# Lecture Notes 4
# Vector Detection and Estimation

- Vector Detection
  - Broadcasting on a tree
  - Detection for Vector AGN Channel
- Vector Linear Estimation
  - Linear Innovation Sequence
  - Kalman Filter

# Vector Detection

- Let the signal $\Theta = \theta_0$ with probability $p_0$ and $\Theta = \theta_1$ with probability $p_1 = 1 - p_0$
- We observe the RV $\mathbf{Y}$, where $\mathbf{Y}|\{\Theta = \theta_0\} \sim f_{\mathbf{Y}|\Theta}(\mathbf{y}|\theta_0)$ and $\mathbf{Y}|\{\Theta = \theta_1\} \sim f_{\mathbf{Y}|\Theta}(\mathbf{y}|\theta_1)$
- We wish to find the estimate $\hat{\Theta}(\mathbf{Y})$ that minimizes the probability of detection error $\mathrm{P}\{\hat{\Theta} \neq \Theta\}$
- The optimal estimate is obtained using the MAP decoder

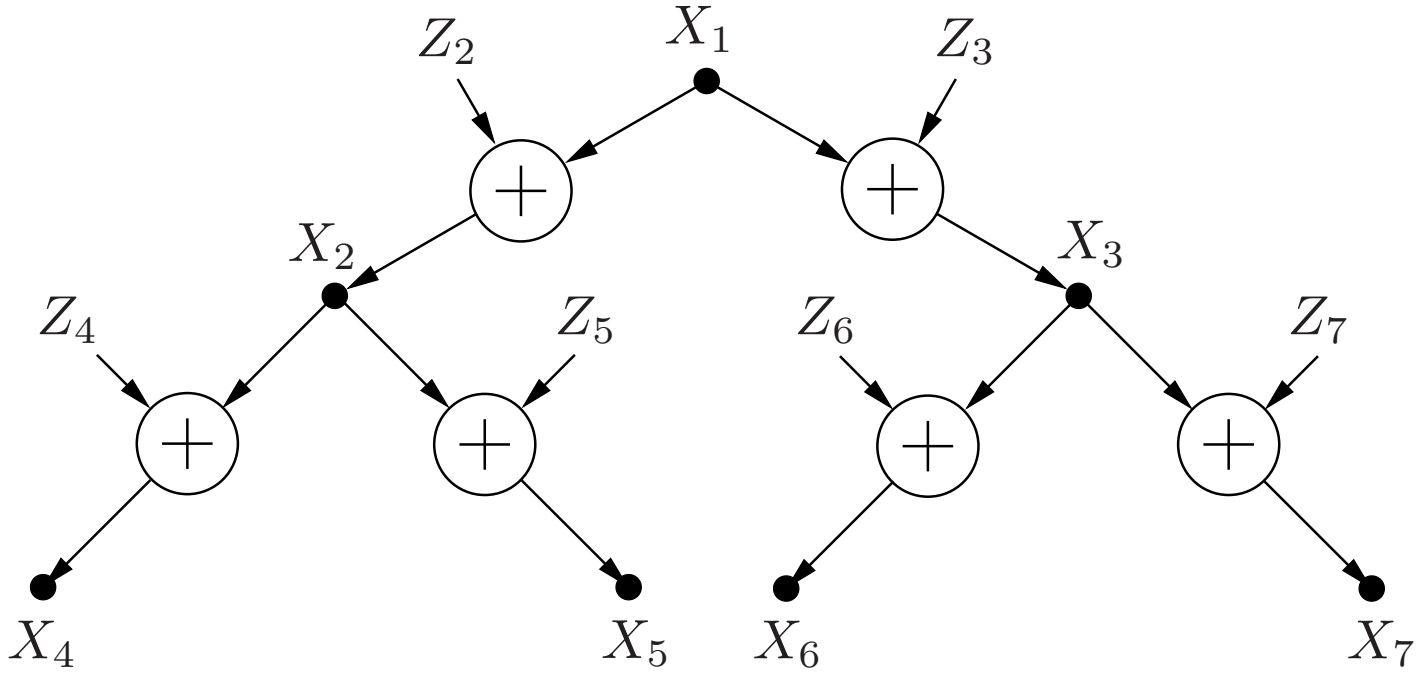
$$\hat{\Theta}(\mathbf{y}) = \begin{cases} \theta_0 & \text{if } \frac{p_{\Theta|\mathbf{Y}}(\theta_0|\mathbf{y})}{p_{\Theta|\mathbf{Y}}(\theta_1|\mathbf{y})} > 1 \\ \theta_1 & \text{otherwise} \end{cases}$$

- When $p_0 = p_1 = 1/2$, the MAP decoder reduces to the ML decoder

$$\hat{\Theta}(\mathbf{y}) = \begin{cases} \theta_0 & \text{if } \frac{f_{\mathbf{Y}|\Theta}(\mathbf{y}|\theta_0)}{f_{\mathbf{Y}|\Theta}(\mathbf{y}|\theta_1)} > 1 \\ \theta_1 & \text{otherwise} \end{cases}$$

# Broadcasting on a Tree

- Consider a complete binary broadcasting tree of finite depth $k$




  - The root node is assigned a r.v. $X_1 \sim \mathrm{Bern}(1/2)$ (the signal)
  - Denote the two children of each non-leaf node $i$ as $l_i$ and $r_i$ (e.g., for $i = 1$, $l_1 = 2$ and $r_1 = 3$)

- A random variable is assigned to each non-root node as follows

$$X_{l_i} = X_i \oplus Z_{l_i},$$
$$X_{r_i} = X_i \oplus Z_{r_i},$$

where $Z_1, Z_2, \ldots$ are i.i.d. $\mathrm{Bern}(\epsilon)$, $\epsilon \leq 1/2$, r.v.s independent of $X_1$

That is, the r.v. assigned to a node is the output of a binary symmetric channel (BSC) whose input is the r.v. of its parent

- Denote the set of leaf r.v.s that are descendants of node $i$ as $\mathbf{X}_i$ (e.g., for $i = 1$, $\mathbf{X}_1 = (X_4, X_5, X_6, X_7)$, and for $i = 4$, $\mathbf{X}_4 = X_4$ in figure)
- We observe the leaf node r.v.s $\mathbf{X}_1$ and wish to find the estimate $\hat{X}_1(\mathbf{X}_1)$ that minimizes the probability of error $P_e = \mathrm{P}\{\hat{X}_1 \neq X_1\}$
- This problem is a simple example of the reconstruction on trees problem, which arises in computational evolutionary biology, statistical physics, and theoretical computer science. A question of interest in these fields is under what condition on the channel noise can $X_1$ be reconstructed with $P_e < 1/2$ as the tree depth $k \to \infty$

  The broadcasting on trees problem is an example of graphical models in which random variables dependencies are specified by a graph (STAT 375, CS 228)

- Since $X_1 \sim \text{Bern}(1/2)$, the optimal estimate is obtained using the ML decoder

$$\hat{X}_1(\mathbf{X}_1) = \begin{cases} 0 & \text{if } \frac{p_{\mathbf{X}_1|X_1}(\mathbf{x}_1|0)}{p_{\mathbf{X}_1|X_1}(\mathbf{x}_1|1)} > 1 \\ 1 & \text{otherwise} \end{cases}$$

- Because of the special structure of the observation r.v.s, the optimal estimate can be computed using a fast iterative message passing algorithm

- Define

$$L_{i,0} = p_{\mathbf{X}_i|X_i}(\mathbf{x}_i|0)$$
$$L_{i,1} = p_{\mathbf{X}_i|X_i}(\mathbf{x}_i|1)$$

