Notes

EE 225A

Druv Pai

Fall 2021

Contents

1 Background

1.1 Digital Signal Processing

1.1.1 Discrete-Time Fourier Transform (DTFT)

Definitions

Definition 1.1.1 (Discrete-Time Fourier Transform). For a complex-valued signal $\{x(n) \in \mathbb{C} \colon n \in \mathbb{Z}\}$, the **discrete-time Fourier transform** $X(\omega) \in \mathbb{C}$ as

$$X(\omega) := \sum_{n \in \mathbb{Z}} x(n) \mathrm{e}^{-\mathrm{i}\omega n}. \tag{1.1.1}$$

Notation 1.1.2. We sometimes also write it as $X\left(\mathrm{e}^{\mathrm{i}\omega}\right)$ since it in fact only depends on $\mathrm{e}^{\mathrm{i}\omega}$, and that $X(\omega)$ is a periodic function with period 2π. We usually only specify $X(\omega)$ for $\omega \in (-\pi, \pi]$.

The inverse DTFT is given by

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) \mathrm{e}^{\mathrm{i}\omega n} \, \mathrm{d}\omega. \tag{1.1.2}$$

Existence

We want to find out what family of signals $\{x(n) \colon n \in \mathbb{N}\}$ the DTFT exists. The following three cases are significant:

(i) The ℓ^1 case:

$$\|x\|_{\ell^1} := \sum_{n \in \mathbb{Z}} |x(n)| < \infty \tag{1.1.3}$$

i.e., series that converge absolutely. For this family, $X(\omega)$ is well defined for every ω, the convergence is uniform ω, and $X(\omega)$ is a continuous function of ω.

(ii) The ℓ^2 case:

$$\|x\|_{\ell^2} := \sum_{n \in \mathbb{Z}} |x(n)|^2 < \infty, \tag{1.1.4}$$

i.e., square-summable sequences. This class is bigger than ℓ^1 and corresponds to the Hilbert space theory of Fourier transforms, and the corresponding DTFT "converges in a mean-squared sense":

$$\lim_{N \to \infty} \int_{-\pi}^{\pi} |X_N(\omega) - X(\omega)|^2 \, \mathrm{d}\omega = 0, \tag{1.1.5}$$

where $X_N(\omega)$ refers to the DTFT of $x(n)$ truncated to between $n = -N$ and $n = N$ (and zero elsewhere), which is a finite sequence so $X_N(\omega)$ is well defined. For $x \in \ell^2$, $X(\omega)$ is defined almost everywhere (in the Lebesgue sense).

Example 1.1.3 (Discrete Sinc function). Let $0 < \omega_0 < \pi$ and define

$$x(n) := \frac{\sin(\omega_0 k)}{\pi_k}. \tag{1.1.6}$$

Then

$$X(\omega) = \begin{cases} 1 & |\omega| < \omega_0 \\ 0 & |\omega| \geq \omega_0. \end{cases} \tag{1.1.7}$$

Note that this function x belongs to ℓ^2 but not ℓ^1. Its DTFT is not continuous in ω. This means that $X_N(\omega)$ does not converge for $|\omega| = \omega_0$.

(iii) The case of tempered distributions: it is bigger than ℓ^1 and ℓ^2, but requires the theory of generalized functions, or distributions. In signal processing literature we use the Dirac delta function to deal with this case. Take the constant signal $x(n) = 1$, which clearly doesn't belong to ℓ^1 or ℓ^2. We have the DTFT pair:

$$x(n) = 1 \leftrightarrow X(\omega) = 2\pi\delta(\omega). \tag{1.1.8}$$

In the IDTFT formula,

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} 2\pi\delta(\omega)\,\mathrm{d}\omega = 1, \tag{1.1.9}$$

as desired.

Properties

The DTFT pair $x(n) \leftrightarrow X(\omega)$ has the following properties:

1. Time delay:

$$x(n-k) \leftrightarrow \mathrm{e}^{-\mathrm{i}\omega k} X(\omega) \tag{1.1.10}$$

2. Time reversal:

$$x(-n) \leftrightarrow X(-\omega) \tag{1.1.11}$$

3. Conjugate:

$$x^*(n) \leftrightarrow X^*(-\omega) \tag{1.1.12}$$

4. Convolution in time:

$$\left(x_1 * x_2\right)(n) := \sum_{k\in\mathbb{Z}} x_1(k) x_2(n-k) \leftrightarrow X_1(\omega) X_2(\omega). \tag{1.1.13}$$

5. Deterministic cross-correlation:

$$c(n) := \sum_{k\in\mathbb{Z}} x(k) y^*(k-n) \leftrightarrow C(\omega) = X(\omega) Y^*(\omega). \tag{1.1.14}$$

1.1.2 z-Transform

The z-transform generalizes the DTFT by replacing the complex number $\mathrm{e}^{\mathrm{i}\omega}$ with complex number $z \in \mathcal{C}$.

Definition

Definition 1.1.4 (z-Transform). Let $\{x(n) \in \mathcal{C} \colon n \in \mathbb{Z}\}$ be a signal and let $z \in \mathcal{C}$. The z-transform of $x(n)$ is defined as

$$X(z) := \sum_{n\in\mathbb{Z}} x(n) z^{-n}. \tag{1.1.15}$$

Existence

The series in Eq. (1.1.15) is called a Laurent series. Complex analysis tells us that there exists an inner radius r and an outer radius R such that:

1. The Laurent series converges *absolutely* on the open annulus $A = \{z \mid r < |z| < R\}$. To say the series converges, we mean that both the positive degree power series and the negative degree power series absolutely converge, which is equivalent to

$$\sum_{n\in\mathbb{Z}} |x(n)|\,|z|^{-n} < \infty. \tag{1.1.16}$$

   Furthermore, this convergence will be uniform on compact sets. Finally, the convergence series defines a holomorphic function $f(z)$ on the open annulus.

2. Outside the annulus, the Laurent series diverges. That is, at each point of the exterior of A, the positive degree power series or the negative degree power series diverges.

3. On the boundary of the annulus, one cannot make a general statement, except to say that there is at least one point on the inner boundary and one point on the outer boundary such that $f(z)$ cannot be holomorphically continued to those points.

It is possible that $r = 0$ or $R = \infty$. It is also not true that $r < R$ in general. We can compute the radii r, R as follows:

$$r = \limsup_{n\to\infty} |x(n)|^{1/n} \tag{1.1.17}$$

$$R = \left(\limsup_{n\to\infty} |x(-n)|^{1/n}\right)^{-1}. \tag{1.1.18}$$

In signal processing, we define the **region of convergence (ROC)** as the subset of $\mathcal{C}$ with $\{z\colon r < |z| < R\}$ since we only care about absolute convergence. By convention, the ROC concept is extended to $|z| = \infty$ by including $|z| = \infty$ in the ROC when $x(n) = 0$ for all $n < 0$ and excluding it otherwise. Similarly, $z = 0$ is in the ROC when $x(n) = 0$ for all $n > 0$ and not in the ROC otherwise.

It is essential to specify the ROC along with the z-transform $X(z)$ to uniquely describe a discrete-time signal. This is because 2 different signals $x_1(n)$ and $x_2(n)$ can have the same z-transform $X(z)$ and can only be distinguished by their different regions of convergence.

Example 1.1.5. Consider the signal

$$x_1(n) := u(n)2^{-n}, \tag{1.1.19}$$

where $u(n) := \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$. The z-transform is given by

$$X_1(z) = \sum_{n\in\mathbb{Z}} x_1(n) z^{-n} = \sum_{n\in\mathbb{N}\cup\{0\}} (2z)^{-n} = \frac{1}{1-\frac{1}{2z}} \tag{1.1.20}$$

with the last equation converging if and only if $\left|\frac{1}{2z}\right| < 1$. Hence the region of convergence is $|z| > \frac{1}{2}$.

Now consider the signal

$$x_1(n) := -u[-n-1]2^{-n}. \tag{1.1.21}$$

The z-transform is

$$X_2(z) = \sum_{n\in\mathbb{Z}} x_2(n) z^{-n} = -\sum_{n\in\mathbb{Z}\setminus(\mathbb{N}\cup\{0\})} (2z)^{-n} = -\sum_{n\in\mathbb{N}} (2z)^{n} = \frac{2z}{1-2z} = \frac{1}{1-\frac{1}{2z}}. \tag{1.1.22}$$

The ROC for the z transform is $|2z| < 1$ which implies $|z| < \frac{1}{2}$. The z-transforms are the same for x_1 and x_2, but the ROCs are different.

Causality

Definition 1.1.6 (Causality). A discrete-time signal $x(n)$ is **causal** if and only if $x(n) = 0$ for $n < 0$.

We know $x(n)$ is causal if and only if the ROC includes ∞.

From the ROCs in the two examples above, we can see that only the first ROC includes ∞, so $x_1(n)$ is causal while $x_2(n)$ is non-causal, which is also obvious from the time domain representation.

BIBO Stability

BIBO stands for bounded-input-bounded-output. By Hölder's inequality,

$$\left|\sum_{n\in\mathbb{N}} x(n)y(n)\right| \le \left(\sup_n |y(n)|\right)\left(\sum_{n\in\mathbb{N}} |x(n)|\right). \tag{1.1.23}$$

Since ℓ^∞ is dual to ℓ^1, the left hand side of Eq. (1.1.23) is bounded for all bounded inputs $y \in \ell^\infty$ if and only if $x \in \ell^1$, which is true if and only if the ROC contains the unit circle $|z| = 1$.

From the ROCs in the previous examples, we can see that only the first ROC includes $|z| = 1$, so $x_1(n)$ is BIBO stable while $x_2(n)$ is BIBO unstable, which can be inferred from the time domain representations as well.

Properties

The z-transform pair $x(n) \leftrightarrow X(z)$ has the following properties:

1. Time delay:
$$x(n-k) \leftrightarrow z^{-k}X(z) \tag{1.1.24}$$

2. Time reversal:
$$x(-n) \leftrightarrow X\left(z^{-1}\right) \tag{1.1.25}$$

3. Conjugate:
$$x^*(n) \leftrightarrow X^*(z^*) \tag{1.1.26}$$

4. Convolution in time:
$$\left(x_1 * x_2\right)(n) := \sum_{k\in\mathbb{N}} x_1(k)x_2(n-k) \leftrightarrow X_1(z)X_2(z) \tag{1.1.27}$$

5. Deterministic cross-correlation:
$$c(n) := \sum_{k\in\mathbb{Z}} x(k)y^*(k-n) \leftrightarrow C(z) = X(z)Y^*\left(z^{-1}\right). \tag{1.1.28}$$

1.2 Linear Estimation

1.2.1 Introduction

The problem of optimal linear estimation, or more precisely, optimal linear estimation under squared error loss, is one of the foundational pillars of statistical signal processing and Bayesian statistics with great theoretical depth and practical applications.

Abstractly, the objective is to find the best estimate of signal X using a *linear* function of the observation Y. Here X and Y do not need to have the same dimension, and could take values in general sets.

We do assume we know the **auto-correlation** and **cross-correlation** functions:

$$R_X := \mathbf{E}(XX^*) \tag{1.2.1}$$

$$R_Y := \mathbf{E}(YY^*) \tag{1.2.2}$$

$$R_{XY} := \mathbf{E}(XY^*) \tag{1.2.3}$$

where we view X and Y as column vectors and $*$ denotes the conjugate transpose operation. Note that we *do not* need assumptions on $\mathbf{E}(X)$ and $\mathbf{E}(Y)$, and we *do not* need to know the joint distribution of (X, Y).

The loss function we use is the *squared error* loss function. This loss has deep connections with Hilbert space, which enables a general theory with beautiful formulas. Our treatment is a bit unconventional, since we use the notation $\langle \cdot, \cdot \rangle$ to denote a function whose output is a rectangular matrix, but the underlying theory is still classical Hilbert space theory. The benefits of this notation will be apparent in the later parts of this course.

Mathematically, define the (possibly matrix-valued) inner product between two random vectors Z_1, Z_2 which may not have the same dimension:

$$\langle Z_1, Z_2 \rangle := \mathbf{E}(Z_1 Z_2^*) \tag{1.2.4}$$

and denote

$$\|Z\|^2 := \langle Z, Z \rangle . \tag{1.2.5}$$

It is consistent with our intuition of inner product for Euclidean spaces. We map a vector in the Euclidean space to a random variable, and a collection of vectors in the Euclidean space to a random vector.

Proposition 1.2.1. The inner product satisfies the following properties:

1. Linearity:
$$\langle \alpha_1 v_1 + \alpha_2 v_2, u \rangle = \alpha_1 \langle v_1, u \rangle + \alpha_2 \langle v_2, u \rangle . \tag{1.2.6}$$
2. Reflexivity:
$$\langle u, v \rangle = \langle v, u \rangle^* . \tag{1.2.7}$$
3. Non-degeneracy:
$$\|v\|^2 = 0 \iff v = 0. \tag{1.2.8}$$

We would like to solve for the linear map $\widehat{X}(Y)$ of Y that satisfies

$$\min_{\widehat{X}} \left\| \widehat{X} - X \right\|^2 . \tag{1.2.9}$$

Here we constrain $\widehat{X}(Y)$ to be a linear function of Y. In other words, it satisfies

$$\widehat{X}(\alpha_1 Y_1 + \alpha_2 Y_2) = \alpha_1 \widehat{X}(Y_1) + \alpha_2 \widehat{X}(Y_2). \tag{1.2.10}$$

Sometimes there exists an explicit matrix W such that $\widehat{X}(Y) = WY$. If Y is a finite-dimensional random vector it is the case. But it is important to keep in mind that it is not always possible. One notable example is the optimal linear predictor of X_t given its causal history $(X_j)_{j\in[t-1]}$ when $X_t = V_t - V_{t-1}$, where V_t are mutually independent $\mathcal{N}(0,1)$ random variables.

1.2.2 Orthogonality Principle

On the face of it, it's not even clear that Eq. (1.2.9) has a solution. Indeed, we are evaluating X with $\widehat{X}$ using a matrix, which only has a partial order. The surprising/incredible property of Hilbert spaces guarantees that it is in fact solvable and has a unique solution.

Theorem 1.2.2 (Orthogonality Principle). The optimization problem Eq. (1.2.9) has a unique solution $\widehat{X}(Y)$ such that for any linear function $\widehat{X}'(Y)$ of Y,

$$\left\| X - \widehat{X} \right\|^2 \leq \left\| X - \widehat{X}' \right\|^2 . \tag{1.2.11}$$

Further, $\widehat{X}$ is the unique linear function of Y satisfying the **orthogonality principle**:

$$\left\langle X - \widehat{X}, Y \right\rangle = 0. \tag{1.2.12}$$

Proof. One can show that there exists some $\widehat{X}$ satisfying Eq. (1.2.12). Now we need to show that it is unique, and that it solves Eq. (1.2.9). Note that Eq. (1.2.12) is equivalent to $\left\langle X-\widehat{X},\widehat{X}'(Y)\right\rangle$ for any linear function $\widehat{X}'(Y)$ of Y.

We first show that it solves the minimization problem. We can write

$$
\begin{align}
\left\|X-\widehat{X}'\right\|^2 &= \left\|X-\widehat{X}+\widehat{X}-\widehat{X}'\right\|^2 \tag{1.2.13}\\
&= \left\|X-\widehat{X}\right\|^2+\left\|\widehat{X}-\widehat{X}'\right\|^2+\underbrace{\left\langle X-\widehat{X},\widehat{X}-\widehat{X}'\right\rangle}_{=0}+\underbrace{\left\langle \widehat{X}-\widehat{X}',X-\widehat{X}\right\rangle}_{=0} \tag{1.2.14}\\
&= \left\|X-\widehat{X}'\right\|^2-\left\|\widehat{X}-\widehat{X}'\right\|^2 \tag{1.2.15}\\
&\leq \left\|X-\widehat{X}'\right\|^2. \tag{1.2.16}
\end{align}
$$

Here, the matrix ordering is based on the spectrum; in particular, we are saying that $\left\|X-\widehat{X}'\right\|^2-\left\|X-\widehat{X}'\right\|^2 \succeq 0$.

Now we show the uniqueness. Suppose both $\widehat{X}_1$ and $\widehat{X}_2$ satisfy the orthogonality principle Eq. (1.2.12). Then

$$
\begin{align}
\left\|\widehat{X}_1-\widehat{X}_2\right\|^2 &= \left\langle \widehat{X}_1-\widehat{X}_2,\widehat{X}_1-\widehat{X}_2\right\rangle \tag{1.2.17}\\
&= \left\langle \widehat{X}_1-\widehat{X}_2,\widehat{X}_1-X+X-\widehat{X}_2\right\rangle \tag{1.2.18}\\
&= \underbrace{\left\langle \widehat{X}_1-\widehat{X}_2,X-\widehat{X}_2\right\rangle}_{=0}-\underbrace{\left\langle \widehat{X}_1-\widehat{X}_2,X-\widehat{X}_1\right\rangle}_{=0} \tag{1.2.19}\\
&= 0. \tag{1.2.20}
\end{align}
$$

□

Now we apply Theorem 1.2.2 to solve Eq. (1.2.9) assuming that we can write $\widehat{X}(Y)=WY$. The orthogonality principle says that

$$\langle X-WY,Y\rangle=0 \tag{1.2.21}$$

which is equivalent to

$$
\begin{align}
\langle X,Y\rangle &= W\langle Y,Y\rangle \tag{1.2.22}\\
R_{XY} &= WR_Y. \tag{1.2.23}
\end{align}
$$

This is the **normal equation**.

A first question is whether Eq. (1.2.23) always has a solution. Theorem 1.2.2 guarantees this even when R_Y is singular, and clearly if R_Y is invertible we have

$$W=R_{XY}R_Y^{-1}. \tag{1.2.24}$$

Proposition 1.2.3. No matter what solution W to Eq. (1.2.23) we pick, the corresponding estimator WY is always the same.

Proof. Suppose W_1,W_2 are such that

$$W_1R_Y=W_2R_Y=R_{XY}. \tag{1.2.25}$$

Then

$$
\begin{align}
\left\|W_1Y-W_2Y\right\|^2 &= \langle W_1Y-W_2Y,W_1Y-W_2Y\rangle \tag{1.2.26}\\
&= (W_1-W_2)\langle Y,Y\rangle(W_1-W_2)^* \tag{1.2.27}
\end{align}
$$

$$
\begin{aligned}
&= \left(W_1 - W_2\right) R_Y \left(W_1 - W_2\right)^* && (1.2.28)\\
&= W_1 R_Y \left(W_1 - W_2\right)^* - W_2 R_Y \left(W_1 - W_2\right)^* && (1.2.29)\\
&= R_{XY} \left(W_1 - W_2\right)^* - R_{XY} \left(W_1 - W_2\right)^* && (1.2.30)\\
&= 0. && (1.2.31)
\end{aligned}
$$

□

If R_Y is singular, then one solution of W (not unique) is $W = R_{XY} R_Y^\dagger$, where † denotes the pseudoinverse. To be precise, R_Y is hermitian. If we do an eigenvalue decomposition $R_Y = U_r \Lambda_r U_r^*$, where R_Y has rank r, U_r has orthonormal columns, and Λ_r is an invertible $r \times r$ matrix, then the pseudoinverse of R_Y can be defined by

$$R_Y^\dagger = U_r \Lambda_r^{-1} U_r^*. \tag{1.2.32}$$

Also we have the representation

$$Y = \sum_{i \in [r]} u_i \left(u_i^* Y\right) \tag{1.2.33}$$

where $(u_i)_{i \in [r]}$ are the columns of U_r. Hence

$$
\begin{aligned}
W R_Y &= R_{XY} R_Y^\dagger R_Y && (1.2.34)\\
&= R_{XY} \left(U_r \Lambda_r^{-1} U_r^*\right) \left(U_r \Lambda_r U_r^*\right) && (1.2.35)\\
&= R_{XY} U_r \Lambda_r^{-1} \Lambda_r U_r^* && (1.2.36)\\
&= R_{XY} U_r U_r^*. && (1.2.37)
\end{aligned}
$$

Since we wanted to show $W R_Y = R_{XY}$, it remains to show $R_{XY} = R_{XY} U_r U_r^*$. We have

$$
\begin{aligned}
R_{XY} &= \langle X, Y \rangle && (1.2.38)\\
&= \left\langle X, \sum_{i \in [r]} u_i \left(u_i^* Y\right) \right\rangle && (1.2.39)\\
&= \sum_{i \in [r]} \langle X, Y \rangle u_i u_i^* && (1.2.40)\\
&= \sum_{i \in [r]} R_{XY} u_i u_i^* && (1.2.41)\\
&= R_{XY} \sum_{i \in [r]} u_i u_i^* && (1.2.42)\\
&= R_{XY} U_r U_r^* && (1.2.43)\\
&= \sum_{i \in [r]} \langle X, Y \rangle u_i u_i^* && (1.2.44)\\
&= \sum_{i \in [r]} R_{XY} u_i u_i^* && (1.2.45)\\
&= R_{XY} \sum_{i \in [r]} u_i u_i^* && (1.2.46)\\
&= R_{XY} U_r U_r^*. && (1.2.47)
\end{aligned}
$$

1.2.3 Error

If $R_Y > 0$,

$$\left\langle \widehat{X}, \widehat{X} \right\rangle = \langle WY, WY \rangle \tag{1.2.48}$$

$$
\begin{aligned}
&= W \langle Y, Y \rangle W^* && (1.2.49)\\
&= W R_Y W^* && (1.2.50)\\
&= R_{XY} R_Y^{-1} R_Y R_Y^{-1} R_{YX} && (1.2.51)\\
&= R_{XY} R_Y^{-1} R_{YX}. && (1.2.52)
\end{aligned}
$$

Since

$$X = \widehat{X} + \left(X - \widehat{X} \right) \quad \text{and} \quad \left\langle \widehat{X}, X - \widehat{X} \right\rangle = 0 \tag{1.2.53}$$

we have

$$
\begin{aligned}
\langle X, X \rangle &= \left\langle \widehat{X}, \widehat{X} \right\rangle + \left\langle X - \widehat{X}, X - \widehat{X} \right\rangle && (1.2.54)\\
R_X &= R_{\widehat{X}} + R_e && (1.2.55)
\end{aligned}
$$

where $e := X - \widehat{X}$ is the **error**.

1.2.4 Affine Estimators

Previously, we did not impose assumptions on the expectations of X and Y in the normal equations, Eq. (1.2.23). That is because we are only allowed to use a linear function of Y, not a linear function of $\begin{bmatrix} Y \\ 1 \end{bmatrix}$, to estimate X. Now we extend our theory to affine estimators.

Define the following quantities:

$$
\begin{aligned}
\mu_X &:= \mathbf{E}(X) && (1.2.56)\\
\mu_Y &:= \mathbf{E}(Y) && (1.2.57)\\
\Sigma_X &:= \mathbf{E}((X - \mu_X)(X - \mu_X)^*) && (1.2.58)\\
\Sigma_Y &:= \mathbf{E}((Y - \mu_Y)(Y - \mu_Y)^*) && (1.2.59)\\
\Sigma_{XY} &:= \mathbf{E}((X - \mu_X)(Y - \mu_Y)^*) && (1.2.60)\\
& && (1.2.61)
\end{aligned}
$$

Proposition 1.2.4. The best affine estimator of X given Y (in the sense of Theorem 1.2.2) is of the form

$$\widehat{X} = W(Y - \mu_Y) + \mu_X, \tag{1.2.62}$$

where $W\Sigma_Y = \Sigma_{XY}$.

Proof. We will show the case where $\Sigma_Y > 0$. Define a new vector $Z = \begin{bmatrix} Y \\ 1 \end{bmatrix}$. Then

$$
\begin{aligned}
R_{XZ} &= \mathbf{E}(XZ^*) && (1.2.63)\\
&= \mathbf{E}\big(X \begin{bmatrix} Y^* & 1 \end{bmatrix}\big) && (1.2.64)\\
&= \big[\mathbf{E}(XY^*), \mathbf{E}(X)\big] && (1.2.65)\\
&= \big[R_{XY}, \mu_X\big] && (1.2.66)\\
&= \big[\Sigma_{XY} + \mu_X \mu_Y^*, \mu_X\big] && (1.2.67)\\
R_Z &= \mathbf{E}(ZZ^*) && (1.2.68)\\
&= \mathbf{E}\left(\begin{bmatrix} YY^* & Y \\ Y^* & 1 \end{bmatrix} \right) && (1.2.69)\\
&= \begin{bmatrix} R_{XY} & \mu_Y \\ \mu_Y^* & 1 \end{bmatrix}. && (1.2.70)
\end{aligned}
$$

The block matrix inversion formula is

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} \left(A - BD^{-1}C\right)^{-1} & -\left(A - BD^{-1}C\right)^{-1} BD^{-1} \\ -D^{-1}C\left(A - BD^{-1}C\right) & D^{-1} + D^{-1}C\left(A - BD^{-1}C\right) BD^{-1} \end{bmatrix} \tag{1.2.71}$$

and applying it toR_Z, we get

$$R_Z^{-1} = \begin{bmatrix} \Sigma_Y^{-1} & -\Sigma_Y^{-1}\mu_Y \\ -\mu_Y^*\Sigma_Y^{-1} & 1 + \mu_Y^*\Sigma_Y^{-1}\mu_Y \end{bmatrix}. \tag{1.2.72}$$

Hence, the optimal linear estimator of X given Z is

$$\begin{aligned} \widehat{X}(Z) &= R_{XZ}R_Z^{-1}Z & (1.2.73) \\ &= \begin{bmatrix} \Sigma_{XY} + \mu_X\mu_Y^* & \mu_X \end{bmatrix} \begin{bmatrix} \Sigma_Y^{-1} & -\Sigma_Y^{-1}\mu_Y \\ -\mu_Y^*\Sigma_Y^{-1} & 1 + \mu_Y^*\Sigma_Y^{-1}\mu_Y \end{bmatrix} \begin{bmatrix} Y \\ 1 \end{bmatrix} & (1.2.74) \\ &= \begin{bmatrix} \Sigma_{XY}\Sigma_Y^{-1} & -\Sigma_{XY}\Sigma_Y^{-1}\mu_Y + \mu_X \end{bmatrix} \begin{bmatrix} Y \\ 1 \end{bmatrix} & (1.2.75) \\ &= \Sigma_{XY}\Sigma_Y^{-1}\left(Y - \mu_Y\right) + \mu_X. & (1.2.76) \end{aligned}$$

In this case we have the error matrices:

$$\begin{aligned} \Sigma_{\widehat{X}} &= \Sigma_{XY}\Sigma_Y^{-1}\Sigma_{YX} & (1.2.77) \\ \Sigma_X &= \Sigma_{\widehat{X}} + \Sigma_e & (1.2.78) \end{aligned}$$

where as before $e = X - \widehat{X}$. □

1.2.5 Scalar Affine Estimators

It is intuitive to look for the scalar case, i.e., X and Y are scalar random variables. Let ρ be the correlation coefficient, defined as

$$\rho(X, Y) = \frac{\mathbf{E}\left(\left(X - \mathbf{E}(X)\right)\left(Y - \mathbf{E}(Y)\right)^*\right)}{\sqrt{\mathbf{E}\left(\|X - \mathbf{E}(X)\|^2\right)\mathbf{E}\left(\|Y - \mathbf{E}(Y)\|^2\right)}}. \tag{1.2.79}$$

We know from Cauchy-Schwarz that $0 \leq \rho \leq 1$. Then Eq. (1.2.62) reduces to

$$\widehat{X} = \rho\frac{\sigma_X}{\sigma_Y}\left(Y - \mu_Y\right) + \mu_X \tag{1.2.80}$$

where σ_X, σ_Y are standard deviations of X, Y respectively. This formula can be intuitively explained as skewing/standardizing $Y - \mu_Y$ by σ_Y, then reskewing by σ_X and scaling by the correlation coefficient ρ to obtain the estimate $\widehat{X} - \mu_X$.

1.2.6 Scalar Additive Noise Model

Consider the noise model $Y = X + Z$, where $X, Y, Z \in \mathcal{C}$ are scalar-valued random variables. We assume that

- The signal power is P: $\mathbf{E}\left(|X|^2\right) = P$.
- The noise power is N: $\mathbf{E}\left(|Z|^2\right) = N$.
- The signal and noise are uncorrelated: $\mathbf{E}(XZ^*) = 0$.

Note that we do *not* assume the means of X, Z. Now we compute

$$R_{XY} = \mathbf{E}(XY^*) = \mathbf{E}(X\left(X + Z\right)^*) = \mathbf{E}\left(|X|^2 + XZ^*\right) = \mathbf{E}\left(|X|^2\right) + \mathbf{E}(XZ^*) = P \tag{1.2.81}$$

$$R_Y = \mathbf{E}\left(|Y|^2\right) = \mathbf{E}((X+Z)(X+Z)^*) = \mathbf{E}\left(|Z|^2 + |Z|^2 + 2XZ^*\right) = P + N. \tag{1.2.82}$$

Thus the optimal linear estimator of X given Y is

$$\widehat{X}(Y) = \frac{P}{P+N} Y. \tag{1.2.83}$$

Intuitively, we shrink the observation Y down based on how strong the signal is compared to the noise. In the extreme case of $P = 0$, there is no signal, and $\widehat{X} = 0$. At the other extreme of $P/N \to \infty$, the signal-to-noise ratio is as strong as it can be, and we obtain $\widehat{X} = Y$.

In the special case of $X \sim \mathcal{N}(0, P)$, $Z \sim \mathcal{N}(0)N$, and X is independent of Z, Eq. (1.2.83) gives the *optimal Bayes estimator* of X given Y (not constraining ourselves to linear estimators) under squared error loss. In that case, $P = \infty$ corresponds to the maximum likelihood estimator, which corresponds to the case of no prior information on X. We remark that the MSE is

$$\mathbf{E}\left(\left|\widehat{X} - X\right|^2\right) = \frac{PN}{P+N} \tag{1.2.84}$$

and as $P \to \infty$, the MSE approaches N.

We observe that

$$\frac{PN}{P+N} \leq \min\left\{P, N\right\}. \tag{1.2.85}$$

This admits an interesting interpretation. P is the MSE achieved by using constant zero as an estimator, and N is the MSE achieved by using Y as the estimator. The optimal error, clearly, should not be bigger than any of these two numbers.

