\mathbf{H})^{-1}\end{aligned}$$

Sequential LMMSE Estimation: Make use of the new observation $x[n]$ to update the previous estimate $\hat{\theta}_{n-1}$ to $\hat{\theta}_n$.

Model & Goal: Suppose I have the following data generation model:

$$\mathbf{x}[n] = \mathbf{H}[n]\theta + \mathbf{w}[n]$$

where

$$\begin{aligned}\mathbf{x}[n]_{(n+1)\times 1} &= \begin{bmatrix}\mathbf{x}[n-1]\\ x[n]\end{bmatrix} && \text{is the accumulated data vector}\\ \mathbf{H}[n]_{(n+1)\times p} &= \begin{bmatrix}\mathbf{H}[n-1]\\ \mathbf{h}_n^T\end{bmatrix} && \text{is the accumulated matrix of observation models}\\ \theta_{p\times 1} & && \text{is the unknown parameter vector to be estimated}\\ \mathbf{w}[n] \sim \mathcal{N}(0,\, \sigma_n^2) & && \text{is the white noise vector with known covariance}\end{aligned}$$

The goal is to benefit from the recursive structure

$$\begin{bmatrix}\mathbf{x}[n-1]\\ x[n]\end{bmatrix} = \begin{bmatrix}\mathbf{H}[n-1]\\ \mathbf{h}_n^T\end{bmatrix}\theta + \mathbf{w}[n]$$

in obtaining the current estimate $\hat{\theta}[n]$ using the previous estimate $\hat{\theta}[n-1]$.

Core Idea: Make a prediction of the current estimate, then update it with the "novel information" provided by the latest observation. This novel information is provided by the so called Innovations Sequence.

Innovations Sequence: The innovations sequence is best described in the framework of vector spaces with subspaces, projections and orthogonality.

Inner Product Space of Random Variables: The set of random vectors form an inner product space with the inner product defined as $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbb{E}\left\{\mathbf{y}^T\mathbf{x}\right\}$.

"Novel Information:" In this sense, the novelty of information one random variable has with respect to another one is orthogonality (uncorrelatedness if one is zero-mean). Therefore by applying the Gram-Schmidt orthogonalization process, I obtain the innovations sequence $\tilde{x}[n]$:

$$\begin{aligned}
\tilde{x}[0] &= x[0] \\
\tilde{x}[1] &= x[1] - \underbrace{\frac{\langle x[1], \tilde{x}[0]\rangle}{\langle \tilde{x}[0], \tilde{x}[0]\rangle}\tilde{x}[0]}_{:=\hat{x}[1\,|\,0]} \\
\tilde{x}[2] &= x[2] - \underbrace{\frac{\langle x[2], \tilde{x}[0]\rangle}{\langle \tilde{x}[0], \tilde{x}[0]\rangle}\tilde{x}[0] - \frac{\langle x[2], \tilde{x}[1]\rangle}{\langle \tilde{x}[1], \tilde{x}[1]\rangle}\tilde{x}[1]}_{:=\hat{x}[2\,|\,1]} \\
&\vdots \\
\tilde{x}[n] &= x[n] - \underbrace{\sum_{i=0}^{n-1} \frac{\langle x[n], \tilde{x}[i]\rangle}{\langle \tilde{x}[i], \tilde{x}[i]\rangle}\tilde{x}[i]}_{:=\hat{x}[n\,|\,n-1]}
\end{aligned}$$

Sequential LMMSE: Here is the sequential LMMSE algorithm based on the idea of innovations sequence:

Input:

- The prior first and second moments of θ: $\mathbb{E}\{\theta\}$ and $\mathbf{C}_\theta$
- The observation model $\mathbf{h}_n$ and σ_n^2 for all n

Initialization:

- $\hat{\theta}_{-1} = \mathbb{E}\{\theta\}$
- $\mathbf{M}_{-1} = \mathbf{C}_\theta$

Loop:

1. Calculate the innovation:

$$\begin{aligned}
\tilde{x}[n] &= x[n] - \hat{x}[n\,|\,n-1] \\
&= x[n] - \mathbf{h}_n^T\hat{\theta}_{n-1}
\end{aligned}$$

2. Calculate the gain vector $\mathbf{k}_n$:

$$\mathbf{k}_n = \frac{\mathbf{M}_{n-1}\mathbf{h}_n}{\sigma_n^2 + \mathbf{h}_n^T\mathbf{M}_{n-1}\mathbf{h}_n}$$

No data needed for this calculation!