- Then the ML estimate can be written as

$$\hat{X}_1(\mathbf{X}_1) = \begin{cases} 0 & \text{if } \frac{L_{1,0}}{L_{1,1}} > 1 \\ 1 & \text{otherwise} \end{cases}$$

- We now show that $L_{1,0}, L_{1,1}$ can be computed (in order of the number of nodes in the tree) by iteratively computing the intermediate likelihoods $L_{i,0}, L_{i,1}$ beginning with the leaf nodes for which $L_{i,0} = 1 - x_i$ and $L_{i,1} = x_i$

- By the law of total probability, for a non-leaf node $i$, we can write

$$
\begin{aligned}
L_{i,0} &= p_{\mathbf{X}_i|X_i}(\mathbf{x}_i|0) \\
&= \quad p_{X_{l_i},X_{r_i}|X_i}(0,0|0)\; p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,0,0) \\
&\quad + p_{X_{l_i},X_{r_i}|X_i}(0,1|0)\; p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,0,1) \\
&\quad + p_{X_{l_i},X_{r_i}|X_i}(1,0|0)\; p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,1,0) \\
&\quad + p_{X_{l_i},X_{r_i}|X_i}(1,1|0)\; p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,1,1) \\
&= \quad p_{X_{l_i}|X_i}(0|0)p_{X_{r_i}|X_i}(0|0)\; p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,0,0) \\
&\quad + p_{X_{l_i}|X_i}(0|0)\; p_{X_{r_i}|X_i}(1|0)p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,0,1) \\
&\quad + p_{X_{l_i}|X_i}(1|0)\; p_{X_{r_i}|X_i}(0|0)p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,1,0) \\
&\quad + p_{X_{l_i}|X_i}(1|0)p_{X_{r_i}|X_i}(1|0)p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,1,1) \text{ conditional independence} \\
&= \quad \bar{\epsilon}^2 \cdot p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,0,0) + \bar{\epsilon}\epsilon \cdot p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,0,1) \\
&\quad + \epsilon\bar{\epsilon} \cdot p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,1,0) + \epsilon^2 \cdot p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|0,1,1),
\end{aligned}
$$

  where $\bar{\epsilon} = 1 - \epsilon$. $L_{i,1}$ can be expressed similarly

- Now, since $\mathbf{X}_i = (\mathbf{X}_{l_i}, \mathbf{X}_{r_i})$, by conditional independence,

$$\begin{aligned} p_{\mathbf{X}_i|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_i|x_i, x_{l_i}, x_{r_i}) &= p_{\mathbf{X}_{l_i},\mathbf{X}_{r_i}|X_i,X_{l_i},X_{r_i}}(\mathbf{x}_{l_i}, \mathbf{x}_{r_i}|x_i, x_{l_i}, x_{r_i}) \\ &= p_{\mathbf{X}_{l_i}|X_{l_i}}(\mathbf{x}_{l_i}|x_{l_i}) p_{\mathbf{X}_{r_i}|X_{r_i}}(\mathbf{x}_{r_i}|x_{r_i}) \end{aligned}$$

- Hence we obtain the following iteratively equations

$$\begin{aligned} L_{i,0} &= (\bar{\epsilon} L_{l_i,0} + \epsilon L_{l_i,1})(\bar{\epsilon} L_{r_i,0} + \epsilon L_{r_i,1}), \\ L_{i,1} &= (\epsilon L_{l_i,0} + \bar{\epsilon} L_{l_i,1})(\epsilon L_{r_i,0} + \bar{\epsilon} L_{r_i,1}), \end{aligned}$$

  where, at the leaf nodes

$$\begin{aligned} L_{i,0} &= p_{X_i|X_i}(x_i|0) = 1 - x_i \\ L_{i,1} &= p_{X_i|X_i}(x_i|1) = x_i \end{aligned}$$

- Hence to compute $L_{1,0}$ and $L_{1,1}$, we start with the likelihoods at each leaf node, then compute the likelihoods for the nodes at level $k-1$, and so on until we arrive at node 1

# Detection for Vector Additive Gaussian Noise Channel

- Consider the vector additive Gaussian noise (AGN) channel

$$\mathbf{Y} = \mathbf{\Theta} + \mathbf{Z},$$

  where the signal $\mathbf{\Theta} = \boldsymbol{\theta}_0$, an $n$-dimensional real vector, with probability $1/2$ and $\mathbf{\Theta} = \boldsymbol{\theta}_1$ with probability $1/2$, and the noise $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{Z}})$ are independent

- We observe $\mathbf{y}$ and wish to find the estimate $\hat{\mathbf{\Theta}}(\mathbf{Y})$ that minimizes the probability of decoding error $\mathrm{P}\{\hat{\mathbf{\Theta}} \neq \mathbf{\Theta}\}$

- First assume that $\Sigma_{\mathbf{Z}} = NI$, i.e., additive white Gaussian noise channel

- The optimal decoding rule is the ML decoder. Define the log likelihood ratio

$$\Lambda(\mathbf{y}) = \ln \frac{f(\mathbf{y}|\boldsymbol{\theta}_0)}{f(\mathbf{y}|\boldsymbol{\theta}_1)}$$

  Then, the ML decoder is

$$\hat{\mathbf{\Theta}}(\mathbf{y}) = \begin{cases} \boldsymbol{\theta}_0 & \text{if } \Lambda(\mathbf{y}) > 0 \\ \boldsymbol{\theta}_1 & \text{otherwise} \end{cases}$$

Now,

$$\Lambda(\mathbf{y}) = \frac{1}{2N}\left[(\mathbf{y}-\boldsymbol{\theta}_1)^T(\mathbf{y}-\boldsymbol{\theta}_1) - (\mathbf{y}-\boldsymbol{\theta}_0)^T(\mathbf{y}-\boldsymbol{\theta}_0)\right]$$

- Hence, the ML decoder reduces to the minimum distance decoder

$$\hat{\boldsymbol{\Theta}}(\mathbf{y}) = \begin{cases} \boldsymbol{\theta}_0 & \text{if } ||\mathbf{y}-\boldsymbol{\theta}_0|| < ||\mathbf{y}-\boldsymbol{\theta}_1|| \\ \boldsymbol{\theta}_1 & \text{otherwise} \end{cases}$$

- We can simplify this further to

$$\hat{\boldsymbol{\Theta}}(\mathbf{y}) = \begin{cases} \boldsymbol{\theta}_0 & \text{if } \mathbf{y}^T(\boldsymbol{\theta}_1-\boldsymbol{\theta}_0) < \frac{1}{2}(\boldsymbol{\theta}_1^T\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0^T\boldsymbol{\theta}_0) \\ \boldsymbol{\theta}_1 & \text{otherwise} \end{cases}$$

Hence, the decision depends only on the value of a scalar r.v. $W = \mathbf{Y}^T(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0)$. Such r.v. is referred to as a sufficient statistic for the optimal decoder. Further,

$$W|\{\boldsymbol{\Theta} = \boldsymbol{\theta}_0\} \sim \mathcal{N}(\boldsymbol{\theta}_0^T(\boldsymbol{\theta}_1-\boldsymbol{\theta}_0), N(\boldsymbol{\theta}_1-\boldsymbol{\theta}_0)^T(\boldsymbol{\theta}_1-\boldsymbol{\theta}_0)),$$

$$W|\{\boldsymbol{\Theta} = \boldsymbol{\theta}_1\} \sim \mathcal{N}(\boldsymbol{\theta}_1^T(\boldsymbol{\theta}_1-\boldsymbol{\theta}_0), N(\boldsymbol{\theta}_1-\boldsymbol{\theta}_0)^T(\boldsymbol{\theta}_1-\boldsymbol{\theta}_0))$$

