

Statistical Signal Processing (SSP)

Lecture 1

Statistical Signal Modeling, Learning and Processing

- SSP = data science of signals
- course arrangements
- course overview
- chapter 1: parameter estimation
 - stochastic parameters
 - the multivariate Gaussian distribution; Gauss-Markov theorem
 - the parameter estimation problem
 - Bayes estimation: the MMSE, absolute value and uniform cost functions
 - examples: Gaussian mean in Gaussian noise, Poisson process

Course Arrangements

- course assistants:
Chandan Sheemar, Roya Gholami, Amel Tibhirt, Hoang Minh Le
- theoretical part: exam (counts for 10/20), open notes and handouts
- practical part:
graded parts: copying forbidden, discussing solutions allowed
 - problem sessions (TDs): 2 not graded
 - computer sessions (TPs): 1 (Matlab) graded (counts for 1/6)
⇒ attendance compulsory
 - homeworks (HVs): 2 graded (count for 1/6 each)
- emphasis (in exam) on problem solving, knowing how to apply the theory
⇒ TDs important
- course notes and copies of viewgraphs will appear throughout the term, also on:
web: <http://moodle.eurecom.fr>
lecture recordings: <http://mediaserver.eurecom.fr>

Some Thoughts

- Some student populations target a bare passing grade. However, it is much easier for professors to write a good letter of recommendation for a student with good grades (examples: many former students working at Qualcomm after internship, a student admitted to UC Berkeley with 3 year scholarship upfront).
- If you have been able to get this far in your engineering studies, there is nothing in this course that you would not be able to master.



Course Overview

- Chapter 0: Background Material
- Chapter 1: Parameter Estimation
- Chapter 2: Spectrum Estimation
- Chapter 3: Optimal Filtering
- Chapter 4: Adaptive Filtering
- Chapter 5: Estimation of Sinusoids in Noise
- Chapter 6: Compressed Sensing - Sparse Bayesian Learning
- The techniques introduced in this course have a proven track record of many decades. They are useful for other application branches such as machine learning, in the EURECOM courses **MALIS**, **ASI**, **MALCOM**, **DeepLearning**, or source coding/processing in **Speech**, **ImCod**, **ImProc**, **ImSecu**. Notions from **MathEng**, **InfoTheo**, **Optim** may be useful.



Chapter 0: Background Material

See also the MathEng (Mathematical Methods for Engineers) and Optim (Fundamentals of Optimization) courses and a *Linear Algebra review* on course website

- probability
- linear algebra
- multivariate Gaussian distribution, Gauss-Markov theorem
- complex Gaussian variables, circularity
- vector spaces, inner products, norms
- optimization



Chapter 1: Parameter Estimation

- **random parameters**, Bayesian estimation, minimum mean squared error (MMSE) estimation, maximum a posteriori (MAP) estimation, equivalent estimators
- MMSE and MAP estimation for vector parameters, Cramer-Rao bound, linear MMSE estimators, orthogonality principle, the linear model
- **deterministic parameters**, uniformly minimum variance unbiased estimators (UMVUE), maximum likelihood (ML) estimation, properties of estimators (bias, efficiency, consistency), Cramer-Rao bound
- best linear unbiased estimator (BLUE), least-squares techniques, method of moments, the linear model
- Advanced: model order selection, case of complex measurements and/or parameters, ML optimization techniques (steepest-descent, Gauss-Newton, alternating opt., scoring), Expectation-Maximization (EM) algorithm, Variational Bayesian techniques (VB), Compressed Sensing (sparse), Empirical Bayes



Chapter 2: Spectrum Estimation

- spectrum estimation = parameter estimation when parameters = spectrum
- non-parametric techniques, periodogram, windowing, spectral leakage, averaged periodogram, smoothed periodogram
- parametric random process models, autoregressive (AR) processes, moving average (MA) processes, autoregressive moving average (ARMA) processes
- parametric techniques, linear prediction, autoregressive modeling, maximum entropy, Levinson & Schur algorithms (relation to triangular covariance matrix factorization), lattice filters
- time and frequency domain localization, short-time Fourier transform, QMF, subbands, perfect reconstruction filter banks, wavelet transform, hierarchical signal representation/approximation



Chapter 3: Optimal Filtering

- optimal filtering = Bayesian parameter estimation when parameters = signal
- **Wiener filtering:** unrealizable (non-causal), causal and FIR; lattice/ladder filters
- application to linear and decision-feedback equalization
- some elements from optimization theory, steepest descent algorithm
- linear state-space models
- **Kalman filtering**
developed for space applications, relation to Levinson/Schur algorithms, Chandrasekhar equations
- applications (channel estimation)



Chapter 4: Adaptive Filtering

- adaptive filtering = optimal filtering in absence of statistical knowledge
- adaptive FIR filtering
- least-mean-square (LMS) algorithm (Stochastic Gradient Descent - SGD)
- recursive least-squares (RLS) algorithm
- tracking of time-varying parameters, performance analysis
- applications



Chapter 5: Estimation of Sinusoids in Noise

- special focus on the estimation of the sinusoid frequencies
- application of deterministic parameter estimation techniques:
 - maximum likelihood (ML): IQML (Iterative Quadratic ML), DIQML (De-noised IQML), PQML (Pseudo QML)
 - subspace fitting (method of moments)
 - linear prediction: Prony's method, Pisarenko's method
 - linear constrained minimum variance (LCMV): Capon's method
- application of adaptive filtering:
 - adaptive notch filtering

Ch 6: Compressed Sensing - Sparse Bayesian Learning

- Compressed sensing:
 - from underdetermined systems to overdetermined systems via sparsity
 - recovery guarantees
 - popular approaches: LASSO, Orthogonal Matching Pursuit (OMP), Iterative (Soft/Hard) Thresholding (IST/IHT)
- Sparse Bayesian Learning
 - empirical Bayes, hierarchical Bayes, non-Gaussian priors via Gaussian Scale Mixtures
 - Expectation Maximization (EM), Variational Bayes (VB)
 - Space Alternating EM (SAGE) & VB (SAVE)



Bibliography

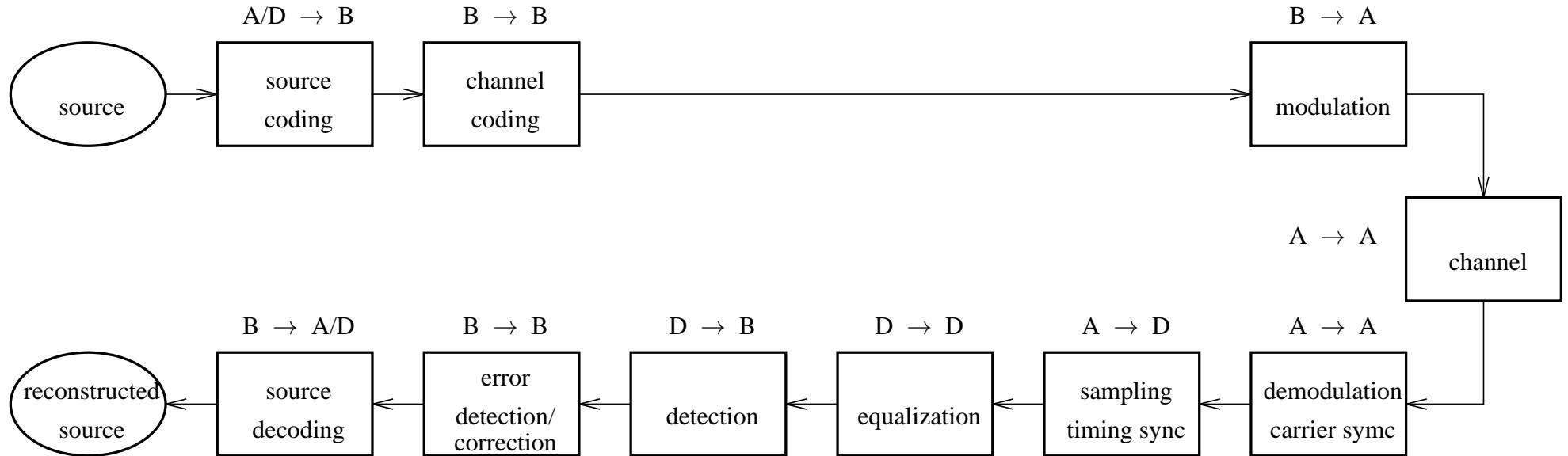
- [1] H.L. Van Trees. *Detection, Estimation and Modulation Theory*, volume 1. Wiley, New York, 1968.
- [2] L. Scharf. *Statistical Signal Processing*. Addison-Wesley, Reading, MA, 1991.
- [3] S.M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.
- [4] B. Porat. *Digital Processing of Random Signals: Theory and Methods*. Prentice Hall, 1994.
- [5] C.W. Therrien. *Discrete Random Signals and Statistical Signal Processing*. Prentice Hall, 1992.
- [6] M.H. Hayes *Statistical Digital Signal Processing and Modeling* Wiley, 1996. Pdf on internet.
- [7] S. Kay *Modern Spectral Estimation: Theory and Application* Prentice Hall, 1988.
- [8] P. Stoica, R. Moses *Spectral Analysis of Signals*. Prentice hall, 2005.
<http://user.it.uu.se/~ps/SAS-new.pdf>
- [9] T. Kailath. *Lectures on Wiener and Kalman Filtering*. Springer-Verlag, Wien – New York, 1981.
- [10] T. Kailath, A.H. Sayed, B. Hassibi *Linear Estimation* Prentice Hall, 2000.
- [11] A.H. Sayed *Adaptive Filters* Wiley-IEEE Press, 2008.



DSP Relevance to Eurecom

- leading thread in DSP course: *statistical signal processing*. Signals as realizations of stochastic processes.
- key building block in telecommunications: *modem*
 - key issue: bit detection: Digital Communications course
 - carrier synchronization: recovery of the carrier frequency and phase
 - timing synchronization: recovery of baud rate and phase
 - channel estimation (or its inverse) for equalization, echo or interference cancellation

MODEM Building Blocks

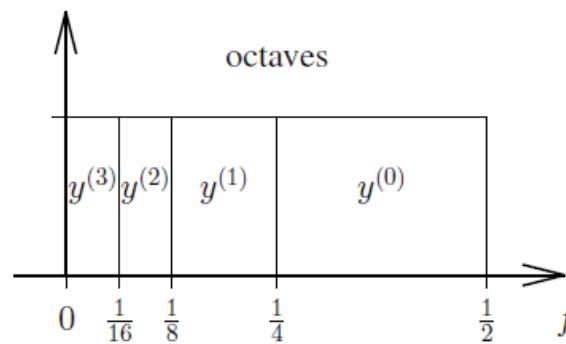
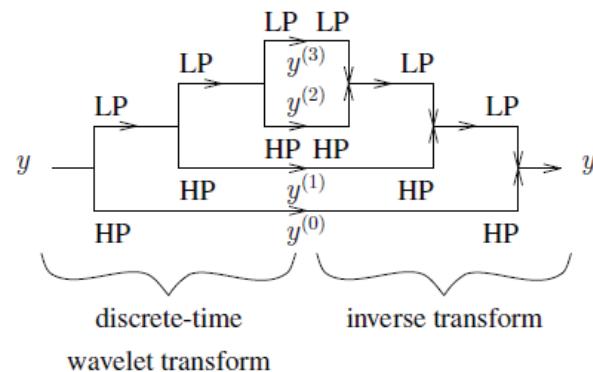
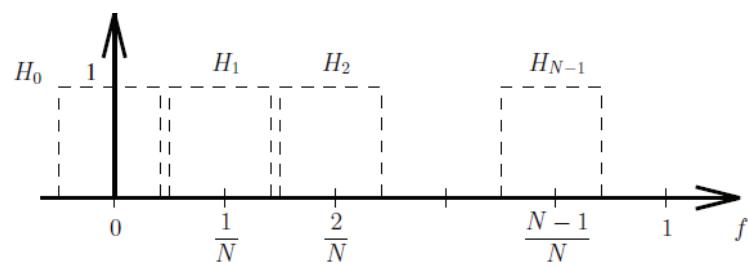
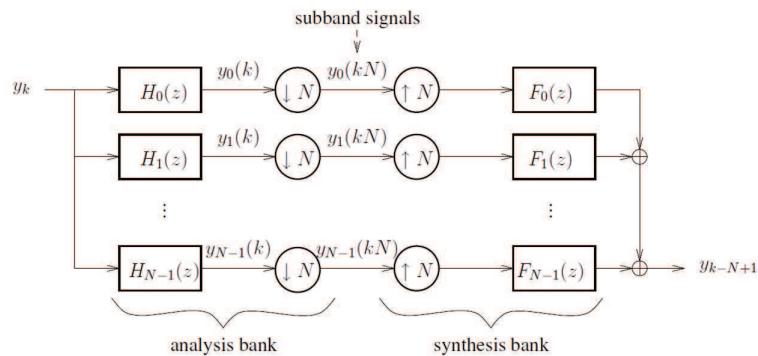


- modem: modulation-demodulation: digital transmission over analog medium
- signals appear in analog (A), digital (D) or binary/coded (B) form
- DigiCom: channel (de)coding, (de)modulation, sampling, detection, equalization
- SSP: source (de)coding, equalization/channel estimation, parameter estimation for timing and carrier synchronization

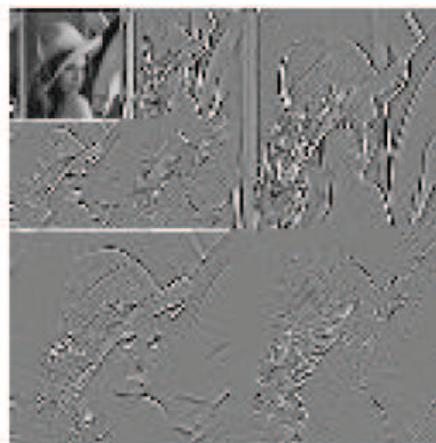
Applications in Corporate Communications

- source compression: (e.g. Unix commands compress and gzip), appetite for information steadily increasing (multimedia)
- hierarchical image compression: nested structure: information at a higher definition level consists of the information at a lower level plus complementary information.
 - result: info split up into streams corresponding to the details at various levels of definition
 - interaction with the network: send different streams separately, with different levels of transmission quality (success of arrival, delay,...)
 - consequences for security: not all streams need to be encrypted (to the same extent)
- communication network performance analysis: queuing theory. parameterized statistical models (e.g. Poisson processes for arrival of packets) need an estimation of their parameter values.
E.g. Amadeus internship: prediction of # reservations/day.

Filterbanks - Wavelets - Hierarchical Images



$A_{2-2}f$	D_{2-2}^1f	D_{2-1}^1f
D_{2-2}^2f	D_{2-2}^3f	
D_{2-1}^2f		D_{2-1}^3f

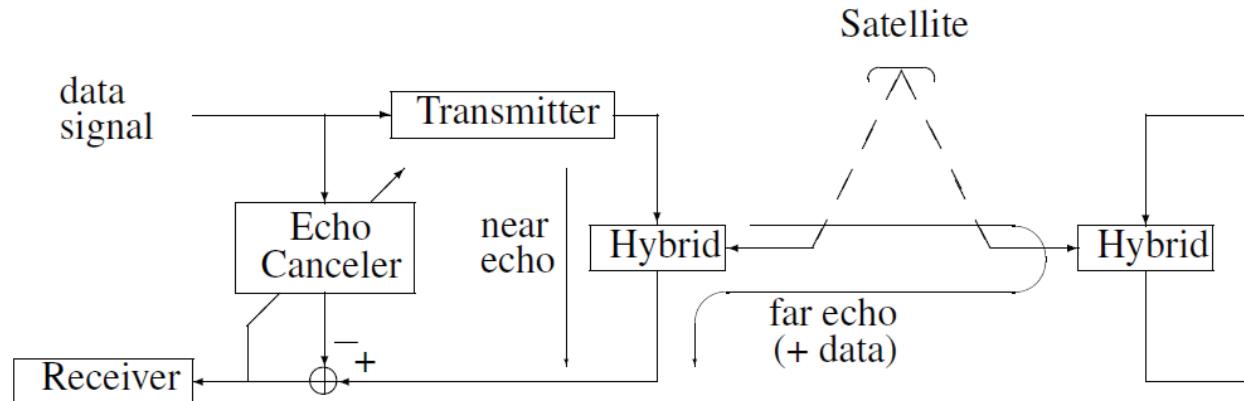




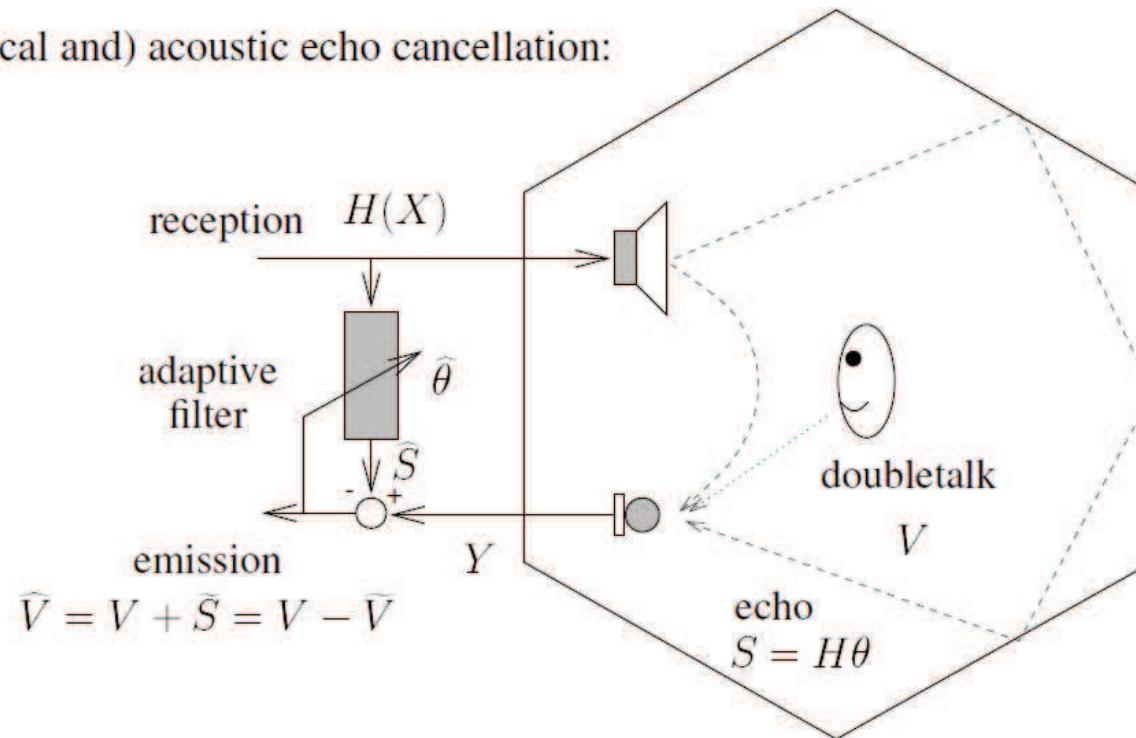
Applications in Multimedia Communications

- lossy source coding for audio and video as opposed to lossless compression of data files
- videoconferencing systems
 - adaptive filters for acoustic echo cancellation (marketing studies: audio more important than video)
 - parametric face models (low bitrate TX via facial cloning), face recognition (biometry), gesture recognition, etc.
- speech recognition for voice-controlled operation: hampered by background noise (other speakers etc.). solution: speech enhancement and directive acoustic microphone arrays
- synchronization of multiple information streams (audio & video & ...)
- transmission of continuous sources (speech, video) over packet-mode networks (internet), (packet) error concealment
- xDSL (Digital Subscriber Loop): high-speed connectivity over the last mile

Electrical and Acoustic Echo Cancellation



(electrical and) acoustic echo cancellation:

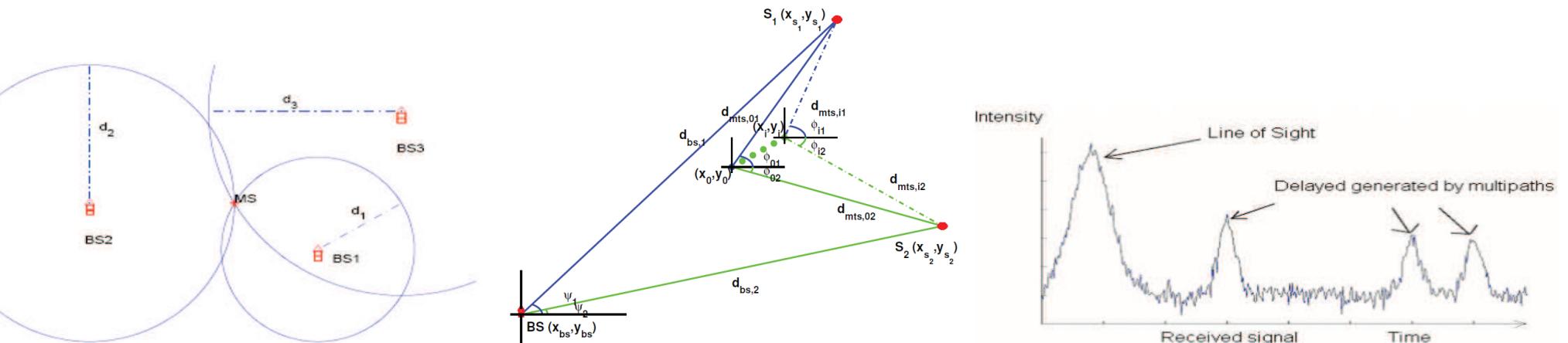




Applications in Mobile Communications

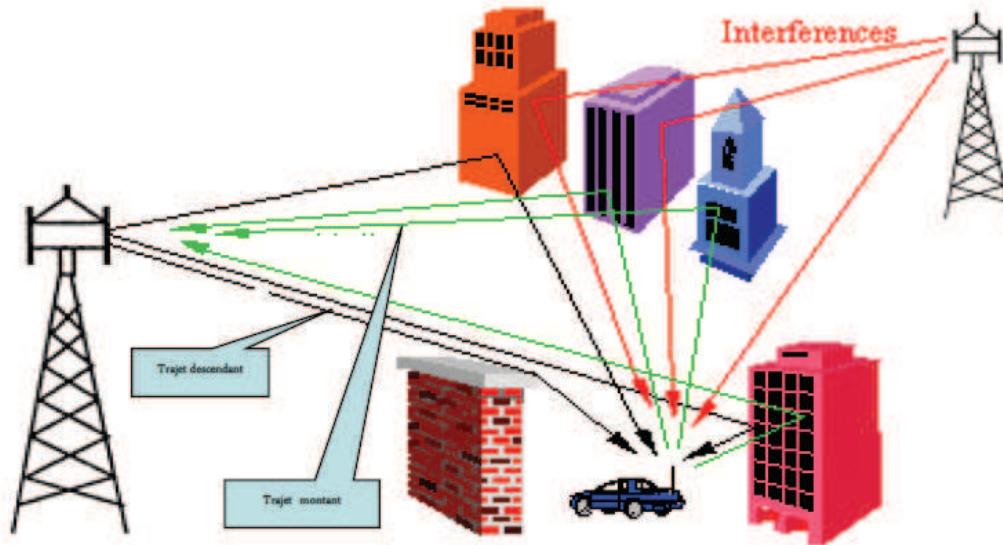
- digital mobile communications more reliable and flexible than analog systems: requires voice coding; very limited bandwidth \Rightarrow sophisticated low bit rate coding techniques required; changing environment \Rightarrow adaptive multi-rate coding.
- National regulations prohibiting simultaneous driving and calling \Rightarrow handsfree telephone systems \Rightarrow audio conferencing systems and acoustic echos problem. Handsfree talking and dialing: speech recognition techniques in a noisy environment
- multipath propagation in a mobile environment: fast channel equalization techniques needed, possibly of limited complexity, carrier and symbol rate synchronization, Doppler and fading tracking
- user localization (911): position estimation on basis of delay (ToA) and/or direction (DoA) estimates (Time/Direction of Arrival)
- cellular communications and frequency reuse: communications limited, not by noise, but by other users (interferers). Sophisticated antenna array processing permits Spatial Division Multiple Access (SDMA): simultaneous interference reduction and multipath reduction, DoA estimation; spatio-temporal processing; interference reduction in Code DMA (CDMA) systems.

Mobile Terminal Location Estimation

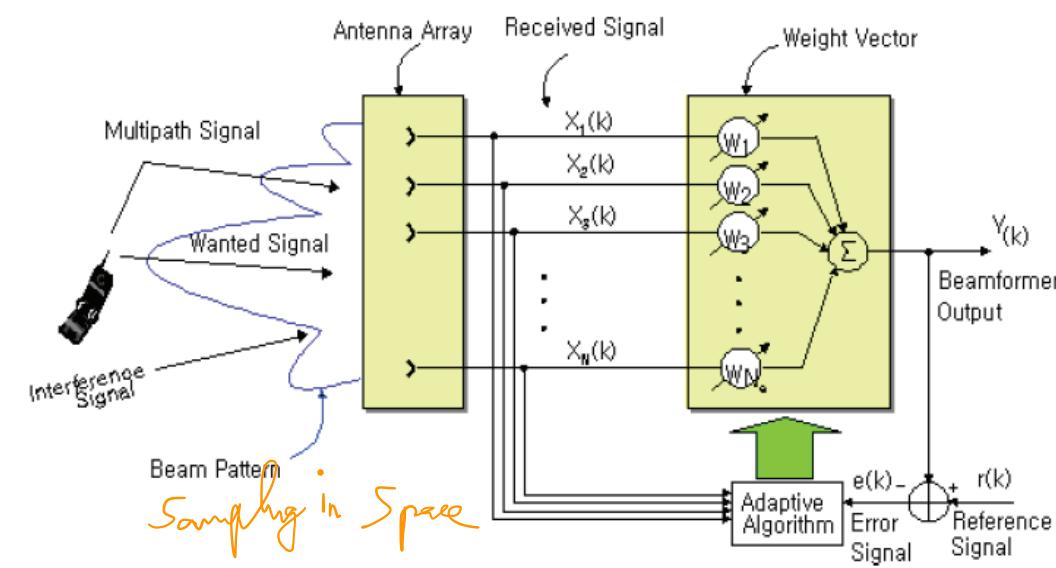


- Time of Arrival (ToA) - pseudo range - triangulation
cheap but imprecise alternative: received signal strength (RSS)
- Non Line of Sight (NLoS), multipath Power Delay Profile (PDP) = location specific **fingerprint**
- NLoS: (dynamic) **single bounce** propagation model: estimate position of mobile (trajectory) + scatterers; recently extended to multi-bounce between known walls
- Multi-Input Multi-Output (**MIMO**) propagation channel model
- multiple antennas: allow estimation of Direction of Arrival (DoA), requires known MS antenna array orientation; alternatively get DoAs from Doppler shifts and speed vector

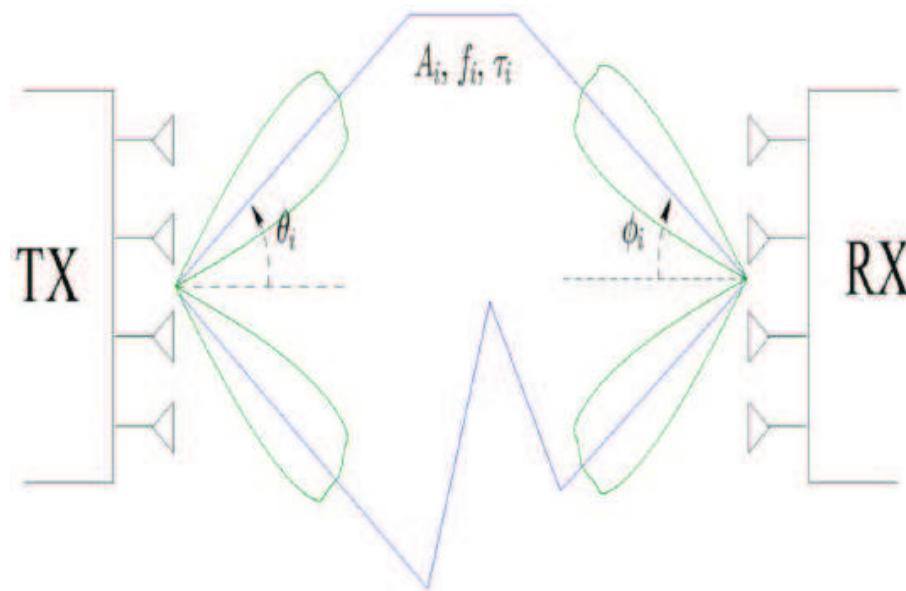
Interfering Cells - Smart Antennas



- spatial filtering \Leftrightarrow FIR filtering
- spatial vs. spatiotemporal
- multi-antenna on both Tx and Rx side: joint transmit/receive
"zero-forcing" feasibility still unknown!
("interference alignment")



Multi Input Multi Output (MIMO) Communications



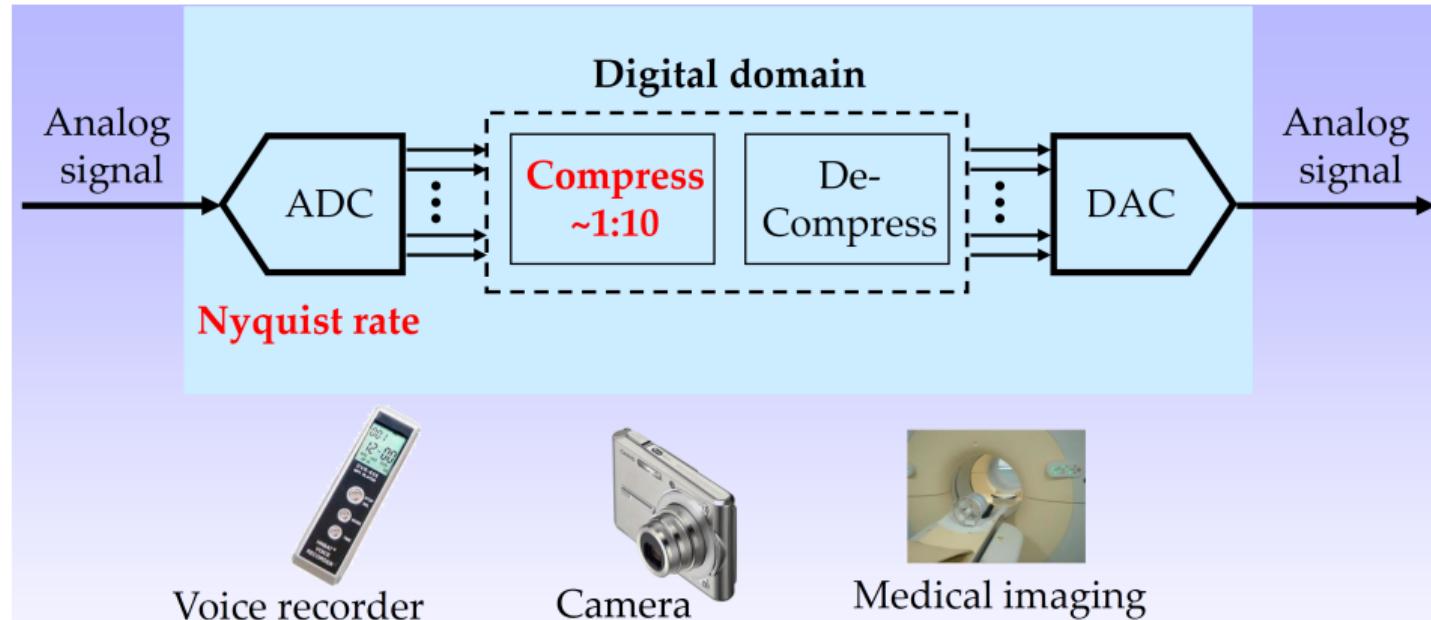
- Narrowband assumption (modulated carrier \approx sinusoid): time delay differences between antennas translate into phase offsets: "antenna array response" vector
- N antennas on both sides \Rightarrow matrix channel response: can send multiple (vector) signals: "spatial multiplexing"
- Requires channel matrix to be invertible. But rank 1 contribution per path. For full rank channel, need $\geq N$ paths with different directions at Tx and Rx. LoS channel: rank 1 matrix only!
- $N \times N$ potential sources of diversity (different fadings): requires "space-time coding" to make every transmitted bit pass thru each antenna.
"Diversity Multiplexing Tradeoff" (DMT).



Other Applications

- audio signal processing
 - dereverberation
 - signal separation
 - music transcription
- navigation, position tracking
- (stock market, economics, weather ...) prediction
- "inverse problems":
reconstruction of 3D objects from 2D images (tomography,...)
- plenty of more applications in "big data" analysis

Compressed Sensing



Compressed Sensing

In some applications, measurements are costly:

- Magnetic resonance imaging:
 - scan time \approx 30 minutes
 - scan time proportional to # samples taken

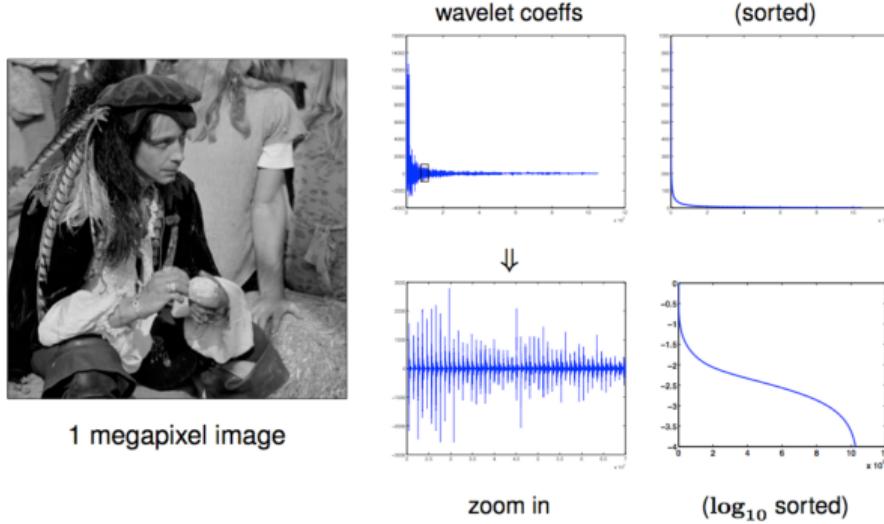


- Imaging outside visible spectrum:
 - CMOS doesn't work
 - high cost per pixel
- Wireless communication:
 - pilots inserted to measure channel
 - more pilots means less payload



Sparsity

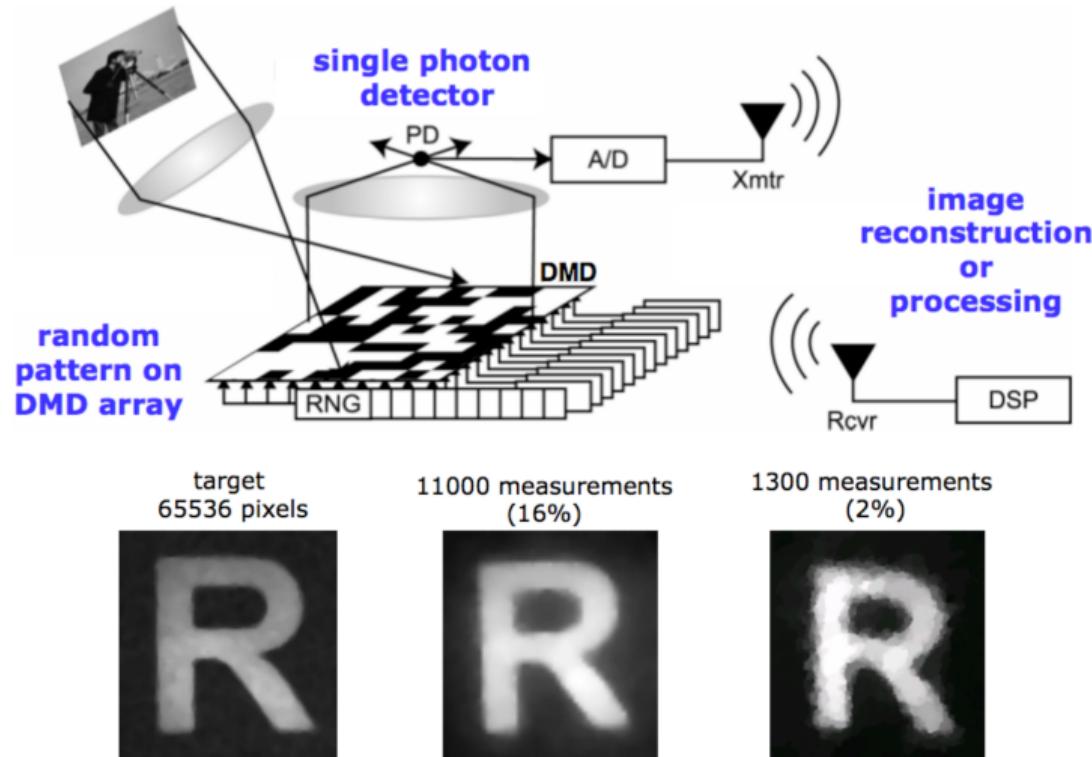
- Many real-world signals are approximately sparse in a known basis.
- For example, natural images are sparse in the discrete wavelet transform (DWT) basis:



Typically: 99% signal energy captured by only 2.5% of DWT coefficients!

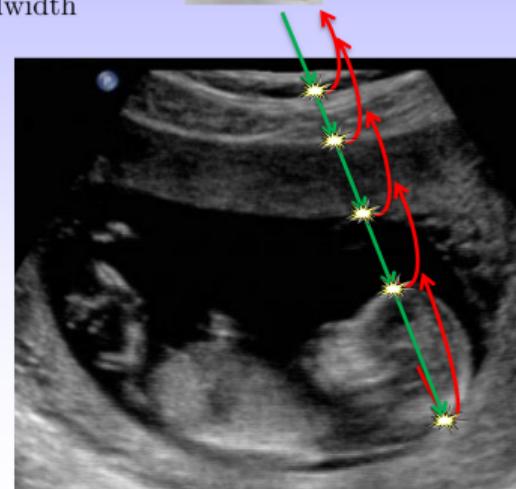
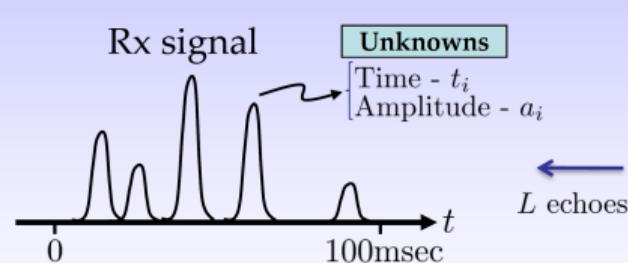
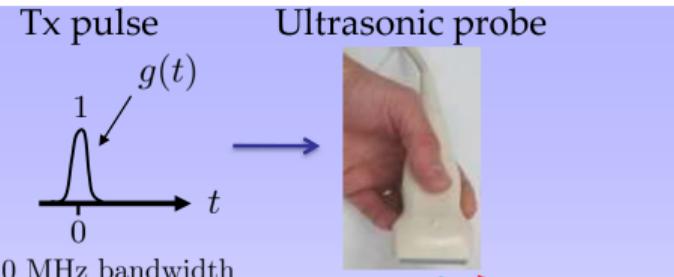
Sparsity will be captured by prior information in a Bayesian approach.

Single Pixel Camera (Rice U.)



Ultra-Sound Imaging

- High sampling rates
- High digital processing rates

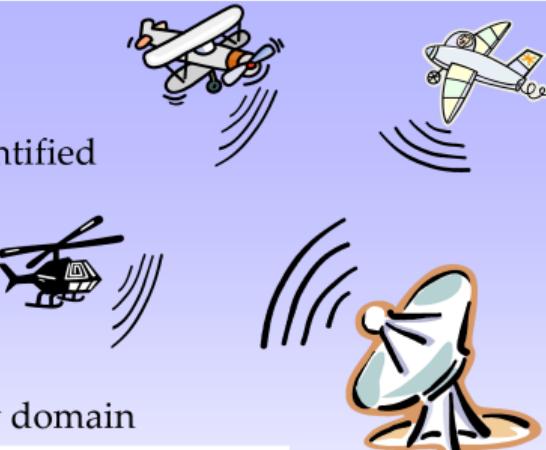


- Echoes result from scattering in the tissue
- The image is formed by identifying the scatterers

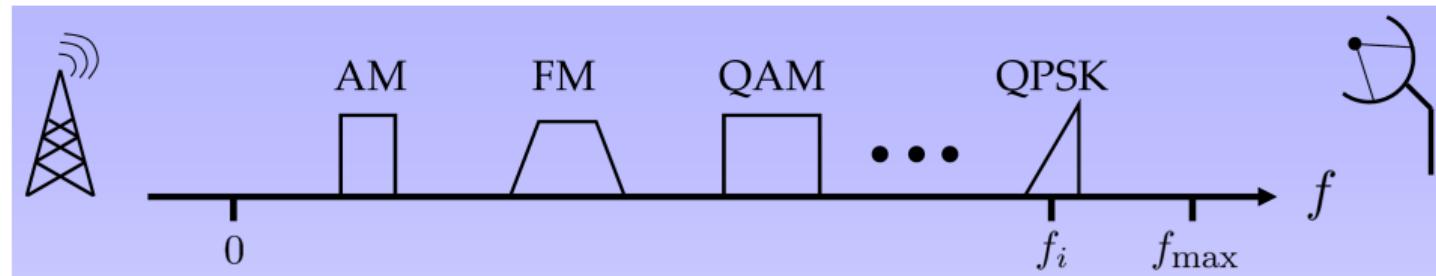
Radar

- Principle:
 - A known pulse is transmitted
 - Reflections from targets are received
 - Target's ranges and velocities are identified

 - Challenge:
 - All processing is done digitally
 - Targets can lie on an arbitrary grid
 - Process of digitizing
→ loss of resolution in range-velocity domain

 - Subspace methods:
- 
- The diagram illustrates various radar configurations. At the top left, two aircraft are shown with their respective signal waveforms. Below them, a helicopter and a ground-based satellite dish are also depicted with their signal patterns. At the bottom, a 2D heatmap plot shows the relationship between Delay ($\times \tau_{\max}$) on the x-axis and Doppler ($\times v_{\max}$) on the y-axis. The plot features several data points: red 'x' marks for 'True Targets' and blue open circles for 'MF peaks'. A color scale bar on the right indicates intensity from 0.2 (blue) to 1.0 (red). The plot shows a central bright peak at approximately (0.5, 0).

Cognitive Radio



- The spectrum occupation by legacy/primary users is sparse.
- Unlicensed secondary users can insert in spectral holes.
- However, have to find the spectral holes, or the occupied spectrum portions.
- Generalized Nyquist says that one can sample at a rate exceeding the sum of the bandwidth, however here also (only) the spectral support needs to be estimated.

The Multivariate Gaussian Distribution

- $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_m]^T$: m jointly Gaussian random variables: $\mathbf{X} \sim \mathcal{N}(m_X, C_{XX})$ with mean

$$m_X = E\mathbf{X} = \int dx_1 \cdots \int dx_m f_{\mathbf{X}}(X) X$$

$$= \begin{bmatrix} \int dx_1 \cdots \int dx_m f_{\mathbf{X}}(X) x_1 \\ \vdots \\ \int dx_1 \cdots \int dx_m f_{\mathbf{X}}(X) x_m \end{bmatrix} = \begin{bmatrix} \int dx_1 x_1 \underbrace{\int dx_2 \cdots \int dx_m f_{\mathbf{X}}(X)}_{f_{\mathbf{X}_1}(x_1)} \\ \vdots \\ \int dx_m x_m \underbrace{\int dx_1 \cdots \int dx_{m-1} f_{\mathbf{X}}(X)}_{f_{\mathbf{X}_m}(x_m)} \end{bmatrix} = \begin{bmatrix} m_{x_1} \\ \vdots \\ m_{x_m} \end{bmatrix}$$

and covariance matrix

$$C_{XX} = E(\mathbf{X} - m_X)(\mathbf{X} - m_X)^T = \int dx_1 \cdots \int dx_m f_{\mathbf{X}}(X) \underbrace{(X - m_X)(X - m_X)^T}_{\geq 0, \text{ rank } 1}$$

where $C_{x_i x_j} = E(\mathbf{x}_i - m_{x_i})(\mathbf{x}_j - m_{x_j})$. As weighted average of positive semi-definite matrices: $C_{XX} = C_{XX}^T \geq 0$ symmetric and positive semidefinite:
 $\forall U \in \mathcal{R}^m : U^T C_{XX} U = U^T E(\mathbf{X} - m_X)(\mathbf{X} - m_X)^T U = E(U^T (\mathbf{X} - m_X))^2 \geq 0$

- joint Gaussian probability density function (pdf):

$$f_{\mathbf{X}}(X) = (2\pi)^{-\frac{m}{2}} (\det C_{XX})^{-\frac{1}{2}} \exp\left(-\frac{1}{2} [X - m_X]^T C_{XX}^{-1} [X - m_X]\right)$$

Multivariate Gaussian Derivation

- goal: derive multivariate Gaussian distribution from univariate Gaussian distribution and two postulates
- $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_m]^T$ real random vector with specified mean and covariance matrix:
with mean $m_X = E\mathbf{X} = [m_{x_1} \cdots m_{x_m}]^T$ and covariance matrix

$$\begin{aligned} C_{XX} &= E(\mathbf{X} - m_X)(\mathbf{X} - m_X)^T = \left[E(\mathbf{x}_i - m_{x_i})(\mathbf{x}_j - m_{x_j}) \right]_{i,j=1}^m \\ &= E(\mathbf{X}\mathbf{X}^T - \mathbf{X}m_X^T - m_X\mathbf{X}^T + m_Xm_X^T) \\ &= (E\mathbf{X}\mathbf{X}^T) - (E\mathbf{X})m_X^T - m_X(E\mathbf{X})^T + m_Xm_X^T \\ &= R_{XX} - m_Xm_X^T - m_Xm_X^T + m_Xm_X^T = R_{XX} - m_Xm_X^T \end{aligned}$$

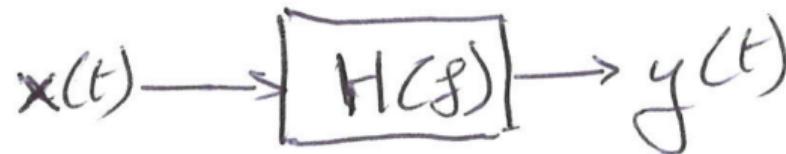
$R_{XX} = E\mathbf{X}\mathbf{X}^T$ = correlation matrix,
linearity of expectation E exploited (linear operations commute)

- by definition of expectation

$$C_{XX} = E(\mathbf{X} - m_X)(\mathbf{X} - m_X)^T = \int dx_1 \cdots \int dx_m f_{\mathbf{X}}(X) (X - m_X)(X - m_X)^T .$$

$$\Rightarrow C_{XX} = C_{XX}^T \text{ symmetric}$$

Eigen Functions/Values of a Linear Time-Invariant System



- Linear Time-Invariant (LTI) system in continuous time, transfer function $H(f)$.
- The input-output relation in the frequency domain: $Y(f) = H(f)X(f)$
where $X(f)$, $Y(f)$ are the Fourier transforms of the time domain signals $x(t)$, $y(t)$.
- Consider a particular input and corresponding output:

$$\begin{aligned} x(t) &= e^{j2\pi f t} \\ \Rightarrow y(t) &= H(f) e^{j2\pi f t} \end{aligned}$$

Complex sinusoids (cisoids) are the eigen functions of LTI systems, with corresponding eigen value $H(f)$, the transfer function at the corresponding frequency f .



Eigendecomposition Covariance Matrix

- eigenvalues λ_i and corresponding eigenvectors V_i of C_{XX} : $C_{XX} V_i = \lambda_i V_i$
fix norm $\|V_i\| = 1$, $\|V_i\|^2 = V_i^T V_i$
- $(C_{XX} - \lambda_i I_m) V_i = 0 \Rightarrow (C_{XX} - \lambda_i I_m)$ singular
 λ_i solution of $\det(C_{XX} - \lambda I_m) = 0$: characteristic equation
- $C_{XX} = C_{XX}^T \Rightarrow \lambda_i \in \mathbb{R}$, $V_i^T V_j = \delta_{ij}$, $i, j = 1, \dots, n$

Kronecker delta : $\delta_{ij} = \begin{cases} 1 & , i = j \\ 0 & , i \neq j \end{cases}$

- matrix $V = [V_1 \cdots V_m]$ ($m \times m$) :
 $[V^T V]_{ij} = V_i^T V_j = \delta_{ij} \Rightarrow V^T V = I_m \quad V = \text{orthogonal matrix}$
- C_{XX} real, λ_i real $\Rightarrow V_i$ can be chosen to be real
- $I_m = V^T V \Rightarrow 1 = \det(I_m) = \det(V^T V) = \det(V^T) \det(V) = (\det V)^2$
 $\Rightarrow \det V = \pm 1$. We can choose the signs of the V_i such that $\det V = 1$.

Eigendecomposition Covariance Matrix (2)

- $C_{XX} \geq 0$ positive semidefinite:

$$\forall U \in \mathcal{R}^m : U^T C_{XX} U = E(U^T (\mathbf{X} - m_X))^2 \geq 0$$

(positive definite would be: $\forall U \in \mathcal{R}^m \setminus \{0\} : U^T C_{XX} U > 0$)

- $U = V_i : U^T C_{XX} U = V_i^T C_{XX} V_i = \lambda_i V_i^T V_i = \lambda_i \geq 0$
- order the $\lambda_i : \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m \geq 0$
- If $\lambda_m = 0$, C_{XX} is singular $\Rightarrow V_m^T C_{XX} V_m = E(V_m^T (\mathbf{X} - m_X))^2 = 0$ mean and variance of $V_m^T (\mathbf{X} - m_X)$ are zero $\Rightarrow V_m^T (\mathbf{X} - m_X) = 0$ in mean square. This means that at least one variable x_i is a linear combination of the other variables and 1. We shall in general exclude this possibility $\Rightarrow C_{XX} > 0$, $\lambda_i > 0$, $i = 1, \dots, m$
- $C_{XX} V_i - V_i \lambda_i = 0$ are the columns of the matrix $C_{XX} V - V \Lambda = 0$ where $\Lambda = \text{diag}\{\lambda_1, \dots, \lambda_m\}$. Using $V^{-1} = V^T$, we find

$$C_{XX} = V \Lambda V^T = [V_1 \cdots V_m] \text{diag}\{\lambda_1, \dots, \lambda_m\} [V_1 \cdots V_m]^T = \sum_{i=1}^m \lambda_i V_i V_i^T$$

Can show: $C_{XX}^\alpha = V \Lambda^\alpha V^T$ for any α (e.g. $\alpha = 2, -1$)

- $\det C = \prod_{i=1}^m \lambda_i$ = volume of parallelopiped spanned by columns of C



Multivariate Gaussian Derivation (2)

- *Postulate* that a linear transformation of jointly Gaussian random variables produces again jointly Gaussian random variables.
- Consider now a linear transformation of variables: $\mathbf{Z} = V^T(\mathbf{X} - m_X)$.
 \mathbf{X} is Gaussian $\Leftrightarrow \mathbf{Z}$ is Gaussian. Then we find for the first two moments

$$\begin{aligned}m_Z &= E V^T(\mathbf{X} - m_X) = V^T(E\mathbf{X} - m_X) = V^T(m_X - m_X) = 0 \\C_{ZZ} &= E (\mathbf{Z} - m_Z)(\mathbf{Z} - m_Z)^T = E\mathbf{Z}\mathbf{Z}^T = EV^T(\mathbf{X} - m_X)(\mathbf{X} - m_X)^TV \\&= V^T \left(E(\mathbf{X} - m_X)(\mathbf{X} - m_X)^T \right) V = V^T C_{XX} V = V^T V \Lambda V^T V = \Lambda\end{aligned}$$

Or hence $E\mathbf{z}_i\mathbf{z}_j = \lambda_i\delta_{ij}$: the \mathbf{z}_i are zero mean and uncorrelated.

- At this point: only specified the first two moments of the \mathbf{z}_i . Now specify the rest of their distribution by stating that the \mathbf{z}_i are jointly Gaussian. We furthermore *postulate* that zero mean uncorrelated Gaussian random variables are independent.

Note that in general $\begin{cases} \text{independent} \\ \text{zero mean} \end{cases} \xrightarrow{\quad} \begin{cases} \text{uncorrelated} \\ \text{zero mean} \end{cases}$



Multivariate Gaussian Derivation (3)

- joint distribution of the independent Gaussian r.v.'s \mathbf{z}_i :

$$f_{\mathbf{Z}}(Z) = \prod_{i=1}^m f_{\mathbf{z}_i}(z_i) = \prod_{i=1}^m \frac{1}{\sqrt{2\pi \lambda_i}} \exp\left(-\frac{z_i^2}{2\lambda_i}\right) = (2\pi)^{-\frac{m}{2}} (\det \Lambda)^{-\frac{1}{2}} \exp\left(-\frac{1}{2} Z^T \Lambda^{-1} Z\right)$$

- Since the Jacobian $(\det V^T)$ of the linear transformation between \mathbf{X} and \mathbf{Z} equals one, we get for the joint distribution of the \mathbf{x}_i $Z = V^T(X - m_X)$

$$\begin{aligned} f_{\mathbf{X}}(X) &= f_{\mathbf{Z}}(V^T(X - m_X)) \\ &= (2\pi)^{-\frac{m}{2}} (\det(V^T C_{XX} V))^{-\frac{1}{2}} \exp\left(-\frac{1}{2} [V^T(X - m_X)]^T \Lambda^{-1} [V^T(X - m_X)]\right) \\ &= (2\pi)^{-\frac{m}{2}} ((\det V^T)(\det C_{XX})(\det V))^{-\frac{1}{2}} \exp\left(-\frac{1}{2} [X - m_X]^T V \Lambda^{-1} V^T [X - m_X]\right) \\ &= (2\pi)^{-\frac{m}{2}} (\det C_{XX})^{-\frac{1}{2}} \exp\left(-\frac{1}{2} [X - m_X]^T C_{XX}^{-1} [X - m_X]\right) \end{aligned}$$

This is the general expression for a multivariate Gaussian probability density function (pdf). Notation: $\mathbf{X} \sim \mathcal{N}(m_X, C_{XX})$: completely specified in terms of the first and second-order moments.

The Multivariate Gaussian Distribution (2)

- $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_m]^T$ and $\mathbf{Y} = [\mathbf{y}_1 \cdots \mathbf{y}_n]^T$: $m+n$ jointly Gaussian random variables:

$$\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m_X \\ m_Y \end{bmatrix}, \begin{bmatrix} C_{XX} & C_{XY} \\ C_{YX} & C_{YY} \end{bmatrix}\right)$$

with mean and covariance matrix

$$\begin{bmatrix} m_X \\ m_Y \end{bmatrix} = E\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix}, \quad C = \begin{bmatrix} C_{XX} & C_{XY} \\ C_{YX} & C_{YY} \end{bmatrix} = E\begin{bmatrix} \mathbf{X} - m_X \\ \mathbf{Y} - m_Y \end{bmatrix}\begin{bmatrix} \mathbf{X} - m_X \\ \mathbf{Y} - m_Y \end{bmatrix}^T$$

- joint Gaussian probability density function (pdf): $f_{\mathbf{X}, \mathbf{Y}}(X, Y)$

$$= (2\pi)^{-\frac{m+n}{2}} (\det C)^{-\frac{1}{2}} \exp\left(-\frac{1}{2} \begin{bmatrix} X - m_X \\ Y - m_Y \end{bmatrix}^T \begin{bmatrix} C_{XX} & C_{XY} \\ C_{YX} & C_{YY} \end{bmatrix}^{-1} \begin{bmatrix} X - m_X \\ Y - m_Y \end{bmatrix}\right)$$

- meaning of the joint pdf:

$$\Pr[\mathbf{X} \in (X, X + dX), \mathbf{Y} \in (Y, Y + dY)] = f_{\mathbf{X}, \mathbf{Y}}(X, Y) dX dY$$

where first $dX = [dx_1 \cdots dx_m]^T$ and then $dX = dx_1 dx_2 \cdots dx_m$



The Conditional Gaussian Distribution

- the conditional pdf $f_{\mathbf{X}|\mathbf{Y}}(X|Y)$ means

$$\Pr [\mathbf{X} \in (X, X + dX) | \mathbf{Y} = Y] = f_{\mathbf{X}|\mathbf{Y}}(X|Y)dX$$

- the conditional pdf is defined by Bayes' rule:

$$f_{\mathbf{X}|\mathbf{Y}}(X|Y) = \frac{f_{\mathbf{X},\mathbf{Y}}(X,Y)}{f_{\mathbf{Y}}(Y)} \Leftrightarrow f_{\mathbf{X},\mathbf{Y}}(X,Y) = f_{\mathbf{Y}}(Y) f_{\mathbf{X}|\mathbf{Y}}(X|Y)$$

get the marginal pdf $f_{\mathbf{Y}}(Y)$ from the joint pdf by integrating out X :

$$\begin{aligned} f_{\mathbf{Y}}(Y) &= \int f_{\mathbf{X},\mathbf{Y}}(X,Y)dX \\ &= \int \cdots \int f_{\mathbf{x}_1, \dots, \mathbf{x}_m, \mathbf{y}_1, \dots, \mathbf{y}_n}(x_1, \dots, x_m, y_1, \dots, y_n) dx_1 \cdots dx_m \\ &= (2\pi)^{-\frac{n}{2}} (\det C_{YY})^{-\frac{1}{2}} \exp \left(-\frac{1}{2} [Y - m_Y]^T C_{YY}^{-1} [Y - m_Y] \right) \end{aligned}$$

- consider the block Upper Diagonal Lower (UDL) triangular factorization of C :

$$C = \begin{bmatrix} C_{XX} & C_{XY} \\ C_{YX} & C_{YY} \end{bmatrix} = \begin{bmatrix} I & K \\ 0 & I \end{bmatrix} \begin{bmatrix} P & 0 \\ 0 & C_{YY} \end{bmatrix} \begin{bmatrix} I & 0 \\ K^T & I \end{bmatrix}$$

with $K = C_{XY}C_{YY}^{-1}$, $P = C_{XX} - C_{XY}C_{YY}^{-1}C_{YX}$ (*Schur complement*)

The Conditional Gaussian Distribution (2)

- from the UDL factorization of C , we can obtain the LDU factorization of C^{-1}

$$\begin{aligned} C^{-1} &= \begin{bmatrix} I & 0 \\ K^T & I \end{bmatrix}^{-1} \begin{bmatrix} P & 0 \\ 0 & C_{YY} \end{bmatrix}^{-1} \begin{bmatrix} I & K \\ 0 & I \end{bmatrix}^{-1} \\ &= \begin{bmatrix} I & 0 \\ -K^T & I \end{bmatrix} \begin{bmatrix} P^{-1} & 0 \\ 0 & C_{YY}^{-1} \end{bmatrix} \begin{bmatrix} I & -K \\ 0 & I \end{bmatrix} \end{aligned}$$

- we can rewrite the exponent of the joint distribution as

$$\begin{aligned} & \begin{bmatrix} X - m_X \\ Y - m_Y \end{bmatrix}^T \begin{bmatrix} C_{XX} & C_{XY} \\ C_{YX} & C_{YY} \end{bmatrix}^{-1} \begin{bmatrix} X - m_X \\ Y - m_Y \end{bmatrix} \\ &= \begin{bmatrix} X - m_X \\ Y - m_Y \end{bmatrix}^T \begin{bmatrix} I & 0 \\ -K^T & I \end{bmatrix} \begin{bmatrix} P^{-1} & 0 \\ 0 & C_{YY}^{-1} \end{bmatrix} \begin{bmatrix} I & -K \\ 0 & I \end{bmatrix} \begin{bmatrix} X - m_X \\ Y - m_Y \end{bmatrix} \\ &= \begin{bmatrix} X - KY - (m_X - Km_Y) \\ Y - m_Y \end{bmatrix}^T \begin{bmatrix} P^{-1} & 0 \\ 0 & C_{YY}^{-1} \end{bmatrix} \begin{bmatrix} X - KY - (m_X - Km_Y) \\ Y - m_Y \end{bmatrix} \\ &= [X - KY - (m_X - Km_Y)]^T P^{-1} [X - KY - (m_X - Km_Y)] \\ &\quad + [Y - m_Y]^T C_{YY}^{-1} [Y - m_Y] \end{aligned}$$

The Gauss-Markov Theorem

- By taking determinants, we also obtain

$$\begin{aligned}\det C &= \det \begin{bmatrix} I & K \\ 0 & I \end{bmatrix} \det \begin{bmatrix} P & 0 \\ 0 & C_{YY} \end{bmatrix} \det \begin{bmatrix} I & 0 \\ K^T & I \end{bmatrix} \\ &= 1 \cdot \det \begin{bmatrix} P & 0 \\ 0 & C_{YY} \end{bmatrix} \cdot 1 = \det P \det C_{YY}\end{aligned}$$

- **Theorem 0.1 (Gauss-Markov)** *If \mathbf{X} and \mathbf{Y} have the joint Gaussian distribution indicated before, then the conditional distribution is*

$$\begin{aligned}f_{\mathbf{X}|\mathbf{Y}}(X|Y) &= (2\pi)^{-\frac{m}{2}} (\det P)^{-\frac{1}{2}} \\ &\exp \left(-\frac{1}{2} [X - KY - (m_X - Km_Y)]^T P^{-1} [X - KY - (m_X - Km_Y)] \right)\end{aligned}$$

The conditional distribution is again Gaussian with conditional mean

$$E_{\mathbf{X}|\mathbf{Y}} \mathbf{X} = E[\mathbf{X} | \mathbf{Y} = Y] = m_X + \underbrace{C_{XY} C_{YY}^{-1}}_{=K} (Y - m_Y)$$

and conditional covariance matrix

$$E_{\mathbf{X}|\mathbf{Y}} [\mathbf{X} - E_{\mathbf{X}|\mathbf{Y}} \mathbf{X}] [\mathbf{X} - E_{\mathbf{X}|\mathbf{Y}} \mathbf{X}]^T = P = C_{XX} - C_{XY} C_{YY}^{-1} C_{YX}$$

Example 1.1 Two correlated Gaussian r.v.'s

- $m = n = 1$, zero means $m_X = m_Y = 0$, rename $C_{XX} = \sigma_x^2$, $C_{YY} = \sigma_y^2$.
- Cauchy-Schwarz inequality : $|\langle \mathbf{x}, \mathbf{y} \rangle|^2 \leq \langle \mathbf{x}, \mathbf{x} \rangle \langle \mathbf{y}, \mathbf{y} \rangle = \|\mathbf{x}\|^2 \|\mathbf{y}\|^2$
- $\langle \mathbf{x}, \mathbf{y} \rangle = E\mathbf{x}\mathbf{y} \Rightarrow C_{XY}^2 \leq C_{XX}C_{YY}$, $\|\mathbf{x}\|^2 = E\mathbf{x}^2$
- isomorphism: $\mathbf{x} = \underline{x}^T \mathbf{e}$, $\mathbf{y} = \underline{y}^T \mathbf{e}$, $R_{\mathbf{e}\mathbf{e}} = I$, $\Rightarrow E\mathbf{x}\mathbf{y} = \langle \mathbf{x}, \mathbf{y} \rangle = \langle \underline{x}, \underline{y} \rangle = \underline{x}^T \underline{y}$
- $E(\mathbf{x} + \lambda \mathbf{y})^2 = \lambda^2 r_{yy} + 2\lambda r_{xy} + r_{xx} \stackrel{\lambda=-r_{xy}/r_{yy}}{\geq} r_{xx} - \frac{r_{xy}^2}{r_{yy}} \geq 0 \Rightarrow r_{xy}^2 \leq r_{xx}r_{yy}$
- introduce normalized correlation coefficient $\rho = \frac{C_{XY}}{\sqrt{C_{XX}C_{YY}}} \in [-1, 1]$
can rewrite C_{XY} as $C_{XY} = \rho\sigma_x\sigma_y$.
- The joint Gaussian distribution of \mathbf{x} and \mathbf{y} can be written as

$$f_{\mathbf{x},\mathbf{y}}(x, y) \leftrightarrow \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}\right)$$

- The conditional pdf of \mathbf{x} given \mathbf{y} can be written as (Gauss-Markov theorem)

$$f_{\mathbf{x}|\mathbf{y}}(x|y) \leftrightarrow \mathcal{N}\left(\rho \frac{\sigma_x}{\sigma_y} y, \sigma_x^2(1-\rho^2)\right)$$

Example 1.1 Two correlated Gaussian r.v.'s (2)

- if \mathbf{x}, \mathbf{y} uncorrelated ($\rho = 0$), then $f_{\mathbf{x}|\mathbf{y}}(x|y) = f_{\mathbf{x}}(x)$: \mathbf{x}, \mathbf{y} independent
- as $|\rho| \rightarrow 1$, $|E_{\mathbf{x}|\mathbf{y}}\mathbf{x}| \nearrow$, conditional variance $\rightarrow 0$:
when $\mathbf{y} = y$ is known, it gives us some information about \mathbf{x} and the residual randomness in \mathbf{x} decreases as $|\rho| \rightarrow 1$.

- Extreme cases:

$\rho = 1$: $\mathbf{x} = \frac{\sigma_x}{\sigma_y}\mathbf{y}$ \mathbf{x} and \mathbf{y} are perfectly correlated,

$\rho = -1$: $\mathbf{x} = -\frac{\sigma_x}{\sigma_y}\mathbf{y}$ \mathbf{x} and \mathbf{y} are perfectly anticorrelated.

- *concentration ellipse*:

$$\begin{aligned} f_{\mathbf{x},\mathbf{y}}(x,y) = c' &\iff \begin{bmatrix} x \\ y \end{bmatrix}^T \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \end{bmatrix} = c \\ &\iff \frac{(x - \rho\frac{\sigma_x}{\sigma_y}y)^2}{\sigma_x^2(1-\rho^2)} + \frac{y^2}{\sigma_y^2} = c \end{aligned}$$



Example 1.1 Two correlated Gaussian r.v.'s (3)

- the ellipse for $c = 1$

$$\frac{\left(x - \rho \frac{\sigma_x}{\sigma_y} y\right)^2}{\sigma_x^2(1 - \rho^2)} + \frac{y^2}{\sigma_y^2} \leq 1$$

contains a significant portion of the probability mass.

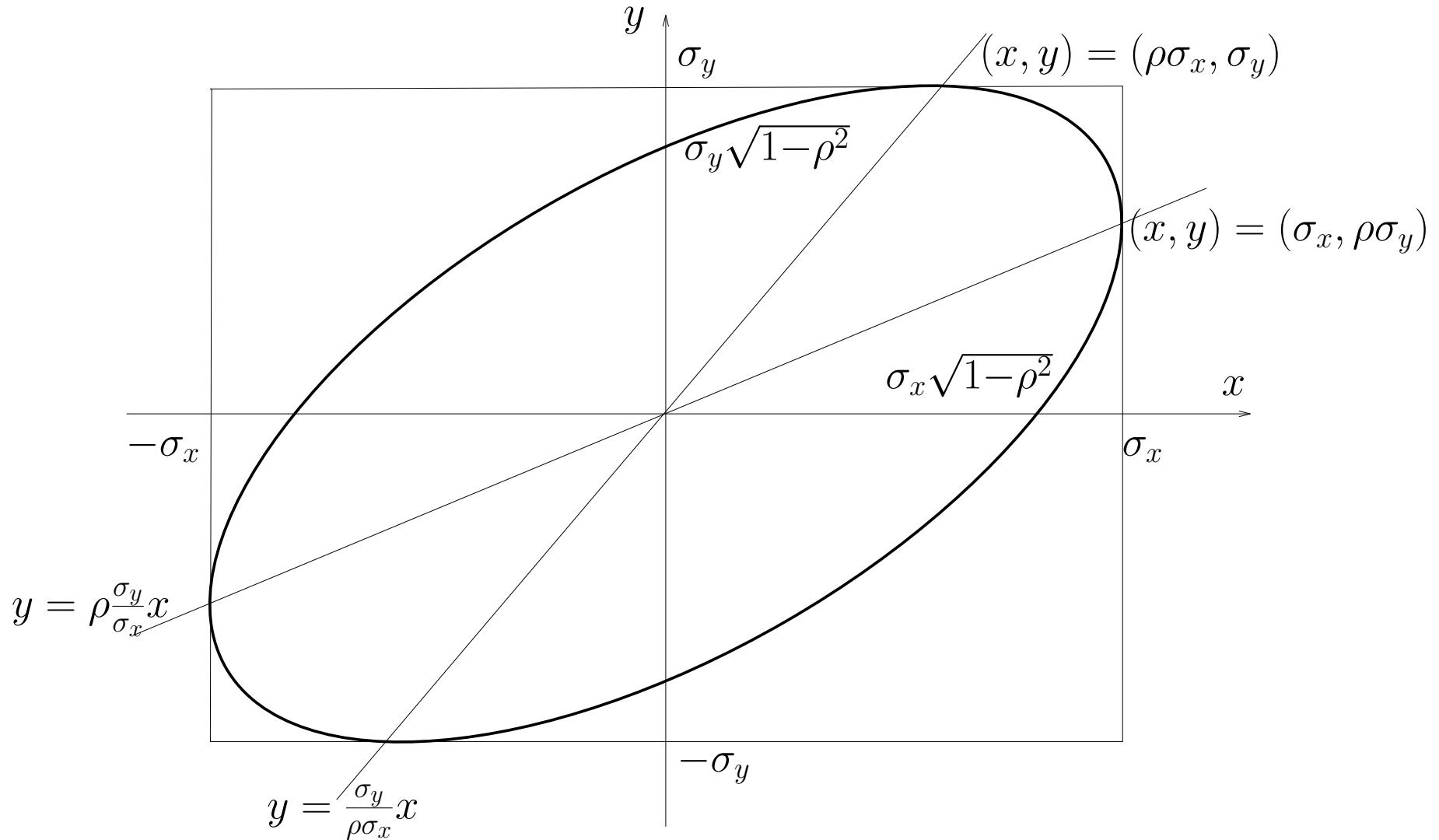
- its volume is

$$V = \pi (\det C)^{\frac{1}{2}} = \pi \sigma_x \sigma_y \sqrt{1 - \rho^2} \in [0, \pi \sigma_x \sigma_y]$$

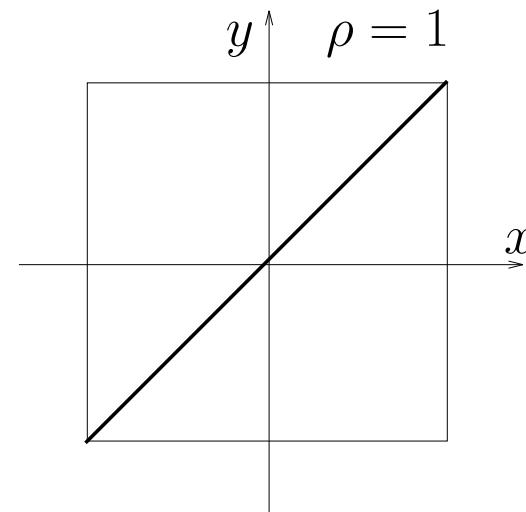
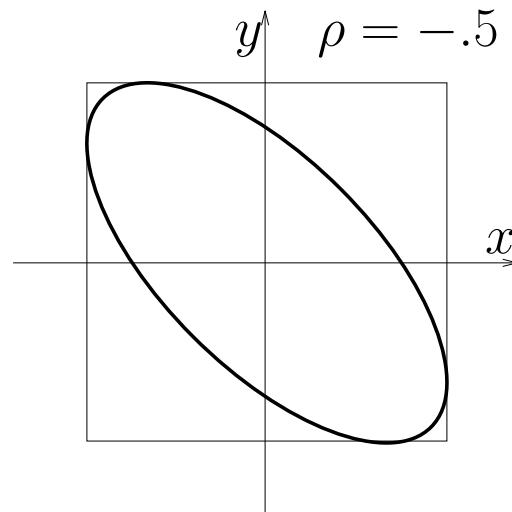
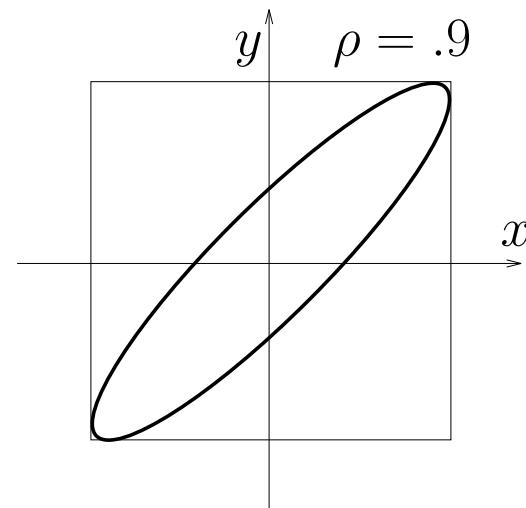
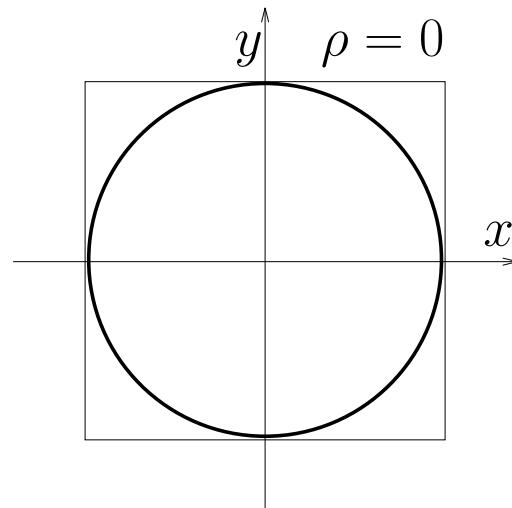
- for strongly correlated variables, the pair (x, y) takes with high probability values in a fairly small area. Hence, in some sense the randomness (or entropy) of the pair (x, y) decreases as $|\rho| \rightarrow 1$. (later: Gaussian r.v.'s: entropy $\sim \det C$)

Example 1.1 Two correlated Gaussian r.v.'s (4)

concentration ellipses



Example 1.1 Two correlated Gaussian r.v.'s (5)



Gaussian r.v.'s and Linear Models

- by interchanging X and Y , we find the conditional pdf of \mathbf{Y} given \mathbf{X}

$$f_{\mathbf{Y}|\mathbf{X}}(Y|X) \leftrightarrow \mathcal{N}(m_Y + C_{YX}C_{XX}^{-1}(X - m_X), C_{YY} - C_{YX}C_{XX}^{-1}C_{XY})$$

- introduce the Gaussian r. vector \mathbf{V} of dimension n also with distribution

$$f_{\mathbf{V}}(V) \leftrightarrow \mathcal{N}(m_V, C_{VV}) = \mathcal{N}(\underbrace{m_Y - C_{YX}C_{XX}^{-1}m_X}_{=m_V}, \underbrace{C_{YY} - C_{YX}C_{XX}^{-1}C_{XY}}_{=C_{VV}})$$

and \mathbf{V} independent of \mathbf{X} ,

- then \mathbf{X} and \mathbf{Y} generated as follows

$$\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} I & 0 \\ C_{YX}C_{XX}^{-1} & I \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{V} \end{bmatrix} \Rightarrow \begin{bmatrix} I & 0 \\ C_{YX}C_{XX}^{-1} & I \end{bmatrix} \begin{bmatrix} m_X \\ m_V \end{bmatrix} = \begin{bmatrix} m_X \\ m_Y \end{bmatrix}$$

$$\begin{bmatrix} I & 0 \\ C_{YX}C_{XX}^{-1} & I \end{bmatrix} \begin{bmatrix} C_{XX} & \overset{=0}{\underset{\mathcal{C}_{XV}}{\overbrace{C_{XV}}} \\ \underset{=0}{\underset{\mathcal{C}_{VX}}{\overbrace{C_{VX}}}} & \underset{=C_{YY}-C_{YX}C_{XX}^{-1}C_{XY}}{\underset{\mathcal{C}_{VV}}{\overbrace{C_{VV}}}} \end{bmatrix} \begin{bmatrix} I & C_{XX}^{-1}C_{XY} \\ 0 & I \end{bmatrix} = \begin{bmatrix} C_{XX} & C_{XY} \\ C_{YX} & C_{YY} \end{bmatrix}$$

(block Lower-Diagonal-Upper (LDU) triangular factorization of C)

are jointly Gaussian and have the correct mean and variance, hence have the correct pdf $f_{\mathbf{X}, \mathbf{Y}}(X, Y)$.

Gaussian r.v.'s and Linear Models (2)

- we can write explicitly

$$\mathbf{Y} = C_{YX}C_{XX}^{-1}\mathbf{X} + \mathbf{V}.$$

This means that for two sets of correlated Gaussian random variables, one (\mathbf{Y}) can be thought of as being generated from the other (\mathbf{X}) through a linear model ($C_{YX}C_{XX}^{-1}$) and being corrupted by independent Gaussian “measurement” noise (\mathbf{V}).

- Going back to the scalar example ($m = n = 1$), the variance of the linear model part ($C_{YX}C_{XX}^{-1}\mathbf{X}$) is $\rho^2\sigma_y^2$ while the variance of the measurement noise (\mathbf{V}) is $(1-\rho^2)\sigma_y^2$. The two are complementary parts of the total variance σ_y^2 of \mathbf{Y} . We could define a signal to noise ratio (SNR) as the ratio of the two parts which is $\frac{\rho^2}{1-\rho^2}$. The SNR increases from 0 to ∞ as $|\rho|$ increases from 0 to 1, as the correlation between \mathbf{X} and \mathbf{Y} increases.

The Parameter Estimation Problem

description of stochastic processes known up to some parameters

- tone detector: a push button telephone emits sinusoids of which the frequencies are distinct for different buttons. The unknown parameter of interest is the sinusoid frequency. This frequency determination problem may in fact be better approached as a detection problem as we shall see shortly
- the carrier phase and timing instants in linear digital modulation schemes
- the impulse response of the transmission channel. We may take a parameterized model (e.g. FIR model) for this impulse response and then the channel identification problem becomes a problem of estimating the parameters in its model
- similar impulse response identification problems occur in the problems of the cancellation of electrical echos caused by hybrids connecting 2-wire and 4-wire sections of telephone line, and acoustic echos in teleconferencing systems

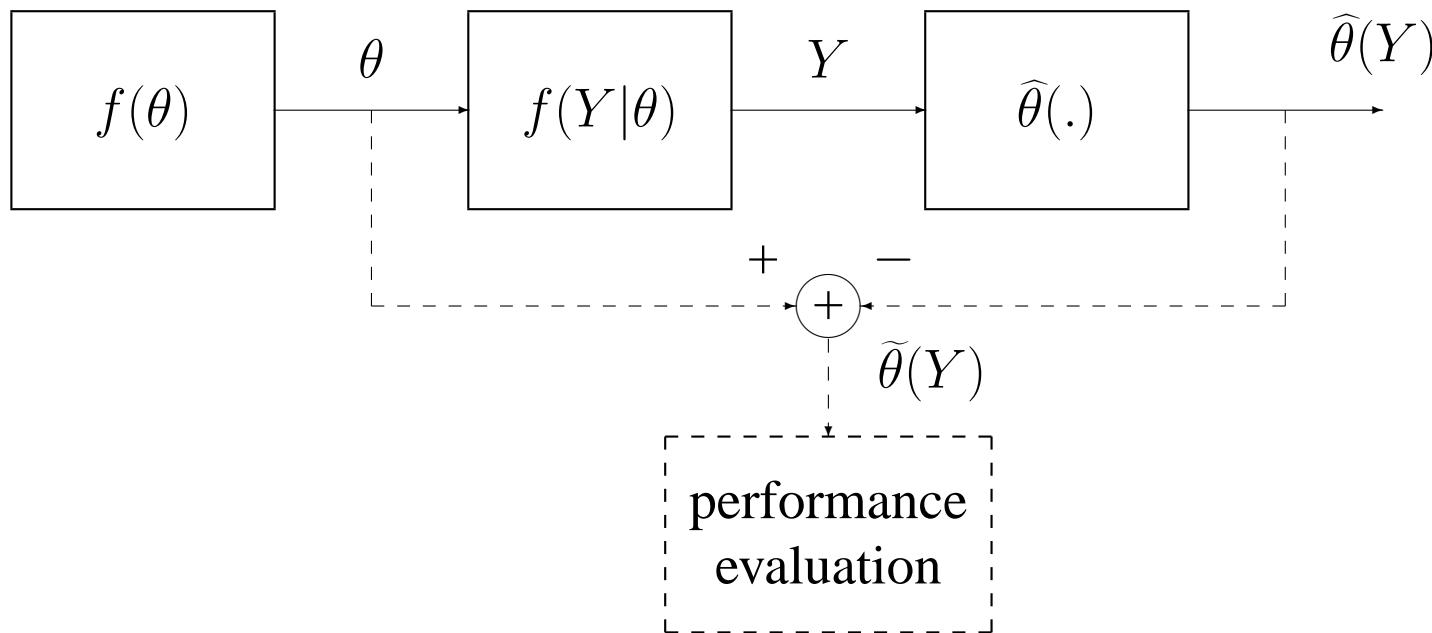
The Parameter Estimation Problem (2)

description of stochastic processes known up to some parameters

- the average rate in a Poisson process occurring in queuing theory and network performance analysis
- the pixel value in the noisy capture of an image by a satellite
- the position and orientation of an object in an image
- security in mobile communications: the (parameterized) distribution of features characterizing the behavior of users. The deviation from typical habits may be a reason for alarm.

Bayesian Framework

- Bayesian approach: consider parameters θ to be random variables
- the parameters have some *a priori* distribution $f_{\theta}(\theta)$. This distribution is called a priori because it summarizes the knowledge we have about θ before making any measurement.
- Next we make a *measurement* Y . The measurement is not a deterministic function of the unknown parameters. The random aspect of this relation is captured by the conditional distribution $f_{Y|\theta}(Y|\theta)$.



Ex 1.2: Additive indep. measurement noise

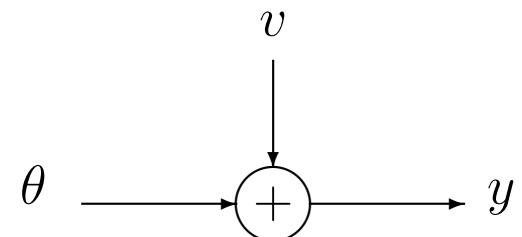
$$\bullet f_{\mathbf{y}|\boldsymbol{\theta}}(y|\theta) = \frac{f_{\mathbf{y},\boldsymbol{\theta}}(y,\theta)}{f_{\boldsymbol{\theta}}(\theta)} = \frac{f_{\mathbf{v},\boldsymbol{\theta}}(y-\theta,\theta)}{f_{\boldsymbol{\theta}}(\theta)} = \frac{f_{\mathbf{v}}(y-\theta) f_{\boldsymbol{\theta}}(\theta)}{f_{\boldsymbol{\theta}}(\theta)} = f_{\mathbf{v}}(y-\theta)$$

- first identity: Bayes' rule
- second identity: transformation of variables

$$\begin{bmatrix} v \\ \theta \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y \\ \theta \end{bmatrix}$$

linear transformation \Rightarrow Jacobian = determinant of the transformation matrix, which equals 1.

- third identity: independence of v and θ





Estimators

- *estimator* : a function $\widehat{\theta}(\cdot)$ to be applied to the measurement \mathbf{Y}
- *estimate* : $\widehat{\theta}(Y)$ is the estimator $\widehat{\theta}(\mathbf{Y})$ evaluated at $\mathbf{Y} = Y$.
- we consider here a *point estimator*, i.e. it delivers one value $\widehat{\theta} = \widehat{\theta}(Y) = t(y_1, \dots, y_n)$ that we hope to be close to θ .
- The function $\widehat{\theta}(\mathbf{Y}) = t(\mathbf{y}_1, \dots, \mathbf{y}_n)$ of the measurement data is called a *statistic*.
- Another type of estimator would be an *interval estimator*: two statistics $t_1(y_1, \dots, y_n)$ and $t_2(y_1, \dots, y_n)$ are used to define an interval such that

$$\Pr \{ \theta \in (t_1(y_1, \dots, y_n), t_2(y_1, \dots, y_n)) \} = \gamma$$

can be determined. γ is called the *confidence coefficient*.



Ex: Estimators for the mean of Gaussian r.v.'s

- Let the $y_i \sim \mathcal{N}(\theta, 5)$ be i.i.d. (independent and identically distributed) normal random variables with unknown mean θ and variance equal to 5.
- The arithmetic mean is a *point estimator* of θ :

$$\hat{\theta} = \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i \sim \mathcal{N}\left(\theta, \frac{5}{n}\right).$$

- Now, since $\bar{y} \sim \mathcal{N}\left(\theta, \frac{5}{n}\right)$ or $\bar{y} = \theta + v$ with $v \sim \mathcal{N}(0, \frac{5}{n})$, we can also state

$$\Pr\left\{\theta \in \left(\bar{y} - 2\sqrt{\frac{5}{n}}, \bar{y} + 2\sqrt{\frac{5}{n}}\right)\right\} = \Pr\left\{\bar{y} \in \left(\theta - 2\sqrt{\frac{5}{n}}, \theta + 2\sqrt{\frac{5}{n}}\right)\right\} = 0.95$$

since

$$\bar{y} - 2\sqrt{\frac{5}{n}} < \theta < \bar{y} + 2\sqrt{\frac{5}{n}} \Leftrightarrow \theta - 2\sqrt{\frac{5}{n}} < \bar{y} < \theta + 2\sqrt{\frac{5}{n}}$$

so that $(\bar{y} - 2\sqrt{\frac{5}{n}}, \bar{y} + 2\sqrt{\frac{5}{n}})$ is an *interval estimator* for θ with confidence coefficient 0.95.



Estimation Considerations

- The design of interval estimators is the result of the following compromise. On the one hand, one wants a small interval so that θ can be known fairly accurately. On the other hand, a small interval necessarily leads to a small confidence coefficient since in general we can only state with a low probability level that θ is contained in a small interval. But we would like the confidence coefficient to be high so that we would be able to say with high probability where θ is located. This again would lead to a large interval etc.
- We will restrict the further discussion to point estimators. In general, the estimators we shall consider will asymptotically (for many measurements, large n) have a Gaussian distribution so that we can easily construct an interval estimator from the point estimator as in the example above.
- As a final remark, we shall assume that the parameters (and their estimators) can take on a continuous range of values so that their distribution is described by a probability density function. In case the parameters can take on only a *discrete set of values* and the task is to decide which one of the values the parameters actually take, then the estimation problem is called a *detection* or *decision* problem (see DIGICOM course).



Bayes Estimation

- nonnegative cost function $\mathcal{C}(\theta, \hat{\theta})$
- since $\theta, \hat{\theta}(Y)$ are random variables, we can never hope to make this cost function small for every outcome of θ and Y . All we can hope for is to minimize the expected value of the cost function, also called the risk

$$\mathcal{R}(\hat{\theta}(.)) = E \mathcal{C}(\theta, \hat{\theta}(Y)) = E_{\theta, Y} \mathcal{C}(\theta, \hat{\theta}(Y)).$$

So the estimator is that function $\theta(.)$ that minimizes the Bayes risk

$$\hat{\theta}(.) = \arg \min_{\hat{\theta}(.)} \mathcal{R}(\hat{\theta}(.))$$

It is common practice to limit the choice of the cost function to a nonnegative cost function of the parameter estimation error

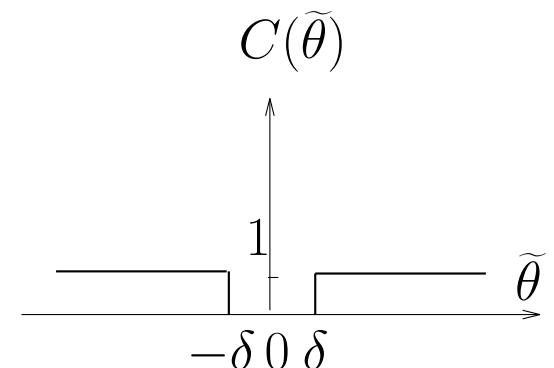
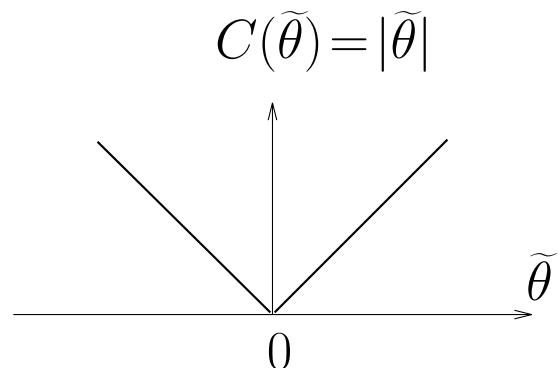
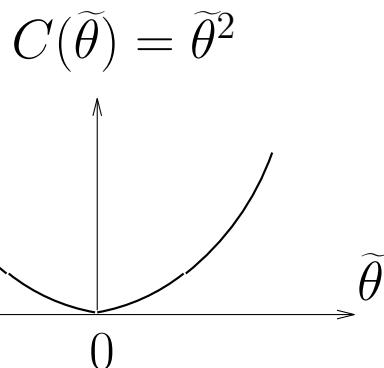
$$\tilde{\theta} = \theta - \hat{\theta}(Y)$$

only, in which case the estimation problem becomes

$$\hat{\theta}(.) = \arg \min_{\hat{\theta}(.)} E \mathcal{C}(\tilde{\theta})$$

Bayes Cost Functions

- Three popular Bayes cost functions: squared parameter deviation, absolute parameter deviation, and the uniform cost function. All three choices assign no cost when there is no error.



Bayes Risk Minimization

- manipulate the Bayes optimization criterion

$$\begin{aligned} \min_{\hat{\theta}(.)} \mathcal{R}(\hat{\theta}(.)) &= \min_{\hat{\theta}(.)} E \mathcal{C}(\hat{\theta}) = \min_{\hat{\theta}(.)} \int \int f(Y, \theta) \mathcal{C}(\theta - \hat{\theta}(Y)) dY d\theta \\ &= \min_{\hat{\theta}(.)} \underbrace{\int f(Y) dY}_{\geq 0} \underbrace{\int f(\theta|Y) \mathcal{C}(\theta - \hat{\theta}(Y)) d\theta}_{\mathcal{R}(\hat{\theta}(.)|Y)} = \min_{\hat{\theta}(.)} E_Y \mathcal{R}(\hat{\theta}(.)|Y) = E_Y \min_{\hat{\theta}(Y)} \mathcal{R}(\hat{\theta}(Y)|Y) \end{aligned}$$

$\mathcal{R}(\hat{\theta}(.)|Y) = \mathcal{R}(\hat{\theta}(Y)|Y)$ is a function of Y .

- Since the contributions of the conditional risk $\mathcal{R}(\hat{\theta}(Y)|Y)$ for every Y are combined via the nonnegative weighting factor $f(Y)$ to obtain the global risk $\mathcal{R}(\hat{\theta}(.))$, we can minimize $\mathcal{R}(\hat{\theta}(.))$ by minimizing $\mathcal{R}(\hat{\theta}(Y)|Y)$ for every particular Y .
- While the minimization of the global risk $\mathcal{R}(\hat{\theta}(.))$ is with respect to (w.r.t.) the estimator function $\hat{\theta}(.)$, the minimization of the conditional risk $\mathcal{R}(\hat{\theta}(Y)|Y)$ is w.r.t. the particular estimate $\hat{\theta}(Y)$, which is simply a number (the estimator function $\hat{\theta}(.)$ evaluated at $\mathbf{Y} = Y$).

Bayes Risk Minimization (2)

- This minimization of the conditional risk $\mathcal{R}(\hat{\theta}(Y)|Y)$ requires the *a posteriori* distribution $f(\theta|Y)$ of θ given the measurement Y .
- The a posteriori distribution describes the randomness left in θ after we have made a measurement of (the related quantity) Y .
- The a posteriori distribution $f(\theta|Y)$ can be determined from the conditional distribution $f(Y|\theta)$ and the a priori distribution $f(\theta)$, which are normally given in the problem description, as follows (using Bayes' rule):

$$f(\theta|Y) = \frac{f(Y, \theta)}{f(Y)} = \frac{f(Y|\theta)f(\theta)}{\int_{\Theta} f(Y, \theta)d\theta} = \frac{f(Y|\theta)f(\theta)}{\int_{\Theta} f(Y|\theta)f(\theta)d\theta}$$

where Θ is the region of support for θ .



The MMSE Criterion

- using the quadratic cost function $\mathcal{C}_{MMSE}(\tilde{\theta}) = |\tilde{\theta}|^2$, minimizing the conditional Bayes risk yields

$$\min_{\hat{\theta}(Y)} \mathcal{R}_{MMSE}(\hat{\theta}(Y)|Y) = \min_{\hat{\theta}(Y)} \int_{-\infty}^{\infty} f(\theta|Y) (\theta - \hat{\theta}(Y))^2 d\theta$$

- to take the derivative w.r.t. $\hat{\theta}$, recall Leibnitz's rule:

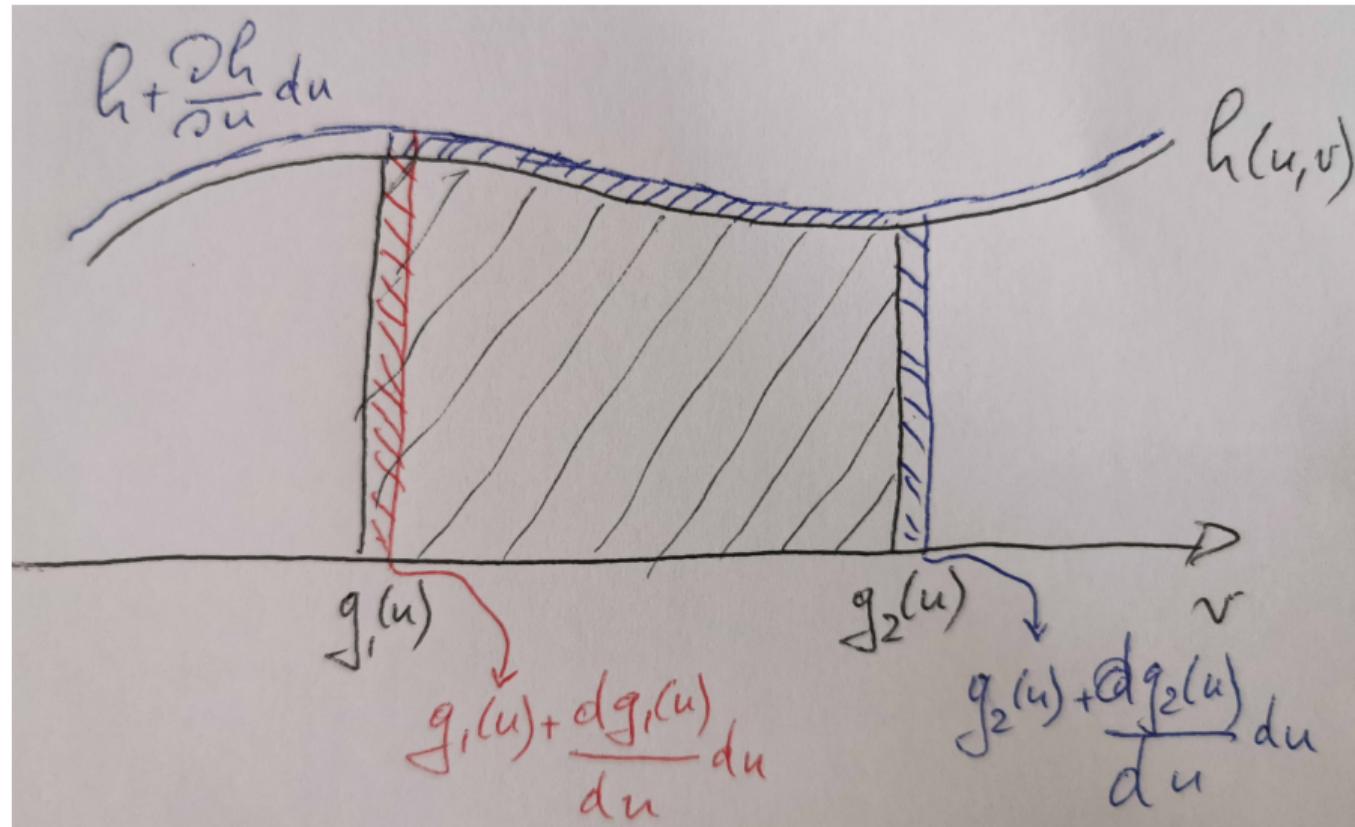
$$\frac{\partial}{\partial u} \int_{g_1(u)}^{g_2(u)} h(u, v) dv = \int_{g_1(u)}^{g_2(u)} \frac{\partial h(u, v)}{\partial u} dv + \frac{dg_2(u)}{du} h(u, g_2(u)) - \frac{dg_1(u)}{du} h(u, g_1(u))$$

- Using Leibnitz's rule, we obtain

$$\frac{\partial}{\partial \hat{\theta}} \mathcal{R}_{MMSE}(\hat{\theta}|Y) = 2 \int_{-\infty}^{\infty} f(\theta|Y) (\hat{\theta} - \theta) d\theta = 0$$

where we put the derivative equal to zero in order to find an extremum.