1.2.7 Vector Additive Noise Model

Consider the setup $Y = HX + Z$, where $Y \in \mathbb{C}^n$, $X \in \mathcal{C}^d$, $H \in \mathcal{C}^{n \times d}$, and $Z \in \mathcal{C}^n$. We assume H has full column rank. This is a generalization of the previous case. Our goal is again to estimate X from Y, and similar to before, we assume that the signal and noise are uncorrelated: $\mathbf{E}(XZ^*) = 0$. From these assumptions, we find that

$$\begin{aligned}
R_Y &= \mathbf{E}((HX+Z)(HX+Z)^*) && (1.2.86)\\
&= H\,\mathbf{E}(XX^*)H^* + \mathbf{E}(ZZ^*) && (1.2.87)\\
&= HR_XH^* + R_Z && (1.2.88)\\
R_{XY} &= \mathbf{E}(X(HX+Z)^*) && (1.2.89)\\
&= \mathbf{E}(XX^*H^*) && (1.2.90)\\
&= R_XH^*. && (1.2.91)
\end{aligned}$$

Hence, we obtain

$$\begin{aligned}
W &= R_{XY}R_Y^{-1} && (1.2.92)\\
&= R_XH\left(HR_XH^* + R_Z\right)^{-1} && (1.2.93)\\
\widehat{X} &= WY && (1.2.94)\\
&= R_XH\left(HR_XH^* + R_Z\right)^{-1}Y. && (1.2.95)
\end{aligned}$$

If we denote by R_e the error auto-correlation matrix,

$$\begin{aligned}
R_e &= R_X - R_{XY}R_Y^{-1}R_{YX} && (1.2.96)\\
&= R_X - R_XH^*\left(HR_XH^* + R_Z\right)^{-1}HR_X. && (1.2.97)
\end{aligned}$$

If both R_X and R_Z are available, there is another set of expressions, usually called the *information form*. Indeed, one can verify that

$$W = \left(H^* R_Z^{-1} H + R_X^{-1}\right)^{-1} H^* R_Z^{-1}. \tag{1.2.98}$$

Recall the Sherman-Morrison-Woodbury formula:

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B\left(C^{-1} + DA^{-1}B\right)DA^{-1} \tag{1.2.99}$$

and apply it to $R_X^{-1} + H^* R_Z^{-1} H$, we have

$$\begin{aligned} \left(R_X^{-1} + H^* R_Z^{-1} H\right)^{-1} &= R_X - R_X H^* \left(H R_X H^* + R_Z\right)^{-1} H R_X & (1.2.100) \\ &= R_e & (1.2.101) \end{aligned}$$

which implies that $W = R_e H^* R_Z^{-1}$, or equivalently

$$R_e^{-1}\widehat{X} = H^* R_Z^{-1} Y. \tag{1.2.102}$$

Notice that the RHS of the above equation does not depend on R_X.

The error autocorrelation reduces to

$$R_e = \frac{1}{\frac{1}{P} + \frac{1}{N}} \tag{1.2.103}$$

in the scalar additive noise case.

The information form has an interesting advantage in deriving formulas combining two estimators together as shown by the following lemma.

Lemma 1.2.5. Let Y_a and Y_b sataisfy

$$\begin{aligned} Y_a &= H_a X + Z_a & (1.2.104) \\ Y_b &= H_b X + Z_b & (1.2.105) \end{aligned}$$

where $\langle X, Z_a\rangle = \langle X, Z_b\rangle = \langle Z_a, Z_b\rangle = 0$. Denote by $\widehat{X}_a$ the optimal linear estimator of X given only Y_a, and $\widehat{X}_b$ the optimal linear estimator of X given only Y_b. Denote the estimation error matrices by $R_{e_a} = \left\langle X - \widehat{X}_a, X - \widehat{X}_a\right\rangle$, $R_{e_b} = \left\langle X - \widehat{X}_b, X - \widehat{X}_b\right\rangle$. Then $\widehat{X}$, the optimal linear estimator of X given both Y_a, Y_b can be found as

$$P^{-1}\widehat{X} = R_{e_a}^{-1}\widehat{X}_a + P_{e_b}^{-1}\widehat{X}_b \tag{1.2.106}$$

where $P = \left\langle X - \widehat{X}, X - \widehat{X}\right\rangle$ satisfies

$$P^{-1} = R_{e_a}^{-1} + P_{e_b}^{-1} - R_X^{-1}. \tag{1.2.107}$$

We will now try to concretely connect the vector case to the scalar case. Assume

$$R_X = PI, \quad R_Z = NI, \tag{1.2.108}$$

where $P, N \in \mathbb{R}$. Then

$$\begin{aligned} W &= PH^* \left(PHH^* + NI\right)^{-1} & (1.2.109) \\ HW &= PHH^* \left(PHH^* + NI\right)^{-1}. & (1.2.110) \end{aligned}$$

Now, using the eigendecomposition of $H = UDU^*$,

$$HH^* = UD^2U^* = U \operatorname{diag}\left(\lambda_1^2, \ldots, \lambda_d^2, 0, \ldots, 0\right) U^*, \tag{1.2.111}$$

so we can simplify:

$$HW = PUD^2U^* \left(PUD^2U^* + NI\right)^{-1} \tag{1.2.112}$$

$$= PUD^2U^* \left(U\left(PD^2+NI\right)U^*\right)^{-1} \tag{1.2.113}$$

$$= U\left(PD^2\left(PD^2+NI\right)^{-1}\right)U^*. \tag{1.2.114}$$

In the scalar case, we had

$$\widehat{X} = \frac{P}{P+N}Y \tag{1.2.115}$$

and in the matrix case we have

$$HW = U\left(PD^2\left(PD^2+NI\right)^{-1}\right)U^* \tag{1.2.116}$$

where

$$PD^2\left(PD^2+NI\right)^{-1} = \operatorname{diag}\left(\frac{P\lambda_1^2}{P\lambda_1^2+N},\dots,\frac{P\lambda_d^2}{P\lambda_d^2+N},0,\dots,0\right). \tag{1.2.117}$$

The second formula is analogous to the first, where we first rotate Y into the eigenbasis, shrink it coordinate-by-coordinate according to the signal-noise ratio, and rotate back into the standard basis. As $P \to \infty$, we get

$$\lim_{P\to\infty} PD^2\left(PD^2+NI\right)^{-1} = \lim_{P\to\infty}\operatorname{diag}\left(\frac{P\lambda_1^2}{P\lambda_1^2+N},\dots,\frac{P\lambda_d^2}{P\lambda_d^2+N},0,\dots,0\right) \tag{1.2.118}$$

$$= \operatorname{diag}(1,\dots,1,0,\dots,0). \tag{1.2.119}$$

And

$$HW = U_dU_d^* = \underbrace{H\left(H^*H\right)^{-1}H^*}_{\text{projection into }\operatorname{Im}(H)} \tag{1.2.120}$$

where U_d is U with the last $n-d$ columns set to zero.

1.2.8 Best Unbiased Linear Estimator

We can show that the estimator

$$\widehat{X}_b := \left(H^*H\right)^{-1}H^*Y \tag{1.2.121}$$

is the *minimum variance unbiased* estimator for X if we assume X is deterministic. In other words, the previous example showing that assuming $R_X = PI$ with $P \to \infty$ is equivalent to assuming no knowledge about X, which is also equivalent to finding the best unbiased estimator.

Theorem 1.2.6 (Gauss-Markov Theorem). Consider model $Y = HX + Z$, where Z is a zero-mean random vector with covariance matrix identity: $\langle Z, Z\rangle = I$, X is a deterministic vector, and H has full column rank. Then Eq. (1.2.121) is the best linear unbiased estimator (BLUE) of X given Y in the sense that if $\widehat{X}$ is another linear unbiased estimator of X given Y, then

$$\left\|\widehat{X}_b - X\right\|^2 \leq \left\|\widehat{X} - X\right\|^2. \tag{1.2.122}$$

By unbiasedness, we mean that the expectation of the estimator is always X, no matter what X actually is.

Proof. Assume $\widehat{X} = KY$ is another linear estimator of X. In order for $\widehat{X}$ to be unbiased, we must require

$$X = \mathbf{E}\left(\widehat{X}\right) \tag{1.2.123}$$

$$= \mathbf{E}(K(HX+Z)) \tag{1.2.124}$$

$$= \mathbf{E}(KHX+KZ) \tag{1.2.125}$$

$$= \mathbf{E}(KHX)+\mathbf{E}(KZ) \tag{1.2.126}$$

$$= KHX + K\,\mathbf{E}(Z) \tag{1.2.127}$$

$$= KHX. \tag{1.2.128}$$

Since X is arbitrary, we know $KH = I$. Then

$$\begin{aligned}
\left\|\widehat{X} - X\right\|^2 &= \langle KY - X, KY - X\rangle && (1.2.129)\\
&= \langle K(HX + Z) - X, K(HX + Z) - X\rangle && (1.2.130)\\
&= \langle KHX + KZ - X, KHX + KZ - X\rangle && (1.2.131)\\
&= \langle X + KZ - X, X + KZ - X\rangle && (1.2.132)\\
&= \langle KZ, KZ\rangle && (1.2.133)\\
&= K\langle Z, Z\rangle K^* && (1.2.134)\\
&= KK^*. && (1.2.135)
\end{aligned}$$

Note that for Eq. (1.2.121), the K matrix is equal to $K_b = H(H^*H)^{-1}$. So we want to show that, for any K such that $KH = I$,

$$KK^* \geq K_bK_b^* = (H^*H)^{-1} H^*H (H^*H)^{-1}. \tag{1.2.136}$$

It is shown by this calculation:

$$\begin{aligned}
KK^* - (H^*H)^{-1} H^*H (H^*H)^{-1} &= KK^* - (H^*H)^{-1} && (1.2.137)\\
&= KK^* - KH(H^*H)^{-1} H^*K^* && (1.2.138)\\
&= K\left(I - H(H^*H)^{-1} H^*\right)K^* && (1.2.139)\\
&\geq 0. && (1.2.140)
\end{aligned}$$

In the last step we used the fact that $H(H^*H)^{-1} H^*$ is a projection matrix (onto $\mathrm{Im}(H)$), so $I - H(H^*H)^{-1} H^*$ is also a projection matrix (onto $\mathrm{Im}(H)^{\perp}$). □

1.3 Gaussian Distribution

A natural question to ask is when this linear estimator is also the actual optimal estimator, i.e. when there is no non-linear estimator better than it. It turns out to be the case when (X, Y) is jointly Gaussian. However, it is not the only case where the optimal linear estimator is also the optimal non-linear estimator. Indeed, if Y only takes values in a set with cardinality two, then any deterministic function of Y can be written as a linear function of Y.

1.3.1 Jointly Gaussian Distribution and Gaussian Random Vectors

Definition 1.3.1 (Gaussian Random Variable). Random variable $X\colon \Omega \to \mathbb{R}$ is Gaussian with mean μ and variance σ^2 if it has probability density function

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}}. \tag{1.3.1}$$

We say $X \sim \mathcal{N}(\mu, \sigma^2)$.

Definition 1.3.2 (Characteristic Function). Let X be a random variable and let $u \in \mathbb{R}$. Then

$$\phi_X(u) := \mathbf{E}\left(\mathrm{e}^{\mathrm{i}uX}\right). \tag{1.3.2}$$

Proposition 1.3.3. If $X \sim \mathcal{N}(\mu, \sigma^2)$ then

$$\phi_X(u) = \exp\left(-\frac{u^2\sigma^2}{2} + \mathrm{i}u\mu\right). \tag{1.3.3}$$

Definition 1.3.4 (Gaussian Random Vector). We say $X\colon \Omega \to \mathbb{R}^d$ is a **Gaussian random vector** if every finite linear combination of the coordinates of X is a Gaussian random variable.

Definition 1.3.5 (Multivariate Gaussian). We say $X\colon \Omega \to \mathbb{R}^d$ is **multivariate Gaussian** with mean μ and covariance matrix Σ if

$$\phi_X(u) = \exp\left(\mathrm{i}\left\langle \mu, u\right\rangle - \frac{\left\langle \Sigma u, u\right\rangle}{2}\right). \tag{1.3.4}$$

If Σ is non-singular, then X has probability density function

$$f_X(x) = \frac{1}{(2\pi)^{d/2}\det(\Sigma)^{1/2}}\exp\left(-\frac{(x-\mu)^*\,\Sigma^{-1}\,(x-\mu)}{2}\right). \tag{1.3.5}$$

Proposition 1.3.6 (Independent ↔ Uncorrelated). If X, Y are jointly multivariate Gaussian, then

$$X \perp\!\!\!\perp Y \iff \Sigma_{XY} = 0. \tag{1.3.6}$$

In other words, uncorrelated is equivalent to independent.

Proposition 1.3.7 (Affine Transformation). If $X \sim \mathcal{N}(\mu, \Sigma)$, then $AX + b \sim \mathcal{N}(A\mu + b, A\Sigma A^*)$.

Remark 1.3.8. If $X \sim \mathcal{N}(\mu, \Sigma)$ and $Z \sim \mathcal{N}(0, I)$, with $\Sigma = U\Lambda U^*$ and $\operatorname{rank}(\Sigma) = r$, then $X \sim U_r\Lambda_r^{1/2}Z$. This reduces the intrinsic dimension of a degenerate Gaussian distribution to a linear combination of lower-rank Gaussians.

Conditional Distribution of Jointly Gaussian Vectors

Theorem 1.3.9. Suppose

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \begin{bmatrix} \Sigma_X & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_Y \end{bmatrix}\right). \tag{1.3.7}$$

Then

$$X \mid Y \sim \mathcal{N}\left(\mu_X + \Sigma_{XY}\Sigma_Y^{-1}\left(Y - \mu_Y\right), \Sigma_X - \Sigma_{XY}\Sigma_Y^{-1}\Sigma_{YX}\right). \tag{1.3.8}$$

It implies that

$$\begin{aligned} \mathbf{E}(X \mid Y) &= \mu_X + \Sigma_{XY}\Sigma_Y^{-1}\left(Y - \mu_Y\right) && (1.3.9) \\ \mathbf{E}((X - \mathbf{E}(X \mid Y))\,(X - \mathbf{E}(X \mid Y))^*) &= \Sigma_X - \Sigma_{XY}\Sigma_Y^{-1}\Sigma_{YX} = \Sigma_e. && (1.3.10) \end{aligned}$$

If Σ_Y is singular then we may replace Σ_Y^{-1} by $\Sigma_Y^{\dagger}$ the pseudoinverse.

Remark 1.3.10. This result shows that to derive the optimal linear estimator, there are "two" possible approaches: one is to use the orthogonality principle, and the other is to assume everything is jointly Gaussian and compute the conditional expectation. Both approaches may be easy to execute for simple problems, but for problems with complicated structures usually one approach will stand out and the idea of using both approaches is usually called the Gaussian trick in the literature.

Remark 1.3.11. For a given mean and covariance structure of (X, Y)m jointly Gaussian distributions attain the *worst* estimation error. Indeed, we have

$$\min_g \mathbf{E}\left(\|X - g(Y)\|^2\right) \le \mathbf{E}\left(\left\|X - \widehat{X}_b\right\|^2\right) = \operatorname{tr}(\Sigma_e) \tag{1.3.11}$$

but in the Gaussian case, this equality is tight.

Note that the value $\Sigma_{X|Y}$ is *constant* in Y, so that the value of Y does not change our uncertainty in our knowledge of X.

Proof of Theorem 1.3.9. To demonstrate the power of the "Gaussian trick", we prove the theorem without using the concrete expressions for the PDF of (X, Y). In particular, (X, Y) does not have a PDF if its joint covariance matrix is singular.

Define the optimal affine estimator $\widehat{X}_L(Y)$ and let $e = X - \widehat{X}_L$. By the orthogonality principle, we know $\mathbf{E}(e) = 0$ and $\operatorname{Cov}(e, Y) = \langle e, Y \rangle = 0$. Since (e, Y) is obtained from (X, Y) from an affine transformation, they are jointly Gaussian. Since $\operatorname{Cov}(e, Y) = 0$, e and Y are independent.

Since $X = e + \widehat{X}_L(Y)$, and we know $\widehat{X}_L$ is a function of Y while e is independent of Y with covariance Σ_e, we know that $e \mid Y \sim e \sim \mathcal{N}(0, \Sigma_e)$. Hence, conditioned on Y, X is nothing but the sum of a deterministic vector $\widehat{X}_L(Y)$ and a Gaussian random vector $e \sim \mathcal{N}(0, \Sigma_e)$, which is distributed as

$$X \sim \mathcal{N}\left(\widehat{X}_L(Y), \Sigma_e\right). \tag{1.3.12}$$

□

1.4 Foundations of Random Processes

1.4.1 Wide-Sense Stationary Processes

Definition 1.4.1 (Discrete-Time Random Process). A **discrete-time random process** is a countably infinite collection of random variables on the same probability space $\{x(n) \colon n \in \mathbb{Z} \text{ or } \mathbb{N}\}$. We define a **mean function** $\mu_n := \mathbf{E}(x(n))$ and a **autocorrelation function** $R_X(n_1, n_2) := \mathbf{E}(x(n_1)x(n_2)^*)$.

Definition 1.4.2 (Wide Sense Stationary). A discrete time random process is **wide sense stationary (WSS)** if:

(1) (Mean is time-invariant.) There exists μ such that $\mu_n = \mu$ for all n.

(2) (Autocorrelation is only a function of the difference.) There exists a one-argument function R_X (abusing notation) such that for all n_1 and n_2, $R_X(n_1, n_2) = R_X(n_1 - n_2)$.

(3) $\|x(n)\|_{L^2(\mathbf{P})} := \mathbf{E}\left(|x(n)|^2\right) < \infty$ for all n.

Remark 1.4.3. The complementary notion to wide-sense stationary (WSS) is something like strict-sense stationary, which means that the joint distribution (as opposed to just the mean and covariance) are invariant to global time shifts.

For jointly Gaussian processes, these are the same properties – wide-sense stationary is equivalent to strict-sense stationary.

Proposition 1.4.4. For a WSS process,

(a) $R_X(k) = R_X(-k)^*$.

(b) $|R_X(k)| \le R_X(0)$.

(c) The following are equivalent:

a) $R_X(d) = R_X(0)$;

b) $\mathbf{P}(x(n + d) = x(n)) = 1$ for all $n \in \mathbb{Z}$;

c) $R_X(d + n) = R_X(n)$ for all $n \in \mathbb{Z}$.

Proof.

Proof of (a).

$$R_X(k) = \mathbf{E}(x(n + k)x(n)^*) \tag{1.4.1}$$

$$\begin{aligned}
&= \mathbf{E}\big((x(n)x(n+k)^*)^*\big) && (1.4.2)\\
&= \mathbf{E}(x(n)x(n+k)^*)^* && (1.4.3)\\
&= R_X(-k)^*. && (1.4.4)
\end{aligned}$$

Proof of (b).

$$\begin{aligned}
|R_X(k)| &= |\mathbf{E}(x(n)x(n-k)^*)| && (1.4.5)\\
&\leq \sqrt{\mathbf{E}(x(n)x(n)^*)}\sqrt{\mathbf{E}(x(n-k)x(n-k)^*)} && (1.4.6)\\
&= \sqrt{R_X(0)R_X(0)} && (1.4.7)\\
&= R_X(0). && (1.4.8)
\end{aligned}$$

Proof of (c). Suppose the first statement is true. Since $R_X(0)$ is real-valued, so is $R_X(d)$, yielding

$$\begin{aligned}
\mathbf{E}\left(|x\,(n+d) - x\,(n)|^2\right) &= \mathbf{E}(x(n+d)x^*(n+d)) - \mathbf{E}(x(n+d)x^*(n)) && (1.4.9)\\
&\quad - \mathbf{E}(x(n)x^*(n+d)) + \mathbf{E}(x(n)x^*(n)) && (1.4.10)\\
&= R_X(0) - R_X(d) - R_X^*(d) + R_X(0) && (1.4.11)\\
&= 0, && (1.4.12)
\end{aligned}$$

which implies the second statement. Since two random variables which are equal with probability 1 have the same expectation, the second statement implies that for any n,

$$R_X(n+d) = \mathbf{E}(x(n+d)x^*(d)) = \mathbf{E}(x(n)x^*(0)) = R_X(n). \tag{1.4.13}$$

The third statement obviously implies the first statement.

□

If we take the discrete time Fourier transform of $R_X(\cdot)$, we get a sequence $S_X(\omega)$ which is called the **power spectral density**.

Definition 1.4.5 (Harmonic Process). Fix $N \in \mathbb{N}$. A **harmonic process** is given by the definitions

$$x(n) = \sum_{m=1}^{N} A_m \cos(\omega_m \cdot n + \phi_m). \tag{1.4.14}$$

Here the collections $(A_m)_{m\in(n)}$ and $(\phi_m)_{m\in(n)}$ are mutually independent random variables, but ω_m is deterministic. Further properties are that $\phi_m \sim \mathcal{U}([-\pi,\pi])$ and $\mathbf{E}(A_m) = 0$ and $\mathbf{E}(A_n A_m) = \sigma_m^2 \delta_{mn}$.

The DFT of a harmonic process ought to be a bunch of Dirac deltas at ω_m, otherwise it is a bad idea to define the power spectral density as we have done. This is in some sense the most fundamental WSS as all others can be represented in this form.

Proposition 1.4.6. If $(x(n))_{n\in\mathbb{N}}$ is harmonic then it is WSS.

Proof. We compute

$$\begin{aligned}
\mu_n &= \mathbf{E}(x(n)) && (1.4.15)\\
&= \sum_{m=1}^{N} \mathbf{E}(A_m \cos(\omega_m \cdot n + \phi_m)) && (1.4.16)
\end{aligned}$$

$$= \sum_{m=1}^{N} \mathbf{E}(A_m)\,\mathbf{E}(\cos(\omega_m \cdot n + \phi_m)) \tag{1.4.17}$$

$$= \sum_{m=1}^{N} 0 \cdot 0 = 0. \tag{1.4.18}$$

$$R_X(k) = \mathbf{E}(X(n+k)X(n)^*) \tag{1.4.19}$$

$$= \sum_{m=1}^{N} \sum_{\ell=1}^{N} \mathbf{E}(A_m A_\ell \cos(\omega_m(n+k) + \phi_m)\cos(\omega_\ell n + \phi_\ell)) \tag{1.4.20}$$

$$= \sum_{m=1}^{N} \sum_{\substack{\ell=1 \\ \ell\neq m}}^{N} \mathbf{E}(A_m A_\ell \cos(\omega_m(n+k) + \phi_m)\cos(\omega_\ell n + \phi_\ell)) \tag{1.4.21}$$

$$+ \sum_{m=1}^{N} \mathbf{E}\big(A_m^2 \cos(\omega_m(n+k) + \phi_m)\cos(\omega_m n + \phi_m)\big) \tag{1.4.22}$$

$$= \sum_{m=1}^{N} \sum_{\substack{\ell=1 \\ \ell\neq m}}^{N} \mathbf{E}(A_m)\,\mathbf{E}(A_\ell)\,\mathbf{E}(\cos(\omega_m(n+k) + \phi_m))\,\mathbf{E}(\cos(\omega_\ell n + \phi_\ell)) \tag{1.4.23}$$

$$+ \sum_{m=1}^{N} \mathbf{E}\big(A_m^2 \cos(\omega_m(n+k) + \phi_m)\cos(\omega_m n + \phi_m)\big) \tag{1.4.24}$$

$$= \sum_{m=1}^{N} \mathbf{E}\big(A_m^2 \cos(\omega_m(n+k) + \phi_m)\cos(\omega_m n + \phi_m)\big) \tag{1.4.25}$$

$$= \sum_{m=1}^{N} \mathbf{E}\big(A_m^2\big)\,\mathbf{E}(\cos(\omega_m(n+k) + \phi_m)\cos(\omega_m n + \phi_m)) \tag{1.4.26}$$

$$= \frac{1}{2}\sum_{m=1}^{N} \mathbf{E}\big(A_m^2\big)\,\mathbf{E}(\cos(\omega_m(2n+k) + 2\phi_m) + \cos(\omega_m k)) \tag{1.4.27}$$

$$= \frac{1}{2}\sum_{m=1}^{N} \mathbf{E}\big(A_m^2\big)\,(\mathbf{E}(\cos(\omega_m(2n+k) + 2\phi_m)) + \cos(\omega_m k)) \tag{1.4.28}$$

$$= \frac{1}{2}\sum_{m=1}^{N} \mathbf{E}\big(A_m^2\big)\cos(\omega_m k) \tag{1.4.29}$$

$$= \frac{1}{2}\sum_{m=1}^{N} \sigma_m^2 \cos(\omega_m k). \tag{1.4.30}$$

This is just a function of k. □

The DFT of this $(R_X(n))_{n\in\mathbb{N}}$ is

$$S_X(\omega) = \frac{\pi}{2}\sum_{m=1}^{N} \sigma_m^2\left(\delta(\omega - \omega_m) + \delta(\omega + \omega_m)\right). \tag{1.4.31}$$

It is only nonzero at ω_m and $-\omega_m$, corresponding to the signal only having power at those frequencies.

Interested readers may have observed that the harmonic process is in fact a **deterministic process**, in the sense that once observing a finite interval of X, one can determine the unique random variables $(A_m)_{m\in(n)}$, $(\phi_m)_{m\in(n)}$, thus determining all the future values of X. We will later show that not only WSS process with line spectrum (a countable collection of delta functions) is deterministic, but also any bandlimited WSS process. Here by bandlimited we mean that $S_X(\omega) = 0$ for a set of positive Lebesgue measure.

1.4.2 Spectral Representation of Random Processes

Given a *deterministic* sequence $\{x(n)\colon n \in \mathbb{Z}\}$, its DTFT is given by Eq. (1.1.1). It is associated with an **autocorrelation function**

$$a(n) = \sum_{k\in\mathbb{Z}} x(n)x^*(k-n). \tag{1.4.32}$$

The DTFT of this function is

$$A(\omega) = X(\omega)X^*(\omega) = |X(\omega)|^2 \geq 0. \tag{1.4.33}$$

This quantity $A(\omega)$ is called the **energy spectral density** (ESD). This name is justified through the IDTFT relation:

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} A(\omega)\,\mathrm{d}\omega = a(0) = \sum_{n\in\mathbb{Z}} |x(n)|^2\,. \tag{1.4.34}$$

If $\{x(n)\colon n \in \mathbb{Z}\}$ is a wide sense stationary (WSS) process, we cannot compute the energy density. In particular, the total energy of $x(n)$ is ∞ except in trivial cases:

$$\mathbf{E}\left(\sum_{n\in\mathbb{Z}} |x(n)|^2\right) = \sum_{n\in\mathbb{Z}} \mathbf{E}\left(|x(n)|^2\right) = \sum_{n\in\mathbb{Z}} \mathbf{E}\left(|x(0)|^2\right) = \infty \tag{1.4.35}$$

unless $x(0) = 0$ (almost surely).

Theorem 1.4.7 (Wiener-Khinchin Theorem). Suppose we truncate sequence $\{x(n)\colon n \in \mathbb{Z}\}$ from $-T$ to T, replacing $x(n)$ by $x_T(n) := x(n)1_{n\in[-T,T]}$. Then the ESD $A_T(\omega)$ obeys

$$\lim_{T\to\infty} \frac{\mathbf{E}(A_T(\omega))}{2T+1} = S_X(\omega). \tag{1.4.36}$$

Proof. Note that the total energy of $\{x_T(n)\colon n \in \mathbb{Z}\}$ is finite on average. The DTFT of the truncated signal is

$$X_T(\omega) = \sum_{n\in\mathbb{Z}} x_T(n)\mathrm{e}^{-\mathrm{i}\omega n} = \sum_{n=-T}^{T} x(n)\mathrm{e}^{-\mathrm{i}\omega n}. \tag{1.4.37}$$

The ESD $A_T(\omega)$ of this truncated signal is

$$A_T(\omega) = |X_T(\omega)|^2 = \left(\sum_{n=-T}^{T} x(n)\mathrm{e}^{-\mathrm{i}\omega n}\right)\left(\sum_{n=-T}^{T} x^*(n)\mathrm{e}^{\mathrm{i}\omega n}\right) \tag{1.4.38}$$

$$= \sum_{m=-T}^{T}\sum_{n=-T}^{T} x_n x_m^* \mathrm{e}^{-\mathrm{i}\omega(n-m)}. \tag{1.4.39}$$

$$\mathbf{E}(A_T(\omega)) = \sum_{m=-T}^{T}\sum_{n=-T}^{T} R_X(n-m)\mathrm{e}^{-\mathrm{i}\omega(n-m)}. \tag{1.4.40}$$

Here we introduce the power spectral density by normalizing using a factor of $2T+1$. This represents the transition from energy to power.

$$\frac{1}{2T+1}\mathbf{E}(A_T(\omega)) = \frac{1}{2T+1}\sum_{m=-T}^{T}\sum_{n=-T}^{T} R_X(n-m)\mathrm{e}^{-\mathrm{i}\omega(n-m)} \tag{1.4.41}$$

$$= \sum_{n=-2T}^{T} R_X(n)\mathrm{e}^{-\mathrm{i}\omega n}\left(1 - \frac{|n|}{2T+1}\right). \tag{1.4.42}$$

If $\|R_X\|_{\ell^1} = \sum_{n\in\mathbb{Z}} |R_X(n)| < \infty$, then it follows from the dominated convergence theorem that

$$\lim_{T\to\infty} \frac{1}{2T+1} \mathbf{E}(A_T(\omega)) = \sum_{n\in\mathbb{Z}} R_X(n)\mathrm{e}^{-\mathrm{i}\omega n} \tag{1.4.43}$$

$$= S_X(\omega). \tag{1.4.44}$$

□

Remark 1.4.8. The Wiener-Khinchin Theorem justifies the name "Power Spectral Density" for $S_X(\omega)$.