- Assuming that the signals have the same power, i.e., $\boldsymbol{\theta}_0^T\boldsymbol{\theta}_0 = \boldsymbol{\theta}_1^T\boldsymbol{\theta}_1 = P$, the optimal decoding rule reduces to the matched filter decoder (receiver)

$$\hat{\boldsymbol{\Theta}}(\mathbf{y}) = \begin{cases} \boldsymbol{\theta}_0 & \text{if } \mathbf{y}^T(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0) < 0 \\ \boldsymbol{\theta}_1 & \text{otherwise,} \end{cases}$$

that is,

$$\hat{\Theta}(\mathbf{y}) = \begin{cases} \boldsymbol{\theta}_0 & \text{if } w < 0 \\ \boldsymbol{\theta}_1 & \text{if } w \geq 0 \end{cases}$$

This is the same as the optimal rule for the scalar case discussed in Lecture notes 1! The minimum probability of error is

$$P_e = Q\left(\frac{P - \boldsymbol{\theta}_0^T\boldsymbol{\theta}_1}{\sqrt{2N(P - \boldsymbol{\theta}_0^T\boldsymbol{\theta}_1)}}\right)$$

$$= Q\left(\sqrt{\frac{P - \boldsymbol{\theta}_0^T\boldsymbol{\theta}_1}{2N}}\right)$$

This is minimized by using antipodal signals $\boldsymbol{\theta}_0 = -\boldsymbol{\theta}_1$, which yields

$$P_e = Q\left(\sqrt{\frac{P}{N}}\right)$$

Exactly the same as scalar antipodal signals

- Now suppose that the noise is not white, i.e., $\Sigma_{\mathbf{Z}} \neq NI$. Then the ML decoder reduces to

$$\hat{\boldsymbol{\Theta}}(\mathbf{y}) = \begin{cases} \boldsymbol{\theta}_0 & \text{if } (\mathbf{y}-\boldsymbol{\theta}_0)^T\Sigma_{\mathbf{Z}}^{-1}(\mathbf{y}-\boldsymbol{\theta}_0) < (\mathbf{y}-\boldsymbol{\theta}_1)^T\Sigma_{\mathbf{Z}}^{-1}(\mathbf{y}-\boldsymbol{\theta}_1) \\ \boldsymbol{\theta}_1 & \text{otherwise} \end{cases}$$

Now, let $\mathbf{y}' = \Sigma_{\mathbf{Z}}^{-1/2}\mathbf{y}$ and $\boldsymbol{\theta}_i' = \Sigma_{\mathbf{Z}}^{-1/2}\boldsymbol{\theta}_i'$ for $i = 0, 1$, then the rule becomes the same as that for the white noise case

$$\hat{\boldsymbol{\Theta}}(\mathbf{y}) = \begin{cases} \boldsymbol{\theta}_0 & \text{if } ||\mathbf{y}'-\boldsymbol{\theta}_0'|| < ||\mathbf{y}'-\boldsymbol{\theta}_1'|| \\ \boldsymbol{\theta}_1 & \text{otherwise} \end{cases}$$

and can be simplified to the scalar case as before

- Thus, the optimal decoder is to first multiply $\mathbf{Y}$ by $\Sigma_{\mathbf{Z}}^{-1/2}$ to obtain $\mathbf{Y}'$ and then to apply the optimal rule for the white noise case with the transformed signals $\boldsymbol{\theta}_i' = \Sigma_{\mathbf{Z}}^{-1/2}\boldsymbol{\theta}_i$, $i = 0, 1$

# Vector Linear Estimation

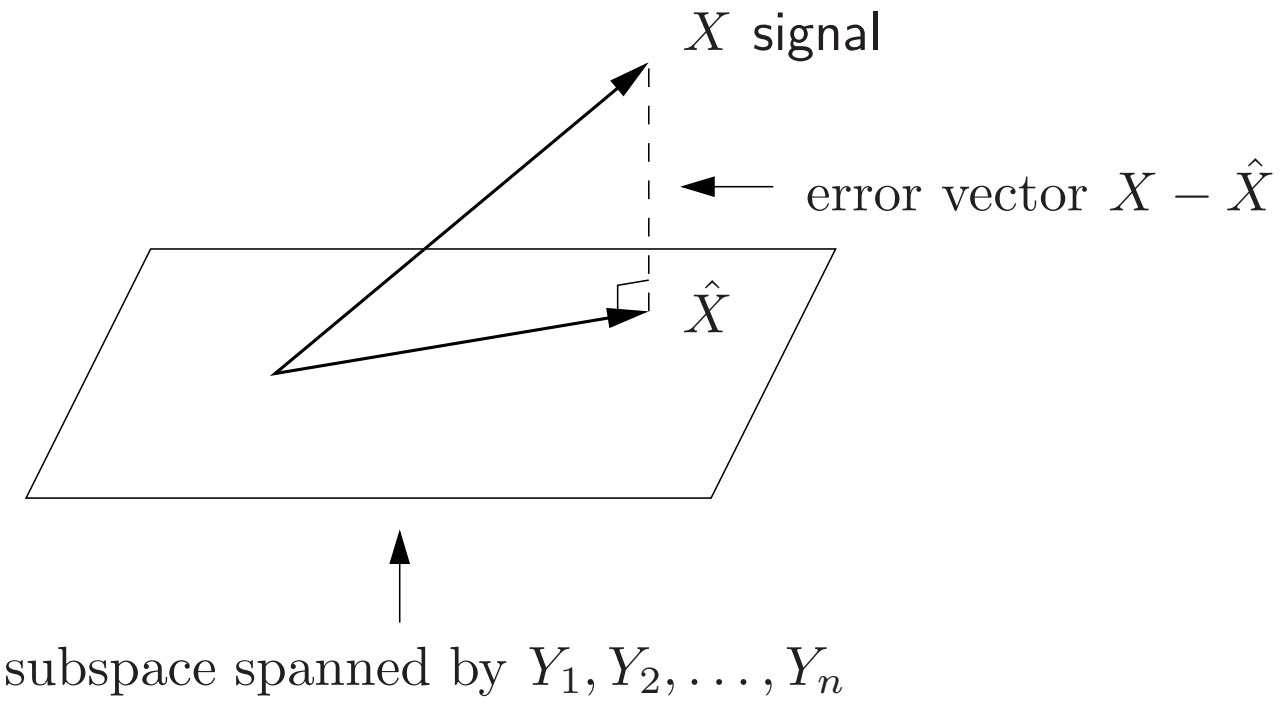
- Let $X \sim f_X(x)$ be a r.v. representing the signal and let $\mathbf{Y}$ be an $n$-dimensional RV representing the observations
- The minimum MSE estimate of $X$ given $\mathbf{Y}$ is the conditional expectation $\mathrm{E}(X \,|\, \mathbf{Y})$. This is often not practical to compute either because the conditional pdf of $X$ given $\mathbf{Y}$ is not known or because of high computational cost
- The MMSE linear (or affine) estimate is easier to find since it depends only on the means, variances, and covariances of the r.v.s involved
- To find the MMSE linear estimate, first assume that $\mathrm{E}(X) = 0$ and $\mathrm{E}(\mathbf{Y}) = \mathbf{0}$. The problem reduces to finding a real $n$-vector $\mathbf{h}$ such that

$$\hat{X} \;=\; \mathbf{h}^T \mathbf{Y} \;=\; \sum_{i=1}^{n} h_i Y_i$$

  minimizes the $\mathrm{MSE} = \mathrm{E}\left[(X - \hat{X})^2\right]$

# MMSE Linear Estimate via Orthogonality Principle

- To find $\hat{X}$ we use the orthogonality principle: we view the r.v.s $X, Y_1, Y_2, \ldots, Y_n$ as vectors in the inner product space consisting of all zero mean r.v.s defined over the underlying probability space
- The linear estimation problem reduces to a geometry problem: find the vector $\hat{X}$ that is closest to $X$ (in norm of error $X - \hat{X}$)