Leibnitz's Rule



The MMSE Criterion (2)

- We can rewrite as

$$\widehat{\theta}(Y) \underbrace{\int_{-\infty}^{\infty} f(\theta|Y) d\theta}_{=1} = \int_{-\infty}^{\infty} \theta f(\theta|Y) d\theta \Rightarrow \widehat{\theta}_{MMSE}(Y) = E(\theta|Y)$$

which is the *mean* of the a posteriori distribution of θ given Y .

- To know whether this extremum is a local minimum, we can apply Leibnitz's rule a second time to obtain

$$\frac{\partial^2}{\partial \widehat{\theta}^2} \mathcal{R}_{MMSE}(\widehat{\theta}|Y) = 2 \int_{-\infty}^{\infty} f(\theta|Y) d\theta = 2 > 0$$

Hence the extremum at $\widehat{\theta}(Y) = E(\theta|Y)$ is a local minimum and it is furthermore the global minimum since it is the unique local minimum. Note that $E(\theta|Y)$ is indeed a function of Y .

The absolute value cost function

- For the absolute value cost function $\mathcal{C}_{ABS}(\hat{\theta}) = |\hat{\theta}|$, the minimization problem of the conditional Bayes risk becomes

$$\begin{aligned}\min_{\hat{\theta}(Y)} \mathcal{R}_{ABS}(\hat{\theta}(Y)|Y) &= \min_{\hat{\theta}(Y)} \int_{-\infty}^{\infty} f(\theta|Y) |\theta - \hat{\theta}(Y)| d\theta \\ &= \min_{\hat{\theta}(Y)} \left[\int_{-\infty}^{\hat{\theta}} f(\theta|Y) (\hat{\theta} - \theta) d\theta + \int_{\hat{\theta}}^{\infty} f(\theta|Y) (\theta - \hat{\theta}) d\theta \right].\end{aligned}$$

- We shall again use Leibnitz's rule to take the derivative. For the first integral, we let $h(\hat{\theta}, \theta) = f(\theta|Y) (\hat{\theta} - \theta)$ and we get

$$h(\hat{\theta}, g_2(\hat{\theta})) = h(\hat{\theta}, \hat{\theta}) = f(\hat{\theta}|Y) (\hat{\theta} - \hat{\theta}) = 0$$

and $\frac{d g_1(\hat{\theta})}{d \hat{\theta}} = 0$ since the lower limit does not depend on $\hat{\theta}$. The corresponding terms for the second integral are similarly equal to zero.

- So the only terms in the derivative remaining are those obtained by differentiating the integrands directly:

$$\frac{\partial}{\partial \hat{\theta}} \mathcal{R}_{ABS}(\hat{\theta}|Y) = \int_{-\infty}^{\hat{\theta}} f(\theta|Y) d\theta - \int_{\hat{\theta}}^{\infty} f(\theta|Y) d\theta = 0$$

where again we put the derivative equal to zero in order to find an extremum.

The absolute value cost function (2)

- So the condition for an extremum becomes

$$F(\hat{\theta}|Y) = \int_{-\infty}^{\hat{\theta}} f(\theta|Y) d\theta = \int_{\hat{\theta}}^{\infty} f(\theta|Y) d\theta = 1 - F(\hat{\theta}|Y)$$

where $F(\theta|Y)$ is the a posteriori cumulative distribution function (cdf) of θ given Y .

- We can rewrite this also as

$$F(\hat{\theta}_{ABS}(Y)|Y) = \frac{1}{2}$$

which means that $\hat{\theta}_{ABS}(Y)$ is the *median* of the a posteriori distribution of θ given Y .

The absolute value cost function (3)

- Again, to know whether this extremum is a local minimum, we can apply Leibnitz's rule a second time to obtain

$$\frac{\partial^2}{\partial \hat{\theta}^2} \mathcal{R}_{ABS}(\hat{\theta}|Y) \Big|_{\hat{\theta}=\hat{\theta}_{ABS}} = 2 f(\hat{\theta}_{ABS}|Y) \geq 0.$$

If $f(\hat{\theta}_{ABS}|Y) = 0$ then, since $f(\theta|Y) \geq 0$, the first nonzero derivative w.r.t. θ of $f(\theta|Y)$ will be of even order and positive (this reasoning can be extended to the case where $f(\theta|Y)$ would not be sufficiently differentiable). Hence the extremum at $\hat{\theta}_{ABS}(Y)$ is a local minimum and it is furthermore the global minimum since it is the unique local minimum. Note again that $\hat{\theta}_{ABS}(Y)$ is indeed a function of Y .

The uniform cost function

- For the uniform cost function we get

$$\begin{aligned} \min_{\hat{\theta}(Y)} \mathcal{R}_{UNIF}(\hat{\theta}(Y)|Y) &= \min_{\hat{\theta}(Y)} \left[\left(\int_{-\infty}^{\hat{\theta}-\delta} + \int_{\hat{\theta}+\delta}^{\infty} \right) f(\theta|Y) d\theta \right] \\ &= \min_{\hat{\theta}(Y)} \left[\underbrace{\int_{-\infty}^{\infty} f(\theta|Y) d\theta}_{=1} - \int_{\hat{\theta}-\delta}^{\hat{\theta}+\delta} f(\theta|Y) d\theta \right]. \end{aligned}$$

- Since we take δ to be arbitrarily small, the optimization problem becomes

$$\max_{\hat{\theta}(Y)} \int_{\hat{\theta}-\delta}^{\hat{\theta}+\delta} f(\theta|Y) d\theta \approx \max_{\hat{\theta}(Y)} 2\delta f(\hat{\theta}|Y) = 2\delta \max_{\hat{\theta}(Y)} f(\hat{\theta}|Y)$$

Hence, for δ arbitrarily small, $\mathcal{R}_{UNIF}(\hat{\theta}(Y)|Y)$ is minimized by choosing $\hat{\theta}$ to be the *mode* (location of the maximum) of the a posteriori distribution of θ given Y .

- For this reason, the estimator corresponding to a uniform cost function is normally called the *Maximum A Posteriori* (MAP) estimator:

$$\hat{\theta}_{MAP}(Y) = \arg \max_{\theta} f(\theta|Y)$$

which is again a function of Y .



MAP Estimator: Remarks

- We may note that the same estimator is obtained by choosing the cost function $\mathcal{C}(\bar{\theta}) = 1 - \delta(\bar{\theta})$. While this cost function is cleaner in that it involves no limiting operation with δ , it might be considered non-intuitive since it is not nonnegative. However, one should keep in mind that adding or subtracting a constant to a cost function does not influence its minimizing argument.
- Instead of maximizing $f(\theta|Y)$, any strictly increasing function of it may be maximized. Since $f(\theta|Y)$ is often given in factored form and often contains exponential distributions, a convenient choice is to maximize

$$\ln f(\theta|Y) = \ln f(Y|\theta) + \ln f(\theta) - \ln f(Y) .$$

- Often $f(\theta|Y)$ satisfies certain regularity conditions so that $\hat{\theta}_{MAP}$ is a solution of

$$\frac{\partial}{\partial \theta} \ln f(\theta|Y) = 0 = \frac{\partial}{\partial \theta} \ln f(Y|\theta) + \frac{\partial}{\partial \theta} \ln f(\theta) .$$

Note that $\frac{\partial \ln f(\theta|Y)}{\partial \theta} = \frac{1}{f(\theta|Y)} \frac{\partial f(\theta|Y)}{\partial \theta}$. The conditions for a maximum (rather than another form of extremum) need to be verified of course.

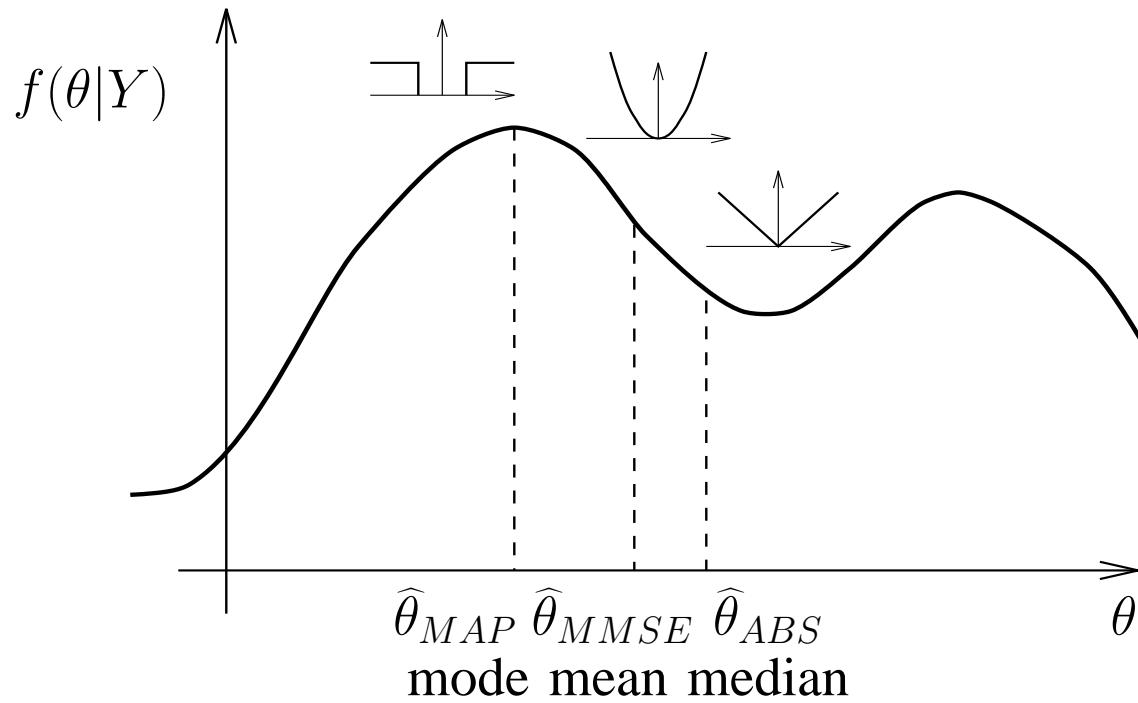


MAP Estimator: Remarks (2)

- The previous equation indicates that, starting from the description of the problem which includes $f(Y|\theta)$ and $f(\theta)$, the calculation of $\hat{\theta}_{MAP}$ should be relatively straightforward.
- The MAP estimator is given by the *global* maximum of $f(\theta|Y)$. If there are several local maxima, all of them need to be examined and compared to find the global maximum.
- Even if $f(\theta|Y)$ satisfies regularity conditions, the maximum may occur at the boundary of the parameter space Θ (which may not necessarily be $(-\infty, \infty)$). In that case, the maximum is not a local extremum.

3 Bayes estimators

- in general, the three Bayes estimators may differ



Ex: Gaussian mean in Gaussian noise

- estimating an unknown dc level in additive Gaussian noise with zero mean: e.g. satellite transmission of a digital image (accumulate many noisy images)
- for any given pixel, the measurement problem can be modeled as

$$y_i = \theta + v_i, \quad i = 1, \dots, n$$

where θ is the true grey value of the pixel, y_i are the consecutive noisy measured grey values, the $v_i \sim \mathcal{N}(0, \sigma_v^2)$ are i.i.d. (Central Limit Theorem).

- Even though the images vary very slowly, the image of one group of $n = 100$ shots will not differ very much from the image of the previous 100 shots. Therefore, we can consider the image estimate (with value m_θ at the pixel considered) from the previous group to be prior information for the problem of estimating the image of the current group (with value θ at the same pixel).

Ex: Gaussian mean in Gaussian noise (2)

- This prior information is not perfect due to the estimation variance associated with the processing of the previous group and also the variation from one image (group) to the next. Therefore we model the prior information as $\theta \sim \mathcal{N}(m_\theta, \sigma_\theta^2)$ where σ_θ^2 reflects the joint effect of estimation variability and variability due to change in time. Also, θ is assumed to be independent of the v_i .
- We can write the measurements in vector form

$$Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \theta \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} + \begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix} = \theta \mathbf{1} + V$$

With $X = \theta$, we can consider the following invertible transformation

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & I_n \end{bmatrix} \begin{bmatrix} \theta \\ V \end{bmatrix}.$$

Now, since θ and V are independent and both Gaussian, they are jointly Gaussian. A linear transformation of jointly Gaussian random variables leads again to Gaussian random variables. Hence, $X = \theta$ and Y are jointly Gaussian.

Ex: Gaussian mean in Gaussian noise (3)

- to determine a Bayes estimator, we need the a posteriori pdf $f(\theta|Y) = f(X|Y)$. Since X and Y are jointly Gaussian, the Gauss-Markov theorem tells us that $f(X|Y)$ is also Gaussian and it also tells us how to compute it from the first and second-order moments of X and Y . So we shall compute these moments.
- First-order moments: $m_X = E X = m_\theta$
 $m_Y = E Y = E(\theta \mathbf{1} + V) = (E\theta) \mathbf{1} + EV = m_\theta \mathbf{1}$
- $C_{VV} = EVV^T = \sigma_v^2 I_n$ since $(EVV^T)_{i,j} = Ev_i v_j = \sigma_v^2 \delta_{i,j} = (\sigma_v^2 I_n)_{i,j}$
- The joint covariance matrix is

$$\begin{aligned} C &= E \begin{bmatrix} X - m_X \\ Y - m_Y \end{bmatrix} \begin{bmatrix} X - m_X \\ Y - m_Y \end{bmatrix}^T = E \begin{bmatrix} \theta - m_\theta \\ (\theta - m_\theta) \mathbf{1} + V \end{bmatrix} \begin{bmatrix} \theta - m_\theta \\ (\theta - m_\theta) \mathbf{1} + V \end{bmatrix}^T \\ &= \begin{bmatrix} \sigma_\theta^2 & \sigma_\theta^2 \mathbf{1}^T \\ \sigma_\theta^2 \mathbf{1} & \sigma_\theta^2 \mathbf{1} \mathbf{1}^T + \sigma_v^2 I \end{bmatrix} = \begin{bmatrix} C_{XX} & C_{XY} \\ C_{YX} & C_{YY} \end{bmatrix} \end{aligned}$$

where we used $E \{(\theta - m_\theta) V\} = E \{(\theta - m_\theta)\} E \{V\} = 0 \cdot 0 = 0$,
 $E [(\theta - m_\theta) \mathbf{1} + V] [(\theta - m_\theta) \mathbf{1} + V]^T = E(\theta - m_\theta)^2 \mathbf{1} \mathbf{1}^T + EVV^T + \mathbf{1} E V^T (\theta - m_\theta)$
 $+ EV(\theta - m_\theta) \mathbf{1}^T = \sigma_\theta^2 \mathbf{1} \mathbf{1}^T + \sigma_v^2 I + 0 + 0$

Ex: Gaussian mean in Gaussian noise (4)

- A key quantity that appears in the expression for $f(X|Y)$ is $C_{XY}C_{YY}^{-1}$. We get

$$\begin{aligned} C_{XY}C_{YY}^{-1} &= \sigma_\theta^2 \mathbf{1}^T [\sigma_v^2 I + \sigma_\theta^2 \mathbf{1} \mathbf{1}^T]^{-1} \\ &= \frac{\sigma_\theta^2}{\sigma_v^2} \mathbf{1}^T \left[I + \frac{\sigma_\theta^2}{\sigma_v^2} \mathbf{1} \mathbf{1}^T \right]^{-1}. \end{aligned}$$

In order to compute the inverse of the matrix in brackets, we shall use the following identity.

- **Lemma 0.1 (Matrix Inversion Lemma)** *If A and C are respectively $n \times n$ and $m \times m$ invertible matrices and B and D are respectively $n \times m$ and $m \times n$ matrices, then*

$$[A + BCD]^{-1} = A^{-1} - A^{-1}B [DA^{-1}B + C^{-1}]^{-1} DA^{-1}$$

if the inverses exists.

- This lemma is very useful when we have to compute $[A + BCD]^{-1}$, we know A^{-1} , and m is (much) smaller than n . We shall apply this lemma with $A = I$, $B = \mathbf{1}$, $C = \frac{\sigma_\theta^2}{\sigma_v^2}$ and $D = \mathbf{1}^T$. Hence $m = 1$.

Ex: Gaussian mean in Gaussian noise (5)

- We obtain

$$C_{XY}C_{YY}^{-1} = \frac{\sigma_\theta^2}{\sigma_v^2} \mathbf{1}^T \underbrace{[I - \mathbf{1}(\mathbf{1}^T \mathbf{1} + \frac{\sigma_v^2}{\sigma_\theta^2})^{-1} \mathbf{1}^T]}_{= n + \frac{\sigma_v^2}{\sigma_\theta^2}} = \frac{\sigma_\theta^2}{\sigma_v^2} \left(1 - \frac{n}{n + \frac{\sigma_v^2}{\sigma_\theta^2}}\right) \mathbf{1}^T = \frac{1}{n + \frac{\sigma_v^2}{\sigma_\theta^2}} \mathbf{1}^T$$

- From the Gauss-Markov theorem, we can now compute the a posteriori mean

$$\begin{aligned} E[\theta|Y] &= m_\theta + C_{XY}C_{YY}^{-1}(Y - m_Y) = m_\theta + \frac{1}{n + \frac{\sigma_v^2}{\sigma_\theta^2}} \mathbf{1}^T (Y - m_\theta \mathbf{1}) \\ &= \left(\frac{1}{\sigma_\theta^2} m_\theta + \frac{n}{\sigma_v^2} \bar{y} \right) / \left(\frac{1}{\sigma_\theta^2} + \frac{n}{\sigma_v^2} \right) = \frac{\frac{1}{\sigma_\theta^2}}{\left(\frac{1}{\sigma_\theta^2} + \frac{n}{\sigma_v^2} \right)} m_\theta + \frac{\frac{n}{\sigma_v^2}}{\left(\frac{1}{\sigma_\theta^2} + \frac{n}{\sigma_v^2} \right)} \bar{y} \end{aligned}$$

where we introduced the sample mean $\bar{y} = \frac{1}{n} \mathbf{1}^T Y = \frac{1}{n} \sum_{i=1}^n y_i$.

- Note that $E[\theta|Y]$ is of the form $\alpha m_\theta + (1-\alpha)\bar{y}$, which is a convex combination between the a priori mean and the sample mean obtained from the data. The respective weighting factors are proportional to the inverse of the variance associated with each of the two components.



Ex: Gaussian mean in Gaussian noise (6)

- We shall call the inverse of the variance the amount of information available:

- $\frac{1}{\sigma_\theta^2}$ = information in the prior distribution $f(\theta)$,
- $\frac{n}{\sigma_v^2}$ = information in the n independent measurements (conditional distribution $f(y_i|\theta) = f_{v_i}(y_i - \theta)$).

- We can consider two extreme cases:

- $\sigma_\theta^2 \ll \frac{\sigma_v^2}{n}$: $E[\theta|Y] \approx m_\theta$. In this case, the measurements are so noisy that the reliable a priori information dominates.
- $\sigma_\theta^2 \gg \frac{\sigma_v^2}{n}$: $E[\theta|Y] \approx \bar{y}$. In this case, the accurate measurements dominate the unreliable a priori information. Note that this second case will eventually occur when $n \rightarrow \infty$.

Ex: Gaussian mean in Gaussian noise (7)

- From the Gauss-Markov theorem, we can also determine the a posteriori variance, which we shall denote by $\sigma_{\hat{\theta}}^2$:

$$\sigma_{\hat{\theta}}^2 = \text{Var} [\theta|Y] = C_{XX} - C_{XY}C_{YY}^{-1}C_{YX} = \sigma_\theta^2 - \frac{1}{n + \frac{\sigma_v^2}{\sigma_\theta^2}} \mathbf{1}^T \mathbf{1} \sigma_\theta^2 = \dots = \left(\frac{1}{\sigma_\theta^2} + \frac{n}{\sigma_v^2} \right)^{-1}$$

or hence

$$\frac{1}{\sigma_{\hat{\theta}}^2} = \frac{1}{\sigma_\theta^2} + \frac{n}{\sigma_v^2}$$

which means that the a priori information and the information in the n independent measurements add up to form the total information in the posterior distribution.

- This allows us to rewrite the a posteriori mean as

$$\frac{1}{\sigma_{\hat{\theta}}^2} E [\theta|Y] = \frac{1}{\sigma_\theta^2} m_\theta + \frac{n}{\sigma_v^2} \bar{y} .$$

Ex: Gaussian mean in Gaussian noise (8)

- Using the Gauss-Markov theorem, we are finally ready to write the a posteriori distribution as

$$f(\theta|Y) \leftrightarrow \mathcal{N}(E[\theta|Y], \sigma_{\hat{\theta}}^2)$$

where $E[\theta|Y]$ and $\sigma_{\hat{\theta}}^2$ are given above.

- Since the posterior distribution is Gaussian, its mean, mode and median coincide. Hence we get

$$\hat{\theta}_{MMSE} = \hat{\theta}_{MAP} = \hat{\theta}_{ABS} = E[\theta|Y] .$$

- Note that the posterior distribution $f(\theta, Y)$ depends on Y only through \bar{y} . In this case we say that the statistic \bar{y} (a function of the data Y) is a *sufficient statistic*, meaning that, for the purpose of estimating θ , the only thing we need to know about y_1, \dots, y_n is \bar{y} .

Equivalences of Bayes Estimators

One characteristic of Bayes estimation is that a Bayes estimator depends on the cost function it is associated with. In the following, we shall examine three cases in which the Bayes estimator coincides with $\widehat{\theta}_{MMSE}$ meaning that the estimator minimizing the risk based on some non-quadratic cost function coincides with $\widehat{\theta}_{MMSE}$, the estimator minimizing a quadratic cost function.

1. Consider a cost function $C(\tilde{\theta})$ with the following properties:

- symmetric: $C(\tilde{\theta}) = C(-\tilde{\theta})$
- convex : $C(\alpha\tilde{\theta}_1 + (1-\alpha)\tilde{\theta}_2) \leq \alpha C(\tilde{\theta}_1) + (1-\alpha) C(\tilde{\theta}_2)$, $\alpha \in [0, 1]$

and let $f(\theta|Y)$ be symmetric about $E[\theta|Y]$. Then $\widehat{\theta}_C = \widehat{\theta}_{MMSE}$.

Equivalences of Bayes Estimators (2)

2. Consider a cost function $C(\tilde{\theta})$ with the following properties:

- symmetric: $C(\tilde{\theta}) = C'(|\tilde{\theta}|)$
- $C'(|\tilde{\theta}|)$ is a non-decreasing function of $|\tilde{\theta}|$ (this condition is weaker than $C(\tilde{\theta})$ being convex)

and let $f(\theta|Y)$ have the following properties

- symmetric about $E[\theta|Y]$
- unimodal (only one local maximum)

and finally let $C(\cdot)$ and $f(\cdot|Y)$ be such that $\lim_{\theta \rightarrow \infty} C(\theta) f(\theta|Y) = 0$, $\forall Y$, then again $\hat{\theta}_C = \hat{\theta}_{MMSE}$.

3. If $f(\theta|Y)$ is Gaussian, then $\hat{\theta}_{MMSE} = \hat{\theta}_{MAP} = \hat{\theta}_{ABS}$.

These equivalences show the relative importance of $\hat{\theta}_{MMSE}$. In general however, $\hat{\theta}_{MAP}$ is the easiest to compute, as is illustrated in the following example.

Ex: Poisson Process

- Consider a communication network node where messages are passing. Let the observation \mathbf{N} be the number of messages that pass during a certain observation period of duration τ . \mathbf{N} has a Poisson distribution

$$\Pr [\mathbf{N} = N | \theta] = (\theta\tau)^N \frac{e^{-\theta\tau}}{N!}, \quad N = 0, 1, 2, \dots$$

where the parameter θ represents the average number of messages passing per second by the node we are considering.

- This average frequency has itself an a priori distribution (over the different nodes in the network) which we take to be exponential with average value $m_\theta = 1/\lambda$:

$$f(\theta) = \begin{cases} \lambda e^{-\lambda\theta} & , \theta > 0 \\ 0 & , \theta \leq 0 . \end{cases}$$



Ex: Poisson Process (2)

- In order to find $\hat{\theta}_{MMSE}$, we shall compute the a posteriori distribution of θ given the measurement N . Using Bayes' rule we get

$$f(\theta|N) = \frac{\Pr[\mathbf{N} = N|\theta] f(\theta)}{\Pr[\mathbf{N} = N]} = \frac{1}{\Pr[\mathbf{N} = N]} \lambda \frac{\tau^N}{N!} \theta^N e^{-\theta(\tau+\lambda)} = k(N) \theta^N e^{-\theta(\tau+\lambda)}, \theta > 0$$

where $k(N)$ depends on N but not on θ . Note that $f(\theta|N) = 0, \theta \leq 0$.

- In order to determine $k(N)$, we use the constraint

$$\int_0^\infty f(\theta|N)d\theta = 1 = k(N) \underbrace{\int_0^\infty \theta^N e^{-\theta(\tau+\lambda)} d\theta}_{g(N)}.$$

Hence $k(N) = 1/g(N)$. We shall calculate $g(N)$ by partial integration:

$$g(N) = \left[\frac{\theta^N e^{-\theta(\tau+\lambda)}}{-(\tau + \lambda)} \right]_0^\infty + \frac{N}{\tau + \lambda} \int_0^\infty \theta^{N-1} e^{-\theta(\tau+\lambda)} d\theta = \frac{N}{\tau + \lambda} g(N-1) = \dots = \frac{N!}{(\tau + \lambda)^N} g(0)$$

where

$$g(0) = \int_0^\infty e^{-\theta(\tau+\lambda)} d\theta = \left[\frac{e^{-\theta(\tau+\lambda)}}{-(\tau + \lambda)} \right]_0^\infty = \frac{1}{\tau + \lambda}$$



Ex: Poisson Process (3)

- which finally leads to

$$g(N) = \frac{N!}{(\tau + \lambda)^{N+1}}, k(N) = 1/g(N) = \frac{(\tau + \lambda)^{N+1}}{N!}.$$

- Now we can calculate

$$\begin{aligned}\widehat{\theta}_{MMSE} &= E[\theta|N] = \int_0^\infty \theta f(\theta|N) d\theta \\ &= \frac{1}{g(N)} \int_0^\infty \theta^{N+1} e^{-\theta(\tau+\lambda)} d\theta = \frac{g(N+1)}{g(N)} = \frac{N+1}{\tau + \lambda}.\end{aligned}$$

Ex: Poisson Process (4)

- To determine $\hat{\theta}_{MAP}$, consider

$$\ln f(\theta|N) = -\ln g(N) + N \ln \theta - \theta(\tau + \lambda) .$$

Differentiation w.r.t. θ yields

$$\frac{\partial}{\partial \theta} \ln f(\theta|Y) = 0 = \frac{N}{\theta} - (\tau + \lambda) .$$

So we obtain

$$\hat{\theta}_{MAP} = \frac{N}{\tau + \lambda} \neq \frac{N+1}{\tau + \lambda} = \hat{\theta}_{MMSE} .$$

- Note however that if $\tau \gg \lambda$ (observation duration much longer than the a priori expected time between observations), then with high probability $N \gg 1$ and hence $\hat{\theta}_{MAP} \approx \hat{\theta}_{MMSE} \approx \frac{N}{\tau}$, which is simply the sample average of the number of messages per second.



Statistical Signal Processing

Lecture 2

chapter 1: parameter estimation
stochastic parameters

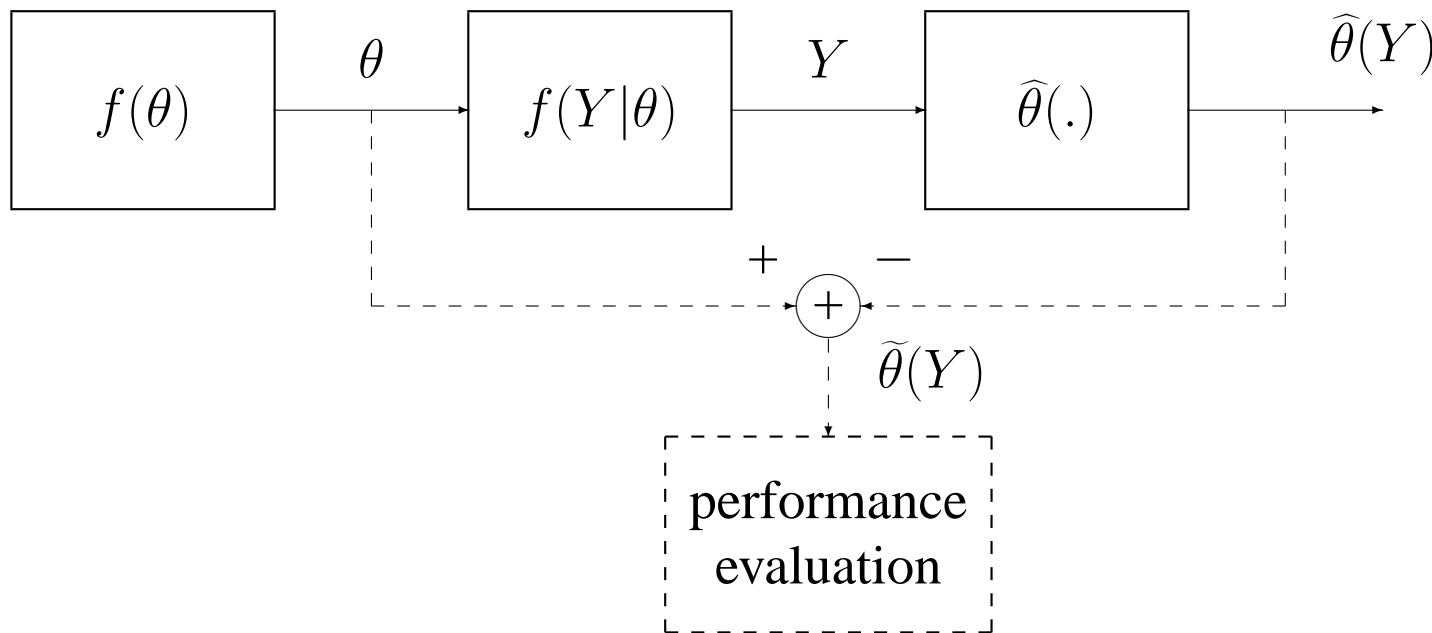
- the parameter estimation problem
- Bayes estimation: the MMSE, absolute value and uniform cost functions
- examples: Gaussian mean in Gaussian noise, Poisson process
- vector parameters
- Fischer Information Matrix

Vector Parameters

$$\bullet \theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_m \end{bmatrix}, \quad \widehat{\theta}(Y) = \begin{bmatrix} \widehat{\theta}_1(Y) \\ \vdots \\ \widehat{\theta}_m(Y) \end{bmatrix}, \quad \widetilde{\theta} = \widetilde{\theta}(\theta, Y) = \theta - \widehat{\theta}(Y)$$

- problem formulation:

- a prior distribution $f_{\boldsymbol{\theta}}(\theta)$
- a conditional distribution $f_{\mathbf{Y}|\boldsymbol{\theta}}(Y|\theta)$
- Bayes' rule : joint distribution $f_{\mathbf{Y},\boldsymbol{\theta}}(Y, \theta) = f_{\mathbf{Y}|\boldsymbol{\theta}}(Y|\theta)f_{\boldsymbol{\theta}}(\theta)$



Bayes Risk Function

- cost $\mathcal{C}(\theta, \hat{\theta}(Y))$
- $\mathcal{C}(\theta, \hat{\theta}(Y))$ often directly a function of the estimation error $\tilde{\theta}$ and in fact often a function of the length of the estimation error $\|\hat{\theta}\| = \sqrt{\hat{\theta}^T \hat{\theta}} = \sqrt{\sum_{i=1}^n \hat{\theta}_i^2}$
- We obtain the estimator function $\hat{\theta}(.)$ by minimizing the risk, which is the expected value of the cost:

$$\begin{aligned}\min_{\hat{\theta}(.)} \mathcal{R}(\hat{\theta}(.)) &= \min_{\hat{\theta}(.)} E \mathcal{C}(\theta, \hat{\theta}(Y)) = \min_{\hat{\theta}(.)} E_{\mathbf{Y}, \boldsymbol{\theta}} \mathcal{C}(\theta, \hat{\theta}(Y)) \\ &= \min_{\hat{\theta}(.)} E_{\mathbf{Y}} E_{\boldsymbol{\theta} | \mathbf{Y}} \mathcal{C}(\theta, \hat{\theta}(Y)) = E_{\mathbf{Y}} [\min_{\hat{\theta}(Y)} E_{\boldsymbol{\theta} | \mathbf{Y}} \mathcal{C}(\theta, \hat{\theta}(Y))] \\ &= E_{\mathbf{Y}} [\min_{\hat{\theta}(Y)} \mathcal{R}(\hat{\theta}(Y) | Y)] .\end{aligned}$$

- $\mathcal{R}(\hat{\theta}(.))$ is a weighted average of $\mathcal{R}(\hat{\theta}(Y) | Y)$, weighted by the nonnegative weighting function $f_{\mathbf{Y}}(Y)$. The minimum of $\mathcal{R}(\hat{\theta}(.))$ w.r.t. $\hat{\theta}(.)$ will hence be obtained by minimizing $\mathcal{R}(\hat{\theta}(Y) | Y)$ w.r.t. $\hat{\theta}(Y)$ for every Y .
- again $\mathcal{R}(\hat{\theta}(Y) | Y)$ depends on the posterior distribution $f_{\boldsymbol{\theta} | \mathbf{Y}}(\theta | Y)$.

Optimization w.r.t. Vector Parameters

- $g(\theta) = [g_1(\theta) \cdots g_l(\theta)]$: $1 \times l$ row vector function, its gradient w.r.t. θ :

$$\frac{\partial g(\theta)}{\partial \theta} = \begin{bmatrix} \frac{\partial g(\theta)}{\partial \theta_1} \\ \vdots \\ \frac{\partial g(\theta)}{\partial \theta_m} \end{bmatrix} = \begin{bmatrix} \frac{\partial g_1(\theta)}{\partial \theta_1} & \dots & \frac{\partial g_l(\theta)}{\partial \theta_1} \\ \vdots & & \vdots \\ \frac{\partial g_1(\theta)}{\partial \theta_m} & \dots & \frac{\partial g_l(\theta)}{\partial \theta_m} \end{bmatrix} \quad m \times l$$

If $g(\theta)$ is a scalar ($l = 1$), then $\frac{\partial g(\theta)}{\partial \theta}$ is a column vector of the same dimensions as θ .

- in particular: $\frac{\partial \theta^T}{\partial \theta} = \left[\frac{\partial \theta_j}{\partial \theta_i} \right] = [\delta_{ij}] = I_m$
- The gradient operator commutes with linear operations. Let X be $m \times 1$

$$\frac{\partial}{\partial \theta} (\theta^T X) = \left(\frac{\partial \theta^T}{\partial \theta} \right) X = I_m X = X.$$

- Since a scalar equals its transpose, we get

$$\frac{\partial}{\partial \theta} (X^T \theta) = \frac{\partial}{\partial \theta} (\theta^T X) = X$$



Optimization w.r.t. Vector Parameters (2)

- If A is $m \times l$: $\frac{\partial}{\partial \theta} (\theta^T A) = \left(\frac{\partial \theta^T}{\partial \theta} \right) A = I_m A = A$

- scalar case: $(uv)' = u'v + u\ v'$

- vector case: let $g(\theta)$ and $h(\theta)$ be $l \times 1$. Since

$$g^T(\theta)h(\theta) = (g^T(\theta)h(\theta))^T = h^T(\theta)g(\theta)$$

we get

$$\frac{\partial}{\partial \theta} (g^T(\theta)h(\theta)) = \left(\frac{\partial g^T(\theta)}{\partial \theta} \right) h(\theta) + \left(\frac{\partial h^T(\theta)}{\partial \theta} \right) g(\theta)$$

- Particular application with $g(\theta) = \theta$ and $h(\theta) = A\theta$:

$$\frac{\partial}{\partial \theta} (\theta^T A\theta) = \left(\frac{\partial \theta^T}{\partial \theta} \right) A\theta + \left(\frac{\partial \theta^T A^T}{\partial \theta} \right) \theta = (A + A^T)\theta$$

- When A is symmetric, this gradient reduces to $2A\theta$.

MMSE Criterion: Vector Parameters

- quadratic cost function $\mathcal{C}_{MMSE}(\theta, \bar{\theta}) = \|\bar{\theta}\|_2^2 = \bar{\theta}^T \bar{\theta} = \sum_{i=1}^n \bar{\theta}_i^2$

- minimizing the conditional Bayes risk :

$$\min_{\hat{\theta}(Y)} \mathcal{R}_{MMSE}(\hat{\theta}(Y)|Y) = \min_{\hat{\theta}(Y)} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(\theta|Y) (\theta - \hat{\theta})^T (\theta - \hat{\theta}) d\theta_1 \cdots d\theta_m$$

- extrema:

$$\begin{aligned} \frac{\partial}{\partial \hat{\theta}} \mathcal{R}_{MMSE}(\hat{\theta}|Y) &= \frac{\partial}{\partial \hat{\theta}} \int f(\theta|Y) (\theta - \hat{\theta})^T (\theta - \hat{\theta}) d\theta \\ &= \int f(\theta|Y) \left(\frac{\partial}{\partial \hat{\theta}} (\theta - \hat{\theta})^T (\theta - \hat{\theta}) \right) d\theta = 2 \int f(\theta|Y) (\hat{\theta} - \theta) d\theta = 0 \end{aligned}$$

- or hence $\hat{\theta}(Y) \underbrace{\int f(\theta|Y) d\theta}_{=1} = \int \theta f(\theta|Y) d\theta \Rightarrow \hat{\theta}_{MMSE}(Y) = E(\theta|Y)$

which is again the *mean* of the a posteriori distribution of θ given Y .

- extremum = minimum?

$$\begin{aligned} \text{Hessian} &= \left[\frac{\partial^2}{\partial \hat{\theta}_i \partial \hat{\theta}_j} \mathcal{R}_{MMSE}(\hat{\theta}|Y) \right] = \frac{\partial}{\partial \hat{\theta}} \left(\frac{\partial}{\partial \hat{\theta}} \mathcal{R}_{MMSE}(\hat{\theta}|Y) \right)^T \\ &= 2 \int f(\theta|Y) \left[\frac{\partial \hat{\theta}^T}{\partial \hat{\theta}} - \frac{\partial \theta^T}{\partial \hat{\theta}} \right] d\theta = 2I \int f(\theta|Y) d\theta = 2I > 0 \end{aligned}$$



MMSE Criterion: Vector Parameters (2)

- MMSE estimation commutes over linear transformations: with $\phi = A\theta$

$$\widehat{\phi}_{MMSE} = E(\phi|Y) = E(A\theta|Y) = A E(\theta|Y) = A \widehat{\theta}_{MMSE}$$

- orthogonality property of MMSE estimators:

$$\widehat{\theta}(\mathbf{Y}) = E(\boldsymbol{\theta}|\mathbf{Y}) \text{ iff } E((\boldsymbol{\theta} - \widehat{\theta}(\mathbf{Y})) g(\mathbf{Y})) = 0, \quad \forall g(.)$$

where $g(.)$ is a scalar function. Equivalently :

$$E(\widehat{\theta}(\mathbf{Y}) g(\mathbf{Y})) = E(\boldsymbol{\theta} g(\mathbf{Y})), \quad \forall g(.)$$

which represents an alternative way of defining $E(\boldsymbol{\theta}|\mathbf{Y})$.

- use orthogonality to show optimality: let $\widehat{\theta}(Y)$ be any function of Y ,

$$\begin{aligned} E \|\theta - \widehat{\theta}(Y)\|_2^2 &= E \|\theta - E(\theta|Y) + E(\theta|Y) - \widehat{\theta}(Y)\|_2^2 \\ &= E \|\theta - E(\theta|Y)\|_2^2 + \underbrace{E \|E(\theta|Y) - \widehat{\theta}(Y)\|_2^2}_{\geq 0} + 2 \underbrace{E ((\theta - E(\theta|Y))^T (E(\theta|Y) - \widehat{\theta}(Y)))}_{=0} \\ &\geq E \|\theta - E(\theta|Y)\|_2^2 \end{aligned}$$

- correlation matrices: $E (\theta - E(\theta|Y))(\theta - E(\theta|Y))^T \leq E (\theta - \widehat{\theta})(\theta - \widehat{\theta})^T = R_{\tilde{\theta}\tilde{\theta}}$

MAP Estimators: Vector Parameters

- introduce: a ball centered around θ_o with radius δ

$$\mathcal{B}_\delta(\theta_o) = \{\theta \in \Theta : \|\theta - \theta_o\|_2 \leq \delta\}$$

- Then the natural extension to the vector case of the uniform cost function is

$$\mathcal{C}_{UNIF}(\theta, \hat{\theta}) = \begin{cases} 0 & , \theta \in \mathcal{B}_\delta(\hat{\theta}) \\ 1 & , \theta \in \Theta \setminus \mathcal{B}_\delta(\hat{\theta}) \end{cases}$$

- The conditional Bayes risk becomes

$$\begin{aligned} \mathcal{R}_{UNIF}(\hat{\theta}(Y)|Y) &= \int_{\Theta} f(\theta|Y) \mathcal{C}_{UNIF}(\theta, \hat{\theta}) d\theta = \int_{\Theta \setminus \mathcal{B}_\delta(\hat{\theta})} f(\theta|Y) d\theta \\ &= \int_{\Theta} f(\theta|Y) d\theta - \int_{\mathcal{B}_\delta(\hat{\theta})} f(\theta|Y) d\theta = 1 - \int_{\mathcal{B}_\delta(\hat{\theta})} f(\theta|Y) d\theta \end{aligned}$$

- The optimization problem $\min_{\hat{\theta}(Y)} \mathcal{R}_{UNIF}(\hat{\theta}(Y)|Y)$ hence leads to

$$\max_{\hat{\theta}(Y)} \int_{\mathcal{B}_\delta(\hat{\theta})} f(\theta|Y) d\theta \approx \text{Vol}(\mathcal{B}_\delta(0)) \max_{\hat{\theta}(Y)} f(\hat{\theta}|Y)$$

the approximation becomes arbitrarily accurate as δ becomes small

MAP Estimators: Vector Parameters (2)

- This leads to the Maximum A Posteriori (likelihood) estimator

$$\widehat{\theta}_{MAP}(Y) = \arg \max_{\theta \in \Theta} f(\theta|Y) \quad \text{posterior likelihood}$$

- The same remarks as in the scalar case hold here also. In particular, one may equivalently obtain $\widehat{\theta}_{MAP}(Y)$ from the optimization problem

$$\widehat{\theta}_{MAP}(Y) = \arg \max_{\theta \in \Theta} \ln f(\theta|Y) \quad \text{posterior log likelihood}$$

- Under certain regularity conditions, $\widehat{\theta}_{MAP}(Y)$ can be found from

$$\frac{\partial}{\partial \theta} \ln f(\theta|Y) = 0 = \frac{\partial}{\partial \theta} \ln f(Y|\theta) + \frac{\partial}{\partial \theta} \ln f(\theta)$$

- Also MAP commutes over linear transformations: $\phi = A\theta$ (A invertible)

$$\begin{aligned} \widehat{\phi}_{MAP}(Y) &= \arg \max_{\phi} f_{\phi|Y}(\phi|Y) = A \arg \max_{\theta} f_{\phi|Y}(A\theta|Y) \\ &= A \arg \max_{\theta} \frac{1}{|\det A|} f_{\theta|Y}(\theta|Y) = A \widehat{\theta}_{MAP}(Y). \end{aligned}$$

This argument can be extended to the case in which $\dim \phi \neq \dim \theta$

Fisher Information Matrix

There exists a lower bound on the correlation matrix of the estimator errors. It is independent of the Bayes estimator (cost) used; it depends only on the posterior distribution. The lower bound is specified in terms of the *information matrix*, which should express in quantitative terms the information carried by the posterior distribution about the parameters θ . For such an information measure, the following properties are desirable:

- The information should increase as the *sensitivity* of $f(\theta|Y)$ to changes in θ increases. Hence, the information should be an increasing function of $\frac{\partial f(\theta|Y)}{\partial \theta}$ or of $\frac{\partial \ln f(\theta|Y)}{\partial \theta}$.
- The information should be *additive* in the sense that it should be the sum of the informations from the prior distribution ($f(\theta)$) and from the data ($f(Y|\theta)$). Furthermore if, given θ , Y_1 and Y_2 are independent ($f(Y_1, Y_2|\theta) = f(Y_1|\theta)f(Y_2|\theta)$), then the informations in Y_1 and Y_2 should add up.
- The information should be positive and should be insensitive to a change of sign of θ .
- The information should be a *deterministic* quantity.

Fisher Information Matrix (2)

- The information matrix is defined as

$$J = E \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right) \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right)^T$$

It can be shown to satisfy all the properties mentioned above.

- The perturbation of Mutual Information (Information Theory) w.r.t. a parameter can be expressed in terms of its Fisher Information.
- With $\frac{\partial \ln f(\theta|Y)}{\partial \theta} = \frac{1}{f(\theta|Y)} \frac{\partial f(\theta|Y)}{\partial \theta}$ we can write the Hessian of $\ln f(\theta|Y)$ as

$$\begin{aligned} \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right)^T &= \frac{1}{f^2(\theta|Y)} \left[f(\theta|Y) \frac{\partial}{\partial \theta} \left(\frac{\partial f(\theta|Y)}{\partial \theta} \right)^T - \left(\frac{\partial f(\theta|Y)}{\partial \theta} \right) \left(\frac{\partial f(\theta|Y)}{\partial \theta} \right)^T \right] \\ &= \frac{1}{f(\theta|Y)} \frac{\partial}{\partial \theta} \left(\frac{\partial f(\theta|Y)}{\partial \theta} \right)^T - \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right) \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right)^T \end{aligned}$$

- For the expectation of the first term, we get

$$\begin{aligned} E \frac{1}{f(\theta|Y)} \frac{\partial}{\partial \theta} \left(\frac{\partial f(\theta|Y)}{\partial \theta} \right)^T &= \int d\theta \int dY f(Y) \frac{\partial}{\partial \theta} \left(\frac{\partial f(\theta|Y)}{\partial \theta} \right)^T \\ &= \frac{\partial}{\partial \theta} \left(\frac{\partial}{\partial \theta} \int d\theta \int dY f(Y) f(\theta|Y) \right)^T = \frac{\partial}{\partial \theta} \left(\frac{\partial}{\partial \theta} 1 \right)^T = 0 \end{aligned}$$

Fisher Information Matrix (3)

- It follows that we can rewrite the information matrix as

$$J = -E \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right)^T$$

This expression will often allow us to obtain J more easily.

- Note also that

$$\frac{\partial \ln f(Y, \theta)}{\partial \theta} = \frac{\partial \ln f(\theta|Y)}{\partial \theta} + \underbrace{\frac{\partial \ln f(Y)}{\partial \theta}}_{=0} = \frac{\partial \ln f(\theta|Y)}{\partial \theta}$$

so that as long as derivatives are taken, we can interchange $f(Y, \theta)$ and $f(\theta|Y)$. Hence

$$\frac{\partial \ln f(Y, \theta)}{\partial \theta} = \frac{\partial \ln f(\theta|Y)}{\partial \theta} = \frac{\partial \ln f(Y|\theta)}{\partial \theta} + \frac{\partial \ln f(\theta)}{\partial \theta}$$

and

$$J = -E \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(\theta, Y)}{\partial \theta} \right)^T = \underbrace{-E \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(Y|\theta)}{\partial \theta} \right)^T}_{J_{data}} - E \underbrace{\frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(\theta)}{\partial \theta} \right)^T}_{J_{prior}}$$

Conditions on the Estimator Bias

- The (conditional) *bias* of an estimator $\hat{\theta}(Y)$ of θ is defined as

$$b_{\hat{\theta}}(\theta) = -E_{\mathbf{Y}|\boldsymbol{\theta}} \hat{\theta} = E_{\mathbf{Y}|\boldsymbol{\theta}} (\hat{\theta}(Y) - \theta) = E_{\mathbf{Y}|\boldsymbol{\theta}} \hat{\theta}(Y) - \theta$$

- An estimator will be called *unbiased* if either

$$E_{\boldsymbol{\theta}} b_{\hat{\theta}}(\theta) = 0 \Leftrightarrow E_{\boldsymbol{\theta}, \mathbf{Y}} \hat{\theta} = 0 \Leftrightarrow E_{\mathbf{Y}} \hat{\theta}(Y) = E_{\boldsymbol{\theta}} \theta = m_{\theta}$$

which means that the unconditional or average bias is zero, or

$$\lim_{\theta \rightarrow \partial \Theta} f(\theta) b_{\hat{\theta}}(\theta) = 0$$

where Θ is the domain for θ and $\partial \Theta$ is its boundary.

- **Lemma 0.1 (Unit Cross Correlation)** *If either condition above is satisfied, then*

$$E \frac{\partial \ln f(Y, \theta)}{\partial \theta} (\hat{\theta} - \theta)^T = I.$$

In words, the cross correlation matrix between $\frac{\partial \ln f(Y, \theta)}{\partial \theta}$ and the estimation error of any unbiased estimator is the identity matrix.

Inner Products

- An inner product $\langle \cdot, \cdot \rangle$ associates a real number $\langle x, y \rangle \in \mathcal{R}$ with two vectors x and y of the vector space \mathcal{V} we are considering, and it has the following properties:

1. linearity: $\forall \alpha_1, \alpha_2, \beta_1, \beta_2 \in \mathcal{R}, \quad \forall x, x_1, x_2, y, y_1, y_2 \in \mathcal{V}$:

$$\begin{aligned}\langle \alpha_1 x_1 + \alpha_2 x_2, y \rangle &= \alpha_1 \langle x_1, y \rangle + \alpha_2 \langle x_2, y \rangle \\ \langle x, \beta_1 y_1 + \beta_2 y_2 \rangle &= \langle x, y_1 \rangle \beta_1 + \langle x, y_2 \rangle \beta_2\end{aligned}\tag{1}$$

2. symmetry: $\langle x, y \rangle = \langle y, x \rangle$

3. non-degeneracy (of the norm induced by the inner product):

$\langle x, x \rangle = \|x\|^2 \geq 0$. If $\|x\| = 0$, then $x = 0$.

- One particular example is a space of random variables with the correlation as inner product: $\langle x, y \rangle = E xy$. Non-degeneracy subtlety:

$$E x^2 = 0 \Rightarrow x = 0 \text{ in m.s.}$$

$x = 0$ “in mean square”. Indeed, $E x^2 = m_x^2 + \sigma_x^2 = 0 \Rightarrow m_x = 0, \sigma_x^2 = 0$. This often (not always) implies $x = 0$ “almost surely” (a.s.) or “almost everywhere” (a.e.) or “with probability 1” (w.p. 1): $\Pr(x = 0) = 1$.

Matrix Inner Products

- We now consider a vector space \mathcal{V} in which the vectors have multiple components such that the inner product is a real matrix.
- Example 1: consider a vector space of random vectors with inner product $\langle X, Y \rangle = E XY^T$ where X and Y are column vectors of random variables (not necessarily with the same number of rows).
- Example 2: a vector space in which the “vectors” are $* \times k$ real matrices where k is fixed and $*$ ($\geq k$) is arbitrary. Inner product: $\langle X, Y \rangle = XY^T$.
- Matrix valued inner products satisfy the following properties, which are natural generalizations of the scalar case.

1. linearity: let $X, X_1, X_2 \in \mathcal{V}$ have m rows and $Y, Y_1, Y_2 \in \mathcal{V}$ have n rows. Then $\forall \alpha_1, \alpha_2 \in \mathcal{R}^{k \times m}$, $\forall \beta_1, \beta_2 \in \mathcal{R}^{l \times n}$, for any k and l ,

$$\begin{aligned}\langle \alpha_1 X_1 + \alpha_2 X_2, Y \rangle &= \alpha_1 \langle X_1, Y \rangle + \alpha_2 \langle X_2, Y \rangle \\ \langle X, \beta_1 Y_1 + \beta_2 Y_2 \rangle &= \langle X, Y_1 \rangle \beta_1^T + \langle X, Y_2 \rangle \beta_2^T\end{aligned}\tag{2}$$

2. symmetry: $\langle X, Y \rangle = \langle Y, X \rangle^T$

3. non-degeneracy: $\langle X, X \rangle = \|X\|^2 \geq 0$. If $\|X\|^2 = 0$, then $X = 0$.

Schur Complements

- **Lemma 0.2 (Schur Complements)** *Let X_1 and X_2 be vectors in a certain vector space with a certain inner product and denote $R_{ij} = \langle X_i, X_j \rangle$, $i, j = 1, 2$ so that $R_{ij} = R_{ji}^T$. Assume that R_{11} is nonsingular. Then, because of property 3 of the inner product (non-degeneracy), we have*

$$\begin{aligned}\|X_2 - R_{21}R_{11}^{-1}X_1\|^2 &= \langle X_2 - R_{21}R_{11}^{-1}X_1, X_2 - R_{21}R_{11}^{-1}X_1 \rangle \\ &= R_{22} - 2R_{21}R_{11}^{-1}R_{12} + R_{21}R_{11}^{-1}R_{11}R_{11}^{-1}R_{12} \\ &= R_{22} - R_{21}R_{11}^{-1}R_{12} \geq 0\end{aligned}$$

with equality iff $X_2 = R_{21}R_{11}^{-1}X_1$.

(matrix version of Cauchy-Schwarz inequality)

- The name for this lemma stems from the following congruence relation

$$\begin{aligned}<\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}, \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}> &= \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = \begin{bmatrix} I & O \\ R_{21}R_{11}^{-1} & I \end{bmatrix} \begin{bmatrix} R_{11} & 0 \\ 0 & R_{22} - R_{21}R_{11}^{-1}R_{12} \end{bmatrix} \begin{bmatrix} I & R_{11}^{-1}R_{12} \\ O & I \end{bmatrix} \\ &= \begin{bmatrix} I \\ R_{21}R_{11}^{-1} \end{bmatrix} R_{11} \begin{bmatrix} I \\ R_{21}R_{11}^{-1} \end{bmatrix}^T + \begin{bmatrix} 0 & 0 \\ 0 & R_{22} - R_{21}R_{11}^{-1}R_{12} \end{bmatrix}.\end{aligned}$$

(block LDU triangular factorization). The matrix $R_{22} - R_{21}R_{11}^{-1}R_{12}$ is called the *Schur complement* of R_{11} within the big matrix on the LHS .



Statistical Signal Processing

Lecture 3

chapter 1: parameter estimation
stochastic parameters

- Bayes estimation: the MMSE, absolute value and uniform cost functions
- examples: Gaussian mean in Gaussian noise, Poisson process
- vector parameters
- Fischer Information Matrix
- Cramer-Rao lower bound on the MSE
- Linear and Affine MMSE estimation

Cramer-Rao Bound

- **Theorem (CRB for Stochastic Parameters)** *If the estimator $\widehat{\theta}(Y)$ of θ is unbiased, then the correlation matrix of the parameter estimation errors $\tilde{\theta}$ is bounded below by the inverse of the information matrix:*

$$R_{\tilde{\theta}\tilde{\theta}} = E(\widehat{\theta} - \theta)(\widehat{\theta} - \theta)^T \geq J^{-1}$$

with equality ($\forall \theta \in \Theta$ or for one θ with $\frac{\partial b_{\widehat{\theta}}^T(\theta)}{\partial \theta} = 0$) iff

$$\widehat{\theta}(Y) - \theta = J^{-1} \frac{\partial \ln f(\theta|Y)}{\partial \theta} \text{ in m.s. (in mean-square)}$$

An estimator that achieves the lower bound is called *efficient*.

(note: $A \geq B \Leftrightarrow A - B \geq 0$: positive semi-definite)

- *Proof:* We shall apply the lemma on Schur complements to a vector space of random vectors with inner product $\langle X, Y \rangle = E XY^T$. In particular, we choose $X_1 = \frac{\partial \ln f(\theta|Y)}{\partial \theta}$ and $X_2 = \widehat{\theta} - \theta$. The Unit Cross Correlation lemma applies for an unbiased estimator and we find

$$E \left[\begin{bmatrix} \frac{\partial \ln f(\theta|Y)}{\partial \theta} \\ \widehat{\theta} - \theta \end{bmatrix} \right] \left[\begin{bmatrix} \frac{\partial \ln f(\theta|Y)}{\partial \theta} \\ \widehat{\theta} - \theta \end{bmatrix} \right]^T = \begin{bmatrix} J & I \\ I & R_{\tilde{\theta}\tilde{\theta}} \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \geq 0.$$

We have $R_{22} \geq R_{21}R_{11}^{-1}R_{12}$ with equality iff $X_2 = R_{21}R_{11}^{-1}X_1$.

Cramer-Rao Bound (2)

- *Efficiency* condition: implication for the posterior distribution? We can integrate

$$\frac{\partial \ln f(\theta|Y)}{\partial \theta} = J\hat{\theta}(Y) - J\theta$$

over θ to yield

$$\ln f(\theta|Y) = c(Y) + \hat{\theta}^T(Y) J \theta - \frac{1}{2} \theta^T J \theta = c'(Y) - \frac{1}{2} (\theta - \hat{\theta})^T J (\theta - \hat{\theta})$$

where $c(Y)$ and $c'(Y)$ are scalar functions of Y . This implies that $f(\theta|Y)$ is Gaussian. Using the constraint $\int_{\Theta} f(\theta|Y) d\theta = 1$, we can determine the proper integration constant and we get

$$f(\theta|Y) = \sqrt{\frac{\det J}{(2\pi)^m}} \exp\left(-\frac{1}{2}(\theta - \hat{\theta}(Y))^T J (\theta - \hat{\theta}(Y))\right)$$

or in other words $f(\theta|Y) \leftrightarrow \mathcal{N}(\hat{\theta}(Y), J^{-1})$. So the posterior distribution should be Gaussian *with constant covariance matrix*. In that case, the posterior mean (which may depend on the data) is an efficient estimator. Note that neither the prior distribution nor the conditional distribution $f(Y|\theta)$ need to be Gaussian for the posterior distribution to be Gaussian. Note also: $\hat{\theta}(Y) = \hat{\theta}_{MMSE}(Y)$.

Cramer-Rao Bound (3)

- *Additivity of the information matrix:* using Bayes' rule, we can write

$$J = -E \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right)^T = -E \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(\theta)}{\partial \theta} \right)^T - E \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(Y|\theta)}{\partial \theta} \right)^T = J_{prior} + J_Y$$

- If $f(\theta)$ is Gaussian then the corresponding information matrix J is the inverse of the covariance matrix C : $J = C^{-1}$.
- If the different data y_i are independent given θ , then

$$f(Y|\theta) = \prod_{i=1}^n f(y_i|\theta) \Rightarrow J_Y = \sum_{i=1}^n J_{y_i}$$

If furthermore the data are i.i.d. given θ , then $J_Y = n J_{y_1}$.

- $\widehat{\theta}_{MAP}$ is generally easier to determine than $\widehat{\theta}_{MMSE}$. If $\widehat{\theta}_{MAP}$ achieves efficiency (and hence is unbiased), then it equals $\widehat{\theta}_{MMSE}$ since $\widehat{\theta}_{MMSE}$ minimizes the MSE criterion but the minimum value of the MSE criterion cannot be lower than $\text{tr}\{J^{-1}\}$.

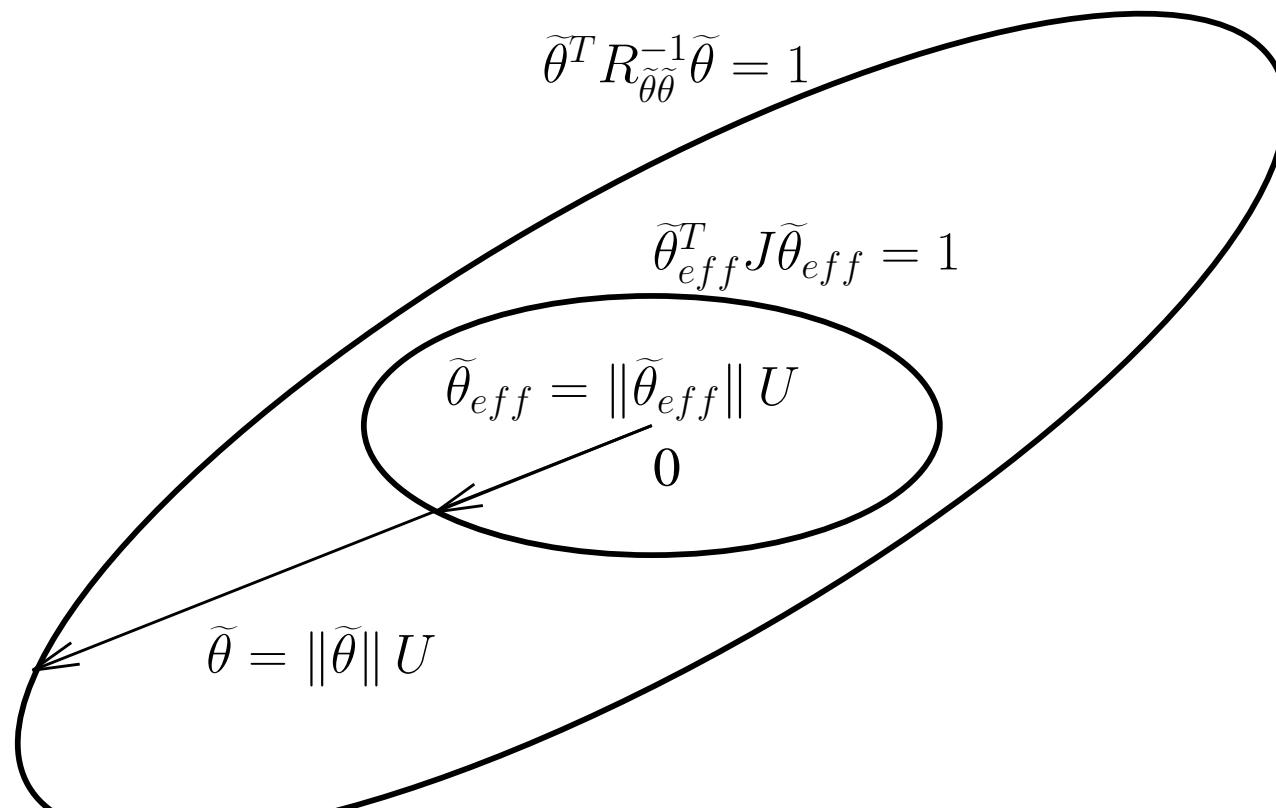
$$\text{MSE} = E \|\widehat{\theta}(Y) - \theta\|^2 = \text{tr} R_{\tilde{\theta}\tilde{\theta}} \quad \text{MMSE} \leftrightarrow \min R_{\tilde{\theta}\tilde{\theta}} \geq J^{-1}$$

- So in general, $R_{\tilde{\theta}\tilde{\theta}}^{MAP} \geq R_{\tilde{\theta}\tilde{\theta}}^{MMSE} \geq J^{-1}$
- $\widehat{\theta}_{MMSE}$ is unbiased: $\widehat{\theta} = E(\theta|Y) = E_{\boldsymbol{\theta}|Y} \theta \Rightarrow E_Y \widehat{\theta} = E_Y E_{\boldsymbol{\theta}|Y} \theta = E_{\boldsymbol{\theta},Y} \theta$

Cramer-Rao Bound (4)

Concentration Ellipsoids

- Gaussian random vector: concentration ellipsoid = volume in which the random vector occurs with a certain probability.
- CRB \Rightarrow the concentration ellipsoid for any unbiased estimator lies outside or on the concentration ellipsoid of an efficient estimator. This means that the estimation errors are the most concentrated in space (around the origin) for an efficient estimator.



Cramer-Rao Bound (5)

Example 1.6 Gaussian mean in Gaussian noise - Example 1.4 Continued

- In example 1.4, the posterior distribution was Gaussian with constant variance.
- We had $\hat{\theta}_{MMSE} = \hat{\theta}_{MAP} = \hat{\theta}_{ABS}$ which is an efficient estimator.
- We found indeed for the estimation error correlation

$$\frac{1}{\sigma_{\hat{\theta}}^2} = J = J_{prior} + J_Y = \underbrace{\frac{1}{\sigma_\theta^2}}_{\text{efficiency}} + \frac{n}{\sigma_v^2}$$

which decomposes indeed into the prior information and n times the information in one measurement (all distributions involved are Gaussian).

Linear MMSE Estimation

- MMSE criterion is a desirable criterion but the resulting Bayes estimator $E[\theta|Y]$ may be complicated to derive.
- In practice often: suboptimal estimators, e.g. restrict the estimator to be a linear function of the data.
- Remark: the MMSE criterion for a vector parameter decomposes:

$$\min_{\hat{\theta}} E(\hat{\theta} - \theta)^T(\hat{\theta} - \theta) = \min_{\hat{\theta}} E \sum_{i=1}^m (\hat{\theta}_i - \theta_i)^2 = \sum_{i=1}^m \min_{\hat{\theta}_i} E(\hat{\theta}_i - \theta_i)^2$$

Hence it suffices to concentrate on the estimator $\hat{\theta}_i$ of a scalar component θ_i of θ and then $\hat{\theta} = [\hat{\theta}_1 \cdots \hat{\theta}_m]^T$.

- Linear Estimators : constrain $\hat{\theta}_i(Y)$ to be linear:

$$\hat{\theta}_i(Y) = F_i^T Y = [f_{i,1} \cdots f_{i,n}] \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \sum_{k=1}^n f_{i,k} y_k$$

where F_i is a $n \times 1$ vector of combination coefficients (of the same dimension as the data vector Y).



Linear MMSE Estimators

- MSE risk function (of F_i now !)

$$\mathcal{R}_{LMMSE}(F_i) = E(\theta_i - \hat{\theta}_i)^2 = E(\theta_i - F_i^T Y)^2$$

and we shall obtain F_i as $F_i = \arg \min_{F_i} \mathcal{R}_{LMMSE}(F_i)$.

- Setting the gradient equal to zero leads to

$$\frac{\partial}{\partial F_i} \mathcal{R}_{LMMSE}(F_i) = -2E(\theta_i - F_i^T Y)Y = 0 \Rightarrow E(\theta_i - F_i^T Y)Y^T = 0$$

$$\Rightarrow F_i^T = (E\theta_i Y^T)(EYY^T)^{-1} = R_{\theta_i Y} R_{YY}^{-1}$$

- The Hessian can be verified to be

$$\frac{\partial}{\partial F_i} \left(\frac{\partial}{\partial F_i} \mathcal{R}_{LMMSE}(F_i) \right)^T = 2R_{YY} > 0$$

Hence, the unique extremum is indeed the global minimum.

Linear MMSE Estimators (2)

- This can be done for each component θ_i of θ independently to yield the minimizer of the overall MSE criterion

$$\widehat{\theta} = \begin{bmatrix} \widehat{\theta}_1 \\ \vdots \\ \widehat{\theta}_m \end{bmatrix} = \begin{bmatrix} F_1^T Y \\ \vdots \\ F_m^T Y \end{bmatrix} = F Y = R_{\theta Y} R_{YY}^{-1} Y = \begin{bmatrix} R_{\theta_1 Y} R_{YY}^{-1} Y \\ \vdots \\ R_{\theta_m Y} R_{YY}^{-1} Y \end{bmatrix}$$

where $F = [F_1 \cdots F_m]^T$.

- correlation matrix of the parameter estimation errors:

$$\begin{aligned} R_{\tilde{\theta}\tilde{\theta}}(F) &= E(\theta - \widehat{\theta})(\theta - \widehat{\theta})^T = E(\theta - FY)(\theta^T - Y^T F^T) \\ &= R_{\theta\theta} - R_{\theta Y} F^T - F R_{Y\theta} + F R_{YY} F^T \end{aligned}$$

- Evaluated at the minimum, this gives

$$R_{\tilde{\theta}\tilde{\theta}}^{LMMSE} = R_{\tilde{\theta}\tilde{\theta}}(R_{\theta Y} R_{YY}^{-1}) = R_{\theta\theta} - R_{\theta Y} R_{YY}^{-1} R_{Y\theta}$$

- the MSE criterion is just the trace of this correlation matrix, hence

$$\min_F E \underbrace{(\theta - FY)^T (\theta - FY)}_{=\|\theta - FY\|^2 = \|\tilde{\theta}\|^2 = \text{tr}\{\tilde{\theta}\tilde{\theta}^T\}} = \text{tr}\{R_{\tilde{\theta}\tilde{\theta}}^{LMMSE}\} = \text{tr}\{R_{\theta\theta} - R_{\theta Y} R_{YY}^{-1} R_{Y\theta}\}$$

Affine MMSE Estimators

- for random variables with non-zero mean, it may be advantageous to add a constant term to the linear estimator:

$$\hat{\theta}_i(Y) = F_i^T Y + g_i \quad \text{where } g_i \text{ is a scalar}$$

- The MSE risk function is now

$$\mathcal{R}_{AMMSE}(F_i, g_i) = E(\theta_i - \hat{\theta}_i)^2 = E(\theta_i - F_i^T Y - g_i)^2$$

and we shall obtain F_i and g_i from the optimization problem

$$\min_{F_i, g_i} \mathcal{R}_{AMMSE}(F_i, g_i).$$

- Setting the gradients equal to zero leads to

$$\begin{aligned} \frac{\partial}{\partial F_i} \mathcal{R}_{AMMSE}(F_i, g_i) &= 0 = -2E(\theta_i - F_i^T Y - g_i)Y && | 1 \\ \frac{\partial}{\partial g_i} \mathcal{R}_{AMMSE}(F_i, g_i) &= 0 = -2E(\theta_i - F_i^T Y - g_i) && | -m_Y \end{aligned}$$

- From the second equation, we get

$$g_i = m_{\theta_i} - F_i^T m_Y$$

Affine MMSE Estimators (2)

- By forming the indicated linear combination of both equations, we get

$$\begin{aligned}
 0 &= E(\theta_i - F_i^T Y - g_i)(Y - m_Y) = E(\theta_i - m_{\theta_i} - F_i^T(Y - m_Y))(Y - m_Y) \\
 &= E(Y - m_Y)(\theta_i - m_{\theta_i} - (Y - m_Y)^T F_i) \\
 &= E\{(Y - m_Y)(\theta_i - m_{\theta_i})\} - E\{(Y - m_Y)(Y - m_Y)^T\} F_i = C_{Y\theta_i} - C_{YY}F_i
 \end{aligned}$$

which leads to

$$\widehat{\theta}_i(Y) = F_i^T Y + g_i = m_{\theta_i} + C_{\theta_i Y} C_{YY}^{-1}(Y - m_Y)$$

- The Hessian can be verified to be (use $R_{XY} = C_{XY} + m_X m_Y^T = R_{YX}^T$, $C_{XY} = C_{YX}^T$)

$$\begin{aligned}
 &\left[\begin{array}{cc} \frac{\partial}{\partial F_i} \left(\frac{\partial}{\partial F_i} \mathcal{R}_{AMMSE} \right)^T & \frac{\partial}{\partial F_i} \left(\frac{\partial}{\partial g_i} \mathcal{R}_{AMMSE} \right)^T \\ \frac{\partial}{\partial g_i} \left(\frac{\partial}{\partial F_i} \mathcal{R}_{AMMSE} \right)^T & \frac{\partial}{\partial g_i} \left(\frac{\partial}{\partial g_i} \mathcal{R}_{AMMSE} \right)^T \end{array} \right] \left\{ \begin{array}{l} F_i = C_{\theta_i Y} C_{YY}^{-1} \\ g_i = m_{\theta_i} - C_{\theta_i Y} C_{YY}^{-1} m_Y \end{array} \right. \\
 &= 2 \begin{bmatrix} R_{YY} & m_Y \\ m_Y^T & 1 \end{bmatrix} = 2 \begin{bmatrix} C_{YY} & 0 \\ 0 & 0 \end{bmatrix} + 2 \begin{bmatrix} m_Y \\ 1 \end{bmatrix} \begin{bmatrix} m_Y \\ 1 \end{bmatrix}^T \\
 &= 2 \begin{bmatrix} I & m_Y \\ 0 & 1 \end{bmatrix} \begin{bmatrix} C_{YY} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I & m_Y \\ 0 & 1 \end{bmatrix}^T > 0
 \end{aligned}$$

Affine MMSE Estimators (3)

- This can again be done for each component θ_i of θ independently to yield the minimizer of the overall MSE criterion

$$\widehat{\theta} = \begin{bmatrix} \widehat{\theta}_1 \\ \vdots \\ \widehat{\theta}_m \end{bmatrix} = \begin{bmatrix} F_1^T Y + g_1 \\ \vdots \\ F_m^T Y + g_m \end{bmatrix} = F Y + g = m_\theta + C_{\theta Y} C_{YY}^{-1} (Y - m_Y)$$

where $F = [F_1 \cdots F_M]^T$ and $g = [g_1 \cdots g_m]^T$.

- the affine estimator is unbiased: $E_Y \widehat{\theta} = m_\theta$ or $E_{Y,\theta} \widehat{\theta} = 0$
- the correlation matrix of the parameter estimation errors evaluated for the optimal estimator is

$$\begin{aligned} R_{\widehat{\theta}\widehat{\theta}}^{AMMSE} &= E (\theta - \widehat{\theta})(\theta - \widehat{\theta})^T \\ &= E [\theta - m_\theta - C_{\theta Y} C_{YY}^{-1} (Y - m_Y)][\theta - m_\theta - C_{\theta Y} C_{YY}^{-1} (Y - m_Y)]^T \\ &= C_{\theta\theta} - C_{\theta Y} C_{YY}^{-1} C_{Y\theta} = C_{\widehat{\theta}\widehat{\theta}}^{AMMSE} \end{aligned}$$

- the MSE criterion is just the trace of this correlation matrix, hence

$$\min_{F,g} E (\theta - F Y - g)^T (\theta - F Y - g) = \text{tr} \{C_{\widehat{\theta}\widehat{\theta}}^{AMMSE}\} = \text{tr} \{C_{\theta\theta} - C_{\theta Y} C_{YY}^{-1} C_{Y\theta}\}$$

Linear MMSE Estimation: Remarks

- when θ and Y are jointly Gaussian, then

$$\hat{\theta}_{MMSE} = E[\theta|Y] = m_\theta + C_{\theta Y} C_{YY}^{-1} (Y - m_Y) = \hat{\theta}_{AMMSE}$$

Hence, in the case of Gaussian variables, the Affine MMSE estimator is optimal!

- Whereas general Bayes estimators require the knowledge of the complete joint distribution $f(Y, \theta)$, the Linear and Affine MMSE estimators only require the joint first and second order moments of θ and Y .
- If θ and Y have non-zero means, it is advantageous to use an affine estimator. Indeed, using $R_{XY} = C_{XY} + m_X m_Y^T$ and the Matrix Inversion Lemma on $R_{YY}^{-1} = (C_{YY} + m_Y m_Y^T)^{-1}$, one can show that

$$\begin{aligned} R_{\tilde{\theta}\tilde{\theta}}^{LMMSE} &= R_{\tilde{\theta}\tilde{\theta}}^{AMMSE} + \underbrace{(m_\theta - C_{\theta Y} C_{YY}^{-1} m_Y)(m_Y^T C_{YY}^{-1} m_Y + 1)^{-1} (m_\theta - C_{\theta Y} C_{YY}^{-1} m_Y)^T}_{\geq 0} \\ &\geq R_{\tilde{\theta}\tilde{\theta}}^{AMMSE} = C_{\tilde{\theta}\tilde{\theta}}^{AMMSE}. \end{aligned}$$

By taking the trace of these expressions, one finds that the affine estimator leads to lower MSE than the linear estimator when $m_Y \neq 0$ or $m_\theta \neq 0$.

If $m_\theta = C_{\theta Y} C_{YY}^{-1} m_Y$, check that we should have $\hat{\theta}_{LMMSE} = \hat{\theta}_{AMMSE}$.

Linear MMSE Estimation: Remarks (2)

Linear MMSE estimator simpler than the Affine MMSE estimator
 \Rightarrow reduce Affine MMSE estimation to Linear MMSE estimation.

- Method 1: introduce a linear estimator for an augmented problem

$$\theta' = \theta, \quad Y' = \begin{bmatrix} Y \\ 1 \end{bmatrix}, \quad \widehat{\theta}'_L = \underbrace{F'}_{1 \times (n+1)} \underbrace{Y'}_{(n+1) \times 1} = [F \ g] \begin{bmatrix} Y \\ 1 \end{bmatrix} = F Y + g = \widehat{\theta}_A$$

The Linear MMSE estimator for the augmented problem $\{\theta, Y'\}$, namely

$$\widehat{\theta}'_{LMMSE} = R_{\theta Y'} R_{Y' Y'}^{-1} Y' = m_\theta + C_{\theta Y} C_{Y Y}^{-1} (Y - m_Y) = \widehat{\theta}_{AMMSE}$$

is in fact the Affine MMSE estimator for the original problem $\{\theta, Y\}$. (Exo!)

- Method 2: When $m_\theta = 0$ and $m_Y = 0$, the affine estimator reduces to the linear estimator \Rightarrow centralize the variables before further treatment

$$\begin{cases} \theta' = \theta - m_\theta \\ Y' = Y - m_Y . \end{cases}$$

Then the linear and the affine MMSE estimators coincide

$$\widehat{\theta}' = R_{\theta' Y'} R_{Y' Y'}^{-1} Y' = C_{\theta' Y'} C_{Y' Y'}^{-1} Y' = C_{\theta Y} C_{Y Y}^{-1} Y'$$

henceforth assume zero mean.

Linear MMSE Estimation: Remarks (3)

- Except in the jointly Gaussian case, the MSE increases by imposing the constraint of linearity on the estimator. This increase can be displayed by decomposing the MSE associated with a linear estimator as follows:

$$\begin{aligned}
 E\|\theta - FY\|^2 &= E(\theta - FY)^T(\theta - FY) \\
 &= E(\theta - E[\theta|Y] + E[\theta|Y] - FY)^T(\theta - E[\theta|Y] + E[\theta|Y] - FY) \\
 &= E(\theta - E[\theta|Y])^T(\theta - E[\theta|Y]) + \underbrace{E(E[\theta|Y] - FY)^T(E[\theta|Y] - FY)}_{\geq 0} \\
 &\quad + 2 \underbrace{E(\theta - E[\theta|Y])^T(E[\theta|Y] - FY)}_{= 0} \geq E(\theta - E[\theta|Y])^T(\theta - E[\theta|Y])
 \end{aligned}$$

- the difference between the LMMSE and the MMSE is the MSE in approximating the conditional mean $E[\theta|Y]$ by a linear function FY .
- In fact, the best linear approximation of $E[\theta|Y]$ is also the best linear approximation of θ since the above implies

$$R_{\theta Y} R_{YY}^{-1} = \arg \min_F E_{Y,\theta} \|\theta - FY\|^2 = \arg \min_F E_Y \|E[\theta|Y] - FY\|^2$$

- And for any F :

$$E_{Y,\theta} \|\theta - E[\theta|Y]\|^2 = E_{Y,\theta} \|\theta - FY\|^2 - E_Y \|E[\theta|Y] - FY\|^2$$

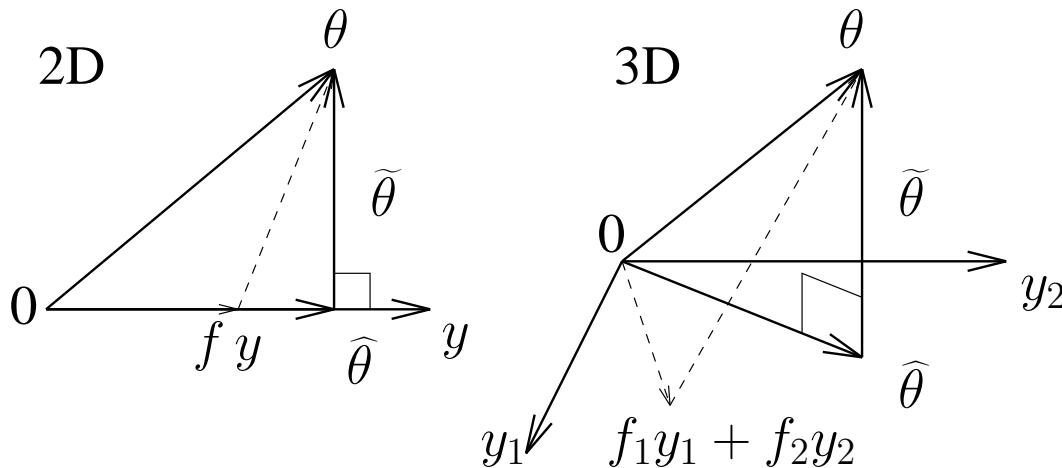
Orthogonality Principle of LMMSE

- Let θ, y_1, \dots, y_n be random variables that span a $(n+1)$ -dimensional vector space with inner product $\langle x, y \rangle = E xy$. We shall form a linear estimate (approximation) of θ in terms of y_1, \dots, y_n : $\hat{\theta} = F^T Y = \sum_{i=1}^n f_i y_i$ determine the combination coefficients f_i by minimizing the MSE

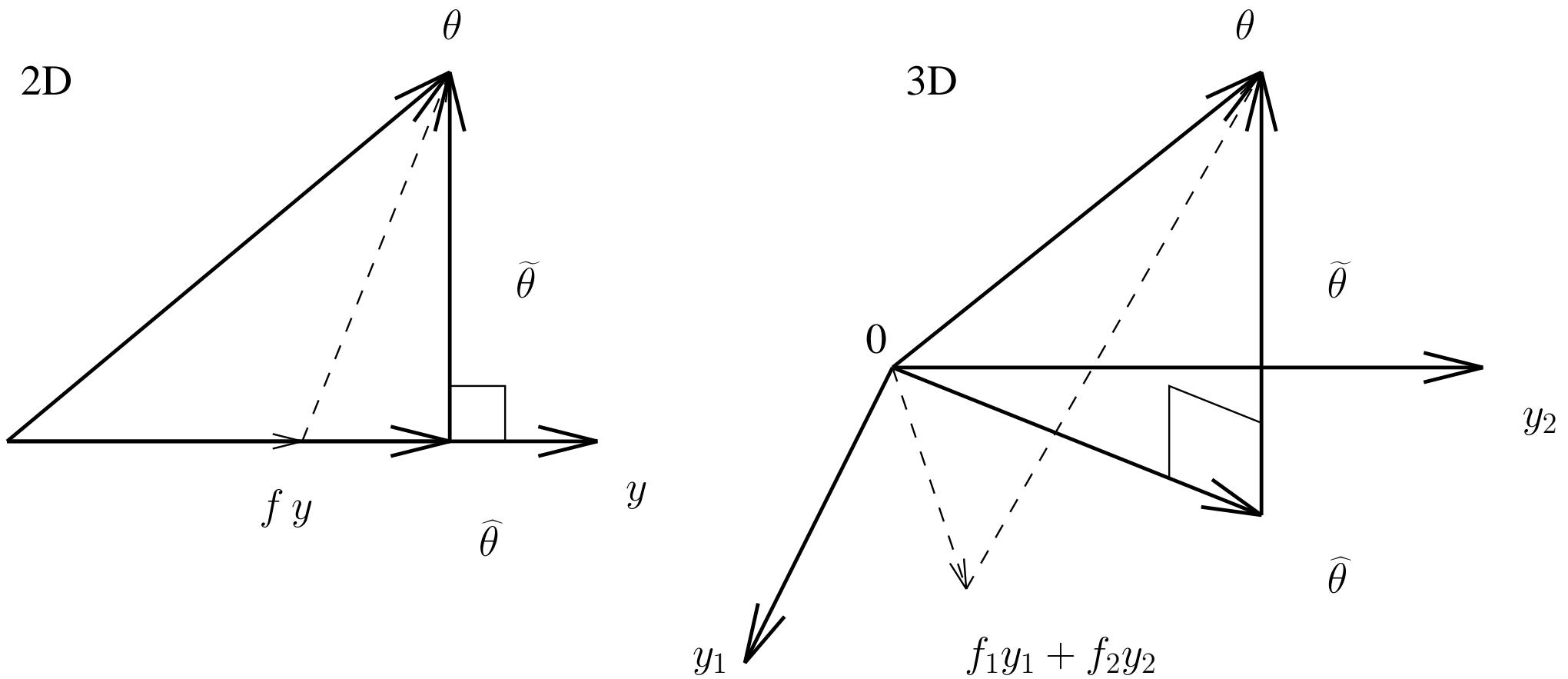
$$\min_{f_i} E (\theta - \hat{\theta})^2 = \min_{f_i} E (\theta - \sum_{i=1}^n f_i y_i)^2 = \min_{f_i} \|\theta - \hat{\theta}\|^2 \Rightarrow$$

$$\frac{\partial}{\partial f_i} E (\theta - \hat{\theta})^2 = -2 E (\theta - \hat{\theta}) y_i = 0 \Rightarrow \langle \theta - \hat{\theta}, y_i \rangle = 0, \quad i = 1, \dots, n$$

Hence the LMMSE estimate also satisfies the orthogonality conditions.



Orthogonality Principle of LMMSE



Orthogonality Principle of LMMSE (2)

- When $\theta = [\theta_1 \dots \theta_m]^T$ is a random vector, then we can again consider the space spanned by the variables $\theta_1, \dots, \theta_m, y_1, \dots, y_n$. We shall now more generally consider vectors of random variables and the matrix inner product between them. If X and Y are random vectors, then we take their inner product to be $\langle X, Y \rangle = E XY^T = R_{XY}$.
- A linear estimator for θ in terms of Y is now

$$\widehat{\theta} = F Y \quad \text{where } F \text{ is } m \times n$$

- We cannot follow directly the same path as for the case of a scalar θ since we have not seen how to take gradients w.r.t. a matrix (and even less how to consider the corresponding Hessian). Geometrical intuition: optimal F is the one that satisfies the orthogonality condition:

$$0 = \langle \theta - \widehat{\theta}, Y \rangle = \langle \theta - FY, Y \rangle = \langle \theta, Y \rangle - F \langle Y, Y \rangle$$

so we find the result we found before using a different route:

$$F = \langle \theta, Y \rangle \langle Y, Y \rangle^{-1} = R_{\theta Y} R_{YY}^{-1}$$

Orthogonality Principle of LMMSE (3)

- We can now show that this F which satisfies the orthogonality condition minimizes the correlation matrix $R_{\tilde{\theta}\tilde{\theta}}$ of the estimation errors. Indeed, let KY be any other linear estimator of θ . With $\langle X, Y \rangle = R_{XY}$, $\|X\|^2 = R_{XX}$, we get

$$\begin{aligned}
 R_{\tilde{\theta}\tilde{\theta}}(K) &= \|\theta - KY\|^2 = \langle \theta - KY, \theta - KY \rangle \\
 &= \langle \theta - FY + FY - KY, \theta - FY + FY - KY \rangle \\
 &= \|\theta - FY\|^2 + \|(F - K)Y\|^2 \\
 &\quad + \underbrace{\langle \theta - FY, Y \rangle}_{=0} (F - K)^T + (F - K) \underbrace{\langle Y, \theta - FY \rangle}_{=0} \\
 &= \|\theta - FY\|^2 + \underbrace{(F - K) \|Y\|^2 (F - K)^T}_{\geq 0 \text{ } (=0 \text{ iff } F=K \text{ } (\|Y\|^2=R_{YY}>0))} \\
 &\geq \|\theta - FY\|^2 = R_{\tilde{\theta}\tilde{\theta}}(F).
 \end{aligned}$$

Since the MSE is the trace of $R_{\tilde{\theta}\tilde{\theta}}$, this shows again that $\hat{\theta} = R_{\theta Y} R_{YY}^{-1} Y$ is the LMMSE estimator, with a proof based on the orthogonality property. This is an example where the whole $R_{\tilde{\theta}\tilde{\theta}}$ gets minimized instead of just its trace (= MSE).



Bayesian Linear Model

- $Y = H\theta + V$, $\theta \sim \mathcal{N}(m_\theta, C_{\theta\theta})$ and $V \sim \mathcal{N}(0, C_{VV})$ independent
- $\begin{bmatrix} \theta \\ Y \end{bmatrix} = \begin{bmatrix} I & 0 \\ H & I \end{bmatrix} \begin{bmatrix} \theta \\ V \end{bmatrix}$ jointly Gaussian, $Y - m_Y = H(\theta - m_\theta) + V$
- $C_{YY} = H C_{\theta\theta} H^T + C_{VV}$, $C_{Y\theta} = H C_{\theta\theta}$
- Gauss-Markov theorem:

$$f(\theta|Y) \leftrightarrow \mathcal{N}(m_{\theta|Y}, C_{\theta|Y})$$

$$\begin{aligned} m_{\theta|Y} &= m_\theta + C_{\theta Y} C_{YY}^{-1} (Y - m_Y) \\ &= m_\theta + C_{\theta\theta} H^T (H C_{\theta\theta} H^T + C_{VV})^{-1} (Y - H m_\theta) \\ &\stackrel{\text{ML}}{=} m_\theta + (C_{\theta\theta}^{-1} + H^T C_{VV}^{-1} H)^{-1} H^T C_{VV}^{-1} (Y - H m_\theta) \end{aligned}$$

$$\begin{aligned} C_{\theta|Y} &= C_{\theta\theta} - C_{\theta Y} C_{YY}^{-1} C_{Y\theta} = C_{\theta\theta} - C_{\theta\theta} H^T (H C_{\theta\theta} H^T + C_{VV})^{-1} H C_{\theta\theta} \\ \stackrel{\text{ML}}{\Rightarrow} C_{\theta|Y}^{-1} &= C_{\theta\theta}^{-1} + H^T C_{VV}^{-1} H \end{aligned}$$

- θ, Y jointly Gaussian $\Rightarrow \hat{\theta}_{MMSE} = \hat{\theta}_{AMMSE} = m_{\theta|Y}$

Bayesian Linear Model (2)

- $f(\theta|Y)$ Gaussian \Rightarrow (A)MMSE estimator = efficient:

$$R_{\tilde{\theta}\tilde{\theta}} = C_{\theta|Y} = J^{-1}, \quad J = J_{prior} + J_Y = C_{\theta\theta}^{-1} + H^T C_{VV}^{-1} H$$

- if noise samples uncorrelated: C_{VV} diagonal: $C_{VV} = \text{diag}\{\sigma_{v_1}^2 \cdots \sigma_{v_n}^2\}$, then the information from the data also decomposes:

$$J_Y = H^T C_{VV}^{-1} H = \sum_{k=1}^n H_{k,:}^T \sigma_{v_k}^{-2} H_{k,:} \quad \text{rank}(J_Y) \leq \min\{n, m\}$$

- Note: for $n < m$, J_Y has rank $\leq n$ and hence the $m \times m$ matrix J_Y is singular. However, due to $J_{prior} > 0$, $J > 0$ and hence non-singular \Rightarrow importance of prior information when only little measurement data is available.
- On the other hand, as $n \rightarrow \infty$, J_{prior} becomes of negligible importance compared to J_Y . So in general, the influence of the prior information disappears asymptotically as the number of measurements becomes large.