3. Calculate current estimate $\hat{\theta}_n$:

$$\begin{aligned}
\hat{\theta}_n &= \hat{\theta}_{n-1} + \mathbf{k}_n\tilde{x}[n] \\
&= \hat{\theta}_{n-1} + \mathbf{k}_n\left(x[n] - \mathbf{h}_n^T\hat{\theta}_{n-1}\right)
\end{aligned}$$

4. Calculate the current BMSE matrix $\mathbf{M}_n$:

$$\mathbf{M}_n = \left(\mathbf{I} - \mathbf{k}_n\mathbf{h}_n^T\right)\mathbf{M}_{n-1}$$

No data needed for this calculation!

This loop rule can be derived by making the assumption that all variables are jointly Gaussian, with the state space rules given by

$$\begin{aligned} \theta_n &= \theta_{n-1} \\ x[n] &= h_n^T \theta_n + w[n] \qquad \text{where} \qquad w[n] \sim \mathcal{N}(0,\, \sigma_n^2) \end{aligned}$$

The joint density of θ_n and $x[n]$ can therefore be written as

$$\begin{bmatrix} \theta_n \\ x[n] \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \hat{\theta}_{n-1} \\ \mathbf{h}_n^T \hat{\theta}_{n-1} \end{bmatrix}, \begin{bmatrix} \mathbf{M}_{n-1} & \mathbf{M}_{n-1}\mathbf{h}_n \\ \mathbf{h}_N^T \mathbf{M}_{n-1} & \mathbf{h}_N^T \mathbf{M}_{n-1}\mathbf{h}_n + \sigma_n^2 \end{bmatrix} \right)$$

Then the rule is directly obtained by writing the conditional density of $\theta_n \,|\, x[n]$:

$$\begin{aligned} \theta_n \,|\, x[n] &\sim \mathcal{N}\left(\hat{\theta}_n,\, \mathbf{M}_n\right) \\ \hat{\theta}_n &= \hat{\theta}_{n-1} + \underbrace{\frac{\mathbf{M}_{n-1}\mathbf{h}_n}{\mathbf{h}_N^T \mathbf{M}_{n-1}\mathbf{h}_n + \sigma_n^2}}_{:=\mathbf{k}_n} \left(x[n] - \mathbf{h}_n^T \hat{\theta}_{n-1}\right) = \hat{\theta}_{n-1} + \mathbf{k}_n \left(x[n] - \mathbf{h}_n^T \hat{\theta}_{n-1}\right) \\ \mathbf{M}_n &= \mathbf{M}_{n-1} - \overbrace{\frac{\mathbf{M}_{n-1}\mathbf{h}_n}{\mathbf{h}_N^T \mathbf{M}_{n-1}\mathbf{h}_n + \sigma_n^2}}^{\mathbf{k}_n} \mathbf{h}_n^T \mathbf{M}_{n-1} = \mathbf{M}_{n-1} - \mathbf{k}_n \mathbf{h}_n^T \mathbf{M}_{n-1} \\ &= (\mathbf{I} - \mathbf{k}_n \mathbf{h}_n^T)\mathbf{M}_{n-1} \end{aligned}$$

Wiener Filter: The problem is to estimate a signal $s[n]$ using observations $x[n]$ corrupted by noise.

$$x[n] = s[n] + w[n]$$

Here, all signals are assumed to be WSS and zero-mean, with covariance/autocorrelation matrices of appropriate order given by $\mathbf{R_w} = \mathbf{C_w}$ and $\mathbf{R_s} = \mathbf{C_s}$.

Filtering, Smoothing, Prediction: Based on the Wiener filter problem formulation, I obtain three problems:

Filtering where you estimate $s[n]$ based on the observations $x[0:n]$.

Smoothing where you estimate $s[k]$ for $k = 0, ..., n-1$ based on the observations $x[0:n]$.

Prediction where you estimate $s[n+k]$ for $k = 1, 2, ...$ based on the observations $x[0:n]$.