Theorem 1.4.9 (Cramer-Khinichin Decomposition). The Cramer-Khinchin decomposition of a signal $\{x(n) \colon n \in \mathbb{Z}\}$ writes

$$\int_{-T}^{T} \mathrm{e}^{\mathrm{i}\omega n}\, \mathrm{d}F(\omega) \tag{1.4.45}$$

where for any interval $[\omega_1, \omega_2] \subseteq [-\pi, \pi]$ and $[\omega_3, \omega_4] \subseteq [-\pi, \pi]$,

$$\mathbf{E}((F(\omega_2) - F(\omega_1))\,(F(\omega_4) - F(\omega_3))^*) = f((\omega_1, \omega_2] \cap (\omega_3, \omega_4]), \tag{1.4.46}$$

where f is called the **structural measure** of the stochastic process $F(\omega)$ and obeys the following Radon-Nikodym derivative (where λ is the Lebesgue measure):

$$\frac{\mathrm{d}f}{\mathrm{d}\lambda}(\omega) = \frac{S_X(\omega)}{2\pi}. \tag{1.4.47}$$

Remark 1.4.10. In some sense we can say that $\mathbf{E}\left(|\mathrm{d}F(\omega)|^2\right) = \frac{S_X(\omega)\,\mathrm{d}\omega}{2\pi}$. Thus the weights of the decomposition are proportional to the power spectral density of $x(n)$'s.

Computing the autocorrelation function,

$$R_X(k) = \mathbf{E}(x(n)x^*(n-k)) \tag{1.4.48}$$

$$= \mathbf{E}\left(\left(\int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega_1 n}\, \mathrm{d}F(\omega_1)\right)\left(\int_{-\pi}^{\pi} \mathrm{e}^{-\mathrm{i}\omega_2 (n-k)}\, \mathrm{d}F^*(\omega_2)\right)\right) \tag{1.4.49}$$

$$= \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega_1 n}\mathrm{e}^{-\mathrm{i}\omega_2(n-k)}\, \mathrm{d}F(\omega_1)\, \mathrm{d}F^*(\omega_2) \tag{1.4.50}$$

$$= \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega_1 n}\mathrm{e}^{-\mathrm{i}\omega_2(n-k)}\, \langle \mathrm{d}F(\omega_1), \mathrm{d}F(\omega_2)\rangle \tag{1.4.51}$$

$$= \int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega k}\, \mathrm{d}[F](\omega) \tag{1.4.52}$$

$$= \int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega k} \frac{S_X(\omega)}{2\pi}\, \mathrm{d}\omega \tag{1.4.53}$$

which is the IDTFT of $S_X(\omega)$. Here $[F]$ is the quadratic variation of F.

Remark 1.4.11. We cannot represent the signal in terms of cosines and sines — they don't fully ensure independence of quantities of ω and $-\omega$. This is because $e^{\mathrm{i}\omega}$ is the eigenfunction of an LTI operator (for example the derivative).

Now we want to concretely do a decomposition using the Harmonic Process. Indeed,

$$x(n) = \sum_{m\in[N]} A_m \cos(\omega_m n + \phi_m) \tag{1.4.54}$$

$$= \sum_{m\in[N]} \frac{A_m}{2}\left(\mathrm{e}^{\mathrm{i}(\omega_m n+\phi_m)} + \mathrm{e}^{-\mathrm{i}(\omega_m n+\phi_m)}\right) \tag{1.4.55}$$

$$= \sum_{m\in[N]} \left(A_m \frac{\mathrm{e}^{\mathrm{i}\phi_m}}{2}\right)\mathrm{e}^{\mathrm{i}\omega_m n} + \sum_{m\in[N]} \left(A_m \frac{\mathrm{e}^{-\mathrm{i}\phi_m}}{2}\right)\mathrm{e}^{-\mathrm{i}\omega_m n}. \tag{1.4.56}$$

In some sense $\mathrm{d}F(\omega_m) = A_m \frac{\mathrm{e}^{\mathrm{i}\phi_m}}{2}$ along with some δ functions to make it precise (since we are going from integral to sum).

Definition 1.4.12 (z-Cross-Spectrum). Suppose $\{x(n)\colon n\in\mathbb{Z}\}$ and $\{y(n)\colon n\in\mathbb{Z}\}$ are jointly WSS. We define the **z-cross-spectrum** as

$$S_{XY}(z) := \sum_{n\in\mathbb{Z}} R_{XY}(n) z^{-k}, \tag{1.4.57}$$

where $R_{XY}(k) = \mathbf{E}\left(x_n y_{n-k}^*\right)$ as usual.

Then we have

$$\begin{aligned} S_{XY}(z) &= S_{YX}^*\left(z^{-*}\right) &(1.4.58)\\ S_{XY}(\omega) &= S_{YX}^*(\omega). &(1.4.59)\end{aligned}$$

Theorem 1.4.13. Let $y(n)$ be a stochastic process obtained by passing a WSS process $x(n)$ through a stable LTI (Linear Time-Invariant) system with impulse response $h(n)$ (transfer function $H(z)$, which is the z-transform of h). In other words,

$$\begin{aligned} Y(n) &= (h * x)(n) &(1.4.60)\\ &= \sum_{k\in\mathbb{Z}} h(k)x(n-k). &(1.4.61)\end{aligned}$$

Then

$$\begin{aligned} S_Y(z) &= H(z)S_X(z)H^*((1/z)^*) &(1.4.62)\\ S_{YX}(z) &= H(z)S_X(z) &(1.4.63)\\ S_{XY}(z) &= H^*((1/z)^*)S_X^*((1/z)^*) &(1.4.64)\\ S_Y(\omega) &= \left|H(\omega)\right|^2 S_X(\omega) &(1.4.65)\\ S_{YX}(\omega) &= H(\omega)S_X(\omega) &(1.4.66)\\ S_{XY}(\omega) &= S_X(\omega)H^*(\omega). &(1.4.67)\end{aligned}$$

Finally, if $z(n)$ is jointly WSS with $(x(n), y(n))$ as defined, we have

$$S_{ZY}(z) = \S_{ZX}(z)H^*((1/z)^*). \tag{1.4.68}$$

2 Wiener Filter

2.1 Noncausal Wiener Filter

Suppose we observe a whole scalar WSS process $y(n)$ and want to estimate a WSS process $x(n)$. We do this by passing it through an LTI filter $h(n)$.

$$\widehat{x}(n) = \sum_{m\in\mathbb{Z}} h(m)y(n-m). \tag{2.1.1}$$

To compute the optimal coefficients $h(n)$, we use the orthogonality principle

$$\langle x(n) - \widehat{x}(n), y(n-k)\rangle = \mathbf{E}((x(n) - \widehat{x}(n))\, y^*(n-k)) = 0 \quad \text{for all } k \in \mathbb{Z}. \tag{2.1.2}$$

It follows that

$$\langle x(n), y^*(n-k)\rangle = \mathbf{E}(x(m)y^*(n-k)) = \sum_{m\in\mathbb{Z}} h(m)\, \mathbf{E}(y(n-m)y^*(n-k)). \tag{2.1.3}$$

This can also be defined as

$$\langle x(n), y^*(n-k)\rangle = R_{XY}(k) = \sum_{m\in\mathbb{Z}} h(m)R_Y(k-m) \quad \text{for all } k. \tag{2.1.4}$$

Taking the DTFT,

$$S_{XY}(\omega) = H(\omega)S_Y(\omega), \tag{2.1.5}$$

which shows that the transfer function of the noncausal Wiener filter is

$$H(\omega) = \frac{S_{XY}(\omega)}{S_Y(\omega)}. \tag{2.1.6}$$

We *did not* assume the WSS processes X and Y are zero-mean.

This derivation is subtly wrong because it is possible that the DTFT does not exist (in which case there exists a non-Wiener solution to the earlier equations), or that $S_Y(\omega) = 0$ while $S_{XY}(\omega) \neq 0$. In engineering this problem never happens and it's okay, but we will understand the theory in a little bit.

This should be reminiscent of the optimal linear estimator in the scalar case:

$$\widehat{X}(Y) = \frac{\mathbf{E}(XY^*)}{\mathbf{E}(YY^*)}Y. \tag{2.1.7}$$

Here it is clear that $\mathbf{E}(XY^*) = 0$ whenever $\mathbf{E}(YY^*) = 0$ due to Cauchy-Schwarz.

Spectral Representation of Jointly Distributed Stochastic Processes

We invoke the Cramer-Khinchin representation of stochastic processes. Write

$$x(n) = -\int_{\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega n}\, \mathrm{d}F_X(\omega) \tag{2.1.8}$$

$$y(n) = -\int_{\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega n}\, \mathrm{d}F_Y(\omega). \tag{2.1.9}$$

Claim 2.1.1. If X and Y are jointly WSS, then

$$\langle \mathrm{d}F_X(\omega_1), \mathrm{d}F_Y(\omega_2)\rangle = 0 \tag{2.1.10}$$

if $\omega_1 \neq \omega_2$.

Then

$$\begin{aligned}
R_{XY}(k) &= \mathbf{E}(x(n)y^*(n-k)) && (2.1.11)\\
&= \mathbf{E}\left(\int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega_1 n}\,\mathrm{d}F_X(\omega) \int_{-\pi}^{\pi} \mathrm{e}^{-\mathrm{i}\omega_2(n-k)}\,\mathrm{d}F_Y^*(\omega_2)\right) && (2.1.12)\\
&= \mathbf{E}\left(\int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega k}\,\mathrm{d}F_X(\omega)\,\mathrm{d}F_Y^*(\omega)\right) && (2.1.13)\\
&= \mathbf{E}\left(\int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega k}\,\mathrm{d}[F_X\,\mathrm{d}F_Y](\omega)\right) && (2.1.14)\\
&= \int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega k}\,\mathrm{d}[F_X\,\mathrm{d}F_Y](\omega) && (2.1.15)\\
&= \int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega k}\frac{S_{XY}(\omega)\,\mathrm{d}\omega}{2\pi}. && (2.1.16)
\end{aligned}$$

Hence $S_{XY}(\omega)$ is proportional to $\mathrm{d}[F_X\,\mathrm{d}F_Y](\omega)$. Here $[F_X, F_Y]$ is the **quadratic covariation** of F_X and F_Y.

Remark 2.1.2. This provides a nice interpretation: a Wiener filter is like a *change of basis* to a basis provided by $\{x(n) \colon n \in \mathbb{N}\}$, namely $[F_X(\omega), F_Y(\omega)]$. This reduces an infinitely large problem to a one-dimensional problem.

Error of Non-Causal Wiener Filter

We have

$$\begin{aligned}
0 &\leq \mathbf{E}\left(\left|x(n) - \widehat{x}(n)\right|^2\right) && (2.1.17)\\
&= \mathbf{E}(x(n)x^*(n) - x(n)\widehat{x}^*(n) - \widehat{x}(n)x^*(n) + \widehat{x}(n)\widehat{x}^*(n)) && (2.1.18)\\
&= \mathbf{E}(x(n)x^*(n)) - \mathbf{E}(\widehat{x}(n)\widehat{x}^*(n)) && (2.1.19)
\end{aligned}$$

since by orthogonality

$$\mathbf{E}(x(n)\widehat{x}^*(n)) = \mathbf{E}(\widehat{x}(n)\widehat{x}^*(n)) = \mathbf{E}(\widehat{x}(n)\widehat{x}^*(n)). \tag{2.1.20}$$

Note that

$$\begin{aligned}
\mathbf{E}(x(n)x^*(n)) - \mathbf{E}(\widehat{x}(n)\widehat{x}^*(n)) &= R_X(0) - R_{\widehat{X}}(0) && (2.1.21)\\
&= \frac{1}{2\pi}\left(\int_{-\pi}^{\pi} S_X\,\mathrm{d}\omega - \int_{-\pi}^{\pi} S_{\widehat{X}}(\omega)\,\mathrm{d}\omega\right) && (2.1.22)\\
&= \frac{1}{2\pi}\left(\int_{-\pi}^{\pi} S_X\,\mathrm{d}\omega - \int_{-\pi}^{\pi} \left|H(\omega)\right|^2 S_Y(\omega)\,\mathrm{d}\omega\right) && (2.1.23)\\
&= \frac{1}{2\pi}\left(\int_{-\pi}^{\pi} S_X\,\mathrm{d}\omega - \int_{-\pi}^{\pi} \left|\frac{S_{XY}(\omega)}{S_Y(\omega)}\right|^2 S_Y(\omega)\,\mathrm{d}\omega\right) && (2.1.24)\\
&= \frac{1}{2\pi}\left(\int_{-\pi}^{\pi} \left(S_X(\omega) - \frac{\left|S_{XY}(\omega)\right|^2}{S_Y^2(\omega)} S_Y(\omega)\right)\mathrm{d}\omega\right) && (2.1.25)\\
&= \frac{1}{2\pi}\left(\int_{-\pi}^{\pi} \left(1 - \frac{\left|S_{XY}(\omega)\right|^2}{S_X(\omega)S_Y(\omega)}\right)\mathrm{d}\omega\right) && (2.1.26)\\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} \left(1 - \rho^2(\omega)\right) S_X(\omega)\,\mathrm{d}\omega. && (2.1.27)
\end{aligned}$$

Here

$$\rho(x, y) = \frac{S_{XY}(\omega)}{\sqrt{S_X(\omega) S_Y(\omega)}} \tag{2.1.28}$$

is the analogue of the Pearson correlation coefficient, and should be viewed as the Pearson correlation coefficient between $\mathrm{d}[F_X](\omega)$ and $\mathrm{d}[F_Y](\omega)$. This is an energy reduction operation on S_X at each frequency.

Fourier Transform as Isometric Isomorphism

Fourier analysis is an indispensable tool for analyzing WSS stochastic processes. The gist of it is captured by the Cramer–Khinchin decomposition. It maps any WSS process to a stochastic process F with orthogonal increments, which allows us to do filter on these increments F instead of on X. Doing filtering on the orthogonal increments F has a clear advantage: it reduces to a scalar linear optimal estimation problem, and we can deal with each frequency separately.

To make this transparent, consider the following argument. Let $(x(n), y(n))$ be jointly WSS. We want to construct $\{y(n) \colon n \in \mathbb{Z}\}$ to linearly estimate X_n. We want to construct the estimator

$$\widehat{x}(n) = \sum_{k \in \mathbb{Z}} h(k) y(n-k). \tag{2.1.29}$$

Applying the representation of $y(n)$, we have

$$\begin{aligned}
\widehat{x}(n) &= \sum_{k \in \mathbb{Z}} h(k) y(n-k) && (2.1.30)\\
&= \sum_{k \in \mathbb{Z}} h(k) \int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega(n-k)} \,\mathrm{d}F_Y(\omega) && (2.1.31)\\
&= \int_{-\pi}^{\pi} \left(\sum_{k \in \mathbb{Z}} h(k) \mathrm{e}^{-\mathrm{i}\omega k} \right) \mathrm{e}^{\mathrm{i}\omega n} \,\mathrm{d}F_Y(\omega) && (2.1.32)\\
&= \int_{-\pi}^{\pi} H(\omega) \mathrm{e}^{\mathrm{i}\omega n} \,\mathrm{d}F_Y(\omega). && (2.1.33)
\end{aligned}$$

Hence, the problem of non-causal Wiener filter is reduced to

$$\text{estimate} \quad \int_{-\pi}^{\pi} \mathrm{e}^{\mathrm{i}\omega n} \,\mathrm{d}F_X(\omega) \quad \text{using} \quad \int_{-\pi}^{\pi} H(\omega) \mathrm{e}^{\mathrm{i}\omega n} \,\mathrm{d}F_2(\omega). \tag{2.1.34}$$

Since F_X and F_Y have jointly orthogonal increments, it implies that the non-causal Wiener filter transfer function $H(\omega)$ should be the optimal linear estimator of $\mathrm{d}F_1(\omega)$ using $\mathrm{d}F_2(\omega)$. It follows from the scalar case that $H(\omega)$ is given by

$$\begin{aligned}
H(\omega) &= \frac{\mathbf{E}(\mathrm{d}[F_X \,\mathrm{d}F_Y](\omega))}{\mathbf{E}(\mathrm{d}[F](\omega))} && (2.1.35)\\
&= \frac{S_{XY}(\omega) \,\mathrm{d}\omega/(2\pi)}{S_Y(\omega) \,\mathrm{d}\omega/(2\pi)} && (2.1.36)\\
&= \frac{S_{XY}(\omega)}{S_Y(\omega)}. && (2.1.37)
\end{aligned}$$

2.2 Causal Wiener Filter

2.2.1 Review

Recall that the z transform of a power spectral density is

$$S_Y(z) := \sum_{n \in \mathbb{Z}} R_Y(n) z^{-n}. \tag{2.2.1}$$

Here $R_Y(n)$ is the auto-correlation function.

2.2.2 Introduction

In non-causal Wiener filtering, we have the whole history of observations $(y(n))_{n\in\mathbb{Z}}$. This is not always possible for practical purposes. Now we want the optimal causal estimate of $x(n)$ based on $y_{-\infty}^{n} = (y(k))_{k\in\mathbb{Z}:\, k\leq n}$. We use the same filter approach:

$$\widehat{x}(n) = \sum_{k=0}^{\infty} h(k)y(n-k). \tag{2.2.2}$$

By the orthogonality principle,

$$\begin{aligned} 0 &= \langle x(n) - \widehat{x}(n), y(n-k)\rangle \quad \text{for all } k \geq 0 && (2.2.3)\\ &= \mathbf{E}((x(n) - \widehat{x}(n))\, y^*(n-k)) \quad \text{for all } k \geq 0 && (2.2.4)\\ \implies R_{XY}(n) &= \sum_{i=0}^{\infty} R_Y(n-i)h(i). && (2.2.5) \end{aligned}$$

We can't take the Fourier transform because these relationships are not guaranteed to be true for $k < 0$.

2.2.3 Spectral Factorization

Definition 2.2.1. A system $L(z)$ is **minimum phase** if both L and L^{-1} are causal and stable.

For rational z-transform, it is equivalent to saying that all zeros and poles of the z-spectrum are strictly inside the unit circle.

Theorem 2.2.2. For $S_Y(z)$ rational in z, finite power, strictly positive on the unit circle ($S_Y(\mathrm{e}^{j\omega}) > 0$, $\omega \in (-\pi, \pi]$), then there exists the unique canonical spectral factorization of $S_Y(z)$:

$$S_Y(z) = L(z)r_e L^*((1/z^*)). \tag{2.2.6}$$

where $L(z)$ is minimal phase, $r_e > 0$, and $L(\infty) = 1$.

Remark 2.2.3. Since we have assumed $S_Y(z)$ is rational, Y has to be zero-mean. The z-transform of the constant sequence is not defined.

Remark 2.2.4. Finite power assumption means

$$\int_{-\pi}^{\pi} S_Y\big(\mathrm{e}^{\mathrm{i}\omega}\big)\,\mathrm{d}\omega < \infty. \tag{2.2.7}$$

Remark 2.2.5. The condition $L(\infty) = 1$ implies that $L(z)$ has the form

$$L(z) = 1 + \sum_{n\in\mathbb{N}} l_n z^{-n} \tag{2.2.8}$$

r_e is a positive constant which captures the prediction error, or the power of the innovations process (hence the name r_e, as we will discuss in lectures later.

If we define

$$\begin{aligned} S_Y^+(\omega) &:= L\big(\mathrm{e}^{\mathrm{i}\omega}\big)\sqrt{r_e} && (2.2.9)\\ S_Y^-(\omega) &:= \big(S_Y^+(\omega)\big)^* && (2.2.10) \end{aligned}$$

then

$$S_Y(\omega) = S_Y^+(\omega)S_Y^-(\omega) \tag{2.2.11}$$

$$\left|S_Y^+(\omega)\right|^2 = S_Y(\omega) \tag{2.2.12}$$

$$\left|S_Y^-(\omega)\right|^2 = S_Y(\omega) \tag{2.2.13}$$

$$S_Y^+(\omega) = \text{causal, stable} \tag{2.2.14}$$

$$\frac{1}{S_Y^+(\omega)} = \text{causal, stable} \tag{2.2.15}$$

$$S_Y^-(\omega) = \text{anti-causal, stable} \tag{2.2.16}$$

$$\frac{1}{S_Y^-(\omega)} = \text{anti-causal, stable.} \tag{2.2.17}$$

Here we use “anti-causal” to say that $S_Y^{-1}(\omega) = 0$ depends only on the present and future, and crucially not the past.

2.2.4 Solving Causal Wiener Filter

Now we try to solve causal Wiener filter. For all n,

$$R_{XY}(n) - \sum_{i=0}^{\infty} h(i) R_Y(n-i) = f(n) = \begin{cases} 0 & k \geq 0 \\ ? & k < 0 \end{cases}. \tag{2.2.18}$$

Here f is strictly anti-causal. Takking DTFT on both sides,

$$F(\omega) = S_{XY}(\omega) - H(\omega) S_Y(\omega) \tag{2.2.19}$$

$$\frac{F(\omega)}{S_Y^{-1}(\omega)} = \frac{S_{XY}(\omega)}{S_Y^-(\omega)} - H(\omega) S_Y^+(\omega). \tag{2.2.20}$$

Definition 2.2.6 (Truncation). Suppose

$$H(\omega) = \sum_{n \in \mathbb{Z}} a_n \mathrm{e}^{-\mathrm{i}\omega n} \tag{2.2.21}$$

then we define

$$[H(\omega)]_+ = \sum_{n=0}^{\infty} a_n \mathrm{e}^{-\mathrm{i}\omega n}. \tag{2.2.22}$$

In this way we have the truncated signal $[H]_+$.

Since $H(\omega)$ is causal and $S_Y^+(\omega)$ is causal, their cascade is causal. Since $F(\omega)$ is strictly anti-causal ($f(0) = 0$) and $\frac{1}{S_Y^{-1}(\omega)}$ is anti-causal, their cascade is strictly anti-causal. Therefore

$$\left[\frac{F(\omega)}{S_Y^{-1}(\omega)}\right]_+ = \left[\frac{S_{XY}(\omega)}{S_Y^-(\omega)} - H(\omega) S_Y^+(\omega)\right]_+ \tag{2.2.23}$$

$$0 = \left[\frac{S_{XY}(\omega)}{S_Y^-(\omega)}\right]_+ - H(\omega) S_Y^+(\omega) \tag{2.2.24}$$

and we have obtained the formula for causal Wiener filter:

$$H(\omega) = \frac{1}{S_Y^+(\omega)} \left[\frac{S_{XY}(\omega)}{S_Y^{-1}(\omega)}\right]_+. \tag{2.2.25}$$

Next, we will discuss an intuitive way to understand causal Wiener filter, which was first proposed by Bode and Shannon.

2.2.5 Concrete Computations

We assumed $S_Y(z)$ is rational.

Proposition 2.2.7. $S_Y(z)$ can be written as

$$r_e \frac{\prod_{i=1}^{m}(z-\alpha_i)\left(z^{-1}-\alpha_i^*\right)}{\prod_{i=1}^{n}(z-\beta_i)\left(z^{-1}-\beta_i^*\right)}, \tag{2.2.26}$$

for $|\alpha_i|<1$, $|\beta_i|<1$, and $r_e>0$.

Proof. Since $S_Y(z) = S_Y^*(1/z^*)$, which is satisfied by the z-spectrum of any WSS process, for every pole (resp. zero) at $z=\alpha$, there must be a pole (resp. zero) at $z=1/\alpha^*$.

Since the process has finite order by assumption, there can be no poles on the unit circle.

Since $S_Y(z)$ is non-negative on the unit circle, any zeros on the unit circle must be of even multiplicity since $\alpha = 1/\alpha^*$ for any $|\alpha|=1$. Our assumption that the z-spectrum is positive on the unit circle rules out the possibility of any unit-circle zeros.

The constant scaling factor in the z-spectrum must be positive since $S_Y(z)$ is positive on the unit circle. □

Then

$$L(z) = z^{n-m}\frac{\prod_{i=1}^{m}(z-\alpha_i)}{\prod_{i=1}^{n}(z-\beta_i)}. \tag{2.2.27}$$

Hence

$$S_Y(z) = L(z) r_e L(z)^*. \tag{2.2.28}$$

Theorem 2.2.8. Any rational $H(z)$ which is stable and whose degree of numerator is $\leq$ that of the denominator has the following partial fraction expansion:

$$H(z) = r_0 + \sum_{i=1}^{m}\sum_{k=1}^{\ell_i}\frac{r_{ik}}{(z-p_i)^k}. \tag{2.2.29}$$

Note we also need $p_i \neq 1$. There are m poles of order ℓ_i.

Theorem 2.2.9. Let $\alpha \in \mathbb{C}$ and $i \geq 1$. Then

$$\left[\frac{1}{(z+\alpha)^i}\right]_+ = \begin{cases} \frac{1}{(z+\alpha)^i} & |\alpha|<1 \\ \frac{1}{\alpha^i} & |\alpha|>1 \end{cases}. \tag{2.2.30}$$

This gives us a really efficiently way to calculate $S_Y^+(\omega)$, $S_Y^{-1}(\omega)$, and $\left[\frac{S_{XY}(\omega)}{S_Y^-(\omega)}\right]_+$. Thus we can compute the causal Wiener filter efficiently.

2.2.6 Whitening and Causal Estimate Interpretation

In literature, L is the "(canonical) modelling filter" and L^{-1} is called "(canonical) whitening filter". Why is L^{-1} called the "whitening filter"?

We would like to decompose the causal Wiener filter into the cascades of two causal filters. We would like to use a causal filter A to process y to obtain a white noise stochastic process w ("whitening"). By white noise we mean $S_W(\omega)=1$. Assuming no information is lost in this process, we then perform the optimal causal filter of $x(n)$ given $w(n)$ ("modelling").

Recall that

$$\left|\frac{1}{S_Y^+(\omega)}\right|^2 S_Y(\omega) = 1. \tag{2.2.31}$$

Hence $\frac{1}{S_Y^+(\omega)}$ can perform the whitening operation.

Similarly, we prefer to call $S_Y^+(\omega)$ instead of $L(z)$ the modelling filter, since if we pass a white noise process w through L, then we obtain a stochastic process with power spectral density

$$S_W(\omega)\left|S_Y^+(\omega)\right|^2 = S_Y(\omega). \tag{2.2.32}$$

What is the benefit of first doing whitening and then computing the causal Wiener filter from w to x? It turns out that one can very easily obtain the causal Wiener filter from a white noise process from the noncausal Wiener filter. Concretely, we have the following theorem.

Theorem 2.2.10. Denote the non-causal Wiener filter of x given y as

$$\widehat{x}_{\mathrm{NC}}(n) = \sum_{k\in\mathbb{Z}} h(k)y(n-k). \tag{2.2.33}$$

Then the causal Wiener filter of x given y is the same as the causal Wiener filter of $\widehat{x}_{\mathrm{NC}}$ given y, denoted as $\widehat{x}_{\mathrm{C}}$. If y is white noise, then

$$\widehat{x}_{\mathrm{C}}(n) = \sum_{k=0}^{\infty} h(k)y(n-k). \tag{2.2.34}$$

Inspired by this theorem, we compute the optimal non-causal Wiener filter of x given w. It is given by

$$\begin{tikzcd} w \arrow[r] & \boxed{\frac{S_{XW}(\omega)}{S_W(\omega)}} \arrow[r] & \widehat{x}_{\mathrm{NC}} \end{tikzcd}$$

Recall that

$$S_{XW}(\omega) = S_{XY}(\omega)\left(\frac{1}{S_Y(\omega)}\right)^* = \frac{S_{XY}(\omega)}{S_Y^-(\omega)} \tag{2.2.35}$$

and that $S_W(\omega) = 1$. So the transfer function is actually

$$\begin{tikzcd} w \arrow[r] & \boxed{\frac{S_{XY}(\omega)}{S_Y^-(\omega)}} \arrow[r] & \widehat{x}_{\mathrm{NC}} \end{tikzcd}$$

The optimal causal Wiener filter of X given W is

$$\begin{tikzcd} w \arrow[r] & \boxed{\left[\frac{S_{XY}(\omega)}{S_Y^-(\omega)}\right]_+} \arrow[r] & \widehat{x}_{\mathrm{C}} \end{tikzcd}$$

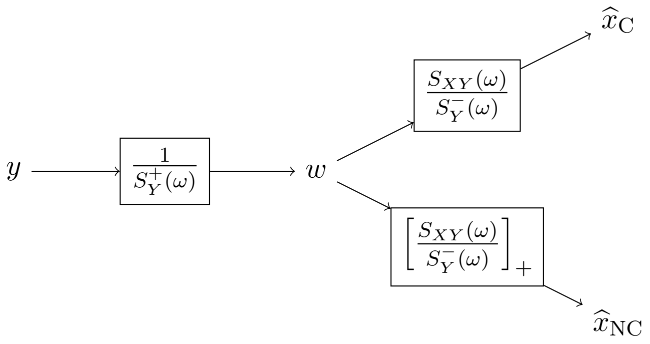
It gives an interesting interpretation of non-causal Wiener filter: it can be viewed as

$$\frac{1}{S_Y^+(\omega)}\,\frac{S_{XY}(\omega)}{S_Y^-(\omega)} \tag{2.2.36}$$

where we first perform a *causal* operation to transform Y into white noise W, and then use the *non-causal* Wiener filter to estimate X using W.

Similarly, for causal Wiener filter, we perform the same causal operation to transform Y into white noise W, then use the causal Wiener filter to estimate X using W. We use the observation that the causal Wiener filter with white noise as observations is easy to obtain from the non-causal Wiener filter with white noise as observations through the $[\cdot]_+$ operation.