Recap: Bayes Parameter Estimation

- obtain the estimator $\hat{\theta}(\cdot)$ by minimizing the risk, the average cost:

$$\min_{\hat{\theta}(\cdot)} \mathcal{R}(\hat{\theta}(\cdot)) = E_{\mathbf{Y}} \left[\min_{\hat{\theta}(Y)} E_{\boldsymbol{\theta}|\mathbf{Y}} \mathcal{C}(\theta, \hat{\theta}(Y)) \right] = E_{\mathbf{Y}} \left[\min_{\hat{\theta}(Y)} \mathcal{R}(\hat{\theta}(Y)|Y) \right]$$

requires the posterior distribution $f_{\boldsymbol{\theta}|\mathbf{Y}}(\theta|Y) = f_{\mathbf{Y}|\boldsymbol{\theta}}(Y|\theta) f_{\boldsymbol{\theta}}(\theta) / f_{\mathbf{Y}}(Y)$

- quadratic cost function (risk=MSE): $\hat{\theta}_{MMSE}(Y) = E(\theta|Y)$
- uniform cost function : $\hat{\theta}_{MAP}(Y) = \arg \max_{\theta} f(\theta|Y) = \arg \max_{\theta} f(Y|\theta) f(\theta)$
- information matrix $J = E \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right) \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right)^T = -E \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(\theta|Y)}{\partial \theta} \right)^T$
- $\hat{\theta}$ unbiased if $E_{\boldsymbol{\theta}} b_{\hat{\theta}}(\theta) = 0$ or $\lim_{\theta \rightarrow \partial \Theta} f(\theta) b_{\hat{\theta}}(\theta) = 0$, $b_{\hat{\theta}}(\theta) = -E_{\mathbf{Y}|\boldsymbol{\theta}} \hat{\theta}$. Then
- CRB: $R_{\tilde{\theta}\tilde{\theta}} \geq J^{-1}$, = (efficiency) iff $\hat{\theta}(Y) - \theta = J^{-1} \frac{\partial \ln f(\theta|Y)}{\partial \theta}$, $f(\theta|Y)$ Gaussian
- in general: $R_{\tilde{\theta}_{MAP}\tilde{\theta}_{MAP}} \geq R_{\tilde{\theta}_{MMSE}\tilde{\theta}_{MMSE}} \geq J^{-1}$
- linear MMSE: $\hat{\theta}_{LMMSE} = R_{\theta Y} R_{YY}^{-1} Y$
- $R_{\tilde{\theta}_{LMMSE}\tilde{\theta}_{LMMSE}} \geq R_{\tilde{\theta}_{MMSE}\tilde{\theta}_{MMSE}}$, Gaussian case: $\hat{\theta}_{AMMSE} = \hat{\theta}_{MMSE} = \hat{\theta}_{MAP}$
- special Gaussian case: linear model: $Y = H\theta + V$



Statistical Signal Processing

Lecture 4

chapter 1: parameter estimation
deterministic parameters

- some optimality properties
- Maximum Likelihood estimation
- Fischer Information Matrix
- Cramer-Rao lower bound on the MSE

Deterministic Parameter Estimation

Two points of view:

- the parameters θ are unknown deterministic quantities
- the parameters θ are stochastic, but their prior distribution $f(\theta)$ is unknown

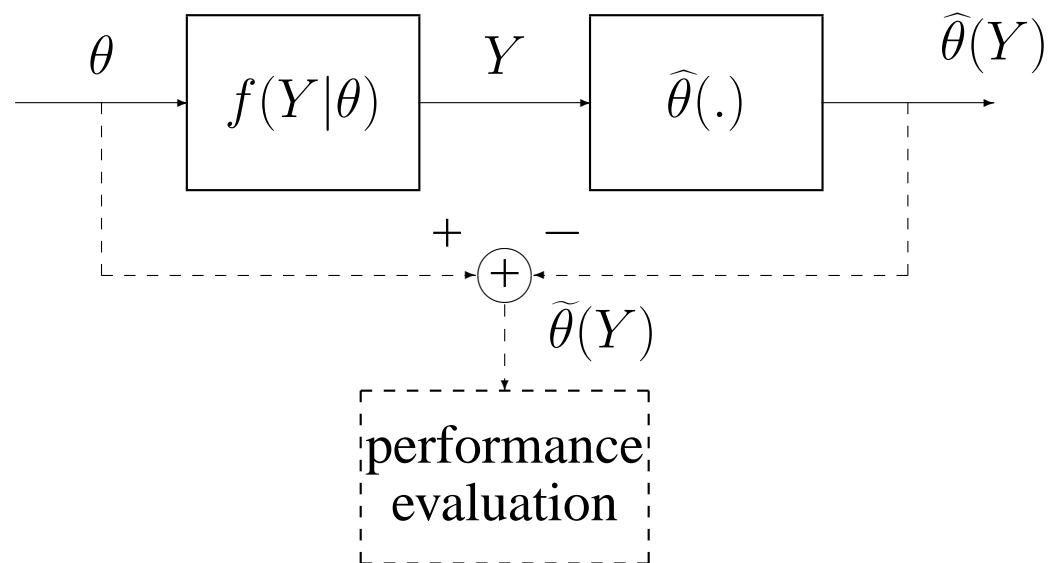
The only stochastic description available is the conditional density $f(Y|\theta)$ describing the stochastic relation between the unknown parameters θ and the observed measurements Y .

- since θ is not necessarily a random vector but just a set of parameters on which the distribution of Y depends, we often find the notations

$$f(Y|\theta) = f(Y;\theta) = f_\theta(Y)$$

but we shall continue to use $f(Y|\theta)$

- expectation now means $E = E_{Y|\theta}$



Deterministic Parameter Estimation (2)

- an estimator $\hat{\theta}(Y)$ of θ is again a function of Y (a statistic), with estimation error $\tilde{\theta} = \theta - \hat{\theta}(Y)$
- to evaluate the quality of an estimator, we shall again introduce the *risk* function MSE as the average value of the SE *cost* function

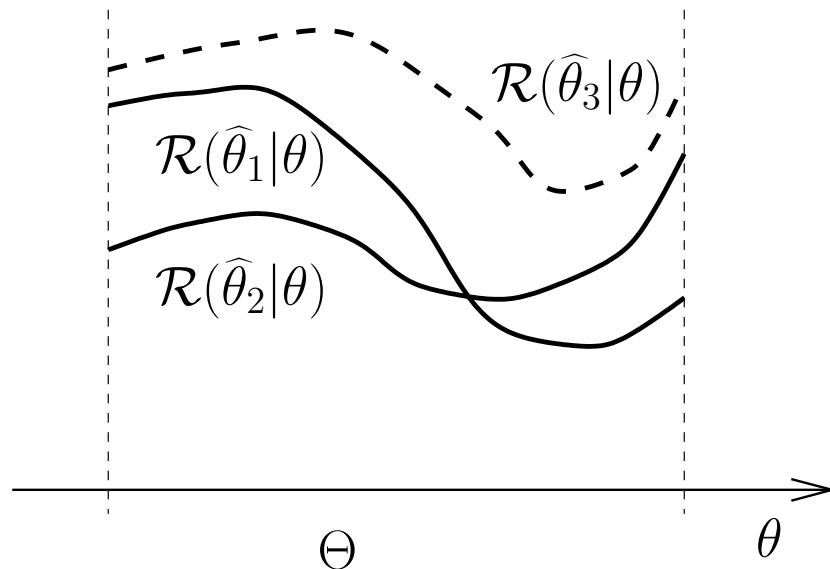
$$\text{MSE} = \mathcal{R}(\hat{\theta}(.)|\theta) = E_{Y|\theta} \|\tilde{\theta}\|^2 = \int f(Y|\theta) \|\theta - \hat{\theta}(Y)\|^2 dY$$

the MSE is a function of θ in general

- minimization of the risk function leads to $\hat{\theta} = \theta$ (and $\mathcal{R} = 0$): not an acceptable strategy since the resulting $\hat{\theta}$ depends on the unknown θ
- ideally, would like $\hat{\theta}(.)$ such that $\mathcal{R}(\hat{\theta}(.)|\theta)$ is minimized $\forall \theta \in \Theta$: impossible!
Consider $\hat{\theta}(Y) = \theta_0 \in \Theta$: ignores the data Y but $\mathcal{R}(\hat{\theta}(.)|\theta_0) = 0$
- we shall still evaluate the performance via the MSE, but in the deterministic case, we shall not be able to derive estimators by minimizing the MSE.

Deterministic Parameter Estimation (3)

- given two estimators $\hat{\theta}_1(Y)$ and $\hat{\theta}_2(Y)$, one is usually not uniformly better than the other one (see figure)
- a uniformly minimum risk estimator does not exist in general
- consider some other desirable properties



Some Optimality Properties

- estimator *bias* : average deviation from the true parameter

$$b_{\hat{\theta}}(\theta) = -E_{Y|\theta}\bar{\theta} = E_{Y|\theta}(\hat{\theta}(Y) - \theta) = E_{Y|\theta}\hat{\theta}(Y) - \theta$$

unbiased estimator: $b_{\hat{\theta}}(\theta) = 0, \forall \theta \in \Theta$

Unbiasedness is a weak property: estimator can be correct on the average, but with large deviations. Good estimators exist that are biased.

- Example: mean of Gaussian i.i.d. variables

$$\text{i.i.d. } y_i \sim \mathcal{N}(\theta, 1), \quad i = 1, \dots, n$$

Consider $\hat{\theta}(Y) = \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$, the sample mean.

$$E_{Y|\theta}\hat{\theta} = E_{Y|\theta}\bar{y} = E_{Y|\theta}\frac{1}{n} \sum_{i=1}^n y_i = \frac{1}{n} \sum_{i=1}^n E_{Y|\theta}y_i = \frac{1}{n} \sum_{i=1}^n \theta = \frac{n\theta}{n} = \theta : \text{unbiased!}$$

- $\hat{\theta}(\cdot)$ is *inadmissible* if another estimator $\hat{\theta}'(\cdot)$ has uniformly lower risk:

$$\forall \theta \in \Theta : \mathcal{R}(\hat{\theta}'|\theta) \leq \mathcal{R}(\hat{\theta}|\theta), \quad \exists \theta_0 \in \Theta : \mathcal{R}(\hat{\theta}'|\theta_0) < \mathcal{R}(\hat{\theta}|\theta_0)$$

$\hat{\theta}$ is *admissible* if no such $\hat{\theta}'$ exists. Example: $\hat{\theta}_3$ in figure above.



Some Optimality Properties (2)

- $\text{MSE} = E_{Y|\theta} \|\tilde{\theta}\|^2 = E_{Y|\theta} \tilde{\theta}^T \tilde{\theta} = \text{tr} \{E_{Y|\theta} \tilde{\theta} \tilde{\theta}^T\} = \text{tr} \{R_{\tilde{\theta} \tilde{\theta}}\},$

$R_{\tilde{\theta} \tilde{\theta}} = E_{Y|\theta} \tilde{\theta} \tilde{\theta}^T$ = estimation error correlation matrix

$$\begin{aligned} R_{\tilde{\theta} \tilde{\theta}} &= E_{Y|\theta} (\hat{\theta} - \theta)(\hat{\theta} - \theta)^T = E_{Y|\theta} [\hat{\theta} (-E_{Y|\theta} \hat{\theta} + E_{Y|\theta} \hat{\theta}) - \theta] [\hat{\theta} (-E_{Y|\theta} \hat{\theta} + E_{Y|\theta} \hat{\theta}) - \theta]^T \\ &= E_{Y|\theta} (\hat{\theta} - E_{Y|\theta} \hat{\theta})(\hat{\theta} - E_{Y|\theta} \hat{\theta})^T + (E_{Y|\theta} \hat{\theta} - \theta)(E_{Y|\theta} \hat{\theta} - \theta)^T \\ &= C_{\hat{\theta} \hat{\theta}} + b_{\hat{\theta}}(\theta) b_{\hat{\theta}}^T(\theta) = C_{\tilde{\theta} \tilde{\theta}} + (E_{Y|\theta} \tilde{\theta}) (E_{Y|\theta} \tilde{\theta})^T \end{aligned}$$

where we used: $C_{\hat{\theta} \hat{\theta}} = C_{\tilde{\theta} \tilde{\theta}}$



Some Optimality Properties (3)

- $\widehat{\theta}(Y)$ is said to be *minimax* if it satisfies

$$\sup_{\theta \in \Theta} \mathcal{R}(\widehat{\theta}|\theta) = \inf_{\widehat{\theta}'} \sup_{\theta \in \Theta} \mathcal{R}(\widehat{\theta}'|\theta)$$

($\inf \approx \min$, $\sup \approx \max$).

A minimax estimator minimizes the maximum risk over Θ .

A minimax $\widehat{\theta}$ is difficult to obtain in general.

Uniformly minimum risk estimators may be found if we restrict the class of estimators.

- $\widehat{\theta}$ is a *uniformly minimum variance unbiased estimator* (UMVUE) if it is unbiased and if for any other unbiased estimator $\widehat{\theta}'$: $R_{\tilde{\theta}\tilde{\theta}} \leq R_{\tilde{\theta}'\tilde{\theta}'}$, $\forall \theta \in \Theta$, or

$$E_{Y|\theta}(\widehat{\theta}(Y) - \theta)(\widehat{\theta}(Y) - \theta)^T \leq E_{Y|\theta}(\widehat{\theta}'(Y) - \theta)(\widehat{\theta}'(Y) - \theta)^T$$

note: variance = $\text{tr} \{ \text{covariance matrix} \}$, $\text{MSE}_{\widehat{\theta}} = \text{tr} \{ R_{\tilde{\theta}\tilde{\theta}} \}$

- UMVUE are highly desirable but they may not exist or be difficult to compute. They can be computed if a *complete sufficient statistic* can be found.

Maximum Likelihood Estimation

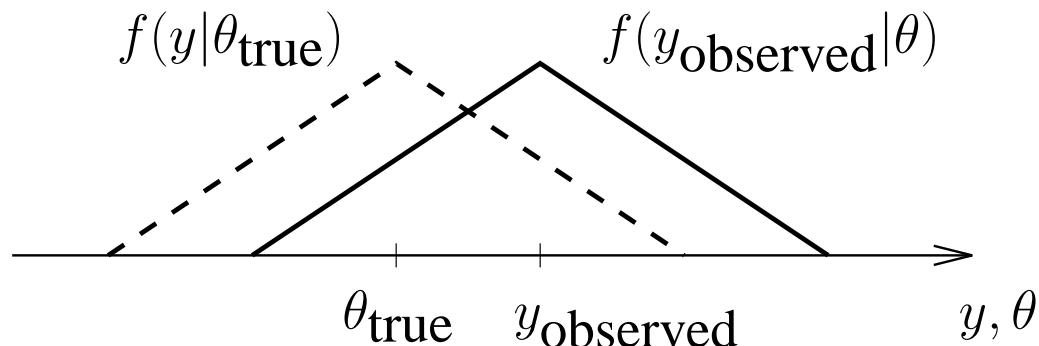
- the maximum likelihood (ML) estimation philosophy is to choose that value of the parameters that renders the observations most likely:

$$\hat{\theta}_{ML}(Y) = \arg \max_{\theta \in \Theta} f(Y|\theta)$$

example:

- $y = \theta + v$, $f_v(v) = \begin{cases} 1 - |v| & , |v| \leq 1 \\ 0 & , |v| > 1 \end{cases}$ $f(y|\theta) = f_v(y - \theta)$

$$\hat{\theta}_{ML}(y) = y$$



ML Estimation: Remarks

- $f(Y|\theta)$ is called the *likelihood function*. In order to emphasize the dependence on θ and the fact that the observation Y is fixed, it is often denoted as

$$l(\theta; Y) = f(Y|\theta) \quad L(\theta; Y) = \ln f(Y|\theta)$$

- since the logarithmic function is strictly monotone, the maximum point of $f(Y|\theta)$ corresponds with the maximum point of $\ln f(Y|\theta)$, called the *log likelihood function*
- Often $f(Y|\theta)$ satisfies certain regularity conditions so that $\hat{\theta}_{ML}$ is a solution of

$$\frac{\partial}{\partial \theta} \ln f(Y|\theta) = 0.$$

The conditions for a maximum (rather than another form of extremum) need to be verified of course.

- The ML estimator is given by the *global* maximum of $f(Y|\theta)$. If there are several local maxima, all of them need to be examined and compared to find the global maximum.



ML Estimation: Remarks (2)

- Even if $f(Y|\theta)$ satisfies regularity conditions, the maximum may occur at the boundary of the parameter space Θ (which may not necessarily be $(-\infty, \infty)$ for every θ_i). In that case, the maximum is not a local extremum.
- The ML estimator can be seen as a limiting case of the MAP estimator when the prior distribution $f(\theta)$ becomes uninformative (uniform distribution). For those components θ_i of θ for which the support is unbounded, this means that $\sigma_{\theta_i}^2 \rightarrow \infty$ (information $\rightarrow 0$). Indeed

$$\begin{aligned}\hat{\theta}_{MAP}(Y) &= \arg \max_{\theta \in \Theta} f(\theta|Y) = \arg \max_{\theta \in \Theta} \frac{f(Y|\theta)f(\theta)}{f(Y)} \\ &= \arg \max_{\theta \in \Theta} f(Y|\theta)f(\theta) \stackrel{f(\theta)=c^t}{=} \arg \max_{\theta \in \Theta} f(Y|\theta) = \hat{\theta}_{ML}(Y)\end{aligned}$$

But in the deterministic case, θ is fixed, whereas in the Bayesian case θ is random, hence e.g. the MSE is different for both formulations
($\text{MSE}_{MAP} = \int_{\Theta} \text{MSE}_{ML}(\theta) f(\theta) d\theta$, averaged with prior distribution for θ).

ML Estimation: Example 1

- Given: $y_i = \mu + \sigma v_i$, $v_i \sim \mathcal{N}(0, 1)$ i.i.d. or $y_i \sim \mathcal{N}(\mu, \sigma^2)$ i.i.d. $\theta = \begin{bmatrix} \mu \\ \sigma^2 \end{bmatrix}$

$$Y = \mu \mathbf{1} + \sigma V, \quad V \sim \mathcal{N}(0, I_n)$$

- Q: $\hat{\theta}_1 = \hat{\mu}_{ML}$, $\hat{\theta}_2 = \hat{\sigma}^2_{ML}$

- A:

$$f(Y|\mu, \sigma^2) = \prod_{i=1}^n f(y_i|\mu, \sigma^2) = \prod_{i=1}^n \frac{\exp[-\frac{(y_i-\mu)^2}{2\sigma^2}]}{\sqrt{2\pi\sigma^2}} = (2\pi)^{-\frac{n}{2}}(\sigma^2)^{-\frac{n}{2}} \exp[-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i-\mu)^2]$$

$$L(\theta; Y) = \ln l(\theta; Y) = -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln \sigma^2 - \frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \mu)^2$$

$$\left\{ \begin{array}{l} \frac{\partial}{\partial \mu} L(\theta; Y) = 0 = \frac{1}{\sigma^2} \sum_{i=1}^n (y_i - \mu) \end{array} \right. \quad (1)$$

$$\left\{ \begin{array}{l} \frac{\partial}{\partial \sigma^2} L(\theta; Y) = 0 = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^n (y_i - \mu)^2 \end{array} \right. \quad (2)$$

$$\left\{ \begin{array}{l} (1) \Rightarrow \hat{\mu}_{ML} = \frac{1}{n} \sum_{i=1}^n y_i = \bar{y} \quad \text{sample mean} \\ (2) \Rightarrow \hat{\sigma}^2_{ML} = \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2 = \overline{(y - \bar{y})^2} \quad \text{sample variance} \end{array} \right.$$



bias calculations

ML Estimation: Example 1 (2)

- $E[\hat{\mu}_{ML}|\mu, \sigma^2] = E[\bar{y}|\mu, \sigma^2] = \frac{1}{n} \sum_{i=1}^n E[y_i|\mu, \sigma^2] = \frac{1}{n} \sum_{i=1}^n \mu = \mu$ unbiased!
- note: with $\bar{y} = \frac{1}{n} \mathbf{1}^T Y$, we get

$$\begin{aligned} n \hat{\sigma}^2_{ML} &= \sum_{i=1}^n (y_i - \bar{y})^2 = \left\| \begin{bmatrix} y_1 - \bar{y} \\ \vdots \\ y_n - \bar{y} \end{bmatrix} \right\|^2 = \|Y - \bar{y}\mathbf{1}\|^2 = (Y - \bar{y}\mathbf{1})^T (Y - \bar{y}\mathbf{1}) \\ &= (Y - \mu\mathbf{1} + \mu\mathbf{1} - \bar{y}\mathbf{1})^T (Y - \mu\mathbf{1} + \mu\mathbf{1} - \bar{y}\mathbf{1}) = (Y - \mu\mathbf{1} - (\bar{y} - \mu)\mathbf{1})^T (\dots) = (Y - \mu\mathbf{1})^T (Y - \mu\mathbf{1}) \end{aligned}$$

$$+ (\bar{y} - \mu)^2 \underbrace{\mathbf{1}^T \mathbf{1}}_{=n} - 2(\bar{y} - \mu) \underbrace{\mathbf{1}^T (Y - \mu\mathbf{1})}_{=n(\bar{y} - \mu)} = \underbrace{(Y - \mu\mathbf{1})^T (Y - \mu\mathbf{1})}_{\sum_{i=1}^n (y_i - \mu)^2} - \frac{1}{n} (Y - \mu\mathbf{1})^T \mathbf{1} \mathbf{1}^T (Y - \mu\mathbf{1})$$

hence $\hat{\sigma}^2_{ML}$ is biased:

$$= \sigma^2 - \frac{1}{n^2} \text{tr}\{\mathbf{1} \mathbf{1}^T \sigma^2 I_n\} = \sigma^2 - \frac{1}{n^2} \sigma^2 \underbrace{\mathbf{1}^T I_n \mathbf{1}}_{=n} = (1 - \frac{1}{n})\sigma^2 = \frac{n-1}{n}\sigma^2 \neq \sigma^2$$

- unbiased variance estimate: $\hat{\sigma}^2_{ub} = \frac{n}{n-1} \hat{\sigma}^2_{ML} = \frac{1}{n-1} \sum_{i=1}^n (y_i - \bar{y})^2$

however, can show: $Var\{\hat{\sigma}^2_{ub}\} \geq Var\{\hat{\sigma}^2_{ML}\}$ (and similarly for MSE).

ML Estimation: Example 2

- given: $y_i \sim \mathcal{U}[\theta - \frac{1}{2}, \theta + \frac{1}{2}]$ i.i.d. $f(y_i|\theta) = \begin{cases} 1 & , y_i \in [\theta - \frac{1}{2}, \theta + \frac{1}{2}] \\ 0 & , \text{elsewhere} \end{cases}$

- Q: $\hat{\theta}_{ML}$

- A: use the indicator function $I_A(x) = \begin{cases} 1 & , x \in A \\ 0 & , x \notin A \end{cases}$

$$f(y_i|\theta) = I_{[\theta - \frac{1}{2}, \theta + \frac{1}{2}]}(y_i) = 1 \text{ if } \theta - \frac{1}{2} \leq y_i \leq \theta + \frac{1}{2} \Leftrightarrow y_i - \frac{1}{2} \leq \theta \leq y_i + \frac{1}{2}$$

hence

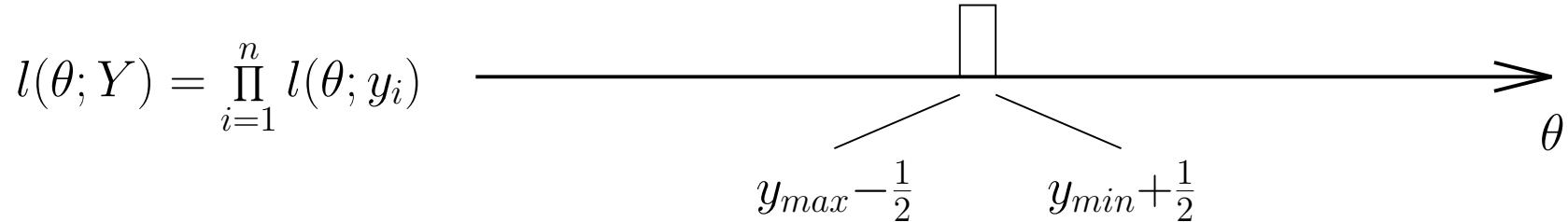
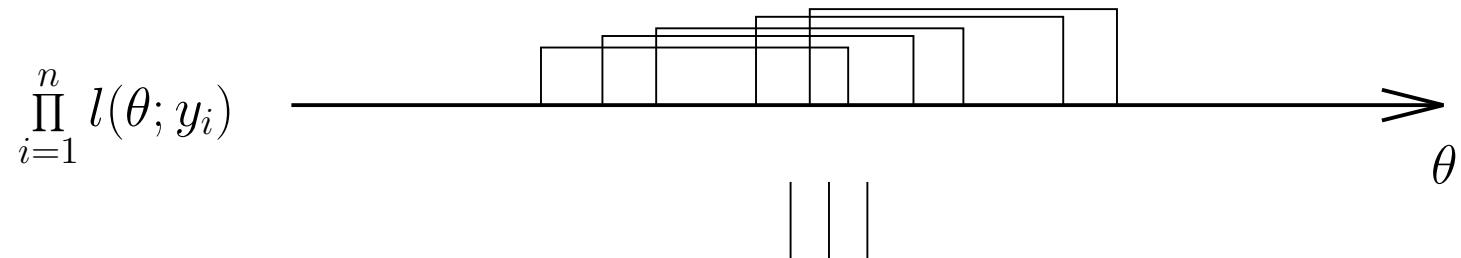
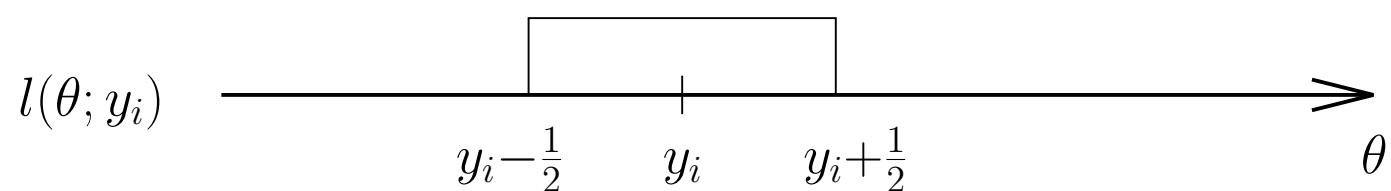
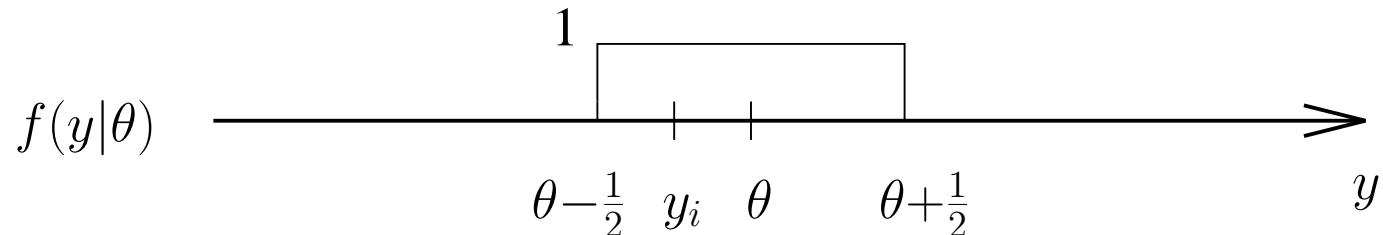
$$f(Y|\theta) = \prod_{i=1}^n f(y_i|\theta) = \prod_{i=1}^n I_{[\theta - \frac{1}{2}, \theta + \frac{1}{2}]}(y_i) = \prod_{i=1}^n I_{[y_i - \frac{1}{2}, y_i + \frac{1}{2}]}(\theta)$$

$$= I_{\bigcap_{i=1}^n [y_i - \frac{1}{2}, y_i + \frac{1}{2}]}(\theta) = I_{[y_{max} - \frac{1}{2}, y_{min} + \frac{1}{2}]}(\theta)$$

hence $\hat{\theta} \in [y_{max} - \frac{1}{2}, y_{min} + \frac{1}{2}]$ a whole interval!

- choose $\hat{\theta}_{ML} = \frac{y_{min} + y_{max}}{2}$

ML Estimation: Example 2 (2)



Fisher Information Matrix

- The information matrix for deterministic parameters is defined as

$$J(\theta) = R_{\frac{\partial L}{\partial \theta}, \frac{\partial L}{\partial \theta}} = E_{Y|\theta} \left(\frac{\partial \ln f(Y|\theta)}{\partial \theta} \right) \left(\frac{\partial \ln f(Y|\theta)}{\partial \theta} \right)^T = -E_{Y|\theta} \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(Y|\theta)}{\partial \theta} \right)^T$$

It can again be shown to satisfy all the properties we specified for an information matrix. The second equality can be shown as before. Note that $J(\theta)$ now depends on the true parameter value θ .

- unbiased estimators: $b_{\hat{\theta}}(\theta) = E_{Y|\theta} \hat{\theta}(Y) - \theta = 0$, $\forall \theta \in \Theta$
- **Lemma 0.1 (Unit Cross Correlation)** *For any unbiased estimator $\hat{\theta}(Y)$*

$$E_{Y|\theta} \frac{\partial \ln f(Y|\theta)}{\partial \theta} (\hat{\theta} - \theta)^T = I .$$

In words, the cross correlation matrix between $\frac{\partial \ln f(Y|\theta)}{\partial \theta}$ and the estimation error of any unbiased estimator is the identity matrix.

Cramer-Rao Bound

- **Theorem (CRB for Deterministic Parameters)** *If the estimator $\widehat{\theta}(Y)$ of θ is unbiased, then the covariance matrix of the parameter estimation errors $\tilde{\theta}$ is bounded below by the inverse of the information matrix:*

$$C_{\tilde{\theta}\tilde{\theta}} = R_{\tilde{\theta}\tilde{\theta}} = E_{Y|\theta} (\widehat{\theta} - \theta)(\widehat{\theta} - \theta)^T \geq J^{-1}(\theta)$$

with equality iff

$$\widehat{\theta}(Y) - \theta = J^{-1}(\theta) \frac{\partial \ln f(Y|\theta)}{\partial \theta} \quad a.e. (\theta)$$

An estimator that achieves the lower bound ($\forall \theta \in \Theta$) is called *efficient*.

Remarks:

- when equality holds, we can integrate to get

$$f(Y|\theta) = h(Y) \exp[c_1^T(\theta)\widehat{\theta}(Y) - c_0(\theta)]$$

where $\frac{\partial c_1^T(\theta)}{\partial \theta} = J(\theta)$ and $\frac{\partial c_0(\theta)}{\partial \theta} = J(\theta)\theta$. Hence $\{f(Y|\theta), \theta \in \Theta\}$ forms an exponential family and $\widehat{\theta}(Y)$ is a sufficient statistic.

Cramer-Rao Bound: Remarks

- the CRB $J^{-1}(\theta)$ only depends on $f(Y|\theta)$, not on $\hat{\theta}(Y)$
- the (deterministic) CRB has two uses:
 - (i) evaluate unbiased estimators: $\hat{\theta}$ with $b_{\hat{\theta}}(\theta) \equiv 0$: if $C_{\tilde{\theta}\tilde{\theta}} - J^{-1}(\theta)$ small enough, then $\hat{\theta}$ good enough
 - (ii) find UMVUE: $\min_{\hat{\theta}: b_{\hat{\theta}} \equiv 0} C_{\tilde{\theta}\tilde{\theta}} \geq J^{-1}(\theta)$.
If $\hat{\theta}$ is efficient ($\forall \theta \in \Theta$), $C_{\tilde{\theta}\tilde{\theta}} = J^{-1}(\theta)$, then $\hat{\theta}$ is UMVUE!

- **Theorem** Suppose $\hat{\theta}_{ML}$ is obtained by $\frac{\partial}{\partial \theta} f(Y|\theta)|_{\theta=\hat{\theta}_{ML}} = 0$. Then if an efficient estimator exists, it is $\hat{\theta}_{ML}$.

Proof: $\hat{\theta}_{eff}$ satisfies

$$\frac{\partial \ln f(Y|\theta)}{\partial \theta} = \underbrace{J(\theta)}_{>0} [\hat{\theta}_{eff} - \theta]$$

For $\theta = \hat{\theta}_{ML}$, LHS = 0, hence RHS = 0 : $\hat{\theta}_{eff} = \hat{\theta}_{ML}$.

- If $J(\theta)$ is singular \Rightarrow (local) unidentifiability. E.g. linear model with $n < m$.

Cramer-Rao Bound: Example

- i.i.d. $y_i \sim \mathcal{N}(\mu, \sigma^2)$, σ^2 known, $\theta = \mu$
- $f(Y|\mu) = \prod_{i=1}^n f(y_i|\mu) = (2\pi\sigma^2)^{-n/2} \exp[-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \mu)^2]$
- $\frac{\partial \ln f(Y|\mu)}{\partial \mu} = \frac{1}{\sigma^2} \sum_{i=1}^n (y_i - \mu)$, $\frac{\partial^2 \ln f(Y|\mu)}{\partial \mu^2} = -\frac{n}{\sigma^2}$
- $J = -E_{Y|\mu} \frac{\partial^2 \ln f(Y|\mu)}{\partial \mu^2} = \frac{n}{\sigma^2}$, $C_{\tilde{\mu}\tilde{\mu}} = E_{Y|\mu}(\hat{\mu} - \mu)^2 \geq J^{-1} = \frac{\sigma^2}{n}$
- $\hat{\mu}_{ML} = \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$, $E_{Y|\mu} \hat{\mu}_{ML} = \mu$: unbiased
- $C_{\tilde{\mu}\tilde{\mu}} = E_{Y|\mu}(\hat{\mu} - \mu)^2 = E_{Y|\mu} \left(\frac{1}{n} \sum_{i=1}^n (y_i - \mu) \right)^2$
 $= \frac{1}{n^2} \left(\underbrace{\sum_{i=1}^n E(y_i - \mu)^2}_{=\sigma^2} + \sum_{i \neq j} \underbrace{\frac{E(y_i - \mu)(y_j - \mu)}{=(Ey_i-\mu)(Ey_j-\mu)=0}}_{=0} \right) = \frac{1}{n^2} n \sigma^2 = \frac{\sigma^2}{n} = J^{-1}$
- efficient: $\frac{\partial \ln f(Y|\mu)}{\partial \mu} = \frac{1}{\sigma^2} \sum_{i=1}^n (y_i - \mu) = \frac{n}{\sigma^2} (\bar{y} - \mu) = J (\hat{\mu}_{ML} - \mu)$

The Deterministic Linear Model

- $Y = H\theta + V$, $V \sim \mathcal{N}(0, C_{VV})$
- $f_{\mathbf{Y}|\boldsymbol{\theta}}(Y|\theta) = f_{\mathbf{V}}(Y - H\theta) = \frac{1}{\sqrt{(2\pi)^n \det C_{VV}}} e^{-\frac{1}{2}(Y-H\theta)^T C_{VV}^{-1} (Y-H\theta)}$
- $\frac{\partial \ln f_{\mathbf{V}}(Y - H\theta)}{\partial \theta} = H^T C_{VV}^{-1} (Y - H\theta) = 0$
 $\Rightarrow \hat{\theta}_{ML} = (H^T C_{VV}^{-1} H)^{-1} H^T C_{VV}^{-1} Y$
- $\frac{\partial}{\partial \theta} \left(\frac{\partial \ln f_{\mathbf{V}}(Y - H\theta)}{\partial \theta} \right)^T = -H^T \underbrace{C_{VV}^{-1} H}_{>0} = -J < 0 \Rightarrow \text{maximum!}$
 assuming H full column rank
- $\tilde{\theta} = \theta - \hat{\theta} = -(H^T C_{VV}^{-1} H)^{-1} H^T C_{VV}^{-1} V$, $E_{Y|\theta} \tilde{\theta} = E_V \tilde{\theta} = 0 \Rightarrow \text{unbiased!}$
- $C_{\tilde{\theta}\tilde{\theta}} = R_{\tilde{\theta}\tilde{\theta}} = E_{Y|\theta} \tilde{\theta} \tilde{\theta}^T = E_V \tilde{\theta} \tilde{\theta}^T = (H^T C_{VV}^{-1} H)^{-1} = J^{-1} : \text{efficient!}$
- $\frac{\partial \ln f_{\mathbf{V}}(Y - H\theta)}{\partial \theta} = H^T C_{VV}^{-1} Y - H^T C_{VV}^{-1} H \theta = J(\hat{\theta} - \theta) : \text{efficient}$



Statistical Signal Processing

Lecture 5

chapter 1: parameter estimation: deterministic parameters

- some optimality properties
- Maximum Likelihood estimation, examples
- Fischer Information Matrix
- Cramer-Rao lower bound on the MSE, example
- linear model
- asymptotic (large sample) properties
- recap: estimator properties and estimators
- simplified estimators: BLUE, (W)LS, method of moments

Asymptotic (Large Sample) Properties

- asymptotic: $n \rightarrow \infty$
- *asymptotically unbiased*: $\lim_{n \rightarrow \infty} b_n(\theta) = 0$, $\forall \theta \in \Theta$
- Example (mean and variance of Gaussian i.i.d. variables):

$$\begin{aligned} E[\widehat{\sigma^2}_{ML} | \mu, \sigma^2] &= \frac{n-1}{n} \sigma^2 \\ b_n &= \frac{n-1}{n} \sigma^2 - \sigma^2 = -\frac{\sigma^2}{n} \xrightarrow{n \rightarrow \infty} 0 \end{aligned}$$

$\widehat{\sigma^2}_{ML}$: biased but asymptotically unbiased

- *consistency* : convergence of (a series of random vectors:) $\widehat{\theta}_n \rightarrow \theta$
 - convergence in probability
 - mean square convergence
 - convergence with probability one
 - convergence in distribution

Consistency

the sequence of estimates $\widehat{\theta}(Y_n)$ is said to be

- *simply or weakly consistent* if

$$\lim_{n \rightarrow \infty} \Pr_{Y_n|\theta} \{ \|\widehat{\theta}(Y_n) - \theta\| < \epsilon \} = 1, \quad \forall \epsilon > 0, \quad \forall \theta \in \Theta$$

- *mean-square consistent* if

$$\lim_{n \rightarrow \infty} \text{MSE}_n = \lim_{n \rightarrow \infty} E_{Y_n|\theta} \|\widehat{\theta}(Y_n) - \theta\|^2 = 0, \quad \forall \theta \in \Theta$$

- *strongly consistent* if

$$\Pr_{Y_\infty|\theta} \{ \lim_{n \rightarrow \infty} \widehat{\theta}(Y_n) = \theta \} = 1, \quad \forall \theta \in \Theta$$

- Any of these 3 consistencies implies asymptotic unbiasedness. E.g. for mean-square:

$$\frac{E_{Y_n|\theta} \|\widehat{\theta}(Y_n) - \theta\|^2}{\text{MSE}} = \underbrace{\|\underbrace{E_{Y_n|\theta} \widehat{\theta}(Y_n)}_{\text{bias}} - \theta\|^2}_{\text{bias}} + \underbrace{E_{Y_n|\theta} \|\widehat{\theta}(Y_n) - E_{Y_n|\theta} \widehat{\theta}\|^2}_{\text{variance}} \rightarrow 0$$

$$\Rightarrow \lim_{n \rightarrow \infty} E_{Y_n|\theta} \widehat{\theta}(Y_n) = \theta$$

Consistency (2)

- Strong and mean-square consistency do not imply each other in general. Either implies weak consistency (e.g. use the Chebyshev inequality to show that mean-square consistency implies weak consistency), but not conversely. Except when Θ is bounded: then weak consistency implies mean-square consistency.
- example: i.i.d. $y_i \sim \mathcal{N}(\mu, \sigma^2)$, $\theta = \mu$, σ^2 known. $\hat{\mu}_{ML} = \bar{y}$

$$Var(\hat{\mu}_{ML}) = \frac{\sigma^2}{n} \xrightarrow{n \rightarrow \infty} 0 \text{ mean-square consistent}$$

- example: i.i.d. $y_i \sim U[\theta - \frac{1}{2}, \theta + \frac{1}{2}]$, $\hat{\theta}_{ML} = \frac{y_{min} + y_{max}}{2}$
$$\begin{cases} y_{min} \rightarrow \theta - \frac{1}{2} & \text{in probability} \\ y_{max} \rightarrow \theta + \frac{1}{2} & \text{in probability} \end{cases} \quad \text{weak consistency}$$

$$\hat{\theta}_{ML} \rightarrow \theta \text{ in probability}$$

mean-square consistency can also be shown

Asymptotic Normality

- if $\widehat{\theta}_n$ consistent, then $\widetilde{\theta} \rightarrow 0$ in some sense
- introduce a magnifying glass: $d_n(\widehat{\theta}_n - \theta)$ where $0 < d_{n-1} \leq d_n \rightarrow \infty$
- *convergence in distribution*: weaker than the 3 forms of convergence of sequences of random vectors mentioned before
- if $d_n(\widehat{\theta}_n - \theta) \xrightarrow{indist.} \xi$, some random vector, then the distribution of ξ useful as a measure for the limiting behavior of $\widehat{\theta}_n$
- usually $d_n = \sqrt{n}$
- $\widehat{\theta}_n$ *consistent asymptotically normal* (CAN) :
if $\widehat{\theta}_n$ simply consistent and $d_n(\widehat{\theta}_n - \theta) \xrightarrow{indist.} \mathcal{N}(0, \Xi(\theta))$
CAN implies asympt. unbiased (which requires that bias $\rightarrow 0$ faster than $\frac{1}{d_n}$),
 Ξ = asymptotic normalized covariance of $\widehat{\theta}_n$. We say that $\widehat{\theta}_n = \theta + \mathcal{O}_p(\frac{1}{d_n})$
- distinguish $\Xi(\theta)$ from $V(\theta) = \lim_{n \rightarrow \infty} d_n^2 C_{\tilde{\theta}\tilde{\theta}}(\theta)$ which may not even exist for a CAN estimate (if $\widehat{\theta}_n$ is simply but not mean-square consistent). $V(\theta)$ exists for a mean-square consistent $\widehat{\theta}_n$, but is not necessarily $= \Xi(\theta)$.
- Hence CAN can be used to formulate *interval estimators* on the basis of *point estimators*.

Asymptotic Optimality of ML

- *asymptotic normalized information matrix* : $J_0(\theta) = \lim_{n \rightarrow \infty} \frac{1}{d_n^2} J_n(\theta)$ if it exists
 $(J_0(\theta) = \text{asymptotic average information per data sample } y_n \text{ if } d_n = \sqrt{n})$
- *best asymptotically normal* (BAN): CAN and $\Xi(\theta) = J_0^{-1}(\theta)$
also called *asymptotically efficient*
- under some regularity conditions (maximum of the likelihood function unique, y_i given θ i.i.d.,...) the ML estimate is strongly consistent and BAN with $d_n = \sqrt{n}$ (\Rightarrow another use of the CRB). In particular, the ML estimate is
 - asymptotically unbiased
 - asymptotically efficient (i.i.d.: $J_n = nJ_1 \Rightarrow J_0 = J_1$)
 - asymptotically normal
- example: i.i.d. $y_i \sim \mathcal{N}(\mu, \sigma^2)$, $\theta = \mu$, σ^2 known. $\hat{\mu}_{ML} = \bar{y}$

$$\hat{\mu}_{ML} \sim \mathcal{N}\left(\mu, \frac{\sigma^2}{n}\right) \longrightarrow \sqrt{n}(\hat{\mu}_{ML} - \mu) \sim \mathcal{N}(0, \sigma^2), \quad J_n = \frac{n}{\sigma^2} \Rightarrow J_0^{-1} = \sigma^2 = \Xi(\theta)$$

Recap: Properties of Estimators $\widehat{\theta}(Y)$

small sample (finite n):

- *bias*: $b_{\widehat{\theta}}(\theta) = E_{Y|\theta} \widehat{\theta}(Y) - \theta = -E_{Y|\theta} \widetilde{\theta} = -m_{\widetilde{\theta}}$ ($= 0$, $\forall \theta \in \Theta$: unbiased)
- *error correlation*: $R_{\widetilde{\theta}\widetilde{\theta}} = E_{Y|\theta} (\widehat{\theta}(Y) - \theta) (\widehat{\theta}(Y) - \theta)^T = C_{\widetilde{\theta}\widetilde{\theta}} + b_{\widehat{\theta}} b_{\widehat{\theta}}^T$

Cramer-Rao Bound : $\widehat{\theta}$ unbiased: $R_{\widetilde{\theta}\widetilde{\theta}} = C_{\widetilde{\theta}\widetilde{\theta}} = C_{\widehat{\theta}\widehat{\theta}}$ MSE = $\text{tr}\{R_{\widetilde{\theta}\widetilde{\theta}}\}$

$$C_{\widetilde{\theta}\widetilde{\theta}} \geq J^{-1}(\theta) , \quad J(\theta) = -E_{Y|\theta} \frac{\partial}{\partial \theta} \left(\frac{\partial \ln f(Y|\theta)}{\partial \theta} \right)^T \quad \text{information matrix}$$

efficient: $C_{\widetilde{\theta}\widetilde{\theta}} = J^{-1}(\theta)$, $\forall \theta \in \Theta \Rightarrow \widehat{\theta}(Y)$ is UMVUE

large sample ($n \rightarrow \infty$):

- *asymptotically unbiased*: $\lim_{n \rightarrow \infty} b_{\widehat{\theta}}(\theta) = 0$, $\forall \theta \in \Theta$
- *consistency* (weak, in mean square, strong): \Rightarrow asymptotically unbiased
- *asymptotic normality*:

$$\text{BAN} \left\{ \begin{array}{l} \diamond \text{ weakly consistent} \\ \diamond \text{ asymptotically normal} \\ \diamond \text{ asymptotically efficient} \end{array} \right\} \text{CAN}$$

Recap: Estimation Techniques

- *Uniformly Minimum Variance Unbiased Estimator (UMVUE)*: complicated (via "sufficient statistics")
- *Maximum likelihood* (ML): $\hat{\theta}_{ML} = \arg \max_{\theta} f(Y|\theta)$

Qualities:

- ◊ if \exists efficient $\hat{\theta} = \hat{\theta}_{eff}$ and $\hat{\theta}_{ML}$ is obtained from $\frac{\partial \ln f(Y|\theta)}{\partial \theta} = 0$
 $\Rightarrow \hat{\theta}_{eff} = \hat{\theta}_{ML} = \hat{\theta}_{UMVUE}$
- ◊ $\hat{\theta}_{ML}$ = BAN

Problems:

- ◊ what if $f(Y|\theta)$ is unknown?
- ◊ if $f(Y|\theta)$ is not concave (local maxima)
- simplified estimators:
 - ◊ *Best Linear Unbiased Estimator (BLUE)* → linear model
 - ◊ *Method of Moments*
 - ◊ *Least-Squares (LS)* → linear model

Best Linear Unbiased Estimator (BLUE)

- deterministic analog of LMMSE in the Bayesian case
- *linear*: $\widehat{\theta}(Y) = F Y \quad (F : m \times n)$
- *unbiased*: $E_{Y|\theta} \widehat{\theta} = F E(Y|\theta) = \theta$
- *best = minimum variance*: $\min C_{\tilde{\theta}\tilde{\theta}}$
- remarks:
 - BLUE inferior to UMVUE unless UMVUE is linear
 - generalizations: $X = g(Y) : \widehat{\theta}(Y) = F X = F g(Y) \quad (\text{linear in } X)$
e.g.: linear in Y inappropriate if $\theta \neq 0$ and $E(Y|\theta) = 0$

Example of $X = g(Y)$

- $y_i \sim \mathcal{N}(0, \sigma^2)$ i.i.d., $\theta = \sigma^2$, $Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$
- linear: $\widehat{\sigma^2} = F Y \Rightarrow E_{Y|\sigma^2} \widehat{\sigma^2} = F E(Y|\sigma^2) = 0 \neq \sigma^2$
no linear unbiased estimator $\widehat{\sigma^2}$ exists
- however, let $x_i = y_i^2$, $X = \begin{bmatrix} y_1^2 \\ \vdots \\ y_n^2 \end{bmatrix}$, $E X = \begin{bmatrix} E y_1^2 \\ \vdots \\ E y_n^2 \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ \vdots \\ \sigma^2 \end{bmatrix} = \sigma^2 \mathbf{1}$
- $\widehat{\sigma^2} = F X \Rightarrow E_{Y|\sigma^2} \widehat{\sigma^2} = F E(X|\sigma^2) = \sigma^2 F \mathbf{1} = \sigma^2 \Rightarrow F \mathbf{1} = 1$
- for this problem: $\widehat{\sigma^2}_{UMVUE} = \frac{1}{n} \mathbf{1}^T X = \widehat{\sigma^2}_{BLUE}$ ($F = \frac{1}{n} \mathbf{1}^T$)



BLUE Assumptions

- unbiased: $FE(Y|\theta) = \theta$, $\forall \theta \in \Theta$

unbiasedness and the requirement that a large class of linear unbiased estimators (many F satisfying $FE(Y|\theta) = \theta$) should exist naturally lead to:

- *assumption 1* : $E(Y|\theta) = H\theta$, ($H : n \times m$)

unbiasedness $\rightarrow FH = I_m$ ($\Rightarrow n \geq m$)

- variance:

$$\begin{aligned} C_{\tilde{\theta}\tilde{\theta}} &= C_{\hat{\theta}\hat{\theta}} = E_{Y|\theta} (\hat{\theta} - E_{Y|\theta}\hat{\theta}) (\hat{\theta} - E_{Y|\theta}\hat{\theta})^T \\ &= E_{Y|\theta} (F Y - F E(Y|\theta)) (F Y - F E(Y|\theta))^T \\ &= F E_{Y|\theta} (Y - E(Y|\theta)) (Y - E(Y|\theta))^T F^T = F C_{YY}(\theta) F^T \end{aligned}$$

- *assumption 2* : $C_{YY}(\theta) = c(\theta) C$

$c(\theta)$ (> 0 , $\forall \theta$) is a scalar function of θ , $C > 0$ is constant w.r.t. θ

BLUE Optimization Problem

- $\min_{\tilde{\theta}: E_{Y|\theta}\tilde{\theta}(Y)=\theta} C_{\tilde{\theta}\tilde{\theta}} \rightarrow \min_{F: FH=I} F C F^T$
- introduce matrix square root B ($n \times n$) of $C = C^T > 0$ ($n \times n$): $C = B B^T$
notation: $B = C^{1/2}$, $C^{T/2} = (C^{1/2})^T$, $C = C^{1/2}C^{T/2}$, $C^{-1} = C^{-T/2}C^{-1/2}$
- Consider a vector space of matrices with n columns with matrix inner product $\langle X_1, X_2 \rangle = X_1 X_2^T$. Take $X_1 = H^T C^{-T/2}$, $X_2 = F C^{1/2}$. With $FH = I$:

$$\left\langle \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}, \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \right\rangle = \begin{bmatrix} H^T C^{-T/2} \\ F C^{1/2} \end{bmatrix} \begin{bmatrix} H^T C^{-T/2} \\ F C^{1/2} \end{bmatrix}^T = \begin{bmatrix} H^T C^{-1} H & I \\ I & F C F^T \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \geq 0$$

- From the Schur Complements Lemma, $R_{22} \geq R_{21}R_{11}^{-1}R_{12}$ with equality iff $X_2 = R_{21}R_{11}^{-1}X_1$.
- Hence $\min_{F: FH=I} F C F^T = (H^T C^{-1} H)^{-1}$
for $F = (H^T C^{-1} H)^{-1} H^T C^{-1} = (H^T C_{YY}^{-1} H)^{-1} H^T C_{YY}^{-1}$.
- Or $\hat{\theta}_{BLUE} = (H^T C^{-1} H)^{-1} H^T C^{-1} Y = (H^T C_{YY}^{-1} H)^{-1} H^T C_{YY}^{-1} Y$
with $C_{\tilde{\theta}\tilde{\theta}} = F C_{YY} F^T = c(\theta) F C F^T = c(\theta) (H^T C^{-1} H)^{-1} = (H^T C_{YY}^{-1} H)^{-1}$

BLUE: Example Cont'd and Recap

Example Cont'd:

- $y_i \sim \mathcal{N}(0, \sigma^2)$ i.i.d., $\theta = \sigma^2$, $x_i = y_i^2$, $\widehat{\sigma^2} = F X$
- BLUE assumptions OK: $E(X|\sigma^2) = \mathbf{1} \sigma^2 = H \theta$, $C_{XX} = 2\sigma^4 I = c(\theta) C$

$$R_{x_i x_j} = E y_i^2 y_j^2 = \begin{cases} \sigma^4 & , i \neq j \\ 3\sigma^4 & , i = j \end{cases} \Rightarrow R_{XX} = 2\sigma^4 I + \sigma^4 \mathbf{1}\mathbf{1}^T, C_{XX} = R_{XX} - m_X m_X^T = 2\sigma^4 I$$

- $\widehat{\sigma^2}_{BLUE} = (H^T C^{-1} H)^{-1} H^T C^{-1} X = \frac{1}{n} \mathbf{1}^T X = \overline{y^2}$
- $C_{\widehat{\sigma^2} \widehat{\sigma^2}}(\sigma^2) = (H^T C_{XX}^{-1} H)^{-1} = \frac{2\sigma^4}{n}$ $H = \mathbf{1}, C = I, c(\theta) = 2\sigma^4$
- note: this example is not a linear model!

Recap: BLUE assumptions:

- $\begin{cases} (1) E(Y|\theta) = H \theta \\ (2) C_{YY}(\theta) = c(\theta) C \end{cases}$

Only need to know the first two moments of $f(Y|\theta)$ which need to satisfy these assumptions. The higher-order moments of $f(Y|\theta)$: don't need to know, can be arbitrary functions of θ . So the problem should more or less look like a linear model problem, up to the second-order moments.

BLUE: Linear Model

- $Y = H\theta + V$, $EV = 0$, $EVV^T = C_{VV}$
(EV and C_{VV} independent of θ , only first two moments of V specified)

- BLUE assumptions satisfied:

$$\begin{cases} E(Y|\theta) = H\theta \\ C_{YY}(\theta) = E_{Y|\theta}(Y - E(Y|\theta))(Y - E(Y|\theta))^T = E_V VV^T = C_{VV} = C \ (c(\theta) = 1) \end{cases}$$

- $\hat{\theta}_{BLUE} = (H^T C_{VV}^{-1} H)^{-1} H^T C_{VV}^{-1} Y$ with $C_{\tilde{\theta}\tilde{\theta}} = (H^T C_{VV}^{-1} H)^{-1}$
- If $V \sim \mathcal{N}(0, C_{VV})$ then $\hat{\theta}_{BLUE} = \hat{\theta}_{ML} = \text{efficient} \Rightarrow = \hat{\theta}_{UMVUE}$

Method of Moments

Principle:

- m unknown parameters $\theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_m \end{bmatrix}$
- $f(Y|\theta)$ depends on $\theta \Rightarrow$ its moments also

- take m moments $\mu = g(\theta) = \begin{bmatrix} g_1(\theta) \\ \vdots \\ g_m(\theta) \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_m \end{bmatrix}$

such that $g(\cdot)$ is invertible, i.e. $\theta = g^{-1}(\mu)$: can determine θ from μ .

- estimate the moments: $\hat{\mu}$ (e.g. sample moments)
- method of moments: $\hat{\theta}_{MM} = g^{-1}(\hat{\mu})$

Method of Moments: Example 1

- $y_i, i = 1, \dots, n$ i.i.d., $f(y|\theta)$ mixture distribution, θ mixture parameter

$$f(y|\theta) = (1-\theta)\phi_1(y) + \theta\phi_2(y), \quad \phi_k(y) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{y^2}{2\sigma_k^2}}, k = 1, 2$$

- $\mu = E(y^2|\theta) = (1-\theta)\sigma_1^2 + \theta\sigma_2^2 = g(\theta) \Rightarrow \theta = g^{-1}(\mu) = \frac{\mu - \sigma_1^2}{\sigma_2^2 - \sigma_1^2}$

- $\widehat{\theta}_{MM} = g^{-1}(\widehat{\mu}) = \frac{\widehat{\mu} - \sigma_1^2}{\sigma_2^2 - \sigma_1^2}, \quad \widehat{\mu} = \frac{1}{n} \sum_{i=1}^n y_i^2$ sample mean squared value

- bias: $E\widehat{\theta} = \frac{1}{\sigma_2^2 - \sigma_1^2} E\widehat{\mu} - \frac{\sigma_1^2}{\sigma_2^2 - \sigma_1^2} = \frac{1}{\sigma_2^2 - \sigma_1^2} \mu - \frac{\sigma_1^2}{\sigma_2^2 - \sigma_1^2} = \theta$: unbiased



Method of Moments: Example 1 (cont'd)

$$\bullet \text{Var}(\sum \text{indep. var's}) = \sum_i \sigma_i^2$$

$$\bullet \begin{aligned} \text{Var}(\hat{\theta}) &= \text{Var}\left(\frac{1}{\sigma_2^2 - \sigma_1^2} \hat{\mu} - \frac{\sigma_1^2}{\sigma_2^2 - \sigma_1^2}\right) = \frac{1}{(\sigma_2^2 - \sigma_1^2)^2} \text{Var}(\hat{\mu}) = \frac{1}{(\sigma_2^2 - \sigma_1^2)^2} \text{Var}\left(\frac{1}{n} \sum_{i=1}^n y_i^2\right) \\ &= \frac{1}{(\sigma_2^2 - \sigma_1^2)^2} \sum_{i=1}^n \text{Var}\left(\frac{1}{n} y_i^2\right) = \frac{1}{(\sigma_2^2 - \sigma_1^2)^2} \sum_{i=1}^n \frac{1}{n^2} \text{Var}(y_i^2) = \frac{1}{(\sigma_2^2 - \sigma_1^2)^2} \frac{1}{n} \text{Var}(y^2) \end{aligned}$$

$$f(y|\theta) = (1-\theta) \phi_1(y) + \theta \phi_2(y)$$

$$\bullet \begin{aligned} \text{Var}(y^2) &= E y^4 - (E y^2)^2, & E y^2 &= (1-\theta) \sigma_1^2 + \theta \sigma_2^2 \\ E y^4 &= (1-\theta) 3\sigma_1^4 + \theta 3\sigma_2^4 \end{aligned}$$

$$\bullet \Rightarrow \text{Var}(\hat{\theta}_{MM}) = \frac{3(1-\theta)\sigma_1^4 + 3\theta\sigma_2^4 - [(1-\theta)\sigma_1^2 + \theta\sigma_2^2]^2}{n(\sigma_1^2 - \sigma_2^2)^2} \xrightarrow{n \rightarrow \infty} 0$$

$\Rightarrow \hat{\theta}_{MM}$ = mean-square consistent

MM Example 2: Sinusoid in White Noise

- $y_k = s_k + v_k = A \cos(\omega k + \phi) + v_k, \quad k = 1, \dots, n$

$$Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad S = \begin{bmatrix} s_1 \\ \vdots \\ s_n \end{bmatrix}, \quad V = \begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix}, \quad \theta = \begin{bmatrix} A \\ \omega \\ \sigma_v^2 \end{bmatrix}, \quad \Theta : A > 0, \omega \in [0, \pi], \sigma_v^2 > 0$$

- distributions: $\phi \sim \mathcal{U}[0, 2\pi]$ independent of θ, V ; $EV = 0, EVV^T = \sigma_v^2 I_n$
 randomness: $f(Y, \phi | \theta) = f(\phi | \theta) f(Y | \theta, \phi) = f(\phi) f_{V|\sigma_v^2}(Y - S(A, \omega, \phi) | \sigma_v^2)$
 below: only first and second moments of V needed, $E = E_{Y, \phi | \theta} = E_{V, \phi | \theta}$

- mean: $E_{Y, \phi | \theta} y_k = AE \cos(\omega k + \phi) + Ev_k = 0$

covariance sequence:

$$\begin{aligned} r_{yy}(i) &= E y_k y_{k+i} = A^2 E \cos(\omega k + \phi) \cos(\omega k + \phi + \omega i) \\ &\quad + AE \cos(\omega k + \phi) Ev_{k+i} + AE \cos(\omega k + \phi + \omega i) Ev_k + Ev_k v_{k+i} \\ &= \frac{A^2}{2} E \cos(2\omega k + 2\phi + \omega i) + \frac{A^2}{2} E \cos(\omega i) + \sigma_v^2 \delta_{i0} \\ &= \frac{A^2}{2} \cos(\omega i) + \sigma_v^2 \delta_{i0} \end{aligned}$$

MM Example 2: Sinusoid in White Noise (2)

- moments: $\mu = \begin{bmatrix} r_{yy}(0) \\ r_{yy}(1) \\ r_{yy}(2) \end{bmatrix} = \begin{bmatrix} \frac{A^2}{2} + \sigma_v^2 \\ \frac{A^2}{2} \cos(\omega) \\ \frac{A^2}{2} \cos(2\omega) \end{bmatrix} = g(\theta)$

- $\theta = g^{-1}(\mu): \quad \omega = \begin{cases} \arccos\left(\frac{r_{yy}(2) + \sqrt{r_{yy}^2(2) + 8r_{yy}^2(1)}}{4r_{yy}(1)}\right), & r_{yy}(1) \neq 0 \\ \frac{\pi}{2}, & r_{yy}(1) = 0 \end{cases}$

$$A = \begin{cases} \sqrt{\frac{2r_{yy}(1)}{\cos(\omega)}}, & r_{yy}(1) \neq 0 \\ \sqrt{-2r_{yy}(2)}, & r_{yy}(1) = 0 \end{cases}, \quad \sigma_v^2 = r_{yy}(0) - \frac{A^2}{2}$$

- sample moments $\hat{\mu}$: $\hat{r}_{yy}(i) = \frac{1}{n} \sum_{k=1}^{n-i} y_k y_{k+i}, \quad i = 0, 1, 2$



Method of Moments: Properties

- $\hat{\mu}$ easy to compute, $\hat{\theta}_{MM} = g^{-1}(\hat{\mu})$ straightforward if μ chosen well, hence $\hat{\theta}_{MM}$ easy to determine and easy to implement
- no optimality properties but usually consistent (since $\hat{\mu}$ consistent)
- if performance of $\hat{\theta}_{MM}$ not satisfactory, can use $\hat{\theta}_{MM}$ as initialization in an iterative optimization procedure that finds $\hat{\theta}_{ML}$



Statistical Signal Processing

Lecture 5a

chapter 1: parameter estimation: deterministic parameters
simplified estimators: BLUE, method of moments, (W)LS:

- problem formulation and solution
- linear model
- applications of the linear model
- interpretations of the LS solution
- performance analysis: bias, MSE, consistency
- acoustic echo cancellation demo, part 1
- model order reduction
- acoustic echo cancellation demo, part 2



Least-Squares (LS) Problem Formulation

- Consider n' data (signal) samples S that depend on m parameters θ

$$S = \begin{bmatrix} s_1 \\ \vdots \\ s_{n'} \end{bmatrix}, \quad \theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_m \end{bmatrix}, \quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_{n'} \end{bmatrix}, \quad V = \begin{bmatrix} v_1 \\ \vdots \\ v_{n'} \end{bmatrix}, \quad E = \begin{bmatrix} e_1 \\ \vdots \\ e_n \end{bmatrix}$$

- **nonlinear model:** model functions $g_k(\theta, S) = 0$, $k = 1, \dots, n$

- example: sinusoid: $s_k = A \cos(\omega k + \phi)$, $\theta = \omega$

can show: $s_k - 2 \cos \omega s_{k-1} + s_{k-2} = g_k(\theta, S) = 0$ (true θ) $\Rightarrow n' = n+2$

indeed, characteristic equation associated with the difference equation:

$$z^2 - 2 \cos \omega z + 1 = 0 \Rightarrow z = e^{\pm j\omega} \Rightarrow s_k = \frac{Ae^{j\phi}}{2} e^{j\omega k} + \frac{Ae^{-j\phi}}{2} e^{-j\omega k} = A \cos(\omega k + \phi)$$

- observed data: $y_k = s_k + v_k$, v_k = measurement/observation noise

- if $v_k \not\equiv 0$ (noisy observations) and/or g_k (model description) approximate, then $g_k(\theta, Y) = e_k(\theta) \not\equiv 0$, (variable θ) e_k = equation error

- LS method: introduced by Gauss in 18th century for the estimation of the parameters of elliptical orbits of planets from noisy observations.

LS Estimation

- **LS strategy:** adjust $\hat{\theta}$ to minimize the sum of squared errors $E^T E = \sum_{k=1}^n e_k^2$
- Let $G(\theta, Y) = [g_1(\theta, Y) \cdots g_n(\theta, Y)]^T$, then

$$\hat{\theta}_{LS} = \arg \min_{\hat{\theta}} G^T(\hat{\theta}, Y)G(\hat{\theta}, Y) = \arg \min_{\hat{\theta}} \sum_{k=1}^n g_k^2(\hat{\theta}, Y) = \hat{\theta}_{LS}(Y)$$

estimator $\hat{\theta}(Y)$ = function of the observations Y

- remark: LS can be formulated without any statistical context!
- **model linear in the parameters:**

$$g_k(\theta, Y) = f_k(Y) - C_k(Y)\theta, \quad f_k(Y) : 1 \times 1, \quad C_k(Y) : 1 \times m, \quad \theta : m \times 1$$

$$\bullet \text{example cont'd: let } \theta = 2 \cos \omega \Rightarrow \begin{cases} f_k(Y) = y_k + y_{k-2} \\ C_k(Y) = y_{k-1} \end{cases}$$

$$\bullet \text{Let } F(Y) = \begin{bmatrix} f_1(Y) \\ \vdots \\ f_n(Y) \end{bmatrix} : n \times 1, \quad H(Y) = \begin{bmatrix} C_1(Y) \\ \vdots \\ C_n(Y) \end{bmatrix} : n \times m$$

$$\bullet \text{LS: } \hat{\theta}_{LS} = \arg \min_{\theta} [F(Y) - H(Y)\theta]^T [F(Y) - H(Y)\theta] = \hat{\theta}_{LS}(Y)$$

LS: Discussion

$$\bullet F(Y) - H(Y)\theta = \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} - \begin{bmatrix} C_1 \\ \vdots \\ C_n \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_m \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} - [H_1 \cdots H_m] \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_m \end{bmatrix} = \begin{bmatrix} e_1 \\ \vdots \\ e_n \end{bmatrix} = E$$

n equations, m unknowns θ (if try to make $E = 0$)

- $n > m$: *overdetermined* case

exact fit impossible \Rightarrow least-squares fit

(assume: H = full rank = full column rank \Rightarrow unique solution)

- $n = m$: *exactly determined* case

if H = full rank $\Rightarrow H^{-1}$ exists $\Rightarrow \hat{\theta} = H^{-1}F$ = unique solution

(no averaging of errors though)

- $n < m$: *underdetermined* case

∞^{m-n} solutions exist, there is a unique solution of minimum norm $\|\hat{\theta}\|$

- assume henceforth: $n > m$, $\text{rank}(H) = m$

then parameters *identifiable*: θ can be found exactly if optimal $E(\theta) = 0$



LS: Solution

- LS: $\widehat{\theta}(Y) = \arg \min_{\theta} \xi_{LS}(\theta, Y)$

$$\begin{aligned}\xi_{LS}(\theta, Y) &= \|F(Y) - H(Y)\theta\|_2^2 \\ &= [F(Y) - H(Y)\theta]^T [F(Y) - H(Y)\theta] \\ &= [F^T(Y) - \theta^T H^T(Y)] [F(Y) - H(Y)\theta]\end{aligned}$$

- $\frac{\partial \xi_{LS}}{\partial \theta} = -2H^T(Y) [F(Y) - H(Y)\theta] = 0 \Rightarrow H^T(Y)H(Y)\theta = H^T(Y)F(Y)$
 $\Rightarrow \widehat{\theta}_{LS} = (H^T(Y)H(Y))^{-1} H^T(Y)F(Y) = \widehat{\theta}_{LS}(Y)$

- Hessian $= \frac{\partial}{\partial \theta} \left(\frac{\partial \xi_{LS}}{\partial \theta} \right)^T = 2H^T(Y)H(Y) > 0$ since $H(Y)$ full column rank
(constant w.r.t. θ)
 \Rightarrow extremum = minimum, only one \Rightarrow global one

LS: Linear Model

- $\begin{cases} F(Y) = Y \\ H(Y) = H \end{cases} \rightarrow \begin{cases} y_k = C_k \theta + v_k, \quad k = 1, \dots, n & v_k = \text{error} \\ Y = H \theta + V \\ = \sum_{i=1}^m H_i \theta_i + V & \begin{cases} H \theta = S = \text{signal component} \\ V = \text{noise} \end{cases} \end{cases}$
- $\hat{\theta}_{LS} = (H^T H)^{-1} H^T Y$
- example 1: amplitude and phase estimation of a noisy sinusoid (ω known)

$$\begin{aligned}
 y_k &= A \cos(\omega k + \phi) + v_k \\
 &= A \cos \phi \cos(\omega k) - A \sin \phi \sin(\omega k) + v_k \\
 &= \underbrace{[\cos(\omega k) \quad \sin(\omega k)]}_{C_k} \underbrace{\begin{bmatrix} A \cos \phi \\ -A \sin \phi \end{bmatrix}}_{\theta} + v_k
 \end{aligned}$$

- example 2: line fitting

$$y_k = a x_k + b + v_k = \underbrace{[x_k \quad 1]}_{C_k} \underbrace{\begin{bmatrix} a \\ b \end{bmatrix}}_{\theta} + v_k$$



Weighted Least-Squares (WLS)

non-linear model

- WLS: $\min_{\theta} E^T W E$, $E = [e_1 \cdots e_n]^T$

$$\hat{\theta}_{WLS} = \arg \min_{\theta} G^T(\theta, Y) W G(\theta, Y), \quad W = W^T > 0 \text{ weighting matrix}$$

- LS: $W = I$, $\Rightarrow E^T E = \sum_{k=1}^n e_k^2$

model linear in parameters

- WLS: $\min_{\theta} \xi_{WLS}(\theta, Y) = \min_{\theta} [F(Y) - H(Y)\theta]^T W [F(Y) - H(Y)\theta]$

$$\begin{aligned} \bullet \frac{\partial \xi_{WLS}}{\partial \theta} &= -2H^T(Y) W [F(Y) - H(Y)\theta] = 0 \\ \Rightarrow \hat{\theta}_{WLS} &= (H^T(Y) W H(Y))^{-1} H^T(Y) W F(Y) = \hat{\theta}_{WLS}(Y) \end{aligned}$$

$$\bullet \text{Hessian} = \frac{\partial}{\partial \theta} \left(\frac{\partial \xi_{WLS}}{\partial \theta} \right)^T = 2H^T(Y) W H(Y) > 0$$

since $W > 0$ and $H(Y)$ full column rank

\Rightarrow extremum = minimum, only one \Rightarrow global one



3 Quantities of Potential Interest

model linear in parameters: $F(Y) = H(Y)\theta + E$

linear model: $Y = H\theta + V \quad (F(Y), H(Y), E) = (Y, H, V)$

- 3 quantities:
- parameters: θ
 - signal: $S = H\theta$
 - error/noise: $E = F(Y) - H(Y)\theta$ or $V = Y - H\theta$

LS estimates:

• parameters: $\hat{\theta} = (H^T H)^{-1} H^T F$

• signal: $\hat{S} = H\hat{\theta} = P_H F$, $P_H = H(H^T H)^{-1} H^T$

projection of F/Y on the *signal subspace* = column space of H

• error/noise: $\hat{E} = F - \hat{S} = F - H\hat{\theta} = P_H^\perp F$, $P_H^\perp = I - P_H$

projection of F/Y on the *noise subspace* = orthogonal complement of column space of H

P = projection matrix if $P = P^T$ (symmetric) and $P P = P$ (idempotent)

eigenvectors/values of P_H (P_H^\perp): $P_H H = H$, $P_H^\perp H = 0$

basis vectors of signal subspace, corresponding to eigenvalue 1 (0),
basis vectors of noise subspace, corresponding to eigenvalue 0 (1).



Applications

- linear model
 - 1. polynomial curve fitting / modal analysis
 - 2. filter design
- model linear in parameters
 - 3. optimal/adaptive filtering



Application 1: Polynomial Curve fitting/Modal Analysis

- measurements $y_k = \text{signal} + \text{noise}$
signal is a linear combination of known basis functions $h_i(k)$ (*modes*)

$$y_k = s_k + v_k = \sum_{i=1}^m \theta_i h_i(k) + v_k = c_k^T \theta + v_k$$

where $c_k^T = [h_1(k) \cdots h_m(k)]$. The linear combination coefficients θ_i are the parameters.

- typical signal model: solution of a homogenous difference equation with constant coefficients;

$$s_k = \sum_{i=1}^{m_0} \left(\sum_{j=1}^{m_i} \alpha_{ij} k^{j-1} \right) \lambda_i^k = c_k^T \theta$$

$$c_k^T = [k^0 \lambda_1^k \cdots k^{m_1-1} \lambda_1^k \quad k^0 \lambda_2^k \cdots k^{m_{m_0}-1} \lambda_{m_0}^k]$$

$$\theta^T = [\alpha_{11} \cdots \alpha_{1m_1} \quad \alpha_{21} \cdots \alpha_{m_0 m_{m_0}}]$$

for m_0 distinct roots λ_i with multiplicity m_i .

Applic. 1: Polynomial Curve fitting/Modal Analysis (2)

- The signal s_k is the solution of the following difference equation

$$\prod_{i=1}^{m_0} (1 - \lambda_i q^{-1})^{m_i} s_k = 0$$

where q^{-1} is the delay operator: $q^{-1}s_k = s_{k-1}$ (q^{-1} transforms to a multiplication by z^{-1} when taking the z -transform). The total order of the difference equation is $m = \sum_{i=1}^{m_0} m_i$.

- particular case 1: $m_0 = 1$ root and $\lambda_1 = 1$: s_k is a polynomial function of k .
In particular, if $m_1 = 1$, then $(1 - q^{-1}) s_k = s_k - s_{k-1} = 0$ and $s_k \equiv b$ is a constant.
If $m_1 = 2$, then $s_k - 2s_{k-1} + s_{k-2} = 0$ and $s_k = a k + b$ (example 1 above).
- particular case 2: m_0 even, λ_i on the unit circle ($\lambda_i = e^{j\omega_i}$) and occurring in complex conjugate pairs, and $m_i = 1$, $\forall i$. Useful reparameterization:

$$s_k = \sum_{i=1}^{m_0/2} \left(\alpha_i e^{j\omega_i k} + \alpha_i^* e^{-j\omega_i k} \right) = \sum_{i=1}^{m_0/2} (a_i \cos(\omega_i k) + b_i \sin(\omega_i k)) .$$

(see example 2 above: $m_0 = 2$)

Application 2: Filter Design

IIR filter design in the time domain

- IIR model transfer function: $\frac{\mathbf{B}(z)}{\mathbf{A}(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_p z^{-p}}{1 + a_1 z^{-1} + \cdots + a_r z^{-r}}$

parameters $\theta = [a_1 \cdots a_r \ b_0 \ b_1 \cdots b_p]^T$, $m = p + q + 1$

- IIR model impulse response: $s_k = \frac{\mathbf{B}(q)}{\mathbf{A}(q)} \delta_{k0}$

Kronecker delta: $\delta_{ij} = \begin{cases} 1 & , \ i = j \\ 0 & , \ i \neq j \end{cases}$

- target impulse response (causal, truncated): $y_k = s_k + v_k$, $k = 0, 1, \dots, n$

error $v_k = y_k - \frac{\mathbf{B}(q)}{\mathbf{A}(q)} \delta_{k0}$ nonlinear in parameters θ

- consider $\mathbf{A}(q) y_k = \mathbf{B}(q) \delta_{k0} + \underbrace{\mathbf{A}(q) v_k}_{e_k}$ or $e_k = y_k + \sum_{i=1}^r a_i y_{k-i} - b_k$
error e_k linear in the parameters $(b_k = 0, k > p)$

Application 2: Filter Design (2)

- with $Y = [y_0 \ y_1 \cdots \ y_n]^T$, $E = [e_0 \ e_1 \cdots \ e_n]^T$, $B = [b_0 \ b_1 \cdots \ b_n]^T$, we can write

$$E = \mathcal{A}Y - B = Y - H\theta , \quad H = [-\mathcal{Y} \ \mathcal{I}]$$

where

$$\mathcal{A}(\theta) = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ a_1 & \ddots & \ddots & \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_r & & \ddots & \ddots & \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & a_r & \cdots & a_1 & 1 \end{bmatrix}, \quad \mathcal{Y} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ y_0 & 0 & \cdots & 0 \\ y_1 & y_0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ y_{n-1} & y_{n-2} & \cdots & y_{n-r} \end{bmatrix}, \quad \mathcal{I} = \begin{bmatrix} I_{p+1} \\ 0 \end{bmatrix}$$

\mathcal{A} and \mathcal{Y} are Toeplitz (elements along a diagonal are the same), hence they are specified by their first row and column; they are also lower triangular, and \mathcal{A} is banded (limited number of non-zero diagonals).

For filtering with \mathcal{A} : Toeplitzness corresponds to time-invariance, triangularity to causality and bandedness to FIR.

- Strictly speaking: model linear in parameters: $F = Y$ but $H(Y)$ depends on Y .
- LS solution: $\hat{\theta}_{LS} = (H^T H)^{-1} H^T Y = \arg \min_{\theta} E^T E$

Application 2: Filter Design (3)

- Assume now that we insist on obtaining the LS solution in the *output error* V rather than the *equation error* $E = \mathcal{A}V$ (corresponding to $e_k = \mathbf{A}(q) v_k$).
- Observe that we have

$$V = \mathcal{A}^{-1} E = \mathcal{A}^{-1} (Y - H\theta)$$

- We can obtain the LS solution $\arg \min_{\theta} V^T V$ iteratively as follows. Note

$$V^T V = \left\| \mathcal{A}^{-1} (Y - H\theta) \right\|_2^2 = (Y - H\theta)^T (\mathcal{A}\mathcal{A}^T)^{-1} (Y - H\theta)$$

Hence the solution $\widehat{\theta}^{(i)}$ at iteration i can be obtained as

$$\widehat{\theta}_{WLS}^{(i)} = \left(H^T W^{(i)} H \right)^{-1} H^T W^{(i)} Y \text{ where } W^{(i)} = \left(\mathcal{A}(\widehat{\theta}^{(i-1)}) \mathcal{A}^T(\widehat{\theta}^{(i-1)}) \right)^{-1}$$

- Initialization: e.g. $\widehat{\theta}_{WLS}^{(0)} = 0$ so that $\widehat{\theta}_{WLS}^{(1)} = \widehat{\theta}_{LS}$ ($\mathcal{A}(0) = I \Rightarrow W^{(1)} = I$).
- Note: $V^T V = E^T W E$: the LS problem in the output error V corresponds to a WLS problem in the equation error E .
- known as Steiglitz-McBride iterations

Application 2: Filter Design (4)

FIR filter design in the frequency domain

- FIR filter $B(z) = C(z)B$, $C(z) = [1 \ z^{-1} \cdots z^{-p}]$, $\theta = B = [b_0 \ b_1 \cdots b_p]^T$
- We wish to fit the frequency response $B(e^{j2\pi f})$ to a desired response y_i at frequency f_i , $i = 1, \dots, n'$:

$$y_i = C(e^{j2\pi f_i})\theta + v_i, \quad i = 1, \dots, n'$$

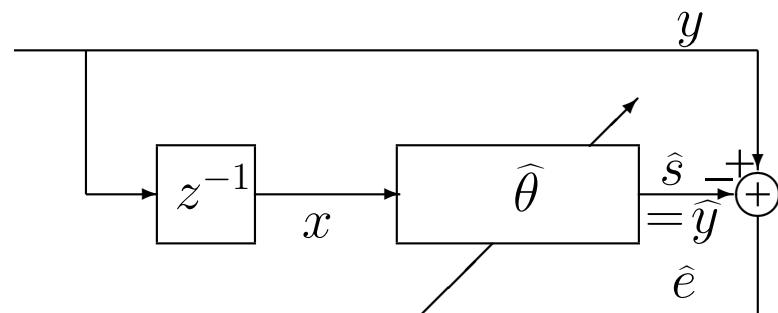
where v_i here is clearly not noise but approximation error.

- Then $\widehat{\theta}_{LS} = (H^T H)^{-1} H^T Y$ where

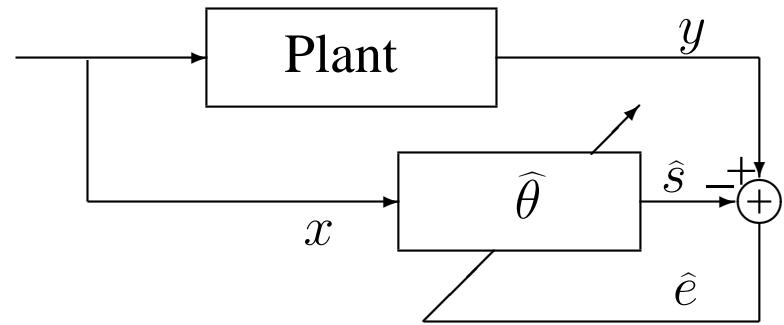
$$Y' = \begin{bmatrix} y_1 \\ \vdots \\ y_{n'} \end{bmatrix}, \quad H' = \begin{bmatrix} C(e^{j2\pi f_1}) \\ \vdots \\ C(e^{j2\pi f_{n'}}) \end{bmatrix}, \quad Y = \begin{bmatrix} \Re Y' \\ \Im Y' \end{bmatrix}, \quad H = \begin{bmatrix} \Re H' \\ \Im H' \end{bmatrix}$$

- For the design of a filter with real coefficients $\theta = B$, the distribution of the frequency points f_i can be limited to the normalized frequency interval $[0, \frac{1}{2}]$.
- A weighting matrix $W = \text{blockdiag}\{W', W'\}$, $W' = \text{diag}\{w_1, \dots, w_{n'}\}$ can be introduced to put a higher weight $w_i > 0$ at frequencies f_i where a tighter fit is desired ($V^T W V = V'^H W' V' = \sum_{i=1}^{n'} w_i |v_i|^2$ where $V^H = (V^*)^T$).

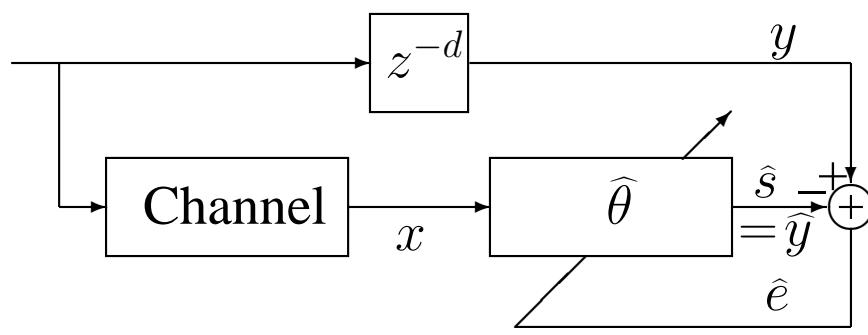
Application 3: Adaptive Filtering



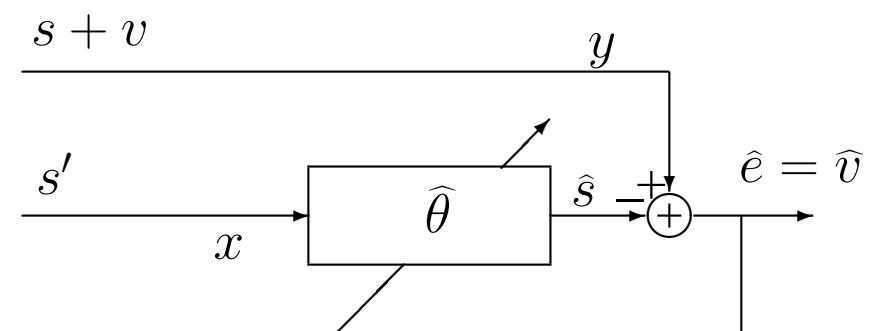
prediction, spectral estimation, whitening



system identification



equalization, deconvolution



interference canceling

Application 3: Adaptive Filtering (2)

- adaptive filtering terminology: y_k = desired-response signal, x_k = filter input
- strictly speaking: adaptive filtering = application of model linear in parameters because H contains signal
- adaptive filtering cases:

I. single-channel FIR filtering (4 cases): previous figure with $\theta = B$ ($m = p+1$) = FIR filter impulse response: $y_{1:n} = [y_1 \cdots y_n]^T = H\theta + V$ with

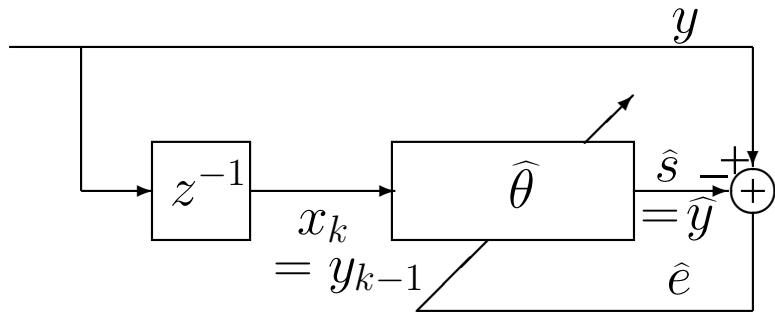
$$H = H(x_{2-m:n}) = \begin{bmatrix} x_1 & x_0 & \cdots & x_{2-m} \\ x_2 & x_1 & \cdots & x_{3-m} \\ \vdots & \vdots & \ddots & \vdots \\ x_n & x_{n-1} & \cdots & x_{n-m+1} \end{bmatrix}, \quad \theta = B = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_p \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{m-1} \end{bmatrix}, \quad Y = \begin{bmatrix} y_{1:n} \\ x_{2-m:n} \end{bmatrix}$$

H is Toeplitz. $E = V$ in this case.

II. multichannel applications: (combinations of:)

- * IIR filters formulated as multichannel FIR filters
- * multirate FIR filters
- * vector input signals: spatial filtering (beamforming)/spatiotemporal filtering of multiple sensor (antennas/sensors) signals
- * other multidimensional signals (images)

Application 3: Adaptive Filtering (3)



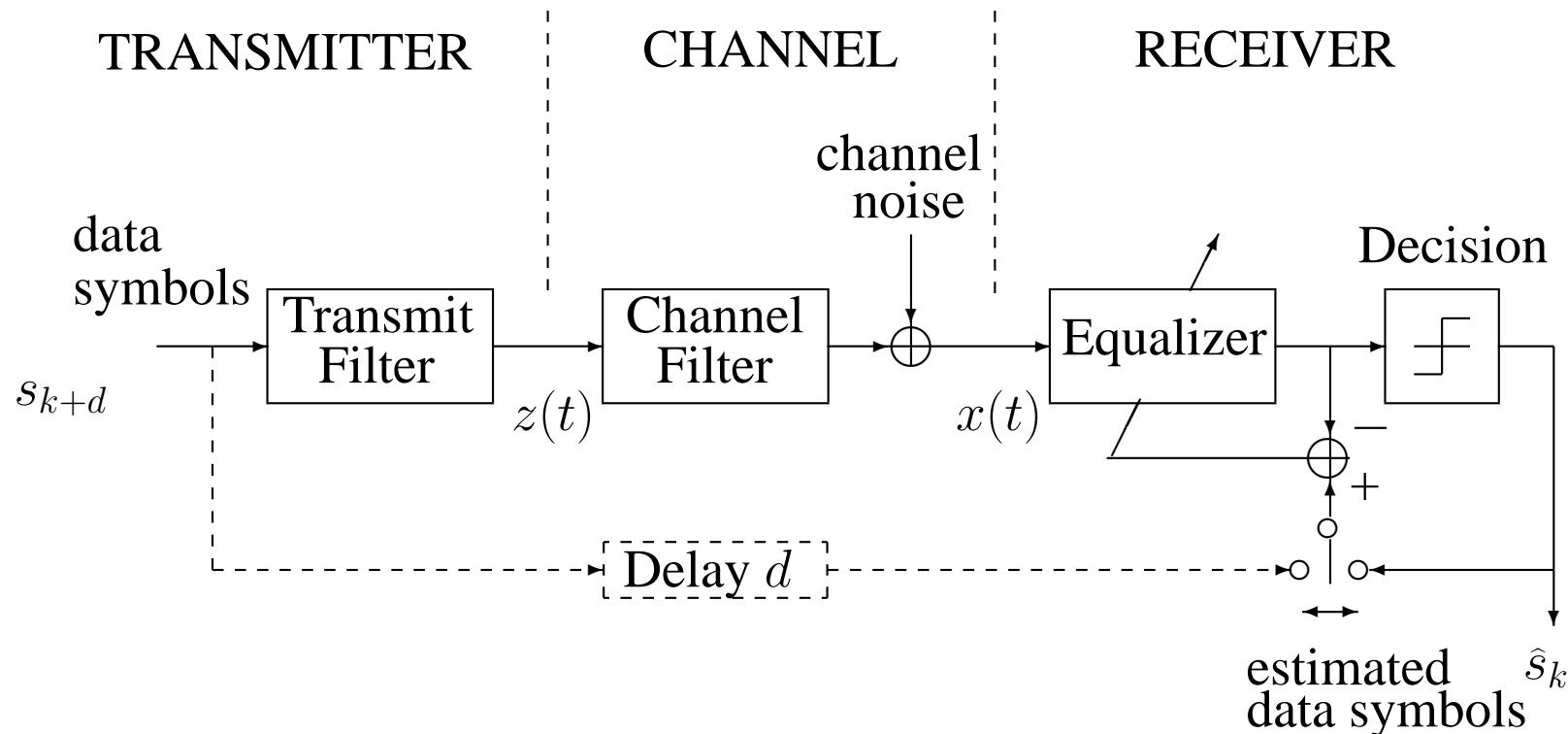
prediction, spectral estimation, whitening

- here $x_k = y_{k-1} \Rightarrow x_k$ noisy also
- **prediction** = s_k , e.g. stock market (multidimensional signals though)
- **whitening**: make prediction error e_k as white as possible (unpredictable part): used in signal coding (e_k easier to quantize than y_k)
- **spectral estimation/modeling**: when prediction error e_k becomes white (uncorrelated), θ contains all the spectral (correlation) information of y_k

Application 3: Adaptive Filtering (4)

- **equalization, deconvolution:**

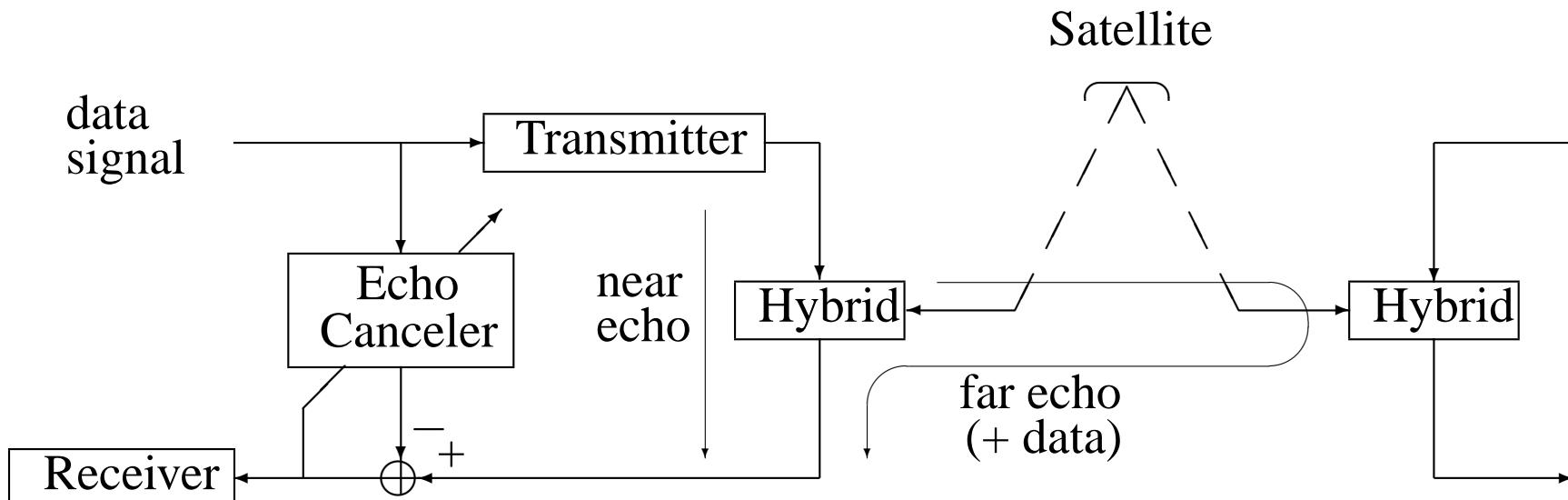
- s_k of interest here (transmitted symbols, original image/object)
- the noise is here situated at the filter input x_k instead of at the filter output y_k
- recovery of original image from a blurred version
- reconstruction of 3D object from 2D images
- channel equalization in communications:



Application 3: Adaptive Filtering (5)

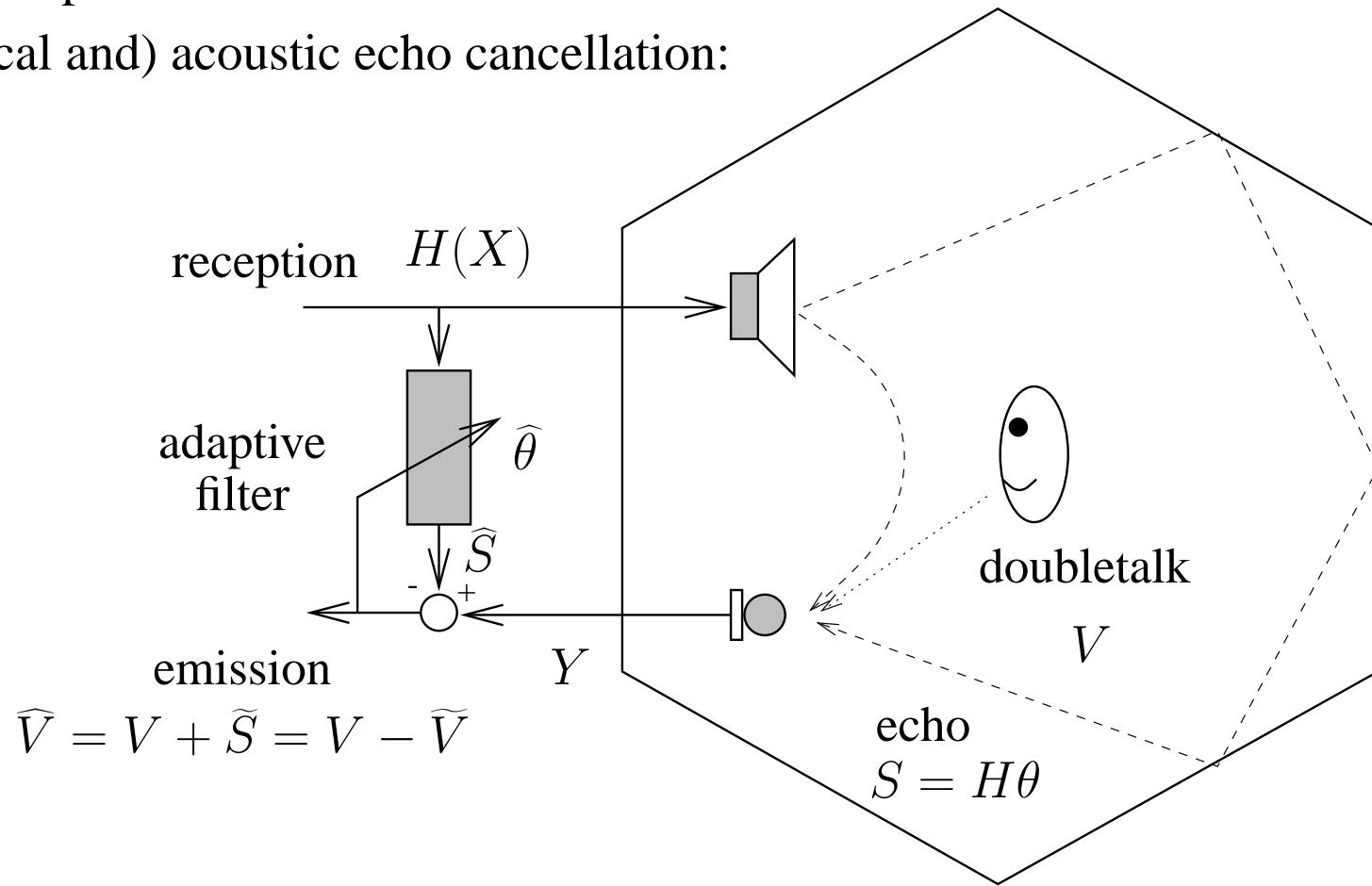
- **interference cancellation:** $e_k = v_k$ signal of interest, corrupted by unmeasurable noise s_k , which is correlated with the measurable noise $s'_k = x_k$ applications:

- acoustic (motor) noise reduction for handsfree telephony systems in cars
- fan/air conditioning system noise reduction in teleconferencing systems
- 50 Hz interference in electrocardiography
- interference from other users in mobile communications
- electrical echo cancellation in telephone lines (voiceband modems/xDSL):



Application 3: Adaptive Filtering (6)

- **system identification:** θ (filter) of interest, examples:
 - channel identification
 - automatic control
 - seismic exploration
 - (electrical and) acoustic echo cancellation:



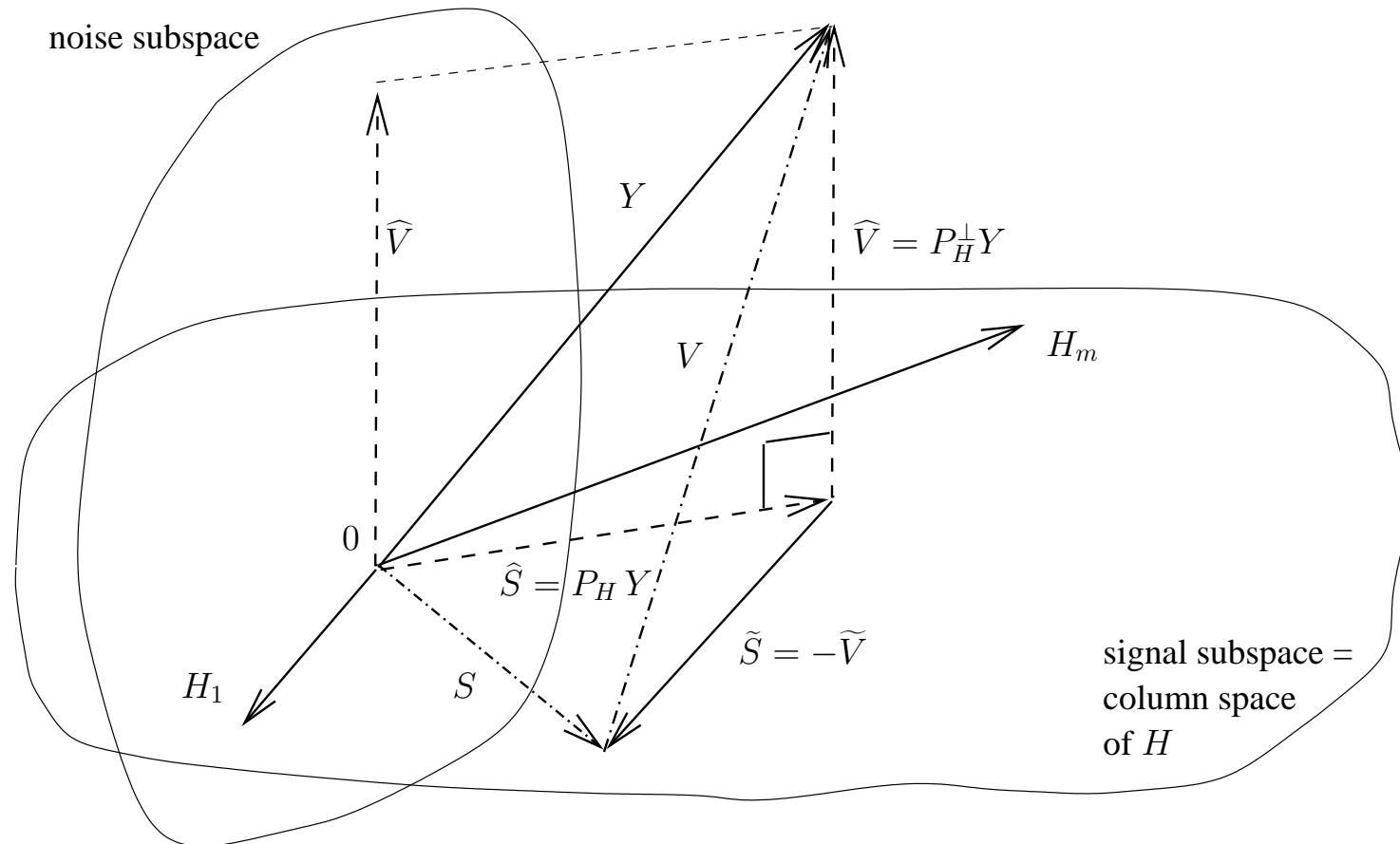
Orthogonality Principle of LS

- we found that $\widehat{\theta}_{LS}$ satisfies

orthogonality conditions of LS

$$H^T (Y - H\widehat{\theta}_{LS}) = H^T \widehat{V}_{LS} = 0 \Leftrightarrow H_i^T \widehat{V}_{LS} = 0, \quad i = 1, \dots, m$$

the smallest fitting error is orthogonal to the signal subspace (column space of H)
linear model notation assumed here





Correlation and Covariance Matrices

- random vectors X and Y

- mean: $m_X = E X$, $m_Y = E Y$ $(E = \text{Expectation})$

- correlation matrix: $R_{XY} = E XY^T$, $R_{XX} = E XX^T$

- covariance matrix:

$$C_{XY} = R_{X-m_X, Y-m_Y} = E (X - m_X)(Y - m_Y)^T = R_{XY} - m_X m_Y^T$$

- vector power (mean square value):

$$\begin{aligned} E \|X\|^2 &= \text{tr} \{ E \|X\|^2 \} = E \text{tr} \{ \|X\|^2 \} = E \text{tr} \{ X^T X \} \\ &= E \text{tr} \{ X X^T \} = \text{tr} \{ E X X^T \} = \text{tr} \{ R_{XX} \} \end{aligned}$$

- notation: $\begin{cases} \theta = \widehat{\theta} + \widetilde{\theta} \\ S = \widehat{S} + \widetilde{S} \\ V = \widehat{V} + \widetilde{V} \end{cases}$

Performance Analysis of LS in the Linear Model

- *a priori* and *a posteriori* decompositions of Y :

$$Y = \underbrace{S + V}_{\text{a priori decomposition}} = \underbrace{\widehat{S} + \widehat{V}}_{\text{a posteriori decomposition}}$$

where $\widehat{S} \perp \widehat{V}$: $\widehat{S}^T \widehat{V} = \widehat{\theta}^T H^T \widehat{V} = 0$

- estimator **bias**: average deviation from the true parameter (E = Expectation)

$$b_{\widehat{\theta}}(\theta) = -E\bar{\theta} = E(\widehat{\theta}(Y) - \theta) = E\widehat{\theta}(Y) - \theta$$

unbiased estimator: $b_{\widehat{\theta}}(\theta) = 0, \forall \theta \in \Theta$ (set of possible values for θ)

Unbiasedness is a weak property: estimator can be correct on the average, but with large deviations (large MSE). Also, good estimators exist that are biased.