General Solution: All of the problems above rely on the general LMMSE estimator solution, which is

$$\hat{\theta} = \mathbf{C}_{\theta\mathbf{x}} \mathbf{C}_{\mathbf{x}}^{-1} \mathbf{x}$$

2.c Kalman Filters (Kay's Version)

2.c.1 Kalman Filter (KF)

State Model: The dynamical state space I have is the following:

$$\mathbf{s}[n] = \mathbf{A}\mathbf{s}[n-1] + \mathbf{B}\mathbf{u}[n] \qquad \text{State Model}$$

where

$\mathbf{s}[n]$ is a vector Gauss-Markov process

$\mathbf{A}$ is the state transition matrix with eigenvalues in the unit circle

$\mathbf{B}$ is the input matrix that "colors" the noise
$\mathbf{u}[n] \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ is the driving noise vector
$\mathbf{s}[-1] \sim \mathcal{N}(\mu_\mathbf{s}, \mathbf{C_s})$ is the initial state independent of $\mathbf{u}[n]$

Propagation of Mean and Covariance: At each step, the state mean and covariance progresses by the following, rather intuitive rules:

$$\mathbb{E}\{\mathbf{s}[n]\} = \mathbf{A}\,\mathbb{E}\{\mathbf{s}[n-1]\}$$
$$\mathbf{C_s}[n] = \mathbf{A}\mathbf{C_s}[n-1]\mathbf{A}^T + \mathbf{B}\mathbf{Q}\mathbf{B}^T$$

However, only using the state model is not a very good estimation scheme, as the covariance matrix will keep on growing as seen above.

Observation Model: To improve our estimate of the state, I make some observations modelled as below:

$$\mathbf{x}[n] = \mathbf{H}[n]\mathbf{s}[n] + \mathbf{w}[n] \qquad \text{Observation Model}$$

where

$\mathbf{x}[n]$ is a vector of observations
$\mathbf{H}[n]$ is the observation function allowed to be time-varying
$\mathbf{w}[n] \sim \mathcal{N}(\mathbf{0}, \mathbf{C}[n])$ is the AWGN on the observations

Kalman Filter (Kay, no derivation): What I aim to do is recursively generate an estimate $\hat{\mathbf{s}}[n]$ of $\mathbf{s}[n]$, using the previous estimate $\hat{\mathbf{s}}[n-1]$ and the newly available observation $\mathbf{x}[n]$. This gives me the Kalman Filter.

Input:

- State Model parameters: $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{Q}$
- Observation Model parameters: $\mathbf{H}[n]$ for all n and $\mathbf{C}[n]$
- Initialization parameters: μ_s and $\mathbf{C_s}$

Initialization:

- $\hat{\mathbf{s}}[-1\,|\,-1] = \mu_\mathbf{s}$
- $\mathbf{M}[-1\,|\,-1] = \mathbf{C_s}$

Loop:

Prediction: Follow the rules of state propagation.

$$\hat{\mathbf{s}}[n\,|\,n-1] = \mathbf{A}\hat{\mathbf{s}}[n-1\,|\,n-1]$$
$$\mathbf{M}[n\,|\,n-1] = \mathbf{A}\mathbf{M}[n-1\,|\,n-1]\mathbf{A}^T + \mathbf{B}\mathbf{Q}\mathbf{B}^T$$

Kalman Gain:

$$\mathbf{K}[n] = \mathbf{M}[n\,|\,n-1]\mathbf{H}^T[n]\Big(\mathbf{C}[n] + \mathbf{H}[n]\mathbf{M}[n\,|\,n-1]\mathbf{H}^T[n]\Big)^{-1}$$

Measurement Update:

$$\hat{\mathbf{s}}[n\,|\,n] = \hat{\mathbf{s}}[n\,|\,n-1] + \mathbf{K}[n]\left(\mathbf{x}[n] - \underbrace{\mathbf{H}[n]\hat{\mathbf{s}}[n\,|\,n-1]}_{\hat{\mathbf{x}}[n\,|\,n-1]}\right)$$
$$\mathbf{M}[n\,|\,n] = (\mathbf{I} - \mathbf{K}[n]\mathbf{H}[n])\,\mathbf{M}[n\,|\,n-1]$$

2.c.2 Extended Kalman Filter (EKF)

Shortcoming of KF: While Kalman Filter is a powerful tool, it fails to account for nonlinear state and observation models.

Core Idea: Make a first-order Taylor approximation of the nonlinear functions and use them.