The full diagram is the following:


y
$\frac{1}{S_Y^+(\omega)}$
w
$\frac{S_{XY}(\omega)}{S_Y^-(\omega)}$
$\widehat{x}_{\mathrm{C}}$
$\left[\frac{S_{XY}(\omega)}{S_Y^-(\omega)}\right]_+$
$\widehat{x}_{\mathrm{NC}}$

2.2.7 Prediction Problem

Suppose we have a zero-mean WSS process $(x(n))_{n\in\mathbb{Z}}$. The goal is to estimate $x(n)$ given the causal history $(x(n-1), x(n-2), \dots)$. We can cast this as a causal Wiener filter with $x(n) = x(n-1)$. The transfer function is

$$H(\omega) = \frac{1}{S_Y^+(\omega)}\left[\frac{S_{XY}(\omega)}{S_Y^-(\omega)}\right]_+ \tag{2.2.37}$$

$$\begin{aligned} R_{XY}(k) &= \mathbf{E}(x(n)y^*(n-k)) && (2.2.38)\\ &= \mathbf{E}(x(n)x^*(n-k-1)) && (2.2.39)\\ &= R_X(k+1) && (2.2.40)\\ R_Y(k) &= \mathbf{E}(y(n)y^*(n-k)) && (2.2.41)\\ &= \mathbf{E}(x(n-1)x^*(n-k-1)) && (2.2.42)\\ &= R_X(k). && (2.2.43) \end{aligned}$$

It implies that

$$\begin{aligned} S_{XY}(\omega) &= \mathrm{e}^{\mathrm{i}\omega}S_X(\omega) && (2.2.44)\\ S_Y(\omega) &= S_X(\omega). && (2.2.45) \end{aligned}$$

Plugging these into the formula,

$$\begin{aligned} H(\omega) &= \frac{1}{S_X^+(\omega)}\left[\frac{\mathrm{e}^{\mathrm{i}\omega}S_X(\omega)}{S_X^-(\omega)}\right]_+ && (2.2.46)\\ &= \frac{\left[\mathrm{e}^{\mathrm{i}\omega}S_X^+(\omega)\right]}{S_X^+(\omega)}. && (2.2.47) \end{aligned}$$

To simplify, write

$$S_X^+(\omega) = p_0 + \sum_{n=1}^{\infty} p_n \mathrm{e}^{-\mathrm{i}\omega n} \tag{2.2.48}$$

where $p_0 = \sqrt{r_e} > 0$. Hence

$$\left[\mathrm{e}^{\mathrm{i}\omega}S_X^+(\omega)\right]_+ = \left(S_X^+(\omega) - p_0\right)\mathrm{e}^{\mathrm{i}\omega}. \tag{2.2.49}$$

Plugging this back into the formula for $H(\omega)$, we obtain

$$H(\omega) = \left(1 - \frac{p_0}{S_X^+(\omega)}\right)\mathrm{e}^{\mathrm{i}\omega}. \tag{2.2.50}$$

We now have the transfer function for optimal causal filter from Y to X. However, the input to the filter is X instead of Y, so the output should be delayed by one unit, resulting in the optimal prediction filter

$$H(\omega) = 1 - \frac{p_0}{S_X^+(\omega)}. \tag{2.2.51}$$

Indeed, the filter diagram is

$$\begin{tikzcd} x_{-1} \arrow[r] & \boxed{1 - \frac{p_0}{S_X^+(\omega)}} \arrow[r] & x \end{tikzcd}$$

It can be shown that if we would like to predict X_n using all the information up to X_{n-r}, the optimal prediction filter is given by

$$H_r(\omega) = \frac{\sum_{k=r}^{\infty} p_k \mathrm{e}^{-\mathrm{i}\omega k}}{S_X^+(\omega)}. \tag{2.2.52}$$

with filter diagram

$$\begin{tikzcd} x_{-r} \arrow[r] & \boxed{\frac{\sum_{k=r}^{\infty} p_k \mathrm{e}^{-\mathrm{i}\omega k}}{S_X^+(\omega)}} \arrow[r] & x \end{tikzcd}$$

We finish our discussion on the prediction filter by computing the spectrum of the error process is $e(n) = x(n) - \widehat{x}(n)$ where $\widehat{x}(n)$ is x convolved with a filter that has transfer function H. The filter diagram is like

$$\begin{tikzcd} x \arrow[r] & \boxed{1 - H(\omega)} \arrow[r] & \widehat{x} \end{tikzcd}$$

It implies that

$$\begin{aligned} S_E(\omega) &= \left|1 - H(\omega)\right|^2 S_X(\omega) && (2.2.53)\\ &= \left|1 - \left(1 - \frac{p_0}{S_X^+(\omega)}\right)\right|^2 S_X(\omega) && (2.2.54)\\ &= \frac{p_0^2}{\left|S_X^+(\omega)\right|^2} S_X(\omega) && (2.2.55)\\ &= p_0^2 && (2.2.56)\\ &= r_e. && (2.2.57) \end{aligned}$$

This implies that the power spectral density of the error is r_e. In particular, this shows the power spectral density of the *innovation* – new information gained at $x(n)$ – is r_e. And E is a white-noise process with power r_e.

2.2.8 Generalized Prediction Problem

Consider the following generalization of the prediction problem. Suppose we have

$$y_n = s_n + v_n \tag{2.2.58}$$

where $\langle v_m, v_n \rangle = r\delta_{ij}$ and $\langle v_i, s_j \rangle = 0$. In other words, $S_V(z) = r$, and $S_{VS}(z) = 0$. Furthermore, we have $S_{SY}(z) = S_S(z)$, $S_Y(z) = S_S(z) + S_V(z) = S_S(z) = r$. Our goal is to estimate s_n using all the information of $(y_k)_{k \in \mathbb{Z}:\, k \le n}$. It is a causal Wiener filter problem, and the solution is given by

$$H(z) = \frac{1}{r_e L(z)} \left[\frac{S_{SY}(z)}{L^*(z^{-*})}\right]_+ \tag{2.2.59}$$

where $S_Y(z) = L(z) r_e L^*(z^{-*})$ is the canonical spectral factorization of Y. We can simplify it by

$$\begin{aligned} H(z) &= \frac{1}{r_e L(z)} \left[\frac{S_{SY}(z)}{L^*(z^{-*})}\right]_+ && (2.2.60)\\ &= \frac{1}{r_e L(z)} \left[\frac{S_Y(z) - r}{L^*(z^{-*})}\right]_+ && (2.2.61) \end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{r_e L(z)} \left[L(z) r_e - \frac{r}{L^*(z^{-*})} \right]_+ && (2.2.62) \\
&= \frac{1}{r_e L(z)} \left(r_e \left[L(z) \right]_+ - r \left[\frac{1}{L^*(z^{-*})} \right] \right) && (2.2.63) \\
&= \frac{1}{r_e L(z)} \left(r_e L(z) - r \left[\frac{1}{L^*(z^{-*})} \right] \right) && (L \text{ is causal.}) \\
&= \frac{1}{r_e L(z)} \left(r_e L(z) - r \right) && \left(\tfrac{1}{L^*(z^{-*})} \text{ is anti-causal, and } L(\infty) = 1.\right) \\
&= 1 - \frac{r}{r_e} L^{-1}(z). && (2.2.64)
\end{aligned}
$$

One might wonder about the filter that yields the optimal estimate of s_n given the history of y_m's. Noting that the optimal linear estimator of s_n given this history is the same as the optimal linear estimator of y_n given this history, so this problem is equivalent to the canonical prediction problem we just studied, which has filter

$$H(z) = 1 - L^{-1}(z). \tag{2.2.65}$$

Here the filter diagram is

$$y \longrightarrow \boxed{1 - \tfrac{r}{r_e} L^{-1}(\omega)} \longrightarrow \widehat{s}$$

2.3 Vector Case for Wiener Filters

We first present solutions to the non-causal and causal Wiener filtering problems in the vector case instead of the WSS process case, and discuss the similarities of this setting to the WSS setting. Then, we talk about the prediction problem in WSS processes and present a solution using spectral factorization. In this section, all the random variables are generally complex-valued.

We want to estimate a vector $X = \begin{bmatrix} X_1 \\ \vdots \\ X_N \end{bmatrix}$ and the observation we have is $Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_N \end{bmatrix}$. We have constrained that X and Y have equal length, since we want them to have shared time indices so that we can formulate a *causal* filter problem.

2.3.1 Non-Causal Wiener Filter

The natural question of non-causal Wiener filter would be to use the whole vector Y to estimate every entry of X. In other words, we would like to use estimator

$$\widehat{X}(Y) = K_s Y \in \mathbb{C}^N. \tag{2.3.1}$$

It follows from the orthogonality principle that

$$\left\langle X - \widehat{X}(Y), Y \right\rangle = \langle X - K_s Y, Y \rangle = 0 \implies \langle (X - K_s Y)_i, Y_\ell \rangle = 0 \quad \text{for all } i, \ell \in [N]. \tag{2.3.2}$$

By algebra, we get

$$\langle X_i, Y_\ell \rangle = \langle (K_s Y)_i, Y_\ell \rangle = \left\langle \sum_{j=1}^{N} K_{s,ij} Y_j, Y_\ell \right\rangle \quad \text{for all } i, \ell \in [N]. \tag{2.3.3}$$

Writing as a summation, we get

$$\langle X_i, Y_\ell \rangle = \sum_{j=1}^{N} K_{s,ij} \langle Y_j, Y_\ell \rangle \quad \text{for all } i, \ell \in [N]. \tag{2.3.4}$$

Writing in terms of the correlation matrices,

$$(R_{XY})_{i\ell} = \sum_{j=1}^{N} K_{s,ij} \left(R_{YY}\right)_{j\ell} = \left(K_s R_{YY}\right)_{i\ell} \tag{2.3.5}$$

which gives us

$$R_{XY} = K_s R_Y \implies K_s = R_{XY} R_Y^{-1} \tag{2.3.6}$$

which is exactly the best linear estimator. Note that we did not assume X, Y are zero-mean.

2.3.2 Causal Wiener Filter

To estimate X, we would like to use

$$\widehat{X}_i = \sum_{j=1}^{i} K_{f,ij} Y_j. \tag{2.3.7}$$

By orthogonality principle,

$$\left\langle X_i - \sum_{j=1}^{i} K_{f,ij} Y_j, Y_\ell \right\rangle = 0 \quad \text{for any } i \in [N] \text{ and } \ell \in [i]. \tag{2.3.8}$$

Expanding,

$$\langle X_i, Y_\ell \rangle = \left\langle \sum_{j=1}^{i} K_{f,ij} Y_j, Y_\ell \right\rangle = \sum_{j=1}^{i} K_{f,ij} \langle Y_j, Y_\ell \rangle \quad \text{for any } i \in [N] \text{ and } \ell \in [i]. \tag{2.3.9}$$

Using correlation matrices,

$$(R_{XY})_{i\ell} = \sum_{j=1}^{i} K_{f,ij} \left(R_Y\right)_{j\ell}. \tag{2.3.10}$$

Now define an *lower-triangular operator*,

$$\left([H]_L\right)_{ij} = \begin{cases} H_{ij} & 1 \le j \le i \\ 0 & \text{o.w.} \end{cases}. \tag{2.3.11}$$

With this notation, our earlier equation turns into the matrix equation

$$\left[R_{XY} - K_f R_Y\right]_L = 0. \tag{2.3.12}$$

Theorem 2.3.1 (LDL Decomposition). If H is positive definite, then there exists a unique lower-triangular+diagonal+ triangular factorization of H:

$$H = LDL^*, \tag{2.3.13}$$

where L is lower triangular with unit diagonal entries, and D is diagonal with positive entries. Both L and D are invertible.

We emphasize that the LDL decomposition solution to the causal filter problem here is not only conceptually important but also numerically efficient.

Now we continue to present the solution of the causal Wiener filter problem. Writing the Wiener-Hopf equation without the $[\cdot]_L$ operator:

$$R_{XY} - K_f R_Y = U^+ \tag{2.3.14}$$

where U^+ is some strictly upper triangular matrix.

Assuming $R_Y > 0$, we have the LDL decomposition as

$$R_Y = LDL^*. \tag{2.3.15}$$

Hence

$$R_{XY} - K_f LDL^* = U^+. \tag{2.3.16}$$

Right-multiplying by $L^{-*} := \left(L^{-1}\right)^* = (L^*)^{-1}$ gives

$$R_{XY}L^{-*}D^{-1} - K_f L = U^+ L^{-*}D^{-1}. \tag{2.3.17}$$

Because K_f is causal, it is lower-triangular. And L is lower-triangular. Hence $K_f L$ is also lower-triangular. Since L is upper-triangular, L^* is upper-triangular, so its inverse L^{-*} is upper-triangular as well. Since D is diagonal and U^+ is strictly upper-triangular, $U^+L^{-*}D^{-1}$ is also strictly upper triangular. If we apply the $[\cdot]_L$ operator to the above equation, we get

$$\left[R_{XY}L^{-*}D^{-1}\right]_L - K_f L = 0. \tag{2.3.18}$$

Hence

$$K_f = \left[R_{XY}L^{-*}D^{-1}\right]_L L^{-1}. \tag{2.3.19}$$

How is it related to the non-causal Wiener filter? In the non-causal case, we have

$$\begin{aligned} K_s &= R_{XY}R_Y^{-1} && (2.3.20)\\ &= R_{XY}L^{-*}D^{-1}L^{-1} && (2.3.21)\\ &= \left[R_{XY}L^{-*}D^{-1}\right]_L L^{-1} + \left[R_{XY}L^{-*}D^{-1}\right]_{\text{strict upper triangular part}} L^{-1} && (2.3.22)\\ &= K_f + \left[R_{XY}L^{-*}D^{-1}\right]_{\text{strict upper triangular part}} L^{-1}. && (2.3.23) \end{aligned}$$

The interested readers must have observed that we have made a strong assumption here: $R_Y > 0$. Indeed, even for positive semidefinite matrices the LDL decomposition also exists (may not be unique), but it cannot be generally used to construct casual filter: in those cases L may not even have a lower-triangular pseudoinverse.

The way to interpret this is exactly the Bode-Shannon interpretation. The operator L^{-1} does the whitening. If we want the causal insight, we use the causal operator $\left[R_{XY}L^{-*}D^{-1}\right]_L$. If we want the non-causal insight, we can add the non-causal part $\left[R_{XY}L^{-*}D^{-1}\right]_{\text{strict upper triangular part}}$ after whitening.

2.4 Wold Decomposition

Let P_{n-1} be the projection operator onto $\operatorname{Span}\Big(1, (x_m)_{m\le n-1}\Big)$. If

$$\sigma^2 := \mathbf{E}\Big(|x_n - P_{n-1}x_n|^2\Big) = r_e = \rho_0^2 \tag{2.4.1}$$

is 0, then the spectral factorization may not exist. Note that this is invariant of n since the process is WSS.

Theorem 2.4.1. If $\sigma^2 > 0$ then the WSS process can be written as

$$x_n = \sum_{j=0}^{\infty} c_j z_{n-j} + v_n \tag{2.4.2}$$

with the following properties:

1. $c_0 = 1$, $\sum_{j=0}^{\infty} |c_j|^2 < \infty$;
2. z_n is white noise with variance σ^2;
3. z_n is a linear function of $(x_t)_{t\le n}$;
4. Z and V are uncorrelated.
5. v_n is deterministic.

2.5 Kolmogorov Cepstral Method

Theorem 2.5.1 (Kolmogorov Cepstral Method). Suppose X is a WSS purely non-deterministic process. Let $S_X(z)$ denote the z-spectrum of X and assume that $\log(S_X(z))$ is analytic in an annulus that includes the unit circle, so that it can be expanded in a Laurent series

$$\log(S_X(z)) = \sum_{n\in\mathbb{Z}} \gamma_n z^{-n}. \tag{2.5.1}$$

The canonical spectral factorization of $S_X(z) = L(z) r_e L^*(z^{-*})$ is given by

$$L(z) = \exp\left(\sum_{n\in\mathbb{N}} \gamma_n z^{-n}\right) \tag{2.5.2}$$

$$r_e = \exp(\gamma_0). \tag{2.5.3}$$

Indeed, $L(z)$ is causal. And since $S_X(\omega) \geq 0$ and $\gamma_j = \gamma_{-j}^*$, the identity is easily verified

$$L(z) r_e L^*\left(z^{-*}\right) = \exp\left(\sum_{n\in\mathbb{N}} \gamma_n z^{-n}\right) \exp(\gamma_0) \exp\left(\sum_{n\in\mathbb{N}} \gamma_n^* z^{n}\right) \tag{2.5.4}$$

$$= \exp\left(\sum_{n\in\mathbb{N}} \gamma_n z^{-n}\right) \exp(\gamma_0) \exp\left(\sum_{n=-\infty}^{1} \gamma_{-n}^* z^{-n}\right) \tag{2.5.5}$$

$$= \exp\left(\sum_{n\in\mathbb{N}} \gamma_n z^{-n}\right) \exp(\gamma_0) \exp\left(\sum_{n=-\infty}^{1} \gamma_n z^{-n}\right) \tag{2.5.6}$$

$$= \exp\left(\sum_{n\in\mathbb{N}} \gamma_n z^{-n} + \gamma_0 + \sum_{n=-\infty}^{1} \gamma_n z^{-n}\right) \tag{2.5.7}$$

$$= \exp\left(\sum_{n\in\mathbb{Z}} \gamma_n z^{-n}\right). \tag{2.5.8}$$

This yields the **Kolmogorov-Szego formula**:

$$r_e = \exp(\gamma_0) = \exp\left(\frac{1}{2\pi}\int_{-\pi}^{\pi} \log\left(S_X\left(\mathrm{e}^{\mathrm{i}\omega}\right)\right) \mathrm{d}\omega\right). \tag{2.5.9}$$

This implies that if $S_X(\omega)$ takes the value 0 on a set of non-zero Lebesgue measure in $[-\pi, \pi)$, $r_e = 0$ and the spectral factorization doesn't exist. This means that there is a perfect estimator.

Example 2.5.2. Suppose

$$S_X(\omega) = \sum_{n\in\mathbb{N}} \alpha_n \delta(\omega - \omega_i). \tag{2.5.10}$$

Then using the filter

$$H(\omega) = 1 - \prod_{n\in\mathbb{N}} \left(1 - \mathrm{e}^{\mathrm{i}(\omega_n - \omega)}\right). \tag{2.5.11}$$

The power-spectral density of the prediction error $x_n - (h * x)_n$ is

$$S_X(\omega)\left|1 - H(\omega)\right|^2 = \left|\prod_{n\in\mathbb{N}} \left(1 - \mathrm{e}^{\omega_n - \omega}\right)\right|^2 \left(\sum_{n\in\mathbb{N}} \alpha_n \delta(\omega - \omega_n)\right). \tag{2.5.12}$$

At each ω, either $\omega = \omega_n$ for some n (in which case the first term is 0) or $\omega \neq \omega_n$ for any n (in which case the second term is 0).

3 State Space Models

3.1 State Estimation in Hidden Markov Processes

3.1.1 Markov Triplet

We begin by introducing the Markov triplet, which is useful in the design and analysis of algorithms for state estimation in Hidden Markov Processes.

Let X, Y, Z denote discrete random variables. We say that (X, Y, Z) form a Markov triplet (which we denote as $X - Y - Z$) given the following relationships:

$$\begin{aligned} X - Y - Z &\iff p(x, z \mid y) = p(x \mid y)p(z \mid y) && (3.1.1)\\ &\iff p(z \mid x, y) = p(z \mid y) && (3.1.2)\\ &\iff p(x \mid y, z) = p(x \mid y). && (3.1.3) \end{aligned}$$

If (x, y, z) form a Markov triplet, their joint distribution enjoys the important property that it can be factored into the product of two functions ϕ_1 and ϕ_2.

Let us prove that these are equivalent. We show that the first and second factorizations are equivalent, as an example. We compute

$$\begin{aligned} p(z \mid x, y) &= \frac{p(x, y, z)}{p(x, y)} = \frac{p(y)p(x, z \mid y)}{p(x, y)} = \frac{p(y)p(x \mid y)p(z \mid y)}{p(x, y)} = \frac{p(x, y)p(z \mid y)}{p(x, y)} && (3.1.4)\\ &= p(z \mid y). && (3.1.5) \end{aligned}$$

The rest are left as an exercise.

Lemma 3.1.1. *$X - Y - Z$ if and only if there are functions ϕ_1, ϕ_2 such that $p(x, y, z) = \phi_1(x, y)\phi_2(y, z)$.*

3.1.2 Hidden Markov Model

We use the notation x_m^n (with $m \leq n$), which stands for the sequence $(x_m, x_{m+1}, \dots, x_n)$. For example, $x_1^n = (x_1, x_2, \dots, x_n)$ is the first n components and $x_t^n = (x_t, x_{t+1}, \dots, x_n)$ is the last $n - t + 1$ components of the sequence $(x_1, x_2, \dots)$. We often shorten $x^n = x_1^n$. For completeness, if $n \leq 0$ then x^n is defined to be the empty set.

Definition 3.1.2 (Markov Process). A **Markov process** $(X_n)_{n\in\mathbb{N}}$ is a stochastic process such that for all $n \geq 2$, $X_n - X_{n-1} - X^{n-2}$, i.e., $\left(X_n, X_{n-1}, X^{n-2}\right)$ is a Markov triplet.

In other words, a process is Markov if the density of a variable conditioned on previous variables depends only on the immediate preceding variable, as in

$$p\left(x_n \mid x^{n-1}\right) = p(x_n \mid x_{n-1}). \tag{3.1.6}$$

The property in Eq. (3.1.6) is called the **Markov property**. For example, the joint density $p(x^n)$ of the first n terms of the Markov property is given by

$$\begin{aligned} p(x^n) &= \prod_{t=1}^{n} p\left(x_t \mid x^{t-1}\right) && (3.1.7)\\ &= \prod_{t=1}^{n} p(x_t \mid x_{t-1}). && (3.1.8) \end{aligned}$$

Definition 3.1.3 (Hidden Markov Process). Let $(X_n)_{n\in\mathbb{N}}$ be a Markov process. $(Y_n)_{n\in\mathbb{N}}$ is a **Hidden Markov Process** if the conditional probability density $p(y^n \mid x^n)$ factors as

$$p(y^n \mid x^n) = \prod_{i=1}^{n} p(y_i \mid x_i). \tag{3.1.9}$$

Then

$$\begin{aligned}
p(x^n, y^n) &= p(x^n)p(y^n \mid x^n) && (3.1.10)\\
&= \left(\prod_{t=1}^{n} p(x_t \mid x^{t-1})\right)\left(\prod_{t=1}^{n} p(y_t \mid y^{t-1}, x^n)\right) && (3.1.11)\\
&= \left(\prod_{t=1}^{n} p(x_t \mid x_{t-1})\right)\left(\prod_{t=1}^{n} p(y_t \mid x_t)\right) && (3.1.12)\\
&= \prod_{t=1}^{n} p(x_t \mid x_{t-1})p(y_t \mid x_t) && (3.1.13)\\
&= \prod_{t=1}^{n} p(x_t \mid x_{t-1})p(y_t \mid x_t). && (3.1.14)
\end{aligned}$$

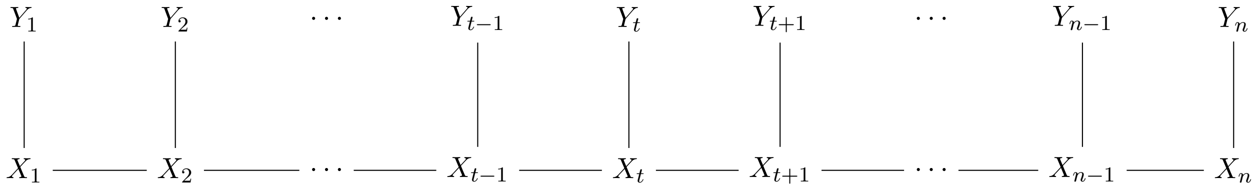
The Hidden Markov Process $(Y_n)_{n\in\mathbb{N}}$ is a noisy observation of an underlying Markov state process $(X_n)_{n\in\mathbb{N}}$ through a "memoryless" noisy channel. We assume that the state transition probabilities $p(x_t \mid x_{t-1})$ and the channel noise probabilities $p(y_i \mid x_i)$ are all well defined. The noisy channel is memoryless, because the observation of random variable Y_i depends only on the state variable X_i through the "channel" conditional density $p(y_i \mid x_i)$.

3.1.3 Undirected Graphical Model

Let us derive a graphical representation of the joint distribution as follows:

1. Create vertices on the graph representing each random variable $X_1, \ldots, X_n, Y_1, \ldots, Y_n$.

2. For each pairing of nodes, create an edge if there is some factor which contains both variables.



Lemma 3.1.4. For three disjoint sets of vertices S_1, S_2, S_3 such that any path from S_1 to S_3 passes through some vertices in S_2. Then $S_1 - S_2 - S_3$ forms a Markov triplet.

From this graph and the above lemma we get the following Markov triplets:

1. $(X^{t-1}, Y^t) - X_t - (X_{t+1}^n, Y_{t+1}^n)$;

2. $(X^{t-1}, Y^{t-1}) - X_t - (X_{t+1}^n, Y_t^n)$;

3. $X^{t-1} - (X_t, Y^t) - (X_{t+1}^n, Y_t^n)$;

4. $X^{t-1} - (X_t, Y^n) - X_{t+1}^n$.

3.1.4 Inference

Given that the noisy process $(Y_n)_{n\in\mathbb{N}}$ is observed, we would like to estimate $(X_n)_{n\in\mathbb{N}}$. One way to do this is to calculate the posterior distribution $p\big(x_t \mid y^t\big)$. We rewrite

$$\begin{aligned} p\big(x_t \mid y^t\big) &= \frac{p\big(x_t, y^t\big)}{p\big(y^t\big)} && (3.1.15)\\ &= \frac{\sum_{x^{t-1}} p\big(x^{t-1}, x_t, y^t\big)}{\sum_{x^t} p\big(x^t, y^t\big)} && (3.1.16)\\ &= \frac{\sum_{x^{t-1}} p\big(x^t, y^t\big)}{\sum_{x^t} p\big(x^t, y^t\big)}. && (3.1.17) \end{aligned}$$

This approach is infeasible since each sum above goes over an exponential number of terms. The factorized structure above suggests we can do much better.

Causal Inference with Forward Recursion

The clever factorization which is the source of the hidden Markov process graph is the basis of a recursive inference procedure termed *forward recursion*. Suppose at some index t we know $p\big(x_t \mid y^{t-1}\big)$ and the noisy channel conditional distribution $p(y_t \mid x_t)$. We can then write

$$\begin{aligned} p\big(x_t \mid y^t\big) &= \frac{p\big(x_t, y_t, y^{t-1}\big)}{\sum_{x_t'} p\big(x_t', y_t, y^{t-1}\big)} && (3.1.18)\\ &= \frac{p\big(y_t, y^{t-1} \mid x_t\big)p\big(x_t\big)}{\sum_{x_t'} p\big(y_t, y^{t-1} \mid x_t'\big)p\big(x_t'\big)} && (3.1.19)\\ &= \frac{p(y_t \mid x_t)p\big(y^{t-1} \mid x_t\big)p(x_t)}{\sum_{x_t'} p(y_t \mid x_t')p\big(y^{t-1} \mid x_t'\big)p(x_t')} && (3.1.20)\\ &= \frac{p(y_t \mid x_t)p\big(x_t \mid y^{t-1}\big)p\big(y^{t-1}\big)}{\sum_{x_t'} p(y_t \mid x_t')p\big(x_t' \mid y^{t-1}\big)p\big(y^{t-1}\big)} && (3.1.21)\\ &= \frac{p(y_t \mid x_t)p\big(x_t \mid y^{t-1}\big)}{\sum_{x_t'} p(y_t \mid x_t')p\big(x_t' \mid y^{t-1}\big)}. && (3.1.22) \end{aligned}$$

In the third step we made use of the independence assumption $Y^{t-1} - X_t - Y_t$. We make the following two definitions:

1. $\alpha_t(x_t) := p\big(x_t \mid y^t\big)$, the posterior of on the state x_t given past and present observations y^t,
2. $\beta_t(x_t) := p\big(x_t \mid y^{t-1}\big)$, the posterior of on the state x_t given past observations y^{t-1}.

In this case,

$$\alpha_t(x_t) = \frac{\beta_t(x_t)p(y_t \mid x_t)}{\sum_{x_t'} \beta_t(x_t')p(y_t \mid x_t')}. \qquad (3.1.23)$$

Here $\alpha_t(x_t)$ depends only on $\beta_t(x_t)$ and the channel probabilities $p(y_t \mid x_t)$. Eq. (3.1.23) is also known as the **measurement update**, because it calculates the posterior probability of the state at the current index t by indocrporating the measurement t. We can also compute the posterior $\beta_{t+1}(x_{t+1})$ as

$$\begin{aligned} \beta_{t+1}(x_{t+1}) &= p\big(x_{t+1} \mid y^t\big) && (3.1.24)\\ &= \sum_{x_t'} p\big(x_{t+1}, x_t' \mid y^t\big) && (3.1.25)\\ &= \sum_{x_t'} p\big(x_t' \mid y^t\big)p\big(x_{t+1} \mid x_t', y^t\big) && (3.1.26) \end{aligned}$$

$$= \sum_{x_t'} p(x_t \mid y^t) p(x_{t+1} \mid x_t). \tag{3.1.27}$$

where in the last step we used the relation $Y^t - X_t - X_{t+1}$. Hence

$$\beta_{t+1}(x_{t+1}) = \sum_{x_t'} \alpha_t(x_t') p(x_{t+1} \mid x_t'). \tag{3.1.28}$$

Eq. (3.1.28) is also known as the **time update**, because it updates the posterior probability by propagating the state forward one time index. Putting Eq. (3.1.23) and Eq. (3.1.28) together gives a recursive algorithm for estimating the posterior probabilities $\alpha_t(x_t)$ on the state x_t for each index t. Note that this algorithm is causal in the sense that $\alpha_t(x_t)$ depends only on the posterior and transition probabilities at the previous indices s with $1 \leq s \leq t$.

Putting all together, this is the forward recursion algorithm:

Algorithm 3.1.5 (Forward recursion algorithm for Hidden Markov Models.). **Initialize:** $\beta_1(x_1) = p(x_1)$ (p

```
for $t = 1$ to $n$ do
    $\alpha_t(x_t) = \frac{\beta_t(x_t)p(y_t|x_t)}{\sum_{x_t'} \beta_t(x_t)p(y_t|x_t)}$ (measurement update).
    $\beta_{t+1}(x_{t+1}) = \sum_{x_t'} \alpha_t(x_t)p(x_{t+1} \mid x_t)$ (time update).
```

The idea is that the measurement update reweights β based on the values of the conditional distribution $p(y \mid x)$, i.e., elaborates our certainty on x_t.