- **MSE** = $\text{tr}\{R_{\tilde{\theta}\tilde{\theta}}\} = E\|\tilde{\theta}\|_2^2$, $R_{\tilde{\theta}\tilde{\theta}} = E\tilde{\theta}\tilde{\theta}^T$ = estimation error correlation matrix

$$\begin{aligned} R_{\tilde{\theta}\tilde{\theta}} &= E(\widehat{\theta} - \theta)(\widehat{\theta} - \theta)^T = E[\widehat{\theta}(-E\widehat{\theta} + E\widehat{\theta}) - \theta][\widehat{\theta}(-E\widehat{\theta} + E\widehat{\theta}) - \theta]^T \\ &= E(\widehat{\theta} - E\widehat{\theta})(\widehat{\theta} - E\widehat{\theta})^T + (E\widehat{\theta} - \theta)(E\widehat{\theta} - \theta)^T = C_{\widehat{\theta}\widehat{\theta}} + b_{\widehat{\theta}}(\theta)b_{\widehat{\theta}}^T(\theta) = C_{\tilde{\theta}\tilde{\theta}} + b_{\tilde{\theta}}(\theta)b_{\tilde{\theta}}^T(\theta) \end{aligned}$$

$\text{tr}\{R_{\tilde{\theta}\tilde{\theta}}\} = \text{tr}\{C_{\tilde{\theta}\tilde{\theta}}\} + \|b_{\tilde{\theta}}\|^2$: Mean Squared Error = variance + bias squared

- (mean square) **consistency**: if $\text{MSE}(n) \xrightarrow{n \rightarrow \infty} 0$, then $\widehat{\theta} \xrightarrow{n \rightarrow \infty} \theta$ (in mean square)

Performance Analysis of LS in the Linear Model (2)

- No statistical information (about V) needed to derive $\hat{\theta}_{WLS}$. However, in order to evaluate its performance (for the linear model), we need to introduce a stochastic context: V random with
$$\begin{cases} EV = 0 \\ E VV^T = C_{VV} \end{cases}$$
- note: $\hat{\theta}_{WLS} - \theta = (H^T W H)^{-1} H^T W (H \theta + V) - \theta = (H^T W H)^{-1} H^T W V$
- $b_{WLS} = E \hat{\theta}_{WLS} - \theta = (H^T W H)^{-1} H^T W EV = 0$: unbiased if $EV = 0$
- $C_{\tilde{\theta}\tilde{\theta}}(W) = C_{\hat{\theta}\hat{\theta}}(W) = (H^T W H)^{-1} H^T W C_{VV} W H (H^T W H)^{-1}$
- optimal weighting: $W = C_{VV}^{-1}$: $C_{\tilde{\theta}\tilde{\theta}}(W) \geq C_{\tilde{\theta}\tilde{\theta}}(C_{VV}^{-1}) = (H^T C_{VV}^{-1} H)^{-1}$
- LS: $C_{\tilde{\theta}\tilde{\theta}} = C_{\tilde{\theta}\tilde{\theta}}(I) = (H^T H)^{-1} H^T C_{VV} H (H^T H)^{-1}$
- white noise: $C_{VV} = \sigma_v^2 I_n \Rightarrow \text{WLS}^{opt} = \text{LS}$ and $C_{\tilde{\theta}\tilde{\theta}} = \sigma_v^2 (H^T H)^{-1}$
- (W)LS in general consistent: $\hat{\theta} \rightarrow \theta$ as $\frac{n}{m} \rightarrow \infty$

Performance Analysis of LS in the Linear Model (3)

- consider LS and white noise ($C_{VV} = \sigma_v^2 I$)

- **signal component:**

$$\widehat{S} = H\widehat{\theta}_{LS} = P_H Y = S + P_H V \Rightarrow \widetilde{S} = S - \widehat{S} = -P_H V$$

* Hence, $E \widehat{S} = S$: unbiased if $E V = 0$.

* $C_{\widetilde{S}\widetilde{S}} = P_H C_{VV} P_H = \sigma_v^2 P_H \Rightarrow E\|\widetilde{S}\|^2 = \text{tr}\{C_{\widetilde{S}\widetilde{S}}\} = \sigma_v^2 \text{tr}\{P_H\} = m \sigma_v^2$ remains finite!

* Even $C_{\tilde{s}_k \tilde{s}_k} = \sigma_{\tilde{s}_k}^2 = \sigma_v^2 [P_H]_{kk}$ ($= \sigma_v^2 \frac{m}{n}$ on the avg.) $\xrightarrow{\frac{n}{m} \rightarrow \infty} 0$: \hat{s}_k consistent.

$$\frac{1}{n} \sum_{k=1}^n [P_H]_{kk} = \frac{1}{n} \text{tr}\{P_H\} = \frac{1}{n} \text{tr}\{H(H^T H)^{-1} H^T\} = \frac{1}{n} \text{tr}\{(H^T H)^{-1} H^T H\} = \frac{1}{n} \text{tr}\{I_m\} = \frac{m}{n}$$

- **noise component:**

$$\widehat{V} = Y - H\widehat{\theta}_{LS} = P_H^\perp Y = P_H^\perp V \Rightarrow \widetilde{V} = V - \widehat{V} = P_H V$$

* Hence, $E \widehat{V} = 0$: unbiased if $E V = 0$ (case of a “random parameter”).

* $C_{\widetilde{V}\widetilde{V}} = C_{\widetilde{S}\widetilde{S}} \Rightarrow E\|\widetilde{V}\|^2 = m \sigma_v^2$ remains finite also!

* Furthermore $C_{\tilde{v}_k \tilde{v}_k} = \sigma_{\tilde{v}_k}^2 = \sigma_{\tilde{s}_k}^2 \xrightarrow{\frac{n}{m} \rightarrow \infty} 0$: \hat{v}_k consistent also. ($\text{SNR} = \frac{\sigma_{v_k}^2}{\sigma_{\tilde{v}_k}^2} = \frac{n}{m}$)

- observe: $R_{\widehat{S}\widehat{V}} = E \widehat{S}\widehat{V}^T = P_H C_{VV} P_H^\perp = \sigma_v^2 P_H P_H^\perp = 0$: a posteriori signal $\widehat{S} = S + P_H V$ and noise components $\widehat{V} = P_H^\perp V$ are uncorrelated



Perf Analysis of LS in FIR System Identification

- recall: $H = [X_1 \ X_2 \ \cdots \ X_n]^T$, $X_i = [x_i \ x_{i-1} \ \cdots \ x_{i-m+1}]^T$
- linear model: H deterministic \rightarrow model linear in parameters: H can be stochast.
- law of large numbers: $\frac{1}{n} H^T H = \frac{1}{n} \sum_{i=1}^n X_i X_i^T \xrightarrow{n \rightarrow \infty} E X_i X_i^T = R_{XX}$ ($m \times m$)
 \Rightarrow approximation: $H^T H \approx n R_{XX}$
- observe: if x_k and v_k are independent and at least one of them is white noise ($R_{XX} = \sigma_x^2 I$ and/or $R_{VV} = \sigma_v^2 I$), then $E H^T R_{VV} H = n \sigma_v^2 R_{XX}$
- hence $C_{\tilde{\theta}\tilde{\theta}} = (H^T H)^{-1} H^T C_{VV} H$ ($H^T H$) $^{-1} \approx \frac{\sigma_v^2}{n} R_{XX}^{-1}$ (\Rightarrow consistency)
- Is LS criterion $= \|\widehat{V}\| = Y^T P_H^\perp Y$ a good indicator of estimation quality?
($\widehat{V} = Y - H\widehat{\theta} = \text{LS error}$)

$$\begin{aligned} E \|\widehat{V}\|^2 &= E Y^T P_H^\perp Y = E V^T P_H^\perp V = E V^T V - E \{V^T P_H V\} \\ &= E \sum_{i=1}^n v_i^2 - \text{tr}\{E P_H V V^T\} = n \sigma_v^2 - \text{tr}\{E P_H C_{VV}\} \\ &= n \sigma_v^2 - \text{tr}\{E (H^T H)^{-1} H^T C_{VV} H\} \xrightarrow{\text{LLN}} n \sigma_v^2 - \text{tr}\{(E H^T H)^{-1} E H^T C_{VV} H\} \\ &= n \sigma_v^2 - \text{tr}\{(n R_{XX})^{-1} n \sigma_v^2 R_{XX}\} = n \sigma_v^2 - \sigma_v^2 \text{tr}\{I_m\} = (n - m) \sigma_v^2 \end{aligned}$$

hence $E \|\widehat{V}\|^2 \rightarrow 0$ as $m \nearrow n$ (or $n \searrow m$). Extreme case: $n = m \Rightarrow \widehat{V} = 0$.
But estimation not good at all.

Perf Analysis of LS in FIR System Identification (2)

- *white noise case:*
$$\begin{aligned} E \|\widehat{V}\|^2 &= E V^T P_H^\perp V = \text{tr} \{P_H^\perp E V V^T\} \\ &= \sigma_v^2 \text{tr} \{P_H^\perp\} = \sigma_v^2 \text{tr} \{I_n - P_H\} = \sigma_v^2 (n-m) \end{aligned}$$

- “signal” and “noise” parts:
$$\begin{cases} Y = S + V \\ \widehat{S} = S - \widetilde{S} \\ \widehat{V} = V - \widetilde{V} \end{cases}$$

- A priori SNR: $\text{SNR}_Y = \frac{E \|S\|^2}{E \|V\|^2} = \frac{n E s_i^2}{n E v_i^2} = \frac{E (\theta^T X_i)^2}{\sigma_v^2} = \frac{\theta^T R_{XX} \theta}{\sigma_v^2}$

A posteriori SNRs:

$$\text{SNR}_{\widehat{S}} = \frac{E \|S\|^2}{E \|\widetilde{S}\|^2} = \frac{n E s_i^2}{m \sigma_v^2} = \frac{n}{m} \text{SNR}_Y$$

$$\text{SNR}_{\widehat{V}} = \frac{E \|V\|^2}{E \|\widetilde{V}\|^2} = \frac{n \sigma_v^2}{m \sigma_v^2} = \frac{n}{m} \quad \text{indep. of } \text{SNR}_Y !$$

- For $n = m$: $\text{SNR}_{\widehat{S}} = \text{SNR}_Y$ (estimation did not improve SNR!),

$\text{SNR}_{\widehat{V}} = 1 = 0\text{dB}$ (LS error $\widehat{V} = 0 \Rightarrow \widetilde{V} = V$)

Perf Analysis of LS in FIR System Identification (3)

- *cross validation*: to get an idea of estimation quality, try estimate $\widehat{\theta}(Y)$ on n' other data $Y' = S' + V'$, $S' = H'\theta$ (independent from Y but identically distributed). In practice: often $n' = 1$ (1 new sample)
- **signal component**:

$$\begin{aligned}\widehat{S}' &= H'\widehat{\theta}_{LS} = H'(H^T H)^{-1}H^T Y = S' + H'(H^T H)^{-1}H^T V \\ \Rightarrow \widetilde{S}' &= S' - \widehat{S}' = -H'(H^T H)^{-1}H^T V\end{aligned}$$

- * can show $E \|\widetilde{S}'\|^2 \approx \frac{n'}{n} m \sigma_v^2$
- * hence $\text{SNR}_{\widetilde{S}'} = \frac{E \|S'\|^2}{E \|\widetilde{S}'\|^2} = \frac{n}{m} \text{SNR}_Y$ as before
- **noise component**:

$$\widehat{V}' = Y' - H'\widehat{\theta}_{LS} = V' + \widetilde{S}' \Rightarrow \widetilde{S}' = -\widehat{V}'$$

- * $\text{SNR}_{\widehat{V}'} = \frac{E \|V'\|^2}{E \|\widehat{V}'\|^2} = \frac{n}{m}$ but $E \|\widehat{V}'\|^2 = n' \sigma_v^2 (1 + \frac{m}{n}) > E \|V'\|^2$ now
- * this time also $R_{\widehat{V}'V'} = 0$ whereas $R_{\widehat{V}V} = P_H R_{VV} \neq 0$ before
- * to predict performance from \widehat{V} : $\frac{1}{n'} \|\widehat{V}'\|^2 \approx \frac{n+m}{n-m} \frac{1}{n} \|\widehat{V}\|^2$ (Akaike's FPEC)
- conclusion: need $\frac{n}{m} = \frac{\# \text{ equations}}{\# \text{ unknowns}} \gg 1$ for good quality estimation

WLS: Performance Analysis

- No statistical information (about V) needed to derive $\hat{\theta}_{WLS}$. However, in order to evaluate its performance (for the linear model), we need to introduce a stochastic context: V random with
$$\begin{cases} E V = 0 \\ E VV^T = C_{VV} \end{cases}$$
- note: $\hat{\theta}_{WLS} - \theta = (H^T W H)^{-1} H^T W (H \theta + V) - \theta = (H^T W H)^{-1} H^T W V$
- $E \hat{\theta}_{WLS} - \theta = (H^T W H)^{-1} H^T W E V = 0$: unbiased
- $C_{\tilde{\theta}\tilde{\theta}}(W) = C_{\hat{\theta}\hat{\theta}}(W) = (H^T W H)^{-1} H^T W C_{VV} W H (H^T W H)^{-1}$
- optimal weighting: $W = C_{VV}^{-1}$: $C_{\tilde{\theta}\tilde{\theta}}(W) \geq C_{\tilde{\theta}\tilde{\theta}}(C_{VV}^{-1}) = (H^T C_{VV}^{-1} H)^{-1}$
- Further statistical knowledge and optimality properties:

WLS = ML if $V \sim \mathcal{N}(0, W^{-1})$ and independent of θ



Rank Reduction in the Linear Model

- reparameterize in terms of a reduced set of parameters $\underbrace{\theta}_{m \times 1} = \underbrace{T}_{m \times r} \underbrace{\phi}_{r \times 1}$

- issue of optimal transformation T

- we shall limit analysis to $T = \begin{bmatrix} I_r \\ 0 \end{bmatrix} : \phi = \theta_{1:r} = \bar{\theta}_r$

$$S = H\theta = [\underline{H}_r \ \underline{H}_r] \begin{bmatrix} \bar{\theta}_r \\ \underline{\theta}_r \end{bmatrix} = \underline{H}_r \bar{\theta}_r + \underline{H}_r \underline{\theta}_r$$

- reduced-rank LS: $\widehat{\bar{\theta}}_r = \arg \min_{\bar{\theta}_r} \|Y - \underline{H}_r \bar{\theta}_r\|^2 = (\underline{H}_r^T \underline{H}_r)^{-1} \underline{H}_r^T Y$

$$\widehat{S} = \widehat{S}_r = \underline{H}_r \widehat{\bar{\theta}}_r = P_{\underline{H}_r} Y , \quad \widehat{V} = \widehat{V}_r = Y - \widehat{S}_r = P_{\underline{H}_r}^\perp Y$$

•

$$\begin{aligned} \widehat{\theta}_r &= (\underline{H}_r^T \underline{H}_r)^{-1} \underline{H}_r^T (\underline{H}_r \bar{\theta}_r + \underline{H}_r \underline{\theta}_r + V) \\ &= \bar{\theta}_r + (\underline{H}_r^T \underline{H}_r)^{-1} \underline{H}_r^T (\underline{H}_r \underline{\theta}_r + V) = \bar{\theta}_r - \widetilde{\bar{\theta}}_r \end{aligned}$$

$$\widehat{\theta} = \begin{bmatrix} \widehat{\bar{\theta}}_r \\ 0 \end{bmatrix} , \quad \widetilde{\theta} = \begin{bmatrix} \widetilde{\bar{\theta}}_r \\ \underline{\theta}_r \end{bmatrix}$$



Rank Reduction in the Linear Model (2)

- estimator bias and variance

$$b_{\tilde{\theta}}(\theta) = \begin{bmatrix} (\bar{H}_r^T \bar{H}_r)^{-1} \bar{H}_r^T \underline{H}_r \underline{\theta}_r \\ -\underline{\theta}_r \end{bmatrix}, \quad C_{\tilde{\theta}\tilde{\theta}} = \begin{bmatrix} C_{\tilde{\theta}_r \tilde{\theta}_r} & 0 \\ 0 & 0 \end{bmatrix}$$

$$C_{\tilde{\theta}_r \tilde{\theta}_r} = (\bar{H}_r^T \bar{H}_r)^{-1} \bar{H}_r^T C_{VV} \bar{H}_r (\bar{H}_r^T \bar{H}_r)^{-1}, \quad R_{\tilde{\theta}\tilde{\theta}} = C_{\tilde{\theta}\tilde{\theta}} + b_{\tilde{\theta}} b_{\tilde{\theta}}^T$$

- signal component

$$\tilde{S} = S - \hat{S}_r = \bar{H}_r \bar{\theta}_r + \underline{H}_r \underline{\theta}_r - (\bar{H}_r \bar{\theta}_r + P_{\bar{H}_r} \underline{H}_r \underline{\theta}_r + P_{\bar{H}_r} V) = P_{\bar{H}_r}^\perp \underline{H}_r \underline{\theta}_r - P_{\bar{H}_r} V$$

bias : $b_{\hat{S}_r}(\theta) = -E \tilde{S} = -P_{\bar{H}_r}^\perp \underline{H}_r \underline{\theta}_r \neq 0$: biased !

$$R_{\tilde{S}\tilde{S}} = C_{\tilde{S}\tilde{S}} + b_{\hat{S}_r \hat{S}_r} b_{\hat{S}_r \hat{S}_r}^T = P_{\bar{H}_r} C_{VV} P_{\bar{H}_r}^\perp (P_{\bar{H}_r}^\perp \underline{H}_r \underline{\theta}_r) (P_{\bar{H}_r}^\perp \underline{H}_r \underline{\theta}_r)^T$$

- noise component

$$\tilde{V} = V - \hat{V}_r = V - (P_{\bar{H}_r}^\perp \underline{H}_r \underline{\theta}_r + P_{\bar{H}_r}^\perp V) = -P_{\bar{H}_r}^\perp \underline{H}_r \underline{\theta}_r + P_{\bar{H}_r} V = -\tilde{S}$$

$$\text{SNR}_{\hat{V}_r} = \frac{E \|V\|^2}{E \|\tilde{V}\|^2} = \frac{n\sigma_v^2}{\|P_{\bar{H}_r}^\perp \underline{H}_r \underline{\theta}_r\|^2 + r\sigma_v^2}$$

Rank Reduction in FIR System Identification

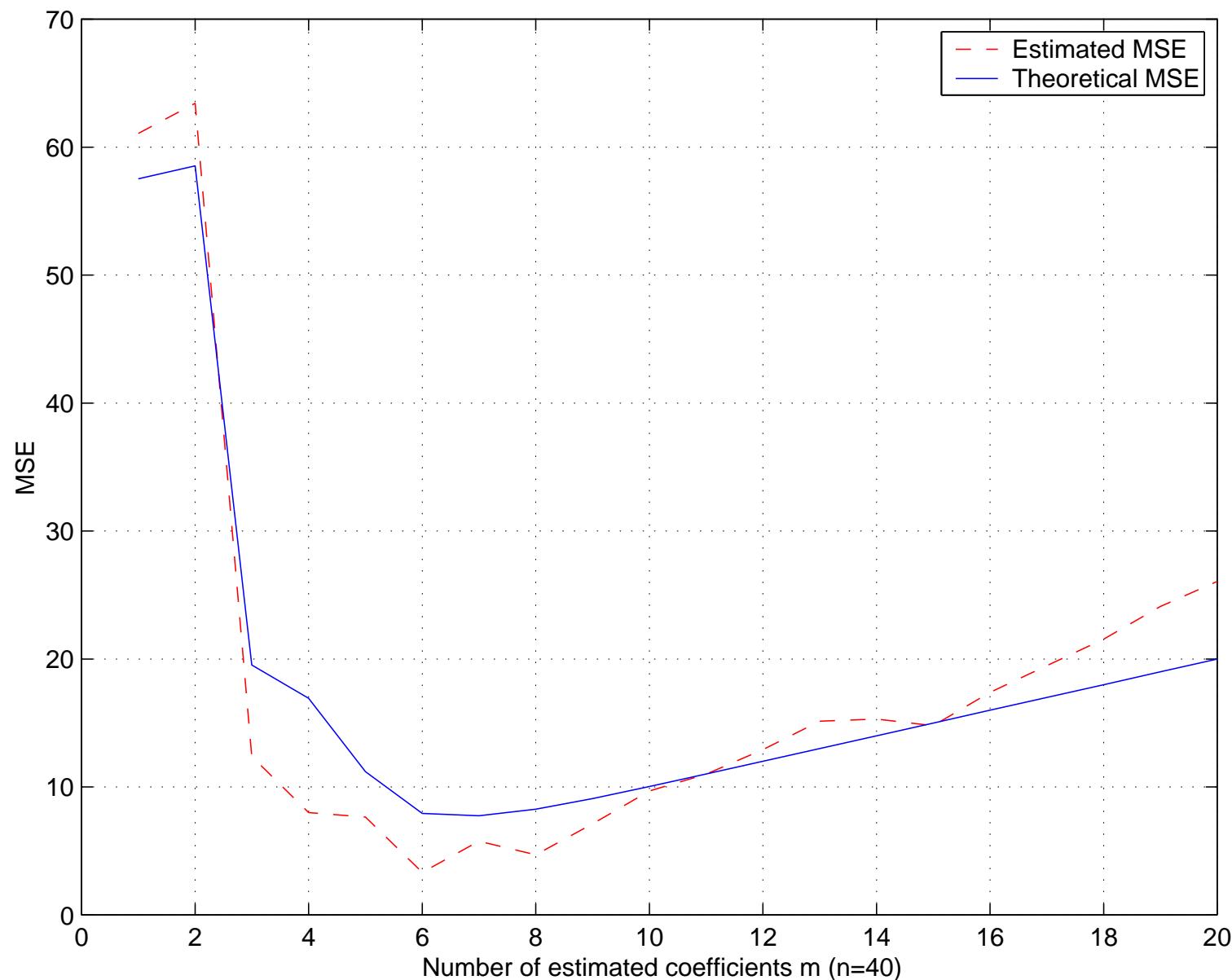
- Assume now H filled with samples of x_k , being white noise.
-

$$\begin{aligned}\text{SNR}_{\widehat{V}_r} &= \frac{n\sigma_v^2}{E_X \|P_{H_r^\perp} H_r \underline{\theta}_r\|^2 + r\sigma_v^2} = \frac{n\sigma_v^2}{|\text{bias}|^2 + r\sigma_v^2} \\ &= \frac{\frac{\sigma_x^2}{\sigma_v^2} \|\underline{\theta}_r\|^2 + \frac{r}{n}}{\text{SNR}_Y \frac{\|\underline{\theta}_r\|^2}{\|\underline{\theta}\|^2} + \frac{r}{n}}\end{aligned}$$

- to maximize $\text{SNR}_{\widehat{V}_r}$, need to minimize $|\text{bias}|^2 + r\sigma_v^2$
- we have $E \|\widehat{V}_m\|^2 = (n-m) \sigma_v^2$, $E \|\widehat{V}_r\|^2 = |\text{bias}|^2 + (n-r) \sigma_v^2$
- Hence can estimate

$$|\text{bias}|^2 + r\sigma_v^2 \approx \|\widehat{V}_r\|^2 - \|\widehat{V}_m\|^2 + (2r-m)\sigma_v^2 \approx \|\widehat{V}_r\|^2 - \|\widehat{V}_m\|^2 + \frac{2r-m}{n-m} \|\widehat{V}_m\|^2$$

Rank Reduction in FIR System Identification (2)



Choice of Estimator

- stochastic (Bayesian) information matrix:

$$J_{stoch} = J_{prior} + E_\theta J_{det}(\theta)$$

as $J_{det} \sim n$, J_{det} dominant as $n \gg 1$.

Hence if lots of data \Rightarrow prior of little relevance \Rightarrow deterministic estimation

If little data \Rightarrow need prior (even if invented) to regularize the problem, to avoid singularity of J_{det}

- Bayesian estimation:

- $\hat{\theta}_{MMSE}$ preferable
- $\hat{\theta}_{MAP}$ easier to calculate
- $\hat{\theta}_{LMMSE}$ simple, acceptable if everything \approx Gaussian (model \approx linear)

- deterministic (classical) estimation:

- Maximum Likelihood (ML) if possible
- if ML too complex or if a good initialization is required for an iterative optimization of ML: Least-Squares or Method of Moments
- Linear Gaussian model: all reasonable estimators identical



Statistical Signal Processing

Lecture 6

Chapter 2: Spectral Estimation

- review of spectral descriptions of stationary processes
- non-parametric spectral estimation
 - the periodogram
 - spectral leakage, spectral resolution, windowing
 - the averaged periodogram
 - the Blackman-Tukey spectral estimator (the smoothed periodogram)

Stationary Stochastic Processes

- discrete-time stochastic process = sequence of random variables
- description of distribution of stochastic processes often limited to first and second order moments
- for zero mean processes: second-order moments crucial
in the frequency domain: *power spectral density function* (psdf)
- for non-stationary processes: no ergodicity, no time-averaging. We shall see *time-frequency representations*.
- The random (complex vector) process $\{\mathbf{y}_k\}$ is *wide-sense stationary* (WSS) if its mean $(^H : \text{Hermitian (complex conjugate) transpose})$

$$E \mathbf{y}_n = m_{\mathbf{y}} \quad \text{sequence}$$

does not depend on n and its (matrix) *autocorrelation function* (acf)

$$r_{\mathbf{y}\mathbf{y}}(k) = E \mathbf{y}_{n+k} \mathbf{y}_n^H \quad \text{“acf at lag } k\text{”}$$

only depends on the time lag between the two samples (not on time). Corresponding central moment : *autocovariance function*

$$c_{\mathbf{y}\mathbf{y}}(k) = E (\mathbf{y}_{n+k} - m_{\mathbf{y}})(\mathbf{y}_n - m_{\mathbf{y}})^H = r_{\mathbf{y}\mathbf{y}}(k) - m_{\mathbf{y}} m_{\mathbf{y}}^H$$

Stationary Stochastic Processes (2)

- two jointly WSS random vector processes $\{\mathbf{x}_k\}$ and $\{\mathbf{y}_k\}$ have a *cross-correlation function* (ccf)

$$r_{\mathbf{xy}}(k) = E \mathbf{x}_{n+k} \mathbf{y}_n^H$$

and a *cross-covariance function*

$$c_{\mathbf{xy}}(k) = E (\mathbf{x}_{n+k} - m_{\mathbf{x}})(\mathbf{y}_n - m_{\mathbf{y}})^H = r_{\mathbf{xy}}(k) - m_{\mathbf{x}} m_{\mathbf{y}}^H$$

- Unless stated otherwise, we shall consider below processes with zero mean so that the autocorrelation function and the autocovariance function are equal. Nevertheless, the rest of this discussion holds for the general case.
- The following symmetry properties follow immediately from the definition:

$$r_{\mathbf{xy}}(k) = r_{\mathbf{yx}}^H(-k)$$

$$r_{\mathbf{yy}}(k) = r_{\mathbf{yy}}^H(-k) .$$

For real scalar processes

$$\begin{aligned} r_{xy}(k) &= r_{yx}(-k) \\ r_{yy}(k) &= r_{yy}(-k) . \end{aligned}$$

Stationary Stochastic Processes (3)

- In this transparency we consider real scalar processes.
- The following boundedness properties follow from the Cauchy-Schwarz inequality (with correlation as inner product):

$$\begin{aligned} r_{xx}(0) r_{yy}(0) &\geq |r_{xy}(k)|^2 \\ r_{yy}(0) &\geq |r_{yy}(k)| \geq 0 \end{aligned}$$

- The acf is furthermore a *positive semidefinite function*, meaning the following. Take any positive integer M . Let $\{a_j, j = 1, \dots, M\}$ be any set of M real numbers, not all zero, and $\{k_j \in \mathcal{Z}, j = 1, \dots, M\}$ any set of M distinct time instants. Then

$$0 \leq E \left| \sum_{j=1}^M a_j y_{k_j} \right|^2 = \sum_{i=1}^M \sum_{j=1}^M a_i a_j r_{yy}(k_i - k_j) = [a_1 \cdots a_M] T \begin{bmatrix} a_1 \\ \vdots \\ a_M \end{bmatrix}$$

This means also that the symmetric matrix T with elements $T_{ij} = r_{yy}(k_i - k_j)$ is positive semidefinite. T is Toeplitz if the k_j are equidistant.

Spectral Descriptions

- We can consider the z -transforms of the acf and the ccf

$$\begin{aligned} S_{yy}(z) &= \sum_{k=-\infty}^{\infty} r_{yy}(k) z^{-k} \\ S_{xy}(z) &= \sum_{k=-\infty}^{\infty} r_{xy}(k) z^{-k}. \end{aligned}$$

- *auto-psdf* $S_{yy}(f) = S_{yy}(e^{j2\pi f})$ and *cross-psdf* $S_{xy}(f) = S_{xy}(e^{j2\pi f})$

$$\begin{aligned} S_{yy}(f) &= \sum_{k=-\infty}^{\infty} r_{yy}(k) e^{-j2\pi fk} && \text{Wiener-Khinchin relation} \\ S_{xy}(f) &= \sum_{k=-\infty}^{\infty} r_{xy}(k) e^{-j2\pi fk}. \end{aligned}$$

auto-psdf and cross-psdf are in principle defined for scalar processes

- Because we normalized the sampling period to 1, the sampling frequency is also normalized to 1 and hence $S_{xy}(f)$ and $S_{yy}(f)$ are periodic with period 1.
- We see that estimating $S_{yy}(f)$ and estimating $r_{yy}(k)$ are equivalent since both are related by the Fourier transform, a one-to-one transformation.

Spectral Descriptions (2)

- A basic real scalar discrete-time random process is white noise with acf

$$r_{yy}(k) = \sigma_y^2 \delta_{k0}$$

where δ_{kn} is the Kronecker delta. This means that the white process is zero mean and all samples are uncorrelated. The psdf becomes

$$S_{yy}(f) = \sigma_y^2$$

which is constant.

- LTI filter transfer fn: $\mathbf{H}(f) = \sum_m \mathbf{h}_m e^{-j2\pi f m} = \mathbf{H}(e^{j2\pi f})$, $\mathbf{H}(z) = \sum_m \mathbf{h}_m z^{-m}$
- The filtering of a WSS process by a linear time-invariant (LTI) filter produces another WSS process. Indeed, let $\mathbf{y}_n = \sum_m \mathbf{h}_{n-m} \mathbf{x}_m$ where $\{\mathbf{x}_k\}$ is WSS. Then we get for the mean

$$E \mathbf{y}_n = \sum_m \mathbf{h}_m E \mathbf{x}_{n-m} = (\sum_m \mathbf{h}_m) m_{\mathbf{x}} = \mathbf{H}(0) m_{\mathbf{x}} = m_{\mathbf{y}}$$

which does indeed not depend on n . Furthermore,

$$\begin{aligned} E \mathbf{y}_{k+n} \mathbf{x}_n^H &= \sum_m \mathbf{h}_{k+n-m} E \mathbf{x}_m \mathbf{x}_n^H = \sum_m \mathbf{h}_{k+n-m} r_{\mathbf{xx}}(m-n) = \sum_m \mathbf{h}_{k-m} r_{\mathbf{xx}}(m) \\ &= \mathbf{h}_k * r_{\mathbf{xx}}(k) = r_{\mathbf{yx}}(k) \end{aligned}$$

Spectral Descriptions (3)

and similarly

$$\begin{aligned} E \mathbf{x}_{k+n} \mathbf{y}_n^H &= \sum_m (E \mathbf{x}_{k+n} \mathbf{x}_m^H) \mathbf{h}_{n-m}^H = \sum_m r_{\mathbf{xx}}(k+n-m) \mathbf{h}_{n-m}^H = \sum_m r_{\mathbf{xx}}(k-m) \mathbf{h}_{-m}^H \\ &= r_{\mathbf{xx}}(k) * \mathbf{h}_{-k}^H = r_{\mathbf{xy}}(k) \end{aligned}$$

$$\begin{aligned} E \mathbf{y}_{k+n} \mathbf{y}_n^H &= \sum_m \mathbf{h}_{k+n-m} \sum_l (E \mathbf{x}_m \mathbf{x}_l^H) \mathbf{h}_{n-l}^H = \sum_m \mathbf{h}_{k+n-m} \sum_l r_{\mathbf{xx}}(m-l) \mathbf{h}_{n-l}^H \\ &= \sum_m \mathbf{h}_{k-m} \sum_l r_{\mathbf{xx}}(m-l) \mathbf{h}_{-l}^H = \sum_m \mathbf{h}_{k-m} r_{\mathbf{xy}}(m) = \mathbf{h}_k * r_{\mathbf{xy}}(k) \\ &= \mathbf{h}_k * r_{\mathbf{xx}}(k) * \mathbf{h}_{-k}^H = r_{\mathbf{yy}}(k) \end{aligned}$$

in which the correlation considered is each time independent of time. This shows that not only $\{\mathbf{y}_k\}$ is WSS but furthermore $\{\mathbf{x}_k\}$ and $\{\mathbf{y}_k\}$ are jointly WSS.

- Taking z - and Fourier transforms of these convolution relations yields

$$\begin{array}{ll} S_{\mathbf{yx}}(z) = \mathbf{H}(z) S_{\mathbf{xx}}(z) & S_{\mathbf{yx}}(f) = \mathbf{H}(f) S_{\mathbf{xx}}(f) \\ S_{\mathbf{xy}}(z) = S_{\mathbf{xx}}(z) \mathbf{H}^\dagger(z) & S_{\mathbf{xy}}(f) = S_{\mathbf{xx}}(f) \mathbf{H}^H(f) \\ S_{\mathbf{yy}}(z) = \mathbf{H}(z) S_{\mathbf{xx}}(z) \mathbf{H}^\dagger(z) & S_{\mathbf{yy}}(f) = \mathbf{H}(f) S_{\mathbf{xx}}(f) \mathbf{H}^H(f) \end{array}$$

w. $\mathbf{H}(z) = \sum_k \mathbf{h}_k z^{-k}$ and for ^{scalar} real h_k : $S_{yy}(f) = |H(f)|^2 S_{xx}(f)$.

paraconjugate (matched filter): $\mathbf{H}^\dagger(z) = \mathbf{H}^H(1/z^*) = z$ -transform of \mathbf{h}_{-k}^H

Power Spectral Density Interpretation

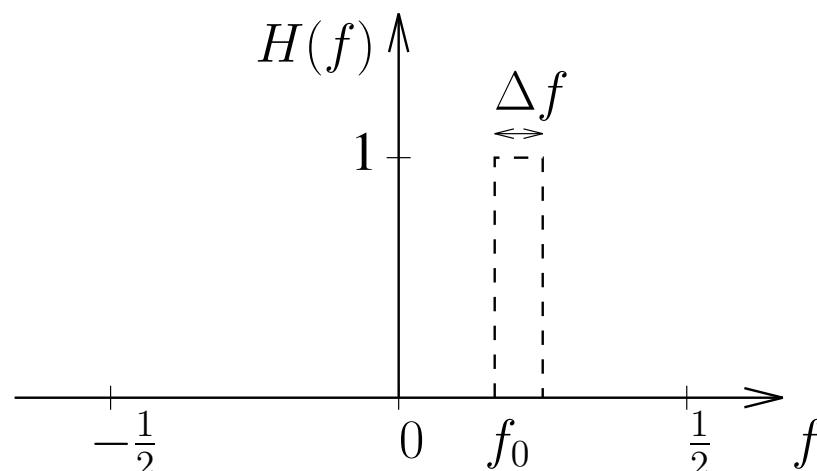
- The last relation leads to the interpretation of $S_{xx}(f)$ as power spectral density function. Indeed, on the one hand we find from the Fourier relationship

$$r_{xx}(0) = \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{xx}(f) df$$

which means that the total power $r_{xx}(0) = E x^2(k)$ is made up of the sum of the contributions at all frequencies. *real scalar process considered here*

- Now let $\{x_k\}$ and $\{y_k\}$ be the input-output pair associated with a LTI filter with transfer function

$$H(f) = \begin{cases} 1 & , f \in [f_0, f_0 + \Delta f] , \Delta f \text{ arbitrarily small} \\ 0 & , \text{elsewhere in } [-\frac{1}{2}, \frac{1}{2}] \end{cases}$$



Power Spectral Density Interpretation (2)

- Assume that $S_{xx}(f)$ is continuous at f_0 . We get

$$r_{yy}(0) = \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{yy}(f) df = \int_{-\frac{1}{2}}^{\frac{1}{2}} |H(f)|^2 S_{xx}(f) df = \int_{f_0}^{f_0 + \Delta f} S_{xx}(f) df = S_{xx}(f_0) \Delta f$$

or $S_{xx}(f_0) = r_{yy}(0)/\Delta f$ which first of all implies $S_{xx}(f) \geq 0$ and secondly justifies the name power spectral density of $S_{xx}(f)$.

- For the special case of $\{x_k\}$ being a white noise input, the psdf of the output process $\{y_k\}$ of a LTI filter with transfer function $H(f)$ becomes

$$S_{yy}(f) = \sigma_x^2 |H(f)|^2$$

- We have shown independently that the acf is a positive semidefinite function and that the psdf is nonnegative. It is possible to show that these two properties imply each other. Indeed, if $r_{xx}(k)$ and $S_{xx}(f)$ are two functions that form a Fourier transform pair, then the following theorem holds.

Theorem (Bochner's theorem) $r_{xx}(k)$ is real, even and positive (semi)definite iff $S_{xx}(f)$ is real, even and positive (nonnegative).

Fourier Transform Correlation

- the Fourier transform $\mathbf{Y}(f) = \sum_k \mathbf{y}_k e^{-j2\pi f k}$ of the stationary process $\{\mathbf{y}_k\}$ is random
- **Theorem (Fourier transform correlation)**

Let $\mathbf{X}(f) = \sum_{k=-\infty}^{\infty} \mathbf{x}_k e^{-j2\pi f k}$ and $\mathbf{Y}(f) = \sum_{k=-\infty}^{\infty} \mathbf{y}_k e^{-j2\pi f k}$. Then

$$E \mathbf{X}(f) \mathbf{Y}^H(f_1) = S_{\mathbf{xy}}(f) \delta_1(f - f_1)$$

where we have introduced the impulse train $\delta_{f_0}(f) = \sum_{n=-\infty}^{\infty} \delta(f - n f_0)$.

- In particular, $E Y(f) Y^*(f_1) = S_{yy}(f) \delta_1(f - f_1)$ which means that a real scalar random process $y_k = \int_{-\frac{1}{2}}^{\frac{1}{2}} Y(f) e^{j2\pi f k} df$ can be regarded as a superposition of exponentials $e^{j2\pi f k}$ with random complex amplitudes $Y(f)$ that are uncorrelated at different frequencies and that have a power proportional to $S_{yy}(f)$.

Filtering Relationships

- Using the above result, it is straightforward to show the effect on the auto- or cross-psdf of linear filtering. Indeed consider the processes \mathbf{u}_k , \mathbf{v}_k , \mathbf{x}_k and \mathbf{y}_k which are related by linear filtering as shown in the figure. Then we get

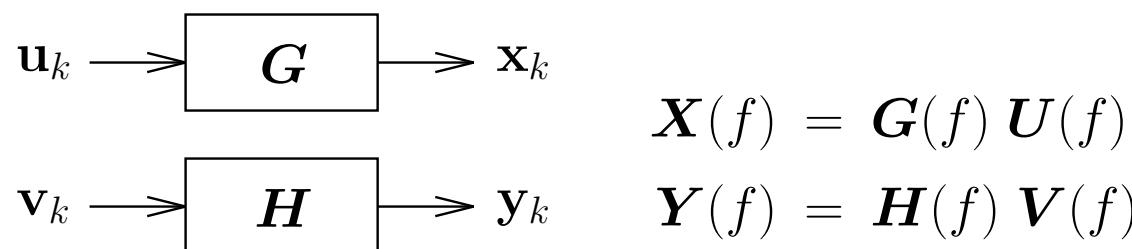
$$\begin{aligned} S_{\mathbf{xy}}(f)\delta_1(f-f_1) &= E \mathbf{X}(f)\mathbf{Y}^H(f_1) = \mathbf{G}(f) \left(E \mathbf{U}(f)\mathbf{V}^H(f_1) \right) \mathbf{H}^H(f_1) \\ &= \mathbf{G}(f)S_{\mathbf{uv}}(f)\mathbf{H}^H(f_1)\delta_1(f-f_1). \end{aligned}$$

Integrating both sides, $\int_{-\frac{1}{2}}^{\frac{1}{2}} df_1$, yields

$$S_{\mathbf{xy}}(f) = \mathbf{G}(f)S_{\mathbf{uv}}(f)\mathbf{H}^H(f).$$

- We can extend this property to the z -transforms:

$$S_{\mathbf{xy}}(z) = \mathbf{G}(z)S_{\mathbf{uv}}(z)\mathbf{H}^\dagger(z)$$



Why Spectral Estimation

- The spectrum transmitted by modems needs to satisfy certain restrictions, especially if the modem needs to comply with a certain standard. For instance voiceband modems over the telephone line need to be restricted to the [300Hz,3400Hz] band. Wireless modems or any radio transmitter needs to occupy a restricted bandwidth since the radio medium is scarce. In these considerations, not only the (effective) bandwidth of the signal is of importance but also the spectral roll-off. Typically, the emitted spectrum needs to fall under a certain spectral mask (plot of the maximum allowed power spectral density as a function of frequency). To verify this, the spectrum of the emitted signal needs to be estimated.
- We shall see that the optimization of the parameters of most source coding techniques depends on the spectrum or equivalently the correlation structure of the source to be coded. Hence, spectral estimation is again required. Such sources can be speech, audio, images or video.
- Spectrum estimation also allows to evaluate certain parameters about transmitted signals such as the frequency offset and the frequency-selective distortion introduced by the channel, or the spectrum of the interfering signals or noise.

Spectral Estimation

- There exist two big methodologies for estimating the psdf of a stationary process: *non-parametric* techniques and *parametric* techniques.
- In non-parametric or *classical* spectral estimation techniques, no constraints are imposed on the possible form of $S_{yy}(f)$. This implies that an infinite number of degrees of freedom need to be estimated: the $r_{yy}(k)$. This will lead to a very high estimation variance if we have to work with a finite amount of data.
- In the parametric or *high-resolution* techniques, a parametric model is assumed for $S_{yy}(f)$ and the spectral estimation problem reduces to the estimation of the parameters describing the parametric model. Since the parametric form can only perfectly model a limited class of functions, there will be some bias in the estimate of $S_{yy}(f)$. However, to compensate for this bias, a (large) reduction in variance is achievable compared to the non-parametric techniques.

Non-Parametric Spectral Estimation

- The first non-parametric technique is based on the Fourier transform. Consider the Fourier transform $Y(f)$ of the WSS process y_k with zero mean. $Y(f)$ is a random function with mean

$$E Y(f) = \sum_{k=-\infty}^{\infty} e^{-j2\pi f k} \underbrace{E y_k}_{} = 0 .$$

The correlation between the Fourier transform at frequencies f and f_1 is

$$E Y(f) Y^*(f_1) = S_{yy}(f) \delta_1(f - f_1) .$$

So the Fourier transform at different frequencies is uncorrelated, while its magnitude squared is proportional to the power spectral density function.

- In practice, we are given only a finite number of N samples $\{y_0, y_1, \dots, y_{N-1}\}$. The *periodogram* was introduced by Schuster in 1898 and is defined as the scaled magnitude squared of the Fourier transform of this finite number of samples:

$$\widehat{S}_{yy}(f) = \widehat{S}_{PER}(f) = \frac{1}{N} \left| \sum_{n=0}^{N-1} y_n e^{-j2\pi f n} \right|^2$$

where $\frac{1}{N}$ has been introduced to avoid that things $\rightarrow \infty$ as $N \rightarrow \infty$.

The Periodogram

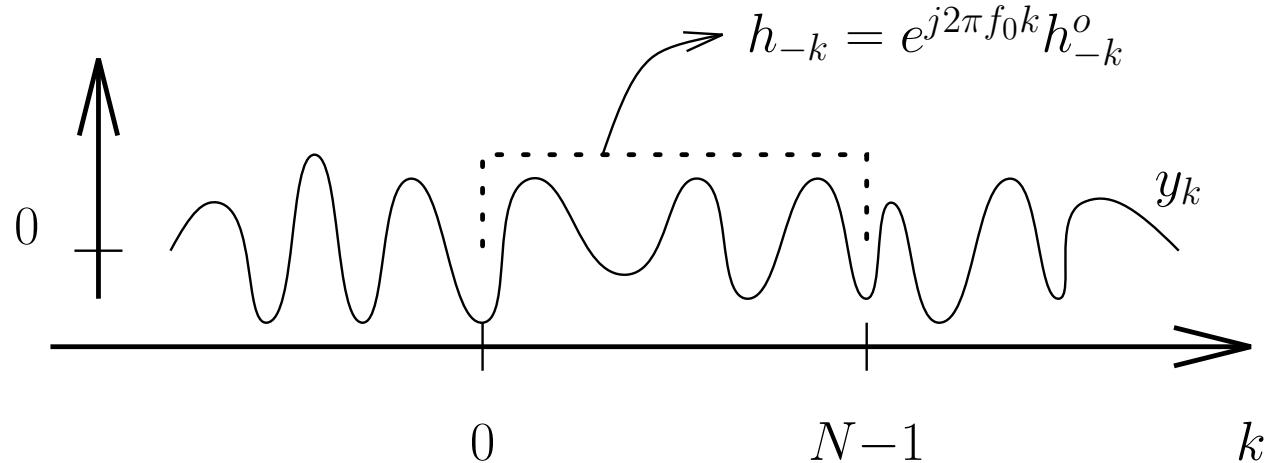
- focus on a particular frequency: denote f as f_0 . Interpretation:

$$\widehat{S}_{PER}(f_0) = \frac{1}{N} \left| \sum_{n=0}^{N-1} y_n e^{-j2\pi f_0 n} \right|^2 = N \left| \sum_{n=0}^{N-1} h_{k-n}^o y_n \right|_{k=0}^2$$

where $h_k^o = h_k e^{j2\pi f_0 k}$ and

$$h_k = \begin{cases} \frac{1}{N}, & k = -(N-1), \dots, -1, 0 \\ 0, & \text{otherwise} \end{cases}$$

h_k^o = anticausal FIR filter = a modulated rectangular window h_k



The Periodogram (2)

- The frequency response of the filter h_k is

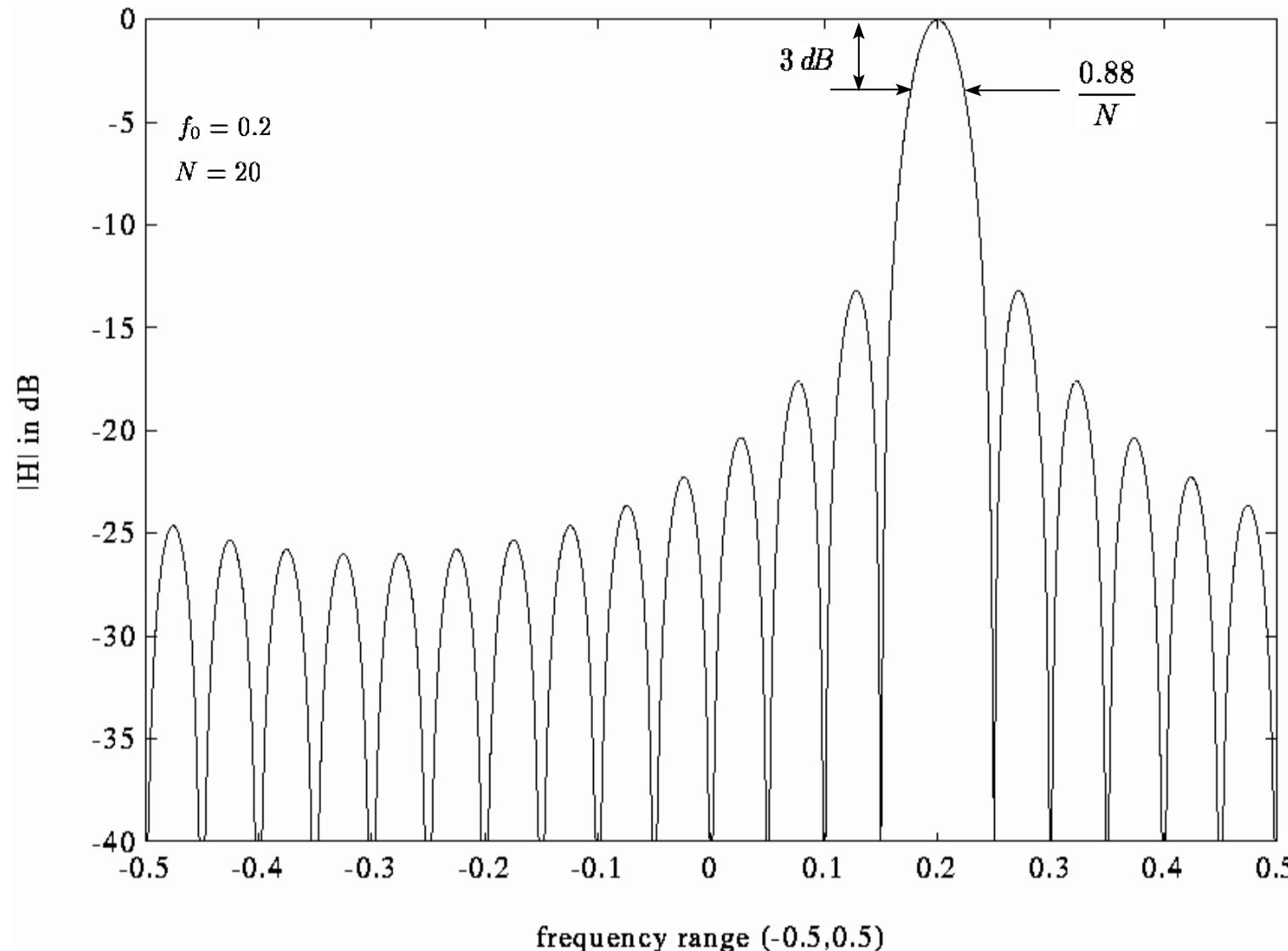
$$\begin{aligned} H(f) &= \sum_{k=-\infty}^{\infty} h_k e^{-j2\pi f k} = \frac{1}{N} \sum_{k=-(N-1)}^{0} e^{-j2\pi f k} = \frac{1}{N} \sum_{k=0}^{N-1} e^{j2\pi f k} \\ &= \frac{1}{N} \frac{1 - e^{j2\pi f N}}{1 - e^{j2\pi f}} = \frac{1}{N} \frac{e^{-j\pi f N} - e^{j\pi f N}}{e^{-j\pi f} - e^{j\pi f}} \frac{e^{j\pi f N}}{e^{j\pi f}} = \frac{\sin N\pi f}{N \sin \pi f} e^{j(N-1)\pi f} \end{aligned}$$

- $H(f)$ is of course periodic with period 1. It satisfies furthermore

$$\begin{aligned} H(k) &= 1 \quad , \quad k \in \mathcal{Z} \\ H(\frac{k}{N}) &= 0 \quad , \quad k \in \mathcal{Z} \setminus N\mathcal{Z} \quad . \end{aligned}$$

- $H^o(f) = H(f-f_0)$ is the frequency response of a *bandpass filter* with center frequency f_0 . The 3dB bandwidth is $\frac{0.88}{N} \Rightarrow$ bandwidth $\approx \frac{1}{N}$.
- Hence the periodogram estimates the power in $\{y_k\}$ at the frequency f_0 by filtering the data with a bandpass filter, sampling the output at time $k = 0$, and computing the magnitude squared. When multiplied by N ($= \frac{1}{\Delta f} = \frac{1}{1/N}$) (to account for the bandwidth of the bandpass filter), this yields the power spectral density estimate $\widehat{S}_{PER}(f_0)$ (compare to psdf interpretation).

The Periodogram (3)



The Periodogram (4)

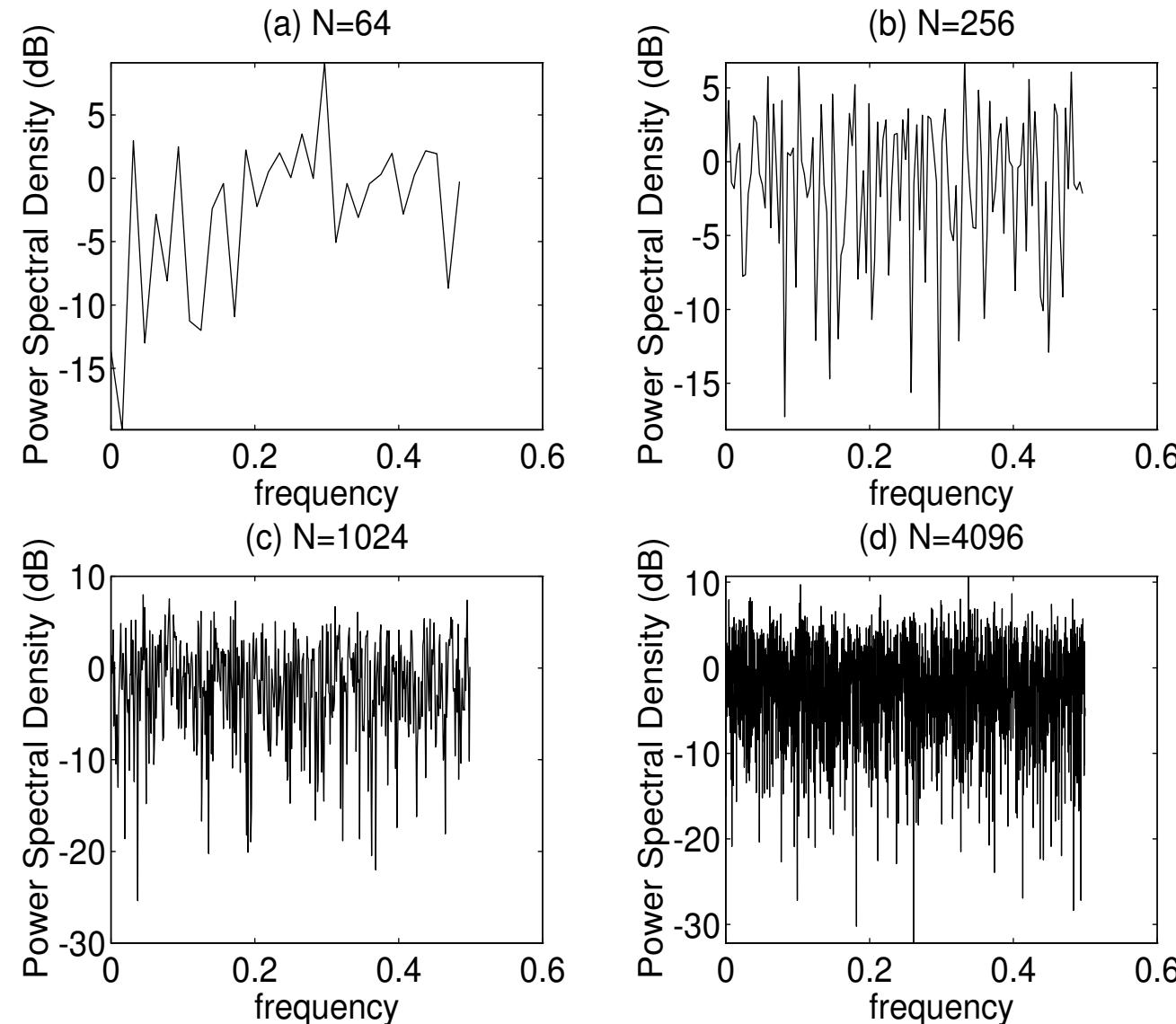
- If $N \rightarrow \infty$, then $\widehat{S}_{PER}(f_0) \rightarrow S_{yy}(f_0)$?
Is the Periodogram a consistent estimator of the psdf?
- White noise example: it appears that the random fluctuations or variance of the periodogram does not decrease with N , \Rightarrow the periodogram is *not* a consistent estimator of the psdf. Nevertheless, for this white noise example, the periodogram appears to fluctuate around a constant value (the true constant value of the psdf), \Rightarrow periodogram unbiased for white noise. But the variance does not tend to zero as $N \rightarrow \infty$.
- Intuitively, in order to have a consistent estimator of a set of parameters, we need to have lots more data than the number of parameters to be estimated. The number of parameters is ∞ here: Wiener-Khinchin relation

$$S_{yy}(f) = r_{yy}(0) + 2 \sum_{n=1}^{\infty} r_{yy}(n) \cos(2\pi f n)$$

$\{r_{yy}(k), k \geq 0\}$ = infinite set of parameters parameterizing $S_{yy}(f)$. Even if $N \rightarrow \infty$, $N \gg$ the number of parameters is impossible. Hence, the variance in estimating those parameters cannot go to zero.

The Periodogram (5)

Illustrating the periodogram inconsistency for white Gaussian noise ($\sigma_y^2 = 1$)
 $10 \log_{10} \widehat{S}_{PER}(f)$ for $N = 64$ (a), 256 (b), 1024 (c), 4096 (d).



Periodogram Mean

- We get for the convolution of h_k^o and y_k (inverse Fourier T of its FT)

$$\sum_n h_{k-n}^o y_n = h_k^o * y_k = \mathcal{F}^{-1} \{ H^o(f) Y(f) \}$$

where h_k^o and $H^o(f) = H(f-f_0)$ depend on f_0 . Hence

$$\left[\sum_{n=0}^{N-1} h_{k-n}^o y_n \right]_{k=0} = \left[\int_{-\frac{1}{2}}^{\frac{1}{2}} H^o(f) Y(f) e^{j2\pi f k} df \right]_{k=0} = \int_{-\frac{1}{2}}^{\frac{1}{2}} H^o(f) Y(f) df .$$

- So we get for the mean of the periodogram

$$\begin{aligned}
 E \widehat{S}_{PER}(f_0) &= N E \left| \left[\sum_{n=0}^{N-1} h_{k-n}^o y_n \right]_{k=0} \right|^2 = N E \left| \int_{-\frac{1}{2}}^{\frac{1}{2}} H^o(f) Y(f) df \right|^2 \\
 &= N E \left(\int_{-\frac{1}{2}}^{\frac{1}{2}} H^o(f) Y(f) df \right) \left(\int_{-\frac{1}{2}}^{\frac{1}{2}} H^{o*}(f_1) Y^*(f_1) df_1 \right) \\
 &= N \int_{-\frac{1}{2}}^{\frac{1}{2}} df H^o(f) \int_{-\frac{1}{2}}^{\frac{1}{2}} df_1 H^{o*}(f_1) \underbrace{E Y(f) Y^*(f_1)}_{= S_{yy}(f) \delta_1(f-f_1)} \\
 &= N \int_{-\frac{1}{2}}^{\frac{1}{2}} df H^o(f) S_{yy}(f) \underbrace{\int_{-\frac{1}{2}}^{\frac{1}{2}} df_1 H^{o*}(f_1) \delta_1(f-f_1)}_{= H^{o*}(f)} = N \int_{-\frac{1}{2}}^{\frac{1}{2}} |H^o(f)|^2 S_{yy}(f) df \\
 &= N \int_{-\frac{1}{2}}^{\frac{1}{2}} |H(f-f_0)|^2 S_{yy}(f) df = \int_{-\frac{1}{2}}^{\frac{1}{2}} \underbrace{N |H(f_0-f)|^2}_{= W_B(f_0-f)} S_{yy}(f) df
 \end{aligned}$$

Periodogram Mean (2)

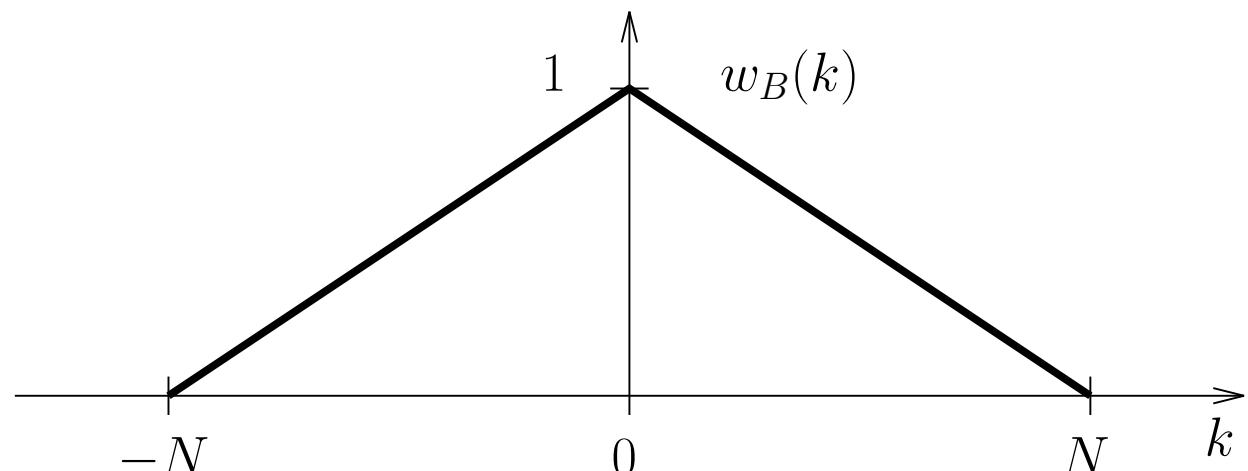
- Hence, $E \widehat{S}_{PER}(f_0) = \int_{-\frac{1}{2}}^{\frac{1}{2}} W_B(f_0 - f) S_{yy}(f) df = W_B(f_0) * S_{yy}(f_0)$
where

$$W_B(f) = \frac{1}{N} \left(\frac{\sin \pi f N}{\sin \pi f} \right)^2 = N |H(f)|^2.$$

- Since $W_B(f)$ is (apart from the scaling factor N) the square of the Fourier transform of a rectangular window, it is also the Fourier transform of the convolution of a rectangular window with itself which is a triangular window. Hence

$$w_B(k) = \mathcal{F}^{-1}\{W_B(f)\} = \begin{cases} 1 - \frac{|k|}{N}, & |k| \leq N-1 \\ 0, & |k| \geq N. \end{cases}$$

Such a window is called a *Bartlett window*.



Periodogram Mean (3)

- We conclude that the average periodogram is the convolution of the true psdf with the Fourier transform of the Bartlett window, yielding on the average a smoothed version of the psdf. Now since in general $W_B(f) * S_{yy}(f) \neq S_{yy}(f)$, the periodogram is biased in general for finite data records.
- However, the periodogram is asymptotically unbiased. Indeed

$$\begin{aligned}\lim_{N \rightarrow \infty} E \widehat{S}_{PER}(f) &= \lim_{N \rightarrow \infty} W_B(f) * S_{yy}(f) = \lim_{N \rightarrow \infty} \mathcal{F} \{ w_B(k) r_{yy}(k) \} \\ &= \mathcal{F} \left\{ \underbrace{\lim_{N \rightarrow \infty} w_B(k)}_{=1} r_{yy}(k) \right\} = \mathcal{F} \{ r_{yy}(k) \} = S_{yy}(f) .\end{aligned}$$

- Note that

$$\begin{aligned}w_B(0) &= \int_{-\frac{1}{2}}^{\frac{1}{2}} W_B(f) df = 1 , \quad W_B(0) = N , \quad W_B(f) \geq 0 , \\ 6dB \text{ bandwidth of } W_B(f) &= \frac{0.88}{N} .\end{aligned}$$

As $N \rightarrow \infty$, $W_B(f)$ becomes a narrow and high peak of constant surface. So $W_B(f) \rightarrow \delta(f)$. For the special case of white noise, the periodogram is unbiased even for finite data records.

Periodogram Covariance

- As far as the covariance of the periodogram is concerned, the following can be shown exactly for white Gaussian noise. The same result holds true approximately for more general processes when the data record is large.

$$Cov [\widehat{S}_{PER}(f_1), \widehat{S}_{PER}(f_2)] \approx S_{yy}(f_1)S_{yy}(f_2) [W_B(f_1 + f_2) + W_B(f_1 - f_2)] \frac{1}{N}$$

- The variance at the frequency f then follows as

$$Var [\widehat{S}_{PER}(f)] = Cov [\widehat{S}_{PER}(f), \widehat{S}_{PER}(f)] \approx S_{yy}^2(f) \left[1 + \frac{W_B(2f)}{N} \right] \geq S_{yy}^2(f).$$

- For frequencies not near 0 or $\pm\frac{1}{2}$, we can further approximate as

$$Var [\widehat{S}_{PER}(f)] \approx S_{yy}^2(f)$$

which is independent of the record length N !

- Furthermore, modulo certain conditions, the following result holds exactly

$$\lim_{N \rightarrow \infty} Var [\widehat{S}_{PER}(f)] = \begin{cases} 2S_{yy}^2(f) , & f = 0, \frac{1}{2} \\ S_{yy}^2(f) , & f \in (0, \frac{1}{2}) \end{cases}.$$

Periodogram Covariance (2)

- Hence the periodogram is an unreliable estimator since its standard deviation is approximately as large as the (nonnegative) quantity to be estimated.
- We get also

$$\text{Cov} [\widehat{S}_{PER}(f_1), \widehat{S}_{PER}(f_2)] \approx 0 \text{ if } f_1 \neq f_2 \text{ are integer multiples of } \frac{1}{N}.$$

The values of the periodogram at integer multiples of $\frac{1}{N}$ are uncorrelated.

- Coupled with the constant variance (as a function of N), this implies that as N increases the periodogram fluctuates rapidly as illustrated for white noise.
- In fact, it can also be shown that

$$\lim_{N \rightarrow \infty} \text{Cov} [\widehat{S}_{PER}(f_1), \widehat{S}_{PER}(f_2)] = 0, \quad f_1 \neq f_2$$

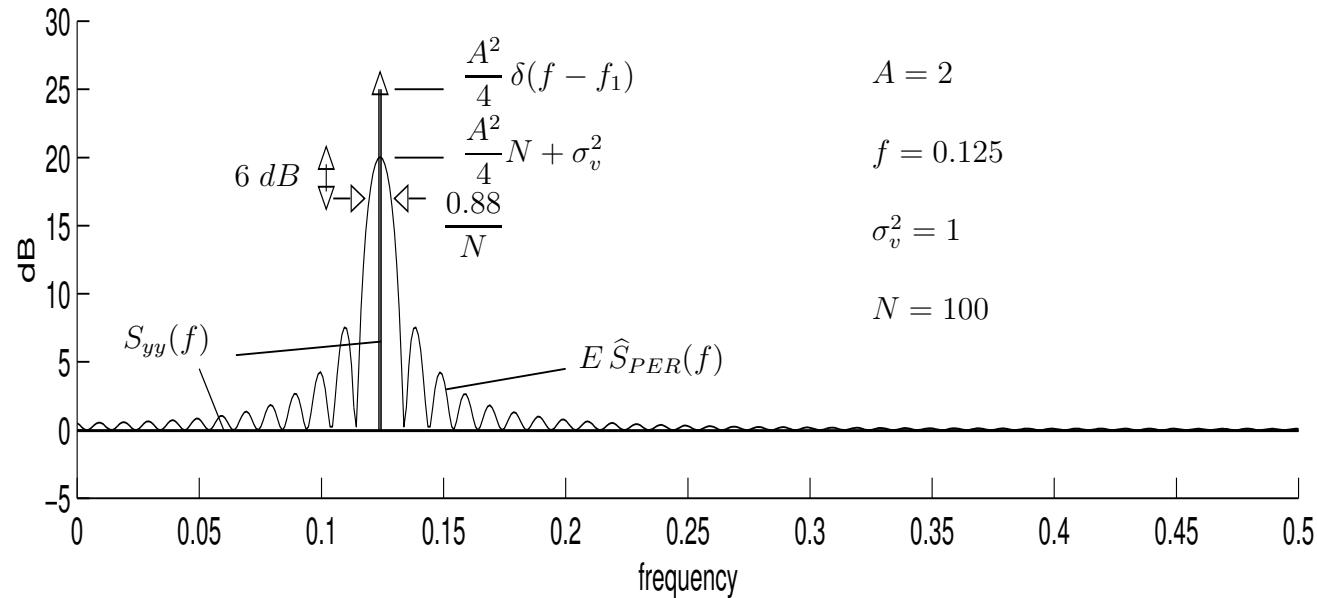
so that asymptotically, the periodogram is uncorrelated at different frequencies, just like the Fourier transform of the whole realization y_k .

Spectral Leakage

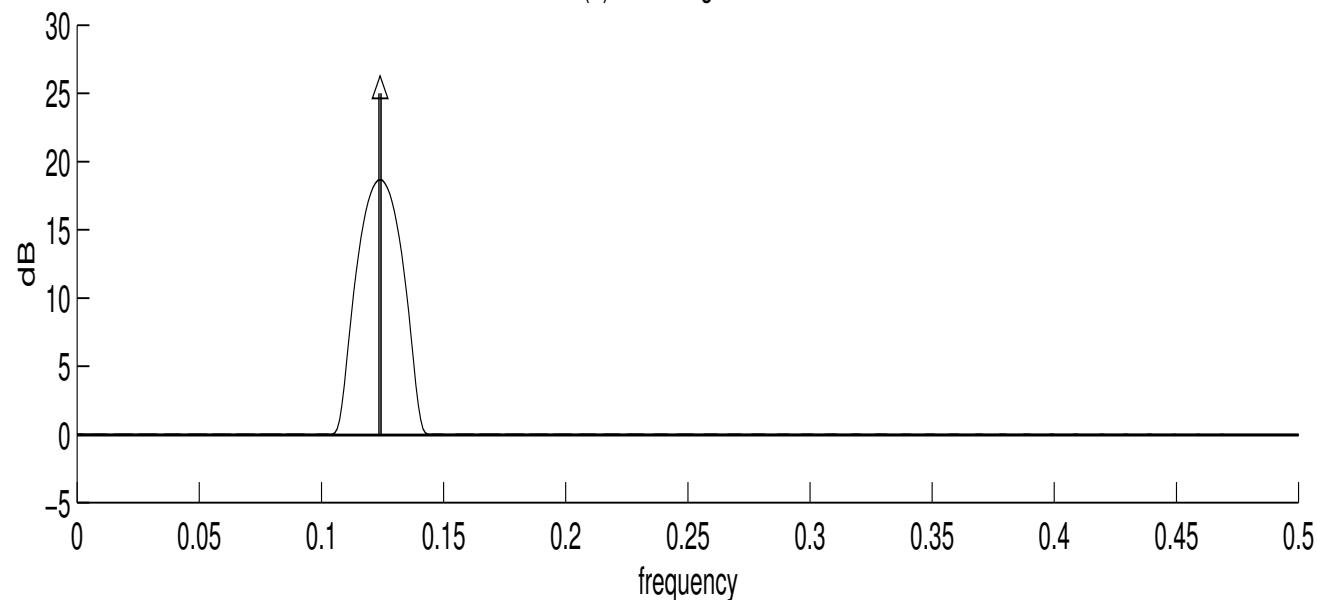
- If the process y_k under consideration would be a (complex) sinusoid with frequency f_0 , $y_k = e^{j(2\pi f_0 k + \phi)}$ with ϕ uniformly distributed over $[0, 2\pi)$, then $r_{yy}(n) = e^{j2\pi f_0 n}$, $S_{yy}(f) = \delta(f - f_0)$ and the periodogram mean would be $W_B(f - f_0) = N |H(f - f_0)|^2$.
- strong sidelobes \Rightarrow the average periodogram has non-zero contributions at frequencies where the psdf has no contributions: *spectral leakage*.
- to reduce this spectral leakage, replace the rectangular window with another window that shows less discontinuities near the edges and hence that has weaker sidelobes.
- The price to pay is that the bandwidth of the main lobe increases, reducing the *resolution*. For maximum resolution, or the ability to observe two comparable level sinusoids, no data windowing should be used. Considering the 3 dB bandwidth of $|H(f)|$, a common rule of thumb is that two equiamplitude sinusoids are resolvable if their normalized frequencies are spaced more than $1/N$ apart. If the data consists of only one sinusoid (or several sinusoids spaced much more than $1/N$ apart) embedded in white Gaussian noise, then the optimal (ML) frequency estimator is the periodogram without data windowing.

Spectral Leakage (2)

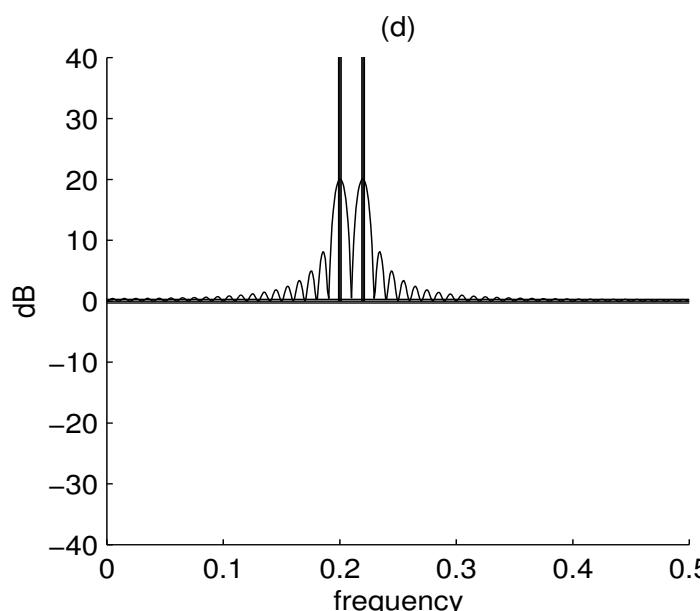
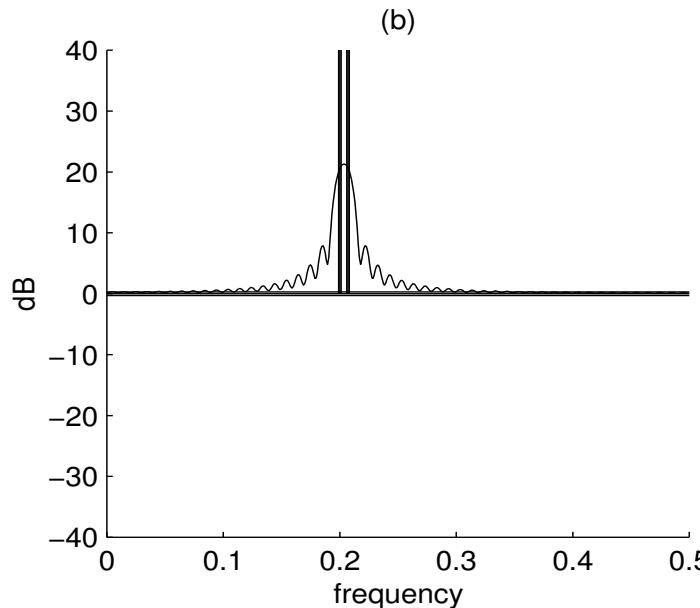
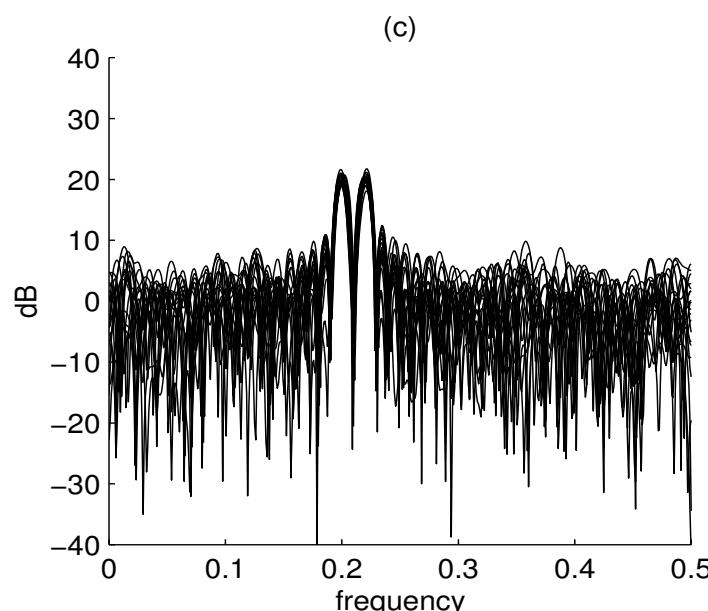
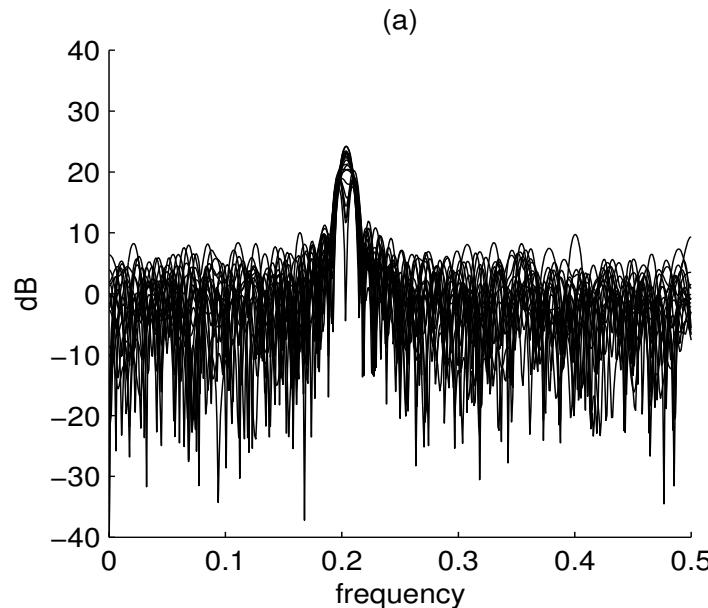
(a) Rectangular window



(b) Hamming window



Illustrating Spectral Resolution



(a), (b) :
 $(f_1, f_2) = (0.2, 0.207)$

(c), (d) :
 $(f_1, f_2) = (0.2, 0.22)$

$N = 100$
 $A_1 = A_2 = 2$
 $\sigma_v^2 = 1$
rectang. wind.

Data Weighting

- Since the windows used in practice are symmetric w.r.t. the center of the data record, we shall assume for this windowing discussion that $N = 2M+1$ and that the available data are y_{-M}, \dots, y_M . So we apply a symmetric window $w_k = w_{-k}$ before computing the periodogram. The periodogram with weighting becomes:

$$\widehat{S}_{PER,w}(f) = \frac{1}{2M+1} \left| \sum_{k=-M}^M w_k y_k e^{-j2\pi f k} \right|^2.$$

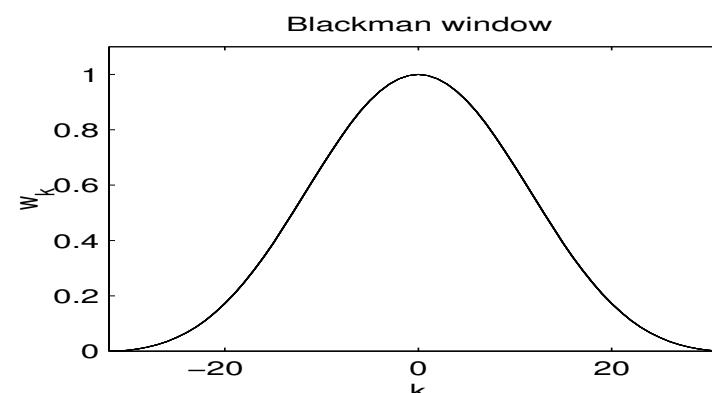
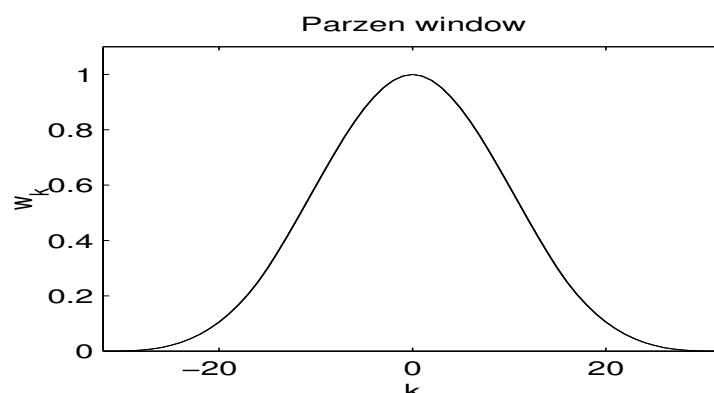
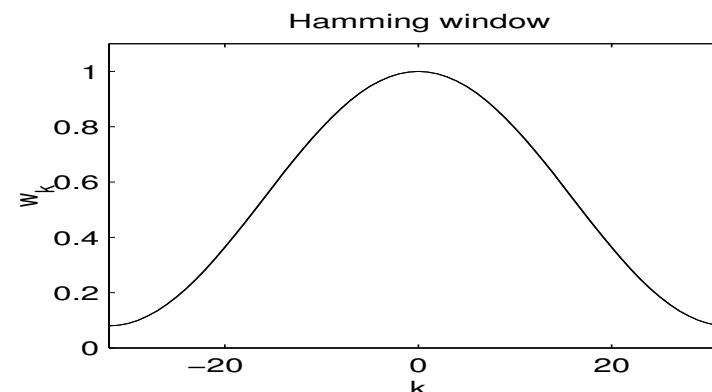
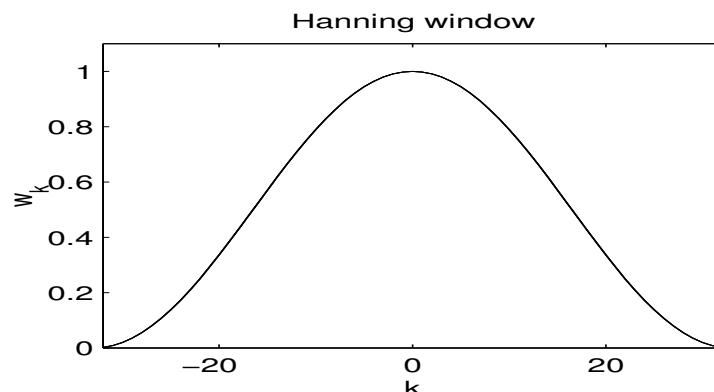
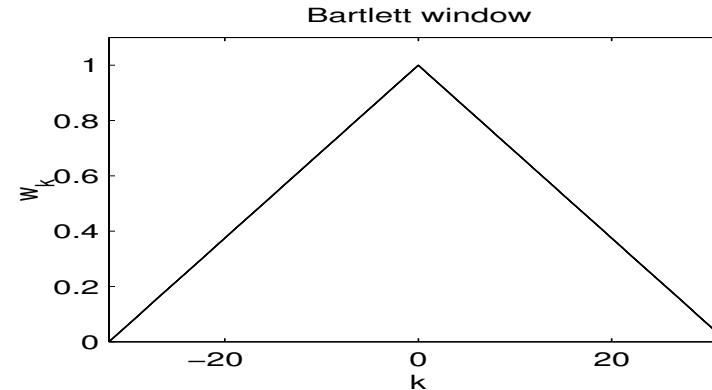
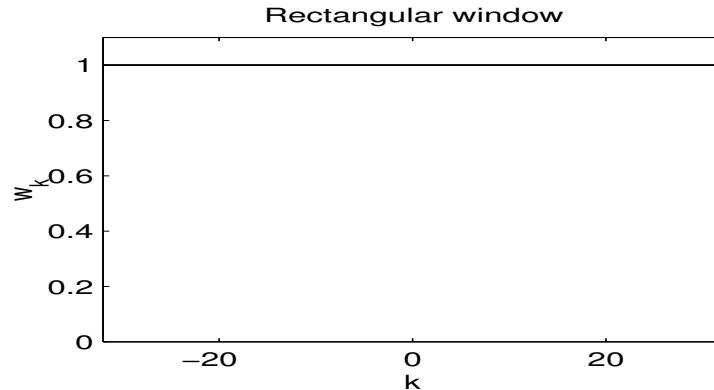
Some popular windows are (all with $w_k = 0$, $|k| > M$):

Name	Definition: $w_k =$	Fourier transform: $W(f) =$
Rectangular	1	$W_R(f) = \frac{\sin \pi f(2M+1)}{\sin \pi f}$
Bartlett	$1 - \frac{ k }{M}$	$W_B(f) = \frac{1}{M} \left(\frac{\sin \pi f M}{\sin \pi f} \right)^2$
Hanning	$\frac{1}{2} + \frac{1}{2} \cos \frac{\pi k}{M}$	$\frac{1}{4} W_R(f - \frac{1}{2M}) + \frac{1}{2} W_R(f) + \frac{1}{4} W_R(f + \frac{1}{2M})$
Hamming	$0.54 + 0.46 \cos \frac{\pi k}{M}$	$0.23 W_R(f - \frac{1}{2M}) + 0.54 W_R(f) + 0.23 W_R(f + \frac{1}{2M})$
Parzen (M even)	$\begin{cases} 2(1 - \frac{ k }{M})^3 - (1 - 2\frac{ k }{M})^3, & k \leq \frac{M}{2} \\ 2(1 - \frac{ k }{M})^3, & \frac{M}{2} < k \leq M \end{cases}$	$\frac{8}{M^3} \left(\frac{3}{2} \frac{\sin^4 \pi f M / 2}{\sin^4 \pi f} - \frac{\sin^4 \pi f M / 2}{\sin^2 \pi f} \right)$

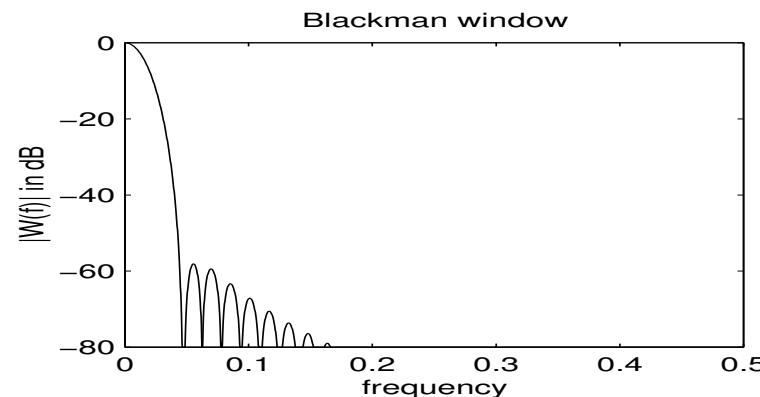
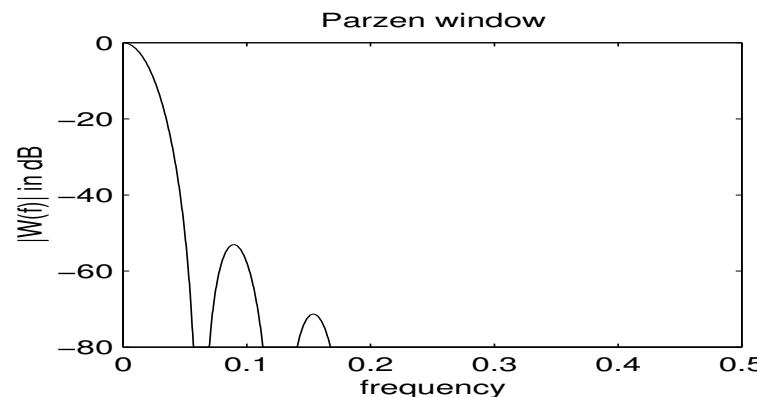
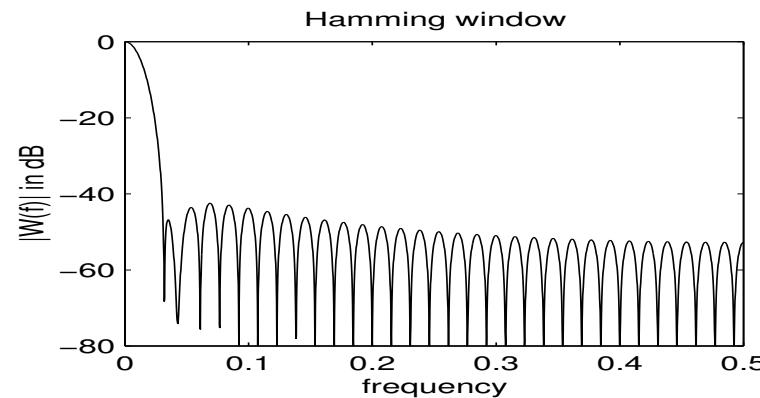
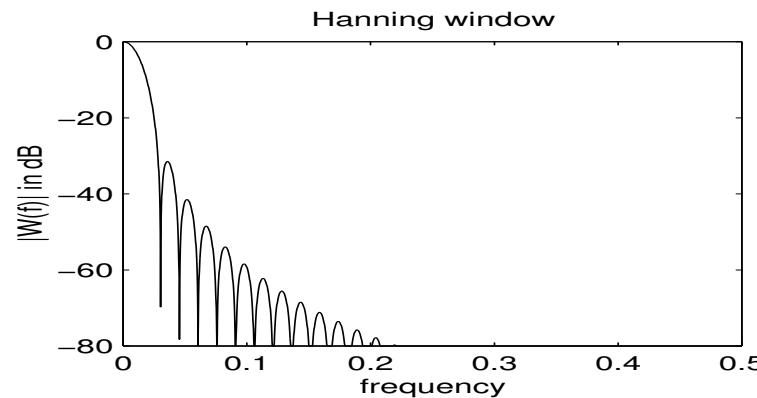
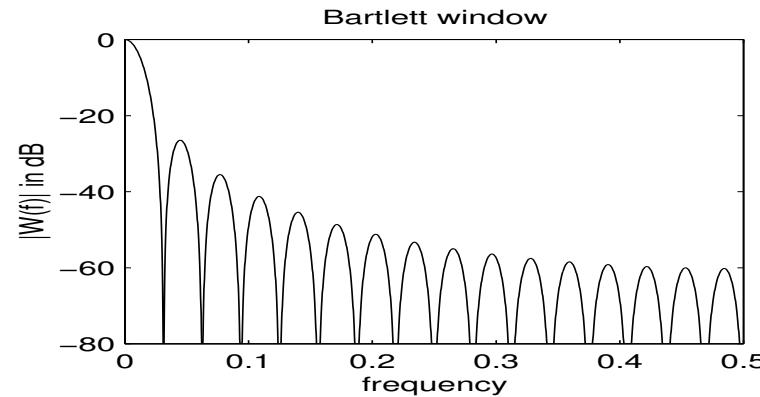
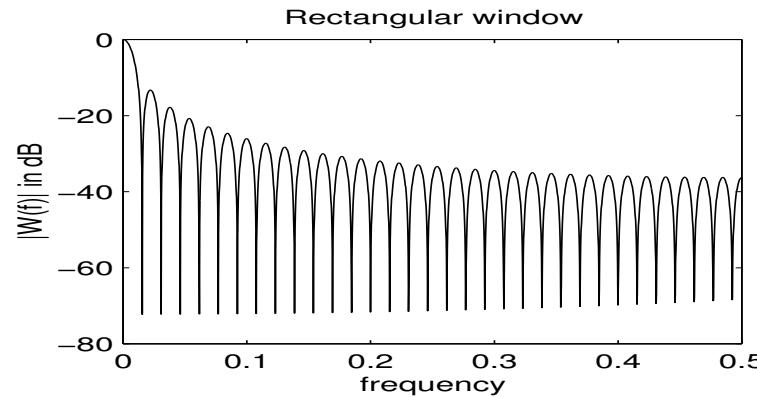
Data Windowing

Name	Definition: $w_k =$	Fourier transform: $W(f) =$	\mathcal{B}_{3dB}	A_{SL}
Rectangular	1	$W_R(f) = \frac{\sin \pi f(2M+1)}{\sin \pi f}$	$\frac{0.88}{N}$	-13
Bartlett	$1 - \frac{ k }{M}$	$W_B(f) = \frac{1}{M} \left(\frac{\sin \pi f M}{\sin \pi f} \right)^2$	$\frac{1.28}{N}$	-27
Hanning	$\frac{1}{2} + \frac{1}{2} \cos \frac{\pi k}{M}$	$\frac{1}{2} W_R(f) + \frac{1}{4} (W_R(f - \frac{1}{2M}) + W_R(f + \frac{1}{2M}))$	$\frac{1.44}{N}$	-32
Hamming	$0.54 + 0.46 \cos \frac{\pi k}{M}$	$0.54 W_R(f) + 0.23 (W_R(f - \frac{1}{2M}) + W_R(f + \frac{1}{2M}))$	$\frac{1.30}{N}$	-43
Parzen (M even)	$\begin{cases} 2(1 - \frac{ k }{M})^3 - (1 - 2\frac{ k }{M})^3, & k \leq \frac{M}{2} \\ 2(1 - \frac{ k }{M})^3, & \frac{M}{2} < k \leq M \end{cases}$	$\frac{8}{M^3} \left(\frac{3}{2} \frac{\sin^4 \pi f M / 2}{\sin^4 \pi f} - \frac{\sin^4 \pi f M / 2}{\sin^2 \pi f} \right)$	$\frac{1.84}{N}$	-53
Blackman	$0.42 + 0.5 \cos \frac{\pi k}{M} + 0.08 \cos \frac{2\pi k}{M}$	$0.42 W_R(f) + 0.25 (W_R(f - \frac{1}{2M}) + W_R(f + \frac{1}{2M})) + 0.04 (W_R(f - \frac{1}{M}) + W_R(f + \frac{1}{M}))$	$\frac{1.68}{N}$	-58

Data Windows in Time



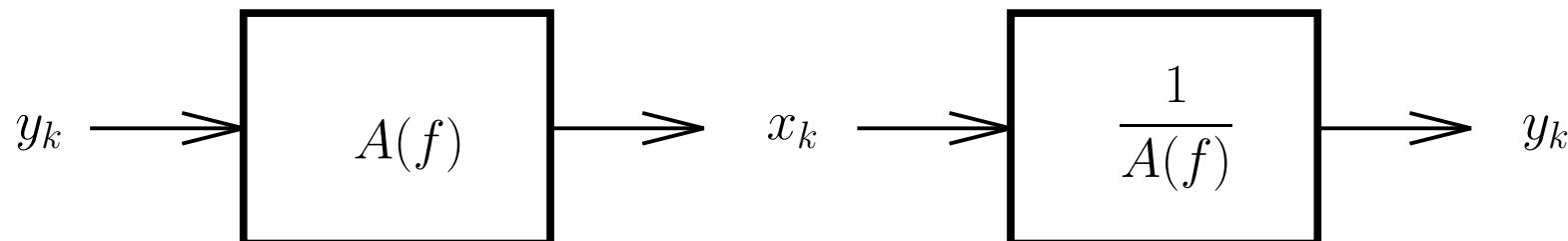
Data Windows in Frequency



Bias Reduction by Data Prewhitening

- If the psdf is constant, then the periodogram is unbiased, even for finite data record length N .
- \Rightarrow idea for bias reduction: Suppose we have some a priori idea about the psdf (or perhaps, after a first application of the periodogram, we have such information). Then we may design a filter $A(f)$ such that the result x_k obtained by filtering y_k with $A(f)$ is approximately white (the technique of linear prediction that we shall see further provides one convenient way of doing this). Then we determine the periodogram $\widehat{S}_{xx,PER}(f)$ of the filtered data. The bias in $\widehat{S}_{xx,PER}(f)$ will be small since x_k is approximately white. The psdf estimate for the original data y_k can then be taken to be

$$\widehat{S}_{yy}(f) = \frac{\widehat{S}_{xx,PER}(f)}{|A(f)|^2}.$$



Use of the DFT and Zero Padding

- The computation of the periodogram: evaluate the Fourier transform of the windowed data for continuous f . The computer can only handle numbers and hence a discrete set of frequencies \Rightarrow Discrete Fourier Transform (DFT) (the Fast Fourier Transform (FFT)). For consideration of the DFT/FFT, it is customary to consider the frequency interval $[0, 1]$ rather than $[-\frac{1}{2}, \frac{1}{2}]$. The DFT evaluates the Fourier transform of a signal of length N at N equispaced frequencies $f_k = k/N$, $k = 0, 1, \dots, N-1$. So we get

$$\widehat{S}_{PER}(f_k) = \frac{1}{N} \left| \sum_{n=0}^{N-1} y_n e^{-j2\pi f_k n} \right|^2 = \frac{1}{N} \left| \sum_{n=0}^{N-1} y_n e^{-j2\pi \frac{k}{N} n} \right|^2, \quad k = 0, 1, \dots, N-1.$$

- We can obtain a finer frequency spacing by padding the data with $N' - N$ zeros and then applying an N' -point DFT. The effective data set becomes

$$y'_n = \begin{cases} y_n & , n = 0, 1, \dots, N-1 \\ 0 & , n = N, N+1, \dots, N'-1 \end{cases}$$

which has the same Fourier transform as the original data set. The frequency spacing of the DFT on the data set y'_n will be $\frac{1}{N'} < \frac{1}{N}$. Zero padding: no extra resolution, only a finer evaluation of the periodogram.