- To minimize $\mathrm{MSE} = \|X - \hat{X}\|^2$, we choose $\hat{X}$ so that the error vector $X - \hat{X}$ is orthogonal to the subspace spanned by the observations $Y_1, Y_2, \ldots, Y_n$, i.e.,

$$\mathrm{E}\left[(X - \hat{X})Y_i\right] = 0\,, \quad i = 1, 2, \ldots, n\,,$$

hence

$$\mathrm{E}(Y_i X) = \mathrm{E}(Y_i \hat{X}) = \sum_{j=1}^{n} h_j \,\mathrm{E}(Y_i Y_j)\,, \quad i = 1, 2, \ldots, n$$

- Define the cross covariance of $\mathbf{Y}$ and $X$ as the $n$-vector

$$\Sigma_{\mathbf{Y}X} = \mathrm{E}\left[(\mathbf{Y} - \mathrm{E}(\mathbf{Y}))(X - \mathrm{E}(X))\right] = \begin{bmatrix} \sigma_{Y_1 X} \\ \sigma_{Y_2 X} \\ \vdots \\ \sigma_{Y_n X} \end{bmatrix}$$

For $n = 1$ this is simply the covariance

- The above equations can be written in vector form as $\Sigma_{\mathbf{Y}}\mathbf{h} = \Sigma_{\mathbf{Y}X}$
- If $\Sigma_{\mathbf{Y}}$ is nonsingular, we can solve the equations to obtain $\mathbf{h} = \Sigma_{\mathbf{Y}}^{-1}\Sigma_{\mathbf{Y}X}$

- Thus, if $\Sigma_{\mathbf{Y}}$ is nonsingular then the best linear MSE estimate is: $\hat{X} = \mathbf{h}^T\mathbf{Y} = \Sigma_{\mathbf{Y}X}^T \Sigma_{\mathbf{Y}}^{-1}\mathbf{Y}$
- Compare this to the scalar case, where $\hat{X} = \dfrac{\mathrm{Cov}(X,Y)}{\sigma_Y^2} Y$
- Now to find the minimum MSE, consider

$$\begin{aligned}
\mathrm{MSE} &= \mathrm{E}\left[(X-\hat{X})^2\right] \\
&= \mathrm{E}\left[(X-\hat{X})X\right] - \mathrm{E}\left[(X-\hat{X})\hat{X}\right] \\
&= \mathrm{E}\left[(X-\hat{X})X\right], \text{ since by orthogonality } (X-\hat{X}) \perp \hat{X} \\
&= \mathrm{E}(X^2) - \mathrm{E}(\hat{X}X) \\
&= \mathrm{Var}(X) - \mathrm{E}\left(\Sigma_{\mathbf{Y}X}^T \Sigma_{\mathbf{Y}}^{-1}\mathbf{Y}X\right) = \mathrm{Var}(X) - \Sigma_{\mathbf{Y}X}^T \Sigma_{\mathbf{Y}}^{-1}\Sigma_{\mathbf{Y}X}
\end{aligned}$$

- Compare this to the scalar case, where minimum MSE is $\mathrm{Var}(X) - \dfrac{\mathrm{Cov}(X,Y)^2}{\sigma_Y^2}$
- If $X$ or $\mathbf{Y}$ have nonzero mean, the MMSE affine estimate $\hat{X} = h_0 + \mathbf{h}^T\mathbf{Y}$ is determined by first finding the MMSE linear estimate of $X - \mathrm{E}(X)$ given $\mathbf{Y} - \mathrm{E}(\mathbf{Y})$ (minimum MSE for $\hat{X}'$ and $\hat{X}$ are the same), which is $\hat{X}' = \Sigma_{\mathbf{Y}X}^T \Sigma_{\mathbf{Y}}^{-1}(\mathbf{Y} - \mathrm{E}(\mathbf{Y}))$, and then setting $\hat{X} = \hat{X}' + \mathrm{E}(X)$ (since $\mathrm{E}(\hat{X}) = \mathrm{E}(X)$ is necessary)

# Example

- Let $X$ be the r.v. representing a signal with mean $\mu$ and variance $P$. The observations are $Y_i = X + Z_i$, for $i = 1, 2, \ldots, n$, where the $Z_i$ are zero mean uncorrelated noise with variance $N$, and $X$ and $Z_i$ are also uncorrelated

  Find the MMSE linear estimate of $X$ given $\mathbf{Y}$ and its MSE

- For $n = 1$, we already know that $\hat{X}_1 = \dfrac{P}{P+N} Y_1 + \dfrac{N}{P+N} \mu$

- To find the MMSE linear estimate for general $n$, first let $X' = X - \mu$ and $Y_i' = Y_i - \mu$. Thus $X'$ and $\mathbf{Y}'$ are zero mean

- The MMSE linear estimate of $X'$ given $\mathbf{Y}'$ is given by $\hat{X}_n' = \mathbf{h}^T \mathbf{Y}'$, where

$$\Sigma_{\mathbf{Y}} \mathbf{h} = \Sigma_{\mathbf{Y}X}, \quad \text{thus}$$

$$\begin{bmatrix} P+N & P & \cdots & P \\ P & P+N & \cdots & P \\ \vdots & \vdots & \ddots & \vdots \\ P & P & \cdots & P+N \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{bmatrix} = \begin{bmatrix} P \\ P \\ \vdots \\ P \end{bmatrix}$$

- By symmetry, $h_1 = h_2 = \cdots = h_n = \dfrac{P}{nP+N}$. Thus

$$\hat{X}'_n = \frac{P}{nP+N}\sum_{i=1}^{n} Y'_i$$

Therefore

$$\hat{X}_n = \frac{P}{nP+N}\left(\sum_{i=1}^{n}(Y_i-\mu)\right)+\mu = \frac{P}{nP+N}\left(\sum_{i=1}^{n} Y_i\right)+\frac{N}{nP+N}\mu$$

- The mean square error of the estimate:

$$\mathrm{MSE}_n = P - \mathrm{E}(\hat{X}'_n X') = \frac{PN}{nP+N}$$

Thus as $n \to \infty$, $\mathrm{MSE}_n \to 0$, i.e., the linear estimate becomes perfect (even though we don't know the complete statistics of $X$ and $\mathbf{Y}$)