Dynamical Model: With the nonlinear state and observation models, the dynamical model is expressed as follows:

$$\begin{aligned}\mathbf{s}[n] &= \mathbf{a}(\mathbf{s}[n-1]) + \mathbf{B}\mathbf{u}[n] \\ \mathbf{x}[n] &= \mathbf{h}_n(\mathbf{s}[n]) + \mathbf{w}[n]\end{aligned}$$

Linearization: It is important to note that the linearizations are not made around a constant value but around the previous prediction or state estimate.

$$\mathbf{a}(\mathbf{s}[n-1]) \approx \mathbf{a}(\hat{\mathbf{s}}[n-1\,|\,n-1]) + \underbrace{\left[\frac{\partial \mathbf{a}}{\partial \mathbf{s}[n-1]}\bigg|_{\mathbf{s}[n-1]=\hat{\mathbf{s}}[n-1\,|\,s-1]}\right]}_{:=\mathbf{A}[n-1]} (\mathbf{s}[n-1] - \hat{\mathbf{s}}[n-1\,|\,s-1])$$

$$\mathbf{h}_n(\mathbf{s}[n]) \approx \mathbf{h}_n(\hat{\mathbf{s}}[n\,|\,n-1]) + \underbrace{\left[\frac{\partial \mathbf{h}_n}{\partial \mathbf{s}[n]}\bigg|_{\mathbf{s}[n]=\hat{\mathbf{s}}[n\,|\,s-1]}\right]}_{:=\mathbf{H}[n]} (\mathbf{s}[n] - \hat{\mathbf{s}}[n\,|\,s-1])$$

Extended Kalman Filter: The linearization above yields not necessarily a linear mapping but an affine one, i.e. a linear one with an offset. This offset however is known at each step, and the derivations work similarly as in KF.

Input:

- State Model parameters: $\mathbf{a}(\cdot)$, $\mathbf{B}$ and $\mathbf{Q}$
- Observation Model parameters: $\mathbf{h}_n(\cdot)$ for all n and $\mathbf{C}[n]$
- Initialization parameters: μ_s and $\mathbf{C_s}$

Initialization:

- $\hat{\mathbf{s}}[-1\,|\,-1] = \mu_\mathbf{s}$
- $\mathbf{M}[-1\,|\,-1] = \mathbf{C_s}$

Loop:

Prediction: Follow the rules of state propagation according to the nonlinear state transition function.

$$\hat{\mathbf{s}}[n\,|\,n-1] = \mathbf{a}(\hat{\mathbf{s}}[n-1\,|\,n-1])$$

Linearization:

$$\mathbf{A}[n-1] = \frac{\partial \mathbf{a}}{\partial \mathbf{s}[n-1]}\bigg|_{\mathbf{s}[n-1]=\hat{\mathbf{s}}[n-1\,|\,s-1]} \qquad \mathbf{H}[n] = \frac{\partial \mathbf{h}_n}{\partial \mathbf{s}[n]}\bigg|_{\mathbf{s}[n]=\hat{\mathbf{s}}[n\,|\,s-1]}$$

Prediction MSE:

$$\mathbf{M}[n\,|\,n-1] = \mathbf{A}[n-1]\mathbf{M}[n-1\,|\,n-1]\mathbf{A}^T[n-1] + \mathbf{B}\mathbf{Q}\mathbf{B}^T$$

Kalman Gain: Same as KF.

$$\mathbf{K}[n] = \mathbf{M}[n\,|\,n-1]\mathbf{H}^T[n]\Big(\mathbf{C}[n] + \mathbf{H}[n]\mathbf{M}[n\,|\,n-1]\mathbf{H}^T[n]\Big)^{-1}$$

Measurement Update: Same as KF. Notice how even if I use the approximation, I obtain the exact evaluation of the nonlinear observation function.

$$\hat{\mathbf{s}}[n\,|\,n] = \hat{\mathbf{s}}[n\,|\,n-1] + \mathbf{K}[n]\left(\mathbf{x}[n] - \mathbf{h}_n(\hat{\mathbf{s}}[n\,|\,n-1])\right)$$
$$\mathbf{M}[n\,|\,n] = \left(\mathbf{I} - \mathbf{K}[n]\mathbf{H}[n]\right)\mathbf{M}[n\,|\,n-1]$$

2.d Bayesian Filtering (Särkkä's Version)

Different Way of Expressing Models: The state space model above in Kay's version is just one way of expressing the state and observation models. In the most generic way possible, I can them in two ways (in Särkkä's notation) as shown below.

Nonlinear State Space: In this form, the noise disturbance is expressed as by iid. samples from a relevant distribution.