The Averaged Periodogram

- The main problem with the periodogram is its large variance. We may introduce an averaging operation in order to reduce the variance.
- Assume we have K independent data records of length L available (K realizations of the same process). So the data in record i are $y_n^{(i)}$, $n = 0, 1, \dots, L-1$ for $i = 0, 1, \dots, K-1$.
- The *averaged periodogram* is defined as

$$\widehat{S}_{AVPER}(f) = \frac{1}{K} \sum_{i=0}^{K-1} \widehat{S}_{PER}^{(i)}(f)$$

where $\widehat{S}_{PER}^{(i)}(f)$ is the periodogram for data record i :

$$\widehat{S}_{PER}^{(i)}(f) = \frac{1}{L} \left| \sum_{n=0}^{L-1} y_n^{(i)} e^{-j2\pi f n} \right|^2 .$$

- Since the data records are i.i.d., the mean of the averaged periodogram is also the mean of the periodogram of any record. Hence

$$E \widehat{S}_{AVPER}(f) = E \widehat{S}_{PER}^{(0)}(f) = W_{B,L}(f) * S_{yy}(f) , \quad W_{B,L}(f) = \frac{1}{L} \left(\frac{\sin \pi f L}{\sin \pi f} \right)^2 .$$

The Averaged Periodogram (2)

- The variance on the other hand will be decreased by a factor K . Indeed

$$\text{Var} [\widehat{S}_{AVPER}(f)] = \frac{1}{K} \text{Var} [\widehat{S}_{PER}^{(0)}(f)] .$$

- In practice, we normally don't have K independent records but only one record of length N on which to base the spectral estimator. A common approach is to segment the data into K nonoverlapping contiguous blocks of length L so that $N = KL$. In this way, the data records become

$$y_n^{(i)} = y_{n+iL}, \quad n = 0, 1, \dots, L-1; \quad i = 0, 1, \dots, K-1 .$$

Contiguous data records cannot be independent unless the y_k are i.i.d. This dependence does not influence the mean of the averaged periodogram. However, the variance reduction obtained by the averaged periodogram is generally less than a factor K . For processes with rapidly decaying acf, the correlation between different data blocks will be weak. If the data are furthermore Gaussian, the data blocks will also be roughly independent.

The Averaged Periodogram (3)

- With the sectioning of a long data record into K shorter data records, the main design issue becomes one of finding a good compromise. Indeed, as the number of data blocks K increases, the variance of the averaged periodogram decreases. However, also the length $L = N/K$ of the data blocks decreases which means that the bandwidth $\approx \frac{1}{L}$ of the Bartlett window increases. This means that the smearing of the spectral estimate (bias) increases.
- One approach is to start with a large value for K (and hence small L) and to observe the averaged periodogram as K decreases. The smearing will decrease and hence more spectral details become apparent. One can stop reducing K when no more details are transpiring. This procedure is called *window closing*. This procedure is not without risk since the variance increases as K decreases and hence spurious details that appear may simply be due to estimation variance.
- We have already discussed before the use of data prewhitening as a technique for reducing the bias problem. This technique may help to alleviate the bias problem in the averaged periodogram.
- If choose e.g. $L = K = \sqrt{N} \Rightarrow$ AVPER = consistent!

Averaged Periodogram: Welch's Variant

- Welch proposed to
 - introduce a window in the computation of the periodogram corresponding to each data block
 - let the data blocks be overlapping (since the data are windowed, the interaction between consecutive data blocks is less severe than their amount of overlap may suggest). The suggested overlap is as much as 50 or even 75 %.
- Due to the windowing, the spectral leakage gets decreased. Furthermore, due to the overlap, a larger number of data records becomes available, leading to some extra variance reduction.

The Blackman-Tukey Spectral Estimator

- The poor performance of the periodogram may also be illuminated by considering the following equivalent form for the periodogram

$$\begin{aligned}\widehat{S}_{PER}(f) &= \frac{1}{N} \left| \sum_{n=0}^{N-1} y_n e^{-j2\pi f n} \right|^2 = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} y_n y_m e^{-j2\pi f(n-m)} \\ &= \sum_{k=-(N-1)}^{N-1} e^{-j2\pi f k} \frac{1}{N} \sum_{n=0}^{N-1-|k|} y_{n+|k|} y_n = \sum_{k=-(N-1)}^{N-1} \widehat{r}_{yy}(k) e^{-j2\pi f k}\end{aligned}$$

where

$$\widehat{r}_{yy}(k) = \begin{cases} \frac{1}{N} \sum_{n=0}^{N-1-k} y_{n+k} y_n & , \quad k = 0, 1, \dots, N-1 \\ \widehat{r}_{yy}(-k) & , \quad k = -(N-1), \dots, -1. \end{cases}$$

Hence the periodogram may be seen to be an estimator of the psdf by equivalently estimating the acf and using the Wiener-Khinchin relation.

- The poor performance of the periodogram may be understood by considering the quality of the acf estimate. At lag $N-1$ for instance, we get $\widehat{r}_{yy}(N-1) = \frac{1}{N} y_{N-1} y_0$ which apart from being strongly biased is also highly variable due to the lack of averaging. Also, $\widehat{r}_{yy}(k) = 0$, $|k| > N-1$.

The Blackman-Tukey Spectral Estimator (2)

- For general lags, the mean of the acf estimate is

$$E \hat{r}_{yy}(k) = \left(1 - \frac{|k|}{N}\right) r_{yy}(k) = w_{B,N}(k) r_{yy}(k), \quad |k| \leq N-1.$$

So the mean is equal to the true value weighted by the Bartlett window and hence the estimator is biased (except for $k = 0$).

- We could use an unbiased acf estimator by replacing the $\frac{1}{N}$ factor in $\hat{r}_{yy}(k)$ by $\frac{1}{N-|k|}$. However, this choice leads to a spectral estimate with higher variance. Furthermore, this $\widehat{S}(f)$ may be negative at certain frequencies since the unbiased acf estimate does not necessarily correspond to a positive semidefinite sequence.
- So the estimated acf has a bias that increases linearly with lag and a variance that increases also with lag (number of terms in the averaging decreases). Hence, the trustworthiness of the estimated acf decreases with lag.

The Blackman-Tukey Spectral Estimator (3)

- This motivated Blackman and Tukey to introduce a weighting sequence w_k that reflects the quality of the acf estimates: w_k decreases with $|k|$. This leads to *Blackman-Tukey (BT) spectral estimator*

$$\widehat{S}_{BT}(f) = \sum_{k=-(N-1)}^{N-1} w_k \widehat{r}_{yy}(k) e^{-j2\pi fk}$$

where the real sequence w_k (*lag window*) satisfies the following properties:

1. $0 \leq w_k \leq w_0 = 1$
2. $w_{-k} = w_k$
3. $w_k = 0$ for $|k| > M$

where $M \leq N-1$. Due to this last property, we may rewrite \widehat{S}_{BT} as

$$\widehat{S}_{BT}(f) = \sum_{k=-M}^M w_k \widehat{r}_{yy}(k) e^{-j2\pi fk}$$

- The BT spectral estimator is equivalent to the periodogram if $w_k = 1$ for $|k| \leq M = N-1$. The BT estimator is also sometimes called a *weighted covariance* estimator. The weighting will reduce the variance of the spectral estimate, but again by increasing the bias (smearing).

The Blackman-Tukey Spectral Estimator (4)

- We must be careful however that the window chosen will always lead to a non-negative spectral estimate. Remark that the BT estimator can be rewritten as

$$\widehat{S}_{BT}(f) = \mathcal{F}\{w_k \widehat{r}_{yy}(k)\} = W(f) * \widehat{S}_{PER}(f)$$

since $\mathcal{F}\{\widehat{r}_{yy}(k)\} = \widehat{S}_{PER}(f)$. Although $\widehat{S}_{PER}(f) \geq 0$, if $W(f)$ is negative at certain frequencies, then the convolution above may produce negative values. To avoid this, it is preferable that $W(f) \geq 0, \forall f$ or equivalently that w_k is a nonnegative sequence. Only the Bartlett and Parzen windows (in the table) have nonnegative Fourier transforms.

- We get for the mean of the BT estimator

$$\begin{aligned} E \widehat{S}_{BT}(f) &= W(f) * E \widehat{S}_{PER}(f) = W(f) * (W_{B,N}(f) * S_{yy}(f)) \\ &= (\underbrace{W(f) * W_{B,N}(f)}_{= \mathcal{F}\{w_k w_{B,k} \approx w_k\}}) * S_{yy}(f) \approx W(f) * S_{yy}(f) \end{aligned}$$

where we assumed that $N \gg M$. So the true psdf gets smeared again, this time leaving a spectral resolution of about $\frac{1}{M}$. Again, prewhitening the data will help reduce the bias.

The Blackman-Tukey Spectral Estimator (5)

- For the variance, we get under the assumption that the psdf is smooth on a frequency scale of $\frac{1}{M}$, for frequencies not near 0 or $\frac{1}{2}$

$$\text{Var} [\widehat{S}_{BT}(f)] \approx S_{yy}^2(f) \frac{1}{N} \sum_{k=-M}^M w_k^2 .$$

- Again the bias-variance trade-off becomes apparent: for a small bias, M should be chosen large since that will cause the spectral window $W(f)$ to behave as a Dirac delta function. On the other hand, a small variance imposes a small M . A maximum value of $M = N/5$ is usually recommended. As an example, for the Bartlett window

$$\text{Var} [\widehat{S}_{BT}(f)] \approx \frac{2M}{3N} S_{yy}^2(f)$$

so that $M = N/5$ results in a variance reduction by a factor 7.5 compared to the periodogram. Much of the art in classical spectral estimation is in choosing an appropriate window, both the type and the length (M).

- Again, if $M \rightarrow \infty$ as $N \rightarrow \infty$ s.t. $\frac{N}{M} \rightarrow \infty$ (e.g. $M = \sqrt{N}$), then \widehat{S}_{BT} consistent.

The Smoothed Periodogram

- From

$$\widehat{S}_{BT}(f) = \mathcal{F}\{w_k \widehat{r}_{yy}(k)\} = W(f) * \widehat{S}_{PER}(f)$$

we can interpret the BT estimator as the convolution of the periodogram with the spectral window $W(f)$. The role of the spectral window is to smoothen the periodogram, thus possibly increasing the bias but also reducing the variance.

- This suggest computing the periodogram and smoothing it in frequency. Assuming a N' point DFT is used to compute the periodogram, a discrete version with uniform spectral weighting (rectangular $W(f)$) is

$$\widehat{S}_{DAN}(f_k) = \frac{1}{2L+1} \sum_{i=-L}^L \widehat{S}_{PER}\left(f_k + \frac{i}{N'}\right), \quad f_k = \frac{k}{N'}$$

where we have approximately the correspondence: bandwidth = $\frac{2L+1}{N'} = \frac{1}{M}$. This smoothed periodogram is called *Daniell's spectral estimator*. Many other smoothed periodograms are possible by choosing different spectral weightings. Remark that it is quite easy to control the nonnegativity of the spectral weighting so that a positive psdf estimate can be guaranteed.



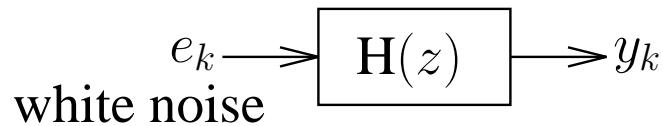
Statistical Signal Processing

Lecture 7

Parametric spectral estimation:

- parametric random process models: AutoRegressive (AR) processes
- linear prediction
- Levinson algorithm
- lattice filters
- AR modeling motivations: LP of an AR(N) process, asymptotics
- AR modeling interpretations, techniques, model order selection

Parametric Random Process Models



The derivation shows the following steps:

$$\begin{aligned} H(z) &= \sum_{k=0}^{\infty} h_k z^{-k} = \frac{\prod_i (1 - z_i z^{-1})}{\prod_k (1 - p_k z^{-1})} \\ &= \frac{\prod_i (1 - z_i z^{-1})}{\prod_k (1 - p_k z^{-1})} = \frac{|z|^2}{|z - z_i|^2} e^{j2\pi f} \\ &\Rightarrow \frac{1}{|z - z_i|^2} e^{j2\pi f} = \frac{1}{|z|^2} e^{-j2\pi f} \end{aligned}$$

- White noise drives a linear time-invariant causal and rational system

$$\begin{aligned} H(z) &= \sum_{k=0}^{\infty} h_k z^{-k} = \frac{\prod_i (1 - z_i z^{-1})}{\prod_k (1 - p_k z^{-1})} = \sum_i \alpha_i \rho_i^{-k} = h_k \\ &= \frac{1}{|z - z_i|^2} e^{j2\pi f} = \frac{1}{|z|^2} e^{-j2\pi f} \end{aligned}$$

- Zero mean input \Rightarrow zero mean output. Variance of the output

$$\sigma_y^2 = \sigma_e^2 \sum_{k=0}^{\infty} h_k^2$$

will be finite if the system is stable ($|p_k| < 1$: poles inside the unit circle). In that case, due to the wide-sense stationarity of the input, the output will be wide-sense stationary with power spectral density function (psdf)

$$S_{yy}(f) = \sigma_e^2 |H(f)|^2$$

only magnitude counts

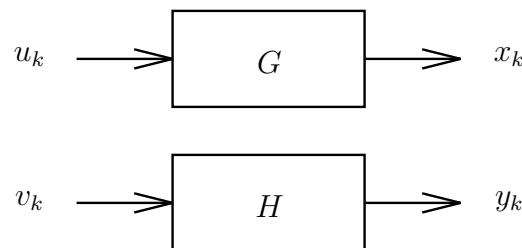
Parametric Random Process Models (2)

- For the modeling of $S_{yy}(f)$, only $|H(f)|$ is important. However, we desire to be able to determine $H(f)$ completely from $S_{yy}(f)$ and hence from $|H(f)|$. Can be done if $H(f)$ minimum-phase ($|z_i| < 1$: zeros inside the unit circle). Minimum-phase \Rightarrow system causally invertible ($H^{-1}(z)$ is causal).
- Due to the uncorrelatedness of the input, we get

$$\mathbf{S}_{ye}(z) = \mathbf{H}(z) \sigma_e^2 \Rightarrow r_{ye}(k) = E y_{n+k} e_n = h_k \sigma_e^2 \quad (= 0, k < 0)$$

\Rightarrow output uncorrelated with future inputs.

- reminder: $\mathbf{S}_{xy}(z) = \mathbf{G}(z) \mathbf{S}_{uv}(z) \mathbf{H}^\dagger(z) \stackrel{z=e^{j2\pi f}}{\Rightarrow} S_{xy}(f) = G(f) S_{uv}(f) H^*(f)$ in the figure:



Autoregressive (AR) Processes

- An Autoregressive process of order n (AR(n)) is obtained by taking an n -th order *all-pole* transfer function

$$= \prod_{i=1}^n (1 - P_i z^{-1})$$

$$H(z) = \frac{1}{A(z)}, \quad A(z) = \sum_{i=0}^n A_i z^{-i}, \quad A_0 = 1$$

$A(z)$ with $A_0 = 1$ is called a monic polynomial in z^{-1} . $A(z)$ as a function of z needs to have all its roots inside the unit circle for $H(z)$ to be minimum-phase.

- The input-output relation is described by the following difference equation

$$y_k = \frac{1}{A(q)} e_k \Rightarrow A(q) y_k = e_k \quad \text{or}$$

$$A(q) y_k = \sum_{i=0}^n A_i q^{-i} y_k = \sum_{i=0}^n A_i y_{k-i} = y_k + A_1 y_{k-1} + \cdots + A_n y_{k-n} = e_k$$

where q^{-1} is the delay operator: $q^{-1} y_k = y_{k-1}$ (and q is the advance operator).

- The name autoregression becomes more obvious when we rewrite this as

$$y_k = -A_1 y_{k-1} - \cdots - A_n y_{k-n} + e_k$$

which expresses y_k as a linear regression on its own past plus independent noise.



Autoregressive (AR) Processes (2)

- The impulse response of the system satisfies the following recursion:

$$\mathbf{A}(z) \mathbf{H}(z) = 1 \Rightarrow \mathbf{A}(q) h_k = \delta_{k0}$$

This allows one to find h_k recursively from $\mathbf{A}(z)$. In particular $h_0 = 1$.

- From the expression for the psdf, we can find

$$\mathbf{S}_{yy}(z) = \frac{\sigma_e^2}{\mathbf{A}(z)\mathbf{A}(1/z)} \Rightarrow \mathbf{A}(z) \mathbf{S}_{yy}(z) = \sigma_e^2 \mathbf{H}(1/z) \quad \text{or} \quad \mathbf{A}(q) r_{yy}(k) = \sigma_e^2 h_{-k}$$

which are the so-called *Yule-Walker equations*.

The Yule-Walker equations for $k = 0, 1, \dots, n$ constitute $n+1$ linear equations that allow one to obtain $r_{yy}(0), \dots, r_{yy}(n)$ from $\sigma_e^2, A_1, \dots, A_n$ or vice versa:

$$\begin{bmatrix} r_{yy}(0) & r_{yy}(1) & \cdots & r_{yy}(n) \\ r_{yy}(1) & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & r_{yy}(1) \\ r_{yy}(n) & \cdots & r_{yy}(1) & r_{yy}(0) \end{bmatrix} \begin{bmatrix} 1 \\ A_1 \\ \vdots \\ A_n \end{bmatrix} = \begin{bmatrix} \sigma_e^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

If for example the $r_{yy}(0), \dots, r_{yy}(n)$ have been obtained from $\sigma_e^2, A_1, \dots, A_n$, then further Yule-Walker equations can be used to obtain the rest of the covariances recursively: $\mathbf{A}(q) r_{yy}(k) = 0, k > n$

Moving Average (MA) Processes

- A Moving Average process of order m (MA(m)) is obtained by taking a m -th order *all-zero* transfer function

$$H(z) = B(z) = \sum_{i=0}^m B_i z^{-i}, \quad B_0 = 1.$$

Again, $B(z)$ is a monic polynomial in z^{-1} . $B(z)$ as a function of z needs again to have all its roots inside the unit circle for $H(z)$ to be minimum-phase.

- The input-output relation is described by the following difference equation

$$y_k = B(q) e_k = e_k + B_1 e_{k-1} + \cdots + B_m e_{k-m}.$$

The name moving average stems from the fact that y_k is computed as a sliding (moving) weighted linear combination (average) of the $m+1$ last inputs.

Moving Average (MA) Processes (2)

- We get for the psdf

$$S_{yy}(z) = \sigma_e^2 \mathbf{B}(z)\mathbf{B}(1/z)$$

or in the time domain

$$r_{yy}(k) = \sigma_e^2 B_k * B_{-k}$$

which implies in particular that

$$r_{yy}(k) = 0, \quad |k| > m.$$

- Due to the particular form of $S_{yy}(z)$, $S_{yy}(z)$ has precisely $2m$ zeros which are such that if z_i is a zero, then so is $1/z_i$. Hence, σ_e^2 and $\mathbf{B}(z)$ can be identified from the $r_{yy}(k)$ by finding the zeros of $S_{yy}(z)$ and assigning the minimum-phase zeros ($|z_i| \leq 1$) to $\mathbf{B}(z)$, and $\sigma_e^2 = S_{yy}(1)/\mathbf{B}^2(1)$. This process is called *spectral factorization*.
- Remark that the Blackman-Tukey spectral estimator implicitly assumes a MA(M) process since the weighted acf is put equal to zero for lags bigger than M .

Autoregressive Moving Average (ARMA) Processes

- An autoregressive moving average (ARMA(n,m)) process is obtained by taking a rational transfer function

$$H(z) = B(z)/A(z)$$

where $A(z)$ and $B(z)$ are monic minimum-phase polynomials as before.

- The input-output relation is described by the following difference equation

$$A(q)y_k = B(q)e_k \text{ or } y_k + A_1y_{k-1} + \cdots + A_ny_{k-n} = e_k + B_1e_{k-1} + \cdots + B_me_{k-m}.$$

This process is clearly a combination of autoregression and moving average.

- The impulse response of the system satisfies the following recursion:

$$A(z)H(z) = B(z) \Rightarrow A(q)h_k = B_k.$$

This allows one to find h_k recursively from $A(z)$ and $B(z)$. In particular $h_0 = 1$.

Autoregressive Moving Average (ARMA) Processes (2)

- From the expression for the psdf, we can find

$$\mathbf{S}_{yy}(z) = \sigma_e^2 \frac{\mathbf{B}(z)\mathbf{B}(1/z)}{\mathbf{A}(z)\mathbf{A}(1/z)}$$

$$\Rightarrow \mathbf{A}(z)\mathbf{S}_{yy}(z) = \sigma_e^2 \mathbf{B}(z)\mathbf{H}(1/z) \quad \text{or} \quad \mathbf{A}(q)r_{yy}(k) = \sigma_e^2 B_k * h_{-k}$$

which are again the *Yule-Walker equations*.

- Given σ_e^2 , $\mathbf{A}(z)$ and $\mathbf{B}(z)$, one can obtain the acf from the Yule-Walker equations in pretty much the same way as for an AR process. Given the acf, one can determine $\mathbf{A}(z)$ from n equations of the form

$$\mathbf{A}(q)r_{yy}(k) = 0, \quad k > m.$$

σ_e^2 and $\mathbf{B}(z)$ can then be obtained by spectral factorization of $\mathbf{A}(z)\mathbf{A}(1/z)\mathbf{S}_{yy}(z)$.

(Forward) Linear Prediction

- Consider predicting the sample y_k (WSS process) linearly from the n previous samples:

$$\text{prediction} \quad \hat{y}_k = - \sum_{i=1}^n A_{n,i} y_{k-i} \quad \text{linear combination}$$

double index of $A_{n,i}$: they will depend on the total number of previous samples n involved in the prediction.

- We shall adjust the coefficients $A_{n,i}$ to minimize the prediction error $y_k - \hat{y}_k$. Of course, for a given sample y_k , it is always possible to find coefficients $A_{n,i}$ such that the prediction error is zero! However, we don't want to choose totally different coefficients for each sample, because then our coefficients would simply be a nonunique nonlinear transformation of our signal and they would not extract any important characteristic of our signal. We want to minimize the prediction error, not instantaneously, but on the average (LMMSE!).
- So we shall minimize the prediction error variance (MSE):

$$\min_{A_{n,i}} \|y_k - \hat{y}_k\|^2 = \min_{A_{n,i}} E (y_k - \hat{y}_k)^2 = \min_{A_{n,i}} E \left(y_k + \sum_{i=1}^n A_{n,i} y_{k-i} \right)^2$$

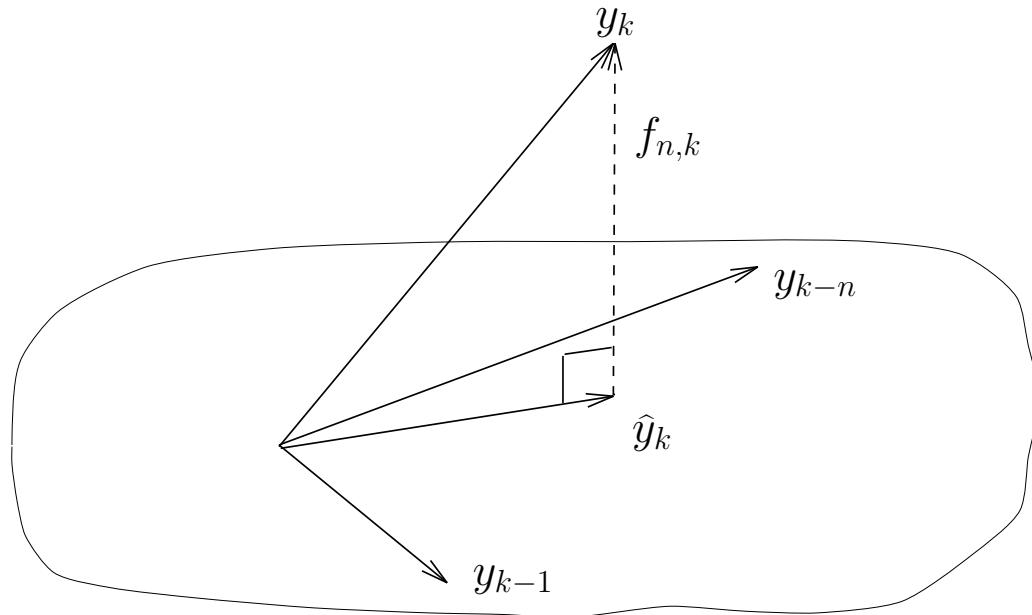
(Forward) Linear Prediction (2)

- Let us introduce the following notation for this forward prediction error of order n at time instant k

$$f_{n,k} = y_k - \hat{y}_k = y_k + \sum_{i=1}^n A_{n,i} y_{k-i} = \sum_{i=0}^n A_{n,i} y_{k-i}$$

where we introduced $A_{n,0} = 1$.

- the solution to the least-squares problem is characterized by the orthogonality condition: the point \hat{y}_k in the subspace spanned by y_{k-1}, \dots, y_{k-n} that is closest to y_k is the one that is the orthogonal projection of y_k onto that subspace.



(Forward) Linear Prediction (3)

- The orthogonality conditions can be written as

$$\langle f_{n,k}, y_{k-i} \rangle = Ef_{n,k}y_{k-i} = \sum_{j=0}^n A_{n,j} E y_{k-i} y_{k-j} = \sum_{j=0}^n r_{|i-j|} A_{n,j} = 0, \quad i = 1, \dots, n$$

where $r_{|i-j|} = E y_{k-i} y_{k-j} = r_{yy}(|i-j|)$. Stationarity $\Rightarrow A_{n,i}$ time invariant (no k).

- Minimal value of the criterion:

$$\sigma_{f,n}^2 = Ef_{n,k}^2 = \min_{A_{n,i}} Ef_{n,k}(y_k + \sum_{i=1}^n A_{n,i} y_{k-i}) = Ef_{n,k}y_k + \sum_{j=1}^n A_{n,j} \underbrace{Ef_{n,k}y_{k-j}}_{=0} = Ef_{n,k}y_k$$

$$\bullet \text{ Introduce : } Y_{n+1}(k) = \begin{bmatrix} y_k \\ y_{k-1} \\ \vdots \\ y_{k-n} \end{bmatrix}, \quad A_n = \begin{bmatrix} 1 \\ A_{n,1} \\ \vdots \\ A_{n,n} \end{bmatrix}, \quad \Rightarrow \quad f_{n,k} = Y_{n+1}^T(k) A_n$$

- Assembling the orthogonality conditions with the expression for the minimal variance:

$$E Y_{n+1}(k) f_{n,k} = \begin{bmatrix} E y_k f_{n,k} \\ E y_{k-1} f_{n,k} \\ \vdots \\ E y_{k-n} f_{n,k} \end{bmatrix} = \begin{bmatrix} \sigma_{f,n}^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

(Forward) Linear Prediction (4)

- On the other hand

$$E Y_{n+1}(k) f_{n,k} = (E Y_{n+1}(k) Y_{n+1}^T(k)) A_n = R_{n+1} A_n = \begin{bmatrix} \sigma_{f,n}^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

\Rightarrow *normal/Yule-Walker equations*, where

$$R_{n+1} = E Y_{n+1}(k) Y_{n+1}^T(k) = \begin{bmatrix} r_0 & r_1 & \cdots & r_n \\ r_1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & r_1 \\ r_n & \cdots & r_1 & r_0 \end{bmatrix}$$

Due to the stationarity of y_k , R_{n+1} is Toeplitz.

- The YW equations are a bit unusual in that they are $n+1$ equations in $n+1$ unknowns, but n unknowns are on the LHS , while 1 unknown ($\sigma_{f,n}^2$) is on the RHS . One solves the last n equations for the n unknowns $A_{n,1}, \dots, A_{n,n}$, which then get substituted in the first equation to find $\sigma_{f,n}^2$.
- To solve a system of n equations in n unknowns takes on the order of n^3 operations (multiplications, additions) in general.

Backward Linear Prediction

- Fast algorithms for solving the normal equations make use of the so-called backward prediction problem.
- Consider now the sense of the time axis as going backward in time, but we shall still work with the $n+1$ most recent samples of y_k . So consider the problem of linearly predicting y_{k-n} backward, i.e. from the n samples that come immediately afterward:

$$\hat{y}_{k-n} = - \sum_{i=1}^n B_{n,i} y_{k-n+i}$$

- We want again to adjust the backward prediction coefficients $B_{n,i}$ to minimize the prediction error variance:

$$\min_{B_{n,i}} \|y_{k-n} - \hat{y}_{k-n}\|^2 = \min_{B_{n,i}} E (y_{k-n} - \hat{y}_{k-n})^2 = \min_{B_{n,i}} E \left(y_{k-n} + \sum_{i=1}^n B_{n,i} y_{k-n+i} \right)^2$$

- notation: backward prediction error of order n at time instant k

$$b_{n,k} = y_{k-n} - \hat{y}_{k-n} = y_{k-n} + \sum_{i=1}^n B_{n,i} y_{k-n+i} = \sum_{i=0}^n B_{n,i} y_{k-n+i}$$

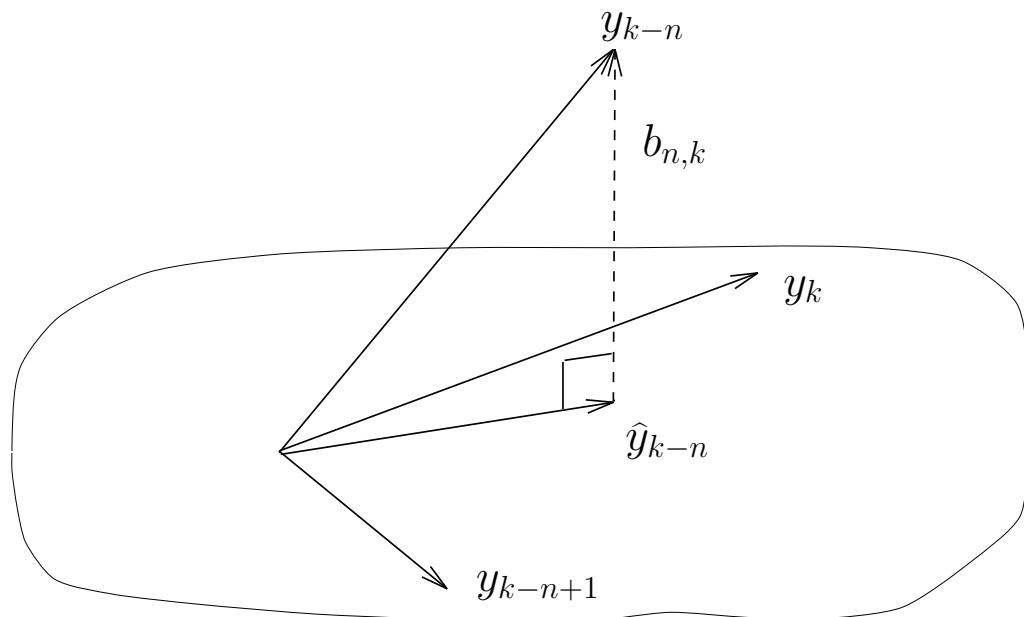
where $B_{n,0} = 1$.

Backward Linear Prediction (2)

- The solution to the least-squares problem is again characterized by the orthogonality condition: the point \hat{y}_{k-n} in the subspace spanned by y_k, \dots, y_{k-n+1} that is closest to y_{k-n} is the one that is the orthogonal projection of y_{k-n} onto that subspace.
- The orthogonality conditions can be written for $i = 1, \dots, n$

$$\langle b_{n,k}, y_{k-n+i} \rangle = E b_{n,k} y_{k-n+i} = \sum_{j=0}^n B_{n,j} E y_{k-n+i} y_{k-n+j} = \sum_{j=0}^n r_{|i-j|} B_{n,j} = 0$$

Again the optimal prediction coefficients are constant (as a function of time).



Backward Linear Prediction (3)

- Minimal value of the criterion:

$$\sigma_{b,n}^2 = Eb_{n,k}^2 = \min_{B_{n,i}} Eb_{n,k}(y_{k-n} + \sum_{i=1}^n B_{n,i} y_{k-n+i}) = Eb_{n,k} y_{k-n} + \sum_{j=1}^n B_{n,j} \underbrace{Eb_{n,k} y_{k-n+j}}_{=0} = Eb_{n,k} y_{k-n}$$

- Assembling the orthogonality conditions with the expression for the minimal variance:

$$EY_{n+1}(k)b_{n,k} = \begin{bmatrix} E y_k b_{n,k} \\ \vdots \\ E y_{k-n+1} b_{n,k} \\ E y_{k-n} b_{n,k} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \sigma_{b,n}^2 \end{bmatrix}$$

$$\bullet \text{ On the other hand, } b_{n,k} = Y_{n+1}^T(k) B_n \text{ where } B_n = \begin{bmatrix} B_{n,n} \\ \vdots \\ B_{n,1} \\ 1 \end{bmatrix}$$

- So we get the normal equations for the backward prediction problem

$$E Y_{n+1}(k) b_{n,k} = (E Y_{n+1}(k) Y_{n+1}^T(k)) B_n = R_{n+1} B_n = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \sigma_{b,n}^2 \end{bmatrix}$$

Forward vs Backward Prediction Quantities

- reverse identity matrix J

$$J = \begin{bmatrix} 0 & & 1 \\ & \ddots & \\ 1 & & 0 \end{bmatrix}, \quad J \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} y_n \\ \vdots \\ y_2 \\ y_1 \end{bmatrix}, \quad [y_1 \ y_2 \cdots y_n] \ J = [y_n \cdots y_2 \ y_1]$$

- A symmetric Toeplitz matrix is *persymmetric* (symmetric w.r.t. the antidiagonal)

$$J \ R_{n+1} \ J = R_{n+1}^T = R_{n+1}$$

where the second identity implies that R_{n+1} is also *centrosymmetric* (symmetric and persymmetric).

- the backward normal equations lead to the forward normal equations

$$R_{n+1} \ J B_n = J R_{n+1} J J B_n = J R_{n+1} B_n = J \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \sigma_{b,n}^2 \end{bmatrix} = \begin{bmatrix} \sigma_{b,n}^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$B_{n,i}$ and $\sigma_{b,n}^2$ satisfy exactly the same equations as the $A_{n,i}$ and $\sigma_{f,n}^2$ and hence $B_n = J A_n$ (or $B_{n,i} = A_{n,i}$) , $\sigma_{b,n}^2 = \sigma_{f,n}^2$

Levinson Algorithm

Goal: fast algorithm for solving the normal equations. The algorithm is recursive in nature. So suppose after recursion n we have A_n and $\sigma_{f,n}^2$. We now try to find the same quantities for order $n+1$. We first look at a trial solution for A_{n+1} which we obtain by appending one zero to A_n :

$$\underbrace{\begin{bmatrix} r_0 & r_1 & \cdots & r_n & r_{n+1} \\ r_1 & \ddots & \ddots & & r_n \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ r_n & & \ddots & \ddots & r_1 \\ r_{n+1} & r_n & \cdots & r_1 & r_0 \end{bmatrix}}_{R_{n+2}} \underbrace{\begin{bmatrix} 1 \\ A_{n,1} \\ \vdots \\ A_{n,n} \\ 0 \end{bmatrix}}_{A_{n+1}} + K_{n+1} \underbrace{\begin{bmatrix} 0 \\ A_{n,n} \\ \vdots \\ A_{n,1} \\ 1 \end{bmatrix}}_{\Delta_{n+1}} = \underbrace{\begin{bmatrix} \sigma_{f,n}^2 \\ 0 \\ \vdots \\ 0 \\ \Delta_{n+1} \end{bmatrix}}_{[\sigma_{f,n+1}^2 \ 0 \cdots 0 \ 0]^T} + K_{n+1} \underbrace{\begin{bmatrix} \Delta_{n+1} \\ 0 \\ \vdots \\ 0 \\ \sigma_{f,n}^2 \end{bmatrix}}_{\Delta_{n+1}}$$

Because R_{n+2} contains R_{n+1} as its upper-left submatrix of one dimension less, and because of the form of our trial solution, the corresponding RHS has the desired form except for the last entry, call it Δ_{n+1} . If we flip our trial solution upside down (the corresponding trial solution for the backward prediction problem), then the RHS also simply gets flipped upside down, because of the centrosymmetry of R_{n+2} . By linearity, a linear combination of the two trial solutions gives the same linear combination of the two RHS 's. We can choose K_{n+1} to get zero for the last element of the RHS .

Levinson Algorithm (2)

- So we should choose

$$\Delta_{n+1} + K_{n+1} \sigma_{f,n}^2 = 0 \quad \Rightarrow \quad K_{n+1} = -\frac{\Delta_{n+1}}{\sigma_{f,n}^2}$$

- Now it becomes clear that the combination of the two trial solutions has itself the right structure (1 as first element) and when multiplied by R_{n+2} gives a right hand side that has the right structure. Hence

$$A_{n+1} = (I + K_{n+1} J) \begin{bmatrix} A_n \\ 0 \end{bmatrix}$$

and in particular, $A_{n+1,n+1} = K_{n+1}$.

- Since we have found A_{n+1} , the top element of the RHS must be $\sigma_{f,n+1}^2$. Hence,

$$\underbrace{\sigma_{f,n+1}^2}_{\geq 0} = \sigma_{f,n}^2 + K_{n+1} \Delta_{n+1} = \underbrace{\sigma_{f,n}^2}_{>0} \underbrace{(1 - K_{n+1}^2)}_{\geq 0}$$

from which it follows that

$$|K_{n+1}| \leq 1$$

Levinson Algorithm (4)

- Levinson algorithm: $\begin{cases} A_n \\ \sigma_{f,n}^2 \end{cases} \rightarrow \begin{cases} A_{n+1} \\ \sigma_{f,n+1}^2 \end{cases}$

$$\begin{aligned}\Delta_{n+1} &= [r_{n+1} \cdots r_1] A_n \\ K_{n+1} &= -\frac{\Delta_{n+1}}{\sigma_{f,n}^2} \\ A_{n+1} &= \begin{bmatrix} A_n \\ 0 \end{bmatrix} + K_{n+1} \begin{bmatrix} 0 \\ J A_n \end{bmatrix} \\ \sigma_{f,n+1}^2 &= \sigma_{f,n}^2 (1 - K_{n+1}^2)\end{aligned}$$

Initialization: $A_0 = [1]$, $\sigma_{f,0}^2 = r_0$.

- Per recursion, the Levinson algorithm needs about $2n$ multiplications and a similar amount of additions. So when the algorithm is run up to some full order N , the total computational complexity is

$$\sum_{n=1}^N 2n = \frac{2N(N+1)}{2} \approx N^2$$

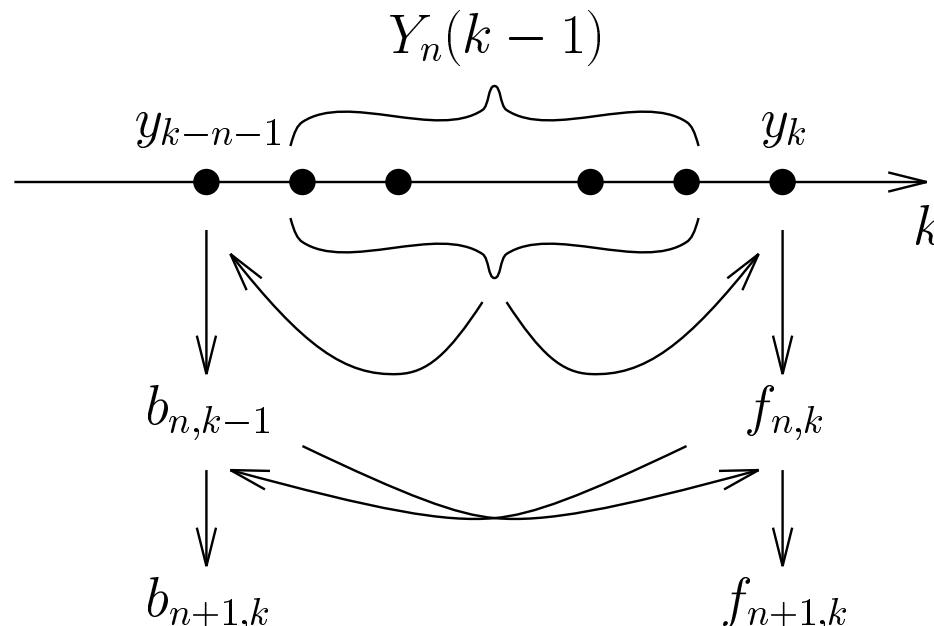
So we really have a fast algorithm for finding A_N .

Levinson Algorithm (5)

- The coefficients $-K_{n+1}$ have an interpretation as correlation coefficients:

$$\begin{aligned}
 -K_{n+1} &= \frac{\Delta_{n+1}}{\sigma_{f,n}^2} = \sigma_{f,n}^{-2} [\sigma_{f,n}^2 0 \cdots 0 \Delta_{n+1}] [0 A_{n,n} \cdots A_{n,1} 1]^T = \sigma_{f,n}^{-2} [A_n^T 0] R_{n+2} \begin{bmatrix} 0 \\ J A_n \end{bmatrix} \\
 &= \frac{E A_n^T Y_{n+1}(k) Y_{n+1}^T(k-1) B_n}{\sigma_{f,n} \sigma_{b,n}} = \frac{E f_{n,k} b_{n,k-1}}{\sqrt{E f_{n,k}^2} \sqrt{E b_{n,k-1}^2}} = \frac{\langle f_{n,k}, b_{n,k-1} \rangle}{\|f_{n,k}\| \|b_{n,k-1}\|}
 \end{aligned}$$

This coefficient is in fact called *Partial Correlation* (PARCOR) coefficient because it describes the partial correlation between y_k and y_{k-n-1} , partial because the influence of $Y_n(k-1)$ in between those two is removed.



Levinson Algorithm (6)

- When we apply the Cauchy-Schwarz formula, then we find immediately $|K_{n+1}| \leq 1$ back.
- prediction filters: consider the z -transforms of the prediction error filter impulse responses:

$$[\mathbf{A}_n(z) \ \mathbf{B}_n(z)] = [1 \ z^{-1} \cdots z^{-n}] [A_n \ B_n]$$

The property $B_n = J A_n$ translates to $\mathbf{B}_n(z) = z^{-n} \mathbf{A}_n(z^{-1}) = z^{-n} \mathbf{A}_n^\dagger(z)$

- The positive definiteness of R_{n+1} has as a consequence that the filter $\mathbf{A}_n(z)$ is minimum-phase, i.e. has all its zeros inside the unit circle (on the unit circle when R_{n+1} is singular). This implies in particular that $1/\mathbf{A}_n(z)$ is guaranteed to be an exponentially stable filter.

Levinson Algorithm (7)

- The positive definiteness of R_{n+1} and the minimum-phase property of the filter $A_n(z)$ are also related to the boundedness of the PARCORs. In fact, we have the following property.

Schur-Cohn Test: A polynomial $A_N(z)$ is minimum-phase if and only if the sequence of PARCORs is bounded: $|K_n| < 1$, $n = N, N-1, \dots, 1$.

- In order to be able to apply this test, we have to know how to find the PARCORs from the filter $A_n(z)$. Whereas the Levinson algorithm is essentially the following *step-up* procedure:

$$A_{n+1} = \begin{bmatrix} A_n \\ 0 \end{bmatrix} + K_{n+1} \begin{bmatrix} 0 \\ J A_n \end{bmatrix} = (I + K_{n+1} J) \begin{bmatrix} A_n \\ 0 \end{bmatrix}$$

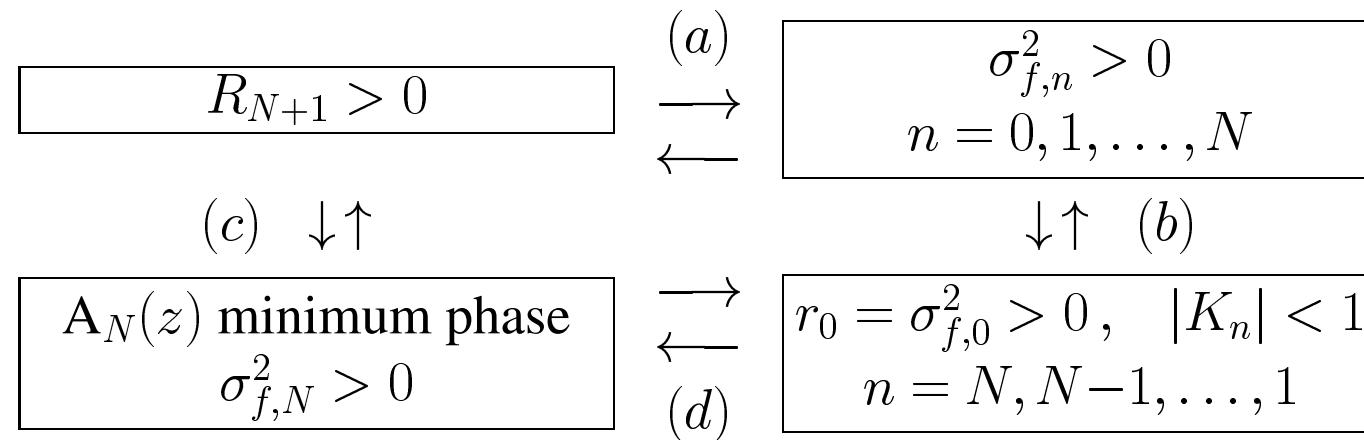
this procedure can be inverted to yield the following *step-down* procedure:

$$\begin{bmatrix} A_n \\ 0 \end{bmatrix} = (I + K_{n+1} J)^{-1} A_{n+1} = \frac{1}{1 - K_{n+1}^2} (I - K_{n+1} J) A_{n+1}, \quad K_{n+1} = A_{n+1,n+1}$$

which can be reiterated to yield all PARCORs, starting from the highest order polynomial.

Levinson Algorithm (8)

- the complete set of equivalences is



- The Schur-Cohn test, which is a subset of equivalence (d), can perhaps most easily be shown by using Rouché's theorem of complex variable theory.
- We mentioned equivalence (c) before.
- Equivalence (b) follows straightforwardly from $\sigma_{f,n}^2 = \sigma_{f,n-1}^2(1 - K_n^2)$.
- Equivalence (a) follows from the triangular factorization of R_n^{-1} interpretation of linear prediction.

Lattice Filters

- We can write out the step-up (Levinson) procedure jointly for forward and backward prediction error filters (using $B_{n+1} = J A_{n+1}$):

$$\begin{aligned} A_{n+1} &= \begin{bmatrix} A_n \\ 0 \\ 0 \end{bmatrix} + K_{n+1} \begin{bmatrix} 0 \\ B_n \\ A_n \end{bmatrix} \\ B_{n+1} &= \begin{bmatrix} 0 \\ B_n \end{bmatrix} + K_{n+1} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{aligned}$$

- If we multiply all sides with $[1 \ z^{-1} \ \dots \ z^{-n-1}]$, we get

$$\begin{aligned} \mathbf{A}_{n+1}(z) &= \mathbf{A}_n(z) + K_{n+1} z^{-1} \mathbf{B}_n(z) \\ \mathbf{B}_{n+1}(z) &= K_{n+1} \mathbf{A}_n(z) + z^{-1} \mathbf{B}_n(z) \end{aligned}$$

which can be rewritten as

$$\begin{bmatrix} \mathbf{A}_{n+1}(z) \\ \mathbf{B}_{n+1}(z) \end{bmatrix} = \begin{bmatrix} 1 & K_{n+1} \\ K_{n+1} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{A}_n(z) \\ \mathbf{B}_n(z) \end{bmatrix}, \quad \begin{bmatrix} \mathbf{A}_0(z) \\ \mathbf{B}_0(z) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

This formula describes one lattice section. From this formula, it is straightforward to draw the realization of the complete lattice filter, by cascading lattice sections and taking the proper initialization into account.

Lattice Filters (1)

- By multiplying

$$\begin{bmatrix} \mathbf{A}_{n+1}(z) \\ \mathbf{B}_{n+1}(z) \end{bmatrix} = \begin{bmatrix} 1 & K_{n+1} \\ K_{n+1} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{A}_n(z) \\ \mathbf{B}_n(z) \end{bmatrix}, \quad \begin{bmatrix} \mathbf{A}_0(z) \\ \mathbf{B}_0(z) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

with $\mathbf{Y}(z)$, the z -transform of y_k , we get

$$\begin{bmatrix} \mathbf{A}_{n+1}(z)\mathbf{Y}(z) \\ \mathbf{B}_{n+1}(z)\mathbf{Y}(z) \end{bmatrix} = \begin{bmatrix} 1 & K_{n+1} \\ K_{n+1} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{A}_n(z)\mathbf{Y}(z) \\ \mathbf{B}_n(z)\mathbf{Y}(z) \end{bmatrix}, \quad \begin{bmatrix} \mathbf{A}_0(z)\mathbf{Y}(z) \\ \mathbf{B}_0(z)\mathbf{Y}(z) \end{bmatrix} = \begin{bmatrix} \mathbf{Y}(z) \\ \mathbf{Y}(z) \end{bmatrix}.$$

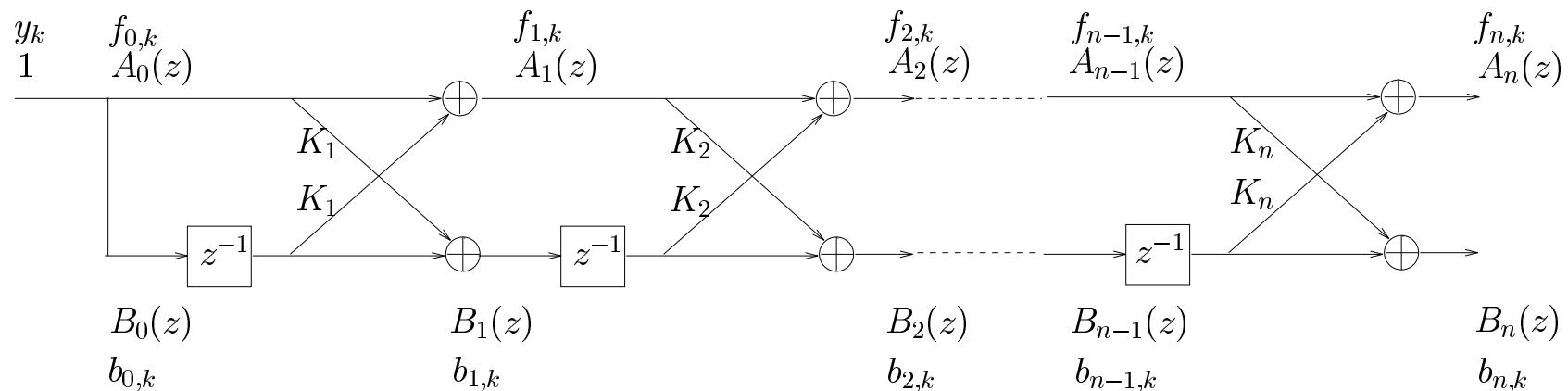
which can be rewritten in the time domain as

$$\begin{bmatrix} f_{n+1,k} \\ b_{n+1,k} \end{bmatrix} = \begin{bmatrix} 1 & K_{n+1} \\ K_{n+1} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & q^{-1} \end{bmatrix} \begin{bmatrix} f_{n,k} \\ b_{n,k} \end{bmatrix}, \quad \begin{bmatrix} f_{0,k} \\ b_{0,k} \end{bmatrix} = \begin{bmatrix} y_k \\ y_k \end{bmatrix}.$$

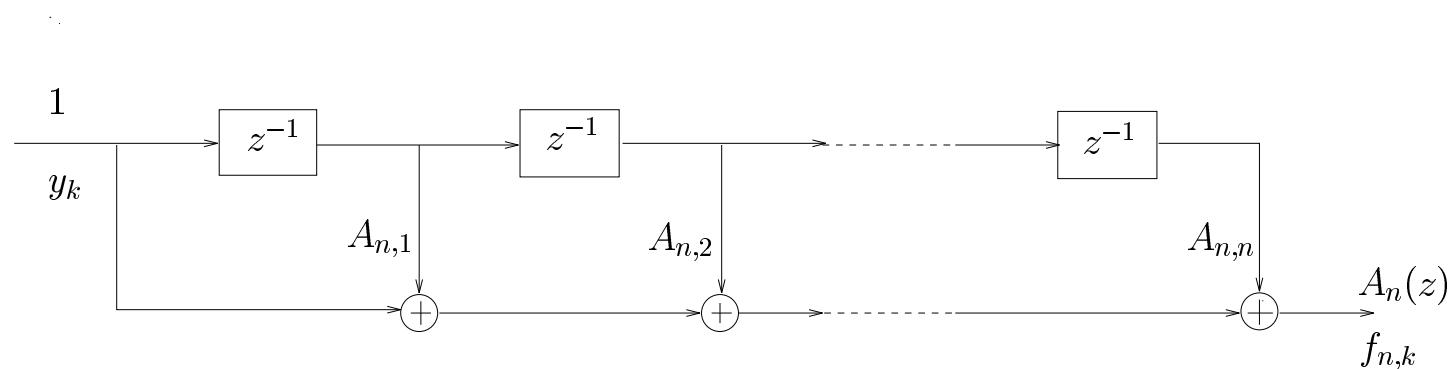
This formula describes one lattice section. From this formula, it is straightforward to draw the realization of the complete lattice filter, by cascading lattice sections and taking the proper initialization into account.

Lattice Filters (2)

- realization of forward and backward prediction via the analysis lattice filter



- equivalent “tapped delay line” or transversal filter realization of the FIR forward prediction error filter



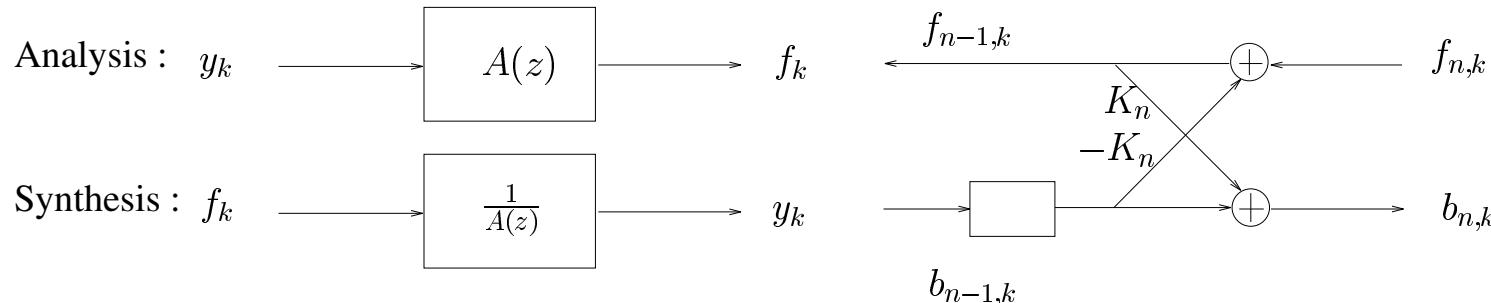
Lattice Filters (3)

Advantages of the lattice filter realization.

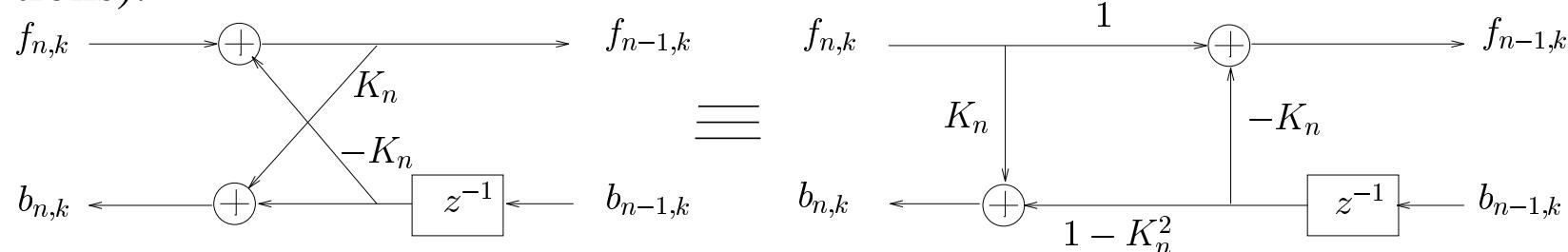
- *order-recursiveness*: since we have that $f_{n,k} = A_n(q)y_k$, $b_{n,k} = B_n(q)y_k$, the prediction errors of all orders show up in the lattice filter if we excite it with the signal y_k .
- The lattice filter has some numerical advantages (especially in fixed-point implementation).
 1. the multipliers that appear in the lattice filter are the PARCORs which are bounded by 1 in magnitude: $|K_n| < 1$
 2. the various prediction errors have lower variance than the input signal: $\sigma_{f,n}^2 \leq \sigma_{f,0}^2 = \sigma_y^2$. Therefore, if the input signal is scaled to be in the interval $[-1, +1]$ (e.g. $\sigma_y < 0.25$ assuming Gaussian signal), then so will be all the signals appearing at all the internal nodes in the filter.
 3. The transfer function has also fairly low sensitivity to perturbations in the filter coefficients K_n .

Synthesis Lattice Filters

- Whereas for linear prediction we are interested in an analysis lattice filter that realizes $A(z)$, for modeling we are interested in the synthesis lattice filter that realizes $1/A(z)$. The synthesis lattice can be obtained from the analysis lattice by straightforward flowgraph manipulations. The roles of input and output get interchanged, we change the direction of the flow:

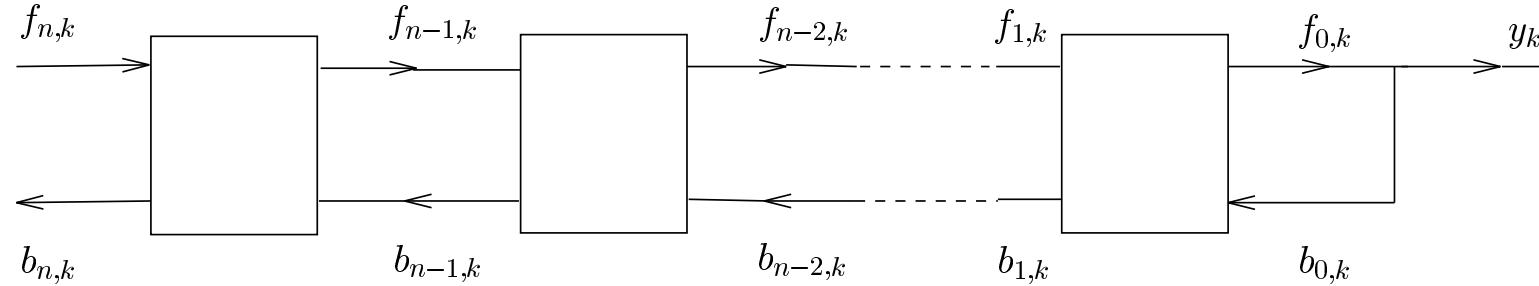


- Since it is usual to have the input on the left and the output on the right, we shall flip the above synthesis lattice section around. This yields the result below, which can also be transformed into the so-called 3-multiplier lattice section on the right (whereas the lattice sections considered so far are 2-multiplier sections):

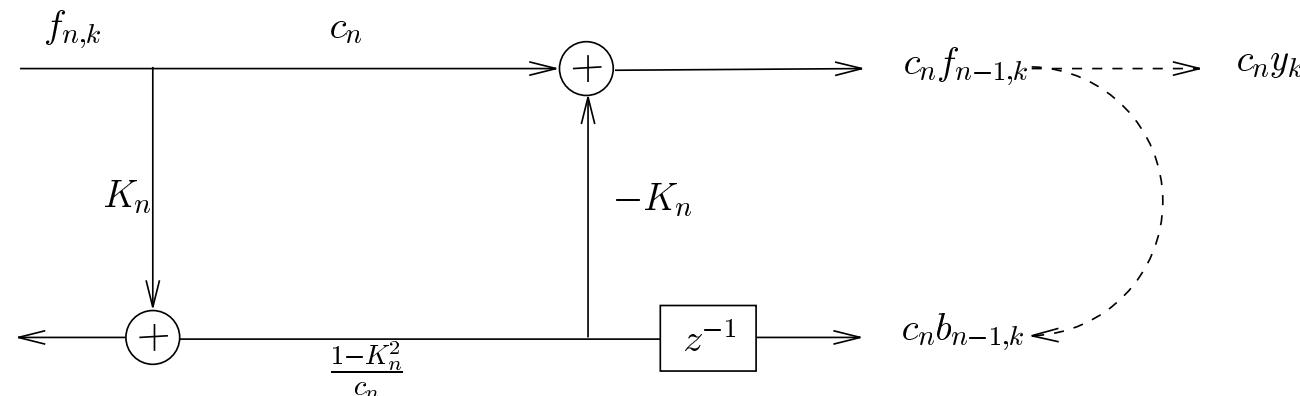


Synthesis Lattice Filters (2)

- general presentation of the all-pole IIR synthesis lattice filter:

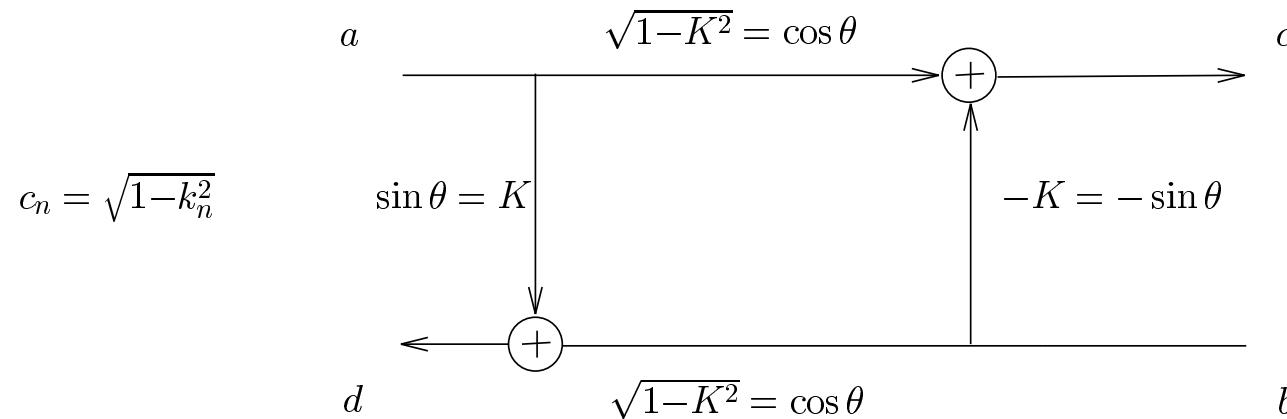


- lattice section transformations:* suppose we factor the gain $1-K_n^2$ of the lower branch into two factors, one of which (c_n) is moved to the upper branch, then this will not change the loop gain of the feedback loop and so the dynamics remain unaltered. However, the net effect of such an operation is that all signals to the right are amplified by the factor c_n . As a result, the overall transfer function becomes $\left(\prod_{n=1}^N c_n \right) / A_N(z)$.



Synthesis Lattice Filters (3)

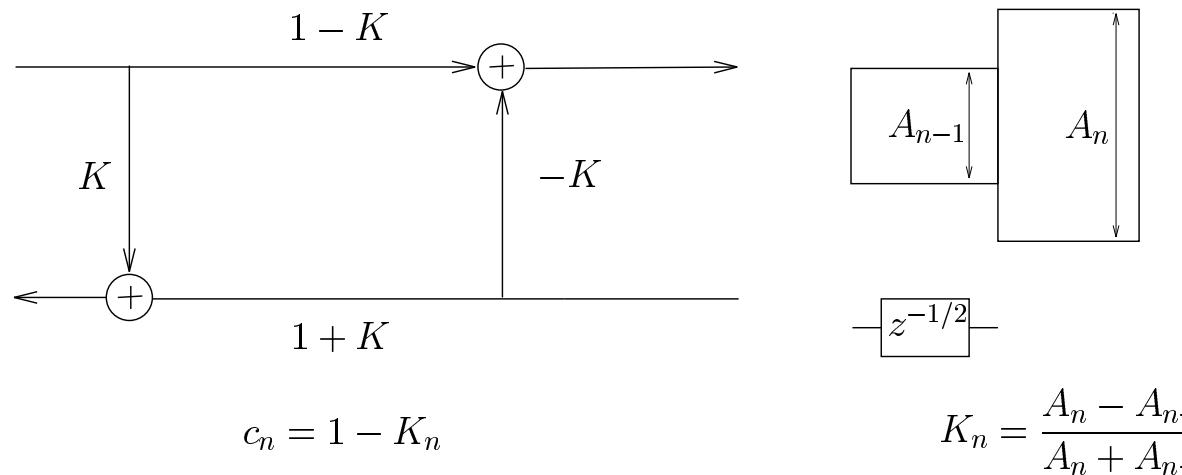
- choice 1 : $c_n = \sqrt{1 - K_n^2} \Rightarrow$ 4-multiplier lattice or *normalized* lattice. This lattice has very good numerical properties since the input-output behavior of the static part of the lattice section is a 2×2 orthogonal rotation, which conserves energy. In a normalized lattice, all signals have the same variance.



$$\begin{bmatrix} c \\ d \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \Rightarrow c^2 + d^2 = a^2 + b^2$$

Synthesis Lattice Filters (4)

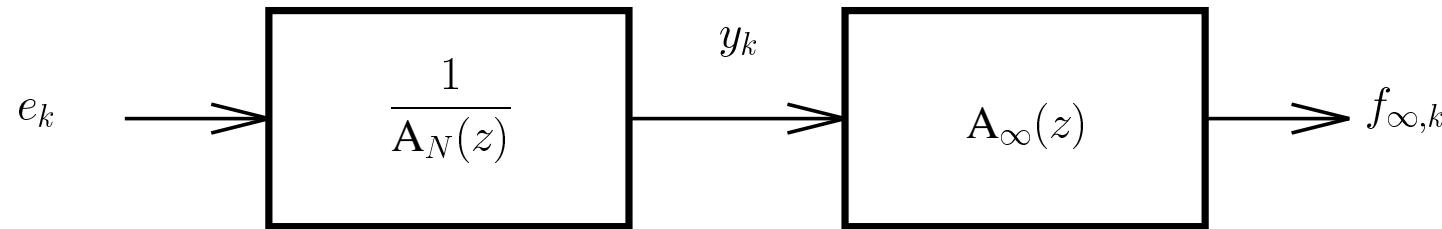
- choice 2 : $c_n = 1 - K_n \Rightarrow$ Kelly-Lochbaum lattice which corresponds exactly to a section of a transmission line. For this reason, the PARCORs are also called *reflection coefficients*. So there exists a close relationship between the all-pole synthesis lattice and a speech production model.



Linear Prediction of AR(N)

- normal equations of LP \equiv Yule-Walker equations for AR(N) process \Rightarrow since solution to linear equations unique, prediction error filter $A_N(z)$ equals denominator of all-pole filter generating the AR process
- alternative point of view: for an AR process, minimizing prediction error variance leads to white prediction errors

white, σ_e^2



- overall transfer from white input e_k to prediction error $f_{\infty,k} = H(q) e_k$

$$H(z) = \frac{A_\infty(z)}{A_N(z)} = \sum_{i=0}^{\infty} h_i z^{-i}, \quad h_0 = H(\infty) = \frac{A_\infty(\infty)}{A_N(\infty)} = \frac{1}{1} = 1$$

$(\frac{1}{A_N(z)}$ is causal since $A_N(z)$ is minimum-phase)

Linear Prediction of AR(N) (2)

$$\begin{aligned} \bullet \quad \sigma_{f,\infty}^2 &= E f_{\infty,k}^2 = E \left(\sum_{i=0}^{\infty} h_i e_{k-i} \right)^2 = E \sum_{i=0}^{\infty} \sum_{l=0}^{\infty} h_i h_l e_{k-i} e_{k-l} \\ &= \sum_{i=0}^{\infty} \sum_{l=0}^{\infty} h_i h_l r_{ee}(i-l) = \sigma_e^2 \sum_{i=0}^{\infty} \sum_{l=0}^{\infty} h_i h_l \delta_{il} = \sigma_e^2 \sum_{i=0}^{\infty} h_i^2 = \sigma_e^2 (1 + \sum_{i=1}^{\infty} h_i^2) \end{aligned}$$

$$\bullet \text{ minimization: } \min_{A_{\infty,i}, i > 0} \sigma_{f,\infty}^2 = \min_{h_i, i > 0} \sigma_{f,\infty}^2 = \sigma_e^2 \quad \text{for } h_i = 0, i > 0.$$

Hence $H(z) = 1$ and $A_{\infty}(z) = A_N(z)$ but also $A_n(z) = A_N(z), n \geq N$

• So for an AR(N) process, we get

$$\begin{cases} K_n = A_{n,n} = -\frac{\Delta_n}{\sigma_{f,n-1}^2} = 0, & n > N \\ f_{n,k} = e_k = \text{white!}, & n \geq N. \end{cases}$$

• special case: white noise = AR(0).

Hence $A_n(z) = 1, K_n = 0, \sigma_{f,n}^2 = r_0, f_{n,k} = y_k, n \geq 0$.

In particular also $\hat{y}_{n,k} = y_k - f_{n,k} = 0$: white noise is unpredictable.

Linear Prediction Asymptotics

- *spectral factorization*: psdf $\mathbf{S}_{yy}(z)$ of a WSS process y_k can be factored as

$$\mathbf{S}_{yy}(z) = \mathbf{S}_{yy}^+(z) \mathbf{S}_{yy}^-(z) , \quad \mathbf{S}_{yy}^-(z) = \mathbf{S}_{yy}^+(1/z) = \mathbf{S}_{yy}^{+\dagger}(z)$$

$\mathbf{S}_{yy}^+(z)$ = the causal minimum-phase *spectral factor* of $\mathbf{S}_{yy}(z)$

$\mathbf{S}_{yy}^-(z)$ = the anticausal maximum-phase spectral factor of $\mathbf{S}_{yy}(z)$

- can interpret

$$y_k = \mathbf{S}_{yy}^+(q) e_k , \quad \sigma_e^2 = 1$$

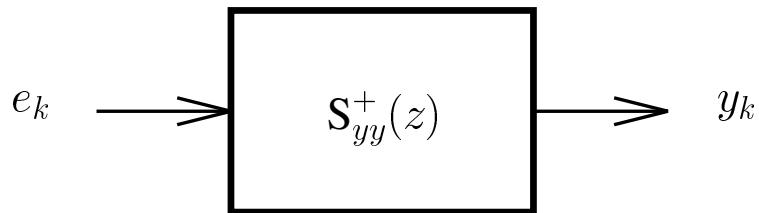
where e_k = white noise

$$\Rightarrow \mathbf{S}_{yy}(z) = \mathbf{S}_{yy}^+(z) \mathbf{S}_{ee}(z) \mathbf{S}_{yy}^+(1/z) = \mathbf{S}_{yy}^+(z) \sigma_e^2 \mathbf{S}_{yy}^+(1/z) = \mathbf{S}_{yy}^+(z) \mathbf{S}_{yy}^-(z)$$

Wold decomposition: any WSS process = MA(∞)

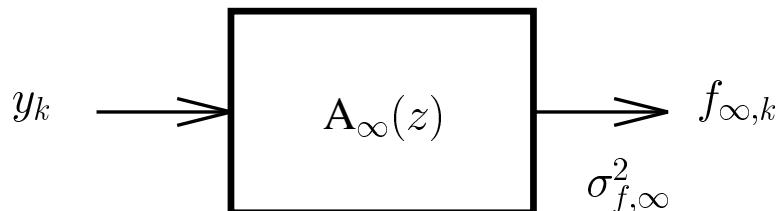
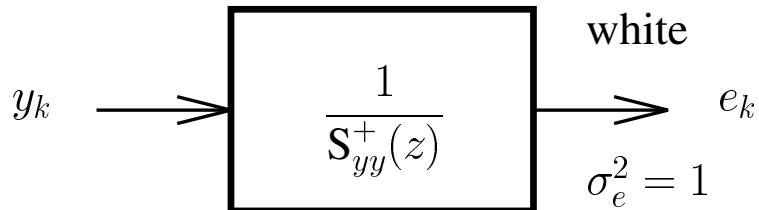
Linear Prediction Asymptotics (2)

white
 $\sigma_e^2 = 1$

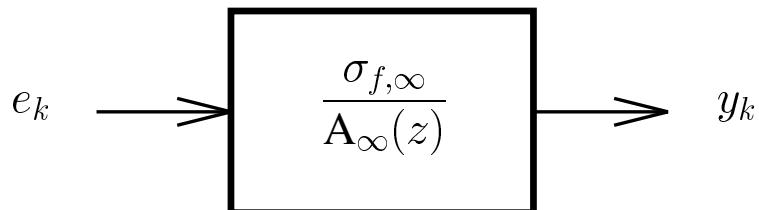


Wold decomposition

WSS process = MA(∞)



white
 $\sigma_e^2 = 1$



Kolmogorov decomposition

WSS process = AR(∞)

Linear Prediction Asymptotics (3)

- consider LP(∞), vector spaces $\text{span} \{y_i, i \leq k\}$ and $\text{span} \{f_{\infty,i}, i \leq k\}$

$$\begin{cases} f_{\infty,k} = \mathbf{A}_{\infty}(q) y_k, \mathbf{A}_{\infty}(z) \text{ causal} \Rightarrow \text{span} \{f_{\infty,i}, i \leq k\} \subset \text{span} \{y_i, i \leq k\} \\ y_k = \frac{1}{\mathbf{A}_{\infty}(q)} f_{\infty,k}, \frac{1}{\mathbf{A}_{\infty}(z)} \text{ causal} \Rightarrow \text{span} \{f_{\infty,i}, i \leq k\} \supset \text{span} \{y_i, i \leq k\} \end{cases}$$

$$\Rightarrow \text{span} \{f_{\infty,i}, i \leq k\} = \text{span} \{y_i, i \leq k\}$$

- Now, by the orthogonality condition of LMMSE estimation, we have

$$f_{\infty,k} \perp \text{span} \{y_i, i < k\} = \text{span} \{f_{\infty,i}, i < k\} \Rightarrow f_{\infty,k} \perp \text{span} \{f_{\infty,i}, i < k\}$$

which implies that $f_{\infty,k}$ is a white process! Hence

$$\sigma_{f,\infty}^2 = \mathbf{S}_{f_{\infty}f_{\infty}}(z) = \mathbf{A}_{\infty}(z) \mathbf{S}_{yy}(z) \mathbf{A}_{\infty}(1/z) \Rightarrow \mathbf{S}_{yy}(z) = \frac{\sigma_{f,\infty}^2}{\mathbf{A}_{\infty}(z) \mathbf{A}_{\infty}(1/z)}$$

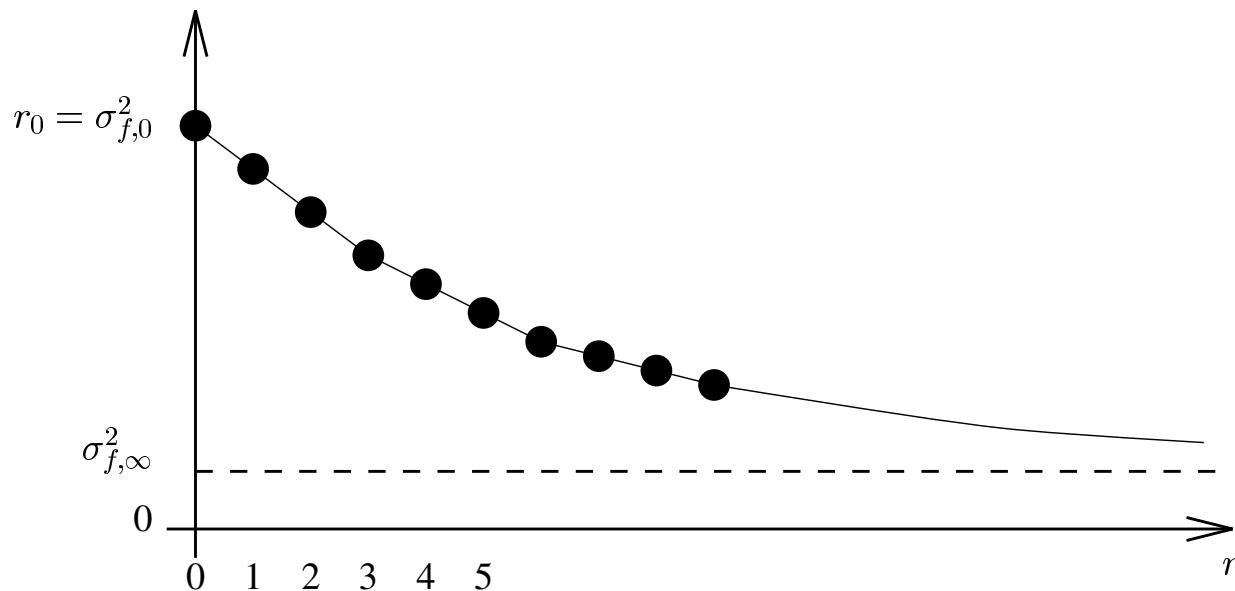
$$\Rightarrow \mathbf{S}_{yy}^+(z) = \frac{\sigma_{f,\infty}^2}{\mathbf{A}_{\infty}(z)}, \quad \mathbf{A}_{\infty}(\infty) = 1 \Rightarrow \sigma_{f,\infty} = \mathbf{S}_{yy}^+(\infty), \quad \mathbf{A}_{\infty}(z) = \frac{\mathbf{S}_{yy}^+(\infty)}{\mathbf{S}_{yy}^+(z)}$$

- *Kolmogorov decomposition:* WSS process = AR(∞)

- in general: $\sigma_{f,\infty}^2 = e^{\int_{-0.5}^{0.5} \ln S_{yy}(f) df}$ (> 0 for purely random processes)

AR Modeling via Linear Prediction

- $0 \leq \sigma_{f,n}^2 = \sigma_{f,n-1}^2(1-K_n^2) \leq \sigma_{f,n-1}^2 \Rightarrow \sigma_{f,n}^2 \searrow \sigma_{f,\infty}^2$



- stationary segment of speech: $K_n \approx 0, n > 10 \Rightarrow$ the curve of $\sigma_{f,n}^2$ decreases rapidly at low orders, but starts to flatten out at n around 8 to 10: $\sigma_{f,10}^2 \gtrsim \sigma_{f,\infty}^2$.

- $\begin{cases} f_{\infty,k} = \text{white noise} \Rightarrow \text{unpredictable} \\ f_{0,k} = y_k = \text{very predictable if } \sigma_{f,\infty}^2 \ll \sigma_{f,0}^2 = r_0 \\ \text{if } \sigma_{f,N}^2 \gtrsim \sigma_{f,\infty}^2 \text{ then } f_{N,k} \text{ no longer very predictable} \Rightarrow \approx \text{white} \end{cases}$

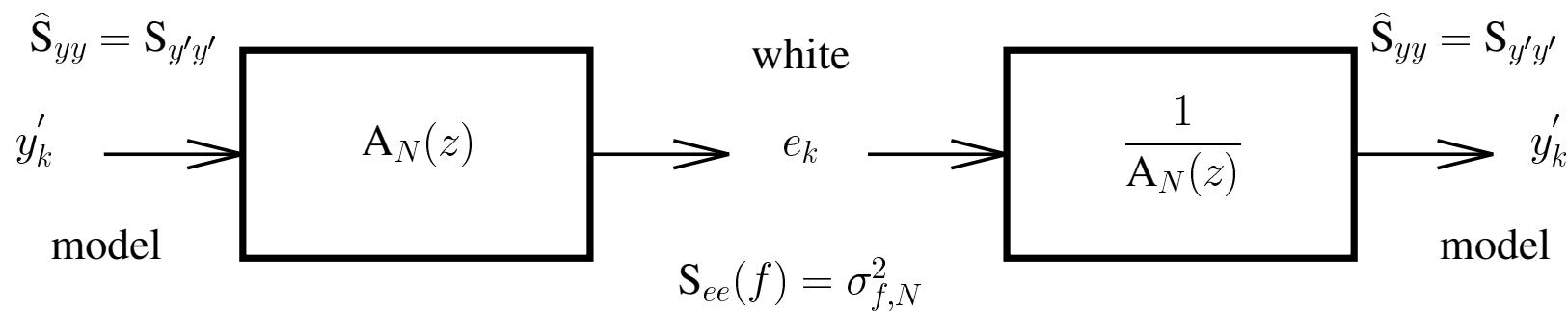
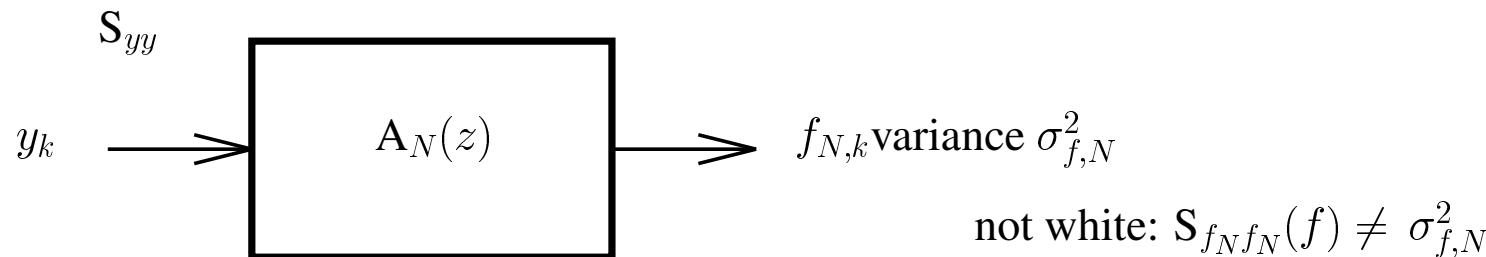
AR Modeling via Linear Prediction (2)

- autoregressive modeling: consider $f_{N,k} \equiv$ white with variance $\sigma_{f,N}^2 = \hat{\mathbf{S}}_{f_N f_N}(z)$

$$\hat{\mathbf{S}}_{yy}(z) = \frac{\hat{\mathbf{S}}_{f_N f_N}(z)}{\mathbf{A}_N(z)\mathbf{A}_N(1/z)} = \frac{\sigma_{f,N}^2}{\mathbf{A}_N(z)\mathbf{A}_N(1/z)} \approx \frac{\mathbf{S}_{f_N f_N}(z)}{\mathbf{A}_N(z)\mathbf{A}_N(1/z)} = \mathbf{S}_{yy}(z)$$

hat = approximation, not estimation

- If $y_k = \text{AR}(n)$ $n \leq N$, then $f_{N,k}$ exactly white \Rightarrow no approximation



AR Modeling: Spectral Interpretation

- prediction error variance minimization in the frequency domain:

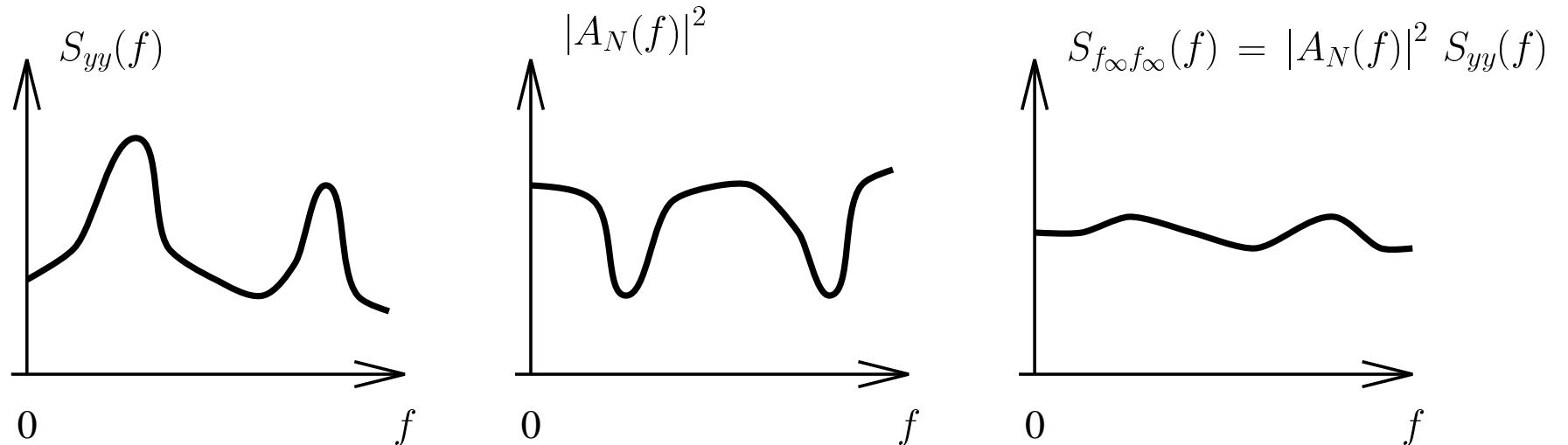
$$\sigma_{f,N}^2 = E f_{N,k}^2 = \int_{-0.5}^{0.5} S_{f_N f_N}(f) df = \min_{A_{N,i}, i=1, \dots, N} \int_{-0.5}^{0.5} |A_N(f)|^2 S_{yy}(f) df$$

- on the other hand: $\int_{-0.5}^{0.5} |A_N(f)|^2 df = \|A_N\|^2 = \sum_{n=0}^N |A_{N,n}|^2 \geq |A_{N,0}|^2 = 1$
monicity $\Rightarrow |A_N(f)|$ cannot be small at all frequencies

- The degrees of freedom $A_{N,i}$ are used to minimize the total contribution of $|A_N(f)|^2 S_{yy}(f) \geq 0$. $\sigma_{f,N}^2$ is made small by making $|A_N(f)|$ small. However, with a limited number of $A_{N,i}$, $|A_N(f)|$ cannot be made small at all frequencies. Hence, $|A_N(f)|$ should preferably be small at those frequencies where $S_{yy}(f)$ is big in order to minimize $\sigma_{f,N}^2$ well.
- This explains why $A_N(z)$ has its zeros on the unit circle if y_k consists of at most N complex sinusoids, in which case $\sigma_{f,N}^2 = 0$.

AR Modeling: Spectral Interpretation (2)

- So $|A_N(f)|$ will be such that $|A_N(f)|^2 S_{yy}(f)$ will not exceed its average value by much when $S_{yy}(f)$ is big.
- But $|A_N(f)|$ will not do much to avoid that $S_{f_\infty f_\infty}(f) = |A_N(f)|^2 S_{yy}(f)$ is small when $S_{yy}(f)$ is small.
- Hence $\widehat{S}_{yy}(f) = \widehat{S}_{yy}(e^{j2\pi f})$ will match $S_{yy}(f)$ closely at the peaks of $S_{yy}(f)$. We say that the AR model follows the ($N/2$ most significant) peaks of $S_{yy}(f)$.