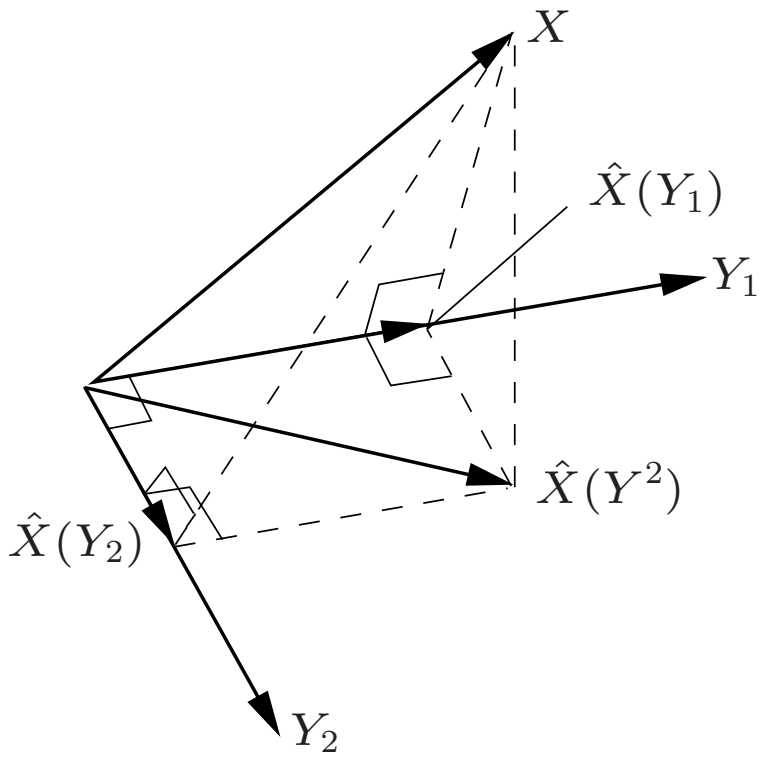
# Linear Innovation Sequence

- Let $X$ be the signal and $\mathbf{Y}$ be the observation vector (all zero mean)
- Suppose the $Y_i$s are orthogonal, i.e., $\mathrm{E}(Y_i Y_j) = 0$ for all $i \neq j$, and let $\hat{X}(\mathbf{Y})$ be the best linear MSE estimate of $X$ given $\mathbf{Y}$ and $\hat{X}(Y_i)$ be the best linear MSE estimate of $X$ given only $Y_i$ for $i = 1, \ldots, n$, then we can write

$$\hat{X}(\mathbf{Y}) = \sum_{i=1}^{n} \hat{X}(Y_i),$$

$$\mathrm{MSE} = \mathrm{Var}(X) - \sum_{i=1}^{n} \frac{\mathrm{Cov}^2(X, Y_i)}{\mathrm{Var}(Y_i)}$$

  This can be proved by evaluating the best linear estimate or using orthogonality:




- Hence if the observations are orthogonal, the computation of the best linear MSE estimate and its MSE are very simple
- In fact, we can compute the estimates and the MSE causally (recursively)

$$\hat{X}(Y^{i+1}) = \hat{X}(Y^i) + \hat{X}(Y_{i+1})$$

$$\mathrm{MSE}_{i+1} = \mathrm{MSE}_i - \frac{\mathrm{Cov}^2(X, Y_{i+1})}{\mathrm{Var}(Y_{i+1})}$$

- Now suppose the $Y_i$s are not orthogonal. We can still express the estimate and its MSE as sums
  - We first whiten $\mathbf{Y}$ to obtain $\mathbf{Z}$. The best linear MSE estimate of $X$ given $\mathbf{Y}$ is exactly the same as that given $\mathbf{Z}$ (why?)
  - The estimate and its MSE can then be computed as

$$\hat{X}(\mathbf{Y}) = \sum_{i=1}^{n} \hat{X}(Z_i)$$

$$\text{MSE} = \text{Var}(X) - \sum_{i=1}^{n} \text{Cov}^2(X, Z_i)$$

- We can compute an orthogonal observation sequence $\tilde{\mathbf{Y}}$ from $\mathbf{Y}$ causally:
  - Given $Y^i$, we compute the error of the best linear MSE estimate of $Y_{i+1}$,

$$\tilde{Y}_{i+1}(Y^i) = Y_{i+1} - \hat{Y}_{i+1}(Y^i)$$

  - Clearly, $\tilde{Y}_{i+1} \perp (\tilde{Y}_1, \tilde{Y}_2, \ldots, \tilde{Y}_i)$, hence we can write

$$\hat{Y}_{i+1}(Y^i) = \sum_{j=1}^{i} \hat{Y}_{i+1}(\tilde{Y}_j)$$

- Interpretation: $\hat{Y}_{i+1}$ is the part of $Y_{i+1}$ predictable by $Y^i$, hence carries no useful new information for estimating $X$ beyond $Y^i$

  $\tilde{Y}_{i+1}$ by comparison is the unpredictable part, hence carries new information

  As such, $\tilde{\mathbf{Y}}$ is called the linear innovation sequence of $\mathbf{Y}$

- Remark: If we normalize $\tilde{\mathbf{Y}}$ (by dividing each $\tilde{Y}_i$ by its standard deviation), we obtain the same sequence as using the Cholesky decomposition in Lecture notes 3

- Example: Let the observation sequence be $Y_i = X + Z_i$ for $i = 1, 2, \ldots, n$, where $X$, $Z_1$, ..., $Z_n$ are zero mean, uncorrelated r.v.s with $\mathrm{E}(X^2) = P$ and $\mathrm{E}(Z_i^2) = N$ for $i = 1, 2, \ldots, n$. Find the linear innovation sequence of $\mathbf{Y}$

- Using the innovation sequence, the MMSE linear estimate of $X$ given $\tilde{Y}^{i+1}$ and its MSE can be computed causally

$$\hat{X}(\tilde{Y}^{i+1}) = \hat{X}(\tilde{Y}^i) + \hat{X}(\tilde{Y}_{i+1}),$$

$$\mathrm{MSE}_{i+1} = \mathrm{MSE}_i - \frac{\mathrm{Cov}^2(X, \tilde{Y}_{i+1})}{\mathrm{Var}(\tilde{Y}_{i+1})}$$

- The innovation sequence will prove useful in deriving the Kalman filter

# Kalman Filter

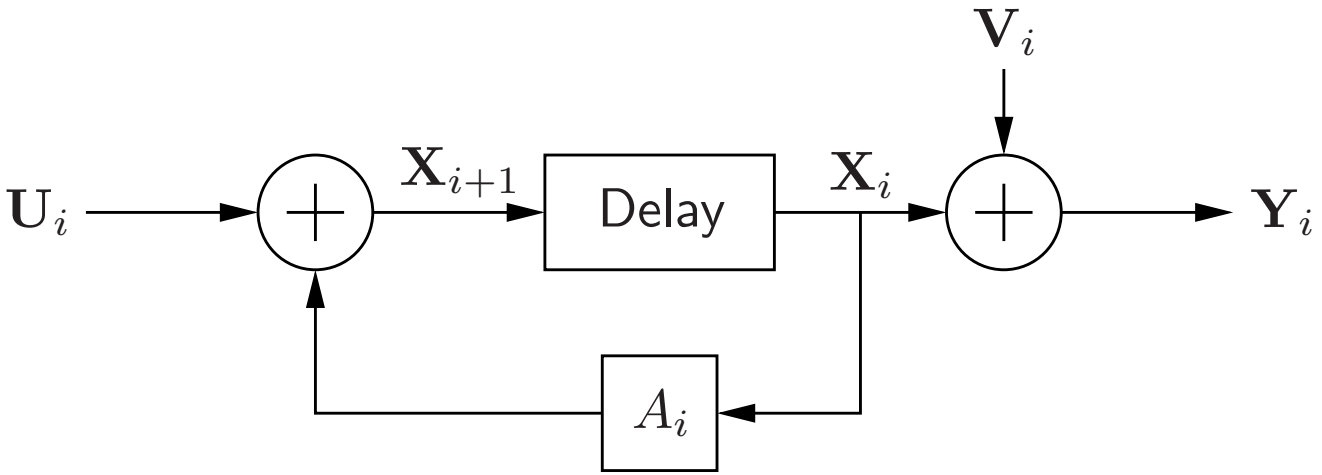
- The Kalman filter is an efficient, recursive algorithm for computing the MMSE linear estimate and its MSE when the signal $\mathbf{X}$ and observations $\mathbf{Y}$ evolve according to a state-space model
- Consider a linear dynamical system described by the state-space model:

$$\mathbf{X}_{i+1} = A_i \mathbf{X}_i + \mathbf{U}_i, \quad i = 0, 1, \ldots, n$$

with noisy observations (output)

$$\mathbf{Y}_i = \mathbf{X}_i + \mathbf{V}_i, \quad i = 0, 1, \ldots, n,$$

where $\mathbf{X}_0$, $\mathbf{U}_0, \mathbf{U}_1, \ldots, \mathbf{U}_n$, $\mathbf{V}_0, \mathbf{V}_1, \ldots, \mathbf{V}_n$ are zero mean, uncorrelated RVs with $\Sigma_{\mathbf{X}_0} = P_0$, $\Sigma_{\mathbf{U}_i} = Q_i$, $\Sigma_{\mathbf{V}_i} = N_i$; $A_i$ is a known sequence of matrices