We can abstract our understanding of the measurement and time update by defining functions F and G such that

$$\alpha_t = F(\beta_t, p(y_t \mid x_t), y_t), \quad \beta_{t+1} = G(\alpha_t, p(x_{t+1} \mid x_t)). \tag{3.1.29}$$

In other words, F does the measurement update and the time update.

Inference for Controlled Markov Models

Suppose we allow some action A_t depending on Y^t, and the joint distribution of (X^n, Y^n, A^n) is

$$p(x^n, y^n, a^n) = \prod_{i=1}^{n} p(x_i \mid x_{i-1}, a_{i-1}) p(a_i \mid y^i) p(y_i \mid x_i), \tag{3.1.30}$$

where we assume $x_0 = 0$, $a_0 = 0$. We observe the sequence Y^n and A^n. Then we have the new updates

$$\alpha_t(x_t) = \frac{\beta_t(x_t) p(y_t \mid x_t)}{\sum_{x_t'} \beta_t(x_t') p(y_t \mid x_t')} \tag{3.1.31}$$

$$\beta_{t+1}(x_{t+1}) = \sum_{x_t'} \alpha_t(x_t') p(x_{t+1} \mid x_t', a_t(y^t)). \tag{3.1.32}$$

Non-Causal Inference with Backward Recursion

The causal forward recursion algorithm makes sense in applications where we are only allowed to use the state measurements up until the current index. In other applications, where causality is not a requirement (such as in image processing), we can also incorporate future measurements. In other words, we can calculate $p(x_t \mid y^n)$, $t \leq n$, instead of just $p(x_t \mid y^t)$. One difficulty is the initialization. We write

$$p(x_t \mid y^n) = \sum_{x_{t+1}} p(x_t, x_{t+1} \mid y^n) \tag{3.1.33}$$

$$= \sum_{x_{t+1}} p(x_{t+1} \mid y^n) p(x_t \mid x_{t+1}, y^t, y_{t+1}^n) \tag{3.1.34}$$

$$= \sum_{x_{t+1}} p(x_{t+1} \mid y^n) p\big(x_t \mid x_{t+1}, y^t\big) \tag{3.1.35}$$

$$= \sum_{x_{t+1}} p(x_{t+1} \mid y^n) \frac{p\big(x_t \mid y^t\big) p\big(x_{t+1} \mid x_t, y^t\big)}{p\big(x_{t+1} \mid y^t\big)} \tag{3.1.36}$$

$$= \sum_{x_{t+1}} p(x_{t+1} \mid y^n) \frac{p\big(x_t \mid y^t\big) p(x_{t+1} \mid x_t)}{p\big(x_{t+1} \mid y^t\big)} \tag{3.1.37}$$

$$\tag{3.1.38}$$

where in the third step we used the independence property $X_t - \big(X_{t+1}, Y^t\big) - \big(X_{t+2}^n, Y_{t+1}^n\big)$, and we can neglect the conditional dependence on y^t in the last term $p\big(x_{t+1} \mid x_t, y^t\big)$ because of the independence property $Y^t - X_t - X_{t+1}$. Thus if we make the additional definition

3. $\gamma_t(x_t) := p(x_t \mid y^n)$, the posterior on the state x_t given all observations y^n.

Then substituting,

$$\gamma_t(x_t) = \sum_{x_{t+1}} \gamma_{t+1}(x_{t+1}) \frac{\alpha_t(x_t) p(x_{t+1} \mid x_t)}{\beta_{t+1}(x_{t+1})}. \tag{3.1.39}$$

We loop from n to 1, backwards.

This is the backwards recursion algorithm:

Algorithm 3.1.6 (Backwards recursion algorithm for Hidden Markov Models.). **Initialize:** $\gamma_n(x_n) = p(x_n$

```
for $t = n-1$ to 1 do
    $\gamma_t(x_t) = \sum_{x_{t+1}} \gamma_{t+1}(x_{t+1}) \frac{\alpha_t(x_t) p(x_{t+1} | x_t)}{\beta_{t+1}(x_{t+1})}$
```

3.1.5 Viterbi Algorithm

The most likely sequence of hidden states $\widehat{x}^n$ is defined as

$$\widehat{x}^n := \arg\max_{x^n} p(x^n \mid y^n). \tag{3.1.40}$$

The Viterbi algorithm computes $\widehat{x}^n$ efficiently using the idea of dynamic programming. It defines the value function

$$V_t(x_t) := \max_{x^{t-1}} \big(x^t, y^t\big) \tag{3.1.41}$$

and iteratively computes it. Clearly

$$V_1(x_1) = p(x_1) p(y_1 \mid x_1) \tag{3.1.42}$$

is known. We have

$$\begin{aligned} p\big(x^t, y^t\big) &= p\big(x^{t-1}, y^{t-1}\big) p\big(x_t, y_t \mid x^{t-1}, y^{t-1}\big) && (3.1.43)\\ &= p\big(x^{t-1}, y^{t-1}\big) p\big(x_t \mid x^{t-1}, y^{t-1}\big) p\big(y_t \mid x_t, x^{t-1}, y^{t-1}\big) && (3.1.44)\\ &= p\big(x^{t-1}, y^{t-1}\big) p(x_t \mid x_{t-1}) p(y_t \mid x_t), && (3.1.45) \end{aligned}$$

where in the last step we used conditional independence relations to simplify the latter two terms. Hence,

$$\begin{aligned} V_t(x_t) &= \max_{x^{t-1}} \big(x^t, y^t\big) && (3.1.46)\\ &= \max_{x^{t-1}} p\big(x^{t-1}, y^{t-1}\big) p(x_t \mid x_{t-1}) p(y_t \mid x_t) && (3.1.47)\\ &= p(y_t \mid x_t) \max_{x^{t-1}} \left[p\big(x^{t-1}, y^{t-1}\big) p(x_t \mid x_{t-1})\right] && (3.1.48)\\ &= p(y_t \mid x_t) \max_{x_{t-1}} \left[p(x_t \mid x_{t-1}) \max_{x^{t-2}} \left[p\big(x^{t-1}, y^{t-1}\big)\right]\right] && (3.1.49) \end{aligned}$$

$$= p(y_t \mid x_t) \max_{x_{t-1}} \left[p(x_t \mid x_{t-1}) V_{t-1}(x_{t-1}) \right]. \tag{3.1.50}$$

Now to trace back the optimum achieving states, we know that

$$\widehat{x}_n = \arg\max_{x_n} V_n(x_n). \tag{3.1.51}$$

It follows from the equation

$$V_t(x_t) = p(y_t \mid x_t) \max_{x_{t-1}} \left[p(x_t \mid x_{t-1}) V_{t-1}(x_{t-1}) \right] \tag{3.1.52}$$

that we can compute $\widehat{x}_{t-1}$ based on $\widehat{x}_t$ using

$$\widehat{x}_{t-1} = \arg\max_{x_{t-1}} \left[p(\widehat{x}_t \mid x_{t-1}) V_{t-1}(x_{t-1}) \right]. \tag{3.1.53}$$

A few remarks are in order. To avoid underflow in numerical computations, usually one computes the logarithmic of all the probabilities and transform the product into sums. If the states x_t take values in a finite set with cardinality K, then the space complexity of the Viterbi algorithm is $O(nK)$ since we need to store n value functions and each one is a vector of dimension K. Its time complexity is $O(nK^2)$ since during each iteration we need to sweep over all the entries of a specific value function, and to compute each entry of a specific value function we need to compute the max operator, which again requires $O(K)$ operations.

Algorithm 3.1.7 (Viterbi Algorithm for Finding Most Likely State Trajectory). **Initialize:** $V_1(x_1) \leftarrow p(x$

```
Initialize: $\widehat{x}^n \leftarrow \emptyset$.
for $t = 2$ to $n$ do
    $V_t(x_t) \leftarrow p(y_t \mid x_t) \max_{x_{t-1}} [p(x_t \mid x_{t-1}) V_{t-1}(x_{t-1})]$.
$\widehat{x}_n \leftarrow \arg\max_{x_n} V_n(x_n)$.
for $t = n$ to $2$ do
    $\widehat{x}_{t-1} = \arg\max_{x_{t-1}} [p(\widehat{x}_t \mid x_{t-1}) V_{t-1}(x_{t-1})]$
```

3.2 Kalman Filter via Hidden Markov Process Analysis

3.2.1 Model

Suppose we have the model

$$X_{t+1} = A_t X_t + W_t \tag{3.2.1}$$

$$Y_t = H_t X_t + N_t \quad \text{for } t \in \mathbb{N}. \tag{3.2.2}$$

Here $X_1 \sim \mathcal{N}(0, \Pi_1)$, $W_t \sim \mathcal{N}(0, \Sigma_{W_t})$, and $N_t \sim \mathcal{N}(0, \Sigma_{N_t})$. Suppose the W_t are mutually independent, the N_t are mutually independent, the W_t are mutually independent of the N_s, and X_1 is mutually independent of everything else. However A_t, H_t, Σ_{W_t}, and Σ_{N_t} are deterministic and known.

Here

1. $(X_t)_{t \in \mathbb{N}}$ is a Markov process.

2. $(Y_t)_{t \in \mathbb{N}}$ is related to $(X_t)_{t \in \mathbb{N}}$ through a memoryless channel.

The collection $(X_t, Y_t)_{t \in \mathbb{N}}$ is a Hidden Markov Process!

The (forward recursion) Kalman filter finds the linear minimum mean square error estimate of X_t given Y^t by applying the forward recursion algorithm tailored to the jointly Gaussian random vectors. This approach is justified as the process is an instance of the Hidden Markov Process with jointly Gaussian random vectors.

Due to Gaussianity, we can characterize the posterior probabilities using the first and second moments only.

3.2.2 Relevant Gaussian Distribution Properties

We will need to review some facts about the Gaussian distribution. Recall that if

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \begin{bmatrix} \Sigma_X & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_Y \end{bmatrix}\right) \tag{3.2.3}$$

then

$$X \mid Y \sim \mathcal{N}\left(\mu_X + \Sigma_{XY}\Sigma_Y^{-1}\left(Y - \mu_Y\right), \Sigma_X - \Sigma_{XY}\Sigma_Y^{-1}\Sigma_{YX}\right) \tag{3.2.4}$$

Here the covariance $\Sigma_{X|Y} := \Sigma_X - \Sigma_{XY}\Sigma_Y^{-1}\Sigma_{YX}$ has a term $\Sigma_{\widehat{X}(Y)} := \Sigma_{XY}\Sigma_Y^{-1}\Sigma_{YX}$. So we can rewrite $\Sigma_{X|Y} = \Sigma_X - \Sigma_{\widehat{X}(Y)}$. Now suppose

$$Y = AX + N \tag{3.2.5}$$

where $X \sim \mathcal{N}(\mu_X, \Sigma_X)$ and $N \sim \mathcal{N}(0, \Sigma_N)$ are independent, then

$$\Sigma_{XY} = \Sigma_X A^*, \quad \Sigma_Y = A\Sigma_X A^* + \Sigma_N, \quad \mu_Y = A\mu_X. \tag{3.2.6}$$

3.2.3 Measurement Update

Using the identities derived above,

$$\mathbf{E}(X \mid Y) = \mu_X + \Sigma_X A^*\left(A\Sigma_X A^* + \Sigma_N\right)^{-1}\left(Y - A\mu_X\right) := F_\mu(\mu_X, \Sigma_X, A, \Sigma_N, Y) \tag{3.2.7}$$

$$\Sigma_{X|Y} = \Sigma_X - \Sigma_X A^*\left(A\Sigma_X A^* + \Sigma_N\right)^{-1} A\Sigma_X := F_\Sigma(\Sigma_X, A, \Sigma_N) \tag{3.2.8}$$

$$\mu_Y = A\mu_X := G_\mu(\mu_X, A) \tag{3.2.9}$$

$$\Sigma_Y = A\Sigma_X A^* + \Sigma_N := G_\Sigma(\Sigma_X, A, \Sigma_N). \tag{3.2.10}$$

Here $F_\mu, F_\Sigma, G_\mu, G_\Sigma$ do the measurement update and time update respectively. Returning to the special case of Kalman filter, we define $\widehat{X}_{i|j} = \mu_{X_i|Y^j}$ and $P_{i|j} = \Sigma_{X_i|Y^j}$. Then

$$\widehat{X}_{t|t} = F_\mu\left(\widehat{X}_{t|t-1}, P_{t|t-1}, H_t, \Sigma_{N_t}, Y_t\right) \tag{3.2.11}$$

$$P_{t|t} = F_\Sigma\left(P_{t|t-1}, H_t, \Sigma_{N_t}\right). \tag{3.2.12}$$

We define the **filtered Kalman gain**

$$K_{f,t} = P_{t|t-1}H_t^*\left(H_t P_{t|t-1} H_t^* + \Sigma_{N_t}\right)^{-1}. \tag{3.2.13}$$

Then we have

$$\widehat{X}_{t|t} = \widehat{X}_{t|t-1} + K_{f,t}\left(Y_t - H_t\widehat{X}_{t|t-1}\right) \tag{3.2.14}$$

$$P_{t|t} = P_{t|t-1} - K_{f,t}H_t P_{t|t-1}. \tag{3.2.15}$$

The signal $Y_t - H_t\widehat{X}_{t|t-1}$ is the error of the prediction of Y_t given all information at time $t-1$. It represents what is *new* about the observation; it is called the *innovation*.

3.2.4 Time Update

We can similarly write

$$\widehat{X}_{t+1|t} = G_\mu\left(\widehat{X}_{t|t}, A_{t+1}\right) \tag{3.2.16}$$

$$P_{t+1|t} = G_\Sigma\left(P_{t|t}, A_{t+1}, \Sigma_{W_t}\right). \tag{3.2.17}$$

We can use our G_μ, G_Σ to find

$$\widehat{X}_{t+1|t} = A_{t+1}\widehat{X}_{t|t} \tag{3.2.18}$$

$$P_{t+1|t} = A_{t+1}P_{t|t}A_{t+1}^* + \Sigma_{W_t}. \tag{3.2.19}$$

We have thus finished the initial derivation of Kalman filter.

3.2.5 Interpretation

Remark 3.2.1. The Kalman filter does not really need the Gaussian assumption; it produces the best linear estimator. The technique of assuming the model has Gaussian properties, saying "the best linear estimator is the conditional mean", and calculating the conditional expectation, is called the "Gaussian trick". This interpretation says that the Gaussian assumption is not really a restriction as much it is a help.

Remark 3.2.2. The limit as $n \to \infty$ in the scalar case is the Wiener filter. Asymptotic behavior of the Kalman filter is very difficult in general.

Remark 3.2.3. We can do updates in only one linear equation because our recursion conditions on all the information seen in the history. If we had to condition on some information given in the past that's not as up-to-date, we would require more update equations.

3.3 Innovations Process

In this lecture, we will discuss the innovations process from both geometric and algebraic views. We also discuss the application of the innovation method in solving least squares.

3.3.1 Geometric Approach

What is the innovation? Suppose we have a sequence of jointly distributed random vectors $Y_0, Y_1, \ldots, Y_n$. We define the **innovation** at time i to be

$$e_i = Y_i - \widehat{Y}_{i|i-1}. \tag{3.3.1}$$

Here $\widehat{Y}_{i|i-1}$ is the optimal linear estimator of Y_i given Y_0^{i-1}. We initialize $e_0 = Y_0$.

If $\mathcal{L}_i = \operatorname{Span}\big(Y_0^i\big)$, then by definition $\widehat{Y}_{i|i-1} = \operatorname{proj}_{Y_i}(\mathcal{L}_{i-1})$. That is, the optimal linear estimator of Y_i given the prior observations is equal to the projection of Y_i into the Hilbert space spanned by the prior observations.

The innovations process satisfies three fundamental properties:

1. $e_i \perp e_j$ for $i \neq j$, $0 \leq i, j \leq n$.

2. $e_i \in \mathcal{L}_i$.

3. $Y_i \in \mathcal{L}_i = \mathcal{L}\big(Y_0^i\big) = \mathcal{L}\big(e_0^i\big)$.

Lemma 3.3.1. Suppose e_0^n is a collection of mutually orthogonal random vectors. Then the optimal linear estimator of any random vector X distributed with e_0^n is given by

$$\widehat{X} = \sum_{j=0}^{N} \langle X, e_j \rangle \, \|e_j\|^{-2} \, e_j \tag{3.3.2}$$

$$= \sum_{j=0}^{N} R_{Xe_j} R_{e_j}^{-1} e_j. \tag{3.3.3}$$

This lemma is easy to prove from the linear algebraic fact that if e_0^n are orthogonal then

$$\operatorname*{proj}_{\operatorname{Span}(e_0^n)}(X) = \sum_{j=0}^{n} \operatorname*{proj}_{\operatorname{Span}(e_i)}(X). \tag{3.3.4}$$

3.3.2 Innovations Process to Solve Prediction Problem

Remember that in the prediction problem, we predict Y_t from Y^{t-1}, getting the estimator $\widehat{Y}_{t|t-1}$ using the following filter:

$$\begin{tikzcd}(Y_t)_{t\in\mathbb{Z}} \arrow[r] & \boxed{1-L^{-1}(z)} \arrow[r] & \left(\widehat{Y}_{t|t-1}\right)_{t\in\mathbb{Z}}\end{tikzcd}$$

where $S_Y(z) = L(z)r_eL^*(z^{-*})$.
From the whitening interpretation, we know

$$\begin{tikzcd}(Y_t)_{t\in\mathbb{Z}} \arrow[r] & \boxed{L^{-1}(z)} \arrow[r] & (e_t)_{t\in\mathbb{Z}}\end{tikzcd}$$

Here $S_e(z) = r_e$. Then we know that

$$e_i = Y_i - \widehat{Y}_{i|i-1} \implies \widehat{Y}_{i|i-1} = Y_i - e_i \tag{3.3.5}$$

Taking the z-transform on both sides, we obtain

$$\mathcal{Z}\left(\widehat{Y}_{i|i-1}\right) = Y(z) - E(z) = Y(z) - \frac{1}{L(z)}Y(z) = \left(1 - L^{-1}(z)\right)Y(z). \tag{3.3.6}$$

This confirms our earlier derivation.
Now let us consider the k-step prediction problem, we predict Y_{t+k} from Y^t. We know that

$$\mathcal{Z}(e) = E(z) = \frac{Y(z)}{L(z)} \implies Y(z) = L(z)E(z). \tag{3.3.7}$$

Since $L(z)$ is causal,

$$\begin{aligned}
Y_i &= \mathcal{Z}^{-1}(Y(z)) && (3.3.8)\\
&= \mathcal{Z}^{-1}(L(z)E(z)) && (3.3.9)\\
&= (\ell * e)_i && (3.3.10)\\
&= \sum_{j=-\infty}^{\infty} \ell_j e_{i-j} && (3.3.11)\\
&= \sum_{j=0}^{\infty} \ell_j e_{i-j} && (3.3.12)\\
&= \ell_0 e_i + \sum_{j=1}^{\infty} \ell_j e_{i-j}. && (3.3.13)
\end{aligned}$$

Since $L(\infty) = 1$, we know $\ell_0 = 1$, so that

$$\begin{aligned}
Y_i &= \ell_0 e_i + \sum_{j=1}^{\infty} \ell_j e_{i-j} && (3.3.14)\\
&= e_i + \sum_{j=1}^{\infty} \ell_j e_{i-j}. && (3.3.15)
\end{aligned}$$

Shifting indices forward by k,

$$Y_{i+k} = e_{i+\lambda} + \sum_{j=1}^{\infty} \ell_j e_{i+k-j}. \tag{3.3.16}$$

Now that we have Y_{i+k} represented as the sum of orthogonal e_i, we see that the prediction coefficients ℓ_j for information we don't have should be 0 – corresponding to no new information coming from that innovation. Then

$$\widehat{Y}_{i+k|i} = \sum_{j=k}^{\infty} \ell_j e_{i+k-j}. \tag{3.3.17}$$

We know that

$$Y_{i+k} - \widehat{Y}_{i+k|i} = e_{i+k} + \sum_{j=1}^{k-1} \ell_j e_{i+k-j}. \tag{3.3.18}$$

The filter for getting $Y_i \mapsto Y_{i+k} - \widehat{Y}_{i+k|i}$ is $H_1(z) = 1 + \sum_{j=1}^{k-1} \ell_j z^{-j}$. From this we see that the final filter will be the following:

$$H(z) = z^k \left(1 - \frac{1 + \sum_{j=1}^{k-1} \ell_j z^{-j}}{L(z)}\right). \tag{3.3.19}$$

and the error can be expressed in terms of the orthogonal decomposition as

$$\begin{aligned} \mathbf{E}\left(\left|Y_{i+k} - \widehat{Y}_{i+k|i}\right|^2\right) &= \mathbf{E}\left(\left|e_{i+k} + \sum_{j=1}^{k-1} \ell_j e_{i+k-j}\right|^2\right) && (3.3.20) \\ &= r_e \left(1 + \sum_{j=1}^{k-1} |\ell_j|^2\right). && (3.3.21) \end{aligned}$$

Innovations Process to Solve Generalized Prediction Problem

Suppose we have the state space model

$$Y_i = S_i + V_i \tag{3.3.22}$$

where $\langle V_i, V_j \rangle = r\delta_{ij}$, $\langle V_i, S_j \rangle = 0$. Then the optimal estimator of S_i using Y^i has transfer function

$$1 - \frac{r}{r_e} L^{-1}(z). \tag{3.3.23}$$

We have already derived this formula before. Now let us try it using innovations. Let us write

$$\begin{aligned} Y_i &= \widehat{Y}_{i|i} && (3.3.24) \\ &= \widehat{S}_{i|i} + \widehat{V}_{i|i} && (3.3.25) \end{aligned}$$

where the last decomposition happens because projection is a linear operator, i.e.,

$$\begin{aligned} \left\langle Y_i - \left(\widehat{S}_{i|i} + \widehat{V}_{i|i}\right), Y_j \right\rangle &= \left\langle \left(S_i - \widehat{S}_{i|i}\right) + \left(V_i - \widehat{V}_{i|i}\right), Y_j \right\rangle && (3.3.26) \\ &= \left\langle S_i - \widehat{S}_{i|i}, Y_j \right\rangle + \left\langle V_i - \widehat{V}_{i|i}, Y_j \right\rangle && (3.3.27) \\ &= 0 + 0 = 0. && (3.3.28) \end{aligned}$$

and the equality is true by the orthogonality principle. We know that

$$\begin{aligned} \|e_j\|^2 &= \langle e_j, e_j \rangle && (3.3.29) \\ &= r_e && (3.3.30) \end{aligned}$$

$$\begin{aligned} \widehat{Y}_{i|i} &= \sum_{j=-\infty}^{i} \langle V_i, e_j \rangle \|e_j\|^{-2} e_j && (3.3.31) \\ &= \sum_{j=-\infty}^{i} \underbrace{\langle V_i, e_j \rangle}_{=0 \text{ for } j<i} r_e^{-1} e_j && (3.3.32) \end{aligned}$$

$$
\begin{aligned}
&= \langle V_i, e_i \rangle r_e^{-1} e_i && (3.3.33)\\
&= \left\langle V_i, Y_i - \widehat{Y}_{i|i-1} \right\rangle r_e^{-1} e_i && (3.3.34)\\
&= \left(\langle V_i, Y_i \rangle - \underbrace{\left\langle V_i, \widehat{Y}_{i|i-1} \right\rangle}_{=0} \right) r_e^{-1} e_i && (3.3.35)\\
&= \langle V_i, Y_i \rangle r_e^{-1} e_i && (3.3.36)\\
&= \langle V_i, S_i + V_i \rangle r_e^{-1} e_i && (3.3.37)\\
&= \left(\underbrace{\langle V_i, S_i \rangle}_{=0} + \underbrace{\langle V_i, V_i \rangle}_{=r} \right) r_e^{-1} e_i && (3.3.38)\\
&= \frac{r}{r_e} e_i. && (3.3.39)
\end{aligned}
$$

Here $\langle V_i, e_j \rangle = 0$ for $j < i$ and $\left\langle V_i, \widehat{Y}_{i|i-1} \right\rangle = 0$ because V_i is a white noise process and is not correlated to Y_j for $j < i$. The error is

$$
\begin{aligned}
S_i - \widehat{S}_{i|i} &= S_i - \left(Y_i - \frac{r}{r_e} e_i \right) && (3.3.40)\\
&= \frac{r}{r_e} e_i - V_i && (3.3.41)\\
\left\| S_i - \widehat{S}_{i|i} \right\|^2 &= \frac{r^2}{r_e^2} \underbrace{\|e_i\|^2}_{=r_e} + \underbrace{\|V_i\|^2}_{=r} - 2\frac{r}{r_e} \underbrace{\langle e_i, V_i \rangle}_{=r} && (3.3.42)\\
&= \frac{r^2}{r_e} + r - \frac{2r^2}{r_e} && (3.3.43)\\
&= r\left(1 - \frac{r}{r_e}\right). && (3.3.44)
\end{aligned}
$$

The interpretation of this is that r is a hidden parameter, but r_e can be estimated by the observations. And we know $r \leq r_e$, so in this way we can bound r.

3.3.3 Algebraic Approach

We want to construct an innovation vector $\epsilon = \begin{bmatrix} e_0 \\ e_1 \\ \vdots \\ e_n \end{bmatrix}$ and have a vector $Y = \begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_n \end{bmatrix}$. We want to find A such that $AY = \epsilon$. Define $R_Y = \langle Y, Y \rangle$. Then $R_\epsilon = A R_Y A^*$.

We want to design A such that R_ϵ is diagonal (hence is white noise). If we impose this restriction then A is not unique. One can see this by parameter counting; A has $(n+1)^2$ parameters, but R_ϵ only has $\frac{(n+1)^2}{2}$ parameters because it is defined to be symmetric.

Now to uniquely solve it, we want $A = L^{-1}$ where L is lower triangular. This is to implement a causal version of A. We implement uniqueness by recognizing $e_i = Y_i - \widehat{Y}_{i|i-1}$, and the coefficient of Y_i is exactly 1, so the entries on the diagonal of L should be 1.

To solve it from here we write $R_Y = LDL^*$. Then

$$R_\epsilon = L^{-1} R_Y L^{-*} = L^{-1} \left(LDL^* \right) L^{-*} = L^{-1} LDL^* L^{-*} = D. \tag{3.3.45}$$

To show uniqueness, suppose we come up with some *other* L_1 which fits the two criterion we have (lower triangular and ones on the diagonal), such that $R_\epsilon = L_1^{-1} R_Y L_1^{-*}$. Then $R_Y = L_1 R_\epsilon L_1^*$ is another LDL decomposition of R_ϵ. This is a contradiction, assuming $R_Y > 0$.

3.4 General State Space Models

The goal of this section is to encapsulate a general problem (state space models) and solve it generally via our techniques.

3.4.1 Exponentially Correlated Process

Suppose $(Y_n)_{n\in\mathbb{Z}}$ is wide-sense stationary and $a \in (0,1)$ is such that

$$R_Y(k) = a^{|k|}. \tag{3.4.1}$$

Then

$$\begin{aligned}
S_Y(z) &= \sum_{k\in\mathbb{Z}} a^{|k|} z^{-k} && (3.4.2)\\
&= \sum_{k=-\infty}^{-1} a^{-k} z^{-k} + \sum_{k=0}^{\infty} a^k z^{-k} && (3.4.3)\\
&= \sum_{k=1}^{\infty} (az)^k + \sum_{k=0}^{\infty} \left(\frac{a}{z}\right)^k && (3.4.4)\\
&= \frac{az}{1-az} + \frac{1}{1-\frac{a}{z}} && (3.4.5)\\
&= \frac{1-a^2}{(1-az^{-1})(1-az)} && (3.4.6)\\
&= \frac{1}{1-az^{-1}} \cdot \left(1-a^2\right) \cdot \frac{1}{1-az}. && (3.4.7)
\end{aligned}$$

This is a canonical spectral factorization,

$$L(z) = \frac{1}{1-az^{-1}}, \quad r_e = 1-a^2. \tag{3.4.8}$$

The radius of convergence is $a < |z| < \frac{1}{a}$. We have

$$\begin{aligned}
L(z) &= \sum_{j=0}^{\infty} \left(az^{-1}\right)^j && (3.4.9)\\
&= \sum_{j=0}^{\infty} a^j z^{-j}, && (3.4.10)
\end{aligned}$$

which implies that the impulse response is $l_j = a^j$ for all $j \geq 0$. Since L is a modeling filter, we can represent the process Y as

$$Y_n = \sum_{j=0}^{\infty} a^j V_{n-j}, \tag{3.4.11}$$

where V is a white noise process with variance $1-a^2$.

Define a new process U such that $U_i = V_{i+1}$. The process U is also a white noise process with variance $1-a^2$, and

$$Y_i = \sum_{j=0}^{\infty} a^j U_{i-j-1}. \tag{3.4.12}$$

Then

$$Y_{i+1} - aY_i = \sum_{j=0}^{\infty} a^j U_{i-j} - \sum_{j=0}^{\infty} a^{j+1} U_{i-j-1} \tag{3.4.13}$$

$$\begin{aligned} &= \sum_{j=0}^{\infty} a^j U_{i-j} - \sum_{j=1}^{\infty} a^{j+1} U_{i-j} \quad (3.4.14) \\ &= U_i, \quad (3.4.15) \end{aligned}$$

which implies the recursive relation

$$Y_{i+1} = aY_i + U_i \quad i \in \mathbb{Z}. \tag{3.4.16}$$

Since $Y_i \in \mathrm{Span}(U_j : j < i)$ and U is white noise, we know

$$\langle U_i, Y_j \rangle = 0 \quad i \geq j. \tag{3.4.17}$$

The interpretation is that we are at a point Y_i at time i. Then we do a time update according to the signal and add some white noise and move to a new state Y_{i+1}.