AR Modeling: Covariance Matching

- Suppose we want to model y_k by an AR(N) process y'_k . This means that we want to approximate $S_{yy}(z)$ by

$$\widehat{S}_{yy}(z) = S_{y'y'}(z) = \frac{\sigma_{f,N}^2}{A_N(z)A_N(1/z)}.$$

\Rightarrow we choose a particular parametric form for $\widehat{S}_{yy}(z)$ in which the parameters $\sigma_{f,N}^2, A_{N,1}, \dots, A_{N,N}$ are to be determined.

- Consider now fixing these parameters by introducing $N+1$ covariance matching constraints:

$$\int_{-0.5}^{0.5} \widehat{S}_{yy}(f) e^{j2\pi fk} df = \hat{r}_k = r_k = \int_{-0.5}^{0.5} S_{yy}(f) e^{j2\pi fk} df, \quad k = 0, 1, \dots, N$$

- Then the parameters $\sigma_{f,N}^2, A_{N,1}, \dots, A_{N,N}$ that make the first $N+1$ covariance lags match are the ones that are found from linear prediction! Indeed, the AR(N) process satisfies the Yule-Walker equations with covariance sequence \hat{r}_k . But since $\hat{r}_k = r_k, k = 0, 1, \dots, N$, these Yule-Walker equations for lags $0, 1, \dots, N$ become the normal equations of linear prediction. Hence, the AR(N) model determined by linear prediction matches the first $N+1$ covariance lags.

AR Modeling: Itakura-Saito Distance Minimization

- Itakura and Saito introduced the following distance measure between two power spectral densities:

$$d(S, \widehat{S}) = \int_{-0.5}^{0.5} \left\{ \frac{S(f)}{\widehat{S}(f)} - \ln \frac{S(f)}{\widehat{S}(f)} - 1 \right\} df$$

note that the function $x - 1 - \ln x \geq 0$ for $x \geq 0$ and is only zero for $x = 1$.

- investigate the three properties of a valid distance function:
 - (i) $d(S, S) = 0$
 - (ii) $d(S, \widehat{S}) \geq 0$, $d(S, \widehat{S}) > 0$ if $S \neq \widehat{S}$
 - (iii) $d(S, \widehat{S}) \neq d(\widehat{S}, S)$. Because the symmetry property is not satisfied, the Itakura-Saito distance is not a true distance function. Nevertheless it is a useful measure.

- Assume again an AR(N) model: $\widehat{\mathbf{S}}_{yy}(z) = \frac{\sigma_{f,N}^2}{\mathbf{A}_N(z)\mathbf{A}_N(1/z)}$.

This time, we shall determine the parameters $\sigma_{f,N}^2, A_{N,1}, \dots, A_{N,N}$ by

$$\min_{\sigma_{f,N}^2, A_{N,i}} d(S_{yy}, \widehat{\mathbf{S}}_{yy}) \Rightarrow \sigma_{f,N}^2, A_{N,1}, \dots, A_{N,N} \text{ from linear prediction again}$$

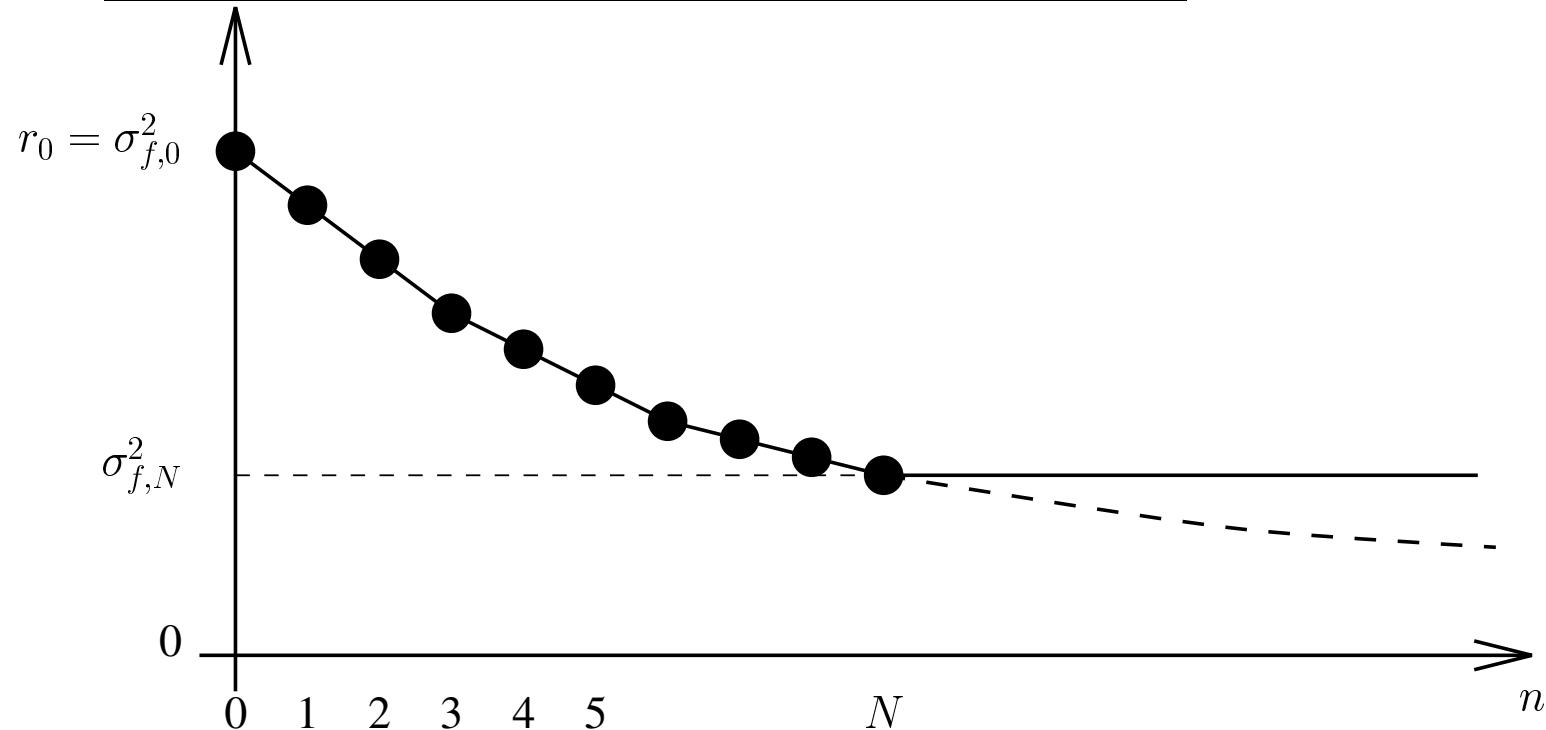
AR Modeling: Maximum Entropy Method

- Suppose: only statistical info about y_k is r_n , $n = 0, 1, \dots, N$ (in practice, r_n estimated from data). In classical spectral estimation theory (Blackman-Tukey spectral estimator), we take $\hat{r}_n = r_n$, $n = 0, 1, \dots, N$,
 $\hat{r}_n = 0$, $n > N \Rightarrow$ spectral estimate with limited resolution capability.
- Suppose: want to model y_k by y'_k with \hat{r}_n such that $\hat{r}_n = r_n$, $n = 0, 1, \dots, N$, and \hat{r}_n , $n > N$ is taken such that the modeled y'_k is as random as possible: we don't want to introduce any further assumptions about the process (the \hat{r}_n) other than $\hat{r}_n = r_n$, $n = 0, 1, \dots, N$. Measure of randomness: *entropy*.
- Key results: when we maximize the entropy w.r.t. a distribution function subject to constraints on some of its second-order moments, the resulting distribution is Gaussian. For a stationary Gaussian process, the entropy rate per sample is given by $\log \sigma_{f,\infty}^2$. Hence, to maximize the entropy, we have to maximize $\sigma_{f,\infty}^2$ for a Gaussian process.
- $\hat{r}_n = r_n$, $n = 0, 1, \dots, N$ are given $\Rightarrow \sigma_{f,n}^2$, $n = 0, 1, \dots, N$ are determined.
Then

$$\sigma_{f,\infty}^2 = \sigma_{f,N}^2 \prod_{n=N+1}^{\infty} (1 - K_n^2) \leq \sigma_{f,N}^2$$

We can choose $\{\hat{r}_n, n > N\} \Rightarrow \{K_n, n > N\}$.

AR Modeling: Maximum Entropy Method (2)



- $\max_{K_n, n > N} \sigma_{f,\infty}^2 = \sigma_{f,N}^2 \max_{K_n, n > N} \prod_{n=N+1}^{\infty} (1 - K_n^2) = \sigma_{f,N}^2$ for $K_n = 0, n > N$
- \Rightarrow Gaussian AR(N) process that satisfies LP normal equations
- Note that we obtain the maximum entropy covariance extension $\hat{r}_n, n > N$:

$$\hat{r}_n = r_n, \quad n = 0, 1, \dots, N, \quad \hat{r}_n = - \sum_{i=1}^N A_{N,i} \hat{r}_{n-i}, \quad n > N$$

This contrasts with MA(N) model matching for which $\hat{r}_n = 0, n > N$.

AR Modeling: Spectral Flatness Measure

- first: spectral flatness measure for a covariance matrix R_N with positive real eigenvalues $\lambda_1 \dots \lambda_N$.
- For white noise, we have $R_N = \sigma_y^2 I_N$ and hence $\lambda_1 = \dots = \lambda_N = \sigma_y^2$.
- For a general covariance matrix, how close is the process to white noise?
Answer: flatness measure FM of the distribution of the λ_i :

$$FM = \frac{\text{geometric avg.}}{\text{arithmetic avg.}} = \frac{\left(\prod_{i=1}^N \lambda_i\right)^{1/N}}{\frac{1}{N} \sum_{i=1}^N \lambda_i} = \frac{e^{\ln\left(\prod_{i=1}^N \lambda_i\right)^{1/N}}}{\frac{1}{N} \sum_{i=1}^N \lambda_i} = \frac{e^{\frac{1}{N} \sum_{i=1}^N \ln \lambda_i}}{\frac{1}{N} \sum_{i=1}^N \lambda_i} \leq 1$$

$$FM = 1 \text{ iff } \lambda_i \equiv \sigma_y^2$$

AR Modeling: Spectral Flatness Measure (2)

- to show $FM \leq 1$: *Jensen's Inequality*

Let $f(\cdot)$ be a convex function and X a random variable. Then

$$f(E X) \leq E f(X) .$$

If $f(\cdot)$ is strictly convex, then strict inequality holds unless the distribution of X is concentrated in one point. If X has a discrete distribution, taking on the M values x_i with probabilities $\alpha_i > 0$, $\sum_{i=1}^M \alpha_i = 1$ then this can be written as

$$f\left(\sum_{i=1}^M \alpha_i x_i\right) \leq \sum_{i=1}^M \alpha_i f(x_i) .$$

Proof: recursively: for $M = 2$: definition of a convex function. Then

$$f\left(\sum_{i=1}^M \alpha_i x_i\right) = f\left(\alpha_1 x_1 + (1-\alpha_1) \sum_{i=2}^M \frac{\alpha_i}{\sum_{k=2}^M \alpha_k} x_i\right) \leq \alpha_1 f(x_1) + (1-\alpha_1) f\left(\sum_{i=2}^M \frac{\alpha_i}{\sum_{k=2}^M \alpha_k} x_i\right)$$

The property for a continuous distribution can be shown by letting $M \rightarrow \infty$.

- Application: $f(x) = -\ln x$ (strictly convex), $x_i = \lambda_i$, $\alpha_i = 1/N$, $M = N \Rightarrow$

$$-\ln\left(\frac{1}{N} \sum_{i=1}^N \lambda_i\right) \leq -\frac{1}{N} \sum_{i=1}^N \ln \lambda_i \Rightarrow \frac{1}{N} \sum_{i=1}^N \lambda_i \geq e^{\frac{1}{N} \sum_{i=1}^N \ln \lambda_i} \Rightarrow FM \leq 1$$

AR Modeling: Spectral Flatness Measure (3)

- As $N \rightarrow \infty$, the distribution of the λ_i behaves similarly as $S_{yy}(f)$, e.g.

$$\lim_{N \rightarrow \infty} \frac{\max_{i=1,\dots,N} \lambda_i}{\min_{i=1,\dots,N} \lambda_i} = \frac{\max_{f \in [0,0.5]} S_{yy}(f)}{\min_{f \in [0,0.5]} S_{yy}(f)}.$$

- Similarly, the flatness measure becomes in the limit as $N \rightarrow \infty$ the spectral flatness measure SFM of y_k

$$FM = \frac{e^{\frac{1}{N} \sum_{i=1}^N \ln \lambda_i}}{\frac{1}{N} \sum_{i=1}^N \lambda_i} \xrightarrow{N \rightarrow \infty} \frac{e^{\int_{-0.5}^{0.5} \ln S_{yy}(f) df}}{\int_{-0.5}^{0.5} S_{yy}(f) df} = \frac{\sigma_{f,\infty}^2}{\sigma_y^2} = SFM = \xi_y \in [0, 1]$$

Note that if e_k is white noise, then $\xi_e = 1$.

- Apply SFM to $f_{N,k}$: $\xi_{f_N} = \frac{\sigma_{f,\infty}^2}{\sigma_{f,N}^2} \Rightarrow \max_{A_{N,i}, i=1,\dots,N} \xi_{f_N} \leftrightarrow \min_{A_{N,i}, i=1,\dots,N} \sigma_{f,N}^2$

Hence, LP = choose $A_{N,i}$ to minimize $\sigma_{f,N}^2$

= make $S_{f_N f_N}(f) = S_{yy}(f) |A_N(f)|^2$ as flat (white) as possible.

AR Modeling: Spectral Estimation Qualities

So far: AR(N) modeled from $r_{yy}(k)$, $k = 0, 1, \dots, N$.

In practice: we estimate $\sigma_{f,N}^2, A_{N,1}, \dots, A_{N,N}$ from M samples y_0, y_1, \dots, y_{M-1} .

Given all the previous observations, we can conclude that the AR(N) model obtained with linear prediction gives a good spectral estimate for N sufficiently high.

More precisely we can state:

- *bias*: the AR(N) model is only unbiased for AR(n) processes with $n \leq N$. Bias smaller for spectral peaks than for spectral valleys. In general, the bias disappears as $N \rightarrow \infty$.
- *variance*: so far we have assumed r_0, \dots, r_N known. In practice, they will have to be estimated from data. However, since they represent $N+1$ parameters to be estimated, we can state that the variance will be roughly proportional to N/M . So the variance will be low if $N \ll M$.
- *resolution*: due to the all-pole filter, we can in principle model arbitrarily closely spaced spectral peaks (sinusoids). There is no spectral smearing. For this reason, the AR modeling is also called a *high resolution* technique.

AR Modeling: Techniques: Least-Squares

- least-squares: replace statistical averages by temporal averages

$$\begin{bmatrix} f_{N,0} \\ \vdots \\ f_{N,N} \\ \vdots \\ f_{N,M-1} \\ \vdots \\ f_{N,M+N-1} \end{bmatrix} = \underbrace{\begin{bmatrix} y_0 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ y_N & & \ddots & 0 \\ \vdots & & \cdots & y_0 \\ y_{M-1} & \cdots & y_{M-N+1} \\ 0 & \ddots & & \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & y_{M-1} \end{bmatrix}}_{\mathcal{Y} \text{ (Toeplitz data matrix)}} \begin{bmatrix} 1 \\ A_{N,1} \\ \vdots \\ A_{N,N} \end{bmatrix} \left. \begin{array}{l} \mathcal{Y}^{pre} \\ \mathcal{Y}^{cov} \end{array} \right\} \left. \begin{array}{l} \mathcal{Y}^{corr} \\ \mathcal{Y}^{po} \end{array} \right\}$$

1. *correlation method* (pre- and postwindowed)

$$\min_{A_{N,i}} \sum_{k=0}^{M+N-1} f_{N,k}^2 \Rightarrow \mathcal{Y}^T \mathcal{Y} A_N = [(M+N)\sigma_{f,N}^2 \ 0 \cdots 0]^T, \quad \mathcal{Y} = \mathcal{Y}^{corr}$$

$\mathcal{Y}^T \mathcal{Y} = \text{Toeplitz} \Rightarrow \text{Levinson, } A_N(z) \text{ minimum-phase}$

AR Modeling: Techniques (2)

2. covariance method (unwindowed)

only take prediction errors calculated with actual data

$$\min_{A_{N,i}} \sum_{k=N}^{M-1} f_{N,k}^2 \Rightarrow \mathcal{Y}^T \mathcal{Y} A_N = [(M-N)\sigma_{f,N}^2 \ 0 \cdots 0]^T, \quad \mathcal{Y} = \mathcal{Y}^{cov}$$

$\mathcal{Y}^T \mathcal{Y} \neq$ Toeplitz \Rightarrow not Levinson, $A_N(z)$ not guaranteed minimum-phase
nevertheless, better estimation quality due to absence of windowing, especially
for short data lengths M

3. modified covariance method (unwindowed)

backward prediction errors:

$$\begin{bmatrix} \vdots \\ b_{N,k} \\ \vdots \end{bmatrix} = \mathcal{Y} \begin{bmatrix} A_{N,N} \\ \vdots \\ A_{N,1} \\ 1 \end{bmatrix}$$

$$\min_{A_{N,i}} \sum_{k=N}^{M-1} (f_{N,k}^2 + b_{N,k}^2) \Rightarrow (\mathcal{Y}^T \mathcal{Y} + J \mathcal{Y}^T \mathcal{Y} J) A_N = [2(M-N)\sigma_N^2 \ 0 \cdots 0]^T, \quad \mathcal{Y} = \mathcal{Y}^{cov}$$

$\mathcal{Y}^T \mathcal{Y} + J \mathcal{Y}^T \mathcal{Y} J$ centro-symmetric \Rightarrow further improved estimate

AR Modeling: Techniques (3)

4. Itakura-Saito method

keep Levinson recursions, but
replace statistical average by temporal average in the PARCOR calculations:

$$K_{n+1} = -\frac{\sum_k f_{n,k} b_{n,k-1}}{\sqrt{\sum_k f_{n,k}^2} \sqrt{\sum_k b_{n,k-1}^2}}$$

5. Burg method

take the Levinson recursions for the prediction errors:

$$\begin{cases} f_{n+1,k} = f_{n,k} + K_{n+1} b_{n,k-1} \\ b_{n+1,k} = b_{n,k-1} + K_{n+1} f_{n,k} \end{cases}$$

and take the modified covariance criterion

$$\min_{K_{n+1}} \sum_k (f_{n+1,k}^2 + b_{n+1,k}^2) \Rightarrow K_{n+1} = -\frac{\sum_k f_{n,k} b_{n,k-1}}{\frac{1}{2} \left(\sum_k f_{n,k}^2 + \sum_k b_{n,k-1}^2 \right)}$$

One can show that

$$|K_n^{Burg}| \leq |K_n^{Ita-S}| \leq 1 \Rightarrow A_N(z) \text{ minimum-phase}$$

first inequality: arithmetic average \geq geometric average

second inequality: Cauchy-Schwarz

AR Modeling: Techniques (4)

6. Maximum-Likelihood

assume the y_k given $\theta = [\sigma_{f,N}^2 \ A_{N,1} \cdots A_{N,N}]^T$ Gaussian and AR(N),
and estimate θ via the ML method

7. Method of Moments

take the normal equations of linear prediction

$$R_{N+1} \ A_N = [\sigma_{f,N}^2 \ 0 \cdots 0]^T$$

and replace r_n , $n = 0, 1, \dots, N$ by sample estimates (such that $\widehat{R}_{N+1} > 0$).

example: correlation method (biased sample moments $\hat{r}_n = \frac{1}{M} \sum_{k=0}^{M-1-n} y_{k+n}y_k$)

another example: unbiased sample moments $\hat{r}_n = \frac{1}{M-n} \sum_{k=0}^{M-1-n} y_{k+n}y_k$ do not guarantee $\widehat{R}_{N+1} > 0$ but usually $\widehat{R}_{N+1} > 0$ since $R_{N+1} > 0$ and the \hat{r}_n are consistent estimates of the r_n

AR Modeling: Order Selection

- given $r_0, r_1, \dots \Rightarrow \sigma_{f,N}^2 \searrow$ as $N \rightarrow \infty$: the higher N the better
 - given data y_0, \dots, y_{M-1} , so far: assumed N given
 - Due to least-squares fit: estimated $\hat{\sigma}_{f,N}^2$ decreases with N and $\hat{\sigma}_{f,N}^2 < \sigma_{f,N}^2$.
example: covariance method with $M = 2N \Rightarrow \hat{\sigma}_{f,N}^2 = 0$! Exact fit possible
(exactly determined equations).
On the basis of $\hat{\sigma}_{f,N}^2$: the higher N the better.
 - When try \widehat{A}_N , estimated with certain data, on other data, $E\widehat{f}_{N,k}^2 > \sigma_{f,N}^2$ due to estimation errors in \widehat{A}_N .
 - order selection criteria: $\min_N c(N)$, $c(N) = g(\hat{\sigma}_{f,N}^2) + d(N)$
where $g(\cdot)$, $d(\cdot)$ are monotonously increasing functions $\Rightarrow g(\hat{\sigma}_{f,N}^2)$ decreases with N whereas $d(N)$ increases with $N \Rightarrow$ a compromise has to be made, leading to a finite optimal N
1. Akaike [’70]: *Final Prediction Error (FPE)*

$$FPE(N) = \frac{M+N}{M-N}\hat{\sigma}_{f,N}^2 = \hat{\sigma}_{f,N}^2 + \frac{2N}{M-N}\hat{\sigma}_{f,N}^2$$

AR Modeling: Order Selection (2)

2. Akaike [’74]: *Akaike Information Criterion (AIC)*

$$AIC(N) = M \ln \widehat{\sigma}_{f,N}^2 + 2N$$

for $\frac{N}{M} \ll 1$, $AIC(N) \approx M \ln FPE(N)$.

3. Rissanen [’78]: *Minimum Description Length (MDL)*

$$MDL(N) = \ln \widehat{\sigma}_{f,N}^2 + (N+1) \frac{\ln M}{M}$$

MDL gives consistent estimates: $\widehat{N}_{MDL} \rightarrow N$ as $M \rightarrow \infty$ for an AR(N) process

- remark that the Levinson-style order-recursive solutions are helpful: find AR estimates for all orders and then choose the best order according to $\min_N c(N)$ where $c = FPE, AIC, MDL$

Linear Time-Frequency Representations

- non-stationary processes \Rightarrow no ergodicity \Rightarrow cannot obtain statistical averages as limits of time averages \Rightarrow no time-averaging
- spectral estimation \rightarrow spectral representation
- fundamental spectral representation: Fourier transform

$$Y(f) = \int_{-\infty}^{\infty} y(t) e^{-j2\pi ft} dt$$

$y(t)$: we know precisely at what time something happens but we don't know at which frequencies

$Y(f)$: we know precisely the different spectral components of the signal, but we don't know when they occur

- non-stationary signals: would like a joint time-frequency representation
e.g.: piano piece: pitch (fundamental frequency) is a piecewise constant function of time (notes being played), $y(t)$ does not tell us which notes are being played, $Y(f)$ shows all the notes but does not tell us when they occur.



Statistical Signal Processing (SSP)

Lecture 7a

ch2: Linear Time-Frequency Representations

- short-term Fourier transform
- filter banks
- discrete Wavelet transform

Linear Time-Frequency Representations

- non-stationary processes \Rightarrow no ergodicity \Rightarrow cannot obtain statistical averages as limits of time averages \Rightarrow no time-averaging
- spectral estimation \rightarrow spectral representation
- fundamental spectral representation: Fourier transform

$$Y(f) = \int_{-\infty}^{\infty} y(t) e^{-j2\pi ft} dt$$

$y(t)$: we know precisely at what time something happens but we don't know at which frequencies

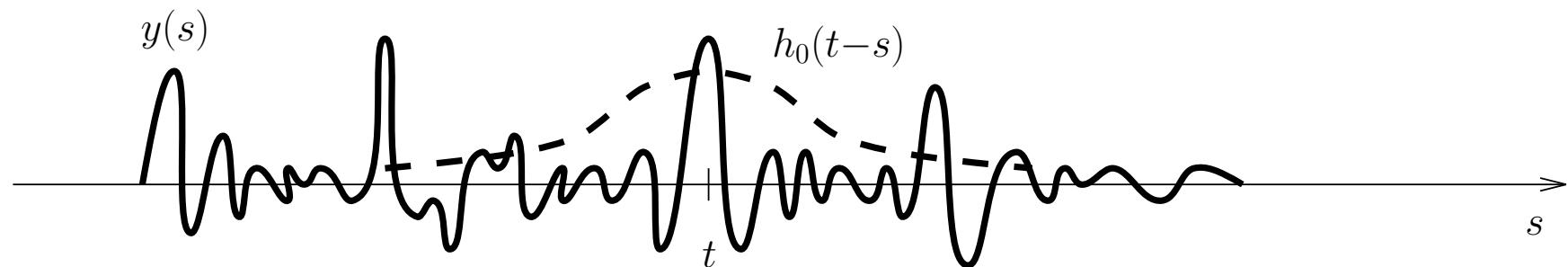
$Y(f)$: we know precisely the different spectral components of the signal, but we don't know when they occur

- non-stationary signals: would like a joint time-frequency representation
e.g.: piano piece: pitch (fundamental frequency) is a piecewise constant function of time (notes being played), $y(t)$ does not tell us which notes are being played, $Y(f)$ shows all the notes but does not tell us when they occur.

Short-Time Fourier Transform (STFT)

- to capture the variation in time of the frequency contents we introduce a sliding window $h_0(t-s)$ centered at t

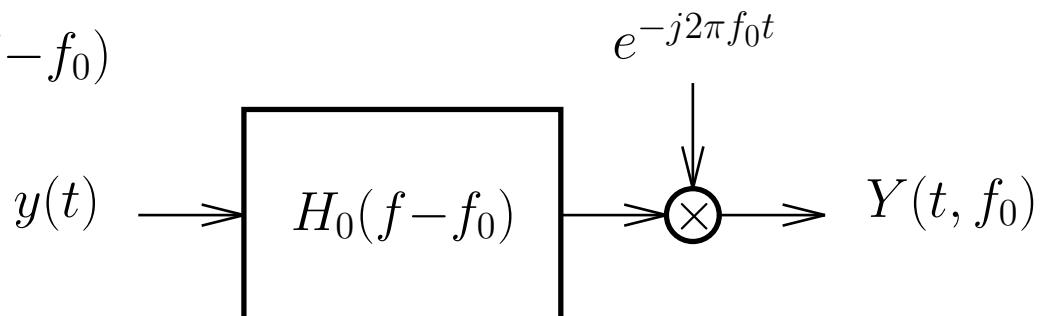
$$Y(t, f) = \int_{-\infty}^{\infty} y(s) h_0(t-s) e^{-j2\pi fs} ds$$



- filterbank interpretation of the STFT

$$Y(t, f_0) = e^{-j2\pi f_0 t} \int_{-\infty}^{\infty} y(s) h_0(t-s) e^{j2\pi f_0(t-s)} ds$$

$h_0(t) e^{j2\pi f_0 t}$ = impulse response of $H_0(f-f_0)$



$y(t)$ is bandpass filtered by $H_0(f-f_0)$ (\Rightarrow centered at frequency f_0), and then down modulated by f_0 to give $Y(t, f_0)$ which is centered around dc

Short-Time Fourier Transform (STFT) (2)

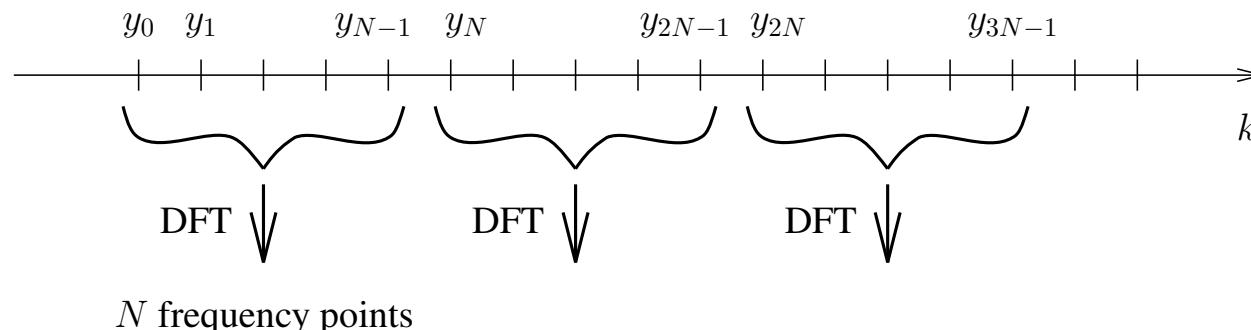
- $\begin{cases} h_0(t) \text{ centered around } t = 0 : \int_{-\infty}^{\infty} t |h_0(t)|^2 dt = 0 \\ H_0(f) \text{ centered around } f = 0 : \int_{-\infty}^{\infty} f |H_0(f)|^2 df = 0 \end{cases}$ // unbiased
- $\begin{cases} \text{precise time location} \Rightarrow h_0(t) \text{ narrow} \\ \text{precise frequency location} \Rightarrow H_0(f) \text{ narrow} \end{cases}$
time multiplication \Rightarrow frequency-domain convolution \Rightarrow smearing
- $\begin{cases} \text{effective duration: } D_{eff} = \sqrt{\frac{\int_{-\infty}^{\infty} t^2 |h_0(t)|^2 dt}{\int_{-\infty}^{\infty} |h_0(t)|^2 dt}} \\ \text{effective bandwidth: } B_{eff} = \sqrt{\frac{\int_{-\infty}^{\infty} f^2 |H_0(f)|^2 df}{\int_{-\infty}^{\infty} |H_0(f)|^2 df}} \end{cases}$ // standard deviation
- *uncertainty principle:* $D_{eff}B_{eff} \geq \frac{1}{4\pi}$

with = for $h_0(t) = e^{-\frac{1}{2}t^2}$: Gaussian window ($\Rightarrow H_0(f)$ Gaussian)

STFT with Gaussian $h_0(t)$: "Gabor representation" [1946]

Discrete-Time STFT

- sampling \Rightarrow time axis discretized
 $f \in [-\frac{1}{2}, \frac{1}{2}] \Rightarrow$ frequency more precise, but precision of Δt limited to the sampling period = 1, hence time is less precise
- uncertainty principle: cannot have high precision in both time and frequency
 $\Rightarrow \begin{cases} \text{sample frequency axis} \\ \text{further down-sample time axis} \end{cases}$
- we shall consider *non-redundant* time-frequency representations
- example: Discrete Fourier Transform (DFT):

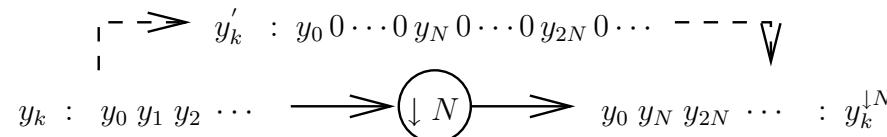


get N frequency points but time axis down-sampled by N
 \Rightarrow conservation of number of degrees of freedom (non-redundant)

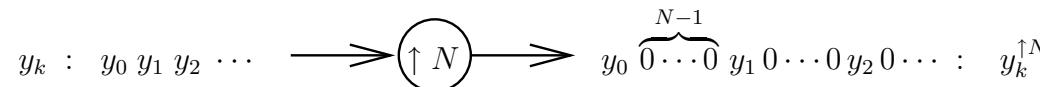
Filterbanks

- *downsampling* (decimation) and *upsampling* (interpolation) operations:

downsampling



upsampling



- upsampling transformation:

$$\mathbf{Y}^{\uparrow N}(z) = \sum_k y_k^{\uparrow N} z^{-k} = \sum_k y_k z^{-Nk} = \mathbf{Y}(z^N)$$

- downsampling transformation:

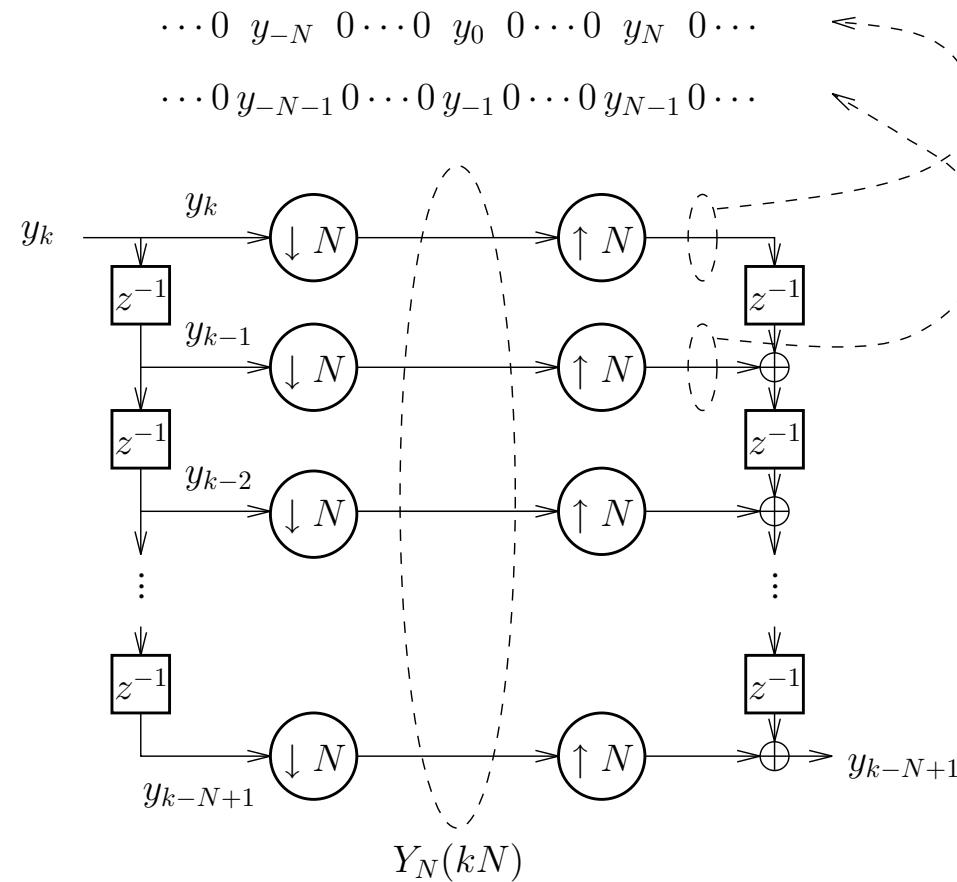
$$\diamondsuit \ y'_k = \begin{cases} y_k & , \frac{k}{N} \in \mathcal{Z} \\ 0 & , \text{otherwise} \end{cases} \Rightarrow \mathbf{Y}^{\downarrow N}(z) = \sum_k y_{Nk} z^{-k} = \sum_k y'_k z^{-k/N} = \mathbf{Y}'(z^{1/N})$$

$$\diamondsuit \ y'_k = c_k y_k, \ c_k = \begin{cases} 1 & , \frac{k}{N} \in \mathcal{Z} \\ 0 & , \text{otherwise} \end{cases}, \ c_k = \frac{1}{N} \sum_{n=0}^{N-1} w_N^{-nk}, \ w_N = e^{-j2\pi/N}$$

$$\diamondsuit \ \mathbf{Y}'(z) = \frac{1}{N} \sum_{n=0}^{N-1} \sum_k y_k w_N^{-nk} z^{-k} = \frac{1}{N} \sum_{n=0}^{N-1} \mathbf{Y}(z w_N^n) \Rightarrow \mathbf{Y}^{\downarrow N}(z) = \frac{1}{N} \sum_{n=0}^{N-1} \mathbf{Y}(z^{1/N} w_N^n)$$

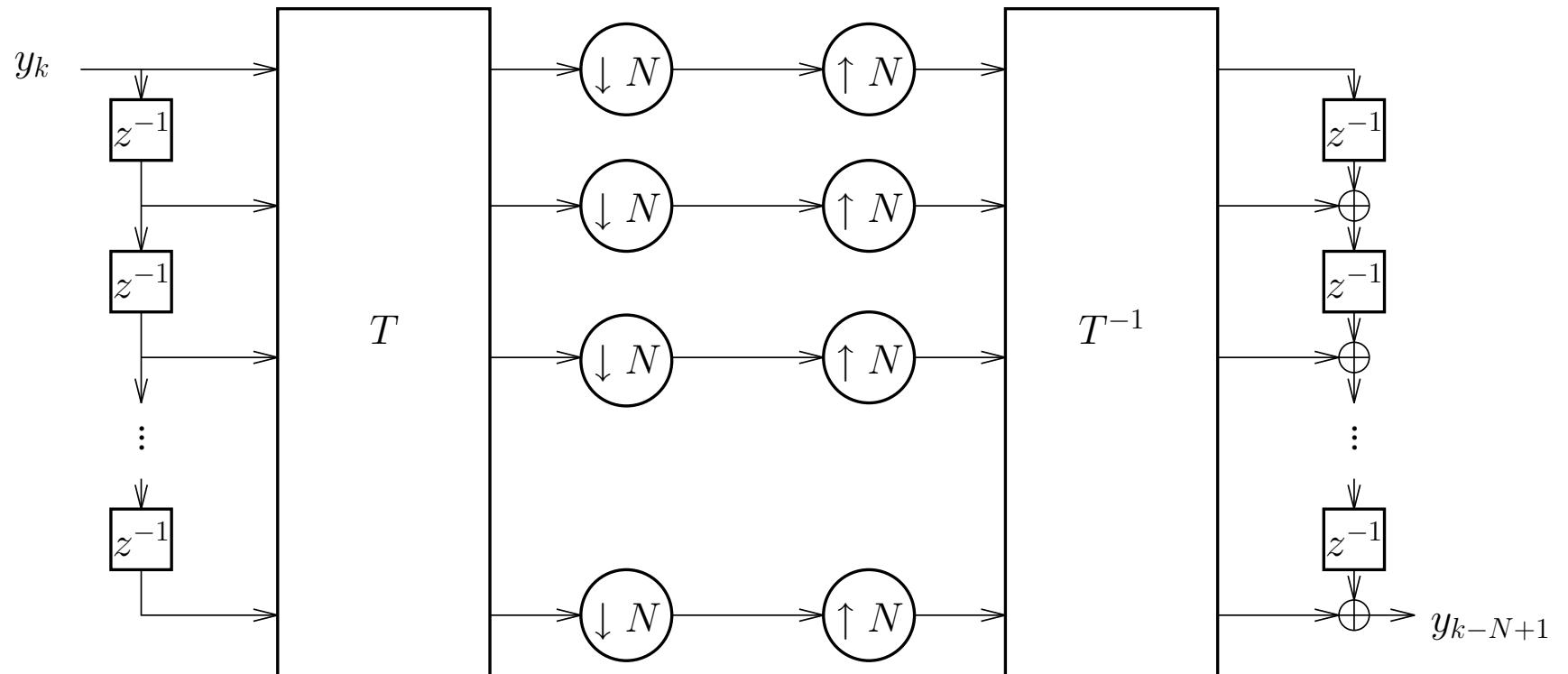
Filterbanks (2)

- *polyphase* representation of the pure delay $z^{-(N-1)}$



Filterbanks (3)

- Can insert $I_N = T^{-1}T$ in the middle. The instantaneous operations multiplication by T or by T^{-1} commute with the down/upsampling, so we get:



Filterbanks (4)

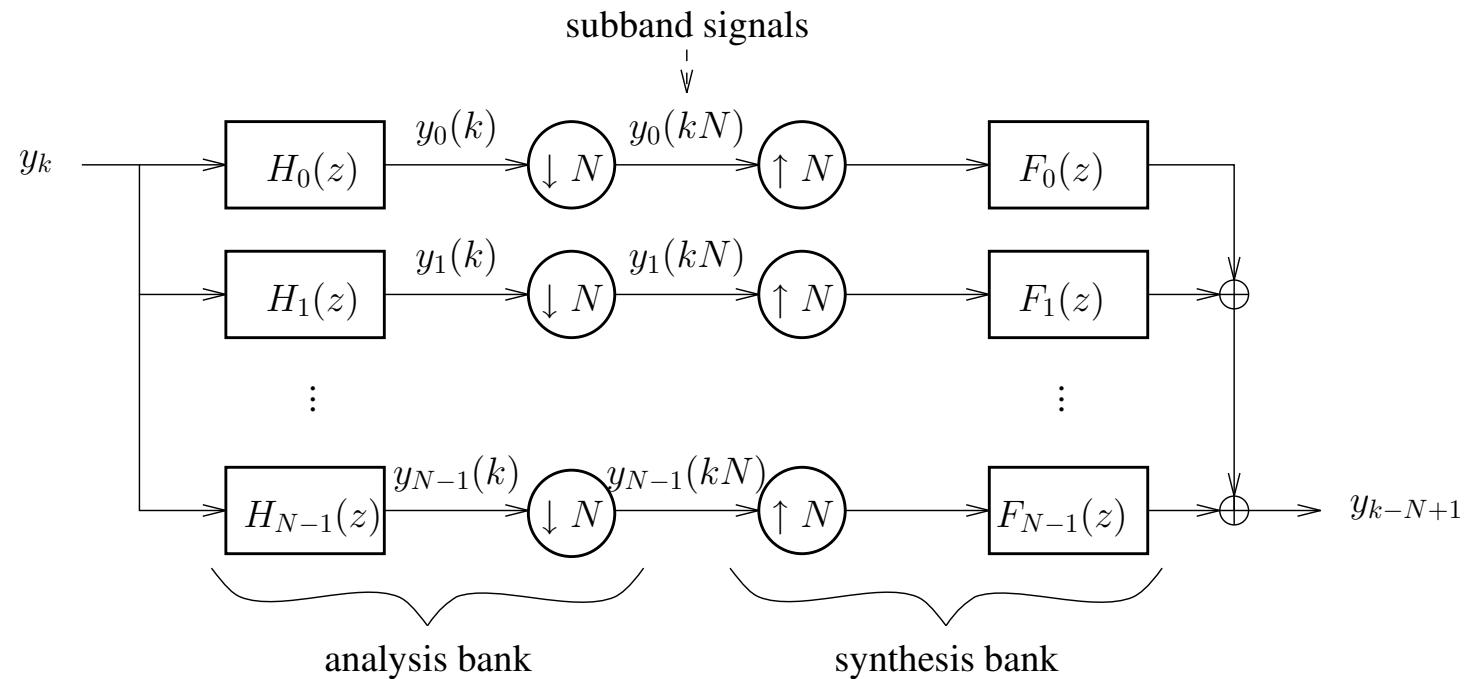
- can interpret the rows of T and the columns of T^{-1} as filters:

$$T = \begin{bmatrix} h_0(0) & h_0(1) & \cdots & h_0(N-1) \\ h_1(0) & h_1(1) & \cdots & h_1(N-1) \\ \vdots & \vdots & & \vdots \\ h_{N-1}(0) & h_{N-1}(1) & \cdots & h_{N-1}(N-1) \end{bmatrix}, \quad T^{-1} = \begin{bmatrix} f_0(N-1) & f_1(N-1) & \cdots & f_{N-1}(N-1) \\ \vdots & \vdots & & \vdots \\ f_0(1) & f_1(1) & \cdots & f_{N-1}(1) \\ f_0(0) & f_1(0) & \cdots & f_{N-1}(0) \end{bmatrix}$$

- can introduce analysis and synthesis filters

$$\left\{ \begin{array}{l} \mathbf{H}(z) = \begin{bmatrix} \mathbf{H}_0(z) \\ \mathbf{H}_1(z) \\ \vdots \\ \mathbf{H}_{N-1}(z) \end{bmatrix} = T \begin{bmatrix} 1 \\ z^{-1} \\ \vdots \\ z^{-(N-1)} \end{bmatrix} \\ \mathbf{F}(z) = [\mathbf{F}_0(z) \cdots \mathbf{F}_{N-1}(z)] = [z^{-(N-1)} \cdots z^{-1} 1] T^{-1} \end{array} \right.$$

Filterbanks (5)



filterbank characteristics:

- *critical subsampling*, subsampling factor = number of subbands, non-redundant
- *perfect reconstruction*: can do an exact inverse transform, the original signal can be recovered exactly from the subsampled subband signals
- *losslessness*: $T^{-1} = T^H = T^{*T}$ unitary T
conservation of energy: $\|TY_N(k)\|^2 = Y_N^T(k) \underbrace{T^H T}_{=I} Y_N(k) = \|Y_N(k)\|^2$
- $T^{-1} = T^H \Rightarrow f_n(k) = h_n^*(N-1-k), k, n = 0, 1, \dots, N-1$

Example 1: DFT

- DFT matrix W : $W_{km} = w_N^{(k-1)(m-1)}$, $w_N = e^{-j2\pi/N}$ $T = W^*$, $T^{-1} = \frac{1}{N}W$
- let $Y_{N,k}(f)$ = FT of $\{y_{k-N+1}, \dots, y_k\}$ of finite duration,
hence $\{Y_{N,k}(n/N), n = 0, 1, \dots, N-1\}$ = its DFT
- subband signals interpretation:

$$\begin{aligned} y_n(k) &= \sum_{i=0}^{N-1} h_n(i)y_{k-i} = \sum_{i=0}^{N-1} w_N^{-ni} y_{k-i} = \sum_{i=0}^{N-1} w_N^{-ni} y_{k-N+1+\underbrace{N-1-i}_{=m}} \\ &= \sum_{m=0}^{N-1} w_N^{mn} \underbrace{w_N^{-nN}}_{=1} w_N^n y_{k-N+1+m} = w_N^n \sum_{m=0}^{N-1} w_N^{mn} y_{k-N+1+m} = w_N^n Y_{N,k}(n/N) \end{aligned}$$

⇒ filterbank = running spectrum analyzer,
trade-off: frequency resolution vs. time resolution

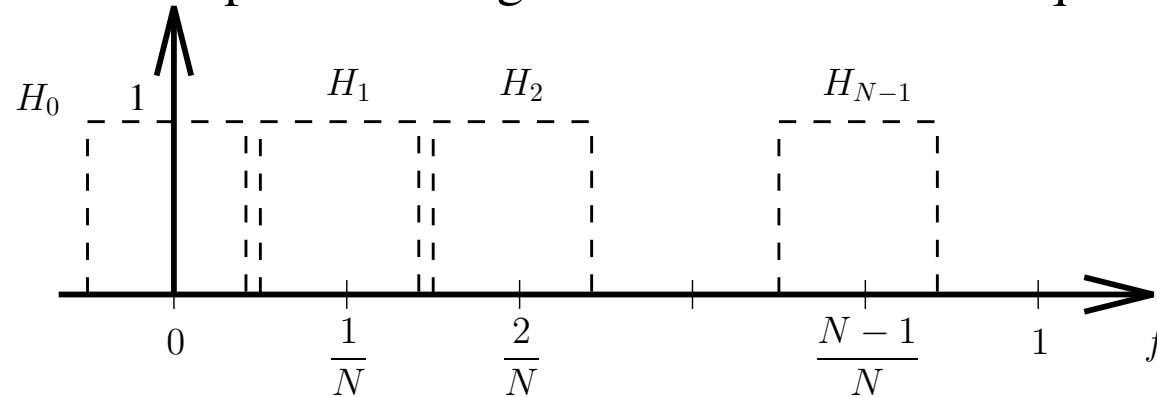
- $H_0(z) = 1+z^{-1}+\dots+z^{-N+1}$ or $|H_0(f)| = \left| \frac{\sin N\pi f}{\sin \pi f} \right|$:
low-pass filter corresponding to a rectangular window $h_0(k)$
- $H_n(z) = H_0(z w_N^n)$: *modulated filterbank*,
 $H_n(f) = H_0(f - \frac{n}{N})$: frequency translation

More General Filterbanks

- so far: analysis and synthesis filters FIR of length N = number of subbands = subsampling factor
- more general: T and T^{-1} get replaced by $\mathbf{E}(z)$ and $\mathbf{R}(z)$ resp. ($N \times N$)
- perfect reconstruction: $\mathbf{R}(z)\mathbf{E}(z) = I_N \Rightarrow \mathbf{R}(z) = \mathbf{E}^{-1}(z)$
- after inserting $\mathbf{R}(z)\mathbf{E}(z) = I_N$ into the polyphase decomposition of $z^{-(N-1)}$, $\mathbf{R}(z)$ and $\mathbf{E}(z)$ becomes $\mathbf{R}(z^N)$ and $\mathbf{E}(z^N)$ resp. after moving them across the downsampling and upsampling operations
- losslessness: $\mathbf{R}(z) = \mathbf{E}^\dagger(z) = \mathbf{E}^H(1/z^*)$: $\mathbf{E}(z)$ paraunitary: $\mathbf{E}^\dagger(z)\mathbf{E}(z) = I$
- analysis filter bank: $\mathbf{H}(z) = \begin{bmatrix} \mathbf{H}_0(z) \\ \vdots \\ \mathbf{H}_{N-1}(z) \end{bmatrix} = \mathbf{E}(z^N) \begin{bmatrix} 1 \\ \vdots \\ z^{-(N-1)} \end{bmatrix}$
- synthesis filter bank: $\mathbf{F}(z) = [\mathbf{F}_0(z) \cdots \mathbf{F}_{N-1}(z)] = [z^{-(N-1)} \cdots z^{-1} \ 1] \mathbf{R}(z^N)$
- lossless case: $\mathbf{H}^\dagger(z)\mathbf{H}(z) = N \Rightarrow$ sum of the variances in the N subbands is $N\sigma_y^2$ per N sample periods, hence σ_y^2 per sample period \Rightarrow power conservation
- further generalizations: non-uniform filter banks, not critically subsampled, not lossless (biorthogonal), linear phase, IIR, ...

Example 2: Brickwall Filterbank

- dual to the DFT example: rectangular filters in the frequency domain:



- subband signal $y_n(k)$ perfectly bandlimited with bandwidth $\frac{1}{N}$, hence the Nyquist theorem allows for downsampling with a factor N without loss of information, can still perfectly reconstruct
- again modulated filterbank: $H_n(f) = H_0(f - \frac{n}{N})$

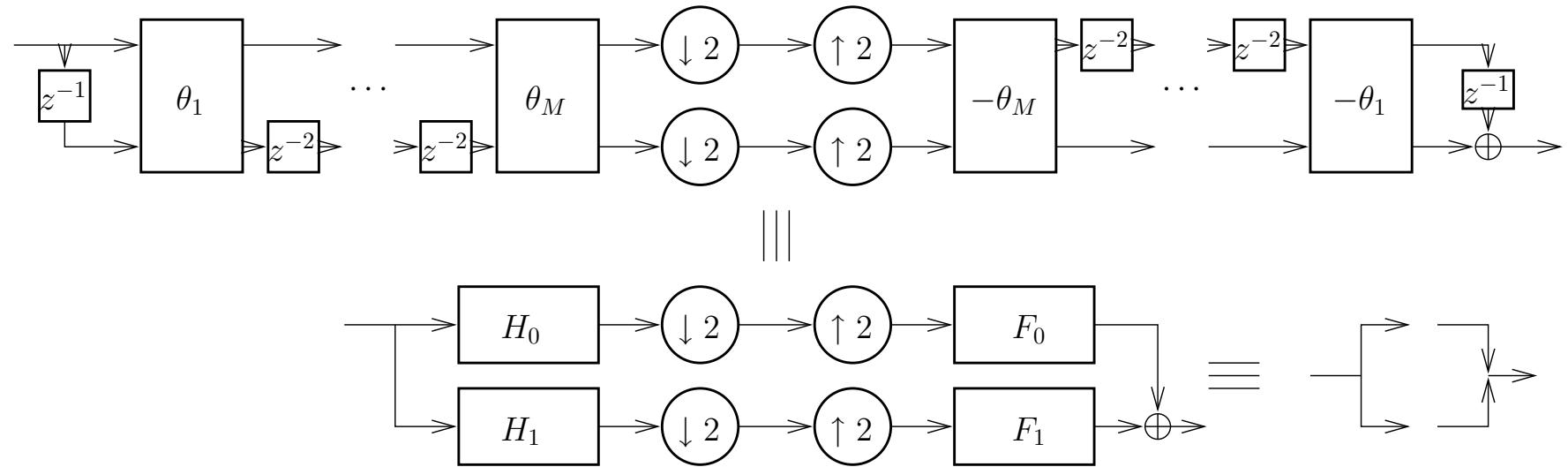
$$h_0(k) = \frac{\sin \frac{\pi k}{N}}{\pi k} \text{ non-causal and of infinite duration}$$

$N = 2$: Quadrature Mirror Filterbanks (QMF)

- general form of lossless FIR two-band filterbank:

$$E(z) = \Theta(\theta_M) \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \cdots \Theta(\theta_2) \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \Theta(\theta_1), \quad \Theta(\theta) = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix}$$

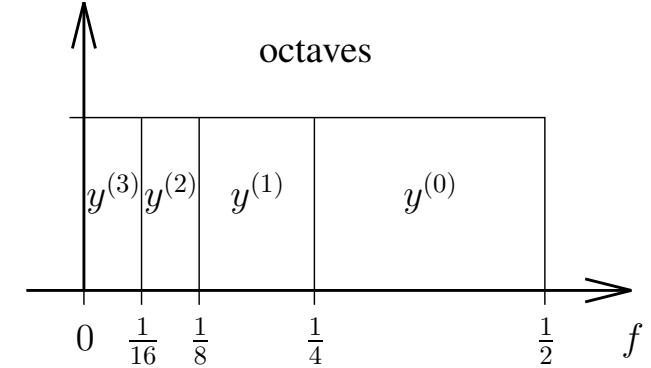
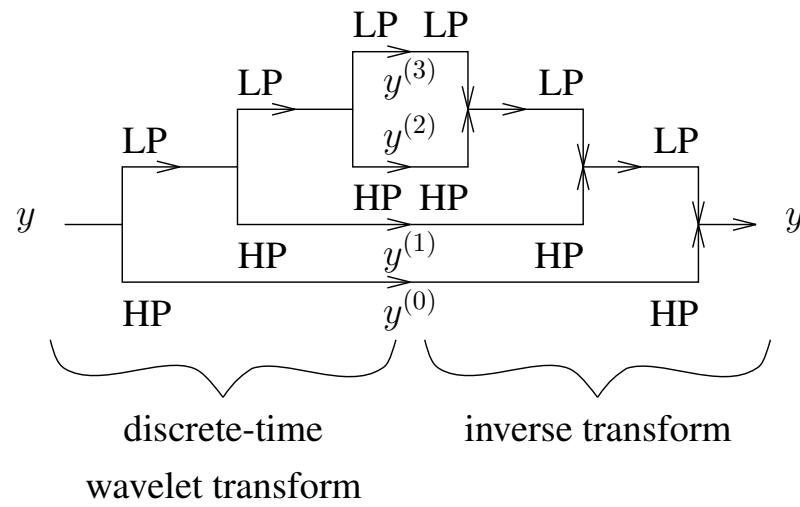
cascade of elementary paraunitary factors



- a delay has been introduced in the synthesis bank to make it causal

Discrete-Time Wavelet Transform (DTWT)

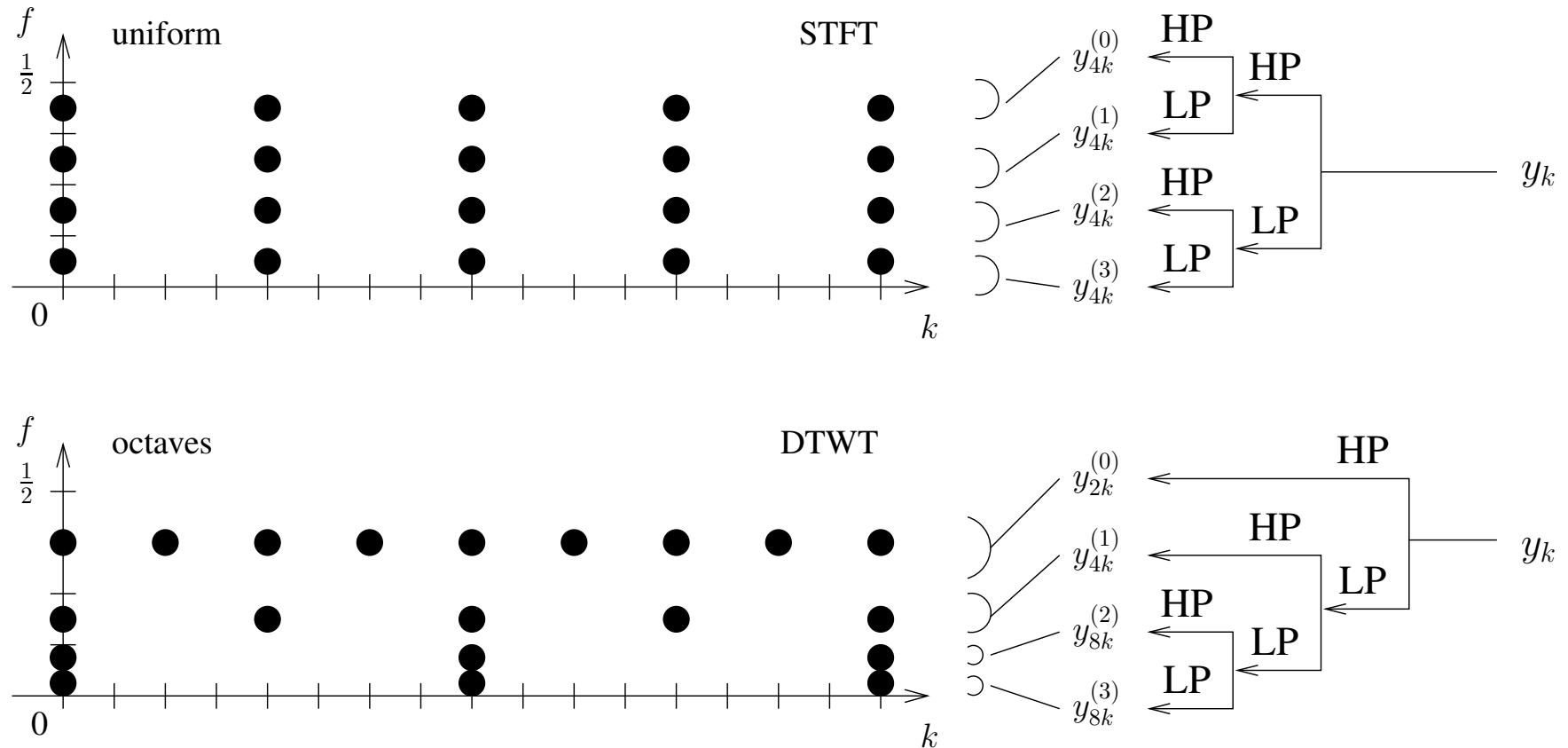
- so far: uniform filterbanks (e.g. discrete STFTF): frequency resolution constant as a function of frequency
- discrete-time wavelet transform: tree-structured two-band filterbank, leads to frequency analysis in octaves



- H_1 : high-pass filter, H_0 : low-pass filter

Discrete-Time Wavelet Transform (2)

- time-frequency sampling patterns of STFT and DTWT:



Discrete-Time Wavelet Transform (3)

- wavelet transform:
 - { → high frequency resolution at low frequencies
 - high temporal resolution at high frequencies
- very well adapted to { → image coding (see source coding chapter)
 - image analysis (// eye)
 - sound analysis (// ear)
- logarithmic frequency = *scale*
- { high frequency components = *detail signals*
 - low frequency components → approximation of signal at a lower scale (as if standing further away ⇒ looks more blurred)

⇒ *multiresolution analysis*:
different scales ↔ different levels of resolution
- hierarchical signal reconstruction by adding detail signals at higher and higher levels of resolution



Decorrelating Transformations (8)

choice of transformation for various source coding applications

- linear prediction
 - speech coding: because of the connection between linear prediction and autoregressive modeling and the fact that speech signals can be approximated well with autoregressive models
 - low complexity solutions for image and video coding
- DCT-based transform coding
 - image and video coding
 - artefacts appearing at low bit rate: blocking (discontinuities at the borders between consecutive image blocks that get transformed)
- subband coding
 - image and video coding
 - artefacts appearing at low bit rate: ringing (large quantization errors at edges in the image get convolved with the synthesis bank impulse response)
 - audio coding (large bandwidth)
 - limitations of hearing that can be expressed in the frequency domain can be incorporated in the coding process



Decorrelating Transformations (9)

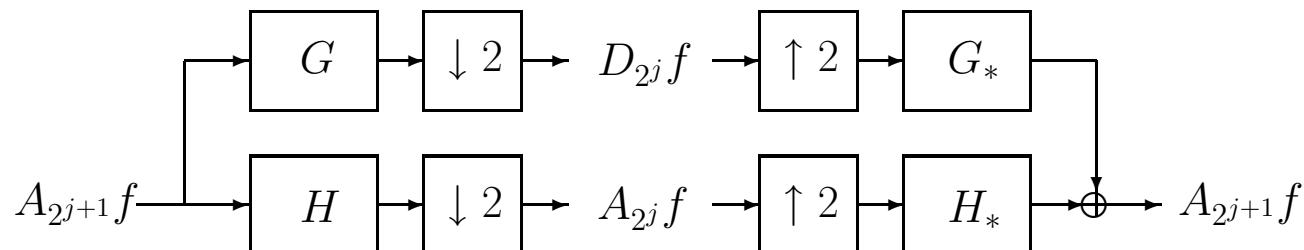
advantages of the Wavelet Transform (WT) for image coding :

- *perfect reconstruction* is possible
- the WT is a *multiresolution* description
 - ⇒ progressive decoding from lowest to highest resolution
- The WT is closer to the human visual system than the DCT transform
 - ⇒ artefacts introduced by WT coding are less annoying than those introduced by DCT coding (esp. at a high compression ratio)
- wavelet transform ⇒ *scale-space representation*:
 - high frequency components precisely located in the pixel domain
 - spatial resolution of the WT increases linearly with frequency
 - ⇒ allows the WT to *zoom* into the strong high frequency components of sharp edges and locate them accurately
 - low frequency components precisely located in frequency domain
 - overall spectrum of most images very much of a low-pass type
 - the frequency resolution is inversely proportional to frequency in the WT
 - ⇒ high decorrelation



Wavelet Transforms and Scale-Space Analysis

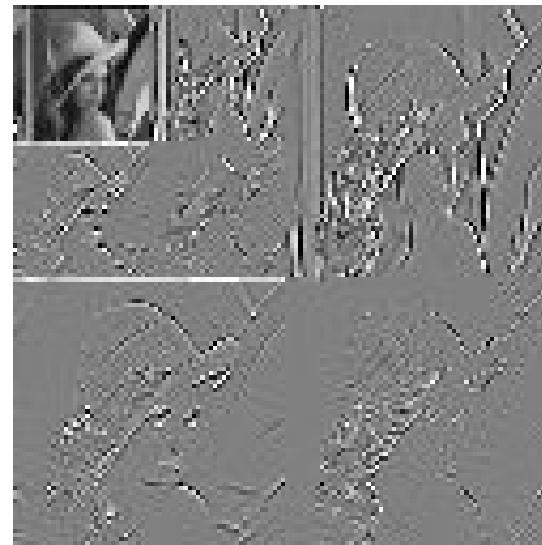
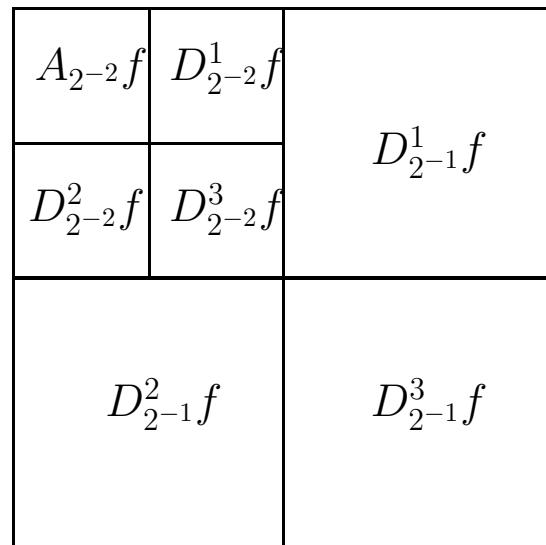
- the approximation $A_{2^j}f$ of a signal f at a resolution 2^j (estimate of f derived from 2^j measurements per unit length): computed by uniformly sampling at the rate 2^j the signal f smoothed by a low-pass filter whose bandwidth is proportional to 2^j . The approximation at resolution 2^j can be computed from the approximation at resolution 2^{j+1} by filtering it with a low-pass digital filter H and subsampling the output by a factor two.
- to extract the details of f which appear in $A_{2^{j+1}}f$ but not in $A_{2^j}f$: the *discrete detail* signal $D_{2^j}f$ at the resolution 2^j can be obtained by passing $A_{2^{j+1}}f$ through a high-pass filter G and subsampling the output by a factor two.
 $D_{2^j}f$ can be obtained by uniformly sampling at the rate 2^j the continuous-time signal f passed through an analog band-pass filter whose bandwidth is proportional to 2^j . This filter is usually referred to as a *wavelet*, and the mapping from the signal f to its discrete details is called the *discrete wavelet transform*
- Conversely, the approximation signal at the resolution 2^{j+1} can be recovered from the approximation and detail signals at the resolution 2^j by means of the “dual filtering system”.





2D Wavelet Transform and Image Coding

- Consider the *separable* extension from 1D to 2D: the 2D impulse responses are obtained as the convolution of two 1D impulse responses, one for each of the horizontal and vertical directions. We first apply the 1D approach to the rows of the image $A_1 f$, yielding two subimages. Then we run the 1D QMF filters on the columns of these two subimages, yielding four subimages, each being one fourth of the original image in size: $A_{2-1} f$ is the low-pass approximation in both directions, on which the decomposition process can be repeated, $D_{2-1}^1 f$ gives the vertical higher frequencies (horizontal edges), $D_{2-1}^2 f$ gives the horizontal higher frequencies (vertical edges), $D_{2-1}^3 f$ gives the higher frequencies in both directions (corners).





Statistical Signal Processing

Lecture 8

chapter 2: Spectrum Estimation

- AR modeling motivations: LP of an AR(N) process, asymptotics
- AR modeling interpretations, techniques, model order selection

chapter 3: Optimal Filtering

Wiener filtering

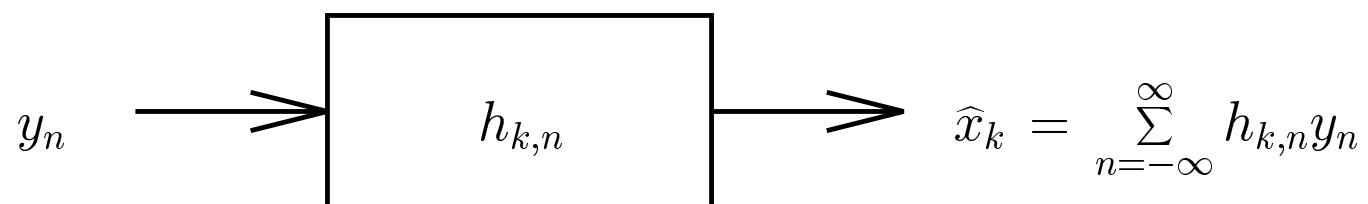
- non-causal Wiener filtering
- signal in noise case
- equalization
- causal Wiener filtering

Noncausal Wiener Filtering

- parameter estimation: estimation of a finite number of random/deterministic parameters given (stochastically) related measurements
- spectrum estimation: estimation of an infinite (nonparametric)/finite (parametric) number of deterministic parameters
- here: estimation of an unknown *random process* $\{x_k, k \in \mathcal{Z}\}$ given a (cor-)related random process $\{y_k, k \in \mathcal{Z}\}$
- can be reduced to the estimation of a random parameter x_k :

$$\{y_n, n \in \mathcal{Z}\} \rightarrow \hat{x}_k$$

- linear estimator = filter:



Noncausal Wiener Filtering (2)

- determine the filter coefficients $h_{k,n}$ from the following LMMSE estimation problem

$$\min_{h_{k,n}} E(x_k - \hat{x}_k)^2 = \min_{h_{k,n}} E(x_k - \sum_{n=-\infty}^{\infty} h_{k,n} y_n)^2$$

- the optimal $h_{k,n}$ satisfy the LMMSE orthogonality conditions

$$E(x_k - \hat{x}_k)y_m = 0, \quad \forall m \in \mathcal{Z} \Rightarrow E\hat{x}_ky_m = \sum_{n=-\infty}^{\infty} h_{k,n} E y_n y_m = E x_k y_m, \quad \forall m \in \mathcal{Z}$$

- assume $\{x_k, k \in \mathcal{Z}\}$ and $\{y_k, k \in \mathcal{Z}\}$ to be jointly (wide sense) stationary (and zero mean): orthogonality conditions \rightarrow normal equations

$$\sum_{n=-\infty}^{\infty} h_{k,n} r_{yy}(n-m) = r_{xy}(k-m), \quad \forall m \in \mathcal{Z}$$

- substitute $k-m \rightarrow m$ and $n-k \rightarrow -n$

$$\sum_{n=-\infty}^{\infty} h_{k,n} r_{yy}(n-k+m) = r_{xy}(m), \quad \sum_{n=-\infty}^{\infty} h_{k,k-n} r_{yy}(m-n) = r_{xy}(m), \quad \forall m \in \mathcal{Z}$$

- solution for the $h_{k,k-n}$ is the same for any $k \Rightarrow h_{k,k-n} = h_{0,-n} = h_n$: stationarity \Rightarrow the optimal linear filter is time-invariant

Noncausal Wiener Filtering (3)

- the optimal linear time-invariant filter satisfies

$$\sum_{n=-\infty}^{\infty} h_n r_{yy}(m-n) = r_{xy}(m), \quad \forall m \in \mathcal{Z}$$

This is an infinite set of equations in an infinite number of unknowns h_n .

- convolution \Rightarrow take z -transform to obtain a simple product. Let

$$\mathbf{S}_{xy}(z) = \sum_{m=-\infty}^{\infty} r_{xy}(m)z^{-m}, \quad \mathbf{H}(z) = \sum_{m=-\infty}^{\infty} h_m z^{-m}$$

then

$$\mathbf{H}(z)\mathbf{S}_{yy}(z) = \mathbf{S}_{xy}(z) \quad \Rightarrow \quad \mathbf{H}(z) = \frac{\mathbf{S}_{xy}(z)}{\mathbf{S}_{yy}(z)}$$

Frequency Domain Interpretation

- the Fourier transform of $\widehat{x}_k = \sum_{m=-\infty}^{\infty} h_m y_{k-m}$ is
(the Fourier transform is the z -transform evaluated at $z = e^{j2\pi f}$)

$$\widehat{X}(f) = H(f) Y(f) , \quad H(f) = \frac{S_{xy}(f)}{S_{yy}(f)}$$

where $Y(f) = \mathbf{Y}(e^{j2\pi f}) = \sum_{m=-\infty}^{\infty} y_m e^{-j2\pi fm}$ etc.

- this resembles the scalar LMMSE problem:

$$\widehat{x} = h y , \quad h = \frac{R_{xy}}{R_{yy}} , \quad \widehat{X}(f) = H(f) Y(f)$$

- can show:

$$H(f) = \frac{R_{X(f)Y(f)}}{R_{Y(f)Y(f)}} = \frac{S_{xy}(f)}{S_{yy}(f)}$$

- Wiener filtering of one random process from another is like a scalar LMMSE estimation of the Fourier transforms of those processes at every frequency.

MMSE Expressions

- The orthogonality property of the LMMSE estimator implies

$$E(x_k - \hat{x}_k)\hat{x}_k = \sum_{n=-\infty}^{\infty} h_n \underbrace{E(x_k - \hat{x}_k)y_{k-n}}_{=0} = 0 \Rightarrow E x_k \hat{x}_k = E \hat{x}_k^2$$

This allows us to demonstrate the following Pythagorean property

$$\begin{aligned} \text{MMSE} &= E \tilde{x}_k^2 = E(x_k - \hat{x}_k)^2 = E(x_k - \hat{x}_k)x_k - \underbrace{E(x_k - \hat{x}_k)\hat{x}_k}_{=0} \\ &= E x_k^2 - E x_k \hat{x}_k = E x_k^2 - \underbrace{E \hat{x}_k^2}_{\geq 0} \leq E x_k^2 \end{aligned}$$

$E x_k^2 = r_{xx}(0)$ is the MSE if we have no observations. In that case, $\hat{x}_k = E x_k = 0$ is our best estimator, leading indeed to $E x_k^2$ as MSE. $E \hat{x}_k^2 = r_{\hat{x}\hat{x}}(0) \geq 0$ is the reduction in MSE by estimating x_k using the Wiener filter on the data $\{y_n, n \in \mathcal{Z}\}$.

- analyzing the MMSE in the frequency domain:

$$\text{MMSE} = r_{xx}(0) - r_{\hat{x}\hat{x}}(0) = \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{xx}(f) df - \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{\hat{x}\hat{x}}(f) df$$

- Since $\hat{x}_k = h_k * y_k$, we have for the power spectral density functions

$$S_{\hat{x}\hat{x}}(f) = |H(f)|^2 S_{yy}(f) = |S_{xy}(f)|^2 / S_{yy}(f)$$

MMSE Expressions in the Frequency Domain

- Let us introduce the *cross-power spectral density coefficient*

$$\rho_{xy}(f) = \frac{S_{xy}(f)}{\sqrt{S_{xx}(f)S_{yy}(f)}}$$

which is defined as zero whenever $S_{xx}(f) = 0$ or $S_{yy}(f) = 0$.

- $\Rightarrow \text{MMSE} = r_{\tilde{x}\tilde{x}}(0) = \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{\tilde{x}\tilde{x}}(f) df = \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{xx}(f) [1 - |\rho_{xy}(f)|^2] df \quad (*)$

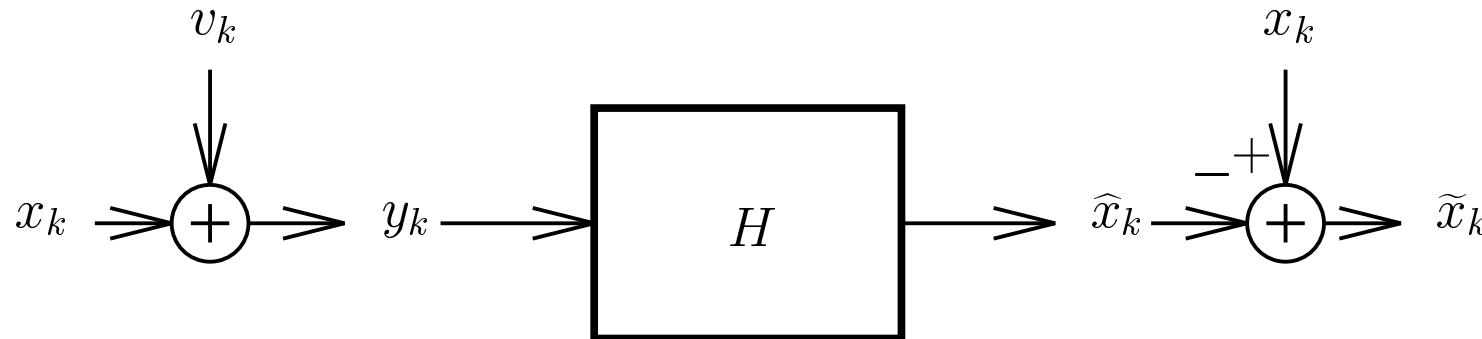
In particular we find that

$$1 - |\rho_{xy}(f)|^2 = \frac{S_{\tilde{x}\tilde{x}}(f)}{S_{xx}(f)} \geq 0 \Rightarrow |\rho_{xy}(f)| \leq 1$$

$\rho_{xy}(f)$ is normalized and can be interpreted as the correlation coefficient between $X(f)$ and $Y(f)$.

- Since now $1 - |\rho_{xy}(f)|^2 \in [0, 1]$, $(*)$ shows how the power spectral density of x_k gets attenuated as a function of frequency to obtain the power spectral density of the estimation error \tilde{x}_k . At frequencies where $X(f)$ and $Y(f)$ are strongly correlated, $S_{\tilde{x}\tilde{x}}(f)$ will be significantly reduced w.r.t. $S_{xx}(f)$ whereas this will not be the case at frequencies where $X(f)$ and $Y(f)$ are hardly correlated.

Signal in Noise



- case of $y_k = x_k + v_k$
- The quantities that determine the Wiener filter are

$$r_{xy}(n) = r_{xx}(n) + \underbrace{r_{xv}(n)}_{=0} = r_{xx}(n) , \quad S_{xy}(z) = S_{xx}(z)$$

$$\begin{aligned} r_{yy}(n) &= r_{xx}(n) + \underbrace{r_{xv}(n)}_{=0} + \underbrace{r_{vx}(n)}_{=0} + r_{vv}(n) \\ &= r_{xx}(n) + r_{vv}(n) , \quad S_{yy}(z) = S_{xx}(z) + S_{vv}(z) \end{aligned}$$

- Wiener filter depends on SNR as a function of frequency

$$H(f) = \frac{S_{xx}(f)}{S_{xx}(f) + S_{vv}(f)} \in [0, 1] \text{ weighting filter}$$

Signal in Noise (2)

- can show

$$\frac{1}{S_{\tilde{x}\tilde{x}}(f)} = \frac{1}{S_{xx}(f)} + \frac{1}{S_{vv}(f)} \geq \max\left\{\frac{1}{S_{xx}(f)}, \frac{1}{S_{vv}(f)}\right\}$$

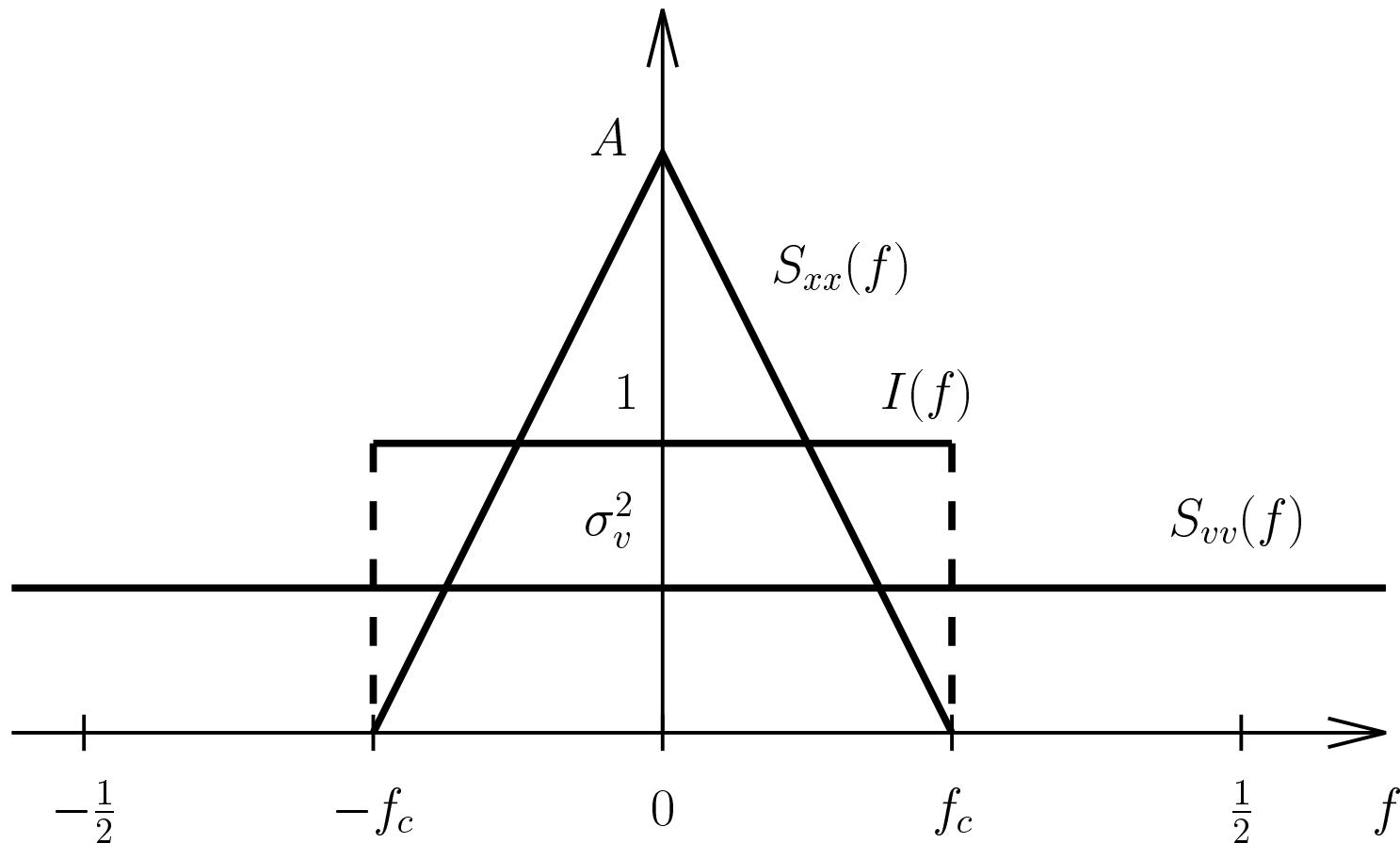
- and for the MMSE

$$\begin{aligned} \text{MMSE} = E \tilde{x}_k^2 &= \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{\tilde{x}\tilde{x}}(f) df = \int_{-\frac{1}{2}}^{\frac{1}{2}} \frac{S_{xx}(f) S_{vv}(f)}{S_{xx}(f) + S_{vv}(f)} df \\ &= \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{xx}(f) \underbrace{\frac{S_{vv}(f)}{S_{xx}(f) + S_{vv}(f)}}_{0 \leq \cdot \leq 1} df = \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{vv}(f) \underbrace{\frac{S_{xx}(f)}{S_{xx}(f) + S_{vv}(f)}}_{0 \leq \cdot \leq 1} df \\ &\leq \min \left\{ \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{xx}(f) df, \int_{-\frac{1}{2}}^{\frac{1}{2}} S_{vv}(f) df \right\} = \min \left\{ \underbrace{E x_k^2}_{\hat{x}_k=0}, \underbrace{E v_k^2}_{\hat{x}_k=y_k} \right\} \end{aligned}$$

and also

$$\text{MMSE} \leq \int_{-\frac{1}{2}}^{\frac{1}{2}} \min \{S_{xx}(f), S_{vv}(f)\} df \leq \min \{E x_k^2, E v_k^2\}$$

Signal in Noise: Example



- example: Bandlimited Random Process in White Noise
- x_k with a triangular power spectral density function

$$S_{xx}(f) = \begin{cases} A(1 - \frac{|f|}{f_c}) & , |f| \leq f_c \\ 0 & , f_c \leq |f| \leq \frac{1}{2} \end{cases}$$

Signal in Noise: Example (2)

- We receive $y_k = x_k + v_k$. The additive noise v_k is white with variance σ_v^2 : $S_{vv}(f) = \sigma_v^2$.
- The task is to filter y_k so that the filter output \hat{x}_k approximates x_k well.
- Classical (non-statistical) filter design is based on the notion of distortion: pass x_k without distortion but for the rest cut out the noise as much as possible. Since x_k is bandlimited, we can choose an ideal low-pass filter $I(f)$ matched to the bandwidth f_c of x_k :

$$I(f) = \begin{cases} 1 & , |f| \leq f_c \\ 0 & , f_c < |f| \leq \frac{1}{2} \end{cases}$$

The output $\hat{x}_k(I)$ of the filter $I(f)$ will be equal to x_k minus an error $\tilde{x}_k(I)$ which is a low-pass filtered version of v_k . Hence the variance of the error is

$$E \tilde{x}_k^2(I) = \int_{-f_c}^{f_c} S_{vv}(f) df = 2\sigma_v^2 f_c$$

Since $f_c \leq 0.5$, $E \tilde{x}_k^2(I) \leq E v_k^2$. Hence, the filtering operation with $I(f)$ has reduced the noise level w.r.t. the measurement y_k while leaving the signal component x_k undistorted.

Signal in Noise: Example (3)

- Wiener approach: trades some signal distortion for a further reduction in overall error variance:

$$H(f) = \frac{S_{xx}(f)}{S_{xx}(f) + S_{vv}(f)} = \begin{cases} \frac{A}{\sigma_v^2} \frac{1 - \frac{|f|}{f_c}}{1 + \frac{A}{\sigma_v^2} \left(1 - \frac{|f|}{f_c}\right)} , & |f| \leq f_c \\ 0 , & f_c < |f| \leq \frac{1}{2} \end{cases}$$

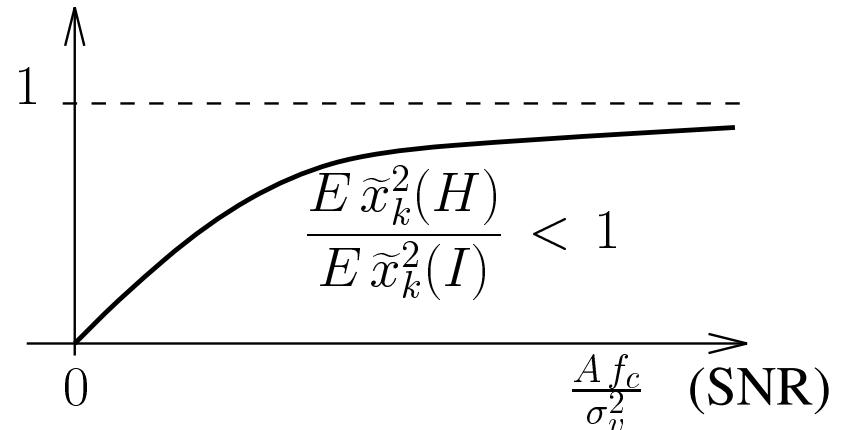
We can analyze the nature of the optimal filter at high or low signal-to-noise ratio (SNR):

$$\text{low SNR: } \frac{A}{\sigma_v^2} \rightarrow 0 : H(f) \rightarrow \frac{S_{xx}(f)}{S_{vv}(f)} = \begin{cases} \frac{A}{\sigma_v^2} \left(1 - \frac{|f|}{f_c}\right) , & |f| \leq f_c \\ 0 , & f_c < |f| \leq \frac{1}{2} \end{cases}$$

$$\text{high SNR: } \frac{A}{\sigma_v^2} \rightarrow \infty : H(f) \rightarrow I(f)$$

So for low SNR, the filter $H(f)$ becomes proportional to the ratio of the psdf's of the signal of interest and the noise. For high SNR, the filter $H(f)$ approaches the classical distortion-based design, passing perfectly all frequencies where the signal of interest is present.

Signal in Noise: Example (4)



- We find for the MMSE:

$$\text{MMSE} = E \tilde{x}_k^2(H) = \left[1 - \frac{\sigma_v^2}{A} \ln\left(1 + \frac{A}{\sigma_v^2}\right) \right] E \tilde{x}_k^2(I)$$

We can again analyze the limiting behavior for high or low SNR:

$$\text{low SNR: } \frac{A}{\sigma_v^2} \rightarrow 0 : E \tilde{x}_k^2(H) \rightarrow E x_k^2 = A f_c$$

$$\text{high SNR: } \frac{A}{\sigma_v^2} \rightarrow \infty : E \tilde{x}_k^2(H) \rightarrow E \tilde{x}_k^2(I) = 2f_c \sigma_v^2$$

- high SNR: optimal filter // classical distortion criterion based design: the variance of the error is the variance of the noise in the signal band.
- low SNR: optimal filter works much better than the classical one. Indeed, the variance of the error becomes equal to the signal variance, even though the noise level is much higher! It is true though that at low SNR, the performance of even the optimal filter is not very good in this example.

Channel Equalization

- digital communications: $y_k = \mathbf{C}(q) x_k + v_k$.
 x_k : symbols, y_k : received signal, v_k : additive noise
 $\mathbf{C}(z) = \sum c_k z^{-k}$: channel (cascade of transmission (pulse shaping) filter, actual channel and receiver filter)
sampling at the symbol rate
- assume: x_k and v_k independent white stationary sequences with zero mean and variances σ_x^2 , σ_v^2 . We also assume here that all signals involved are real and scalar.
- The x_k have a discrete distribution and take on values in a finite alphabet. The problem of deciding on the basis of the y_k which discrete values x_k have been sent is called the *detection* problem.
- If the channel impulse response c_k has only one non-zero sample, then the optimal detection can be done instantaneously since the noise samples v_k are independent. This means that if w.l.o.g. we consider c_0 to be the non-zero sample, x_k can be detected from the sample

$$y_k = c_0 x_k + v_k .$$

Channel Equalization

- If c_k contains more than one non-zero sample, then the symbols appear superimposed in the received signal y_k : *intersymbol interference* (ISI). The problem of detecting the symbols in the presence of ISI is called *equalization*. Optimal (maximum-likelihood) equalization can be done for FIR channels using the so-called Viterbi algorithm (which corresponds to dynamic programming).
- performance of Viterbi: bound by the Matched Filter Bound (MFB). The MFB expresses the maximum SNR achievable for the detection of a certain symbol x_k assuming that all other symbols have been correctly detected. This means that all other symbols are known so that their contribution can be subtracted from the received signal y_k . If we assume w.l.o.g. that the symbol we want to detect is x_0 , then the resulting received signal with the contributions of the $x_k, k \neq 0$, removed is

$$y_k = c_k x_0 + v_k \longrightarrow Y = C x_0 + V .$$

If the white noise v_k is Gaussian, then it turns out that the optimal detection of x_0 can be done on the basis of the unconstrained ML estimate

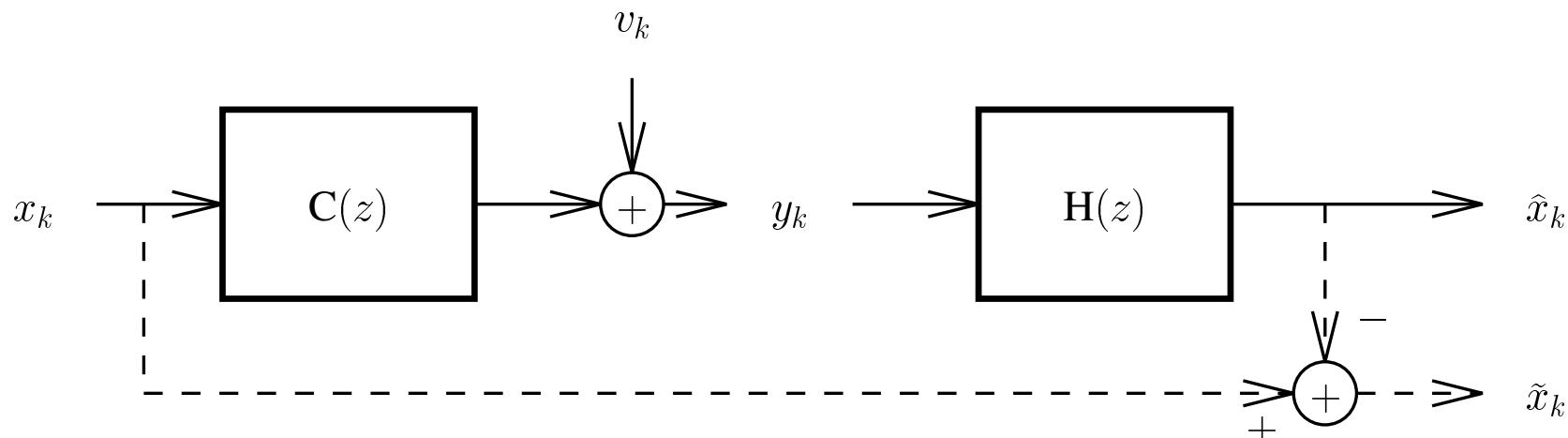
$$\hat{x}_0^{ML} = \frac{\sum_k c_k y_k}{\sum_k c_k^2} = (C^T C)^{-1} C^T Y . \quad \text{MFB} = \frac{\sigma_x^2}{\sigma_{\tilde{x}}^2} = \frac{\sigma_x^2}{\sigma_v^2 / \|C\|^2} = \frac{\sigma_x^2}{\sigma_v^2} \|C\|^2$$

Channel Equalization

- other interpretation: Consider filtering the y_k with a filter $H(z)$ to obtain the signal \hat{x}_k and consider in particular the output at time 0, \hat{x}_0 . The part of \hat{x}_0 due to x_0 is called the signal part while the part due to the v_k is called the noise part.
- the filter $H(z)$ that maximizes the SNR in \hat{x}_0 is the *matched filter* $H(z) = C^\dagger(z) = C(1/z)$ ($= C^H(1/z^*)$ in general), matched to the channel $C(z)$. The SNR at the output of the matched filter is the MFB

$$\text{MFB} = \frac{\sigma_x^2}{\sigma_v^2} \frac{1}{2\pi j} \oint \frac{dz}{z} C^\dagger(z) C(z) = \frac{\sigma_x^2}{\sigma_v^2} \int_{-\frac{1}{2}}^{\frac{1}{2}} df C^*(f) C(f) = \frac{\sigma_x^2}{\sigma_v^2} \sum_k |c_k|^2 = \frac{\sigma_x^2}{\sigma_v^2} \|C\|^2$$

The MFB is proportional to the energy in the channel response. (The term MFB is also often used to denote the corresponding probability of error in the detection of \hat{x}_0^{ML}).



Simpler Equalizers

- MFB optimization problem: $\hat{x}_0 = x_0 \frac{1}{2\pi j} \oint \frac{dz}{z} \mathbf{H}(z) \mathbf{C}(z) + \mathbf{H}(q) v_k|_{k=0}$

$$\text{SNR} = \frac{\sigma_x^2}{\sigma_v^2} \frac{\left| \frac{1}{2\pi j} \oint \frac{dz}{z} \mathbf{H}(z) \mathbf{C}(z) \right|^2}{\frac{1}{2\pi j} \oint \frac{dz}{z} \mathbf{H}(z) \mathbf{H}^\dagger(z)} = \frac{\sigma_x^2}{\sigma_v^2} \frac{\langle \mathbf{H}^\dagger(\cdot), \mathbf{C}(\cdot) \rangle^2}{\|\mathbf{H}^\dagger(\cdot)\|^2} \leq \frac{\sigma_x^2}{\sigma_v^2} \|\mathbf{C}(\cdot)\|^2$$

Cauchy-Schwartz inequality, equality (max) reached for $\mathbf{H}^\dagger(z) = \mathbf{C}(z)$.

- The Viterbi equalizer can be fairly complex however. A class of simple sub-optimal equalizers can be obtained as the cascade of a simple linear estimator followed by an instantaneous detector (decision element).
- The class of so-called *linear equalizers* (LEs) performs linear estimation on the basis of the signal y_k alone.
- The more sophisticated class of *decision-feedback equalizers* (DFEs) performs linear estimation on the basis of all y_k plus also the previously detected x_k (hence, feedback of previous decisions).