- This state space model is used in many applications:
  - Navigation, e.g., of a car:

    State: is location, speed, heading, acceleration, tilt, steering wheel position of vehicle

    Observations: inertial (accelerometer, gyroscopes), electronic compass, GPS
  - Phase locked loop:

    State: phase and frequency offsets

    Observations: noisy observation of phase
  - Computer vision, e.g., face tracking:

    State: Pose, motion, shape (size, articulation), appearance (light, color)

    Observations: video frame sequence
  - Economics . . .

- The goal is to compute the MMSE linear estimate of the state from causal observations:
  - Prediction: Find the estimate $\hat{\mathbf{X}}_{i+1|i}$ of $\mathbf{X}_{i+1}$ from $\mathbf{Y}^i$ and the covariance matrix of its MSE $\Sigma_{i+1|i}$
  - Filtering: Find the estimate $\hat{\mathbf{X}}_{i|i}$ of $\mathbf{X}_i$ from $\mathbf{Y}^i$ and the covariance matrix of its MSE $\Sigma_{i|i}$
- The Kalman filter provides clever recursive equations for computing these estimates and their error covariance matrices

# Scalar Kalman Filter

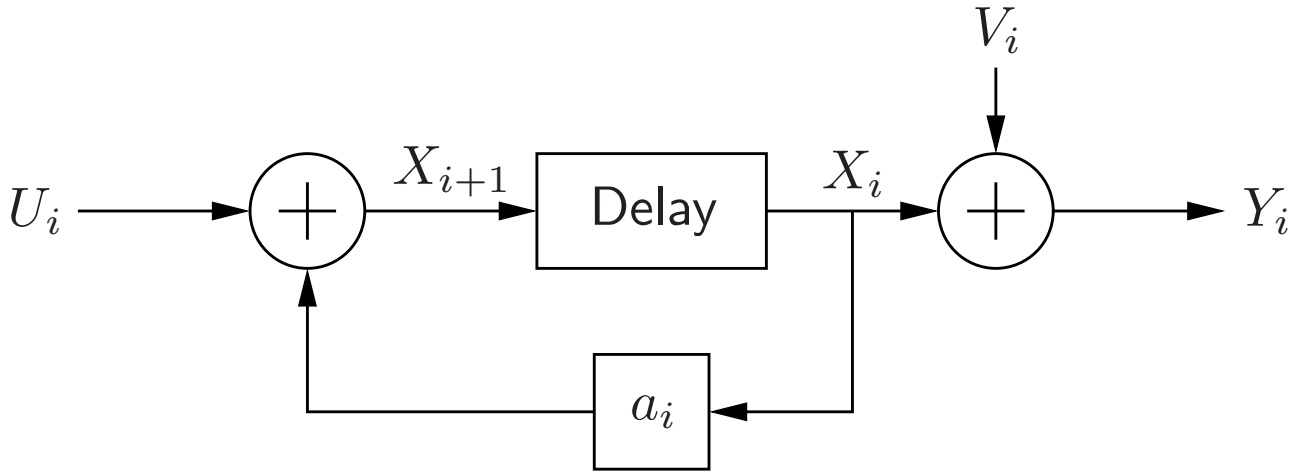
- Consider the scalar state space system:

$$X_{i+1} = a_i X_i + U_i, \quad i = 0, 1, \ldots, n$$

with noisy observations

$$Y_i = X_i + V_i, \quad i = 0, 1, \ldots, n,$$

where $X_0$, $U_0, U_1, \ldots, U_n$, $V_0, V_1, \ldots, V_n$ are zero mean, uncorrelated r.v.s with $\mathrm{Var}(X_0) = P_0$, $\mathrm{Var}(U_i) = Q_i$, $\mathrm{Var}(V_i) = N_i$, and $a_i$ is a known sequence



- Kalman filter (prediction):

  Initialization: $\hat{X}_{0|-1} = 0$, $\sigma^2_{0|-1} = P_0$

  Update equations: For $i = 0, 1, 2, \ldots, n$, the estimate is

$$\hat{X}_{i+1|i} = a_i \hat{X}_{i|i-1} + k_i(Y_i - \hat{X}_{i|i-1}),$$

  where the filter gain is

$$k_i = \frac{a_i \sigma^2_{i|i-1}}{\sigma^2_{i|i-1} + N_i}$$

  The MSE of $\hat{X}_{i+1|i}$ is

$$\sigma^2_{i+1|i} = a_i(a_i - k_i)\sigma^2_{i|i-1} + Q_i$$

- Example: Let $a_i = 1$, $Q_i = 0$, $N_i = N$, and $P_0 = P$ (so $X_0 = X_1 = X_2 = \cdots = X$), and $Y_i = X + V_i$ (this is the same as the earlier estimation example)

  Kalman filter:

  Initialization: $\hat{X}_{0|-1} = 0$ and $\sigma^2_{0|-1} = P$

  The update in each step is

$$\hat{X}_{i+1|i} = (1 - k_i)\hat{X}_{i|i-1} + k_i Y_i$$

  with

$$k_i = \frac{\sigma^2_{i|i-1}}{\sigma^2_{i|i-1} + N},$$

  and the MSE is

$$\sigma^2_{i+1|i} = (1 - k_i)\sigma^2_{i|i-1}$$

We can solve for $\sigma^2_{i+1|i}$ explicitly

$$\sigma^2_{i+1|i} = \left(1 - \frac{\sigma^2_{i|i-1}}{\sigma^2_{i|i-1} + N}\right)\sigma^2_{i|i-1} = \frac{N\sigma^2_{i|i-1}}{\sigma^2_{i|i-1} + N}$$