To compute the innovations,

$$\begin{aligned} e_{i+1} &= Y_{i+1} - \widehat{Y}_{i+1|i} \quad (3.4.18) \\ &= Y_{i+1} - aY_i \quad (3.4.19) \\ &= U_i \quad (3.4.20) \\ &= V_{i+1}. \quad (3.4.21) \end{aligned}$$

3.4.2 Going Beyond Stationarity

Consider a state space model

$$Y_{i+1} = a_i Y_i + U_i, \quad i \geq 0. \tag{3.4.22}$$

We assume

$$\begin{aligned} \langle Y_0, Y_0 \rangle &= \Pi_0 \quad (3.4.23) \\ \langle U_i, U_j \rangle &= Q_i \delta_{ij}, \quad i, j \in \mathbb{N}_0 \quad (3.4.24) \\ \langle U_i, Y_0 \rangle &= 0, \quad i \in \mathbb{N}_0. \quad (3.4.25) \end{aligned}$$

Then the innovation representation

$$e_{i+1} = Y_{i+1} - a_i Y_i \tag{3.4.26}$$

still holds.

Note that we no longer have an assumption on $a_i \in (0, 1)$. Indeed, for $a_i > 1$, we generally cannot make this process stationary. We show it by contradiction: if $a_i = a$, $\langle Y_i, Y_i \rangle = \Pi$, and $\langle U_i, U_j \rangle = Q\delta_{ij}$, then the state space model implies

$$\Pi = a^2 \Pi + Q \tag{3.4.27}$$

which does not have a PSD solution for Π when $a > 1$ except in trivial cases.

3.4.3 Autoregressive Process

To demonstrate the generality of state space models, here we show that the autoregressive process can be written in the state-space model form with appropriately defined states.

Consider the AR(n) model

$$Y_{i+1} = \sum_{j=0}^{n-1} a_{j,i} Y_{i-j} + U_i, \quad i \geq 0 \tag{3.4.28}$$

where $\langle U_i, U_j \rangle = Q_j \delta_{ij}$ for all $i, j \geq 0$, $\langle U_i, Y_j \rangle = 0$ for all $j \leq i$.

Define

$$X_i = \begin{bmatrix} Y_i \\ Y_{i-1} \\ \vdots \\ Y_{i-(n-1)} \end{bmatrix}. \tag{3.4.29}$$

Suppose we have initial value X_0, i.e., we know $\{Y_0, Y_{-1}, \dots, Y_{-n+1}\}$. We can construct $\Pi_0 = \langle X_0, X_0 \rangle$. We can put it into our state space model by writing

$$X_{i+1} = \begin{bmatrix} Y_{i+1} \\ Y_i \\ \vdots \\ Y_{i-(n-2)} \end{bmatrix} \tag{3.4.30}$$

$$= \begin{bmatrix} a_{0,i} & a_{1,i} & \cdots & a_{n-1,i} \\ & & & 0 \\ & I & & \vdots \\ & & & 0 \end{bmatrix} \begin{bmatrix} Y_i \\ Y_{i-1} \\ \vdots \\ Y_{i-(n-1)} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} U_i. \tag{3.4.31}$$

This allows us to reduce the autoregressive process to so-called *standard form.*

3.4.4 Standard State-Space Model

The standard state space model is

$$X_{i+1} = F_i X_i + G_i U_i \tag{3.4.32}$$
$$Y_i = H_i X_i + V_i \tag{3.4.33}$$

with X_i the state and Y_i the observation. We assume

$$\left\langle \begin{bmatrix} X_0 \\ U_i \\ V_i \end{bmatrix}, \begin{bmatrix} X_0 \\ U_i \\ V_i \end{bmatrix} \right\rangle = \begin{bmatrix} \Pi_0 & 0 & 0 \\ 0 & Q_i \delta_{ij} & S_i \delta_{ij} \\ 0 & S_i^* \delta_{ij} & R_i \delta_{ij} \end{bmatrix}. \tag{3.4.34}$$

Remark 3.4.1. The V_i does not have a coefficient in the standard state-space model because most of the time we consider the observation noise to be "well-mixed", and not having good low-rank structure. If it does, we may introduce $\widehat{V_i} = J_i V_i$, and work out the results that way.

As in the previous autoregressive example, the state evolution noise does have low rank structure most of the time, so we choose to give it a coefficient G_i.

Proposition 3.4.2. Define

$$\Pi_i = \langle X_i, X_i \rangle \tag{3.4.35}$$

and define $\phi(i,j)$, $i \geq j$, by

$$\phi(i,j) = F_{i-1} F_{i-2} \cdots F_j, \quad i > j, \quad \phi(i,i) = I. \tag{3.4.36}$$

The following is true for the standard state-space model:

1. Orthogonality: if $i \geq j$ then
$$\langle U_i, X_j \rangle = 0, \quad \langle V_i, X_j \rangle = 0. \tag{3.4.37}$$

2. Orthogonality: if $i > j$ then
$$\langle U_i, Y_j \rangle = 0, \quad \langle V_i, Y_j \rangle = 0. \tag{3.4.38}$$

3. Non-orthogonality: if $i = j$ then
$$\langle U_i, Y_i \rangle = S_i, \quad \langle V_i, Y_i \rangle = R_i. \tag{3.4.39}$$

4. Time-evolution of state autocorrelation matrix:
$$\Pi_{i+1} = F_i \Pi_i F_i^* + G_i Q_i G_i^*. \tag{3.4.40}$$

5. Time-evolution of state cross-correlation matrix:

$$\langle X_i, X_j \rangle = \begin{cases} \phi(i,j)\Pi_j, & i \geq j \\ \Pi_i \Phi^*(j,i), & i \leq j \end{cases}.$$

6. Time-evolution of observation cross-correlation matrix:

$$\langle Y_i, Y_j \rangle = \begin{cases} H_i \phi(i, j+1) N_j, & i > j \\ R_i + H_i \Pi_i H_i^*, & i = j \\ N_i^* \phi(j, i+1)^* H_j^*, & i < j \end{cases} \quad \text{with } N_i = F_i \Pi_i H_i^* + G_i S_i.$$

3.5 Wide-Sense Markovity

Given a Markov triplet $X - Y - Z$, we know that the Markov property stipulates that

$$p(X \mid Y, Z) = p(X \mid Y). \tag{3.5.1}$$

Definition 3.5.1 (Wide-Sense Markovity). We say that (X, Y, Z) is a **wide-sense Markov triplet** if and only if

$$\widehat{X}(Y) = \widehat{X}(Y, Z), \tag{3.5.2}$$

where $\widehat{X}(M)$ is the optimal linear estimator of X given the information in M.

A **stochastic process** $(Y_i)_{i \in \mathbb{N}_0}$ is a **wide-sense Markov process** if and only if for any i, $\left(Y_{i+1}, Y_i, Y^{i-1}\right)$ is a wide-sense Markov triplet.

A corollary is that Y is a wide-sense Markov process if and only if

$$\widehat{Y}_{i|i-1} = \widehat{Y}_{i|Y_{i-1}}. \tag{3.5.3}$$

Theorem 3.5.2. A process X is wide-sense Markov if and only if

1. it has state space representation $X_{i+1} = F_i X_i + G_i U_i$, $i \geq 0$, and
2. It has the following second-order statistic for $i, j \geq 0$:

$$\left\langle \begin{bmatrix} U_i \\ X_0 \end{bmatrix}, \begin{bmatrix} U_j \\ X_0 \end{bmatrix} \right\rangle = \begin{bmatrix} Q_i \delta_{ij} & 0 \\ 0 & \Pi_0 \end{bmatrix}. \tag{3.5.4}$$

Remark 3.5.3. The interpretation is that U_i, U_j are not correlated with X_0, meaning that noise forward in time is not correlated with X_0, if and only if a state space process is wide-sense Markov.

Proof. We need to show that

$$X_{i+1} = F_i X_i + G_i U_i \quad \text{and} \quad \left\langle \begin{bmatrix} U_i \\ X_0 \end{bmatrix}, \begin{bmatrix} U_j \\ X_0 \end{bmatrix} \right\rangle = \begin{bmatrix} Q_i \delta_{ij} & 0 \\ 0 & \Pi_0 \end{bmatrix} \implies X \text{ is wide-sense Markov.} \tag{3.5.5}$$

By linearity of expectation, uncorrelatedness of U_i and X_0 and uncorrelatedness of U_i and U_j for $i \neq j$, we have

$$\langle U_i, X_j \rangle = 0. \tag{3.5.6}$$

Then

$$\widehat{X}_{i+1\ midi} = F_i \underbrace{\widehat{X}_{i|i}}_{=X_i} + G_i \underbrace{\widehat{U}_{i|i}}_{=0} \tag{3.5.7}$$

$$= F_i X_i \tag{3.5.8}$$

$$= F_i \widehat{X}_{i|X_i} \tag{3.5.9}$$

$$= \widehat{X}_{i+1|X_i}. \tag{3.5.10}$$

Thus $\widehat{X}_{i+1|i} = \widehat{X}_{i+1|X_i}$, so X is a wide-sense Markov process.

Now we need to show that

$$X \text{ is wide-sense Markov} \implies X_{i+1} = F_i X_i + G_i U_i \quad \text{and} \quad \left\langle \begin{bmatrix} U_i \\ X_0 \end{bmatrix}, \begin{bmatrix} U_j \\ X_0 \end{bmatrix} \right\rangle = \begin{bmatrix} Q_i \delta_{ij} & 0 \\ 0 & \Pi_0 \end{bmatrix}. \tag{3.5.11}$$

Given a wide-sense Markov process $(X_n)_{n \in \mathbb{N}_0}$, we define the following innovation sequence:

$$e_0 = X_0 \tag{3.5.12}$$

$$e_{i+1} = X_{i+1} - \widehat{X}_{i+1|i} \tag{3.5.13}$$

$$= X_{i+1} - \widehat{X}_{i+1|X_i} \tag{3.5.14}$$

$$= X_{i+1} - F_i X_i \tag{3.5.15}$$

where F_i is the matrix representation of the linear function that outputs the optimal linear estimator $\widehat{X}_{i+1|X_i}$. Define

$$U_i := e_{i+1}. \tag{3.5.16}$$

We can rewrite the above expression as

$$X_{i+1} = F_i X_i + U_i. \tag{3.5.17}$$

We need to verify the following claims.

1. $\langle U_i, U_j \rangle = 0$ for $i \neq j$. Indeed,

$$\langle U_i, U_j \rangle = \langle e_{i+1}, e_{j+1} \rangle = 0 \tag{3.5.18}$$

since innovations are orthogonal.

2. $\langle U_i, X_0 \rangle = 0$ for $i \in \mathbb{N}_0$. Indeed,

$$\langle U_i, X_0 \rangle = \langle U_i, e_0 \rangle = \langle e_{i+1}, e_0 \rangle = 0, \quad i \in \mathbb{N}_0. \tag{3.5.19}$$

3. There is a matrix Q_i such that $\langle U_i, U_i \rangle = Q_i$. We compute Q_i. Define $\Pi_i = \langle X_i, X_i \rangle$ the autocorrelation matrix of X_i. Then

$$\Pi_{i+1} = \langle X_{i+1}, X_{i+1} \rangle \tag{3.5.20}$$

$$= \langle F_i X_i + U_i, F_i X_i + U_i \rangle \tag{3.5.21}$$

$$= F_i \langle X_i, U_i \rangle + F_i \langle X_i, X_i \rangle F_i^* + \langle U_i, U_i \rangle + F_i^* \langle U_i, X_i \rangle \tag{3.5.22}$$

$$= F_i \langle X_i, X_i \rangle F_i^* + \langle U_i, U_i \rangle \tag{3.5.23}$$

$$= F_i \Pi_i F_i^* + \langle U_i, U_i \rangle \tag{3.5.24}$$

$$\implies \langle U_i, U_i \rangle = \Pi_{i+1} - F_i \Pi_i F_i^*. \tag{3.5.25}$$

This gives a state space representation in the desired form. □

3.6 Prediction Kalman Filter via Innovations Process

3.6.1 Model

Consider a system with state space representation

$$X_{i+1} = F_i X_i + G_i U_i, \quad i \in \mathbb{N}_0 \tag{3.6.1}$$

$$Y_i = H_i X_i + V_i \tag{3.6.2}$$

and joint covariance structure

$$\left\langle \begin{bmatrix} U_i \\ V_i \\ X_0 \end{bmatrix}, \begin{bmatrix} U_j \\ V_j \\ X_0 \end{bmatrix} \right\rangle = \begin{bmatrix} Q_i\delta_{ij} & S_i\delta_{ij} & 0 \\ S_i^*\delta_{ij} & R_i\delta_{ij} & 0 \\ 0 & 0 & \Pi_0 \end{bmatrix} \tag{3.6.3}$$

where Q represents the correlation of U, R represents the correlation of V, and S captures the correlation between U and V.

The goal of this section is to derive a recursion for $\widehat{X}_{i+1|i}$ in terms of $\widehat{X}_{i|i-1}$.

Remark 3.6.1. We *did not* make assumptions on the means of the random variables X_0, U_i, V_j. That is because we seek optimal linear estimators of states X_i using observations Y_j. The Wiener-Hopf equations only rely on information in this matrix but not the means of these random variables.

With this structure, we have the following properties:

1. $\langle U_i, X_j \rangle = 0$, $\langle V_i, X_j \rangle = 0$ for $j \leq i$.
2. $\langle U_i, Y_j \rangle = 0$, $\langle V_i, Y_j \rangle = 0$ for $j \leq i-1$ (there may be correlation between U_i and V_i)
3. $\langle U_i, Y_i \rangle = S_i$, $\langle V_i, Y_i \rangle = R_i$
4. $\Pi_i := \langle X_i, X_i \rangle$, $\Pi_{i+1} = F_i \Pi_i F_i^* + G_i Q_i G_i^*$.

With $\widehat{Y}_{i|i-1}$ as the one-step optimal linear predictor of Y_i as previously defined, we can define the prediction error

$$e_i := Y_i - \widehat{Y}_{i|i-1} \quad \text{where} \quad Y_i = H_i X_i + V_i. \tag{3.6.4}$$

We should know in the context of Kalman's recursion that we already know $\widehat{Y}_{i|i-1}$.

The optimal linear predictor of the output can therefore be expressed as

$$\widehat{Y}_{i|i-1} = H_i \widehat{X}_{i|i-1} + \underbrace{\widehat{V}_{i|i-1}}_{=0} \tag{3.6.5}$$

$$= H_i \widehat{X}_{i|i-1}. \tag{3.6.6}$$

$$\implies e_i = Y_i - H_i \widehat{X}_{i|i-1}. \tag{3.6.7}$$

Likewise, the optimal linear one-step predictor of the state $\widehat{X}_{i+1|i}$ is a linear combination of the orthogonal innovations by projecting onto the subspace spanned by those vectors:

$$\widehat{X}_{i+1|i} = \widehat{X}_{i+1|e_0,\ldots,e_i} \tag{3.6.8}$$

$$= \sum_{j=0}^{i} \langle X_{i+1}, e_j \rangle \langle e_j, e_j \rangle^{-1} e_j \tag{3.6.9}$$

$$= \underbrace{\sum_{j=0}^{i-1} \langle X_{i+1}, e_j \rangle \langle e_j, e_j \rangle^{-1} e_j}_{=\widehat{X}_{i+1|i-1}} + \langle X_{i+1}, e_i \rangle \langle e_i, e_i \rangle^{-1} e_i \tag{3.6.10}$$

$$= \widehat{X}_{i+1|i-1} + \langle X_{i+1}, e_i \rangle \underbrace{\langle e_i, e_i \rangle^{-1}}_{:=R_{e,i}^{-1}} e_i \tag{3.6.11}$$

Since X is a wide-sense Markov process, we have

$$\widehat{X}_{i+1|i-1} = F_i \widehat{X}_{i|i-1} + G_i \underbrace{\widehat{U}_{i|i-1}}_{=0} \tag{3.6.12}$$

$$= F_i\widehat{X}_{i|i-1}. \tag{3.6.13}$$

This gives us the Kalman filter:

$$\widehat{X}_{i+1|i} = F_i\widehat{X}_{i|i-1} + K_{p,i}e_i \quad \text{for all } i \in \mathbb{N}_0. \tag{3.6.14}$$

Here

$$K_{p,i} := \langle X_{i+1}, e_i\rangle\, R_{e,i} \tag{3.6.15}$$

is called the **prediction gain** or **predicted Kalman gain**.

An equivalent representation of the one-step optimal linear predictor of the state $\widehat{X}_{i+1|i}$ in terms of the measured output Y_i is given by

$$\widehat{X}_{i+1|i} = F_{p,i}\widehat{X}_{i|i-1} + K_{p,i}Y_i \quad \text{for all } i \in \mathbb{N}_0 \qquad \text{where} \quad F_{p,i} := F_i - K_{p,i}H_i. \tag{3.6.16}$$

For completeness, we write down the algorithm.

Algorithm 3.6.2 (Prediction Kalman Filter).

```
**Initialize:** $\Pi_0 \leftarrow \langle X_0, X_0\rangle$
**Initialize:** $P_0 \leftarrow \langle X_0, X_0\rangle$
**Initialize:** $\widehat{X}_{0|-1} \leftarrow 0$
**for** $i \in \mathbb{N}_0$ **do**
    $e_i \leftarrow Y_i - H_i\widehat{X}_{i|i-1}$
    $R_{e,i} \leftarrow H_iP_{i|i-1}H_i^* + R_i$
    $K_{p,i} \leftarrow \left(F_iP_{i|i-1}H_i^* + G_iS_i\right)R_{e,i}^{-1}$
    $P_{i+1|i} \leftarrow F_iP_{i|i-1}F_i^* + G_iQ_iG_i^* - K_{p,i}R_{e,i}K_{p,i}^*$    (Discrete-Time Ricatti Equation)
    $\widehat{X}_{i+1|i} = F_i\widehat{X}_{i|i-1} + K_{p,i}e_i$
```

Here $P_{i|i-1} = \left\langle X_i - \widehat{X}_{i|i-1}, X_i - \widehat{X}_{i|i-1}\right\rangle$ is the covariance of the prediction error.

Now we verify how to compute the quantities $R_{e,i}$ and $K_{p,i}$.

For convenience, define $\widetilde{X}_{i|i-1} = X_i - \widehat{X}_{i|i-1}$. So then $P_{i|i-1} = \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1}\right\rangle$.

We first compute $R_{e,i}$. Then

$$e_i = Y_i - H_i\widehat{X}_{i|i-1} \tag{3.6.17}$$
$$= H_iX_i + V_i - H_i\widehat{X}_{i|i-1} \tag{3.6.18}$$
$$= H_i\widetilde{X}_{i|i-1} + V_i \tag{3.6.19}$$
$$\implies R_{e,i} = \langle e_i, e_i\rangle \tag{3.6.20}$$
$$= \left\langle H_i\widetilde{X}_{i|i-1} + V_i, H_i\widetilde{X}_{i|i-1} + V_i\right\rangle \tag{3.6.21}$$
$$= H_i\left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1}\right\rangle H_i^* + \langle V_i, V_i\rangle \quad \text{since } \left\langle \widetilde{X}_{i|i-1}, V_i\right\rangle = 0 \tag{3.6.22}$$
$$= H_iP_{i|i-1}H_i^* + R_i. \tag{3.6.23}$$

Now we compute $K_{p,i} = \langle X_{i+1}, e_i\rangle R_{e,i}^{-1}$. We just need to compute $\langle X_{i+1}, e_i\rangle$. By wide-sense Markovity of X,

$$\langle X_{i+1}, e_i\rangle = \langle F_iX_i + G_iU_i, e_i\rangle \tag{3.6.24}$$
$$= F_i\langle X_i, e_i\rangle + G_i\langle U_i, e_i\rangle. \tag{3.6.25}$$
$$\langle X_i, e_i\rangle = \left\langle X_i, H_i\widetilde{X}_{i|i-1} + V_i\right\rangle \tag{3.6.26}$$
$$= \left\langle X_i, H_i\widetilde{X}_{i|i-1}\right\rangle + \underbrace{\langle X_i, V_i\rangle}_{=0} \tag{3.6.27}$$

$$= \left\langle X_i, H_i\widetilde{X}_{i|i-1}\right\rangle \tag{3.6.28}$$

$$= \left\langle \widehat{X}_{i|i-1} + \widetilde{X}_{i|i-1}, H_i\widetilde{X}_{i|i-1}\right\rangle \tag{3.6.29}$$

$$= \left\langle \widehat{X}_{i|i-1} + \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1}\right\rangle H_i^* \tag{3.6.30}$$

$$= \left(\underbrace{\left\langle \widehat{X}_{i|i-1}, \widetilde{X}_{i|i-1}\right\rangle}_{=0} + \underbrace{\left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1}\right\rangle}_{=P_{i|i-1}}\right) H_i^* \tag{3.6.31}$$

$$= P_{i|i-1}H_i^*. \tag{3.6.32}$$

$$\langle U_i, e_i\rangle = \left\langle U_i, H_i\widetilde{X}_{i-1} + V_i\right\rangle \tag{3.6.33}$$

$$= \left\langle U_i, H_i\widetilde{X}_{i|i-1}\right\rangle + \langle U_i, V_i\rangle \tag{3.6.34}$$

$$= \left\langle U_i, \widetilde{X}_{i|i-1}\right\rangle H^* + S_i \tag{3.6.35}$$

$$= \left\langle U_i, X_i - \widehat{X}_{i|i-1}\right\rangle H^* + S_i \tag{3.6.36}$$

$$= \left(\underbrace{\langle U_i, X_i\rangle}_{=0} - \underbrace{\left\langle U_i, \widehat{X}_{i|i-1}\right\rangle}_{=0}\right) H^* + S_i \tag{3.6.37}$$

$$= S_i. \tag{3.6.38}$$

$$\implies \langle X_{i+1}, e_i\rangle = F_i\langle X_i, e_i\rangle + G_i\langle U_i, e_i\rangle \tag{3.6.39}$$

$$= F_iP_{i|i-1}H_i^* + G_iS_i \tag{3.6.40}$$

$$\implies K_{p,i} = \langle X_{i+1}, e_i\rangle R_{e,i}^{-1} \tag{3.6.41}$$

$$= \left(F_iP_{i|i-1}H_i^* + G_iS_i\right) R_{e,i}^{-1}. \tag{3.6.42}$$

The update Ricatti equation is determined by finding the autocorrelation of both sides of the Kalman filter

$$\widehat{X}_{i+1|i} = F_i\widehat{X}_{i|i-1} + K_{p,i}e_i \quad \text{where} \quad \left\langle e_i, \widehat{X}_{i|i-1}\right\rangle = 0. \tag{3.6.43}$$

The update Ricatti Equation is determined by finding the autocorrelation of both sides of the Kalman Filter $\widehat{X}_{i+1|i} = F_i\widehat{X}_{i|i-1} + K_{p,i}e_i$ where $\left\langle e_i, \widehat{X}_{i|i-1}\right\rangle = 0$. Define

$$\Sigma_{i|i-1} := \left\langle \widehat{X}_{i|i-1}, \widehat{X}_{i|i-1}\right\rangle. \tag{3.6.44}$$

We have

$$\Sigma_0 = \left\langle \widehat{X}_{0|-1}, \widehat{X}_{0|-1}\right\rangle = 0 \tag{3.6.45}$$

$$\Sigma_{i+1} = \left\langle \widehat{X}_{i+1|i}, \widehat{X}_{i+1|i}\right\rangle \tag{3.6.46}$$

$$= \left\langle F_i\widehat{X}_{i|i-1} + K_{p,i}e_i, F_i\widehat{X}_{i|i-1} + K_{p,i}e_i\right\rangle \tag{3.6.47}$$

$$= F_i\Sigma_{i|i-1}F_i^* + K_{p,i}R_{e,i}K_{p,i}^*. \tag{3.6.48}$$

The desired term is $P_{i|i-1} = \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1}\right\rangle$. Since

$$X_i = \widehat{X}_{i|i-1} + \widetilde{X}_{i|i-1} \quad \text{and} \quad \left\langle \widehat{X}_{i|i-1}, \widetilde{X}_{i|i-1}\right\rangle = 0, \tag{3.6.49}$$

we have

$$\langle X_i, X_i\rangle = \left\langle \widehat{X}_{i|i-1} + \widetilde{X}_{i|i-1}, \widehat{X}_{i|i-1} + \widetilde{X}_{i|i-1}\right\rangle \tag{3.6.50}$$

$$\begin{align}
\implies \Pi_i &= \Sigma_{i|i-1} + P_{i|i-1} \tag{3.6.51}\\
\implies P_{i+1|i} &= \Pi_{i+1} - \Sigma_{i+1} \tag{3.6.52}\\
&= (F_i\Pi_iF_i^* + G_iQ_iG_i^*) - \left(F_i\Sigma_{i|i-1}F_i^* + K_{p,i}R_{e,i}K_{p,i}^*\right) \tag{3.6.53}\\
&= F_i\left(\Pi_i - \Sigma_{i|i-1}\right)F_i^* + G_iQ_iG_i^* - K_{p,i}R_{e,i}K_{p,i}^* \tag{3.6.54}\\
&= F_iP_{i|i-1}F_i^* + G_iQ_iG_i^* - K_{p,i}R_{e,i}K_{p,i}^*. \tag{3.6.55}
\end{align}$$

Remark 3.6.3. One benefit of this way to derive the Kalman filter is that we may permit $S_i \neq 0$; backward recursion does not allow that.

Remark 3.6.4. The actual implementation has some subtleties in the implementation. For this we use the *array algorithm* in practice.

Remark 3.6.5. When the system is LTI, there are better algorithms than the Kalman filter.

Remark 3.6.6. There are a few key sanity checks we can make; notice that several things in $P_{i+1|i}$ are PSD, and so on.

We can decompose the update into measurement and time updates.

3.6.2 Measurement Update

The measurement update is

$$\begin{align}
\widehat{X}_{i|i} &= \widehat{X}_{i|i-1} + K_{f,i}e_i \tag{3.6.56}\\
P_{i|i} &= P_{i|i-1} - K_{f,i}R_{e,i}K_{f,i}^* \tag{3.6.57}\\
K_{f,i} &= P_{i|i-1}H_i^*R_{e,i}^{-1}. \tag{3.6.58}
\end{align}$$

Note that this is really a filter – we try finding X_i given the observations including Y_i (which is a linear function of X_i).

The **filtered Kalman gain** $K_{f,i}$ is much simpler than the prediction Kalman gain, and the reason is that we do not have to do one step prediction; we already have the innovation for timestep i.

We will try to give a proof. Write

$$\begin{align}
\widehat{X}_{i|i} &= \sum_{j=0}^{i} \langle X_i, e_j\rangle R_{e,j}^{-1}e_j \tag{3.6.59}\\
&= \left(\sum_{j=0}^{n-1} \langle X_i, e_j\rangle R_{e,j}^{-1}e_j\right) + \langle X_i, e_i\rangle R_{e,i}^{-1}e_i \tag{3.6.60}\\
&= \widehat{X}_{i|i-1} + \langle X_i, e_i\rangle R_{e,i}^{-1}e_i. \tag{3.6.61}
\end{align}$$

We compute

$$\begin{align}
e_i &= Y_i - H_i\widehat{X}_{i|i-1} \tag{3.6.62}\\
&= H_iX_i + V_i - H_i\widehat{X}_{i|i-1} \tag{3.6.63}\\
&= H_i\widetilde{X}_{i|i-1} + V_i. \tag{3.6.64}
\end{align}$$

Plugging this back in,

$$\langle X_i, e_i\rangle = \left\langle X_i, H_i\widetilde{X}_{i|i-1} + V_i\right\rangle \tag{3.6.65}$$

$$= \left\langle X_i, H_i\widetilde{X}_{i|i-1} \right\rangle + \underbrace{\langle X_i, V_i \rangle}_{=0} \tag{3.6.66}$$

$$= \left\langle X_i, \widetilde{X}_{i|i-1} \right\rangle H_i^* \tag{3.6.67}$$

$$= \left\langle \widetilde{X}_{i|i-1} + \widehat{X}_{i|i-1}, \widetilde{X}_{i|i-1} \right\rangle H_i^* \tag{3.6.68}$$

$$= \left(\left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1} \right\rangle + \underbrace{\left\langle \widehat{X}_{i|i-1}, \widetilde{X}_{i|i-1} \right\rangle}_{=0} \right) H_i^* \tag{3.6.69}$$

$$= \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1} \right\rangle H_i^* \tag{3.6.70}$$

$$= P_{i|i-1} H_i^* \tag{3.6.71}$$

and plugging that in shows the recursion for $\widehat{X}_{i|i}$ as desired.

Now for $P_{i|i}$, we write

$$P_{i|i} = \left\langle \widetilde{X}_{i|i}, \widetilde{X}_{i|i} \right\rangle \tag{3.6.72}$$

$$= \left\langle \widetilde{X}_{i|i}, X_i - \widehat{X}_{i|i} \right\rangle \tag{3.6.73}$$

$$= P_{i|i-1} - K_{f,i} R_{e,i} K_{f,i}^*. \tag{3.6.74}$$

3.6.3 Time Update

The time update is

$$\widehat{U}_{i|i} = S_i R_{e,i}^{-1} e_i \tag{3.6.75}$$

$$\widehat{X}_{i+1|i} = F_i \widehat{X}_{i|i} + G_i \widehat{U}_{i|i} \tag{3.6.76}$$

$$P_{i+1|i} = F_i P_{i|i} F_i^* + G_i \left(Q_i - S_i R_{e,i}^{-1} S_i^* \right) G_i^* \tag{3.6.77}$$

$$- F_i K_{f,i} S_i^* G_i^* - G_i S_i K_{f,i}^* F_i^* \tag{3.6.78}$$

This is crazy complicated, and so we do not make any derivation. Let us attempt to give an interpretation.

When $S_i = 0$, many terms simplify, and we get the usual formulas. This is a good sanity check. We can also recover estimates of U_i given Y_i, that is, $\widehat{U}_{i|i}$, by exploiting correlation between U_i and V_i. Then we can use it to improve our filter. The way to get this correction term is due to the following:

$$\widehat{U}_{i|i} = \sum_{j=0}^{i} \underbrace{\langle U_i, e_j \rangle}_{=0 \text{ for } j<i} R_{e,j}^{-1} e_j \tag{3.6.79}$$

$$= \langle U_i, e_i \rangle R_{e,i}^{-1} e_i \tag{3.6.80}$$

$$= \left\langle U_i, H_i \widetilde{X}_{i|i-1} + V_i \right\rangle R_{e,i}^{-1} e_i \tag{3.6.81}$$

$$= \left(\underbrace{\left\langle U_i, \widetilde{X}_{i|i-1} \right\rangle}_{=0} H_i + \langle U_i, V_i \rangle \right) R_{e,i}^{-1} e_i \tag{3.6.82}$$

$$= S_i R_{e,i}^{-1} e_i. \tag{3.6.83}$$

3.7 Filtered Kalman Filter

Our goal is to derive a recursion for $\widehat{X}_{i+1|i+1}$ given $\widehat{X}_{i|i}$. Contrast this to predictive Kalman filter, where the goal is to derive a recursion for $\widehat{X}_{i+1|i}$ given $\widehat{X}_{i|i-1}$.