Zero-Forcing Linear Equalizers

- Linear equalizers are simply linear filters $H(z)$ filtering the signal y_k and their output $\hat{x}_k = H(q)y_k$ gets processed by a decision element.
- classical point of view of filtering a signal in noise: zero distortion for the signal part. The signal part in \hat{x}_k is $H(q)C(q)x_k$. To obtain zero distortion (signal part of \hat{x}_k equal to x_k) would mean that

$$H(z)C(z) = 1 \Rightarrow H_{ZF}(z) = \frac{1}{C(z)} = \frac{1}{C_{min}(z)} \frac{1}{C_{max}(z)}$$

Since this equalizer forces the resulting ISI to zero, this solution is called the *zero-forcing* (ZF) equalizer.

- $C_{min}(z)$ and $C_{max}(z)$ are the minimum-phase and maximum-phase factors of $C(z)$ (in general not minimum-phase nor maximum-phase) The inverses of $C_{min}(z)$ and $C_{max}(z)$ are causal and anti-causal resp. This means that in general $H_{ZF}(z)$ will have an impulse response that extends from $-\infty$ to $+\infty$ (even if $C(z)$ is FIR).

Zero-Forcing Linear Equalizers (2)

- In practice, $H(z)$ will normally be approximated with an FIR filter. This FIR approximation will have to be non-causal in order to get a good approximation. Such finite non-causality can be dealt with by introducing a corresponding delay. The FIR approximation will have to be longer as the zeros of $C(z)$ approach the unit circle. The ZF equalizer does not exist if $C(z)$ has zeros on the unit circle.
- error signal $\tilde{x}_k = x_k - \hat{x}_k = H(q)v_k = \frac{1}{C(q)}v_k$ only contains noise, due to ZF

$$MSE_{ZF-LE} = E \tilde{x}_k^2 = \frac{\sigma_v^2}{2\pi j} \oint \frac{dz}{z} \frac{1}{C^\dagger(z)C(z)}$$

The MSE can get very big as some zeros of $C(z)$ approach the unit circle. This phenomenon is called *noise enhancement*. The perfect cancellation of the ISI is obtained at the cost of enhancing the noise. The SNR of the linear equalizer is defined as

$$SNR_{ZF-LE} = \frac{\sigma_x^2}{MSE_{ZF-LE}} \text{ or more precisely } SINR = \frac{E(E\hat{x}_k|_{x_k})^2}{E(\hat{x}_k - E\hat{x}_k|_{x_k})^2}$$

Using the Cauchy-Schwarz inequality (or Jensen's inequality), one can show

$$SNR_{ZF-LE} \leq MFB .$$

MMSE Linear Equalizers

- The optimal filtering point of view simply takes the MSE as optimality criterion.
Hence we get the Wiener filter

$$H_{MMSE-LE}(z) = \frac{S_{xy}(z)}{S_{yy}(z)} = \frac{\sigma_x^2 C^\dagger(z)}{\sigma_x^2 C^\dagger(z)C(z) + \sigma_v^2} \quad \left(\xrightarrow{\sigma_v^2 \rightarrow 0} H_{ZF-LE}(z) = \frac{1}{C(z)} \right)$$

Remark: the factor $C^\dagger(z)$ in the numerator represents the matched filter.

$$\begin{aligned} MSE_{MMSE-LE} &= E(x_k - \hat{x}_k)x_k = \sigma_x^2 - \frac{1}{2\pi j} \oint \frac{dz}{z} S_{\hat{x}\hat{x}}(z) = \sigma_x^2 - \frac{1}{2\pi j} \oint \frac{dz}{z} H(z)S_{yx}(z) \\ &= \sigma_x^2 \left(1 - \frac{\sigma_x^2}{2\pi j} \oint \frac{dz}{z} C^\dagger(z)S_{yy}^{-1}(z)C(z) \right) = \frac{\sigma_x^2 \sigma_v^2}{2\pi j} \oint \frac{dz}{z} S_{yy}^{-1}(z) = \frac{\sigma_v^2}{2\pi j} \oint \frac{dz}{z} \frac{1}{C^\dagger(z)C(z) + \frac{\sigma_v^2}{\sigma_x^2}}. \end{aligned}$$

From the previous expression, it is clear that

$$MSE_{MMSE-LE} \leq \min \{ \sigma_x^2, MSE_{ZF-LE} \} \Rightarrow SNR_{MMSE-LE} \geq \max \{ 1, SNR_{ZF-LE} \}.$$

This means that the MMSE-LE always does at least as well as the ZF-LE, at least in terms of MSE. Using the Cauchy-Schwarz inequality, one can show that

$$SNR_{MMSE-LE} \leq MFB + 1.$$

Remark that the MMSE equalizer converges to the ZF equalizer as $\sigma_v^2 \rightarrow 0$

Unbiased MMSE Linear Equalizers

- remark: $SNR_{MMSE-LE}$ can be larger than the MFB. This is due to the fact that the MMSE LE gives a *biased* estimate of x_k : the coefficient of x_k appearing in \hat{x}_k is not equal to 1.
- Although the unconstrained MMSE LE gives the lowest MSE, the bias in \hat{x}_k will increase the *probability of error* in the decision process. The decision element expects indeed to see x_k plus some random deviations at its input, whereas a bias (in the form of αx_k) is not a random perturbation w.r.t. x_k . Random perturbations are perturbations due to the $x_i, i \neq k$, and the v_i . The unbiased MMSE (UMMSE) LE minimizes the MSE subject to the constraint that the estimator be unbiased, i.e. $E [\hat{x}_k | x_k] = x_k$, or $\frac{1}{2\pi j} \oint \frac{dz}{z} \mathbf{H}(z) \mathbf{C}(z) = \sum_k h_k c_{-k} = 1$.
- The MMSE equalizer takes the Bayesian viewpoint in which all the x_k and the v_k are considered random. The UMMSE equalizer takes the more deterministic viewpoint in which the symbol to be estimated x_k is considered deterministic, but the other $x_i, i \neq k$, and the v_i are still random (this the viewpoint of the BLUE estimator, considered here for an infinite number of measurements y_k).

Unbiased MMSE Linear Equalizers (2)

- The UMMSE equalizer design problem is hence

$$\min_{h_i: \sum_i h_i c_{-i} = 1} E (x_k - \sum_i h_i y_{k-i})^2 .$$

We can turn this constrained optimization problem into an unconstrained optimization problem by introducing a Lagrange multiplier λ :

$$\min_{h_i, \lambda} f(\mathbf{H}(.), \lambda) = \min_{h_i, \lambda} \left\{ E (x_k - \sum_i h_i y_{k-i})^2 + \lambda (\sum_i h_i c_{-i} - 1) \right\} .$$

By setting derivatives equal to zero, we find

$$\begin{aligned} \frac{\partial f}{\partial h_i} &= 2 E \left(x_k - \sum_n h_n y_{k-n} \right) (-y_{k-i}) + \lambda c_{-i} = 0 \\ \frac{\partial f}{\partial \lambda} &= \sum_i h_i c_{-i} - 1 = 0 \end{aligned}$$

The first equation leads to

$$\begin{aligned} -r_{xy}(i) + \sum_n h_n r_{yy}(i-n) + \frac{\lambda}{2} c_{-i} &= 0 \\ \Rightarrow -\mathbf{S}_{xy}(z) + \mathbf{H}(z)\mathbf{S}_{yy}(z) + \frac{\lambda}{2}\mathbf{C}^\dagger(z) &= 0 \Rightarrow \mathbf{H}(z) = (\sigma_x^2 - \frac{\lambda}{2})\mathbf{C}^\dagger(z)\mathbf{S}_{yy}^{-1}(z) . \end{aligned}$$

Unbiased MMSE Linear Equalizers (3)

- From the constraint, we find

$$\frac{1}{2\pi j} \oint \frac{dz}{z} \mathbf{H}(z) \mathbf{C}(z) = 1 = (\sigma_x^2 - \frac{\lambda}{2}) \frac{1}{2\pi j} \oint \frac{dz}{z} \mathbf{C}^\dagger(z) \mathbf{S}_{yy}^{-1}(z) \mathbf{C}(z).$$

Hence

$$\mathbf{H}_{UMMSE}(z) = \left(\frac{1}{2\pi j} \oint \frac{dz}{z} \mathbf{C}^\dagger(z) \mathbf{S}_{yy}^{-1}(z) \mathbf{C}(z) \right)^{-1} \mathbf{C}^\dagger(z) \mathbf{S}_{yy}^{-1}(z)$$

Hence, the UMMSE equalizer is simply proportional to the MMSE equalizer, with the proportionality factor adjusted for unbiasedness.

- We find for the MSE

$$\begin{aligned} MSE_{UMMSE-LE} &= E(x_k - \hat{x}_k)^2 = \frac{1}{2\pi j} \oint \frac{dz}{z} (\mathbf{S}_{xx}(z) - \mathbf{S}_{x\hat{x}}(z) - \mathbf{S}_{\hat{x}x}(z) + \mathbf{S}_{\hat{x}\hat{x}}(z)) \\ &= \frac{1}{2\pi j} \oint \frac{dz}{z} (\mathbf{S}_{xx}(z) - \mathbf{S}_{xx}(z) - \mathbf{S}_{xx}(z) + \mathbf{S}_{\hat{x}\hat{x}}(z)) = \underbrace{\left(\frac{1}{2\pi j} \oint \frac{dz}{z} \mathbf{C}^\dagger(z) \mathbf{S}_{yy}^{-1}(z) \mathbf{C}(z) \right)^{-1}}_{= \sigma_{\hat{x}}^2_{UMMSE}} - \sigma_x^2 \end{aligned}$$

from which we can find the SNR

$$SNR_{UMMSE-LE} = \frac{\sigma_x^2}{MSE_{UMMSE-LE}} = \frac{\frac{\sigma_x^2}{2\pi j} \oint \frac{dz}{z} \mathbf{C}^\dagger(z) \mathbf{S}_{yy}^{-1}(z) \mathbf{C}(z)}{1 - \frac{\sigma_x^2}{2\pi j} \oint \frac{dz}{z} \mathbf{C}^\dagger(z) \mathbf{S}_{yy}^{-1}(z) \mathbf{C}(z)} = \frac{1}{\frac{\sigma_v^2}{2\pi j} \oint \frac{dz}{z} \mathbf{S}_{yy}^{-1}(z)} - 1$$

Unbiased MMSE Linear Equalizers (4)

- we find that

$$SNR_{UMMSE} = SNR_{MMSE} - 1 \quad !$$

Even though the UMMSE LE has a lower SNR than the MMSE LE, it can be shown that its probability of error is lower (using the Gaussian assumption on the interfering symbols - Central Limit Theorem). Since the UMMSE has the highest SNR of all unbiased LEs, it has the lowest probability of error.

- We also have

$$SNR_{ZF-LE} \leq SNR_{UMMSE-LE} \leq MFB = \frac{1}{\sigma_v^2 2\pi j} \oint \frac{dz}{z} S_{yy}(z) - 1$$

where the second inequality is due to the Cauchy-Schwarz inequality

$$\left(\frac{1}{2\pi j} \oint \frac{dz}{z} S_{yy}^{-1}(z) \right)^{-1} \leq \frac{1}{2\pi j} \oint \frac{dz}{z} S_{yy}(z)$$

with equality iff $S_{yy}(f)$ and hence $|C(f)|$ is constant as a function of f ($c_k = c_i \delta_{ki}$ for one certain i). In that case also $SNR_{ZF-LE} = MFB$.

Causal Wiener Filtering

- temporal processing: often need to constrain the filter to be causal (real time implementation)
- using the data $\{y_n, n \leq k\}$, estimate $x_{k+\lambda}$:

$\lambda = 0$: *filtering*

$\lambda > 0$: *prediction*

$\lambda < 0$: *smoothing*

- Instead of y_k , consider its whitened version $f_{\infty,k}$ which is obtained by filtering y_k with $A_{\infty}(z) = S_{yy}^+(\infty)/S_{yy}^+(z)$, a causal and causally invertible filter.

$f_{\infty,k}$ = *innovations* of y_k = white noise with variance $\sigma_{f,\infty}^2$

- $S_{yy}(z) = S_{yy}^+(z) S_{yy}^+(z^{-1})$: *spectral factorization*. Subject to certain conditions, a psdf can be factored into its causal minimum-phase factor $S_{yy}^+(z)$ and its anti-causal maximum-phase counterpart $S_{yy}^+(z^{-1})$.

- $S_{yy}^+(\infty) = \sigma_{f,\infty}$ since $S_{yy}(z) = \frac{\sigma_{f,\infty}^2}{A_{\infty}(z)A_{\infty}(z^{-1})}$

Causal Wiener Filtering (2)

- LMMSE estimation:

$$\min_{h_{k,n}^f} E(x_{k+\lambda} - \hat{x}_{k+\lambda})^2, \quad \hat{x}_{k+\lambda|k} = \sum_{m=-\infty}^k h_{k,m} y_m = \sum_{m=-\infty}^k h_{k,m}^f f_{\infty,m}$$

- orthogonality conditions $\Rightarrow \infty$ normal equations: decoupled!

$$E(x_{k+\lambda} - \hat{x}_{k+\lambda}) f_{\infty,n} = 0 = r_{x f_\infty}(\lambda + k - n) - h_{k,n}^f \sigma_{f,\infty}^2$$

Hence

$$h_{k,n}^f = \frac{r_{x f_\infty}(\lambda + k - n)}{\sigma_{f,\infty}^2} = h_{0,n-k}^f = h_{k-n}^f, \quad n \leq k$$

- $h_k^f = \frac{r_{x f_\infty}(\lambda + k)}{\sigma_{f,\infty}^2}, \quad k \geq 0 \quad \rightarrow \quad \mathbf{H}^f(z) = \frac{1}{\sigma_{f,\infty}^2} \left\{ \mathbf{S}_{x f_\infty}(z) z^\lambda \right\}_+$

$\{.\}_+$: “take the causal part of”

- $\mathbf{H}(z) = \mathbf{H}^f(z) \mathbf{A}_\infty(z) = \frac{1}{\sigma_{f,\infty}^2} \left\{ \mathbf{S}_{x f_\infty}(z) z^\lambda \right\}_+ \frac{\sigma_{f,\infty}}{\mathbf{S}_{yy}^+(z)} = \frac{1}{\mathbf{S}_{yy}^+(z)} \left\{ \frac{\mathbf{S}_{xy}(z) z^\lambda}{\mathbf{S}_{yy}^+(z^{-1})} \right\}_+$

- if drop causality constraint: $\mathbf{H}(z) = \frac{\mathbf{S}_{xy}(z) z^\lambda}{\mathbf{S}_{yy}(z)}$



Statistical Signal Processing

Lecture 9

chapter 3: Optimal Filtering

Wiener filtering

- signal in noise
- equalization
- causal Wiener filtering
- FIR Wiener filtering
 - iterative solution: steepest-descent algorithm

chapter 4: Adaptive Filtering

- LMS algorithm
- Normalized LMS (NLMS) algorithm



FIR Wiener Filtering

- LMMSE estimation of x_k from y_k using a causal FIR filter

$$\bullet \hat{x}_k = \sum_{i=0}^{N-1} h_i y_{k-i} = Y_k^T H = H^T Y_k \text{ where } Y_k = \begin{bmatrix} y_k \\ y_{k-1} \\ \vdots \\ y_{k-N+1} \end{bmatrix}, \quad H = \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{N-1} \end{bmatrix}$$

$$\begin{aligned} \text{MSE} &= \xi(H) = E(x_k - \hat{x}_k)^2 = E(x_k - H^T Y_k)^2 \\ &= E(x_k^2 - 2H^T Y_k x_k + H^T Y_k Y_k^T H) \\ &= \sigma_x^2 - 2H^T E(Y_k x_k) + H^T E(Y_k Y_k^T) H \\ &= \sigma_x^2 - 2H^T R_{Yx} + H^T R_{YY} H \end{aligned}$$

- MSE $\xi(H)$ is a quadratic cost function in H . Minimization \Rightarrow gradient = 0

$$\frac{\partial \xi}{\partial H} \triangleq \begin{bmatrix} \frac{\partial \xi}{\partial h_0} \\ \vdots \\ \frac{\partial \xi}{\partial h_{N-1}} \end{bmatrix} = 2(R_{YY} H^o - R_{Yx}) = 0 \Rightarrow H^o = R_{YY}^{-1} R_{Yx}$$

- extremum = minimum because the Hessian $\left[\frac{\partial^2 \xi}{\partial h_i \partial h_j} \right]_{i,j=0}^{N-1} = 2R_{YY} > 0$



FIR Wiener Filtering (2)

- The minimum MSE (MMSE) can be found to be:

$$\begin{aligned}\xi^o &= \xi(H^o) = \sigma_x^2 - 2R_{Yx}^T H^o + H^{oT} R_{YY} H^o \\ &= \dots = \sigma_x^2 - R_{xY} R_{YY}^{-1} R_{Yx} = \sigma_x^2 - R_{xY} H^o\end{aligned}$$

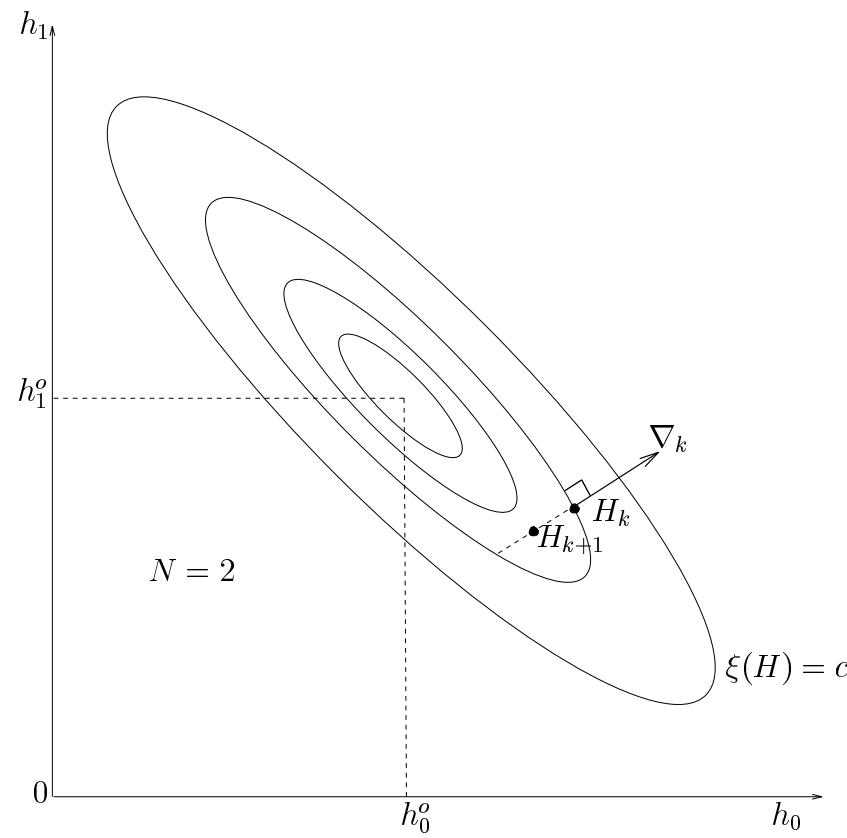
- alternatively, the criterion can be written as

$$\xi(H) = \xi^o + (H - H^o)^T R_{YY} (H - H^o)$$

which brings out clearly the parabolic character.



FIR Wiener Filtering (3)





Steepest Descent Algorithm

- iterative algorithm to solve the normal equations $R_{YY} H^o = R_{Yx}$, updates in the direction of steepest descent Let μ be the stepsize parameter. It determines how far we shall go in the direction of steepest descent. If H_k represents the approximate solution at iteration step k , then we get the following recursion:

$$H_{k+1} = H_k - \frac{\mu}{2} \nabla_k$$

where

$$\nabla_k = \left. \frac{\partial \xi}{\partial H} \right|_{H=H_k} = 2R_{YY}H_k - 2R_{Yx} = 2R_{YY}(H_k - H^o) = -2R_{YY}\tilde{H}_k$$

where we have introduced the filter approximation error $\tilde{H}_k = H^o - H_k$.

- So the iterative algorithm becomes

$$H_{k+1} = H_k - \mu(R_{YY}H_k - R_{Yx}) = (I - \mu R_{YY})H_k + \mu R_{Yx}.$$

note: $\frac{1}{2}\nabla_k = R_{YY}H_k - R_{Yx}$, the error in the satisfaction of the system of normal equations of which we are seeking the solution, is used to adjust the approximation to the solution.

- Complexity: multiplying R_{YY} by a vector.



Steepest Descent Algorithm (2)

- Question: will the sequence H_k converge, and if so, to H^o ?
- For the analysis of the convergence process, we shall assume knowledge of $H^o = R_{YY}^{-1}R_{Yx}$ but we shall see that the conditions for convergence do not depend on H^o . The conditions for convergence are conditions on the stepsize μ . From the derivation of the algorithm, it appears intuitively clear that the algorithm should work for a small positive μ since in that case we can be sure to make descending steps. If the steps we take are too big however, it is clear that we pass the valley and climb up the hill on the other side of the valley. So for large μ we do not expect convergence. Let us see what the analysis gives.
- Translation \Rightarrow set of homogeneous equations

$$\widetilde{H}_{k+1} = (I - \mu R_{YY}) \widetilde{H}_k$$

This is a set of coupled equations since R_{YY} is not a diagonal matrix in general.



Steepest Descent Algorithm (3)

- Eigen decomposition of the covariance matrix R :

$$R_{YY} = EY_k Y_k^T = V \Lambda V^T = \sum_{i=1}^N \lambda_i V_i V_i^T$$

where $\Lambda = \text{diag } \{\lambda_1 \cdots \lambda_N\}$, $V = [V_1 \ V_2 \cdots V_N]$ are matrices containing the eigenvalues and eigenvectors respectively. Note : $V^T V = I = V V^T$ or in other words $V_i^T V_j = \delta_{ij}$. We also assume that the eigenvalues are ordered: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N > 0$.

- Rotating the principal axes using V : $\widetilde{H}_k = V (V^T \widetilde{H}_k) = \sum_{i=1}^N V_i v_i(k)$

$$V^T \widetilde{H}_{k+1} = V^T (I - \mu R_{YY}) \widetilde{H}_k = (V^T - \mu \Lambda V^T) \widetilde{H}_k = (I - \mu \Lambda) V^T \widetilde{H}_k$$

which is now a system of decoupled equations ($I - \mu \Lambda$ is a diagonal matrix).

- If we introduce the error components $v_i(k)$ as

$$(V^T \widetilde{H}_k) \triangleq [v_1(k) \ v_2(k) \cdots v_N(k)]^T$$

then we find the following decoupled dynamics for the i th *natural mode*:

$$v_i(k+1) = (1 - \mu \lambda_i) v_i(k), \quad i = 1, \dots, N.$$



Steepest Descent Algorithm (4)

- The solution of these homogeneous difference equations of first order is readily obtained as

$$v_i(k) = (1 - \mu\lambda_i)^k v_i(0), \quad i = 1, \dots, N.$$

$v_i(k)$ = geometric series with geometric ratio = $1 - \mu\lambda_i$.

For *stability* or *convergence* of the steepest-descent algorithm:

$$-1 < 1 - \mu\lambda_i < 1, \quad i = 1, \dots, N, \Rightarrow \lim_{k \rightarrow \infty} (1 - \mu\lambda_i)^k = 0, \quad i = 1, \dots, N$$

Then $H_k \rightarrow H^o$, irrespective of H_0 .

- necessary and sufficient condition for convergence: $0 < \mu < \frac{2}{\lambda_1} = \min_{i \in \{1, \dots, n\}} \frac{2}{\lambda_i}$
- In absence of knowledge of the λ_i , the following reasoning leads to a safe operating region: $\lambda_1 < \sum_{i=1}^N \lambda_i = \text{tr } R_{YY} = N\sigma_y^2$, hence sufficient condition

$$0 < \mu < \frac{2}{N\sigma_y^2} < \frac{2}{\lambda_1}$$

In practice, it will be much easier to determine $\sigma_y^2 = (R_{YY})_{ii}$ than λ_1 .



Steepest Descent Algorithm (5)

- By taking the unit of time to be the duration of one iteration cycle, we can associate a time constant τ_i with the i th natural mode (for what follows, we assume $1-\mu\lambda_i > 0$):

$$|1 - \mu\lambda_i|^k = e^{-\frac{k}{\tau_i}} , \quad |1 - \mu\lambda_i| = e^{-\frac{1}{\tau_i}} \Rightarrow \tau_i = \frac{-1}{\ln|1 - \mu\lambda_i|}$$

which can be approximated, for small values of μ , as $\tau_i \approx \frac{1}{\mu\lambda_i}$, $\mu \ll 1$. This shows that the smaller the stepsize parameter μ , the slower will be the convergence of the steepest-descent algorithm.

- To see how the natural modes determine the evolution of the filter estimate H_k in the original coordinate system, consider

$$H = H^o - \widetilde{H} = H^o - V(V^T \widetilde{H}) = H^o - \sum_{i=1}^N V_i v_i .$$

So, when represented in the basis composed of the eigen vectors, the coordinates of \widetilde{H} are $[v_1 \cdots v_N]$. The evolution of H_k in terms of the natural modes becomes

$$\begin{aligned} H_k &= H^o - \widetilde{H}_k = H^o - V(V^T \widetilde{H}_k) = H^o - \sum_{i=1}^N V_i v_i(k) \\ &= H^o - \sum_{i=1}^N V_i v_i(0) (1 - \mu\lambda_i)^k . \end{aligned}$$



Steepest Descent Algorithm (6)

- $$\begin{aligned}\xi(H) &= \xi^o + \widetilde{H}^T R_{YY} \widetilde{H} = \xi^o + \widetilde{H}^T V \Lambda V^T \widetilde{H} \\ \bullet \text{ MSE: } &= \xi^o + [v_1 \cdots v_N] \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_N \end{bmatrix} \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix} = \xi^o + \sum_{i=1}^N \lambda_i v_i^2.\end{aligned}$$
- Hence, the set of points H of constant cost function, $\xi(H) = c'$, is given by

$$\sum_{i=1}^N \frac{v_i^2}{1/\lambda_i} = c = c' - \xi^o$$

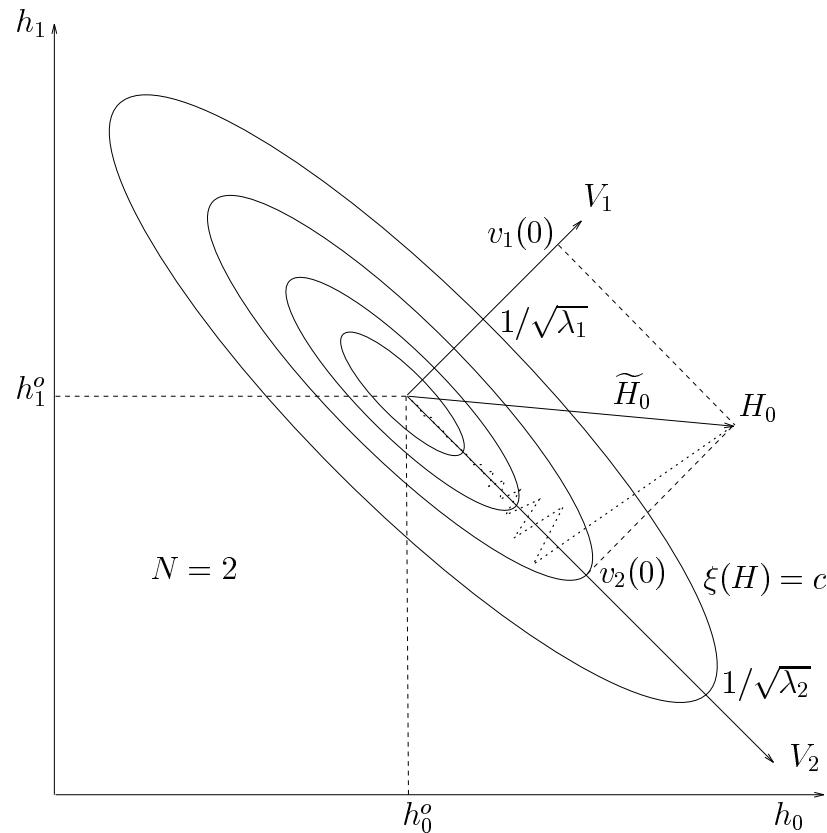
This is an ellipsoid (ellips when $N = 2$) centered at H^o and with principal axes coinciding with the eigenvectors of R_{YY} . In the next figure, the intersections with the principal axes are indicated for the ellips corresponding to $c = 1$. The convergence of the H_k is traced for a case in which $1 - \mu\lambda_1 < 0$, $1 - \mu\lambda_2 > 0$ and $|1 - \mu\lambda_1| \ll |1 - \mu\lambda_2|$. So for the component $v_1(k)$ along the V_1 -axis, we get alternating signs and a fairly fast convergence. For the component $v_2(k)$ along the V_2 -axis, we get a more slowly decaying exponential of constant sign.

- The evolution of the MSE :

$$\begin{aligned}\xi_k &= \xi(H_k) = \xi^o + \widetilde{H}_k^T R_{YY} \widetilde{H}_k = \xi^o + \widetilde{H}_k^T V \Lambda V^T \widetilde{H}_k = \xi^o + \sum_{i=1}^N \lambda_i v_i^2(k) \\ &= \xi^o + \sum_{i=1}^N \lambda_i (1 - \mu\lambda_i)^{2k} v_i^2(0) \stackrel{\Delta}{=} \xi^o + \xi_k^e \geq \xi^o.\end{aligned}$$



Steepest Descent Algorithm (7)





Steepest Descent Algorithm (8)

- The MSE is the sum of the minimum MSE (MMSE) ξ^o and what is called the *Excess MSE* (EMSE) ξ^e .
- The curve obtained by plotting the MSE ξ_k as a function of iteration number k is called the *learning curve*.
- In general, the learning curve of the steepest-descent algorithm consists of a sum of N exponentials, each one of which corresponds to a natural mode of the algorithm. Again, when the stepsize satisfies the convergence condition, we get that irrespective of the initial conditions

$$\lim_{k \rightarrow \infty} \xi_k = \xi^o, \quad \lim_{k \rightarrow \infty} \xi_k^e = 0.$$

Note that the time constants for the MSE are half of the corresponding time constants for the natural modes in the convergence of the filter estimate.

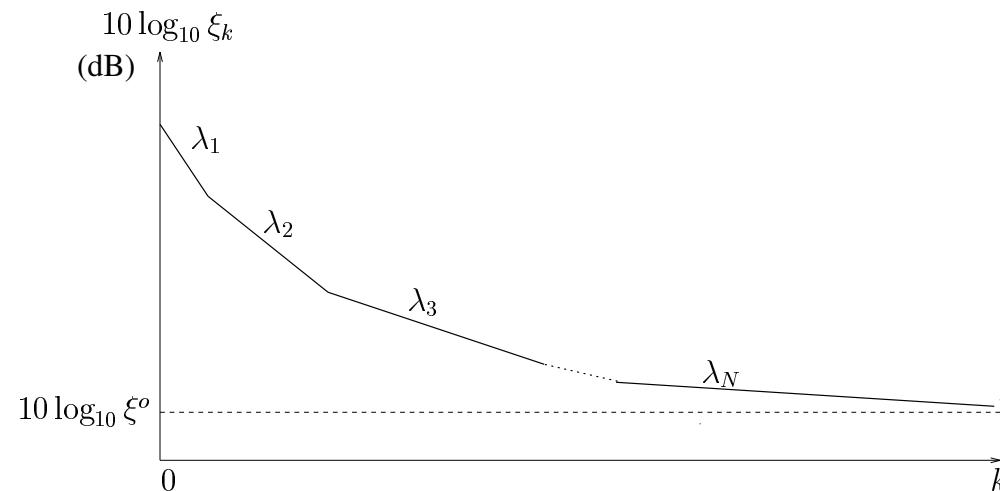
- The learning curve is dominated by different modes at different stages (if the eigenvalues are well separated). The (approximate) piecewise linear form of the learning curve holds when $\lambda_1 \gg \lambda_2 \gg \dots \gg \lambda_N$ and $1 - \mu\lambda_1 > 0$. In that case $\tau_1 \ll \tau_2 \ll \dots \ll \tau_N$.
- When the MSE is plotted in dB, an exponential decay of ξ_k corresponds to a linear decay of $10 \log_{10} \xi_k$.



Steepest Descent Algorithm (9)

learning curve

- In the beginning, the learning curve follows the decay of the fastest mode, associated with λ_1 . After about $2\tau_1 = 4\frac{\tau_1}{2}$ (the time constants for ξ_k are half those for H_k), the first mode has died out while the other modes have hardly decreased in this short time span. Now the decay associated with mode two dominates the learning curve until about $2\tau_2$ etc.
- If $1-\mu\lambda_1 < 0$, then a similar reasoning still applies. But then the time constants are not ordered according to the λ_i but according to the magnitudes $|1-\mu\lambda_i|$.





Steepest Descent Algorithm (10)

- fastest convergence : $\min_{\mu} \max_i |1 - \mu \lambda_i|$

which leads to the following condition

$$1 - \mu^o \lambda_1 = - (1 - \mu^o \lambda_N) \Rightarrow \mu^o = \frac{2}{\lambda_1 + \lambda_N} < \frac{2}{\lambda_1}$$

- (optimal) slowest mode : $\min_{\mu} \max_i |1 - \mu \lambda_i| = \frac{\lambda_1 - \lambda_N}{\lambda_1 + \lambda_N}$

This shows that in general the convergence of the steepest-descent algorithm slows down as the eigenvalue spread (the ratio λ_1/λ_N) increases.

- Convergence is fastest when all eigenvalues are equal. In that case

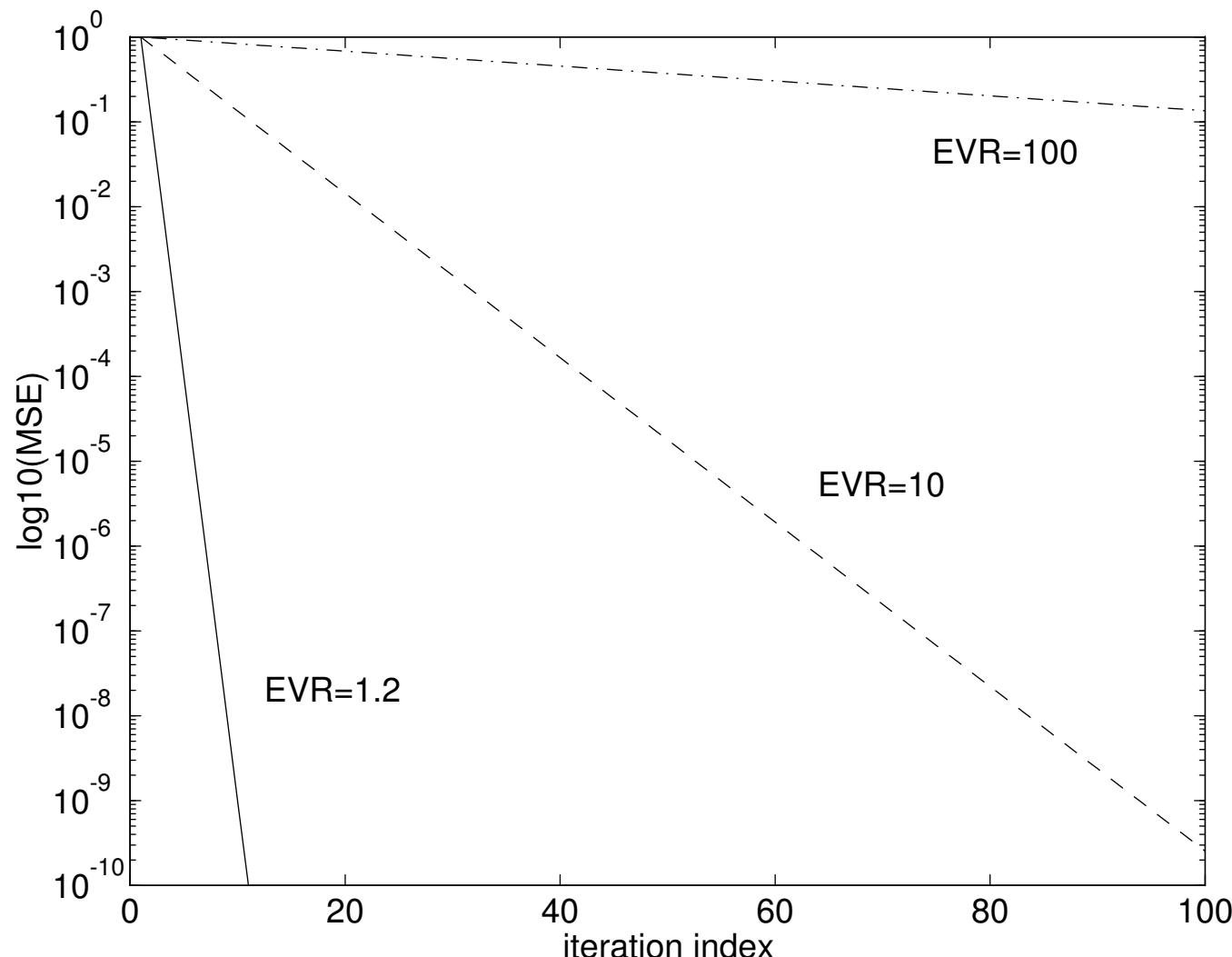
$$\min_{\mu} \max_i |1 - \mu \lambda_i| = 0$$

for $\mu^o = 1/\lambda_1$. This means that convergence occurs in one iteration! Indeed, when all eigenvalues are equal, then the ellipsoids become spheres. In this case, all negative gradients (at any point in H space) point towards H^o . So it suffices to take the right stepsize to end up at H^o in one step.



Steepest Descent Algorithm (11)

EVR = $\frac{\lambda_1}{\lambda_N}$, decay as $\left(\frac{\lambda_1 - \lambda_N}{\lambda_1 + \lambda_N}\right)^{2k}$





LMS Algorithm

- *Adaptive Filtering:* R_{YY} and R_{Yx} not available but a realization $\{x_k, y_k, k \geq 0\}$ is available. From these samples, we could of course try to estimate R_{YY} and R_{Yx} , and below we shall see that the *Recursive Least-Squares* (RLS) algorithm effectively does just that. However, we may also find that such an approach becomes too complicated, too computationally demanding. The *Least Mean Square* (LMS) algorithm takes the following alternative approach (invented by Widrow and Hopf).
- So actually: not x_k of interest but $\hat{x}_k, \tilde{x}_k, H$ or x_k at other time instants (training).
- Let's just drop the mathematical expectation operator E in the MSE criterion, and instead consider the instantaneous squared error. So the new cost function is simply

$$J_k(H) = \epsilon_k^2(H) = (x_k - H^T Y_k)^2.$$

Apply steepest-descent on this criterion, iteration index = time index now, 1 iteration per sample.

So we have dropped the statistical averaging operation, but we hope to replace it by some amount of time domain averaging inherent in the low-pass filtering nature of the adaptation process.

- Just as this criterion can be considered an instantaneous estimate of the MSE, the true criterion of interest, the gradient of J_k can be considered to be an estimate of the true gradient.



LMS Algorithm (2)

$$\bullet \widehat{\nabla}_k = \left. \frac{\partial J_k}{\partial H} \right|_{H=H_{k-1}} = \begin{bmatrix} \frac{\partial \epsilon_k^2}{\partial h_0} \\ \frac{\partial \epsilon_k^2}{\partial h_1} \\ \vdots \\ \frac{\partial \epsilon_k^2}{\partial h_{N-1}} \end{bmatrix}_{H=H_{k-1}} = 2\epsilon_k^p \begin{bmatrix} \frac{\partial \epsilon_k}{\partial h_0} \\ \frac{\partial \epsilon_k}{\partial h_1} \\ \vdots \\ \frac{\partial \epsilon_k}{\partial h_{N-1}} \end{bmatrix}_{H=H_{k-1}} = -2\epsilon_k^p Y_k$$

where $\epsilon_k^p \triangleq \epsilon_k(H_{k-1}) = x_k - H_{k-1}^T Y_k$ is the *predicted* or *a priori* error signal, obtained as the difference of the desired response and the filter output at time k , but using the filter estimate at time $k-1$.

- With this simple estimate of the gradient, we can specify a steepest-descent type of adaptive algorithm. We get

$$H_k = H_{k-1} - \frac{\mu}{2} \widehat{\nabla}_k = H_{k-1} + \mu \epsilon_k^p Y_k$$

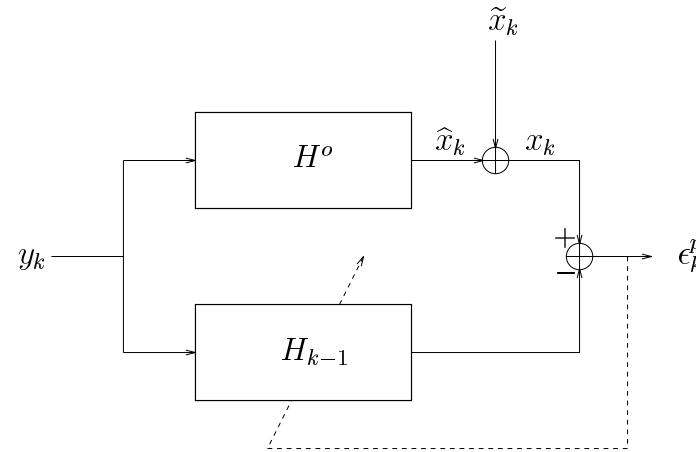
The last 2 equations specify the LMS algorithm.

- Note: the LMS algorithm does one iteration per sample period.
- computational simplicity: $2N$ operations per sample
- The main question now is, does H_k converge to the Wiener solution H^o in any sense?



LMS Algorithm (3)

a model for the purpose of analysis



- $x_k = \hat{x}_k + \tilde{x}_k = Y_k^T H^o + \tilde{x}_k$ where \tilde{x}_k is independent white noise with variance ξ^o
- If we now consider estimating x_k from Y_k by some arbitrary FIR filter H , then the associated MSE becomes

$$\begin{aligned}\xi(H) &= E(x_k - H^T Y_k)^2 = E(\tilde{x}_k + H^{oT} Y_k - H^T Y_k)^2 \\ &= E(\widetilde{H}^T Y_k + \tilde{x}_k)^2, \quad \widetilde{H} = H^o - H \\ &= E\tilde{x}_k^2 + 2\widetilde{H}^T EY_k \tilde{x}_k + \widetilde{H}^T (EY_k Y_k^T) \widetilde{H} \\ &= \xi^o + \widetilde{H}^T R_{YY} \widetilde{H}\end{aligned}$$

This shows among other things that $H = H^o$ is indeed the Wiener solution.



LMS Algorithm (4)

- *independence assumption* : treat H_{k-1} and Y_k as if they were independent
- *learning curve* :

$$\begin{aligned} E(\epsilon_k^p)^2 &= E(x_k - H_{k-1}^T Y_k)^2 = E(\tilde{x}_k + H^o T Y_k - H_{k-1}^T Y_k)^2 \\ &= E(\tilde{H}_{k-1}^T Y_k + \tilde{x}_k)^2, \quad \tilde{H}_k = H^o - H_k \\ &= E\tilde{x}_k^2 + 2E\tilde{H}_{k-1}^T Y_k \tilde{x}_k + E\tilde{H}_{k-1}^T (Y_k Y_k^T) \tilde{H}_{k-1} \\ &= E\tilde{x}_k^2 + 2(E\tilde{H}_{k-1}^T Y_k)(E\tilde{x}_k) + \text{tr}(E(Y_k Y_k^T)(\tilde{H}_{k-1} \tilde{H}_{k-1}^T)) \\ &= E\tilde{x}_k^2 + 2(E\tilde{H}_{k-1}^T Y_k)0 + \text{tr}((EY_k Y_k^T)(E\tilde{H}_{k-1} \tilde{H}_{k-1}^T)) \\ &= \xi^o + \text{tr}(R_{YY} C_{k-1}) = E\xi(H_{k-1}) \end{aligned}$$

where we used the independence assumption and introduced $C_k \triangleq E\tilde{H}_k \tilde{H}_k^T$.

- If $C_k \rightarrow 0$ then $\tilde{H}_k \rightarrow 0$ or $H_k \rightarrow H^o$ in mean square.
- Note that convergence of the correlation matrix implies convergence of the mean and covariance of the estimation error:

$$C_k = E\tilde{H}_k \tilde{H}_k^T = (E\tilde{H}_k)(E\tilde{H}_k)^T + E(\tilde{H}_k - E\tilde{H}_k)(\tilde{H}_k - E\tilde{H}_k)^T$$

- In summary, in order to investigate the learning curve (MSE) of the LMS algorithm (the curve of $E\xi(H_k)$), we need to investigate how C_k evolves.



LMS Algorithm (5)

- From the LMS algorithm we get

$$\begin{aligned}\widetilde{H}_k &= H^o - H_k = H^o - H_{k-1} - \mu \epsilon_k^p Y_k \\ &= \widetilde{H}_{k-1} - \mu(Y_k^T \widetilde{H}_{k-1} + \tilde{x}_k) Y_k \\ \widetilde{H}_k &= (I - \mu Y_k Y_k^T) \widetilde{H}_{k-1} - \mu \tilde{x}_k Y_k\end{aligned}$$

- Compared to the steepest-descent algorithm, we now have a driving term (input).
- Also, in the system transition matrix, R_{YY} got replaced by $Y_k Y_k^T$. The system matrix $I - \mu Y_k Y_k^T$ is now stochastic. This makes the system very hard to analyze. However, it is possible to introduce a simplification when the stepsize is small.
- independence assumption \Rightarrow analysis of first order moment $E\widetilde{H}_k$

$$E\widetilde{H}_k = (I - \mu R_{YY}) E\widetilde{H}_{k-1}$$

The mean $E\widetilde{H}_k$ in LMS converges like \widetilde{H}_k in steepest-descent. However, it is not because the mean of \widetilde{H}_k converges to zero that \widetilde{H}_k converges to zero. To see how $E(\epsilon_k^p)^2$ converges, we have to analyze the second-order moments of \widetilde{H}_k .



LMS Algorithm (6)

Averaging Theorem

- Consider a (stationary) stochastic dynamic (F_k) system

$$\zeta_k = (I - \mu F_k) \zeta_{k-1} + W_k$$

- Now consider the averaged system $\bar{\zeta}_k = (I - \mu E F_k) \bar{\zeta}_{k-1} + W_k$
- difference between the two systems:

$$\begin{aligned}\zeta_k &= (I - \mu E F_k + \mu(E F_k - F_k)) \zeta_{k-1} + W_k \\ \Rightarrow (\zeta_k - \bar{\zeta}_k) &= (I - \mu E F_k)(\zeta_{k-1} - \bar{\zeta}_{k-1}) + \mu(E F_k - F_k) \zeta_{k-1}\end{aligned}$$

and hence $\zeta_k - \bar{\zeta}_k \sim \mu$

- The averaging theorem consists of the following weak convergence result:

distribution of $\{\zeta_k\}$ \rightarrow distribution of $\{\bar{\zeta}_k\}$ as $\mu \rightarrow 0$

if the averaged system is exponentially stable.

- So if we want to study first and second order moments of ζ_k , we may as well study those of $\bar{\zeta}_k$ which is generated by the simpler averaged system. The results become correct in the limit as the stepsize becomes very small.



LMS Algorithm (7)

- Averaged LMS system: $\widetilde{H}_k = (I - \mu R_{YY}) \widetilde{H}_{k-1} - \mu \tilde{x}_k Y_k$
- mean: $E\widetilde{H}_k = (I - \mu R_{YY}) E\widetilde{H}_{k-1}$ same as steepest-descent!
- learning curve:

$$\begin{aligned}\xi_k &\triangleq E\xi(H_k) = E(\epsilon_{k+1}^p)^2 = \xi^o + \text{tr}(R_{YY}C_k) \\ &= \xi^o + \text{tr}(V\Lambda V^T C_k) = \xi^o + \text{tr}(\Lambda V^T C_k V) \\ &= \xi^o + \sum_{i=1}^N \lambda_i (V^T C_k V)_{ii}\end{aligned}$$

but

$$V^T C_k V = V^T E\widetilde{H}_k \widetilde{H}_k^T V = E(V^T \widetilde{H}_k)(V^T \widetilde{H}_k)^T = E \begin{bmatrix} v_1(k) \\ \vdots \\ v_N(k) \end{bmatrix} \begin{bmatrix} v_1(k) \\ \vdots \\ v_N(k) \end{bmatrix}^T$$

and hence $(V^T C_k V)_{ii} = Ev_i^2(k)$ which leads to

$$\xi_k = \xi^o + \sum_{i=1}^N \lambda_i Ev_i^2(k).$$



LMS Algorithm (8)

- in transformed coordinates: $V^T \widetilde{H}_k = (I - \mu \Lambda) V^T \widetilde{H}_{k-1} - \mu \tilde{x}_k V^T Y_k$
- put

$$V^T \widetilde{H}_k = \begin{bmatrix} v_1(k) \\ \vdots \\ v_N(k) \end{bmatrix}, \quad V^T Y_k = \begin{bmatrix} w_1(k) \\ \vdots \\ w_N(k) \end{bmatrix},$$
$$E \begin{bmatrix} w_1(k) \\ \vdots \\ w_N(k) \end{bmatrix} \begin{bmatrix} w_1(k) \\ \vdots \\ w_N(k) \end{bmatrix}^T = E V^T Y_k Y_k^T V = V^T R_{YY} V = \Lambda,$$

then

$$v_i(k) = (1 - \mu \lambda_i) v_i(k-1) - \mu \tilde{x}_k w_i(k).$$

- Taking the expected value of the square of both sides, we get

$$E v_i^2(k) = (1 - \mu \lambda_i)^2 E v_i^2(k-1) + \mu^2 E \tilde{x}_k^2 w_i^2(k) = (1 - \mu \lambda_i)^2 E v_i^2(k-1) + \mu^2 \xi^o \lambda_i$$

- Since we have the same dynamics as in the steepest-descent algorithm, we shall again have stability (convergence) when $0 < \mu < \frac{2}{\lambda_1}$.



LMS Algorithm (9)

- However, the results here are based on the averaging theorem which assumes that μ is small. If based on such a theory, we derive a condition for the maximum possible value for μ to have convergence, then we can expect that result to be erroneous. Indeed, the range of stable stepsize values for convergence of the second-order moments in practice turns out to be quite a bit more conservative than $\frac{2}{\lambda_1}$.
- In any case, for μ within the stable range, $Ev_i^2(k)$ will converge exponentially fast to some value $Ev_i^2(\infty)$:

$$Ev_i^2(\infty) = \frac{\mu}{2 - \mu\lambda_i} \xi^o .$$

- The learning curve converges to the value

$$\xi_\infty = \xi^o + \sum_{i=1}^N \lambda_i Ev_i^2(\infty) = \xi^o + \xi^e = \xi^o + \xi^o \mu \sum_{i=1}^N \frac{\lambda_i}{2 - \mu\lambda_i} = \xi^o(1 + M)$$

where

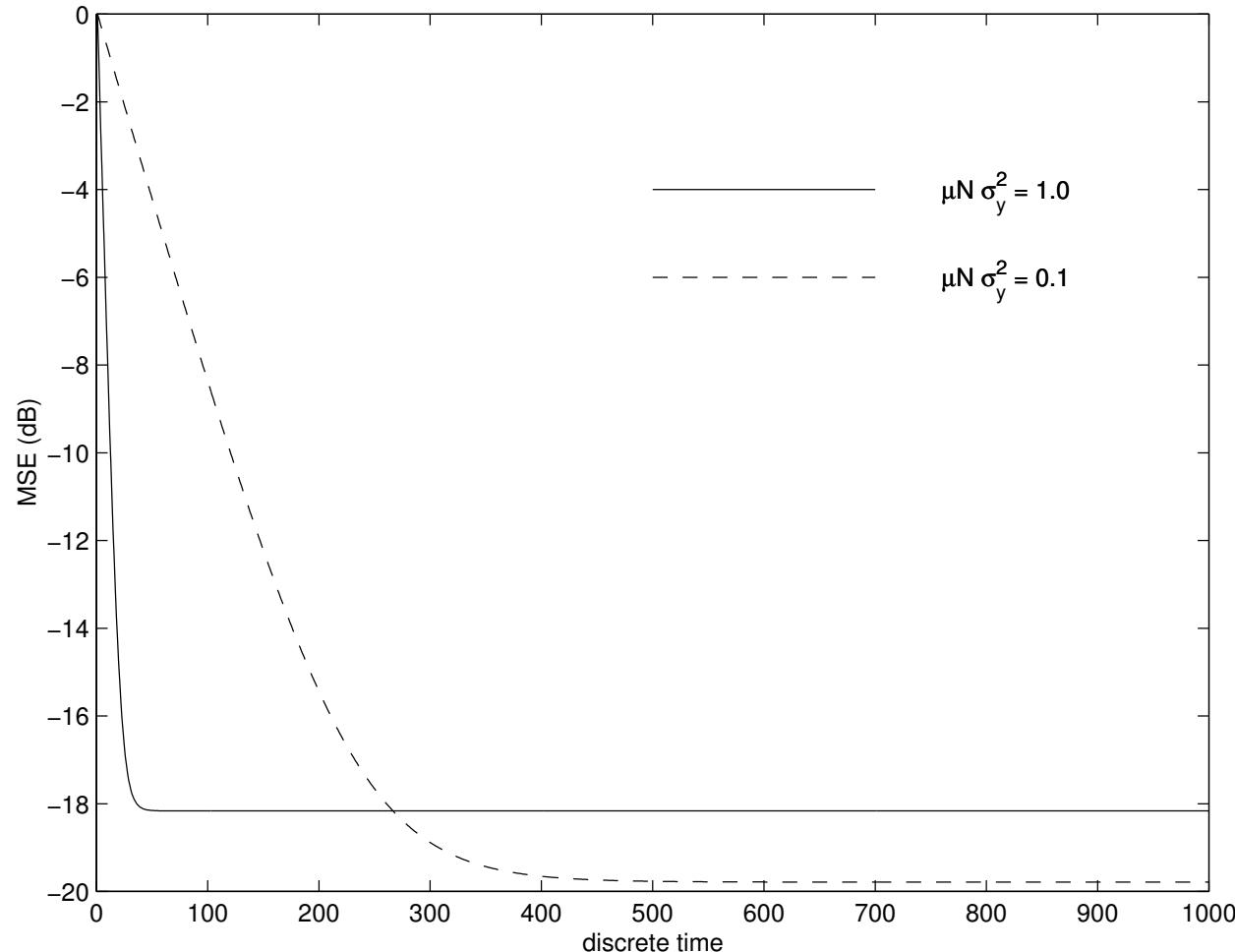
$$M = \frac{\xi^e}{\xi^o} = \mu \sum_{i=1}^N \frac{\lambda_i}{2 - \mu\lambda_i} \approx \frac{\mu}{2} \sum_{i=1}^N \lambda_i = \frac{\mu N \sigma_y^2}{2}$$

is called the *misadjustment* factor, ξ^e the *excess MSE*, and the indicated approximation holds for small μ .



LMS Algorithm (11)

$$\xi_k = \underbrace{\xi^o_{MMSE}}_{\text{mean} \rightarrow 0} + \underbrace{\sum_{i=1}^N \lambda_i (1 - \mu \lambda_i)^{2k} E v_i^2(0)}_{\text{variance} \rightarrow \text{EMSE}} + \xi^o \mu \sum_{i=1}^N \frac{1 - (1 - \mu \lambda_i)^{2k}}{2 - \mu \lambda_i} \lambda_i, \quad k \geq 1$$





LMS Algorithm (10)

- Although $EH_{\infty} = H^o$ (the estimate is asymptotically unbiased), the estimate H_k of H^o is not *consistent* because its variance does not decrease to zero. Due to the use of a finite nonzero stepsize, the LMS algorithm will, even in steady-state, continue taking steps in the noisy gradient direction (even though the true gradient would be zero at convergence). This residual amount of variance is measured by the misadjustment factor M or by the Excess MSE $\xi_{\infty}^e = \xi^o M$. The design of a constant stepsize μ results from a compromise: small values for μ lead to low misadjustment but slow convergence dynamics, large values lead to the opposite.
- *conditions for exact convergence* *gear shifting*

What if we require $M = 0$. Then we need $\mu = 0$ from our previous steady-state considerations. However, $\mu = 0$ will clearly not allow any adaptation. So what we need to do is to vary μ with k . The time-varying μ_k will have some finite value initially and decrease to zero as $k \rightarrow \infty$. One can show that under the following conditions

$$\mu_k \geq 0, \sum_{k=1}^{\infty} \mu_k = \infty, \sum_{k=1}^{\infty} \mu_k^2 < \infty \Rightarrow \mu_k \xrightarrow{k \rightarrow \infty} 0$$

the misadjustment converges to zero and $H_k \rightarrow H^o$, the Wiener solution, in mean square. A typical stepsize sequence : $\mu_k = \frac{c}{k}$



Normalized LMS Algorithm

- LMS: applies steepest descent approach to instantaneous squared filtering error

$$\epsilon_k^2(H) = (x_k - H^T Y_k)^2$$

$$\begin{cases} \epsilon_k^p = x_k - H_{k-1}^T Y_k \\ H_k = H_{k-1} + \mu \epsilon_k^p Y_k \end{cases},$$

where $\epsilon_k^p = \epsilon_k(H_{k-1})$ is the *a priori* (or *predicted*) filtering error.

- From our previous analysis: $\mu_{\max} \sim \frac{1}{\sigma_y^2}$. Therefore in order to be robust w.r.t. possible variations in the level of the input signal y_k , it is desirable to normalize the stepsize, to divide the stepsize by a quantity that behaves roughly as the variance σ_y^2 . The so-called Normalized LMS (NLMS) algorithm takes

$$\mu_k = \frac{\bar{\mu}}{\|Y_k\|^2}. \quad \text{Note: } E \|Y_k\|^2 = N\sigma_y^2$$

- NLMS:

$$\begin{cases} \epsilon_k^p = x_k - H_{k-1}^T Y_k \\ H_k = H_{k-1} + \frac{\bar{\mu}}{\|Y_k\|^2} \epsilon_k^p Y_k \end{cases}$$



Normalized LMS Algorithm (2)

- However, the consequences of the stepsize normalization reach much farther.
- Consider the *a posteriori* filtering error: $\epsilon_k = \epsilon_k(H_k)$.

$$\begin{cases} \text{LMS : } \epsilon_k = (1 - \mu \|Y_k\|^2) \epsilon_k^p \\ \text{NLMS : } \epsilon_k = (1 - \bar{\mu}) \epsilon_k^p . \end{cases}$$

NLMS allows a precise control of the magnitude of the *a posteriori* filtering error ϵ_k , and in particular $\bar{\mu} = 1$ makes $\epsilon_k \equiv 0$. LMS only allows to make ϵ_k small on the average, with possibly large values occurring at certain time instants, depending on the variations of $\|Y_k\|^2$ with time.

- remember model $x_k = H^o T Y_k + \tilde{x}_k$, (H^o = FIR Wiener filter), $\widetilde{H}_k = H^o - H_k$,

$$\begin{cases} \epsilon_k^p = \widetilde{H}_{k-1}^T Y_k + \tilde{x}_k \\ \text{LMS : } \widetilde{H}_k = \Phi_k^{LMS} \widetilde{H}_{k-1} - \mu \tilde{x}_k Y_k , \quad \Phi_k^{LMS} = I - \mu Y_k Y_k^T \\ \text{NLMS : } \widetilde{H}_k = \Phi_k^{NLMS} \widetilde{H}_{k-1} - \frac{\bar{\mu}}{\|Y_k\|^2} \tilde{x}_k Y_k , \quad \Phi_k^{NLMS} = I - \bar{\mu} \frac{Y_k Y_k^T}{Y_k^T Y_k} . \end{cases}$$

state transition matrix Φ_k determines the dynamics and in particular the stability of the error system \widetilde{H}_k .



Normalized LMS Algorithm (3)

- *long-term dynamics*: we used the averaging theorem, leading to the approximations

$$\begin{cases} \Phi_k^{LMS} \approx I - \mu R_{YY} \\ \Phi_k^{NLMS} \approx I - \frac{\bar{\mu}}{\text{tr} R_{YY}} R_{YY} , \end{cases}$$

where we have used for NLMS an additional approximation that is valid for large N . The analysis based on the averaging theorem has revealed the dependence of the dynamics on the eigenvalue spread λ_1/λ_N of R_{YY} .

- *instantaneous dynamics*: Φ_k can be shown to have the following eigenvalues

$$\begin{cases} \nu_1(k) = \nu_2 = \cdots = \nu_{N-1}(k) = 1 & \text{for LMS and NLMS} \\ \nu_N^{LMS}(k) = 1 - \mu \|Y_k\|^2 \\ \nu_N^{NLMS}(k) = 1 - \bar{\mu} , \end{cases}$$

and the eigenvector $W_N(k)$ corresponding to $\nu_N(k)$ is proportional to Y_k (what are the eigenvectors corresponding to $\nu_1(k), \nu_2(k), \dots, \nu_{N-1}(k)$?).

- Remark that we can write for both algorithms

$$\epsilon_k = \nu_N(k) \epsilon_k^p .$$



Normalized LMS Algorithm (4)

- $W_i(k)$ = eigenvector of Φ_k associated with eigenvalue $\nu_i(k)$. The eigenvectors form an orthonormal basis for \mathcal{R}^N and we can always write $\widetilde{H}_k = WW^T\widetilde{H}_k = \sum_{i=1}^N W_i (W_i^T \widetilde{H}_k)$. We can write for both algorithms

$$\begin{cases} W_i^T \widetilde{H}_k = W_i^T \widetilde{H}_{k-1}, & i = 1, 2, \dots, N-1 \\ W_N^T \widetilde{H}_k = \nu_N(k) W_N^T \widetilde{H}_{k-1} - \mu_k \tilde{x}_k W_N^T Y_k, & \mu_k = \begin{cases} \mu & \text{for LMS,} \\ \bar{\mu} & \text{for NLMS.} \end{cases} \end{cases}$$

Hence, for the components of \widetilde{H}_{k-1} along directions orthogonal to Y_k , nothing changes. The component of \widetilde{H}_{k-1} along the direction of Y_k however gets multiplied by $\nu_N(k)$.

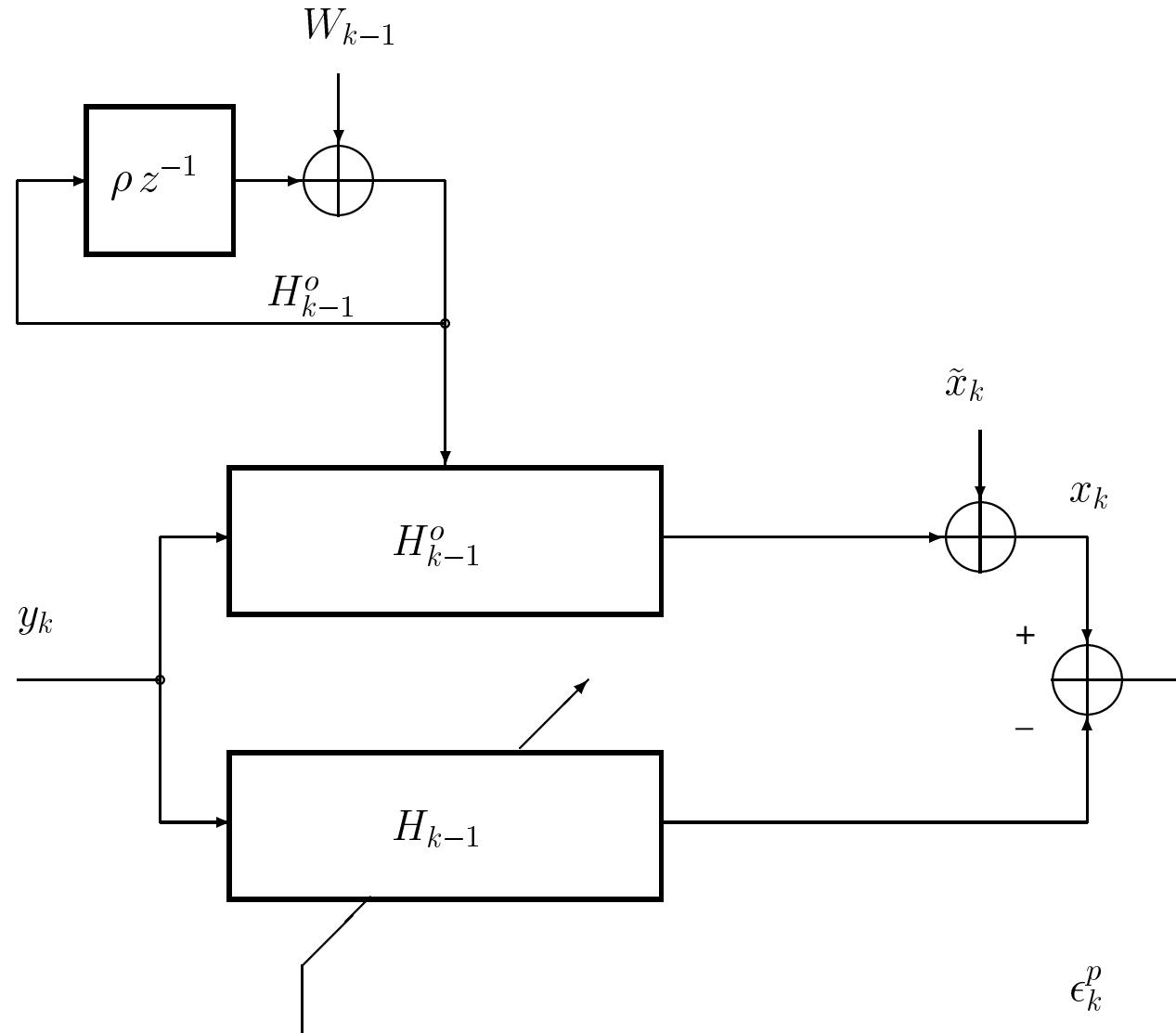
- For NLMS, $\nu_N^{NLMS}(k)$ can be controlled very precisely by $\bar{\mu}$. The choice for $\bar{\mu}$: trade-off between fast dynamics = small $|1-\bar{\mu}|$ = big $\bar{\mu}$, and a small noise amplification factor $\bar{\mu}$ (second (driving) term). In particular, $\bar{\mu} = 1$ results in complete elimination of the accumulated error component of \widetilde{H}_{k-1} in the direction of Y_k (also, $\epsilon_k = 0$ for $\bar{\mu} = 1$). However, with $\tilde{x}_k \neq 0$, some noise gets injected in the component of \widetilde{H}_k along Y_k .



Normalized LMS Algorithm (5)

- For LMS, if we want to make $|\nu_N^{LMS}(k)|$ small on the average , we need to choose μ fairly big. However, due to the statistical fluctuations of $\|Y_k\|$, $|\nu_N^{LMS}(k)|$ may get quite a bit bigger than 1 at certain times k , if μ is big. That means that even though LMS is converging on the average, it may actually take diverging steps at certain isolated time instants. This happens especially when μ is relatively large, when we want the average convergence to be as fast as possible.
- This problem of instantaneous diverging steps does not occur in the NLMS algorithm. Therefore, NLMS always converges faster than LMS when both algorithms have a stepsize that is optimized for fastest convergence speed.

Tracking Time-Varying Filters





Statistical Signal Processing

Lecture 10

chapter 3: Optimal Filtering

Wiener filtering

- FIR Wiener filtering
 - iterative solution: steepest-descent algorithm

chapter 4: Adaptive Filtering

- LMS algorithm
- Normalized LMS (NLMS) algorithm
- tracking behavior of LMS and RLS
- optimal tracking via Kalman filtering

chapter 5: Sinusoids in Noise



RLS Algorithm

- LS: replace the statistical averages by a time averages:

$$\xi_k(H) = \sum_{i=1}^k (x_i - H^T Y_i)^2 + (H - H_0)^T R_0 (H - H_0) ,$$

where the second term with $R_0 = R_0^T > 0$ allows for a proper initialization of the algorithm (the first term alone has a singular Hessian ($= 2 \sum_{i=1}^k Y_i Y_i^T$) for $k < N$).

- We can rewrite

$$\begin{aligned}\xi_k(H) &= H^T \left(\sum_{i=1}^k Y_i Y_i^T \right) H - 2H^T \left(\sum_{i=1}^k Y_i x_i \right) + \sum_{i=1}^k x_i^2 + (H - H_0)^T R_0 (H - H_0) \\ &= H^T \left(R_0 + \sum_{i=1}^k Y_i Y_i^T \right) H - 2H^T \left(R_0 H_0 + \sum_{i=1}^k Y_i x_i \right) + \sum_{i=1}^k x_i^2 + H_0^T R_0 H_0 \\ &= H^T R_k H - 2H^T P_k + \sum_{i=1}^k x_i^2 + H_0^T R_0 H_0\end{aligned}$$

where

$$\begin{aligned}R_k &= R_0 + \sum_{i=1}^k Y_i Y_i^T &= R_{k-1} + Y_k Y_k^T \\ P_k &= R_0 H_0 + \sum_{i=1}^k Y_i x_i &= P_{k-1} + Y_k x_k .\end{aligned}$$



Recursive Least-Squares Algorithm (2)

- By putting the gradient of $\xi_k(H)$ equal to zero and noting that the Hessian $2R_k > 0$, we find that the LS filter H_k that minimizes the LS criterion solves the following normal equations

$$R_k H_k = P_k .$$

To solve this set of equations at each time instant k would take $\mathcal{O}(N^3)$ operations at each time instant. In what follows, we shall derive the Recursive LS algorithm, which allows us, using information obtained at time $k-1$, to obtain H_k with only $\mathcal{O}(N^2)$ operations.

- we can rewrite $P_k = P_{k-1} + Y_k x_k$ as

$$\begin{aligned} R_k H_k &= R_{k-1} H_{k-1} + Y_k x_k \\ &= \left(R_k - Y_k Y_k^T \right) H_{k-1} + Y_k x_k \\ &= R_k H_{k-1} + Y_k \epsilon_k^p \end{aligned}$$

where $\epsilon_k^p = x_k - H_{k-1}^T Y_k$ as in the LMS algorithm. This leads immediately to

$$H_k = H_{k-1} + R_k^{-1} Y_k \epsilon_k^p$$

where $R_k^{-1} Y_k$ is called the Kalman gain (the RLS algorithm is a special case of the so-called Kalman filter).



Recursive Least-Squares Algorithm (3)

- Clearly, the RLS algorithm requires the recursive update of R_k^{-1} . This can be obtained using the Matrix Inversion Lemma:

$$\begin{aligned} R_k^{-1} &= \left(R_{k-1} + Y_k Y_k^T \right)^{-1} \\ &= R_{k-1}^{-1} - R_{k-1}^{-1} Y_k \left(1 + Y_k^T R_{k-1}^{-1} Y_k \right)^{-1} Y_k^T R_{k-1}^{-1} . \end{aligned}$$

This equation allows us to obtain R_k^{-1} from R_{k-1}^{-1} and Y_k using $\mathcal{O}(N^2)$ operations. When multiplying both sides with Y_k to the right, we obtain

$$R_k^{-1} Y_k = R_{k-1}^{-1} Y_k \left(1 + Y_k^T R_{k-1}^{-1} Y_k \right)^{-1} .$$

We find for the *a posteriori* error

$$\epsilon_k = x_k - H_k^T Y_k = \left(1 - Y_k^T R_k^{-1} Y_k \right) \epsilon_k^p = \left(1 + Y_k^T R_{k-1}^{-1} Y_k \right)^{-1} \epsilon_k^p .$$

- All this can be formulated as the RLS algorithm:

$$\left\{ \begin{array}{l} \epsilon_k^p = x_k - H_{k-1}^T Y_k \\ \epsilon_k = \epsilon_k^p \left(1 + Y_k^T R_{k-1}^{-1} Y_k \right)^{-1} \\ H_k = H_{k-1} + R_{k-1}^{-1} Y_k \epsilon_k \\ R_k^{-1} = R_{k-1}^{-1} - R_{k-1}^{-1} Y_k \left(1 + Y_k^T R_{k-1}^{-1} Y_k \right)^{-1} Y_k^T R_{k-1}^{-1} . \end{array} \right.$$



Recursive Least-Squares Algorithm (4)

- The initial values for R_k^{-1} and H_k are R_0^{-1} and H_0 . Compared to the LMS algorithm, the scalar stepsize μ gets replaced by a matrix stepsize R_k^{-1} . The RLS algorithm takes $\mathcal{O}(N^2)$ operations while the LMS algorithm takes only $2N$ operations. However, it converges much faster.
- *performance analysis* : with $x_k = H^{oT}Y_k + \tilde{x}_k$ (and $R_0 = 0$), we get

$$R_k H_k = P_k = \sum_{i=1}^k Y_i x_i = R_k H^o + \sum_{i=1}^k Y_i \tilde{x}_i . \text{ Hence}$$

$$\widetilde{H}_k = H^o - H_k = -R_k^{-1} \sum_{i=1}^k Y_i \tilde{x}_i$$

From this, we obtain

$$C_k \stackrel{\triangle}{=} E \widetilde{H}_k \widetilde{H}_k^T = \sigma_{\tilde{x}}^2 R_k^{-1} .$$

Since R_k^{-1} behaves as $1/k$, we see that C_k converges to zero as $1/k$.

- *Exponential Weighting* In order to be able to track a possibly time-varying $H^o = H_k^o$, one introduces an exponential forgetting factor $\lambda \in (0, 1)$ into the cost function to obtain

$$\xi_k(H) = \sum_{i=1}^k \lambda^{k-i} (x_i - H^T Y_i)^2 + \lambda^k (H - H_0)^T R_0 (H - H_0) .$$

This implies that the past (and in particular the initial conditions H_0, R_0) is forgotten exponentially fast with a window with time constant $1/(1-\lambda)$.



Recursive Least-Squares Algorithm (5)

- Wiener filtering: x_k and y_k are two joint stochastic processes and we're trying to estimate x_k from the y_k using a LMMSE estimator. For an FIR Wiener filter, there are a finite set of coefficients H^o involved in this LMMSE estimator.
RLS approach: replaced statistical averages with temporal averages.

Parameter estimation interpretation

- Assume now our usual model for the *measurements* x_k ,

$$x_k = H^{oT} Y_k + \tilde{x}_k$$

where the \tilde{x}_k are iid with zero mean and variance $\sigma_{\tilde{x}}^2$. Consider here $\{y_k\}$ as a deterministic signal, so the only randomness comes from the $\{\tilde{x}_k\}$. The H^o are the unknown parameters governing the model.

- The analysis of the RLS algorithm is much simpler than that of the LMS algorithm since for each k the RLS solution H_k coincides with the solution of a Least-Squares problem with a closed-form solution: $H_k = R_k^{-1} P_k$.
- Assume now $k \geq N$, $H_0 = 0$, $R_0 = 0$, and that R_k is nonsingular. The performance of the least-squares estimate is simple to analyze and leads to

$$C_k \triangleq E \widetilde{H}_k \widetilde{H}_k^T = \sigma_{\tilde{x}}^2 R_k^{-1} .$$

- If \tilde{x}_k Gaussian, $\Rightarrow H_k = \text{ML estimate of } H^o$ (efficient, $C_k = \text{CRB}$).



Recursive Least-Squares Algorithm (6)

A Bayesian Context - A Priori Information

- Instead of treating the filter coefficients H^o as unknown constant parameters, we could also consider H^o as a stochastic parameter vector about which we have some prior information, possibly from previous adaptive filtering experience. Assume now that, prior to obtaining the measurements x_1, x_2, \dots , we know that H^o has a distribution with mean $E H^o = H_0$ and covariance $E (H^o - H_0)(H^o - H_0)^T = C_0$. So now the randomness in the x_k comes from both the \tilde{x}_k and H^o .
- The problem formulation can now be recognized to be one of a *Bayesian Linear Model*. The AMMSE estimator can be shown to be the filter estimate resulting from the original RLS criterion with $R_0 = \sigma_{\tilde{x}}^2 C_0^{-1}$. $C_k = E \tilde{H}_k \tilde{H}_k^T$ now satisfies

$$C_k^{-1} = \sigma_{\tilde{x}}^{-2} R_k = \sigma_{\tilde{x}}^{-2} \sum_{i=1}^k Y_i Y_i^T + C_0^{-1} .$$

Note that C_k^{-1} is an increasing function of C_0^{-1} and hence C_k is a decreasing function of C_0^{-1} and hence of R_0 .

- So we see that H_0 and R_0 in the LS cost function have the interpretation of the prior mean and the inverse of the prior covariance of H^o . We'll choose R_0 small if we don't have a lot of confidence in our prior guess H_0 (C_0 big). In practice, R_0 is often chosen as $R_0 = \eta I_N$.



Other Adaptive Filtering Algorithms

- Fast RLS algorithms: Fast Transversal Filter (FTF) algorithm ($8N$), Fast Lattice/QR Algorithms ($\mathcal{O}(N)$ complexity)
- LMS with prewhitened input
- block processing/frequency domain LMS
- subband structures
- Fast Newton Transversal Filter (FNTF): replace R^{-1} in RLS by a banded matrix (appropriate for AR processes, hence speech)
- projection algorithms (like NLMS) on an extended subspace of L input vectors (FAP: Fast Affine Projection: complexity $2N + \mathcal{O}(L^2)$ or $2N + \mathcal{O}(L)$)
- Fast Subsampled Updating (FSU) versions of LMS and FTF: introduce some delay to reduce complexity below $\mathcal{O}(N)$
- multistage Wiener filter / polynomial expansion:

$$r_0 R^{-1} = \left[\frac{1}{r_0} R \right]^{-1} = \left[\underbrace{I}_{\text{diagonal}} + \underbrace{\left(\frac{1}{r_0} R - I \right)}_{\text{off-diagonal part}} \right]^{-1} = \sum_{i=0}^{\infty} \left(I - \frac{1}{r_0} R \right)^i = \sum_{i=0}^{\infty} \alpha_i R^i$$

- convergence speed (RLS best) versus tracking speed (FAP best?)

Initial Convergence RLS

- Consider now $H_0 \neq 0, R_0 \neq 0$,

$$R_k = R_0 + R_{1:k}, R_{1:k} = Y_{1:k}Y_{1:k}^T, Y_{1:k} = [Y_1 \cdots Y_k], P_k = R_0H_0 + P_{1:k}$$

- $\tilde{H}_k = H^o - H_k = H^o - R_k^{-1}P_k = (R_0 + R_{1:k})^{-1}(R_0\bar{H}_0 - \sum_{i=1}^k Y_i\tilde{x}_i)$

- $C_k = E \tilde{H}_k \tilde{H}_k^T$ hence

$$\begin{aligned} C_k &= (R_0 + R_{1:k})^{-1}R_0\bar{H}_0\bar{H}_0^TR_0(R_0 + R_{1:k})^{-1} + \sigma_{\tilde{x}}^2(R_0 + R_{1:k})^{-1}R_{1:k}(R_0 + R_{1:k})^{-1} \\ &= \underbrace{\sigma_{\tilde{x}}^2(R_0 + R_{1:k})^{-1}}_{\sim \frac{1}{k}} + \underbrace{(R_0 + R_{1:k})^{-1}(R_0\bar{H}_0\bar{H}_0^TR_0 - \sigma_{\tilde{x}}^2R_0)(R_0 + R_{1:k})^{-1}}_{\sim \frac{1}{k^2} \text{ due to initialization}} \end{aligned}$$

- noiseless case: $\sigma_{\tilde{x}}^2 = 0$

$$C_k = (R_0 + R_{1:k})^{-1}R_0\bar{H}_0\bar{H}_0^TR_0(R_0 + R_{1:k})^{-1}$$

- initial convergence: $1 \leq k < N$, consider $R_0 = \eta I$

$$C_k = \eta^2 (\eta I + R_{1:k})^{-1}\bar{H}_0\bar{H}_0^T(\eta I + R_{1:k})^{-1}$$

Initial Convergence RLS (2)

- *Singular Value Decomposition (SVD):* $Y_{1:k}$, $N \times k$ assumed full column rank

$$Y_{1:k} = V\Sigma U^T \quad V^T V = I_k, U^{-1} = U^T, \Sigma = \text{diag}\{\sigma_1, \dots, \sigma_k\}$$

$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k \geq 0$ “singular values” full column rank $\leftrightarrow \sigma_k > 0$

- *Moore-Penrose pseudo-inverse:* $Y_{1:k}^+ = U\Sigma^{-1}V^T = (Y_{1:k}^T Y_{1:k})^{-1} Y_{1:k}^T$

- projection on column space: $P_{Y_{1:k}} = Y_{1:k} Y_{1:k}^+ = VV^T$

- $V^+ = V^T \quad P_{Y_{1:k}} = VV^T = VV^+ = P_V \quad P_V^+ = P_V$

- eigendecomposition: $R_{1:k} = Y_{1:k} Y_{1:k}^T = V\Sigma^2 V^T$

- let V^\perp be such that $[V \ V^\perp]$ is orthogonal:

$$[V \ V^\perp][V \ V^\perp]^T = I = VV^T + V^\perp V^{\perp T} = P_V + P_{V^\perp} = P_V + P_V^\perp$$

$P_V^\perp = I - P_V = P_{V^\perp}$, V^\perp spans orthogonal complement of V

- SVD alternatively: $Y_{1:k} = [V \ V^\perp] \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} U^T$, $\begin{bmatrix} \Sigma \\ 0 \end{bmatrix}^+ = [\Sigma^+ \ 0]$, $\sigma^+ = \begin{cases} 1/\sigma & , \sigma > 0 \\ 0 & , \sigma = 0 \end{cases}$
- eigendecomposition projection:

$$P_V = V I V^T = [V \ V^\perp] \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} [V \ V^\perp]^T \quad \text{eigenvalues are 1 or 0}$$

Initial Convergence RLS (3)

- let $\eta \ll \sigma_k^2$ be small, then

$$\begin{aligned}
\eta(\eta I + R_{1:k})^{-1} &= \eta(\eta V^\perp V^{\perp T} + \eta VV^T + V\Sigma^2V^T)^{-1} \approx \eta(\eta V^\perp V^{\perp T} + V\Sigma^2V^T)^{-1} \\
&= \eta \left([V \ V^\perp] \begin{bmatrix} \Sigma^2 & 0 \\ 0 & \eta I \end{bmatrix} [V \ V^\perp]^T \right)^{-1} = \eta [V \ V^\perp] \begin{bmatrix} \Sigma^2 & 0 \\ 0 & \eta I \end{bmatrix}^{-1} [V \ V^\perp]^T \\
&= \eta V\Sigma^{-2}V^T + V^\perp V^{\perp T} = \eta R_{1:k}^+ + \mathbf{P}_{R_{1:k}}^\perp
\end{aligned}$$

where $R_{1:k}^+ = Y_{1:k}(Y_{1:k}^T Y_{1:k})^{-2} Y_{1:k}^T$ and $\mathbf{P}_{R_{1:k}} = \mathbf{P}_{Y_{1:k}}$

- hence

$$C_k = (\eta R_{1:k}^+ + \mathbf{P}_{R_{1:k}}^\perp) \widetilde{H}_0 \widetilde{H}_0^T (\eta R_{1:k}^+ + \mathbf{P}_{R_{1:k}}^\perp)$$

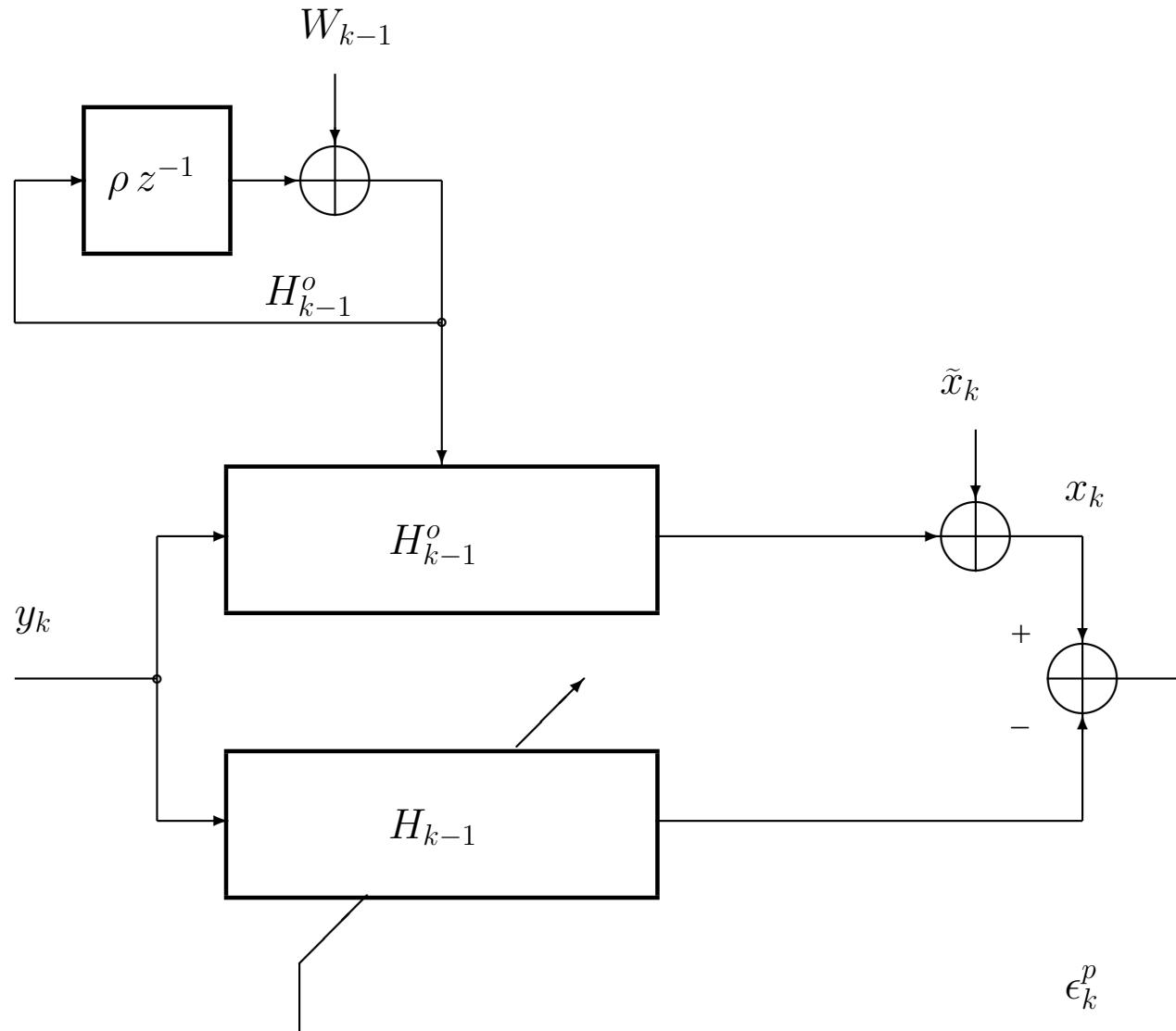
$\eta R_{1:k}^+ \widetilde{H}_0$ reduced to $\mathcal{O}(\eta)$ in k -dim. subspace, column space of $Y_{1:k}$

$\mathbf{P}_{R_{1:k}}^\perp \widetilde{H}_0$ unchanged in $(N - k)$ -dim. orthogonal complement

- C_k rank 1 (noiseless case): only the mean of \widetilde{H}_k needs to converge
- RLS: the mean of \widetilde{H}_k has essentially converged (filter estimate unbiased) after $k = N \Rightarrow$ very fast (mean dominates initial convergence in general)

LMS: the mean needs to converge exponentially, dynamics of steepest-descent

Tracking Time-Varying Filters



Time-Varying System Identification Set-Up

- system processes:

$$\begin{aligned}x_k &= Y_k^T H_{k-1}^o + \tilde{x}_k & E\tilde{x}_k\tilde{x}_i &= \xi^o \delta_{ki} \\H_k^o &= \rho H_{k-1}^o + W_k & EW_k W_i^T &= Q \delta_{ki} \\\widetilde{H}_k &= H_k^o - H_k & EW_k \tilde{x}_i &= 0\end{aligned}$$

- time-varying filter modeled as AR(1) process, requires $|\rho| < 1$ for stationarity
→ stationary case of nonstationarity
- adaptive filter a priori error signal:

$$\epsilon_k^p = x_k - Y_k^T H_{k-1} = Y_k^T \widetilde{H}_{k-1} + \tilde{x}_k$$

- learning curve: (independence assumption)

$$\xi_k = E(\epsilon_k^p)^2 = \xi^o + \xi_k^e = \xi^o(1 + \mathcal{M}), \quad \xi_k^e = \text{tr}\{R_{YY} C_{k-1}\}, \quad C_k = E \widetilde{H}_k \widetilde{H}_k^T$$

- consider $\frac{1}{1-\rho} \gg$ adaptation time constants so that we can take $\rho = 1$ for the analysis

Tracking Analysis LMS

- filter deviation recursion:

$$\begin{aligned}\widetilde{H}_k &= \widetilde{H}_{k-1} - \mu \epsilon_k^p Y_k + W_k \\ &= (I - \mu Y_k Y_k^T) \widetilde{H}_{k-1} - \mu \widetilde{x}_k Y_k + W_k \\ &\approx (I - \mu R_{YY}) \widetilde{H}_{k-1} - \mu \widetilde{x}_k Y_k + W_k\end{aligned}$$

where we introduced the averaging approach in the last step

- filter error correlation matrix recursion:

$$C_k = (I - \mu R_{YY}) C_{k-1} (I - \mu R_{YY}) + \mu^2 \xi^o R_{YY} + Q$$

- the stationary nonstationarity combined with a constant stepsize leads to a steady-state, for which we get (with small μ):

$$R_{YY} C_\infty + C_\infty R_{YY} = \mu \xi^o R_{YY} + \frac{1}{\mu} Q$$

- steady-state misadjustment: $\mathcal{M}_{LMS} = \underbrace{\frac{\mu}{2} \text{tr} R_{YY}}_{\text{estimation noise}} + \underbrace{\frac{1}{2\mu\xi^o} \text{tr} Q}_{\text{lag noise}}$

Tracking Analysis RLS

- filter deviation recursion: $\lambda < 1$ to allow tracking

$$\begin{aligned}\widetilde{H}_k &= \widetilde{H}_{k-1} - R_k^{-1} Y_k \epsilon_k^p + W_k \\ &= (I - R_k^{-1} Y_k Y_k^T) \widetilde{H}_{k-1} - R_k^{-1} Y_k \bar{x}_k + W_k\end{aligned}$$

- after averaging in steady-state, assuming small $1 - \lambda$

$$(I - R_k^{-1} Y_k Y_k^T = \lambda R_k^{-1} R_{k-1} \approx \lambda I, \quad R_k^{-1} \approx (1 - \lambda) R_{YY}^{-1})$$

$$\widetilde{H}_k = \lambda \widetilde{H}_{k-1} - (1 - \lambda) R_{YY}^{-1} Y_k \bar{x}_k + W_k \quad (\text{dynamics indep. of } R_{YY})$$

- filter error correlation matrix recursion:

$$C_k = \lambda^2 C_{k-1} + (1 - \lambda)^2 \xi^o R_{YY}^{-1} + Q$$

- which leads to the steady-state value (assuming small $1 - \lambda$)

$$C_\infty = \frac{1 - \lambda}{2} \xi^o R_{YY}^{-1} + \frac{1}{2(1 - \lambda)} Q$$

- steady-state misadj.: $\mathcal{M}_{RLS} = \underbrace{\frac{1 - \lambda}{2} N}_{\text{estimation noise}} + \underbrace{\frac{1}{2(1 - \lambda)\xi^o} \text{tr}\{R_{YY}Q\}}_{\text{lag noise}}$

Tracking Optimization & LMS-RLS Comparison

- stepsize μ , $1 - \lambda$ design result of compromise between:
 - estimation noise: finite stepsize prevents convergence, consistency
 - lag noise: small stepsize leads to lowpass filtering and to a filter estimate that lags behind the true filter

- LMS: $\mu^{opt} = \sqrt{\frac{\text{tr}Q}{\xi^o \text{tr}R_{YY}}}$, $\mathcal{M}_{LMS}^{opt} = \sqrt{\frac{\text{tr}R_{YY} \text{tr}Q}{\xi^o}}$

- RLS: $\lambda^{opt} = 1 - \sqrt{\frac{\text{tr}\{R_{YY}Q\}}{N \xi^o}}$, $\mathcal{M}_{RLS}^{opt} = \sqrt{\frac{N \text{tr}\{R_{YY}Q\}}{\xi^o}}$

- comparison:

$$\frac{\mathcal{M}_{LMS}^{opt}}{\mathcal{M}_{RLS}^{opt}} = \sqrt{\frac{\text{tr}R_{YY} \text{tr}Q}{N \text{tr}\{R_{YY}Q\}}}$$

$$Q = \begin{cases} q I & : \text{equal performance, at least for small } q \\ q R_{YY} & : \text{LMS is better} \\ q R_{YY}^{-1} & : \text{RLS is better} \end{cases}$$

- faster initial convergence of RLS could be exploited for *jumping* parameters, if windowsize properly adapted



Optimal Tracking via Kalman Filtering

- *state-space model:*

state = AR(1) vector process

$$\text{state equation} \quad H_k^o = A_k H_{k-1}^o + W_k$$

$$\text{measurement equation} \quad x_k = Y_k^T H_{k-1}^o + \tilde{x}_k \quad , \quad E \begin{bmatrix} W_k \\ \tilde{x}_k \end{bmatrix} \begin{bmatrix} W_i \\ \tilde{x}_i \end{bmatrix}^T = \begin{bmatrix} Q_k & P_k^T \\ P_k & \xi_k^o \end{bmatrix} \delta_{ki}$$

time-varying (at the very least due to Y_k), usually $P_k = 0$

state noise: W_k , measurement noise: \tilde{x}_k , state transition matrix A_k

- Kalman filter (KF): estimates recursively in time the *state* H_{k-1}^o on the basis of the *measurements* x_0, \dots, x_k in a LMMSE sense.
Sources of randomness: W_k, \tilde{x}_k . Signal y_k treated as deterministic.
- special case: $H_k^o = H_{k-1}^o \Rightarrow$ Kalman filter \rightarrow RLS algorithm
- RLS with exponential weighting can be interpreted as KF for the case of some non-zero Q_k
- in the time-invariant case (x_k and H_k^o jointly stationary apart from initial conditions), the KF converges to the causal Wiener filter.