$$\frac{1}{\sigma^2_{i+1|i}} = \frac{1}{N} + \frac{1}{\sigma^2_{i|i-1}}$$

$$\sigma^2_{i+1|i} = \frac{1}{(i+1)/N + 1/P} = \frac{NP}{(i+1)P + N}$$

The gain is

$$k_i = \frac{P}{(i+1)P + N}$$

The recursive estimate is

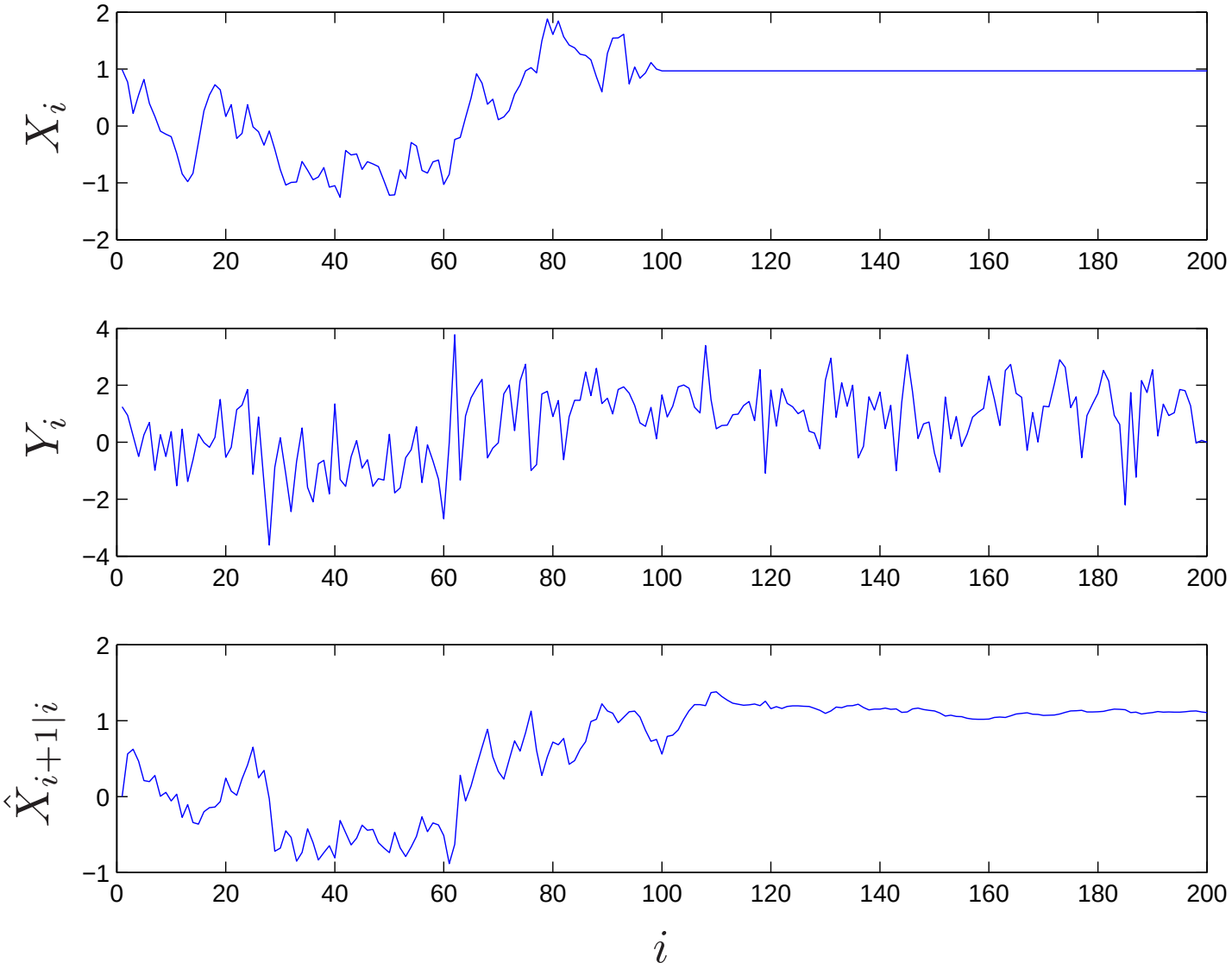
$$\hat{X}_{i+1|i} = \frac{iP + N}{(i+1)P + N}\hat{X}_{i|i-1} + \frac{P}{(i+1)P + N}Y_i$$

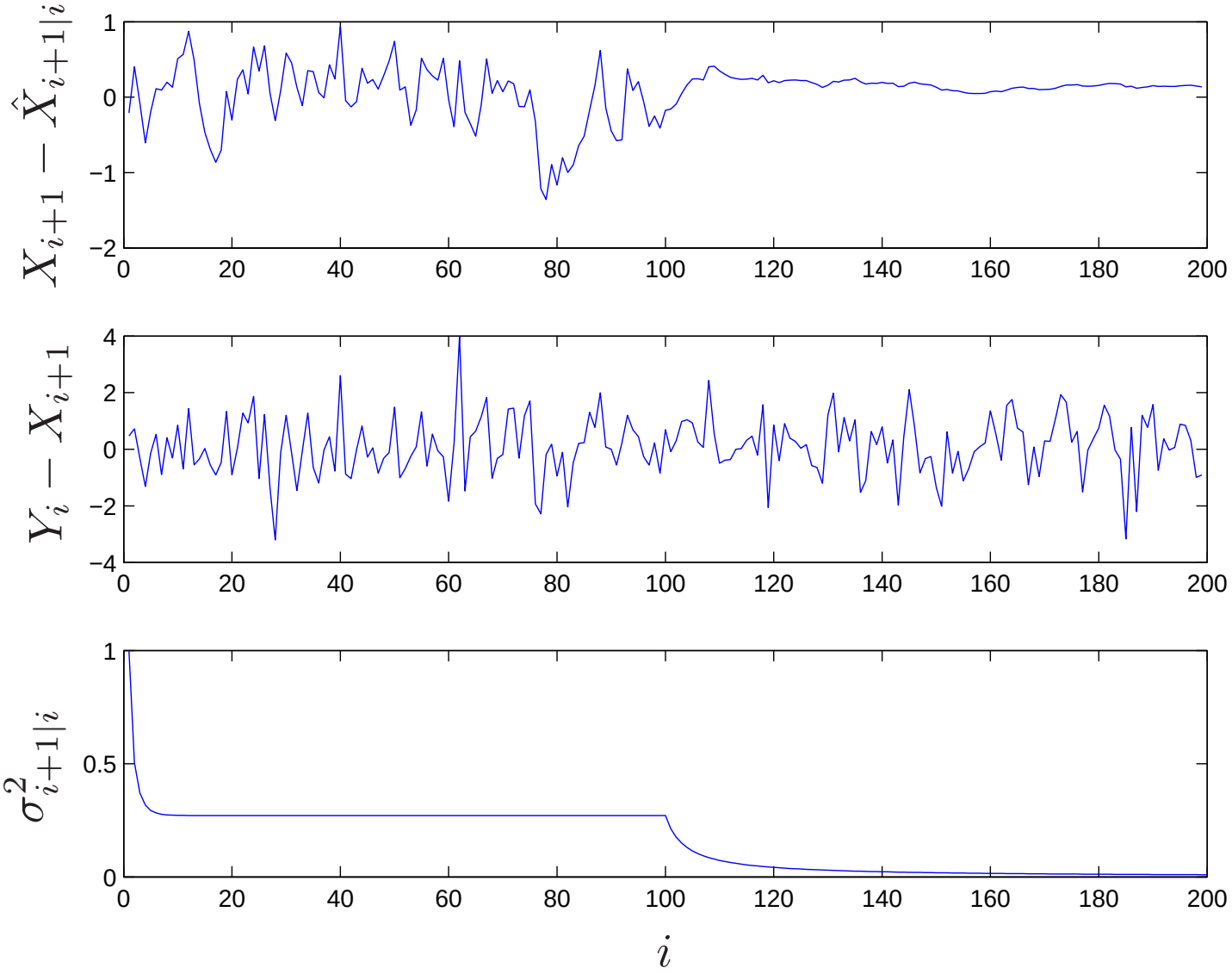
We thus obtain the previous result (with $\mu = 0$) in a recursive form

- Example: Let $n = 200$, $P_0 = 1$, $N_i = 1$

  For $i = 1$ to 100: $a_i = \alpha$, $Q_i = (1 - \alpha^2)$ with $\alpha = 0.95$ (memory factor)

  For $i = 100$ to 200: $a_i = 1$, $Q_i = 0$ (i.e., state remains constant)




$X_{i+1} - \hat{X}_{i+1|i}$
1
0
−1
−2
0
20
40
60
80
100
120
140
160
180
200
$Y_i - X_{i+1}$
4
2
0
−2
−4
$\sigma^2_{i+1|i}$
0.5
$i$