We are only able to write down the filtered Kalman filter in the case $S_i = 0$. The recursion turns out to be

$$e_{i+1} = Y_{i+1} - H_i F_i \widehat{X}_{i|i} \tag{3.7.1}$$

$$\widehat{X}_{i+1|i+1} = F_i \widehat{X}_{i|i} + K_{f,i+1} e_{i+1} \tag{3.7.2}$$

$$P_{i+1|i+1} = F_i P_{i|i} F_i^* + G_i Q_i G_i^* - K_{f,i+1} R_{e,i+1} K_{f,i+1}^* \tag{3.7.3}$$

with initialization

$$\widehat{X}_{0|0} = \Pi_0 H_0^* R_{e,0}^{-1} Y_0 \tag{3.7.4}$$

$$e_0 = Y_0 \tag{3.7.5}$$

$$P_{0|0} = \Pi_0 - \Pi_0 H^* R_{e,0}^{-1} H_0 \Pi_0. \tag{3.7.6}$$

This is already more complicated than the case of predictive Kalman filter. Even the initialization is different; we are using the de-autocorrelation of Y_0 to get a good initialization. Other than that, the complexity is roughly the same.

Remark 3.7.1. The filtered Kalman filter limit may be different from the predictive Kalman filter limit.

3.8 Kalman Smoother

Previously we have been finding estimators of the type $\widehat{X}_{i+1|i}$ (predictive Kalman filter) and $\widehat{X}_{i|i}$ (filtered Kalman filter).

Estimating $\widehat{X}_{i+m|i}$ for $m > 0$ is quite trivial using the relationship $X_{i+1} = F_i X_i + G_i U_i$. Even in the case of $S_i \neq 0$, we have derived $\widehat{X}_{i+1|i}$ using the time-update formula, and if we want $\widehat{X}_{i+m|i}$ for $m \geq 2$, we have

$$\widehat{X}_{i+m|i} = F_{i+m-1} F_{i+m-2} \cdots F_{i+1} \widehat{X}_{i+1|i}. \tag{3.8.1}$$

It is unclear how to obtain $\widehat{X}_{i+m|i}$ for $m < 0$. We derive this now.

We assume we are dealing with a fixed time horizon N. We denote $\widehat{X}_{i|N}$ to be the optimal linear estimate of X_i given all the data $\{Y_0, Y_1, \ldots, Y_N\}$.

Our goal is to recursively compute $\widehat{X}_{i|N}$. We guess that the approach should correspond to forward-backward recursion. We have already seen the forward-recursion in the predictive Kalman filter. Our goal is basically to derive the backward recursion.

Let us compute the innovation. We have

$$e_0 = Y_0 \tag{3.8.2}$$

$$e_j = Y_j - \widehat{Y}_{j|j-1} \tag{3.8.3}$$

$$= Y_j - H_j \widehat{X}_{j|j-1} \tag{3.8.4}$$

$$= H_j X_j + V_j - H_j \widehat{X}_{j|j-1} \tag{3.8.5}$$

$$= H_j \widetilde{X}_{j|j-1} + V_j. \tag{3.8.6}$$

Our goal is to get an estimate for $\widehat{X}_{i|N}$ using the formula for general optimal estimation given orthogonal random variables. We have

$$\widehat{X}_{i|N} = \sum_{j=0}^{N} \langle X_i, e_j \rangle R_{e_j}^{-1} e_j \tag{3.8.7}$$

$$= \sum_{j=0}^{i-1} \langle X_i, e_j \rangle R_{e_j}^{-1} e_j + \sum_{j=i}^{N} \langle X_i, e_j \rangle R_{e_j}^{-1} e_j \tag{3.8.8}$$

$$= \widehat{X}_{i|i-1} + \sum_{j=i}^{N} \langle X_i, e_j \rangle R_{e_j}^{-1} e_j. \tag{3.8.9}$$

Note that R_{e_j} is the same as what we have computed before:

$$\begin{aligned}
R_{e_j} &= \langle e_j, e_j \rangle && (3.8.10)\\
&= \left\langle H_j \widetilde{X}_{j|j-1} + V_j, H_j \widetilde{X}_{j|j-1} + V_j \right\rangle && (3.8.11)\\
&= H_j \left\langle \widetilde{X}_{j|j-1}, \widetilde{X}_{j|j-1} \right\rangle H_j^* + \langle V_j, V_j \rangle && (3.8.12)\\
&= H_j P_{j|j-1} H_j^* + R_j && (3.8.13)
\end{aligned}$$

with $P_{j|j-1} = \left\langle \widetilde{X}_{j|j-1}, \widetilde{X}_{j|j-1} \right\rangle$ being computed by the predictive Kalman filter. The only thing that is left to compute is $\langle X_i, e_j \rangle$, for $j \geq i$. Indeed,

$$\begin{aligned}
\langle X_i, e_j \rangle &= \left\langle X_i, H_j \widetilde{X}_{j|j-1} + V_j \right\rangle && (3.8.14)\\
&= \left\langle X_i, H_j \widetilde{X}_j \right\rangle + \underbrace{\langle X_i, V_j \rangle}_{=0} && (3.8.15)\\
&= \left\langle X_i, H_j \widetilde{X}_j \right\rangle && (3.8.16)\\
&= \left\langle X_i, \widetilde{X}_j \right\rangle H_j^* && (3.8.17)\\
&= \left\langle \widehat{X}_{i|i-1} + \widetilde{X}_{i|i-1}, \widetilde{X}_{j|j-1} \right\rangle H_j^* && (3.8.18)\\
&= \left(\underbrace{\left\langle \widehat{X}_{i|i-1}, \widetilde{X}_{j|j-1} \right\rangle}_{=0} + \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{j|j-1} \right\rangle \right) H_j^* && (3.8.19)\\
&= \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{j|j-1} \right\rangle H_j^* && (3.8.20)\\
&= P_{ij} H_j^*. && (3.8.21)
\end{aligned}$$

Here we use that $\left\langle \widehat{X}_{i|i-1}, \widetilde{X}_{j|j-1} \right\rangle = 0$, but this requires some justification. Indeed, $\widehat{X}_{i|i-1} \in \mathrm{Span}(Y_0, \ldots, Y_{i-1})$, but by orthogonality principle, $\widetilde{X}_j \perp \mathrm{Span}(Y_0, \ldots, Y_{j-1})$, hence $\widetilde{X}_j \perp \mathrm{Span}(Y_0, \ldots, Y_{i-1}) \subseteq \mathrm{Span}(Y_0, \ldots, Y_{j-1})$.

We also denote $P_{ij} = \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{j|j-1} \right\rangle$. Note that $P_{ii} = P_{i|i-1}$, so it is a generalization of the error covariances we already defined. We now compute P_{ij}.

We claim that for any $j \geq i$,

$$P_{ij} = P_{i|i-1} \Phi_p^*(j, i) \quad \text{where} \quad \Phi_p(j, i) = \begin{cases} I & j = i \\ F_{p,j-1} F_{p,j-2} \cdots F_{p,i} & j > i \end{cases}. \tag{3.8.22}$$

We can compute $P_{i|i-1}$ by the Kalman filter, and $F_{p,i} = F_i - K_{p,i} H_i$.

Indeed, we know $P_{ij} = \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{j|j-1} \right\rangle$. We want to write $\widetilde{X}_{j|j-1}$ in terms of $\widetilde{X}_{i|i-1}$. We first use our Kalman filter iteration of the predicted estimator,

$$\begin{aligned}
\widehat{X}_{i+1|i} &= F_i \widehat{X}_{i|i-1} + K_{p,i} e_i \qquad = F_i \widehat{X}_{i|i-1} + K_{p,i} \left(H_i \widetilde{X}_{i|i-1} + V_i \right) && (3.8.23)\\
\widetilde{X}_{i+1|i} &= X_{i+1} - \widehat{X}_{i+1|i} && (3.8.24)\\
&= F_i X_i + G_i U_i - \widehat{X}_{i+1|i} && (3.8.25)\\
&= F_i X_i + G_i U_i - F_i \widehat{X}_{i|i-1} - K_{p,i} \left(H_i \widetilde{X}_{i|i-1} + V_i \right) && (3.8.26)
\end{aligned}$$

$$= F_i \left(X_i - \widehat{X}_{i|i-1} \right) + G_i U_i - K_{p,i} H_i \widetilde{X}_{i|i-1} - K_{p,i} V_i \tag{3.8.27}$$

$$= F_i \widetilde{X}_{i|i-1} + G_i U_i - K_{p,i} H_i \widetilde{X}_{i|i-1} - K_{p,i} V_i \tag{3.8.28}$$

$$= \left(F_i - K_{p,i} H_i \right) \widetilde{X}_{i|i-1} + G_i U_i - K_{p,i} V_i \tag{3.8.29}$$

$$= F_{p,i} \widetilde{X}_{i|i-1} + G_i U_i - K_{p,i} V_i \tag{3.8.30}$$

using our definition of $F_{p,i} = F_i - K_{p,i} H_i$.

Here we can see that if $j > i$, the noises are new and so best estimated by 0. So if $j > i$ then we can write the leading term

$$\widehat{X}_{j+1|j} = \Phi_p(j,i) \widetilde{X}_{i|i-1} + \text{noise} \tag{3.8.31}$$

and show that the noise is orthogonal to $\widetilde{X}_{i|i-1}$. We formalize this calculation now.

Indeed,

$$\widetilde{X}_{j|j-1} = \Phi_p(j,i) \widetilde{X}_{i|i-1} + \sum_{k=i}^{j-1} \Phi_p(j,k+1) \left(G_k U_k - K_{p,k} V_k \right) \tag{3.8.32}$$

$$P_{ij} = \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{j|j-1} \right\rangle \tag{3.8.33}$$

$$= \left\langle \widetilde{X}_{i|i-1}, \Phi_p(j,i) \widetilde{X}_{i|i-1} + \sum_{k=i}^{j-1} \Phi_p(j,k+1) \left(G_k U_k - K_{p,k} V_k \right) \right\rangle \tag{3.8.34}$$

$$= \left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1} \right\rangle \Phi_p(j,i)^* + \sum_{k=i}^{j-1} \left\langle \widetilde{X}_{i|i-1}, \Phi_p(j,k+1) \left(G_k U_k - K_{p,k} V_k \right) \right\rangle \tag{3.8.35}$$

$$= P_{i|i-1} \Phi_p(j,i)^* + \sum_{k=i}^{j-1} \left\langle \widetilde{X}_{i|i-1}, G_k U_k - K_{p,k} V_k \right\rangle \Phi_p(j,k+1)^* \tag{3.8.36}$$

$$= P_{i|i-1} \Phi_p(j,i)^* + \sum_{k=i}^{j-1} \left\langle \widetilde{X}_{i|i-1}, G_k U_k - K_{p,k} V_k \right\rangle \Phi_p(j,k+1)^* \tag{3.8.37}$$

$$= P_{i|i-1} \Phi_p(j,i)^* + \sum_{k=i}^{j-1} \left(\left\langle \widetilde{X}_{i|i-1}, G_k U_k \right\rangle - \left\langle \widetilde{X}_{i|i-1}, K_{p,k} V_k \right\rangle \right) \Phi_p(j,k+1)^* \tag{3.8.38}$$

$$= P_{i|i-1} \Phi_p(j,i)^* + \sum_{k=i}^{j-1} \left(\underbrace{\left\langle \widetilde{X}_{i|i-1}, U_k \right\rangle}_{=0} G_k^* - \underbrace{\left\langle \widetilde{X}_{i|i-1}, V_k \right\rangle}_{=0} K_{p,k}^* \right) \Phi_p(j,k+1)^* \tag{3.8.39}$$

$$= P_{i|i-1} \Phi_p(j,i)^*. \tag{3.8.40}$$

In essence, we are done, but we want to make a recursion so that it's feasible to implement the Kalman smoother. Indeed,

$$\widehat{X}_{i|N} = \widehat{X}_{i|i-1} + \sum_{j=i}^{N} P_{ij} H_j^* R_{e_j}^{-1} e_j \tag{3.8.41}$$

$$= \widehat{X}_{i|i-1} + P_i \sum_{j=i}^{N} \Phi_p(j,i)^* H_j^* R_{e_j}^{-1} e_j. \tag{3.8.42}$$

Define $\lambda_{i|N} = \sum_{j=i}^{N} \Phi_p(j,i)^* H_j^* R_{e_j}^{-1} e_j$. We can use backwards recursion to compute each $\lambda_{i|N}$:

$$\lambda_{i|N} = \sum_{j=i}^{N} \Phi_p(j,i)^* H_j^* R_{e_j}^{-1} e_j \tag{3.8.43}$$

$$= F_{p,i}^{*}\lambda_{i+1|N} + H_i^{*}R_{e_i}^{-1}e_i \tag{3.8.44}$$

with initialization $\lambda_{N+1|N} = 0$.

We see that this is a backwards recursion, because in our forward pass we calculated all of the $\widehat{X}_i$'s and e_i's. Once we have these, we initialize $\lambda_{N+1|N} = 0$, then calculate the other $\lambda_{i|N}$ going backwards. We then use these $\lambda_{i|N}$'s to calculate $\widehat{X}_{i|N}$.

Altogether, we have

$$\widehat{X}_{i|N} = \widehat{X}_{i|i-1} + P_{i|i-1}\lambda_{i|N} \tag{3.8.45}$$

$$\lambda_{i|N} = F_{p,i}^{*}\lambda_{i+1|N} + H_i^{*}R_{e_i}^{-1}e_i \tag{3.8.46}$$

with initialization $\lambda_{N+1|N} = 0$.

The last thing we need to find is $P_{i|N} := \left\langle X_i - \widehat{X}_{i|N}, X_i - \widehat{X}_{i|N} \right\rangle$, the error covariance for the optimal non-causal estimate.

Indeed, we claim that

$$P_{i|N} = P_{i|i-1} - P_{i|i-1}\Lambda_{i|N}P_{i|i-1} \quad \text{where} \quad \Lambda_{i|N} = \langle \lambda_{i|N}, \lambda_{i|N} \rangle = F_{p,i}^{*}\Lambda_{i+1|N}F_{p,i} + H_i^{*}R_{e_i}^{-1}H_i. \tag{3.8.47}$$

and $\Lambda_{N+1|N} = 0$.

Indeed, we first prove the recursive formula for $\Lambda_{i|N}$. We know $\lambda_{i|N} = F_{p,i}^{*}\lambda_{i+1|N} + H_i^{*}R_{e_i}^{-1}e_i$. Since $\lambda_{i+1|N}$ is a linear combination of e_j for $j = i+1, \ldots, N$, then $\langle \lambda_{i+1|N}, e_i \rangle = 0$. So

$$\begin{aligned}
\langle \lambda_{i|N}, \lambda_{i|N} \rangle &= \langle F_{p,i}^{*}\lambda_{i+1|N} + H_i^{*}R_{e_i}^{-1}e_i, F_{p,i}^{*}\lambda_{i+1|N} + H_i^{*}R_{e_i}^{-1}e_i \rangle && (3.8.48)\\
&= \langle F_{p,i}^{*}\lambda_{i+1|N}, F_{p,i}^{*}\lambda_{i+1|N} \rangle + \langle H_i^{*}R_{e_i}^{-1}e_i, H_i^{*}R_{e_i}^{-1}e_i \rangle && (3.8.49)\\
&= F_{p,i}^{*} \langle \lambda_{i+1|N}, \lambda_{i+1|N} \rangle F_{p,i} + H_i^{*}R_{e_i}^{-1} \langle e_i, e_i \rangle R_{e_i}^{-*}H_i && (3.8.50)\\
&= F_{p,i}^{*}\Lambda_{i+1|N}F_{p,i} + H_i^{*}R_{e_i}^{-1}R_{e_i}R_{e_i}^{-1}H_i && (3.8.51)\\
&= F_{p,i}^{*}\Lambda_{i+1|N}F_{p,i} + H_i^{*}R_{e_i}^{-1}H_i. && (3.8.52)
\end{aligned}$$

Now we find the formula for $P_{i|N}$. Starting from our formula for $\widehat{X}_{i|N}$,

$$\begin{aligned}
\widehat{X}_{i|N} &= \widehat{X}_{i|i-1} + P_{i|i-1}\lambda_{i|N} && (3.8.53)\\
\widehat{X}_{i|N} - X_i &= \widehat{X}_{i|i-1} + P_{i|i-1}\lambda_{i|N} - X_i && (3.8.54)\\
-\widetilde{X}_{i|N} &= -\widetilde{X}_{i|i-1} + P_{i|i-1}\lambda_{i|N} && (3.8.55)\\
\widetilde{X}_{i|i-1} &= \widetilde{X}_{i|N} + P_{i|i-1}\lambda_{i|N}. && (3.8.56)
\end{aligned}$$

Since $\widetilde{X}_{i|N}$ is the estimation error, it is orthogonal to all of the Y_i's and hence all of the e_i's. And $\lambda_{i|N}$ is a linear combination of the e_i's. So

$$\begin{aligned}
\left\langle \widetilde{X}_{i|i-1}, \widetilde{X}_{i|i-1} \right\rangle &= \left\langle \widetilde{X}_{i|N} + P_{i|i-1}\lambda_{i|N}, \widetilde{X}_{i|N} + P_{i|i-1}\lambda_{i|N} \right\rangle && (3.8.57)\\
&= \left\langle \widetilde{X}_{i|N}, \widetilde{X}_{i|N} \right\rangle + \langle P_{i|i-1}\lambda_{i|N}, P_{i|i-1}\lambda_{i|N} \rangle && (3.8.58)\\
P_{i|i-1} &= P_{i|N} + P_{i|i-1} \langle \lambda_{i|N}, \lambda_{i|N} \rangle P_{i|i-1}^{*} && (3.8.59)\\
&= P_{i|N} + P_{i|i-1}\Lambda_{i|N}P_{i|i-1}^{*} && (3.8.60)\\
\implies P_{i|N} &= P_{i|i-1} - P_{i|i-1}\Lambda_{i|N}P_{i|i-1}^{*}. && (3.8.61)
\end{aligned}$$

3.9 Array Algorithm

In predictive Kalman filter we propagate $P_{i|i-1} \to P_{i+1|i}$. The key idea is that $P_{i|i-1}$ is PSD, so one can store $P_{i|i-1}^{1/2}$ and multiply. The computation of array algorithms involve standard numerically stable methods of triangularization, which we also introduce here. The Gram–Schmidt approach, which achieves triangularization in infinite-precision arithmetics, is numerically unstable and hence not used for triangularization in practice.

The need for such an algorithm can be motivated by the following Kalman filtering problem. Consider the canonical state space model

$$X_{i+1} = X_i \tag{3.9.1}$$

$$Y_i = X_i + V_i \tag{3.9.2}$$

where $\langle V_i, V_j \rangle = \delta_{ij}$. The Kalman recursion on the predicted error variance $P_{i|i-1}$ is

$$P_{i+1|i} = P_{i|i-1} - \frac{P_i^2}{1 + P_i^2}. \tag{3.9.3}$$

If $P_{i|i-1}$ is large,

$$P_{i+1} = P_i - \frac{P_i^2}{1 + P_i} = \frac{P_i}{1 + P_i} \approx 1. \tag{3.9.4}$$

However, if implemented naively, rounding errors could lead to the second term in the recursion being

$$\frac{P_{i|i-1}^2}{1 + P_{i|i-1}} \approx \frac{P_{i|i-1}^2}{P_{i|i-1}} = P_{i|i-1} \tag{3.9.5}$$

The resulting recursion is therefore computed incorrectly:

$$P_{i+1|i} = P_{i|i-1} - \frac{P_{i|i-1}^2}{1 + P_{i|i-1}} \approx P_{i|i-1} - P_{i|i-1} = 0. \tag{3.9.6}$$

Hence, we would like to formulate a method which performs this recursion robustly without being susceptible to numerical errors.

In fact, our approach for this special case is the following: we form the 2 × 2 pre-array of numbers

$$A = \begin{bmatrix} 1 & \sqrt{P_i} \\ 0 & \sqrt{P_i} \end{bmatrix} \tag{3.9.7}$$

and then we will construct an orthogonal symmetric matrix Θ such that we *triangularize* the matrix A:

$$A\Theta = \begin{bmatrix} 1 & \sqrt{P_{i|i-1}} \\ 0 & \sqrt{P_{i|i-1}} \end{bmatrix} \Theta = \begin{bmatrix} C & 0 \\ D & E \end{bmatrix} \tag{3.9.8}$$

where C, D, E are real numbers. Indeed, $E = \sqrt{P_{i+1|i}}$.

Hence, as long as we can numerically triangularize in a stable way, we would safely obtain $P_{i+1|i}$. One way to do it is through the Givens rotation to be introduced earlier, which has the concrete form

$$\Theta = \frac{1}{\sqrt{1+\rho^2}} \begin{bmatrix} 1 & -\rho \\ \rho & 1 \end{bmatrix}, \tag{3.9.9}$$

where $\rho = \sqrt{P_{i|i-1}}$. If we do the matrix multiplication, we would end up using the formula

$$\sqrt{P_{i+1|i}} = \sqrt{P_{i|i-1}} \cdot \frac{1}{\sqrt{1 + P_{i|i-1}}} \tag{3.9.10}$$

which is definitely not as bad as the original approach since $\sqrt{P_{i|i-1}} \ll P_{i|i-1}$.

3.9.1 Triangularization

Given a row vector $x \in \mathbb{R}^{1\times n}$, we define a triangularization matrix Θ as one that is symmetric and orthogonal, i.e.,

$$\Theta = \Theta^* \quad \text{and} \quad \Theta\Theta^* = \Theta^2 = I_n \tag{3.9.11}$$

such that for some $\alpha \in \mathbb{R}$,

$$x\Theta = \alpha e_1^* \in \mathbb{R}^{1\times n}. \tag{3.9.12}$$

By the definition of Θ, we can constrain α to one of two values:

$$xx^* = x\Theta\Theta^* x^* = (x\Theta)(x\Theta)^* = (\alpha e_1^*)(\alpha e_1^*)^* \tag{3.9.13}$$

$$= \alpha^2 e_1^* e_1 = \alpha^2 \tag{3.9.14}$$

which implies that $\alpha = \pm \|x^*\|$.

We will now define two distinct transformations that can triangularize a matrix A in a numerically stable manner. The complex case requires more care.

3.9.2 Householder Reflection

Given a row vector x and corresponding transformed equivalent vector αe_1^*, where $\|x\| = \|\alpha e_1\|$, we define

$$g := x - \alpha e_1^*. \tag{3.9.15}$$

The projection of x onto g is given by

$$P_g(x) = \langle x, g\rangle \|g\|^{-2} g = \frac{xg^*}{gg^*} g. \tag{3.9.16}$$

We claim that $g = 2P_g(x)$. It suffices to consider the case of $g \neq 0$ otherwise the problem is trivial. Then it suffices to show that $2xg^* = gg^*$ Indeed,

$$2xg^* = 2xx^* - 2\alpha x e_1 \tag{3.9.17}$$

$$gg^* = xx^* + \alpha^2 e_1^* e_1 - 2\alpha e_1^* x^* \tag{3.9.18}$$

and these two terms are equal when $\alpha^2 = xx^*$, which is our very first result. We therefore have

$$\alpha e_1^* = x - g \tag{3.9.19}$$

$$= x - 2\frac{xg^*}{gg^*} g \tag{3.9.20}$$

$$= x\left(I - 2\frac{g^* g}{gg^*}\right) \tag{3.9.21}$$

$$= x\Theta. \tag{3.9.22}$$

Therefore, for any row vector x, we can determine the Householder triangularization matrix

$$\Theta = I - 2\frac{g^* g}{gg^*} \quad \text{where} \quad g = x - \alpha e_1. \tag{3.9.23}$$

Consider a matrix

$$A = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} \tag{3.9.24}$$

where $A \in \mathbb{R}^{m\times n}$ and $a_k \in \mathbb{R}^{1\times n}$. We now define

$$\Theta_0 = I - 2\frac{g^* g}{gg^*} \quad \text{where} \quad g = a_1 - \alpha_1 e_1^* \quad \text{and} \quad \alpha_1 = \pm \|a_1^*\|. \tag{3.9.25}$$

Therefore

$$A\Theta_0 = \begin{bmatrix} a_1\Theta_0 \\ a_2\Theta_0 \\ \vdots \\ a_m\Theta_0 \end{bmatrix} \tag{3.9.26}$$

$$= \begin{bmatrix} \alpha_1 & 0 & \cdots & 0 \\ ? & & & \\ \vdots & & A_1 & \\ ? & & & \end{bmatrix} \tag{3.9.27}$$

Here ? denotes arbitrary entries and $A_1 \in \mathbb{R}^{(m-1)\times(n-1)}$. We can subsequently define a Householder transform Θ_1' for A_1 such that

$$A_1\Theta_1' = \begin{bmatrix} \alpha_2 & 0 & \cdots & 0 \\ ? & & & \\ \vdots & & A_2 & \\ ? & & & \end{bmatrix} \tag{3.9.28}$$

We can then define

$$\Theta_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \Theta_1' & \\ 0 & & & \end{bmatrix} \tag{3.9.29}$$

such that

$$A\Theta_0\Theta_1 = \begin{bmatrix} \alpha_1 & 0 & \cdots & 0 \\ ? & & & \\ \vdots & & A_1\Theta_1' & \\ ? & & & \end{bmatrix} \tag{3.9.30}$$

$$= \begin{bmatrix} \alpha_1 & 0 & 0 & \cdots & 0 \\ ? & \alpha_2 & & & \\ \vdots & \vdots & & A_2 & \\ ? & ? & & & \end{bmatrix} \tag{3.9.31}$$

This process can be repeated to yield a lower triangular matrix, shown below for the case where $n > m$:

$$A\Theta = \begin{bmatrix} \alpha_1 & 0 & \cdots & 0 & 0 & \cdots & 0 & \\ ? & \alpha_2 & \cdots & 0 & 0 & \cdots & 0 & \\ ? & ? & \cdots & \alpha_n & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{3.9.32}$$

where $Theta = \Theta_0\Theta_1\cdots\Theta_{q-1}$ where $q = \min\{m, n\}$, and

$$\Theta_j = \begin{bmatrix} I_j & 0 \\ 0 & \Theta_j' \end{bmatrix}. \tag{3.9.33}$$

3.9.3 Givens Rotation

Any 2×2 rotation matrix can be written as

$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}, \tag{3.9.34}$$

which corresponds to rotating the vector counterclockwise by degree θ. We define a Givens rotation matrix

$$\Theta_g = \begin{bmatrix} \Theta_{g,11} & \Theta_{g,12} \\ \Theta_{g,21} & \Theta_{g,22} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{3.9.35}$$

such that, for $a \neq 0$,

$$\begin{bmatrix} a & b \end{bmatrix} \Theta_g = \begin{bmatrix} \alpha & 0 \end{bmatrix}. \tag{3.9.36}$$

We therefore have

$$\Theta_g = \frac{1}{\sqrt{1+\rho^2}} \begin{bmatrix} 1 & -\rho \\ \rho & 1 \end{bmatrix} \quad \text{where} \quad \rho = \frac{b}{a} \tag{3.9.37}$$

for which

$$\alpha = \operatorname{sign}(a)\sqrt{a^2+b^2} \tag{3.9.38}$$

if $a \neq 0$. And if $a = 0$ we take

$$\Theta_g = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{3.9.39}$$

for which $\alpha = b$.

For an n-dimensional row vector $x = \begin{bmatrix} x_1 & \cdots & x_i & \cdots & x_j & \cdots & x_n \end{bmatrix}$ where $1 \leq i < j \leq n$ such that $x_i \neq 0$, we can define the block Givens transform matrix

$$\Theta = \begin{bmatrix} I_{i-1} & 0 & 0 & 0 & 0 \\ 0 & \Theta_{g,11} & 0 & \Theta_{g,12} & 0 \\ 0 & 0 & I_{j-i-1} & 0 & 0 \\ 0 & \Theta_{g,21} & 0 & \Theta_{g,22} & 0 \\ 0 & 0 & 0 & 0 & I_{n-j} \end{bmatrix}. \tag{3.9.40}$$

We therefore have

$$x\Theta = \begin{bmatrix} x_1 & \cdots & \alpha_{ij} & \cdots & 0 & \cdots & x_n \end{bmatrix} \quad \text{where} \quad \alpha_{ij} = \operatorname{sign}(x_i)\sqrt{x_i^2 + x_j^2}. \tag{3.9.41}$$

This provides a computationally efficient way to zero out right-handed nonzero coefficients in sparse matrices, thereby aiding in triangularization.

3.9.4 Array Algorithms

For any $P \succeq 0$, we have the non-unique decomposition $P = AA^*$. Let $P^{1/2}$ be any such A. *Note:* The notation $P^{1/2}$ is often used in literature to denote the "true square root" of a PSD matrix P with diagonal representation $P = U\Lambda U^*$, where $P^{1/2} = U\Lambda^{1/2}U^*$ is unique.