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{q}_{k-1}) \qquad \text{State Model}$$
$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{r}_k) \qquad \text{Observation Model}$$

Generic Markovian State Space: In this form, the state and observation at each time k are samples from a conditional distribution that implicitly models the noise disturbance.

$$\mathbf{x}_k \sim p(\mathbf{x}_k\,|\,\mathbf{x}_{k-1}) \qquad \text{State Model}$$
$$\mathbf{y}_k \sim p(\mathbf{y}_k\,|\,\mathbf{x}_k) \qquad \text{Measurement Model}$$
$$\mathbf{x}_0 \sim p(\mathbf{x}_0) \qquad \text{Initial/Prior Distribution}$$

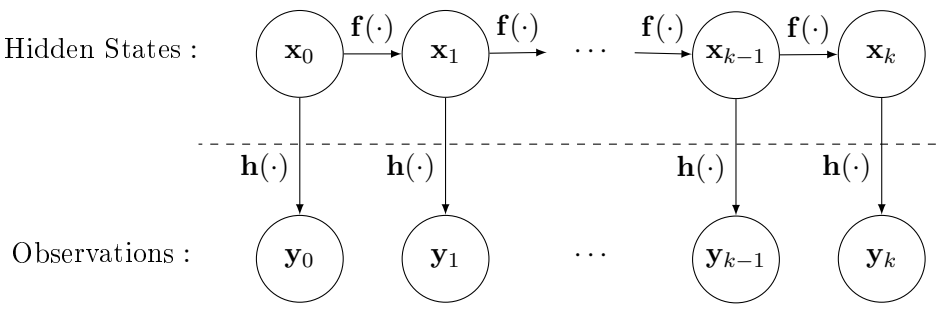
Hidden Markov Models (HMM): The Markov assumption specifies the conditional (in)dependency of the state progression. It assumes that the state $\mathbf{x}_k$ depends only on the previous state $\mathbf{x}_{k-1}$ and is conditionally independent of all the previous states given $\mathbf{x}_{k-1}$:

$$p(\mathbf{x}_k\,|\,\mathbf{x}_{k-1}, \ldots, \mathbf{x}_0) = p(\mathbf{x}_k\,|\,\mathbf{x}_{k-1})$$

In a Hidden Markov Model, such as the linear Gaussian model below, the state observations are "hidden" and can only be inferred through another random variable $\mathbf{y}_k$. Observations $\mathbf{y}_k$ at each time instance only depends on the state $\mathbf{x}_k$ at that time and is conditionally independent from other observations and all previous states:

$$p(\mathbf{y}_k\,|\,\mathbf{x}_k, \ldots, \mathbf{x}_0, \mathbf{y}_{k-1}, \ldots, \mathbf{y}_0) = p(\mathbf{y}_k\,|\,\mathbf{x}_k)$$

This dependency can be summarized in the diagram below, where arrows indicate conditional dependency:



General Bayesian Filtering: The aim of Bayesian filtering is to generate the current conditional

state distribution $p(\mathbf{x}_k \,|\, \mathbf{y}_{0:k})$ using the previous conditional state distribution $p(\mathbf{x}_{k-1} \,|\, \mathbf{y}_{0:k-1})$ and the observation model $p(\mathbf{y}_k \,|\, \mathbf{x}_k)$.

Input:

- State space model distributions $p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})$ and $p(\mathbf{y}_k \,|\, \mathbf{x}_k)$
- Initial distribution $p(\mathbf{x}_0)$

Initialization:

- $p(\mathbf{x}_0 \,|\, \mathbf{x}_{-1}) = p(\mathbf{x}_0)$

Loop:

Time Update: Chapman-Kolmogorov Equation

$$p(\mathbf{x}_k \,|\, \mathbf{y}_{0:k-1}) = \int p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} \,|\, \mathbf{y}_{0:k-1}) d\mathbf{x}_{k-1}$$

Measurement Update: Bayes' Rule

$$\begin{aligned} p(\mathbf{x}_k \,|\, \mathbf{y}_{0:k}) &= \frac{p(\mathbf{y}_k \,|\, \mathbf{x}_k) p(\mathbf{x}_k \,|\, \mathbf{y}_{0:k-1})}{\int p(\mathbf{y}_k \,|\, \mathbf{x}_k) p(\mathbf{x}_k \,|\, \mathbf{y}_{0:k-1}) d\mathbf{x}_k} \\ &\propto p(\mathbf{y}_k \,|\, \mathbf{x}_k) p(\mathbf{x}_k \,|\, \mathbf{y}_{0:k-1}) \end{aligned}$$