# Derivation of the Kalman Filter

- We use innovations. Let $\tilde{Y}_i$ be the innovation r.v. for $Y_i$, then we can write

$$\hat{X}_{i+1|i} = \hat{X}_{i+1|i-1} + k_i \tilde{Y}_i,$$
$$\sigma^2_{i+1|i} = \sigma^2_{i+1|i-1} - k_i \mathrm{Cov}(X_{i+1}, \tilde{Y}_i)$$

  where $\hat{X}_{i+1|i-1}$ and $\sigma_{i+1|i-1}$ are the MMSE linear estimate of $X_{i+1}$ given $Y^{i-1}$ and its MSE, and

$$k_i = \frac{\mathrm{Cov}(X_{i+1}, \tilde{Y}_i)}{\mathrm{Var}(\tilde{Y}_i)}$$

- Now, since $X_{i+1} = a_i X_i + U_i$, by linearity of MMSE linear estimate, we have

$$\hat{X}_{i+1|i-1} = a_i \hat{X}_{i|i-1}$$

  and

$$\sigma^2_{i+1|i-1} = a_i^2 \sigma^2_{i|i-1} + Q_i$$

- Now, the innovation r.v. for $Y_i$ is $\tilde{Y}_i = Y_i - \hat{Y}_i(Y^{i-1})$

  Since $Y_i = X_i + V_i$ and $V_i$ is uncorrelated with $Y_j$, $j = 1, 2, \ldots, i-1$,

$$\hat{Y}_i(Y^{i-1}) = \hat{X}_{i|i-1}$$

  Hence,

$$\tilde{Y}_i = Y_i - \hat{X}_{i|i-1}$$

  This yields

$$\hat{X}_{i+1|i} = a_i \hat{X}_{i|i-1} + k_i \tilde{Y}_i = a_i \hat{X}_{i|i-1} + k_i (Y_i - \hat{X}_{i|i-1})$$
$$\sigma^2_{i+1|i} = \sigma^2_{i+1|i-1} - k_i \mathrm{Cov}(X_{i+1}, \tilde{Y}_i),$$

  Now, consider

$$k_i = \frac{\mathrm{Cov}(X_{i+1}, \tilde{Y}_i)}{\mathrm{Var}(\tilde{Y}_i)}$$
$$= \frac{\mathrm{Cov}(a_i X_i + U_i, X_i - \hat{X}_{i|i-1} + V_i)}{\mathrm{Var}(X_i - \hat{X}_{i|i-1} + V_i)}$$
$$= \frac{\mathrm{Cov}(a_i X_i, X_i - \hat{X}_{i|i-1})}{\mathrm{Var}(X_i - \hat{X}_{i|i-1} + V_i)}$$

$$= \frac{a_i \mathrm{Cov}(X_i, X_i - \hat{X}_{i|i-1})}{\mathrm{Var}(X_i - \hat{X}_{i|i-1} + V_i)}$$

$$= \frac{a_i \mathrm{Cov}(X_i - \hat{X}_{i|i-1}, X_i - \hat{X}_{i|i-1})}{\mathrm{Var}(X_i - \hat{X}_{i|i-1} + V_i)} \quad \text{since } (X_i - \hat{X}_{i|i-1}) \perp \hat{X}_{i|i-1}$$

$$= \frac{a_i \mathrm{Var}(X_i - \hat{X}_{i|i-1})}{\mathrm{Var}(X_i - \hat{X}_{i|i-1}) + N_i}$$

$$= \frac{a_i \sigma^2_{i|i-1}}{\sigma^2_{i|i-1} + N_i}$$

The MSE is

$$\begin{aligned}
\sigma^2_{i+1|i} &= \sigma^2_{i+1|i-1} - k_i \mathrm{Cov}(a_i X_i + U_i, X_i - \hat{X}_{i|i-1} + V_i) \\
&= \sigma^2_{i+1|i-1} - k_i a_i \sigma^2_{i|i-1} \\
&= a_i(a_i - k_i)\sigma^2_{i|i-1} + Q_i
\end{aligned}$$

This completes the derivation of the scalar Kalman filter

# Vector Kalman Filter

- The above scalar Kalman filter can be extended to the vector state space model:

  Initialization: $\hat{\mathbf{X}}_{0|-1} = 0$, $\Sigma_{0|-1} = P_0$

  Update equations: For $i = 0, 1, 2, \ldots, n$, the estimate is

$$\hat{\mathbf{X}}_{i+1|i} = A_i \hat{\mathbf{X}}_{i|i-1} + K_i(\mathbf{Y}_i - \hat{X}_{i|i-1}),$$

  where the filter gain matrix

$$K_i = A_i \Sigma_{i|i-1}(\Sigma_{i|i-1} + N_i)^{-1}$$

  The covariance of the error is

$$\Sigma_{i+1|i} = A_i \Sigma_{i|i-1} A_i^T - K_i \Sigma_{i|i-1} A_i^T + Q_i$$

- Remark: If $\mathbf{X}_0$, $\mathbf{U}_0, \mathbf{U}_1, \ldots, \mathbf{U}_n$ and $\mathbf{V}_0, \mathbf{V}_1, \ldots, \mathbf{V}_n$ are Gaussian (zero mean, uncorrelated), then the Kalman filter yields the best MSE estimate of $\mathbf{X}_i$, $i = 0, \ldots, n$

# Filtering

- Now assume the goal is to compute the MMSE linear estimate of $X_i$ given $Y^i$, i.e., instead of predicting the next state, we are interested in estimating the current state
- We denote this estimate by $\hat{X}_{i|i}$ and its MSE by $\sigma^2_{i|i}$
- The Kalman filter can be adapted to this case as follows:

  Initialization:

$$\hat{X}_{0|0} = \frac{P_0}{P_0 + N_0} Y_0$$

$$\sigma^2_{0|0} = \frac{P_0 N_0}{P_0 + N_0}$$

  Update equations: For $i = 1, 2, \ldots, n$, the estimate is

$$\hat{X}_{i|i} = a_{i-1}(1 - k_i)\hat{X}_{i-1|i-1} + k_i Y_i$$

with filter gain

$$k_i = \frac{a_{i-1}^2 \sigma_{i-1|i-1}^2 + Q_{i-1}}{a_{i-1}^2 \sigma_{i-1|i-1}^2 + Q_{i-1} + N_i}$$

and MSE recursion

$$\sigma_{i|i}^2 = (1 - k_i)\left(a_{i-1}^2 \sigma_{i-1|i-1}^2 + Q_{i-1}\right)$$

- Vector case

  Initialization:

$$\hat{\mathbf{X}}_{0|0} = P_0(P_0 + N_0)^{-1}\mathbf{Y}_0$$
$$\Sigma_{0|0} = P_0(I - (P_0 + N_0)^{-1}P_0)$$

  Update equations: For $i = 1, 2, \ldots, n$, the estimate is

$$\hat{\mathbf{X}}_{i|i} = (I - K_i)A_{i-1}\hat{\mathbf{X}}_{i-1|i-1} + K_i\mathbf{Y}_i$$

  with filter gain

$$K_i = (A_{i-1}\Sigma_{i-1|i-1}A_{i-1}^T + Q_{i-1})\left(A_{i-1}\Sigma_{i-1|i-1}A_{i-1}^T + Q_{i-1} + N_i\right)^{-1}$$

  and MSE recursion

$$\Sigma_{i|i} = (A_{i-1}\Sigma_{i-1|i-1}A_{i-1}^T + Q_{i-1})(I - K_i^T)$$