We can now define the following matrices:

$$P^{*/2} = \left(P^{1/2}\right)^*, \quad P^{-1/2} = \left(P^{1/2}\right)^{-1}, \quad P^{-*/2} = \left(P^{-1/2}\right)^* \tag{3.9.42}$$

using which we have that

$$P = P^{1/2}P^{*/2}, \quad P^{-*} = P^{-1/2}P^{-*/2}. \tag{3.9.43}$$

Consider again the canonical Kalman filtering problem

$$X_{i+1} = F_iX_i + G_iU_i, \quad i \in \mathbb{N}_0 \tag{3.9.44}$$

$$Y_i = H_iX_i + V_i \tag{3.9.45}$$

where we assume $\langle U_i, V_j \rangle = 0$ and $\langle V_i, V_j \rangle = R_i\delta_{ij}$ and $R_i > 0$ for any i. The iteration on the error covariance matrix P is propagated using its corresponding square root forms for efficient computation as follows:

1. Time update: We have

$$\widehat{X}_{i+1|i} = F_i \widehat{X}_{i|i} \tag{3.9.46}$$

and

$$\begin{aligned} P_{i+1|i} &= F_i P_{i|i} F_i^* + G_i Q_i G_i^* \qquad &(3.9.47)\\ &= \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix}^* \qquad &(3.9.48)\\ &= \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \Theta\Theta^* \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix}^* \qquad &(3.9.49) \end{aligned}$$

where Θ is the Householder transform. Therefore

$$\begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \Theta = \begin{bmatrix} L & 0 \end{bmatrix} \tag{3.9.50}$$

where L is lower triangular. The time-updated error covariance is given by

$$P_{i+1|i} = \begin{bmatrix} L & 0 \end{bmatrix} \begin{bmatrix} L & 0 \end{bmatrix}^* = LL^* \implies L = P_{i+1|i}^{1/2}. \tag{3.9.51}$$

It is therefore sufficient to obtain L using any triangularization algorithm and propagate $P_{i|i}^{1/2}$ and $P_{i+1|i}^{1/2}$ through the Kalman filter time-update iterations.

2. THe measurement update for the error covariance is given by

$$P_{i|i} = P_{i|i-1} - P_{i|i-1} H_i^* R_{e,i}^{-1} H_i P_i. \tag{3.9.52}$$

Consider the following matrix, upon which we perform triangularization to obtain its lower triangular representation

$$\begin{bmatrix} R_i^{1/2} & H_i P_i^{1/2} \\ 0 & P_i^{1/2} \end{bmatrix} \Theta = \begin{bmatrix} L_1 & 0 \\ M & L_2 \end{bmatrix} \tag{3.9.53}$$

where L_1 and L_2 are also lower triangular. We can manipulate in the following manner:

$$\begin{bmatrix} L_1 & 0 \\ M & L_2 \end{bmatrix} \begin{bmatrix} L_1 & 0 \\ M & L_2 \end{bmatrix}^* = \begin{bmatrix} R_i^{1/2} & H_i P_i^{1/2} \\ 0 & P_i^{1/2} \end{bmatrix} \begin{bmatrix} R_i^{1/2} & H_i P_i^{1/2} \\ 0 & P_i^{1/2} \end{bmatrix}^* \tag{3.9.54}$$

to obtain

$$L_1 L_1^* = R_{e,i}, \quad M L_1^* = P_i H_i^*, \quad L_2 L_2^* = P_{i|i}. \tag{3.9.55}$$

Therefore $P_{i|i}^{1/2} = L_2$, which is lower triangular.

3.10 CKMS Recursion

The general state space model is

$$\begin{aligned} X_{i+1} &= F_i X_i + G_i U_i \qquad &(3.10.1)\\ Y_i &= H_i X_i + V_i \qquad &(3.10.2) \end{aligned}$$

with

$$\left\langle \begin{bmatrix} U_i \\ V_i \\ X_0 \end{bmatrix}, \begin{bmatrix} U_j \\ V_j \\ X_0 \end{bmatrix} \right\rangle = \begin{bmatrix} Q_i\delta_{ij} & S_i\delta_{ij} & 0 \\ S_i^*\delta_{ij} & R_i\delta_{ij} & 0 \\ 0 & 0 & \Pi_0 \end{bmatrix}. \tag{3.10.3}$$

The interesting question is: what happens when the process $(X_n)_{n\in\mathbb{N}_0}$ is wide-sense stationary, and $F_i = F$, $G_i = G$, $H_i = H$, $Q_i = Q$, $R_i = R$, and $S_i = S$ (i.e., all constants)?

It turns out that there is a computationally simpler method for this process. The purpose of this section is to discuss the method.

The state space model is now

$$X_{i+1} = FX_i + GU_i \tag{3.10.4}$$

$$Y_i = HX_i + V_i \tag{3.10.5}$$

with

$$\left\langle \begin{bmatrix} U_i \\ V_i \\ X_0 \end{bmatrix}, \begin{bmatrix} U_j \\ V_j \\ X_0 \end{bmatrix} \right\rangle = \begin{bmatrix} Q\delta_{ij} & S\delta_{ij} & 0 \\ S^*\delta_{ij} & R\delta_{ij} & 0 \\ 0 & 0 & \Pi_0 \end{bmatrix}. \tag{3.10.6}$$

Suppose $X_i \in \mathbb{C}^n$ and $U_i \in \mathbb{C}^m$, and $p \in \mathbb{C}^p$. Suppose further we have $p, m \leq n$. These properties are valid because our state summarizes everything that we know. We have $p \leq n$ because Y is a reduced/partial observation of the state. We have $m \leq n$ because the noise U only hits X through GU, so having the dimension of the state transition noise be larger than the state doesn't make sense because what the state sees could be represented in a lower dimension, given that the state only sees the dimension of G's rows.

3.10.1 Computational Complexity of Kalman Filter

We first look at the computational complexity of the regular Kalman Filter, for a baseline to compare our "fast" version to. For the predictor version of the Kalman Filter, we have the following system:

$$e_i = Y_i - H\widehat{X}_{i|i-1} \tag{3.10.7}$$

$$\widehat{X}_{i+1|i} = F\widehat{X}_{i|i-1} + K_i R_{e,i}^{-1} e_i \quad \text{with} \quad \widehat{X}_{0|-1} = 0. \tag{3.10.8}$$

Note here that we use the term $K_i R_{e,i}^{-1}$ in place of $K_{p,i}$ because it will later facilitate our derivation of the fast recursion. Note that K_i is neither the predicted/filtered Kalman gain. We can compute these new terms according to the following formulas, derived during the predicted version Kalman filter:

$$K_i = FP_{i|i-1}H^* + GS \tag{3.10.9}$$

$$R_{e,i} = R + HP_{i|i-1}H^* \tag{3.10.10}$$

$$P_{i+1|i} = FP_{i|i-1}F^* + GQG^* - K_i R_{e,i}^{-1} K_i^* a \quad \text{with} \quad P_{0|-1} = \Pi_0. \tag{3.10.11}$$

Now, in general, the biggest computational complexity is in computing the $P_{i+1|i}$. For example, the first term $FP_{i|i-1}F^*$ consists of three matrix multiplications, each having dimension $n \times n$. With the naive/brute-force computation, the complexity is $O(n^3)$. This computation, in general, is the bottle neck for naive/brute force computation.

3.10.2 Reducing Computational Complexity

The key idea is that *propagating* $P_{i|i-1}$ *is bad.* We define recursions for δP_i instead of $P_{i|i-1}$ directly, which we define as follows:

$$\delta P_i := P_{i+1|i} - P_{i|i-1}, \quad i \in \mathbb{N}_0. \tag{3.10.12}$$

Previously we used the Ricatti equation to go from $P_{i|i-1} \to P_{i+1|i}$, but now we need to know how to go from δP_i to δP_{i+1}. This is addressed in the following result.

Recall that we have defined

$$F_{p,i} := F - K_i R_{e,i}^{-1} H. \tag{3.10.13}$$

We claim that (this is the *Stokes Identity*)

$$\delta P_{i+1} = F_{p,i} \left[\delta P_i - (\delta P_i) H^* R_{e,i+1}^{-1} H (\delta P_i) \right] F_{p,i}^*. \tag{3.10.14}$$

This equation immediately implies that $\mathrm{rank}(\delta P_{i+1}) \leq \mathrm{rank}(\delta P_i)$. This is probably the most significant observation in the derivation of the "fast" algorithm. And now, if we can show that the rank of δP_0 is low rank, then we have the opportunity to eliminate unnecessary computation in this recursion via low rank matrix computation.

Now to derive δP_0, we have the following (where $P_0 = \Pi_0$ and P_1 is computed via the Ricatti equations):

$$\delta P_0 = P_1 - P_0 \tag{3.10.15}$$

$$= F\Pi_0 F^* + GQG^* - K_0 R_{e,0}^{-1} K_0^* - \Pi_0. \tag{3.10.16}$$

Our goal is to claim that this expression is low rank, which we can do by breaking it into a number of cases. Note that in all of these cases (and the entire lecture), we assume that $R > 0$.

Case 1. $\Pi_0 = 0$.

Then

$$\delta P_0 = 0 + GQG^* - K_0 R_{e,0}^{-1} K_0^* + 0 \tag{3.10.17}$$

$$= G\left(Q - SR^{-1}S^*\right)G^*. \tag{3.10.18}$$

Hence $\mathrm{rank}(\delta P_0) \leq \mathrm{rank}(G) \leq m \leq n$, so the rank of δP_0 in this case is small.

Case 2. $(X_n)_{n\in\mathbb{N}_0}$ is a stationary process.

By stationary process we mean that we choose the initial variance Π_0 that satisfies the Lyapunov equation

$$\Pi_0 = F\Pi_0 F^* + GQG^* \tag{3.10.19}$$

which has a PSD solution if all eigenvalues of F lie in the unit disk, $|\lambda_i(F)| < 1$. It's important to note that *initializing at* Π_0 *does not make a Kalman filter a time-invariant filter.* Our Kalman filter still takes time to converge to a particular time-invariant filter. Now, in this case we can easily compute that

$$\delta P_0 = -K_0 R_{e,0}^{-1} K_0^*. \tag{3.10.20}$$

Now note that $K_0 = F\Pi_0 H^* + GS$, and so has dimension $n \times p$. Hence $\mathrm{rank}(\delta P_0) \leq p$.

Suppose now that $\mathrm{rank}(\delta P_0) = r \leq n$. We factorize δP_0 to get

$$\delta P_0 = L_0 M_0 L_0^*, \quad \delta P_i = L_i M_i L_i^* \tag{3.10.21}$$

and we require $L_0 \in \mathbb{C}^{n\times r}$, and $M_0 \in \mathbb{C}^{r\times r}$ is Hermitian, then we have that

$$L_{i+1} = \left(F - K_i R_{e,i}^{-1} H\right) L_i = F_{p,i} L_i \tag{3.10.22}$$

$$M_{i+1} = M_i - M_i L_i H^* R_{e,i+1}^{-1} H L_i M_i, \quad i \in \mathbb{N}_0. \tag{3.10.23}$$

Proving this theorem is quite simple because we can use the Stokes identity.

Given that the rank of $\delta P_0 = r$, we can factorize δP_0 with computational complexity $O\big(n^2 r\big)$, where previously we need $O\big(n^3\big)$. Now for the computational complexity of L_{i+1}, then $F_{p,i}$ has dimension $n \times n$, while L_i has dimension $n \times a$. This results in a complexity of $O\big(n^2 r\big)$. Finally, the complexity of M_{i+1} can be shown to have complexity $O\big(\max\left\{r^3, npr, np^2\right\}\big)$.

Now, after this propagation of δP_i, we compute the full recursion to solidify our understanding.

Algorithm 3.10.1 (CKMS Recursion). **Initialize:**

```
$K_0 \leftarrow F\Pi_0 H^* + GS$
$R_{e,0} \leftarrow R + H\Pi_0 H^*$
$\delta P_0 \leftarrow P_1 - P_0$
for $i \in \mathbb{N}_0$ do
    $\delta P_i \leftarrow L_i M_i L_i^*$
```

$$\begin{aligned}
&P_{i+1|i} \leftarrow P_{i|i-1} + \delta P_i \\
&K_{i+1} \leftarrow K_i + F L_i M_i L_i^* H^* \\
&L_{i+1} \leftarrow F L_i - K_i R_{e,i}^{-1} H L_i \\
&R_{e,i+1} \leftarrow R_{e,i} + H L_i M_i L_i^* H^* \\
&M_{i+1} \leftarrow M_i - M_i L_i H^* R_{e,i+1}^{-1} H L_i M_i
\end{aligned}$$

Here an unrolled recursion for $P_{i+1|i}$ is

$$P_{i+1|i} = P_0 + \sum_{j=0}^{i} \delta P_j = \Pi_0 + \sum_{j=0}^{i} L_j M_j L_j^*. \tag{3.10.24}$$

4 Optimization

4.1 Recursive Least Squares

The Recursive Least Squares (RLS) algorithm is a well-known adaptive filtering algorithm that efficiently updates or “downdate” the least square estimate. We present the algorithm and its connections to Kalman filter in this lecture.

Let’s consider the model

$$Y_i = H_i X + V_i \tag{4.1.1}$$

where

$$\begin{aligned}
Y_i &= \begin{bmatrix} y_0 \\ \vdots \\ y_i \end{bmatrix} \in \mathbb{C}^{i+1} && (4.1.2)\\
H_i &= \begin{bmatrix} h_0^* \\ \vdots \\ h_i^* \end{bmatrix} \in \mathbb{C}^{(i+1)\times n} && (4.1.3)\\
V_i &= \begin{bmatrix} v_0 \\ \vdots \\ v_i \end{bmatrix} \in \mathbb{C}^{i+1} && (4.1.4)\\
X &\in \mathbb{C}^n. && (4.1.5)
\end{aligned}$$

We also assume that

$$\begin{aligned}
\langle X, X\rangle &= \Pi_0 && (4.1.6)\\
\langle V_i, V_i\rangle &= I_{i+1} && (4.1.7)\\
\langle X, v_i\rangle &= 0. && (4.1.8)
\end{aligned}$$

The theory of linear estimation in lecture 2 yields an expression for such optimal estimator at step i:

$$\widehat{X}_i = \left(\Pi_0^{-1} + H_i^* H_i\right)^{-1} H_i^* Y_i. \tag{4.1.9}$$

Suppose we already obtained the estimator $\widehat{X}_{i-1}$ and a new observation y_i arrived. How can we update the estimate to $\widehat{X}_i$? Can we use $\widehat{X}_{i-1}$ to reduce the computational burden?

If we define

$$P_i = \left(\Pi_0^{-1} + H_i^* H_i\right)^{-1} \tag{4.1.10}$$

then the update equation becomes

$$\widehat{X}_i = P_i H_i^* Y_i. \tag{4.1.11}$$

We can write a recurrence relation for P_i. Namely, we claim that

$$\begin{aligned}
P_{-1} &= \Pi_0 && (4.1.12)\\
P_i &= P_{i-1} - \frac{P_{i-1} h_i h_i^* P_{i-1}}{1 + h_i^* P_{i-1} h_i}. && (4.1.13)
\end{aligned}$$

Indeed,

$$\begin{aligned} H_i^* H_i &= H_{i-1}^* H_{i-1} + h_i h_i^* \qquad (4.1.14)\\ \implies P_i &= \left(P_{i-1}^{-1} + h_i h_i^*\right)^{-1} \qquad (4.1.15)\end{aligned}$$

and the Sherman-Morrison-Woodbury identity applied to this case yields the desired result.

Note that using this formula, we also obtain a recursive pattern for the estimates

$$\begin{aligned} \widehat{X}_i &= P_i H_i^* Y_i \qquad (4.1.16)\\ &= \left(P_{i-1} - \frac{P_{i-1} h_i h_i^* P_{i-1}}{1 + h_i^* P_{i-1} h_i}\right)\left(H_{i-1}^* Y_{i-1} + h_i y_i\right) \qquad (4.1.17)\\ &= \widehat{X}_{i-1} - \frac{P_{i-1} h_i h_i^* X_{i-1}}{1 + h_i^* P_{i-1} h_i} + P_{i-1} h_i y_i - \frac{P_{i-1} h_i h_i^* P_{i-1} h_i y_i}{1 + h_i^* P_{i-1} h_i} \qquad (4.1.18)\\ &= \widehat{X}_{i-1} + \frac{P_{i-1} h_i}{1 + h_i^* P_{i-1} h_i}\left(y_i - h_i^* \widehat{X}_{i-1}\right) \qquad (4.1.19)\end{aligned}$$

with $\widehat{X}_{-1} = 0$.

We can formulate this as a Kalman filter algorithm with the state space model

$$\begin{aligned} X_{i+1} &= X_i, \quad X_0 = X \qquad (4.1.20)\\ Y_i &= h_i^* X_i + v_i \qquad (4.1.21)\\ \langle X_0, X_0\rangle &= \Pi_0 \qquad (4.1.22)\\ \langle v_i, v_j\rangle &= \delta_{ij} \qquad (4.1.23)\\ \langle X_0, v_i\rangle &= 0, \quad i \geq 0. \qquad (4.1.24)\end{aligned}$$

At time i, the observation model of the Kalman filter is equivalent to the recursive least squares model, and thus the recursive least squares solution is in fact exactly the same as the Kalman filter solution.

Hence, the RLS algorithm can be viewed as

- A special case of Kalman filter under a particular state-space model.
- A recursive algorithm to solve the optimal linear estimator under the given linear model.
- A recursive algorithm to solve the deterministic least squares problem

$$\min_X \left(X^* \Pi_0^{-1} X + \|Y_i - H_i X\|^2\right). \qquad (4.1.25)$$

Now we consider the opposite problem, which is trying to compute a linear estimator by forgetting some observations. In other words, we want to find the best estimator of X given $Y_{1:i}$ which is the vector Y_i without the first element y_0. Here again, theory provides a closed expression for that optimal linear estimator $\widehat{X}_{1:i}$:

$$\widehat{X}_{1:i} = \left(\Pi_0^{-1} + H_{1:i}^* H_{1:i}\right)^{-1} H_{1:i}^* Y_{1:i}. \qquad (4.1.26)$$

Similarly to the recursive least squares algorithm, one can derive a backward recursive equation that translates the operation of forgetting the first observation.

In particular, if

$$\begin{aligned} P_i &= \left(\Pi_0^{-1} + H_i^* H_i\right)^{-1} \qquad (4.1.27)\\ P_{1:i} &= \left(\Pi_0^{-1} + H_{1:i}^* H_{1:i}\right)^{-1} \qquad (4.1.28)\\ \widehat{X}_i &= P_i H_i^* Y_i \qquad (4.1.29)\\ \widehat{X}_{1:i} &= P_{1:i} H_{1:i}^* Y_{1:i} \qquad (4.1.30)\end{aligned}$$

then we have the following equality:

$$P_{1:i} = P_i - \frac{P_i h_0 h_0^* P_i}{-1 + h_0^* P_i h_0}. \tag{4.1.31}$$

To show this it suffices to see that

$$H_{1:i}^* H_{1:i} = H_i^* H_i - h_0 h_0^* \tag{4.1.32}$$

which yields the desired result using Sherman-Morrison-Woodbury identity.

We also obtain a recursive formula:

$$\widehat{X}_{1:i} = \widehat{X}_i + \frac{P_i h_0}{-1 + h_0^* P_i h_0} \left(y_0 - h_0^* \widehat{X}_i \right). \tag{4.1.33}$$

Note that this time we can't find the analogous state space model from which the Kalman filter derives the expression, as it would imply that the noise v_i has a negative definite covariance matrix ($\langle v_i, v_j \rangle = -\delta_{ij}$).

4.2 Gradient Descent for Least Squares

The motivation is to solve the optimization problem

$$\min_{x \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (y_i - a_i^* x)^2 . \tag{4.2.1}$$

This is least squares and thus has a closed form

$$x = (A^* A)^{-1} A^* y \tag{4.2.2}$$

where

$$A = \begin{bmatrix} a_1^* \\ \vdots \\ a_n^* \end{bmatrix} \in \mathbb{R}^{n \times d}, \quad y = \begin{bmatrix} y_1 \\ \dots \\ y_n \end{bmatrix} \in \mathbb{R}^n. \tag{4.2.3}$$

The cost of a direct method (e.g. QR/LU decomposition) is $O(nd^2)$, with the caveat $n \geq d$. *This is too slow!*

This motivates an iterative method for convex optimization to optimize our objective function. Ideally this method would have small per-iteration complexity, and also able to be distributed.

More generally, suppose we are trying to solve the optimization

$$\min_{x \in \mathbb{R}^d} f(x) \tag{4.2.4}$$

for $f \in C^1(\mathbb{R}^d)$ and convex. Our goal is to analyze the gradient descent method.

Definition 4.2.1. A function $f \colon \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex if

$$f(tx + (1-t)y) \leq t f(x) + (1-t) f(y) - \frac{\mu}{2} \|x - y\|_2^2 , \quad x, y \in \mathbb{R}^d, t \in [0,1]. \tag{4.2.5}$$

Equivalently, if $f \in C^2(\mathbb{R}^d)$, then

$$\nabla^2 f \succeq \mu I_d. \tag{4.2.6}$$

Definition 4.2.2. A function $f \in C^1(\mathbb{R}^d)$ has L-Lipschitz gradient if

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L \|x - y\|_2 , \quad x, y \in \mathbb{R}^d. \tag{4.2.7}$$

Equivalently,

- If $f \in C^2(\mathbb{R}^d)$, then

$$\left\|\nabla^2 f(x)\right\|_{\text{op}} \leq L. \tag{4.2.8}$$

- $$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2} \left\| y - x \right\|_2^2 . \tag{4.2.9}$$

- $$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \frac{1}{L} \left\| \nabla f(x) - \nabla f(y) \right\|_2^2 . \tag{4.2.10}$$

Notice that

$$\sum_{i=1}^{n} (y_i - a_i^* x)^2 = \left\| y - Ax \right\|_2^2 . \tag{4.2.11}$$

So the least squares objective function is

$$f(x) = \frac{1}{n} \left\| y - Ax \right\|_2^2 . \tag{4.2.12}$$

Then f is $\frac{\lambda_{\min}(A^*A)}{n}$-strongly convex and has $\frac{\lambda_{\max}(A^*A)}{n}$-Lipschitz gradient.

Another example is the logistic regression objective function:

$$f(x) = \frac{1}{n} \sum_{i=1}^{n} \log\left(1 + \mathrm{e}^{y_i \alpha_i^* x}\right). \tag{4.2.13}$$

Then $\nabla^2 f(x) = \frac{1}{n} A^* D A$ where $D \succeq 0$ is diagonal. In particular, f is 0-strongly convex (i.e., just convex), and has $\frac{\lambda_{\max}(A^*A)}{n}$-Lipschitz gradient.

The gradient descent algorithm initializes x_0 and then iterates by the rule

$$x_{k+1} = x_k - h_k \nabla f(x_k). \tag{4.2.14}$$

Theorem 4.2.3. Suppose f is convex and has L-Lipschitz gradient. Suppose further that the step sizes $h_k = h < \frac{2}{L}$. Let $f_* = \min_x f(x)$ and let x_* be any x such that $f(x_*) = f_*$. Then

$$\begin{aligned} f(x_k) - f_* &\leq \frac{2(f(x_k) - f_*) \left\| x_0 - x_* \right\|_2}{2 \left\| x_0 - x_* \right\|_2^2 + kh(2 - Lh)(f(x_k) - f_*)} && (4.2.15) \\ &\leq \frac{2L}{k} \left\| x_0 - x_* \right\|_2 \quad \text{if } h = \frac{1}{L}. && (4.2.16) \end{aligned}$$

Theorem 4.2.4. Suppose f is μ-strongly convex and has L-Lipschitz gradient. Suppose further that the step sizes $h_k = h = \frac{1}{\mu + L}$. Let $f_* = \min_x f(x)$ and let x_* be any x such that $f(x_*) = f_*$. Then

$$\begin{aligned} \left\| x_k - x_* \right\|_2^2 &\leq \left(1 - \frac{2h\mu L}{\mu + L}\right)^k \left\| x_0 - x_* \right\|_2^2 && (4.2.17) \\ f(x_k) - f_* &\leq \frac{L}{2} \left(1 - \frac{2h\mu L}{\mu + L}\right)^{2k} \left\| x_0 - x_* \right\|_2^2 . && (4.2.18) \end{aligned}$$

Definition 4.2.5. Suppose f is μ-strongly convex and has L-Lipschitz gradient. Define the **condition number** of f to be

$$\kappa = \frac{L}{\mu} = \frac{\max_x \lambda_{\max}\left(\nabla^2 f(x)\right)}{\min_x \lambda_{\min}\left(\nabla^2 f(x)\right)}. \tag{4.2.19}$$

The convergence rate for gradient descent with a convex function is linear (like $\frac{1}{k} = \epsilon$); the convergence rate for gradient descent with a strongly convex function is exponential (like $(1 - \frac{1}{\kappa})^k = \epsilon$).

Gradient descent is a dynamical system. Let

$$S(x) = x - h \nabla f(x). \tag{4.2.20}$$

Then $x_k = S(S(\cdots S(x_0)))$. To analyze this we construct a Lyapunov function.

Definition 4.2.6. Let $L\colon \mathbb{R}^d \to \mathbb{R}$ be a function. Then L is a **Lyapunov function** of S if $L(S(x)) \leq L(x)$.

Two ways this could be true and lead to easy analysis are

$$L(S(x)) \leq L(x) - \epsilon, \quad \epsilon > 0 \tag{4.2.21}$$

$$L(S(x)) \leq cL(x), \quad c < 1. \tag{4.2.22}$$

Proof of Theorem 4.2.3. Let $\Delta_k = f(x_k) - f_*$ and $\gamma_k = \|x_k - x_*\|_2$. The key inequality that we want to show is that there exists $\epsilon > 0$ with

$$\begin{aligned} -\frac{1}{\Delta_{k+1}} &\leq -\frac{1}{\Delta_k} - \epsilon && (4.2.23)\\ &\leq -\frac{1}{\Delta_0} - \epsilon k. && (4.2.24) \end{aligned}$$

We show this now.

$$\begin{aligned} \Delta_{k+1} &= f(x_{k+1}) - f_* && (4.2.25)\\ &= f(x_k) + \langle \nabla f(x_k), x_{k+1} - x_k \rangle + \frac{L}{2}\|x_{k+1} - x_k\|_2^2 - f_* && (4.2.26)\\ &= \Delta_k + \langle \nabla f(x_k), x_{k+1} - x_k \rangle + \frac{L}{2}\|x_{k+1} - x_k\|_2^2 && (4.2.27)\\ &= \Delta_k - \omega\|\nabla f(x_k)\|_2^2 \quad \text{where } \omega = h\left(1 - \frac{Lh}{2}\right) \quad . && (4.2.28)\\ \Delta_k &= f(x_k) - f(x_*) && (4.2.29)\\ &\leq \langle \nabla f(x_k), x_k - x_* \rangle && (4.2.30)\\ &\leq \|\nabla f(x_k)\|_2 \gamma_k && (4.2.31)\\ &\leq \|\nabla f(x_k)\|_2 \gamma_0 \quad \text{given that } \gamma_\cdot \text{ is non-increasing} && (4.2.32)\\ \implies \Delta_{k+1} &\leq \Delta_k - \omega\|\nabla f(x_k)\|_2^2 && (4.2.33)\\ &\leq \Delta_k - \frac{\omega}{\gamma_0}\Delta_k^2 && (4.2.34) \end{aligned}$$

which implies

$$\frac{1}{\Delta_{k+1}} \geq \frac{1}{\Delta_k} + \frac{w}{\gamma_0^2}\frac{\Delta_k}{\Delta_{k+1}} \geq \frac{1}{\Delta_k} + \frac{\omega}{\gamma_0^2}. \tag{4.2.35}$$

□

We omit the proof of Theorem 4.2.4; it goes similarly.

We now briefly mention stochastic gradient descent. Supposing that n is very large and we cannot compute the whole gradient at once, we cannot run the full gradient descent. That is, we iterate using only a part of the full sample at once, using a small-sample unbiased estimator for the gradient instead of the gradient itself.

More formally, this motivates the construction of a framework for stochastic optimization. Fix a probability space $(\Omega, \mathcal{F}, \mathbf{P})$ and define $F\colon \mathbb{R}^d \times \Omega \to \mathbb{R}$. Let the objective function be $f(x) = \mathbf{E}(F(x,\omega))$ where $\omega \sim \mathbf{P}$. The goal is

$$\min_x f(x) = \min_x \mathbf{E}(F(x,\omega)). \tag{4.2.36}$$

We can treat this using query optimization. Given x, we have access to an oracle that returns $(F(x,\omega), \nabla_x F(x,\omega))$ for a fresh $\omega \sim \mathbf{P}$. It is easy to see that $\mathbf{E}(F(x,\omega)) = f(x)$ and $\mathbf{E}(\nabla_x F(x,\omega)) = \nabla f(x)$.

As an example, for the linear regression and logistic regression objectives,

$$F(x,\omega) = (y - a^* x)^2, \quad F(x,\omega) = \log\left(1 + \mathrm{e}^{ya^*x}\right), \quad \omega = (y,a) \tag{4.2.37}$$

where (y,a) is drawn from the empirical distribution on the collected data.

The stochastic gradient descent algorithm is to draw $\omega_k \sim \mathbf{P}$ at timestep k and update with the rule

$$x_{k+1} = x_k - h_k \nabla_x F(x, \omega_k). \tag{4.2.38}$$

Theorem 4.2.7. Suppose f is convex and $\mathbf{E}\left(\|\nabla_x F(x,\omega)\|_2^2\right) \leq M^2$. Let $f_* = \min_x f(x)$ and x_* be one such minimizer. Let $\overline{x} = \frac{1}{T+1}\sum_{t=0}^{T} x_t$ and the step sizes $h_k = h$. Then

$$\mathbf{E}(f(\overline{x})) - f(x_*) \leq \frac{\|x_0 - x_*\|_2^2}{2h(T+1)} + \frac{hM^2}{2}. \tag{4.2.39}$$

Supposing that $h = \frac{\|x_0 - x_*\|_2}{M\sqrt{T+1}}$, then

$$\mathbf{E}(f(\overline{x})) - f(x_*) \leq \frac{\|x_0 - x_*\|_2 M}{\sqrt{T+1}}. \tag{4.2.40}$$

Theorem 4.2.8. Suppose f is μ-strongly convex and $\mathbf{E}\left(\|\nabla_x F(x,\omega)\|_2^2\right) \leq M^2$. Let $f_* = \min_x f(x)$ and x_* be one such minimizer. Let the step sizes $h_k = h$. Then there is a constant Q with

$$\mathbf{E}\left(\|x_k - x_*\|_2^2\right) \leq \frac{Q}{k+1}. \tag{4.2.41}$$