Linear Gaussian State Space Models: A specific instance of HMM are linear Gaussian state space models:

$$\begin{aligned} \mathbf{x}_k &= \mathbf{A}_{k-1}\mathbf{x}_{k-1} + \mathbf{q}_{k-1} & \mathbf{q}_{k-1} &\sim \mathcal{N}(\mathbf{0},\, \mathbf{Q}_{k-1}) \\ \mathbf{y}_k &= \mathbf{H}_k\mathbf{x}_k + \mathbf{r}_k & \mathbf{r}_k &\sim \mathcal{N}(\mathbf{0},\, \mathbf{R}_k) \end{aligned}$$

Equivalently, in terms of pdf's:

$$\begin{aligned} p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}) &= \mathcal{N}(\mathbf{x}_k; \mathbf{A}_{k-1}\mathbf{x}_{k-1}, \mathbf{Q}_{k-1}) \\ p(\mathbf{y}_k \,|\, \mathbf{x}_k) &= \mathcal{N}(\mathbf{y}_k; \mathbf{H}_k\mathbf{x}_k, \mathbf{R}_k) \end{aligned}$$

Kalman Filter (Särkkä, with derivation): Here I make the derivation of the Kalman Filter equations, based on the joint Gaussian marginalization and conditioning rules given in the preliminaries.
Because the Gaussian distribution is uniquely determined by its mean vector and covariance matrix, the recursion runs over those parameters.

Input:

- State model parameters: $\mathbf{A}_{k-1}$ and $\mathbf{Q}_{k-1}$
- Measurement model parameters: $\mathbf{H}_k$ and $\mathbf{R}_k$
- Initial mean and covariance: $\mathbf{m}_0$ and $\mathbf{P}_0$

Initialization:

- Start with the initial mean and covariance: $\mathbf{m}_0$ and $\mathbf{P}_0$

Loop:

Time Update: I know that the previous state $\mathbf{x}_{k-1}$ and the current state $\mathbf{x}_k$ are jointly Gaussian due to the current linear relation between them:

$$\begin{bmatrix} \mathbf{x}_{k-1} \\ \mathbf{x}_k \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{A}_{k-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{k-1} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{q}_{k-1} \end{bmatrix}$$

where

$$\begin{bmatrix}\mathbf{x}_{k-1}\\ \mathbf{0}\end{bmatrix} \mid \mathbf{y}_{1:k-1} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{m}_{k-1}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{P}_{k-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\right) \qquad \begin{bmatrix}\mathbf{0}\\ \mathbf{q}_{k-1}\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{Q}_{k-1}\end{bmatrix}\right)$$

Then I obtain the following joint conditional distribution:

$$\begin{bmatrix}\mathbf{x}_{k-1}\\ \mathbf{x}_{k}\end{bmatrix} \mid \mathbf{y}_{1:k-1} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{m}_{k-1}\\ \mathbf{A}_{k-1}\mathbf{m}_{k-1}\end{bmatrix}, \begin{bmatrix}\mathbf{P}_{k-1} & \mathbf{P}_{k-1}\mathbf{A}_{k-1}^T\\ \mathbf{A}_{k-1}\mathbf{P}_{k-1} & \mathbf{A}_{k-1}\mathbf{P}_{k-1}\mathbf{A}_{k-1}^T + \mathbf{Q}_{k-1}\end{bmatrix}\right)$$

Finally by the marginalization rule, I obtain the time update equation:

$$\begin{aligned}\mathbf{x}_k \mid \mathbf{y}_{1:k-1} &\sim \mathcal{N}(\mathbf{m}_{k\mid k-1}, \mathbf{P}_{k\mid k-1})\\ \mathbf{m}_{k\mid k-1} &= \mathbf{A}_{k-1}\mathbf{m}_{k-1}\\ \mathbf{P}_{k\mid k-1} &= \mathbf{A}_{k-1}\mathbf{P}_{k-1}\mathbf{A}_{k-1}^T + \mathbf{Q}_{k-1}\end{aligned}$$

Measurement Update: I already know that the observation vector $\mathbf{y}_k$ and the current state are jointly Gaussian because they are related by the following linear relation:

$$\begin{bmatrix}\mathbf{x}_k\\ \mathbf{y}_k\end{bmatrix} = \begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{H}_k & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{x}_k\\ \mathbf{0}\end{bmatrix} + \begin{bmatrix}\mathbf{0}\\ \mathbf{r}_k\end{bmatrix}$$

where

$$\begin{bmatrix}\mathbf{x}_k\\ \mathbf{0}\end{bmatrix} \mid \mathbf{y}_{1:k-1} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{m}_{k\mid k-1}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{P}_{k\mid k-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\right) \qquad \begin{bmatrix}\mathbf{0}\\ \mathbf{r}_k\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{R}_k\end{bmatrix}\right)$$

Therefore their joint conditional distribution becomes the following:

$$\begin{bmatrix}\mathbf{x}_k\\ \mathbf{y}_k\end{bmatrix} \mid \mathbf{y}_{1:k-1} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{m}_{k\mid k-1}\\ \mathbf{H}_k\mathbf{m}_{k\mid k-1}\end{bmatrix}, \begin{bmatrix}\mathbf{P}_{k\mid k-1} & \mathbf{P}_{k\mid k-1}\mathbf{H}_k^T\\ \mathbf{H}_k\mathbf{P}_{k\mid k-1} & \mathbf{H}_k\mathbf{P}_{k\mid k-1}\mathbf{H}_k^T + \mathbf{R}_k\end{bmatrix}\right)$$

Now, using the conditioning rule of joint Gaussians, I obtain the measurement update equation:

$$\begin{aligned}\mathbf{x}_k \mid \overbrace{\mathbf{y}_{1:k-1}, \mathbf{y}_k}^{\mathbf{y}_{1:k}} &\sim \mathcal{N}(\mathbf{m}_k, \mathbf{P}_k)\\ \mathbf{m}_k &= \mathbf{m}_{k\mid k-1} + \mathbf{P}_{k\mid k-1}\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_{k\mid k-1}\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1}(\mathbf{y}_k - \mathbf{H}_k\mathbf{m}_{k\mid k-1})\\ \mathbf{P}_k &= \mathbf{P}_{k\mid k-1} - \mathbf{P}_{k\mid k-1}\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_{k\mid k-1}\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1}\mathbf{H}_k\mathbf{P}_{k\mid k-1}\end{aligned}$$

Now, calling

$$\mathbf{K}_k = \mathbf{P}_{k\mid k-1}\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_{k\mid k-1}\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1}$$

as the Kalman gain, the equations simplify significantly:

$$\begin{aligned}\mathbf{m}_k &= \mathbf{m}_{k\mid k-1} + \mathbf{K}_k(\mathbf{y}_k - \mathbf{H}_k\mathbf{m}_{k\mid k-1})\\ \mathbf{P}_k &= \mathbf{P}_{k\mid k-1} - \mathbf{K}_k\mathbf{H}_k\mathbf{P}_{k\mid k-1} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k\mid k-1}\end{aligned}$$

Extended Kalman Filter (EKF): As before, this time the state transition function are not linear but general, but the noise is still additive WGN:

$$\begin{aligned}\mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{q}_{k-1} \qquad & \mathbf{q}_{k-1} &\sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})\\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k) + \mathbf{r}_k & \mathbf{r}_k &\sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)\end{aligned}$$

The key point will be to linearize them around the previous mean:

$$\mathbf{f}(\mathbf{x}_{k-1}) \approx \mathbf{f}(\mathbf{m}_{k-1}) + \mathbf{F}_{k-1}\big|_{\mathbf{m}_{k-1}}(\mathbf{x}_{k-1} - \mathbf{m}_{k-1})$$
$$\mathbf{h}(\mathbf{x}_k) \approx \mathbf{h}(\mathbf{m}_{k\,|\,k-1}) + \mathbf{H}_k\big|_{\mathbf{m}_{k\,|\,k-1}}(\mathbf{x}_k - \mathbf{m}_{k\,|\,k-1})$$

where $\mathbf{F}_{k-1}\big|_{\mathbf{m}_{k-1}}$ and $\mathbf{H}_k\big|_{\mathbf{m}_{k\,|\,k-1}}$ are the Jacobian matrices evaluated at $\mathbf{x}_{k-1} = \mathbf{m}_{k-1}$ and $\mathbf{x}_k = \mathbf{x}_{k\,|\,k-1}$, respectively.

Input:

- State model parameters: $\mathbf{f}(\cdot)$ and $\mathbf{Q}_{k-1}$
- Measurement model parameters: $\mathbf{h}(\cdot)$ and $\mathbf{R}_k$
- Initial mean and covariance: $\mathbf{m}_0$ and $\mathbf{P}_0$

Initialization:

- Start with the initial mean and covariance: $\mathbf{m}_0$ and $\mathbf{P}_0$

Loop:

Time Update: This time, the conditional joint distribution of $\mathbf{x}_{k-1}$ and $\mathbf{x}_k$ is not exact but approximate:

$$\begin{bmatrix}\mathbf{x}_{k-1}\\ \mathbf{x}_k\end{bmatrix} \,|\, \mathbf{y}_{1:k-1} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{m}_{k-1}\\ \mathbf{f}(\mathbf{m}_{k-1})\end{bmatrix}, \begin{bmatrix}\mathbf{P}_{k-1} & \mathbf{P}_{k-1}\mathbf{F}_{k-1}^T\big|_{\mathbf{m}_{k-1}}\\ \mathbf{F}_{k-1}\big|_{\mathbf{m}_{k-1}}\mathbf{P}_{k-1} & \mathbf{F}_{k-1}\big|_{\mathbf{m}_{k-1}}\mathbf{P}_{k-1}\mathbf{F}_{k-1}^T\big|_{\mathbf{m}_{k-1}} + \mathbf{Q}_{k-1}\end{bmatrix}\right)$$

Notice that a new $\mathbf{F}_{k-1}\big|_{\mathbf{m}_{k-1}}$ has to be computed at each loop iteration, which is a computational load. Then, again by the marginalization rule, I obtain the time update equation:

$$\begin{aligned}\mathbf{x}_k \,|\, \mathbf{y}_{1:k-1} &\sim \mathcal{N}(\mathbf{m}_{k\,|\,k-1}, \mathbf{P}_{k\,|\,k-1})\\ \mathbf{m}_{k\,|\,k-1} &= \mathbf{f}(\mathbf{m}_{k-1})\\ \mathbf{P}_{k\,|\,k-1} &= \mathbf{F}_{k-1}\big|_{\mathbf{m}_{k-1}}\mathbf{P}_{k-1}\mathbf{F}_{k-1}^T\big|_{\mathbf{m}_{k-1}} + \mathbf{Q}_{k-1}\end{aligned}$$

Measurement Update: Similar to the time update, the conditional joint distribution of $\mathbf{y}_k$ and $\mathbf{x}_k$ is approximate:

$$\begin{bmatrix}\mathbf{x}_k\\ \mathbf{y}_k\end{bmatrix} \,|\, \mathbf{y}_{1:k-1} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{m}_{k\,|\,k-1}\\ \mathbf{h}(\mathbf{m}_{k\,|\,k-1})\end{bmatrix}, \begin{bmatrix}\mathbf{P}_{k\,|\,k-1} & \mathbf{P}_{k\,|\,k-1}\mathbf{H}_k^T\big|_{\mathbf{m}_{k\,|\,k-1}}\\ \mathbf{H}_k\big|_{\mathbf{m}_{k\,|\,k-1}}\mathbf{P}_{k\,|\,k-1} & \mathbf{H}_k\big|_{\mathbf{m}_{k\,|\,k-1}}\mathbf{P}_{k\,|\,k-1}\mathbf{H}_k^T\big|_{\mathbf{m}_{k\,|\,k-1}} + \mathbf{R}_k\end{bmatrix}\right)$$

Again, a new matrix $\mathbf{H}_k\big|_{\mathbf{m}_{k\,|\,k-1}}$ has to be computed at each loop iteration. Again, using the conditioning rule of joint Gaussians, I obtain the measurement update equation:

$$\begin{aligned}\mathbf{x}_k \,|\, \overbrace{\mathbf{y}_{1:k-1}, \mathbf{y}_k}^{\mathbf{y}_{1:k}} &\sim \mathcal{N}(\mathbf{m}_k, \mathbf{P}_k)\\ \mathbf{K}_k &= \mathbf{P}_{k\,|\,k-1}\mathbf{H}_k^T\big|_{\mathbf{m}_{k\,|\,k-1}}\left(\mathbf{H}_k\big|_{\mathbf{m}_{k\,|\,k-1}}\mathbf{P}_{k\,|\,k-1}\mathbf{H}_k^T\big|_{\mathbf{m}_{k\,|\,k-1}} + \mathbf{R}_k\right)^{-1}\\ \mathbf{m}_k &= \mathbf{m}_{k\,|\,k-1} + \mathbf{K}_k(\mathbf{y}_k - \mathbf{h}(\mathbf{m}_{k\,|\,k-1}))\\ \mathbf{P}_k &= \mathbf{P}_{k\,|\,k-1} - \mathbf{K}_k\mathbf{H}_k\big|_{\mathbf{m}_{k\,|\,k-1}}\mathbf{P}_{k\,|\,k-1} = \left(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k\big|_{\mathbf{m}_{k\,|\,k-1}}\right)\mathbf{P}_{k\,|\,k-1}\end{aligned}$$