Chapter 5: Sinusoids in White Noise

- $x_k = \sum_{i=1}^M A_i \cos(\omega_i k + \phi_i), \quad y_k = x_k + v_k \quad \omega_i = 2\pi f_i$

- the support of $S_{xx}(f) = \sum_{i=1}^M \frac{A_i^2}{4} (\delta(f - f_i) + \delta(f + f_i))$ has measure zero,

R_{XX} is singular for dimension $> 2M$

- $P(q)x_k = 0, \quad P(z) = \prod_{i=1}^M (1 - 2 \cos \omega_i z^{-1} + z^{-2}), \quad q^{-1}x_k = x_{k-1}$

- Hence, x_k is perfectly predictable from the previous $2M$ samples. $P(z)$ and hence the ω_i can be found by linear prediction: *Prony* method.
(Baron Prony, 18th century)

Normal equations:

$$P R_{XX} = [0 \cdots 0 \ \sigma^2], \quad \sigma^2 = 0$$

where

$$\begin{aligned} R_{XX} &= E XX^T \\ X &= [x_0 \cdots x_{2M}]^T \\ P &= [P_{2M} \cdots P_1 \ P_0] \\ P_0 &= 1 \\ P_i &= P_{2M-i}, \quad i = 0, \dots, M-1 \end{aligned}$$



Sinusoids in Noise: Signal and Noise Subspaces

- signal structure

$$X_k = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 1 & 0 \\ \cos \omega_1 & \sin \omega_1 & \cdots & \cos \omega_M & \sin \omega_M \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \cos \omega_1 k & \sin \omega_1 k & \cdots & \cos \omega_M k & \sin \omega_M k \end{bmatrix} \begin{bmatrix} A_1 \cos \phi_1 \\ -A_1 \sin \phi_1 \\ \vdots \\ A_M \cos \phi_M \\ -A_M \sin \phi_M \end{bmatrix} = \mathcal{V} S$$

- one calls

$$\begin{aligned} \text{Range } \{\mathcal{V}\} &= \text{signal subspace} \\ (\text{Range } \{\mathcal{V}\})^\perp &= \text{noise subspace} \end{aligned}$$

- covariance structure:

$$Y_k = X_k + V_k = \mathcal{V}_k S + V_k \Rightarrow R_{YY} = \mathcal{V} R_{SS} \mathcal{V}^T + \sigma_v^2 I$$

if angles uniform and uncorrelated:

$$R_{SS} = \frac{1}{2} \text{diag} \{A_1^2, A_1^2, \dots, A_M^2, A_M^2\}$$



Sinusoids in Noise: Signal and Noise Subspaces (2)

- Consider the eigendecomposition of R_{YY} ($\lambda_1 \geq \lambda_2 \geq \dots$):

$$R_{YY} = \sum_{i=1}^{2M} \lambda_i V_i V_i^T + \sum_{i=2M+1}^{k+1} \lambda_i V_i V_i^T = V_S \Lambda_S V_S^T + V_N \Lambda_N V_N^T$$

where $\Lambda_N = \sigma_v^2 I_{k+1-2M}$.

- Assuming \mathcal{V} and R_{SS} to have full rank, the sets of eigenvectors V_S and V_N are orthogonal: $V_S^T V_N = 0$, and $\lambda_i > \sigma_v^2$, $i = 1, \dots, 2M$.
- Equivalent descriptions of the signal and noise subspaces:

$$\text{Range}\{V_S\} = \text{Range}\{\mathcal{V}\}, \quad V_N^T \mathcal{V} = 0$$

- Linear prediction in the noisy case: minimize variance subject to norm constraint: *Pisarenko* method
with $k = 2M$: noise subspace dimension = 1

$$\min_{\|P\|=1} P R_{YY} P^T = \min_{\|P\|=1} P R_{XX} P^T + \sigma_v^2 \Rightarrow P R_{XX} = [0 \cdots 0], \quad P^T = V_{2M+1}$$



Sinusoids in Noise: Signal Subspace Fitting

- two equivalent signal subspace descriptions: \mathcal{V} and $V_{\mathcal{S}}$
- with an estimated covariance matrix, $V_{\mathcal{S}}$ is approximate, so consider

$$\min_{\omega, T} \|\mathcal{V}(\omega) - V_{\mathcal{S}}T\|_F^2 \quad \|A\|_F^2 = \text{tr } AA^T$$

where $\omega = [\omega_1 \cdots \omega_M]$.

- separable problem: orthogonality of LS: $V_{\mathcal{S}}^T(\mathcal{V} - V_{\mathcal{S}}T) = 0 \Rightarrow T = V_{\mathcal{S}}^T\mathcal{V}$
- $\mathcal{V} - V_{\mathcal{S}}T = (I - V_{\mathcal{S}}V_{\mathcal{S}}^T)\mathcal{V} = (I - P_{V_{\mathcal{S}}})\mathcal{V} = P_{V_{\mathcal{S}}}^\perp \mathcal{V}$
- projection on column space of X : $P_X = X(X^T X)^{-1}X^T$, $P = P^T$, $PP = P$

$$\|P_{V_{\mathcal{S}}}^\perp \mathcal{V}\|_F^2 = \text{tr } \mathcal{V}^T P_{V_{\mathcal{S}}}^\perp \mathcal{V} = \text{tr } \mathcal{V}^T P_{V_{\mathcal{N}}} \mathcal{V} = \|V_{\mathcal{N}}^T \mathcal{V}\|_F^2$$

- hence
- $$= \sum_{i=2M+1}^k \|V_i^T \mathcal{V}\|^2 = \sum_{j=1}^M \sum_{i=2M+1}^k |V_i(\omega_j)|^2 \quad \text{multi-D optim.}$$

- approximate solution: plot as a function of ω and find M largest peaks of

$$\frac{1}{\sum_{i=2M+1}^k |V_i(\omega)|^2} \quad \text{MUSIC!}$$



Sinusoids in Noise: Noise Subspace Parameterization

- $P(q) \cos \omega_i k = 0, P(q) \sin \omega_i k = 0, \Rightarrow \mathcal{G}(P)^T \mathcal{V} = 0$ where

$$\mathcal{G}^T(P) = \begin{bmatrix} P_0 & P_1 & \cdots & P_{2M} & 0 & \cdots & 0 \\ 0 & P_0 & P_1 & \cdots & P_{2M} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & & \ddots & \vdots \\ 0 & \cdots & 0 & P_0 & P_1 & \cdots & P_{2M} \end{bmatrix} \quad \text{Toeplitz , } (k-2M) \times (k+1)$$

- noise subspace fitting:

$$\min_{P,T} \|\mathcal{G}(P) - V_{\mathcal{N}} T\|_F$$

- separable problem $\Rightarrow T = V_{\mathcal{N}}^T \mathcal{G}, \mathcal{G} - V_{\mathcal{N}} T = \mathbf{P}_{V_{\mathcal{N}}}^{\perp} \mathcal{G}$ and hence

$$\|\mathbf{P}_{V_{\mathcal{N}}}^{\perp} \mathcal{G}\|_F^2 = \operatorname{tr} \mathcal{G}^T \mathbf{P}_{V_{\mathcal{N}}}^{\perp} \mathcal{G} = \operatorname{tr} \mathcal{G}^T \mathbf{P}_{V_{\mathcal{S}}} \mathcal{G} = \|V_{\mathcal{S}}^T \mathcal{G}\|_F^2$$

$$= \sum_{i=1}^{2M} \|\mathcal{G}^T V_i\|^2$$

Let $\mathcal{G}^T V_i = \mathcal{W}_i P^T$ where $\mathcal{W}_i = \mathcal{W}(V_i)$ is Hankel, then we get (with $P = P J$)

$$\min_P P \left[\left(\sum_{i=1}^{2M} \mathcal{W}_i^T \mathcal{W}_i \right) + J \left(\sum_{i=1}^{2M} \mathcal{W}_i^T \mathcal{W}_i \right) J \right] P^T$$

subject to $P_0 = 1$ or $\|P\| = 1$.



Sinusoids in Noise: Maximum Likelihood Estimation

- additive noise v_k white and Gaussian \rightarrow likelihood criterion

$$\min_{\omega, S} \|Y - \mathcal{V}(\omega) S\|^2$$

- separable: $\mathcal{V}^T(Y - \mathcal{V}S) = 0 \Rightarrow S = (\mathcal{V}^T\mathcal{V})^{-1}\mathcal{V}^TY$

$$\Rightarrow \|Y - \mathcal{V}S\|^2 = Y^T \mathbf{P}_{\mathcal{V}}^\perp Y = Y^T \mathbf{P}_{\mathcal{G}(P)} Y = P \mathcal{Y}^T (\mathcal{G}(P)^T \mathcal{G}(P))^{-1} \mathcal{Y} P^T$$

where $\mathcal{G}(P)Y = \mathcal{Y}(Y)P^T$ (commutativity of convolution, \mathcal{Y} Hankel)

- IQML (Iterative Quadratic Maximum Likelihood), iteration n :

$$\min_{P^{(n)}} P^{(n)} \mathcal{Y}^T (\mathcal{G}(P^{(n-1)})^T \mathcal{G}(P^{(n-1)}))^{-1} \mathcal{Y} P^{(n)T}$$

subject to $P_0 = 1$ or $\|P\| = 1$

- with a consistent initialization, only one iteration is required to get a BAN estimate at high SNR
- denoised IQML : $Y^T \mathbf{P}_{\mathcal{G}(P)} Y = \text{tr} \left\{ \mathbf{P}_{\mathcal{G}(P)} Y Y^T \right\} \rightarrow \text{tr} \left\{ \mathbf{P}_{\mathcal{G}(P)} (Y Y^T - \widehat{\sigma}_v^2 I) \right\}$
- Pseudo-QML (PQML): $\frac{\partial}{\partial P} Y^T \mathbf{P}_{\mathcal{G}(P)} Y = 2Q(P)P \rightarrow 2Q(\widehat{P})P = \frac{\partial}{\partial P} P^T Q(\widehat{P})P$



Sinusoids in Noise: Adaptive Notch Filtering

- notch filter model

$$P(q)x_k = 0 \Rightarrow P(q)y_k = P(q)v_k \Rightarrow v_k = \frac{P(q)}{P(q/\rho)} y_k \text{ as } \rho \rightarrow 1$$

- notch filter output

$$\epsilon_k = H(q) y_k = H(q)x_k + H(q)v_k , \quad H(q) = \frac{P(q)}{P(q/\rho)}$$

- notch filter $H(z)$: zeros = $e^{\pm j\omega_i}$, poles = $\rho e^{\pm j\omega_i}$,
- notch filter output variance ($H(f) = H(e^{j2\pi f})$)

$$E \epsilon_k^2 = \int_{-\frac{1}{2}}^{\frac{1}{2}} |\mathbf{H}(f)|^2 S_{xx}(f) df + \int_{-\frac{1}{2}}^{\frac{1}{2}} |\mathbf{H}(f)|^2 S_{vv}(f) df = \sum_{i=1}^M \frac{A_i^2}{2} |\mathbf{H}(f_i)|^2 + \sigma_v^2$$

- notch adaptation by output variance minimization

$$\min_P E \epsilon_k^2$$

adaptively: Recursive Prediction Error Method (RPEM): solve ML recursively

Statistical Signal Processing

Lecture 11

Statistical Signal Modeling, Learning and Processing

- chapter 4: Adaptive Filtering

Tracking stationary Time-Varying Parameters

- model, LMS & RLS performance
- optimal approach: Kalman filtering

- chapter 3: Optimal Filtering

Kalman Filtering

- state-space models
- basic Kalman filter derivation
- extensions

- chapter 5: Sinusoids in Noise prototype problem

Outline

1 State-Space Models

2 Kalman Filter (KF) derivation

3 Kalman Filter

4 Kalman and Wiener

5 Extensions

State-Space Model

- The signal model can be written as

state update equation:

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{G}_k \mathbf{w}_k \quad (1)$$

measurement equation:

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k$$

where

$$E \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{w}_k \\ \mathbf{v}_k \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{x}}_0 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \quad E \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{w}_k \\ \mathbf{v}_k \end{bmatrix} \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{w}_l \\ \mathbf{v}_l \end{bmatrix}^T = \begin{bmatrix} \mathbf{P}_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\delta_{kl}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_{\delta_{kl}} \end{bmatrix} \quad (2)$$

- State vector \mathbf{x}_k : the state summarizes the past to produce the current output \mathbf{y}_k .
- State update: vector AR(1) process.
- Kalman filter: recursive LMMSE estimation of \mathbf{x}_k on basis of $\mathbf{y}_{1:k}$.

State-Space Model example: AR(n) in noise

- Consider an AR(n) signal in white noise:

$$\begin{aligned}s_k &= -\sum_{i=1}^n a_i s_{k-i} + w_{k-1} \\ y_k &= s_k + v_k\end{aligned}\tag{3}$$

- corresponding n -dimensional state-space model:

$$\begin{aligned}\mathbf{x}_k &= \begin{bmatrix} s_k \\ s_{k-1} \\ \vdots \\ s_{k-n+1} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_{n-1} & -a_n \\ 1 & 0 & \cdots & & 0 \\ & \ddots & & & \vdots \\ 0 & \cdots & & 1 & 0 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \\ \mathbf{H} &= [1 \ 0 \ \cdots \ 0]\end{aligned}\tag{4}$$

- State-space model not unique: if \mathbf{T} invertible,
 $\mathbf{x}'_k = \mathbf{T}\mathbf{x}_k$, $\mathbf{F}' = \mathbf{TFT}^{-1}$, $\mathbf{G}' = \mathbf{TG}$, $\mathbf{H}' = \mathbf{HT}^{-1}$
also valid state-space model for y_k .
Depends whether \mathbf{x}_k (components) have meaning.

State-Space Model example: Position Tracking

White Noise Acceleration

Let \mathbf{p}_k be the position at sampling instant k , \mathbf{v}_k the velocity (not to be confused with the measurement noise) and \mathbf{a}_k the acceleration. In the case of e.g. 3D positioning, \mathbf{p}_k is of the form $\mathbf{p} = [x \ y \ z]^T$. By simple discretization of the differential equations of motion, we get

$$\begin{aligned}\mathbf{p}_{k+1} &= \mathbf{p}_k + \mathbf{v}_k \\ \mathbf{v}_{k+1} &= \mathbf{v}_k + \mathbf{a}_k \\ \mathbf{a}_k &= \mathbf{w}_k.\end{aligned}\tag{5}$$

In the case of modeling the acceleration as (temporally) white noise, the acceleration is the process noise. To simplify the equations, we assume here that the unit of time for velocity and acceleration is the sampling period. The physical speed and acceleration are then $t_s \mathbf{v}_k$ and $t_s^2 \mathbf{a}_k$ where t_s is the sampling period expressed in seconds, assuming \mathbf{p}_k is expressed in meters. We get for the state-space model

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{p}_k \\ \mathbf{v}_k \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{I}_n & \mathbf{I}_n \\ \mathbf{0}_{n,n} & \mathbf{I}_n \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{0}_{n,n} \\ \mathbf{I}_n \end{bmatrix}, \quad \mathbf{y}_k = \hat{\mathbf{p}}_k, \quad \mathbf{H} = [\mathbf{I}_n \ \mathbf{0}_{n,n}] \tag{6}$$

where $\mathbf{0}_{n,m}$ is a $n \times m$ matrix of zeros.

The only unknown system parameter in this case is the acceleration covariance matrix \mathbf{Q} .

State-Space Model example: Position Tracking

AR(1) (Markov) Acceleration

In this case we assume a first-order autoregressive model for the acceleration $\mathbf{a}_{k+1} = \mathbf{A} \mathbf{a}_k + \mathbf{w}_k$ where now \mathbf{A} and \mathbf{Q} are unknown (need to be estimated also).

We have (pseudo-inverse) $\mathbf{G}^+ = [\mathbf{0}_{n,2n} \ \mathbf{I}_n]$ and $\mathbf{a}_k = \mathbf{G}^+ \mathbf{x}_k$. Note that $\mathbf{G}^+ \mathbf{F} = \mathbf{A} \mathbf{G}^+$.

We get for the state-space model

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{p}_k \\ \mathbf{v}_k \\ \mathbf{a}_k \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{I}_n & \mathbf{I}_n & \mathbf{0}_{n,n} \\ \mathbf{0}_{n,n} & \mathbf{I}_n & \mathbf{I}_n \\ \mathbf{0}_{n,n} & \mathbf{0}_{n,n} & \mathbf{A} \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{0}_{n,n} \\ \mathbf{0}_{n,n} \\ \mathbf{I}_n \end{bmatrix}, \quad \mathbf{H} = [\mathbf{I}_n \ \mathbf{0}_{n,2n}]. \quad (7)$$

Outline

1 State-Space Models

2 Kalman Filter (KF) derivation

3 Kalman Filter

4 Kalman and Wiener

5 Extensions

KF derivation: Innovations approach

- notation $\mathbf{y}_{1:k} = \{\mathbf{y}_1, \dots, \mathbf{y}_k\}$.
- The KF performs **Gram-Schmidt orthogonalization** (decorrelation) of the measurements \mathbf{y}_k .
 $\tilde{\mathbf{y}}_{k|k-1}$ = LMMSE predictor of \mathbf{y}_k on the basis of $\mathbf{y}_{1:k-1}$,
leading to the orthogonalized prediction error (or **innovation**) $\tilde{\mathbf{y}}_k = \tilde{\mathbf{y}}_{k|k-1} = \mathbf{y}_k - \tilde{\mathbf{y}}_{k|k-1}$.
- Correlation matrix notation $R_{\mathbf{x}\mathbf{y}} = \mathbb{E} \mathbf{x} \mathbf{y}^T$. Denote $R_{\tilde{\mathbf{y}}_k \tilde{\mathbf{y}}_k} = \mathbf{S}_k$.
- Innovations approach idea: linear estimation in terms of $\mathbf{y}_{1:k}$ is equivalent to estimation in terms of $\tilde{\mathbf{y}}_{1:k}$ since one set is obtained from the other by an invertible linear transformation. Now, since the $\tilde{\mathbf{y}}_k$ are decorrelated, estimation in terms of $\tilde{\mathbf{y}}_{1:k}$ simplifies:

$$\hat{\mathbf{x}}_{|k} = \sum_{i=1}^k R_{\mathbf{x}\tilde{\mathbf{y}}_i} R_{\tilde{\mathbf{y}}_i \tilde{\mathbf{y}}_i}^{-1} \tilde{\mathbf{y}}_i = \hat{\mathbf{x}}_{|k-1} + R_{\mathbf{x}\tilde{\mathbf{y}}_k} \mathbf{S}_k^{-1} \tilde{\mathbf{y}}_k . \quad (8)$$

Used to obtain **predicted** estimates $\hat{\mathbf{x}}_{k|k-1}$ with estimation error $\tilde{\mathbf{x}}_{k|k-1} = \mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}$ and error covariance matrix $\mathbf{P}_{k|k-1} = R_{\tilde{\mathbf{x}}_{k|k-1} \tilde{\mathbf{x}}_{k|k-1}}$ and also **filtered** estimates $\hat{\mathbf{x}}_{k|k}$ with estimation error $\tilde{\mathbf{x}}_{k|k} = \mathbf{x}_k - \hat{\mathbf{x}}_{k|k}$ and error covariance matrix $\mathbf{P}_{k|k} = R_{\tilde{\mathbf{x}}_{k|k} \tilde{\mathbf{x}}_{k|k}}$.

Kalman Filter: Time Update

- Notation: $\mathcal{L}\{\mathbf{y}_{1:k}\}$ = linear span of $\mathbf{y}_1, \dots, \mathbf{y}_k$ = linear vector space spanned by $\mathbf{y}_1, \dots, \mathbf{y}_k$ (using matrices of appropriate dimension as combination coefficients). Then e.g. $\mathcal{L}\{\mathbf{y}_{1:k}\} = \mathcal{L}\{\tilde{\mathbf{y}}_{1:k}\}$.
- $\mathbf{v}_k \perp \mathcal{L}\{\mathbf{v}_{1:k-1}\}$ means decorrelation of \mathbf{v}_k from the indicated space due to the whiteness of the measurement noise process.
- We have $\mathbf{x}_k \in \mathcal{L}\{\mathbf{x}_0, \mathbf{w}_{1:k-1}\}$, $\mathbf{y}_k \in \mathcal{L}\{\mathbf{x}_0, \mathbf{w}_{1:k-1}, \mathbf{v}_k\}$, hence
 $\mathcal{L}\{\mathbf{y}_{1:k-1}\} \subset \mathcal{L}\{\mathbf{x}_0, \mathbf{w}_{1:k-2}, \mathbf{v}_{1:k-1}\} \perp \mathbf{v}_k$.
- Hence

$$\begin{aligned}\hat{\mathbf{y}}_{k|k-1} &= \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{v}}_{k|k-1} = \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1} \\ \tilde{\mathbf{y}}_{k|k-1} &= \mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1} = \mathbf{H}_k \tilde{\mathbf{x}}_{k|k-1} + \mathbf{v}_k \\ \mathbf{S}_k &= R_{\tilde{\mathbf{y}}_{k|k-1} \tilde{\mathbf{y}}_{k|k-1}} = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k\end{aligned}\tag{9}$$

- $\hat{\mathbf{w}}_{k|k} = 0$ since $\mathcal{L}\{\mathbf{y}_{1:k}\} \subset \mathcal{L}\{\mathbf{x}_0, \mathbf{w}_{1:k-1}, \mathbf{v}_{1:k}\} \perp \mathbf{w}_k$. Hence

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} &= \mathbf{F}_k \hat{\mathbf{x}}_{k|k} + \mathbf{G}_k \hat{\mathbf{w}}_{k|k} = \mathbf{F}_k \hat{\mathbf{x}}_{k|k} \\ \tilde{\mathbf{x}}_{k+1|k} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k} = \mathbf{F}_k \tilde{\mathbf{x}}_{k|k} + \mathbf{G}_k \mathbf{w}_k \\ \mathbf{P}_{k+1|k} &= \mathbf{F}_k \mathbf{P}_{k|k} \mathbf{F}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T\end{aligned}\tag{10}$$

Kalman Filter: Measurement Update



$$\begin{aligned}
 \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + R_{\mathbf{x}_k \tilde{\mathbf{y}}_k} \mathbf{S}_k^{-1} \tilde{\mathbf{y}}_k \\
 R_{\mathbf{x}_k \tilde{\mathbf{y}}_k} &= \mathbb{E} \mathbf{x}_k (\mathbf{H}_k \tilde{\mathbf{x}}_{k|k-1} + \mathbf{v}_k)^T = \mathbb{E} \mathbf{x}_k \tilde{\mathbf{x}}_{k|k-1}^T \mathbf{H}_k^T + \mathbb{E} \mathbf{x}_k \mathbf{v}_k^T = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \\
 \text{Let } \mathbf{K}_k &= \mathbf{P}_{k|k-1} \mathbf{H}_k^T \mathbf{S}_k^{-1} \quad \text{Kalman gain} \\
 \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \tilde{\mathbf{y}}_k \tag{11} \\
 R_{\hat{\mathbf{x}}_{k|k} \hat{\mathbf{x}}_{k|k}} &= R_{\hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{x}}_{k|k-1}} + \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T \quad \text{because } \tilde{\mathbf{y}}_k \perp \mathcal{L}\{\tilde{\mathbf{y}}_{1:k}\} \\
 \mathbf{P}_{k|k} &= R_{\mathbf{x}_k \mathbf{x}_k} - R_{\hat{\mathbf{x}}_{k|k} \hat{\mathbf{x}}_{k|k}} = R_{\mathbf{x}_k \mathbf{x}_k} - R_{\hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{x}}_{k|k-1}} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T \\
 &= \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T
 \end{aligned}$$

- $R_{\mathbf{x}_k \mathbf{x}_k} = R_{\hat{\mathbf{x}}_{k|k} \hat{\mathbf{x}}_{k|k}} + \mathbf{P}_{k|k}$: orthogonality property LMMSE : $R_{\hat{\mathbf{x}} \hat{\mathbf{x}}} = \mathbf{0}$
- Note: cannot do $\tilde{\mathbf{x}}_{k|k} = \tilde{\mathbf{x}}_{k|k-1} - \mathbf{K}_k \tilde{\mathbf{y}}_k \Rightarrow \mathbf{P}_{k|k} \neq \mathbf{P}_{k|k-1} + \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T$ because $\mathbf{x}_k \not\perp \mathcal{L}\{\tilde{\mathbf{y}}_k\}$!

Outline

1 State-Space Models

2 Kalman Filter (KF) derivation

3 Kalman Filter

4 Kalman and Wiener

5 Extensions

Kalman Filter summary

- two-step recursive procedure to go from $|k-1$ to $|k$:

- Measurement Update**

$$\begin{aligned}\tilde{y}_{k|k-1} &= y_k - H_k \hat{x}_{k|k-1} \\ S_k &= H_k P_{k|k-1} H_k^T + R_k \\ K_k &= P_{k|k-1} H_k^T S_k^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k \tilde{y}_{k|k-1} \\ P_{k|k} &= P_{k|k-1} - K_k H_k P_{k|k-1}\end{aligned}\tag{12}$$

- Time Update (prediction)**

$$\begin{aligned}\hat{x}_{k+1|k} &= F_k \hat{x}_{k|k} \\ P_{k+1|k} &= F_k P_{k|k} F_k^T + G_k Q_k G_k^T\end{aligned}\tag{13}$$

- There are various other ways to formulate these update equations, including performing both steps in one step.

$$\begin{aligned}\hat{x}_{k+1|k} &= F_k(I - K_k H_k) \hat{x}_{k|k-1} + F_k K_k y_k \\ P_{k+1|k} &= F_k P_{k|k-1} F_k^T + G_k Q_k G_k^T - F_k P_{k|k-1} H_k^T S_k^{-1} H_k P_{k|k-1} F_k^T\end{aligned}\text{Riccati equation}\tag{14}$$

- The choice of the initial conditions crucially affects the initial convergence (transient behavior). In the usual case of total absence of prior information on the initial state, one can choose $\hat{x}_{0|0} = \hat{x}_0 = 0$, $P_{0|0} = P_0 = p_0 I$ with p_0 a (very) large number.

Outline

1 State-Space Models

2 Kalman Filter (KF) derivation

3 Kalman Filter

4 Kalman and Wiener

5 Extensions

Kalman Filtering for causal Wiener Filtering

- time-invariant state-space model: ARMA signal in white noise
- steady-state Kalman filter solution leads to causal Wiener filter
- but the Kalman filter also applies to time-varying state-space models

Outline

1 State-Space Models

2 Kalman Filter (KF) derivation

3 Kalman Filter

4 Kalman and Wiener

5 Extensions

Kalman Filter variations

- error feedback, **stabilizing closed-loop**, controllability, observability
- **numerical stability**: symmetry and positive (semi-)definiteness of P_k , square-root Kalman filters, exponential forgetting
- **smoothing**: fixed-lag, fixed-interval
- **nonlinear filtering**: Extended Kalman Filter, ...
special case: adapting hyperparameters (when added to the state)

Statistical Signal Processing

Lecture 12

Statistical Signal Modeling, Learning and Processing

chapter 6: Learning Topics

- Expectation-Maximization (EM) algorithm
- compressed sensing and sparsity
- compressed sensing algorithms: LASSO etc
- Sparse Bayesian Learning (SBL), Empirical Bayes
- Variational Bayes (VB)

Outline

1 Parametric Signal Models

2 Compressed Sensing

3 SBL

Main Messages

- There are many Bayesian estimation problems, many of which are LMMSE (Wiener, Kalman), which contain hyperparameters to be tuned, using various approaches.
- Information combining: from weighted least-squares to message passing in a more general overall Bayesian formulation (e.g. cooperative location estimation)
- Empirical Bayes (EB) as principled framework for bias-variance trade-off
- but not necessarily using empirical Bayes for hyperparameter estimation: SURE, Cross Validation
- compressive sensing, sparse models, generalization of model order selection to support region, model complexity and structure
- Sparse Bayesian Learning (SBL) is one EB instance, allowing to exploit (approximate) sparsity for compressed sensing
 - can be extended to time-varying scenarios with sparse variations also
 - can be extended to dictionary learning, in particular with Kronecker structured dictionaries
- message passing (approximate iterative) inference techniques: easy to get the mean (estimate) correct but more difficult to get correct posterior variances

Kalman Filter

Linear state-space model:

state update equation:

$$\mathbf{x}_{k+1} = \mathbf{F}_k(\theta) \mathbf{x}_k + \mathbf{G}_k(\theta) \mathbf{w}_k$$

measurement equation:

$$\mathbf{y}_k = \mathbf{H}_k(\theta) \mathbf{x}_k + \mathbf{v}_k$$

for $k = 1, 2, \dots$, with uncorrelated

- initial state $\mathbf{x}_0 \sim \mathcal{N}(\hat{\mathbf{x}}_0, \mathbf{P}_0)$,
- measurement noise $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k(\theta))$,
- state noise $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k(\theta))$.

State model known up to some parameters θ .

Often $\mathbf{F}_k(\theta)$, $\mathbf{G}_k(\theta)$, $\mathbf{H}_k(\theta)$ linear in θ : bilinear case.



Numerous Applications

- LMMSE wireless channel estimation:

x_k = FIR filter response, θ : Power Delay Profile, AR(1) dynamics in e.g. diagonal F and Q

- Bayesian adaptive filtering (BAF):

analogous to LMMSE channel estimation, except measurement equation: only one 1D measurement is available per instance. An extremely simplified form of BAF is the so-called **Proportionate LMS (P-LMS)** algorithm.

- Position tracking (GPS):

$$\mathbf{x}_{t+1} = \begin{bmatrix} 1 & \Delta t & \frac{1}{2}\Delta t^2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{x}_t = \begin{bmatrix} x_t + \Delta t \cdot v_t + \frac{1}{2}\Delta t^2 a \\ v_t + \Delta t \cdot a \\ a \end{bmatrix}$$

θ : acceleration model parameters (e.g. white noise, AR(1))

- Blind Audio Source Separation (BASS):

x_k = source signals,

θ : (short+long term) AR parameters, reverb filters



Static LMMSE (Wiener) Applications

- Direction of Arrival (DoA) estimation: x = decorrelated sources, apart from the DoA parameters there could also be antenna array calibration parameters or source and noise covariance parameters.
- Blind and semi-blind channel estimation. In the techniques that exploit the (white) second-order statistics of x , (the unknown part of) x gets modeled as Gaussian. Numerous variations: single-carrier, OFDM and CDMA versions, single- and multi-user, single- and multi-stream, with black box FIR channel models or propagation based parameterized channel models.
Image Deblurring, NMRI Imaging
- LMMSE receiver (Rx) design: x = Tx symbol sequence to be recovered on the basis of Rx signal, in single- or multi-user settings and other variations as in the channel estimation case. The crosscorrelation between Tx and Rx signals depends on the channel response, which is part of the parameters. The Rx signal covariance matrix on the other hand can be modeled in various ways, non-parametric or parametric. Account for the channel estimation error in the LMMSE Rx design. Another approach: consider the LMMSE filter directly as the parameters.
LMMSE Tx design, partial CSIR/CSIT.

Adaptive Kalman Filter solutions

- Extended Kalman Filter (**EKF**)
- other generic nonlinear Kalman Filter extensions:
Unscented Kalman Filter (**UKF**), Cubature Kalman Filter (**CKF**), Gaussian Sum Filter, Particle Filter (**PF**)
- Recursive Prediction Error Method (**RPEM**) Kalman Filter
- Second-Order Extended Kalman Filter (**SOEKF**)
- Expectation-Maximization (**EM**)/Variational Bayes (**VB**) Kalman Filter

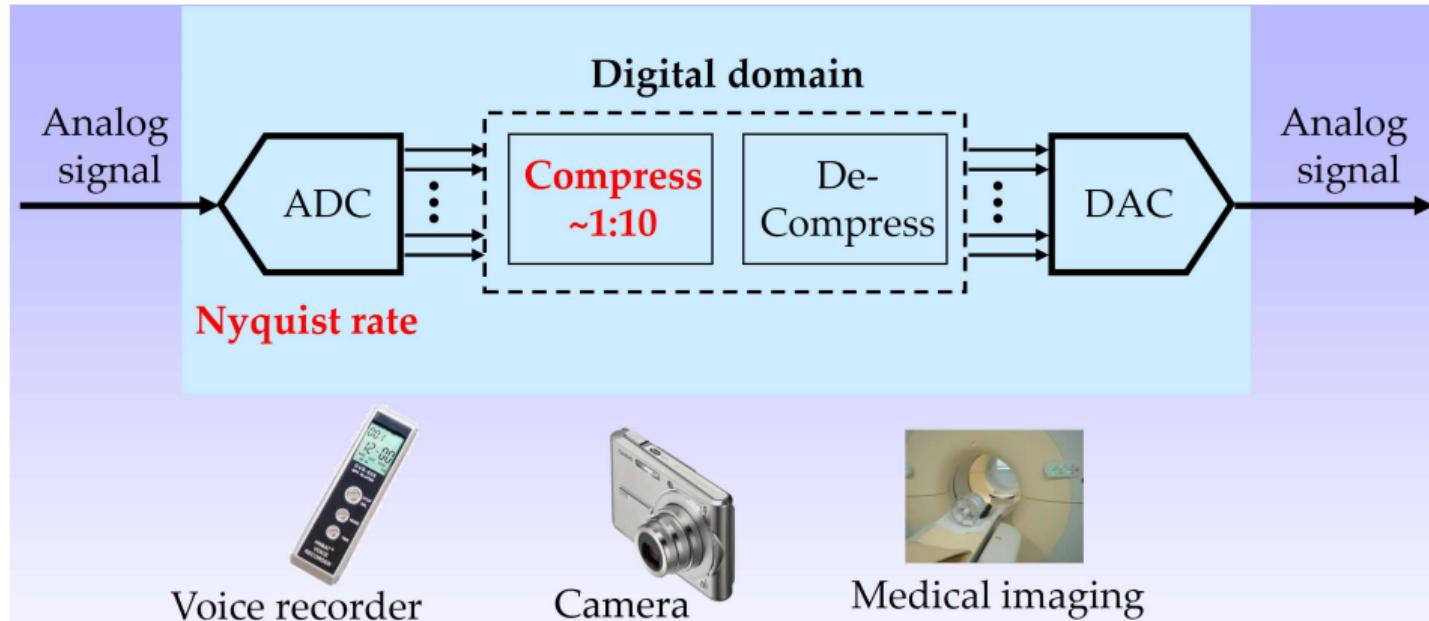
Outline

1 Parametric Signal Models

2 Compressed Sensing

3 SBL

Compressed Sensing



Compressed Sensing

In some applications, measurements are costly:

- Magnetic resonance imaging:
 - scan time \approx 30 minutes
 - scan time proportional to # samples taken

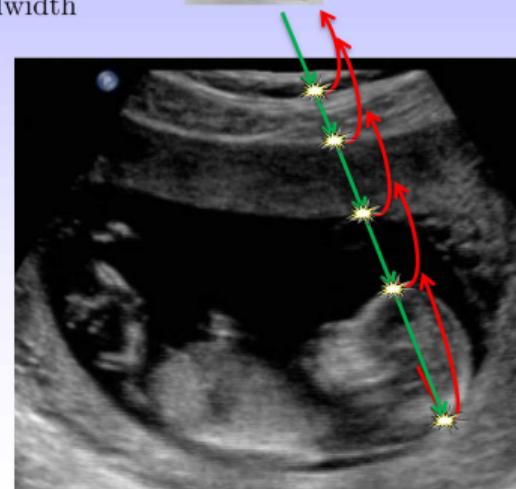
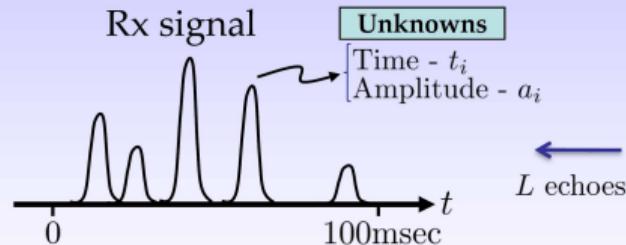


- Imaging outside visible spectrum:
 - CMOS doesn't work
 - high cost per pixel
- Wireless communication:
 - pilots inserted to measure channel
 - more pilots means less payload



Ultra-Sound Imaging

- High sampling rates
- High digital processing rates

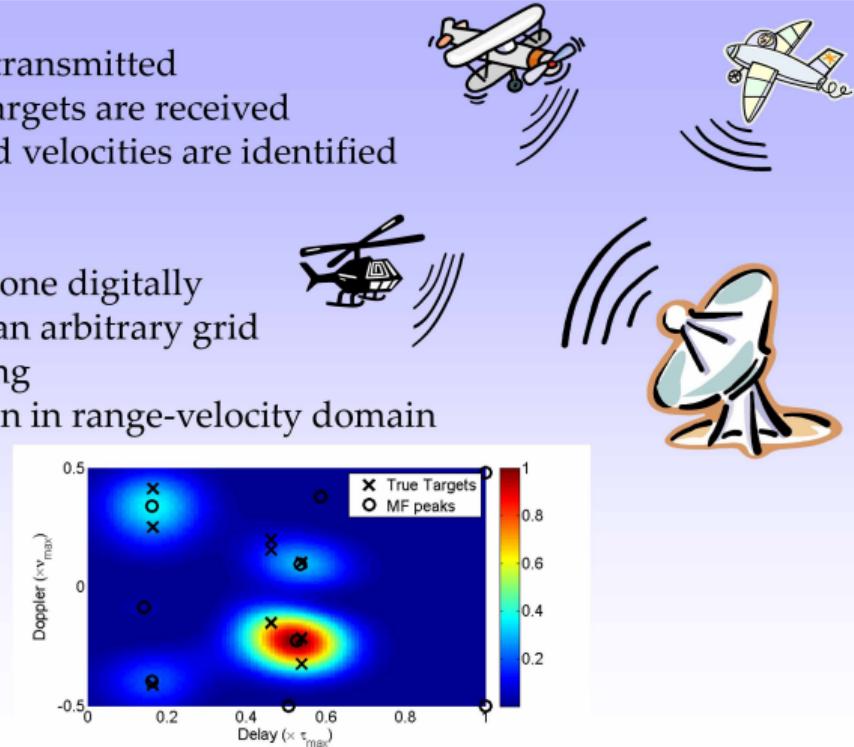


- Echoes result from scattering in the tissue
- The image is formed by identifying the scatterers

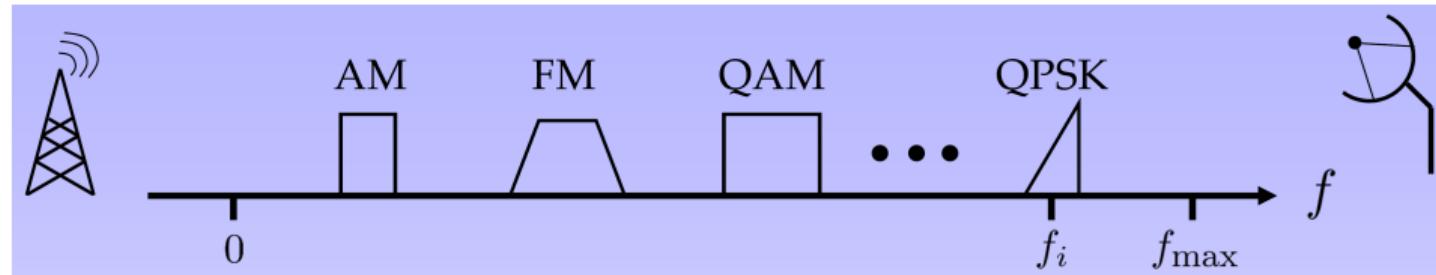
Radar

- Principle:
 - A known pulse is transmitted
 - Reflections from targets are received
 - Target's ranges and velocities are identified

 - Challenge:
 - All processing is done digitally
 - Targets can lie on an arbitrary grid
 - Process of digitizing
→ loss of resolution in range-velocity domain

 - Subspace methods:
- 

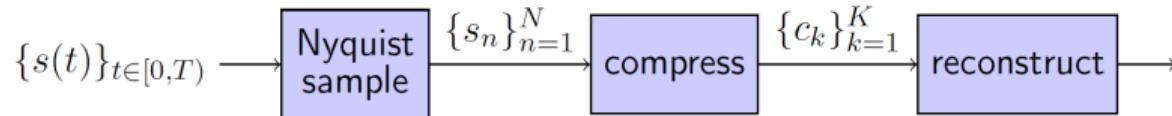
Cognitive Radio



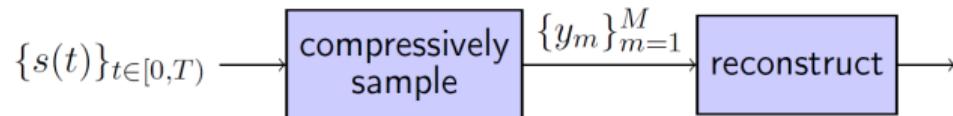
- The spectrum occupation by legacy/primary users is sparse.
- Unlicensed secondary users can insert in spectral holes.
- However, have to find the spectral holes, or the occupied spectrum portions.
- Generalized Nyquist says that one can sample at a rate exceeding the sum of the bandwidth, however here also (only) the spectral support needs to be estimated.

Compressive vs Classical Sampling

- Classical approach:



- New approach:



$$\text{Nyquist rate } \frac{N}{T} \quad \gg \quad \text{compressive sampling rate } \frac{M}{T} \quad \gtrsim \quad \text{information rate } \frac{K}{T}$$

Linear Measurements Model

- 1 For now, assume noiseless linear measurements, e.g.,

$$y_m = \int_0^T \phi_m(t) s(t) dt, \quad m = 1, \dots, M$$

- 2 Also assume signal $s(t)$ is bandlimited, in which case Nyquist says

$$s(t) = \sum_{n=1}^N s_n \operatorname{sinc}\left(\frac{t}{T_s} - n + 1\right), \quad t \in [0, T).$$

Putting these together, we get the convenient discrete representation

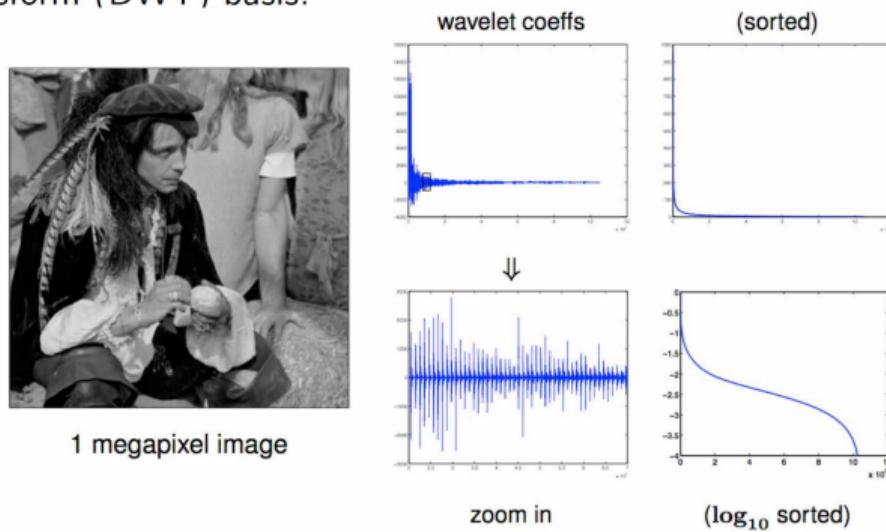
$$y_m = \sum_{n=1}^N s_n \underbrace{\int_0^T \phi_m(t) \operatorname{sinc}\left(\frac{t}{T_s} - n + 1\right) dt}_{\triangleq \Phi_{m,n}}$$

or, in matrix/vector form, $\boxed{\mathbf{y} = \Phi s}$ for $s \in \mathbb{R}^N$ and $\mathbf{y} \in \mathbb{R}^M$.



Sparsity

- Many real-world signals are approximately sparse in a known basis.
- For example, natural images are sparse in the discrete wavelet transform (DWT) basis:



Typically: 99% signal energy captured by only 2.5% of DWT coefficients!

Sparsity will be captured by prior information in a Bayesian approach.

K -sparse in a Dictionary Ψ

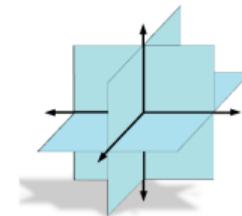
- We say that a signal class \mathcal{S} is K -sparse in the dictionary Ψ if each $s \in \mathcal{S}$ can be written as

$$s = \Psi x$$

for some K -sparse vector x (i.e., x has at most K nonzero elements).

- Usually orthonormal dictionaries Ψ are used (e.g., DWT, DCT, DFT), but overcomplete dictionaries may also be considered.

- Geometrically, a K -sparse vector $x \in \mathbb{R}^N$ lives in a union of $\binom{N}{K}$ subspaces, each of dimension K :



Combining Sparsity with Compressed Measurements

Recall...

- Linear measurement model: $y = \Phi s$ for $\Phi \in \mathbb{R}^{M \times N}$
- Sparse signal model: $s = \Psi x$ for K -sparse $x \in \mathbb{R}^N$

Together...

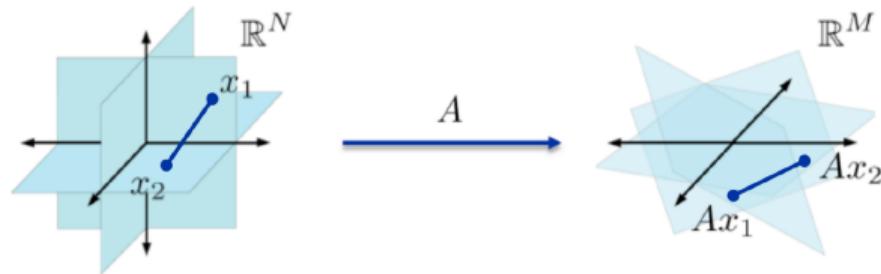
- Compressive sensing model: $y = \underbrace{\Phi \Psi}_{\triangleq A} x$ for $A \in \mathbb{R}^{M \times N}$

Questions:

- 1 What properties of A ensure the recovery of x ?
- 2 Given dictionary Ψ , how can we design Φ to ensure a good A ?

Restricted Isometry Property

- Recall model: $y = Ax$ for $A \in \mathbb{R}^{M \times N}$ and K -sparse $x \in \mathbb{R}^N$.
- Note: if signals $x_1 \neq x_2$ map to the same y , they can't be recovered!



- In general, for our measurement system to be **information preserving**, we want that $\|x_1 - x_2\|_2 \approx \|Ax_1 - Ax_2\|_2$ for all K -sparse x_1, x_2 , or

$$1 - \delta \leq \frac{\|Ad\|_2^2}{\|d\|_2^2} \leq 1 + \delta \text{ for all } 2K\text{-sparse } d. \quad \text{"RIP"}$$

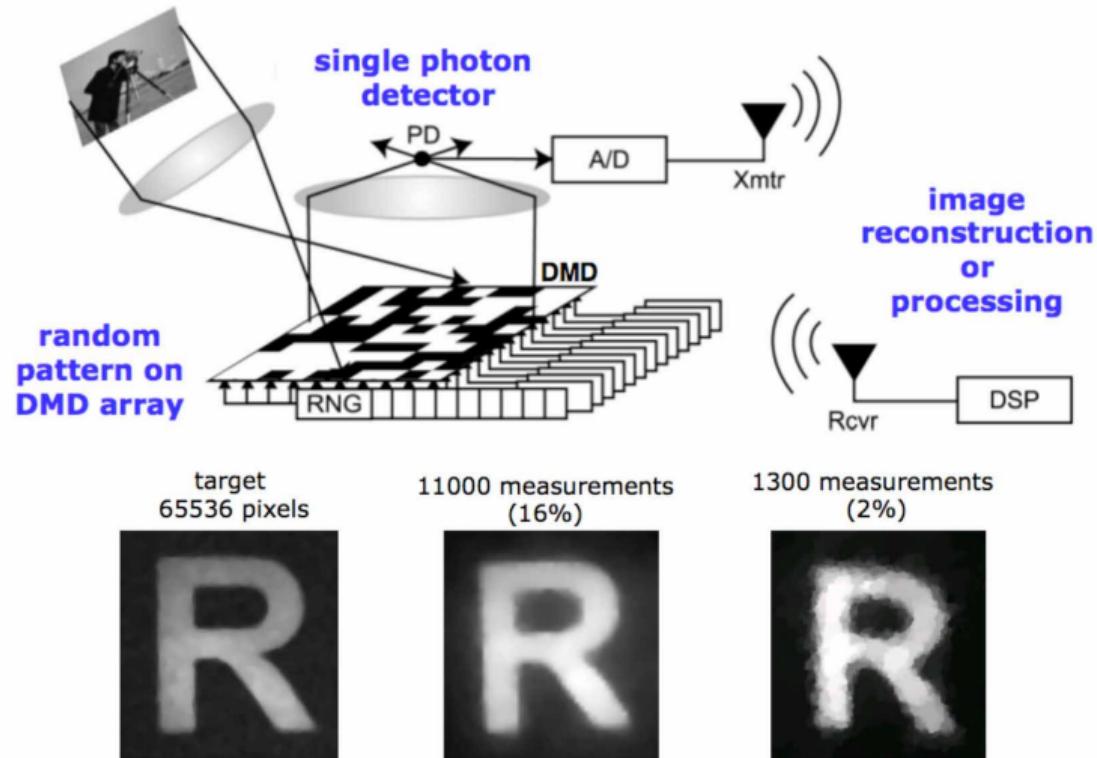
RIP from Randomness

- Testing a given matrix for RIP is an NP-hard (combinatorial) problem.
- Fortunately, if A is **randomly** drawn with **independent zero-mean sub-Gaussian** entries (e.g., normal, ± 1), then *with high probability* it will satisfy RIP if

$$M \geq O\left(K \log \frac{N}{K}\right).$$

- Similarly, if Φ is constructed randomly in the same way, then $A = \Phi\Psi$ will satisfy RIP for any orthonormal Ψ .
- In practice, **semi-random** Φ are preferable, e.g.,
Create $\Phi = JFD$, where D is a diagonal matrix with random ± 1 s,
 F is the N -FFT matrix, and J randomly selects M outputs.

Single Pixel Camera (Rice U.)



Best Sparse Fit - the ℓ_0 technique

Find the sparsest x that explains y up to a specified tolerance of ϵ :

$$\hat{x} = \arg \min_{x} \underbrace{\|x\|_0}_{\# \text{ nonzero coefs}} \text{ s.t. } \|y - Ax\|_2 \leq \epsilon.$$

Unfortunately, this is **NP-hard**; we'd need to check all $\binom{N}{K} \approx N^K$ possible supports!

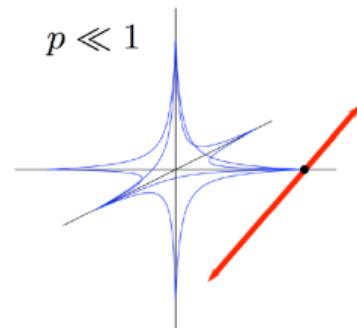
Let's think about this problem geometrically...

Geometry of constrained ℓ_p minimization

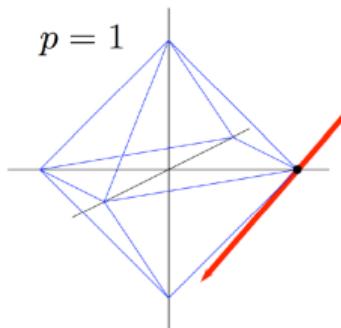
Now consider, for some fixed $p > 0$, the optimization problem:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \underbrace{\|\mathbf{x}\|_p}_{\sqrt[p]{\sum_n |x_n|^p}} \quad \text{s.t. } \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq \epsilon.$$

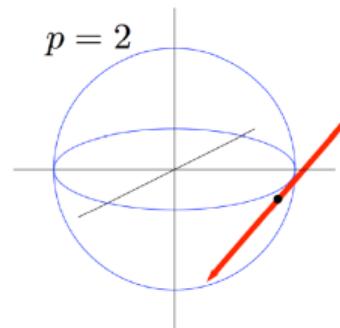
The solution can be found by growing the ℓ_p -ball until it touches the ϵ -rod:



Solution definitely sparse
but problem is **NP hard**.



Solution usually sparse
and problem is **convex**!



Solution is **not sparse**;
 \Leftrightarrow LS when $\epsilon = 0$.

This suggests to use the ℓ_1 norm as a surrogate for the ℓ_0 norm!

LASSO

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ s.t. } \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq \epsilon$$

- Convex! Can be solved very efficiently.
- For \mathbf{A} satisfying $2K$ -RIP, LASSO guarantees that

$$\|\hat{\mathbf{x}} - \mathbf{x}\|_2 \leq \frac{C_1}{\sqrt{K}} \|\mathbf{x} - \mathbf{x}_K\|_1 + C_2 \|\mathbf{w}\|_2$$

where \mathbf{x}_K is the best K -sparse approximation of \mathbf{x} and C_1, C_2 are constants that depend on the RIP δ . Wow!

- In the special case when \mathbf{x} is K -sparse, this simplifies to

$$\|\hat{\mathbf{x}} - \mathbf{x}\|_2 \leq C_2 \|\mathbf{w}\|_2.$$

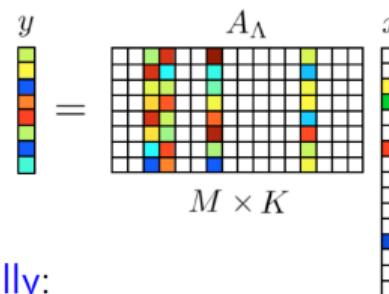
Greedy Search

Main ideas:

- If we can correctly recover the **support** Λ of x (i.e., the locations of nonzeros), then determining the non-zero amplitudes is easy, e.g.,

$$\mathbf{x}_\Lambda = (\mathbf{A}_\Lambda^H \mathbf{A}_\Lambda)^{-1} \mathbf{A}_\Lambda^H \mathbf{y}$$

(least squares)



- Estimate the support **sequentially**:
 - Find the column of \mathbf{A} most “similar” to \mathbf{y} and store its index.
 - Subtract the effect of this column from \mathbf{y} .
 - Repeat (until residual is sufficiently small)!

Famous algorithms include MP, OMP, IHT, CoSaMP, Subspace Pursuit

Bayesian Methods

In the Bayesian approach, one . . .

- models the signal using a **prior** pdf $p(\mathbf{x})$,
- models the measurement process using a **likelihood** function $p(\mathbf{y}|\mathbf{x})$,
- performs inference via **Bayes rule**, yielding the **posterior** pdf

$$p(\mathbf{x}|\mathbf{y}) = Z^{-1} p(\mathbf{y}|\mathbf{x}) p(\mathbf{x}) \text{ where } Z \text{ is a scaling constant,}$$

- often summarizes the posterior pdf by a **point estimate** like

$$\hat{\mathbf{x}} = \int \mathbf{x} p(\mathbf{x}|\mathbf{y}) d\mathbf{x} \quad \text{MMSE estimate}$$

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) \quad \text{MAP estimate}$$

and possibly other statistics that quantify **estimate uncertainty**.

LASSO as Bayesian Method

If we assume ...

- additive white Gaussian noise of variance σ^2
- i.i.d Laplacian signal with rate λ/σ^2

then

- likelihood: $p(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{M/2}} \exp(-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{Ax}\|_2^2)$
- prior: $p(\mathbf{x}) = \frac{1}{(2\sigma^2/\lambda)^M} \exp(-\frac{\lambda}{\sigma^2} \|\mathbf{x}\|_1)$

for which the maximum a posteriori (MAP) estimate is

$$\begin{aligned}\hat{\mathbf{x}} &= \arg \max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) = \arg \max_{\mathbf{x}} \log (Z^{-1} p(\mathbf{y}|\mathbf{x}) p(\mathbf{x})) \\ &= \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{Ax}\|_2^2 + \lambda \|\mathbf{x}\|_1\end{aligned}$$

which is an unconstrained version of the LASSO problem.

Relevance Vector Machine - Sparse Bayesian Learning

- The RVM is based on the *conditionally Gaussian* priors

$$p(\mathbf{x}|\boldsymbol{\alpha}) = \prod_{n=1}^N \mathcal{N}(x_n; 0, \alpha_n^{-1}) \quad \text{and} \quad p(\boldsymbol{\alpha}) = \prod_{n=1}^N \Gamma(\alpha_n; 0, 0)$$

$$p(\mathbf{w}|\boldsymbol{\beta}) \sim \prod_{m=1}^M \mathcal{N}(w_m; 0, \beta^{-1}) \quad \text{and} \quad \beta \sim \Gamma(0, 0)$$

Note that, as “precision” $\alpha_n \rightarrow \infty$, the coefficient x_n is zeroed.

- The *conditional* posterior is (due to Gaussianity) simply

$$p(\mathbf{x}|\mathbf{y}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad \text{for} \quad \begin{cases} \boldsymbol{\mu} = \beta \boldsymbol{\Sigma} \mathbf{A}^T \mathbf{y} \\ \boldsymbol{\Sigma} = (\beta \mathbf{A}^T \mathbf{A} + \mathcal{D}(\boldsymbol{\alpha}))^{-1}. \end{cases}$$

- In practice, $\{\boldsymbol{\alpha}, \beta\}$ are estimated using the EM algorithm and then plugged into $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ to approximate the posterior $p(\mathbf{x}|\mathbf{y})$.
- The RVM (also known as “SBL” and “BCS”) is relatively slow.

Other Bayesian Methods

- Bayesian matching pursuits:
 - Greedy methods that use probabilistic support selection.
- Approximate message passing (AMP):
 - Inspired by methods from statistical physics and information theory.
 - Near-optimal in terms of speed and accuracy if \mathbf{A} is large & random.

Alternating Minimization

- also called "cyclic minimization" or "block coordinate descent"
- Cost function to be minimized: $f(\theta)$
- Partition $\theta = \theta_{1:m} = \{\theta_1, \dots, \theta_m\} = \{\theta_{1:k-1}, \theta_k, \theta_{k+1:m}\}$.
- Sweep through the partition. In sweep i :

$$\theta_k^{(i)} = \arg \min_{\theta_k} f(\theta_{1:k-1}^{(i)}, \theta_k, \theta_{k+1:m}^{(i-1)}) \quad k = 1, \dots, m$$

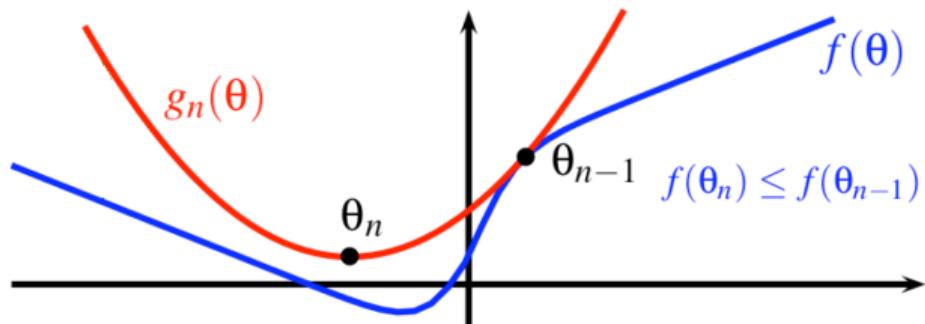
Partitioning done in such a way that these reduced optimization problems are "easy".

- Guaranteed to converge to a local minimum:

$$f(\theta_{1:k}^{(i)}, \theta_{k+1:m}^{(i-1)}) \leq f(\theta_{1:k-1}^{(i)}, \theta_{k:m}^{(i-1)})$$

- Not necessary to do regular sweeps through the partition, can minimize in any order.

Majorization Minimization



- Cost function to be minimized: $f(\theta)$. At iteration $n-1$: have $\theta^{(n-1)}$.
- A **majorizer** at iteration $n-1$ is a function $g_n(\theta)$ such that

$$g_n(\theta^{(n-1)}) = f(\theta^{(n-1)})$$

$$g_n(\theta) \geq f(\theta), \forall \theta$$

- where $g_n(\theta)$ is possibly convex or in any case "easy" to minimize or **just decrease**.
- Then

$$\theta^{(n)} = \arg \min_{\theta} g_n(\theta)$$

$$f(\theta^{(n)}) \leq g_n(\theta^{(n)}) \leq g_n(\theta^{(n-1)}) = f(\theta^{(n-1)})$$

Hence guaranteed to converge to a local minimum.



Expectation Maximization (EM) Algorithm

- Maximum Likelihood estimation: $f(\theta) = -\ln p(\mathbf{y}|\theta)$
- Often resulting from eliminating random \mathbf{x} in $p(\mathbf{y}, \mathbf{x}|\theta)$. θ : deterministic (actual) parameters, \mathbf{x} : random, "unobserved" data, $\{\mathbf{x}, \mathbf{y}\}$: "complete" data.
Estimating θ from $\{\mathbf{x}, \mathbf{y}\}$ easier than from \mathbf{y} alone.
- Majorizer: $g_i(\theta) = f(\theta^{(i-1)}) - E_{\mathbf{x}|\mathbf{y}, \theta^{(i-1)}} \left\{ \ln \left(\frac{p(\mathbf{y}, \mathbf{x}|\theta)}{p(\mathbf{y}, \mathbf{x}|\theta^{(i-1)})} \right) \right\}$, $g_i(\theta^{(i-1)}) = f(\theta^{(i-1)})$

$$\begin{aligned}
 g_i(\theta) &\geq f(\theta^{(i-1)}) - \ln \left(E_{\mathbf{x}|\mathbf{y}, \theta^{(i-1)}} \left\{ \frac{p(\mathbf{y}, \mathbf{x}|\theta)}{p(\mathbf{y}, \mathbf{x}|\theta^{(i-1)})} \right\} \right) \quad \text{Jensen's inequality} \\
 &= f(\theta^{(i-1)}) - \ln \left(E_{\mathbf{x}|\mathbf{y}, \theta^{(i-1)}} \left\{ \frac{p(\mathbf{y}, \mathbf{x}|\theta)}{p(\mathbf{x}|\mathbf{y}, \theta^{(i-1)}) p(\mathbf{y}|\theta^{(i-1)})} \right\} \right) \\
 &= f(\theta^{(i-1)}) - \ln \left(\int p(\mathbf{x}|\mathbf{y}, \theta^{(i-1)}) \frac{p(\mathbf{y}, \mathbf{x}|\theta)}{p(\mathbf{x}|\mathbf{y}, \theta^{(i-1)}) p(\mathbf{y}|\theta^{(i-1)})} d\mathbf{x} \right) \\
 &= f(\theta^{(i-1)}) - \ln \left(\frac{1}{p(\mathbf{y}|\theta^{(i-1)})} \underbrace{\int p(\mathbf{y}, \mathbf{x}|\theta) d\mathbf{x}}_{p(\mathbf{y}|\theta)} \right) = f(\theta^{(i-1)}) - \ln \left(\frac{p(\mathbf{y}|\theta)}{p(\mathbf{y}|\theta^{(i-1)})} \right) = -\ln(p(\mathbf{y}|\theta)) = f(\theta)
 \end{aligned}$$

- EM algorithm: at iteration i :

Expectation step: $\bar{g}_i(\theta) = E_{\mathbf{x}|\mathbf{y}, \theta^{(i-1)}} \{ \ln p(\mathbf{y}, \mathbf{x}|\theta) \}$
 Maximization step: $\theta^{(i)} = \arg \max_{\theta} \bar{g}_i(\theta)$

where $g_i(\theta) = \text{constant} - \bar{g}_i(\theta)$. EM converges to (local) Maximum Likelihood estimate.

Time Varying Sparse State Tracking

Sparse signal \mathbf{x}_t is modeled using an AR(1) process with diagonal correlation coefficient matrix \mathbf{F} .

$$\begin{array}{c} \mathbf{y}_t \\ \vdots \\ N \times 1 \end{array} = \mathbf{A}_t \quad \begin{matrix} \mathbf{A}_t \\ \text{N} \times M, N \ll M \end{matrix} \quad \begin{array}{c} \mathbf{x}_t \\ \vdots \\ M \times 1 \end{array} + \mathbf{v}_t \quad \begin{array}{c} \mathbf{v}_t \\ \vdots \\ N \times 1 \end{array}$$

$$\begin{array}{c} \mathbf{x}_t \\ \vdots \\ M \times 1 \end{array} = \mathbf{F} \quad \begin{matrix} \mathbf{F} \\ M \times M \end{matrix} \quad \begin{array}{c} \mathbf{x}_{t-1} \\ \vdots \\ M \times 1 \end{array} + \mathbf{w}_t \quad \begin{array}{c} \mathbf{w}_t \\ \vdots \\ M \times 1 \end{array}$$

Define: $\Xi = \text{diag}(\xi)$, $\mathbf{F} = \text{diag}(\mathbf{f})$.

f_i : correlation coefficient and $x_{i,t} \sim \mathcal{CN}(x_{i,t}; 0, \frac{1}{\xi_i})$. Further, $\mathbf{w}_t \sim \mathcal{CN}(\mathbf{w}_t; \mathbf{0}, \boldsymbol{\Gamma}^{-1} = \Xi^{-1}(\mathbf{I} - \mathbf{FF}^H))$

and $\mathbf{v}_t \sim \mathcal{CN}(\mathbf{v}_t; \mathbf{0}, \gamma^{-1}\mathbf{I})$. **VB leads to Gaussian SAVE-Kalman Filtering (GS-KF).**

Applications: Localization, Adaptive Filtering.

Compressed Sensing Problem

- **Noiseless case:** Given underdetermined \mathbf{y} , \mathbf{A} , the optimization problem is

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ subject to } \mathbf{y} = \mathbf{Ax}.$$

Can recover \mathbf{x} and its support for small $N - \|\mathbf{x}\|_0$
(small overdetermination if support were known)

- **Noisy case:**

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ subject to } \|\mathbf{y} - \mathbf{Ax}\|_2 \leq \epsilon.$$

- ℓ_0 norm minimization: an NP-hard problem.
- Constrained problem \Rightarrow **Lagrangian, Convex Relaxation** (using p norm, $p > 1$):

$$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{Ax}\|_2^2 + \lambda \|\mathbf{x}\|_p.$$

Restricted Isometry Property (RIP): $\mathbf{A}^T \mathbf{A}$ sufficiently diagonally dominant

- Most identifiability work considered **noiseless data & exact sparsity**

Sparse Signal Recovery Algorithms

Convex Relaxation based Methods:

- Basis pursuit (ℓ_1 norm)¹.
- LASSO(ℓ_1 norm)²
- Dantzig selector³
- FOCUSS (ℓ_p norm, with $p < 1$).

Greedy Algorithms:

- Matching Pursuit⁴
- Orthogonal Matching Pursuit (OMP)⁵
- CoSaMP⁶

Iterative Methods:

- Iterative Shrinkage and Thresholding Algorithm (ISTA)⁷ or Fast ISTA.
- Approximate Message Passing variants (xAMP)- more details to follow.
- Very recent: Deep learning based methods such as Learned ISTA (LISTA)⁸, Learned AMP (LAMP) and Learned Vector AMP (LVAMP)⁹.

¹Chen, Donoho, Saunders'99, ²Tibshirani'96, ³Candes, Tao'07

⁴Mallat, Zhang'93, ⁵Tropp, Gilbert'07, ⁶Needell, Tropp'09

⁷Daubechies, Defrise, Mol'04, ⁸Gregor, Cun'10, ⁹Borgerding, Schniter, Rangan'17

James-Stein Estimator

- Bayesian interpretation of (possibly overdetermined) Compressed Sensing:

$$\min_x \|y - Ax\|_2^2 - 2\sigma_v^2 \ln p(x)$$

- Stein and James¹⁰ showed that for i.i.d. linear Gaussian model $p(x) = \mathcal{N}(x; \mathbf{0}, \xi^{-1} \mathbf{I})$, it is possible to construct a nonlinear estimate of x with lower MSE than that of ML for all values of the true unknown x .
- A popular design strategy: is to minimize Stein's unbiased risk estimate (SURE), which is an unbiased estimate of the MSE.
- SURE directly approximates the MSE of an estimate from the data, without requiring knowledge of the hyperparameters (ξ), it is an instance of empirical Bayes.
- Stein's landmark discovery lead to the study of biased estimators that outperform minimum variance unbiased estimators (MVU) in terms of MSE, e.g. work by Yonina Eldar¹¹.
- Shrinkage estimators and penalized maximum likelihood (PML) estimators.

¹⁰James, Stein'61

¹¹Eldar'08

Kernel Methods in Automatic Control

- Kernel methods in linear system identification¹² ($\mathbf{y} = \mathbf{Ax} + \mathbf{v}$, $\mathbf{v} \sim \mathcal{N}(\mathbf{v}; \mathbf{0}, \gamma^{-1}\mathbf{I})$).
- Traditional methods: maximum likelihood (ML) or prediction error methods (PEM)
- ML/PEM optimal in the large data limit.
- Questions: Model structure design for ML/PEM. Achieving a good bias-variance trade off.
- Solution: Parameterized Kernel design and hyperparameter estimation. Methods for hyperparameter estimation include cross-validation (CV), empirical Bayes (EB), C_p statistics and Stein's unbiased risk estimate (SURE).
- Regularized least square estimator (\mathbf{P} is symmetric and +ve semidefinite kernel matrix):

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathcal{R}^M} \|\mathbf{y} - \mathbf{Ax}\|^2 + \frac{1}{\gamma} \mathbf{x}^T \mathbf{P}^{-1} \mathbf{x}.$$

- Parameterized family of matrices, $\mathbf{P}(\boldsymbol{\eta})$, where $\boldsymbol{\eta} \in \mathcal{R}^P$. $\boldsymbol{\eta}$ are the hyperparameters.
- Can be interpreted as a constrained form of SBL, with a zero-mean Gaussian prior for \mathbf{x} of which the covariance matrix is a linear combination of some fixed matrices (SBL being a special case with fixed matrices $\mathbf{e}_i \mathbf{e}_i^T$).
- A good overview of Kernel methods, connection with machine learning¹³.

¹²Pillonetto, Nicolao'10

¹³Pillonetto, Dinuzzo, Chen, Nicolao, Ljung'14

Kernel Hyperparameter Estimation

- Empirical Bayes (EB=Type II ML):

$$\begin{aligned}\widehat{\boldsymbol{\eta}}_{EB} &= \arg \min_{\boldsymbol{\eta}} f_{EB}(\boldsymbol{P}(\boldsymbol{\eta})), \\ f_{EB}(\boldsymbol{P}(\boldsymbol{\eta})) &= \mathbf{y}^T \mathbf{Q}^{-1} \mathbf{y} + \ln \det(\mathbf{Q}) \text{ with } \mathbf{Q} = \mathbf{A} \boldsymbol{P} \mathbf{A}^T + \frac{1}{\gamma} \mathbf{I}_N.\end{aligned}$$

- Stein's Unbiased Risk Estimator (SURE) method:

- SURE: MSE of signal reconstruction ($MSE_x(\boldsymbol{P}) = E(\|\hat{\mathbf{x}} - \mathbf{x}\|^2)$):

$$SURE_x : \widehat{\boldsymbol{\eta}}_{Sx} = \arg \min_{\boldsymbol{\eta}} f_{Sx}(\boldsymbol{P}(\boldsymbol{\eta})), \text{ with}$$

$$\begin{aligned}f_{Sx}(\boldsymbol{P}(\boldsymbol{\eta})) &= \frac{1}{\gamma^2} \mathbf{y}^T \mathbf{Q}^{-T} \mathbf{A} (\mathbf{A}^T \mathbf{A})^{-2} \mathbf{A}^T \mathbf{Q}^{-1} \mathbf{y} + \frac{1}{\gamma} \text{tr}\{2\mathbf{R}^{-1} - (\mathbf{A}^T \mathbf{A})^{-1}\}, \\ \mathbf{R} &= \mathbf{A}^T \mathbf{A} + \frac{1}{\gamma} \boldsymbol{P}^{-1}.\end{aligned}$$

- The SURE estimator converge to the best possible hyperparameter in terms of MSE in the asymptotic limit, "asymptotically consistent".
- EB estimator converges to another best hyperparameter minimizing the expectation of the EB estimation criterion (EEB).
- Convergence of EB is faster than that of the SURE estimator.

¹³Mu, Chen, Ljung'18

Outline

1 Parametric Signal Models

2 Compressed Sensing

3 SBL

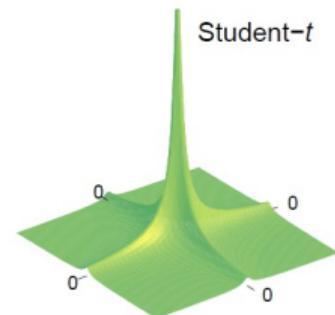
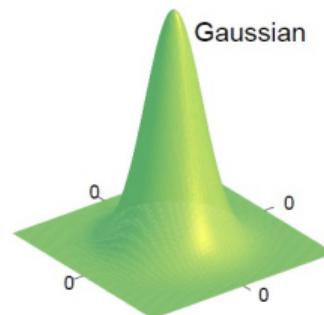
Sparse Bayesian Learning - SBL

- Bayesian Compressed Sensing: 2-layer hierarchical prior for \mathbf{x} as in ¹⁴, inducing sparsity for \mathbf{x} (conjugate priors: posterior pdf of same family as prior pdf) :

$$p_{\mathbf{x}}(x_{i,t}|\xi_i) = \mathcal{N}(x_{i,t}; 0, \xi_i^{-1}), \quad p(\xi_i|a, b) = \Gamma^{-1}(a)b^a \xi_i^{a-1} e^{-b\xi_i}$$

⇒ sparsifying Student-t marginal

$$p_{\mathbf{x}}(x_{i,t}) = \frac{b^a \Gamma(a + \frac{1}{2})}{(2\pi)^{\frac{1}{2}} \Gamma(a)} (b + x_{i,t}^2/2)^{-(a + \frac{1}{2})}$$



- Sparsification of the Innovation Sequence: we apply the (Gamma) prior not to the precision of the state \mathbf{x} but of its innovation \mathbf{w} , allowing to sparsify at the same time the components of \mathbf{x} AND their variation in time (innovation).
- The inverse of the noise variance γ is also assumed to have a Gamma prior,
 $p_{\gamma}(\gamma|c, d) = \Gamma^{-1}(c)d^c \gamma^{c-1} e^{-d\gamma}$.

¹⁴Tipping'01

Original SBL Algorithm (Type II ML)

- Original SBL¹⁵, for a fixed estimate of the **hyperparameters** $(\hat{\xi}, \hat{\gamma})$, the posterior of x is Gaussian, i.e.

$$p_x(x|y, \hat{\xi}, \hat{\gamma}) = \mathcal{N}(x; \hat{x}, \Sigma_L)$$

leading to the **(Linear) MMSE estimate for x**

$$\begin{aligned} \hat{x} &= \hat{\gamma}(\hat{\gamma}\mathbf{A}^T\mathbf{A} + \hat{\Xi})^{-1}\mathbf{A}^Ty, \\ \Sigma_L &= (\hat{\gamma}\mathbf{A}^T\mathbf{A} + \hat{\Xi})^{-1}. \end{aligned} \quad (1)$$

- The **hyperparameters** are estimated from the likelihood function by marginalizing over the sparse coefficients x , the marginalized likelihood being denoted as $p_y(y|\xi, \gamma)$. ξ, γ are estimated by maximizing $p_y(y|\xi, \gamma)$ and this procedure is called as Type-II ML. **Type-II ML is solved using EM**, which leads to the following updates for the hyperparameters.

$$\hat{\xi}_i = \frac{a + \frac{1}{2}}{\left(\frac{\langle x_i^2 \rangle}{2} + b \right)}, \text{ where } \langle x_i^2 \rangle = \hat{x}_i^2 + \sigma_i^2. \quad \langle \gamma \rangle = \frac{c + \frac{N}{2}}{\left(\frac{\langle \|y - Ax\|^2 \rangle}{2} + d \right)},$$

$$\text{where, } \langle \|y - Ax\|^2 \rangle = \|y\|^2 - 2y^T\mathbf{A}\hat{x} + \text{tr}(\mathbf{A}^T\mathbf{A}(\hat{x}\hat{x}^T + \Sigma)), \\ \Sigma = \text{diag}(\Sigma_L) = \text{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_M^2), \quad \hat{x} = [\hat{x}_1, \hat{x}_2, \dots, \hat{x}_M]^T.$$

¹⁵Tipping'01, Wipf,Rao'04

Type I vs Type II ML

- Type I \Rightarrow standard MAP estimation (involves integrating out the hyperparameters)

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x}} [\log p_{\mathbf{y}}(\mathbf{y}|\mathbf{x}) + p_{\mathbf{x}}(\mathbf{x})].$$

- Type II \Rightarrow hyperparameters ($\Psi = \{\xi, \gamma\}$) are estimated using an evidence maximization approach

$$\hat{\Psi} = \arg \max_{\Psi} p_{\Psi}(\Psi|\mathbf{y}) = \arg \max_{\Psi} p_{\Psi}(\Psi) \int p_{\mathbf{y}}(\mathbf{y}|\Psi) = \arg \max_{\Psi} p_{\Psi}(\Psi) \int p_{\mathbf{y}}(\mathbf{y}|\mathbf{x}, \gamma) p_{\mathbf{x}}(\mathbf{x}|\xi) d\mathbf{x}.$$

- Why Type II is better than Type I? ¹⁶ In the evidence maximization framework instead of looking for the mode of the true posterior $p_{\mathbf{x}}(\mathbf{x}|\mathbf{y})$, the true posterior is approximated as $p_{\mathbf{x}}(\mathbf{x}|\mathbf{y}; \hat{\Psi})$, where $\hat{\Psi}$ is obtained by maximizing the true posterior mass over the subspaces spanned by the non zero indexes.
- Type I methods seek the mode of the true posterior and use that as the point estimate of the desired coefficients. Hence, if the true posterior distribution has a skewed peak, then the Type I estimate (Mode) is not a good representative of the whole posterior.

¹⁶Giri, Rao'16

Variational Bayesian (VB) Inference

- The computation of the posterior distribution of the parameters is usually intractable. As in SAGE, **SAVE is simply VB with partitioning of the unknowns at the scalar level**. Define $\theta = \{x, \xi, \gamma\}$, θ_i represents each scalar and $\theta_{\bar{i}}$ denotes θ excluding θ_i .

$$q(\theta) = q_\gamma(\gamma) \prod_{i=1}^M q_{x_i}(x_i) \prod_{i=1}^M q_{\xi_i}(\xi_i).$$

- VB compute the factors q by **minimizing the Kullback-Leibler distance** between the true posterior distribution $p(\theta|y)$ and the $q(\theta)$. From ¹⁷,

$$KLD_{VB} = D_{KL}(q(\theta)||p(\theta|y)) = \int q(\theta) \ln \frac{q(\theta)}{p(\theta|y)} d\theta.$$

- The KL divergence minimization is equivalent to **maximizing the evidence lower bound (ELBO)**¹⁸.

$$\ln p(y) = L(q) + KLD_{VB} = -D_{KL}(q(\theta)||p(\theta, y)) + D_{KL}(q(\theta)||p(\theta|y)), \text{ where,}$$

$\ln p(y)$ is the evidence, and $\min KLD_{VB}$ becomes equivalent to $\max L(q)$, the ELBO.

- We get for the element-wise VB recursions: **(Expectation Maximization (EM))** = special case:

$$\ln(q_i(\theta_i)) = \langle \ln p(y, \theta) \rangle_{\theta_{\bar{i}}} + c_i,$$

θ_s random, hidden
 θ_d deterministic)

¹⁸Beal'03, ¹⁹Tzikas, Likas, Galatsanos'08

Low Complexity-Space Alternating Variational Estimation (SAVE)

- Mean Field (MF) approximation: VB partitioned to scalar level (MF vs VB // SAGE vs EM), results in a SBL algorithm **without any matrix inversions**.
- The resulting **alternating optimization of the posteriors for each scalar in θ** leads to

$$\ln(q_i(\theta_i)) = \langle \ln p(\mathbf{y}, \theta) \rangle_{k \neq i} + c_i,$$

$$p(\mathbf{y}, \theta) = p_{\mathbf{y}}(\mathbf{y}|\mathbf{x}, \xi, \gamma)p_{\mathbf{x}}(\mathbf{x}|\xi)p_{\xi}(\xi)p_{\gamma}(\gamma).$$

where $\theta = \{\mathbf{x}, \xi, \gamma\}$ and θ_i represents each scalar in θ .

$$\begin{aligned} \ln p(\mathbf{y}, \theta) &= \frac{N}{2} \ln \gamma - \frac{\gamma}{2} \|\mathbf{y} - \mathbf{Ax}\|^2 + \sum_{i=1}^M \left(\frac{1}{2} \ln \xi_i - \frac{\xi_i}{2} x_i^2 \right) + \\ &\quad \sum_{i=1}^M ((a-1) \ln \xi_i + a \ln b - b \xi_i) + (c-1) \ln \gamma + c \ln d - d \gamma + \text{constants}. \end{aligned}$$

- Gaussian approximate posterior for x_i :**

$$\begin{aligned} \ln q_{x_i}(x_i) &= -\frac{\gamma}{2} \left\{ \langle \|\mathbf{y} - \mathbf{A}_{\bar{i}} \mathbf{x}_{\bar{i}}\|^2 \rangle - (\mathbf{y} - \mathbf{A}_{\bar{i}} \langle \mathbf{x}_{\bar{i}} \rangle)^T \mathbf{A}_i \mathbf{x}_i - \right. \\ &\quad \left. x_i \mathbf{A}_i^T (\mathbf{y} - \mathbf{A}_{\bar{i}} \langle \mathbf{x}_{\bar{i}} \rangle) + \|\mathbf{A}_i\|^2 x_i^2 \right\} - \frac{\xi_i}{2} x_i^2 + c_{x_i} = -\frac{1}{2\sigma_i^2} (x_i - \hat{x}_i)^2 + c'_{x_i}. \end{aligned}$$

SAVE Iterations Continued...

- The SAVE iterations for \mathbf{x} get obtained as

$$\sigma_i^2 = \frac{1}{\gamma \|\mathbf{A}_i\|^2 + \xi_i}, \quad \hat{\mathbf{x}}_i = \sigma_i^2 \mathbf{A}_i^T (\mathbf{y} - \mathbf{A}_i \hat{\mathbf{x}}_i) \gamma.$$

- Hyperparameter estimates: same as EM iterations. Gamma posterior for ξ_i and γ .
- No matrix inversions.
- Update of all the variables, $\mathbf{x}, \xi_i, \gamma$, requires simple addition and multiplication operations
- $\mathbf{y}^T \mathbf{A}$, $\mathbf{A}^T \mathbf{A}$ and $\|\mathbf{y}\|^2$ can be precomputed, so only need to be computed once.

- Remarks:** From LMMSE expression in (1), i^{th} row of $\gamma \mathbf{A}^T \mathbf{A} \hat{\mathbf{x}} + \Xi \hat{\mathbf{x}} = \gamma \mathbf{A}^T \mathbf{y}$:

$$\gamma \mathbf{A}_i^T \mathbf{A} \hat{\mathbf{x}} + \xi_i \hat{\mathbf{x}}_i = \gamma \mathbf{A}_i^T \mathbf{y} \text{ or } (\gamma \|\mathbf{A}_i\|^2 + \xi_i) \hat{\mathbf{x}}_i = \gamma \mathbf{A}_i^T (\mathbf{y} - \mathbf{A}_i \hat{\mathbf{x}}_i)$$

Hence **SAVE does linear PIC iterations to compute the LMMSE estimate.**

- However, for the posterior variances : $\sigma_i^2 = ((\Sigma_L^{-1})_{i,i})^{-1} \leq (\Sigma_L)_{ii}$, i with equality only for diagonal Σ_L

Convergence of SAVE

Theorem 1

The convergence condition for the sparse coefficients x_i of the SAVE algorithm²⁰ can be written as $\rho(\mathbf{D}^{-1}\mathbf{H}) < 1$, where $\mathbf{D} = \text{diag}(\hat{\gamma}\mathbf{A}^T\mathbf{A} + \hat{\Xi})$, $\mathbf{H} = \text{offdiag}(\hat{\gamma}\mathbf{A}^T\mathbf{A})$. $\rho(\cdot)$ denotes the spectral radius. Moreover, if SAVE converges, assuming the estimate of hyperparameters are consistent, the posterior mean (point estimate) always converges to the exact value (LMMSE). However, the predicted posterior variance is quite suboptimal.

Remark: To fix the convergence of SAVE (when $\rho(\mathbf{D}^{-1}\mathbf{H}) > 1$), we can use the diagonal loading method²¹. The modified iterations (with a diagonal loading factor matrix Λ) can be written as,

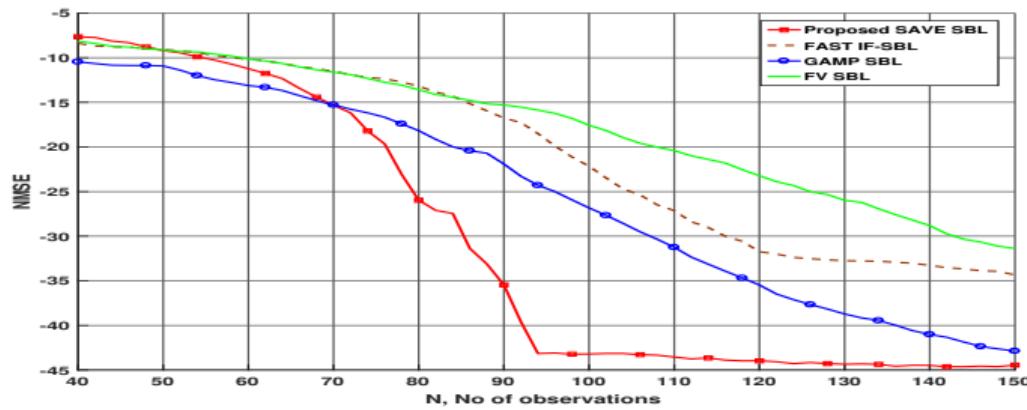
$$\begin{aligned} (\mathbf{D} + \tilde{\Xi})\mathbf{x}^{(t+1)} &= -(\mathbf{H} - \tilde{\Xi})\mathbf{x}^{(t)} + \hat{\gamma}\mathbf{A}^T\mathbf{y}, \implies \\ \mathbf{x}^{(t+1)} &= -(\mathbf{D} + \tilde{\Xi})^{-1}(\mathbf{H} - \tilde{\Xi})\mathbf{x}^{(t)} + (\mathbf{D} + \tilde{\Xi})^{-1}\hat{\gamma}\mathbf{A}^T\mathbf{y}. \end{aligned}$$

The convergence condition gets modified as $\rho((\mathbf{D} + \tilde{\Xi})^{-1}(\mathbf{H} - \tilde{\Xi})) < 1$. Another point worth noting here is that, if the power delay profile Ξ is also estimated using MF, $\hat{\gamma}\text{diag}(\mathbf{A}^T\mathbf{A}) + \hat{\Xi}$, where $\tilde{\Xi} = \Xi + \tilde{\Xi}$, with $\tilde{\Xi} > 0$. In this case, $\tilde{\Xi}$ may represent an automatic correction factor (diagonal loading) to force convergence of SAVE for cases where $\rho(\mathbf{D}^{-1}\mathbf{H}) > 1$.

²⁰Thomas,Slock'18

²¹Johnson, Bickson, Dolev'09

NMSE Results



NMSE vs the number of observations M ($N = 200, K = 40$).

- For sufficient amount of data, **SAVE has significantly lower MSE than the other fast algorithms.**
- Why? The resulting problem of alternating optimization of x and the hyperparameters ξ and γ appears to be characterized by many local optima. Apparently, **the component-wise VB approach appears to allow to avoid a lot of bad local optima, explaining the better performance, apart from lower complexity.**
- At very low amount of data, suboptimal approaches such as AMP which don't introduce individual hyper parameters per x component and assume that the x_i behave i.i.d. behave better because of the lower number of hyper parameters to be estimated.

An Overview of Fast SBL Algorithms

- Fast SBL using Type II ML by Tipping²²: greedy approach handling one x_i at a time, plus replacing precisions by their convergence values, leading to pruning of the small x_i components, i.e. explicit sparsity.
- Fast SBL using VB by Shutin et. al.²³: Shutin uses VB while Tipping is Type II ML as in original SBL. They do both replace precisions by their convergence values. Shutin also added some extra view points in terms of the pruning condition being interpreted as relating between sparsity properties of SBL and a measure of SNR. Main message of the both being faster convergence compared to original SBL, not much reduction in per iteration complexity.
- BP-SBL²⁴: In SBL, with fixed hyperparameters, MAP or MMSE estimate (follows from the Gaussian posterior) of x can be efficiently computed using belief propagation (BP), since all the messages involved are Gaussian (without any approx.).
- Inverse Free SBL (IF-SBL)²⁵: Optimization using a relaxed ELBO.
- Hyperparameter free sparse estimation²⁶: Does not require hyperparameter tuning compared to SBL. Uses covariance matching, equivalent to square root LASSO.

²¹Tipping, Faul'03, ²²Shutin, Buchgraber, Kulkarni, Poor'11

²³Tan, Li'10, ²⁴Duan, Yang, Fang, Li'17

²⁵Zachariah, Stoica'15

Complexity Comparisons-SBL Algorithms

Algorithm	Complexity per Iteration	Convergence (No of iterations)	Sparsity	Optimization function	Local Optimum
Type I	$O(M^3)$		Exact sparsity	Type I ML (Depending upon the prior used, type 1 ML corresponds to LASSO/Re-weighted l1/l2 min. problems)	
Type II SBL	$O(M^3)$		Exact sparsity (α_i converges to ∞)	Type II ML solved using EM	
Fast SBL using Type II ML (Tipping,Faul'03) (Focus more on Convergence speed)	$O(L^3)$, $L \leq M$	$\ll L$	Exact sparsity (Using an entry dependent thresholding condition which follows from the computation of stationary point of α_i)	Type II ML (stationary points of α_i are computed to accelerate convergence)	Convergence to a local optimum.
Fast SBL (using VB) by Shutin (Focus more on Convergence speed)	$O(L^3)$, $L \leq M$	$\ll L$	Exact sparsity (Using a pruning condition similar as in Tipping's)	Maximization of ELBO in VB	Convergence to a local optimum of ELBO (Mean field free energy)
Hyperparameter free SBL (Zachariah, Stoica'15)	$O(M^2)$	$\ll M$	The final objective function is a weighted square root LASSO. So the sum of l2 norm of (y and Ax) and weighted l1 norm of x which promotes sparsity here.	MMSE estimator for x with Covariance matching for PDP, finally giving rise to an objective function which can be interpreted as weighted square root LASSO.	Convergence to a local optimum
BP-SBL (Tan, Li'10)	$O(MN)$ (Similar complexity as xAMP, see matrix form of the BP-SBL in the upcoming slides)	$\log(MN)$	Does not give exact sparsity	Posterior of x computed using BP and EM for hyper-parameters	Convergence to local optimum of Bethe Free Energy (BFE)
GAMP-SBL (Shoukairi, Schniter, Rao'18)	$O(MN)$	$\ll M$	Does not give exact sparsity	Using GAMP for posterior of x, EM for hyperparameters	Convergence to local optimum of LSL-BFE
SAVE	$O(MN)$	$\ll M$	Does not give exact sparsity	Maximization of ELBO in VB	Convergence to a local optimum of ELBO
Inverse Free SBL (Duan, Yang, Fang, Li'17)	$O(MN)$	$\ll M$ (similar to GAMP SBL)	Does not give Exact sparsity	Maximization of an approximate ELBO in VB	Convergence to a local opt in the approx ELBO

References I

-  M. Bayati, A. Montanari, "The Dynamics of Message Passing on Dense Graphs, with Applications to Compressed Sensing," in *IEEE Trans. on Info. Theo.*, Feb. 2011.
-  M. J. Beal, "Variational Algorithms for Approximate Bayesian Inference," in *Thesis, Univeristy of Cambridge, UK*, May 2003.
-  M. Borgerding, P. Schniter, S. Rangan, "AMP-Inspired Deep Networks for Sparse Linear Inverse Problems," *IEEE Trans. on Sig. Process.*, Aug. 2017.
-  B. Çakmak, M. Opper, "Expectation Propagation for Approximate Inference: Free Probability Framework," in *IEEE Inter. Sympo. On Info. Theo. (ISIT)*, 2018.
-  B. Çakmak, O. Winther, B. H. Fleury, "S-AMP: Approximate Message Passing for General Matrix Ensembles," in *IEEE Intl. Sympo. Info. Theo.*, 2014.
-  E. Candes, T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n ," *The annals of Statistics*, 2007.
-  S. S. Chen, D. L. Donoho, M. A. Saunders, "Atomic decomposition by Basis Pursuit," *SIAM J. Sci. Comput.*, 1999.
-  R. Couillet, J. Hoydis, M. Debbah", "Random Beamforming over Quasi-Static and Fading Channels: A Deterministic Equivalent Approach," in *IEEE Trans. On Info. Theo.*, 2012.
-  I. Daubechies, M. Defrise, C. D. Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint..," *Comm. on Pure and Applied Mathematics*, 2004.
-  J. Du, S. Ma, Y-C. Wu, S. Kar, J. M. F. Moura, "Convergence Analysis of Distributed Inference with Vector-Valued Gaussian Belief Propagation," in *Jrnl. of Mach. Learn. Res.*, Apr. 2018.
-  H. Duan, L. Yang, J. Fang, H. Li, "Fast Inverse-Free Sparse Bayesian Learning via Relaxed Evidence Lower Bound Maximization," in *IEEE Sig. Process. Lett.*, Jun. 2017.
-  Y. C. Eldar "Rethinking Biased Estimation: Improving Maximum Likelihood and the CramérRao Bound," *Found. and Tren. in Sig. Process.*, 2008.

References II

-  S. Fortunati, F. Gini, M. S. Greco, C. D. Richmond, "Performance Bounds for Parameter Estimation under Misspecified Models: Fundamental Findings and Applications," in *IEEE Sig. Proc. Mag.*, Nov. 2017.
-  A. E. Gelfand, S. K. Sahu, "Identifiability, Improper Priors, and Gibbs Sampling for Generalized Linear Models," in *Journ. of the Americ. Stat. Assoc.*, Mar. 1999.
-  R. Giri, B. Rao, "Type I and Type II Bayesian Methods for Sparse Signal Recovery Using Scale Mixtures," in *IEEE Trans. Sig. Process.*, July 2016.
-  K. Gregor, Y. LeCun, "Learning Fast Approximations of Sparse Coding," *Intl. Conf. on Mach. Learn.*, 2010.
-  W. James, C. Stein "Estimation with quadratic loss," *Proc. 4th Berkeley Symp. Mathematical Statistics Probability*, 1961.
-  J. K. Johnson, D. Bickson, D. Dolev, "Fixing Convergence of Gaussian Belief Propagation," in *IEEE Intl. Symp. on Info. Theo.*, 2009.
-  M. Luo, Q. Guo, D. Huang, J. Xi, "Sparse Bayesian Learning using Approximate Message Passing with Unitary Transformation," in *IEEE VTS Asia Pac. Wire. Commun. Symp., APWCS*, Aug. 2019.
-  D. M. Malioutov, J. K. Johnson, A. S. Willsky, "Walk-Sums and Belief Propagation in Gaussian Graphical Models," in *Jrnl. of Mach. Learn. Res.*, Oct. 2006.
-  S. Mallat, Z. Zhang, "Matching Pursuits with time-frequency dictionaries," *IEEE Trans. Sig. Process.*, 1993.
-  J. Ma, L. Ping "Orthogonal AMP," in *IEEE Access*, Mar. 2017.
-  T. P. Minka, "Expectation propagation for approximate Bayesian inference , " in *Proc. of Conf. on Uncert. in Art. Intell. (UAI)*, 2001.
-  B. Mu, T. Chen, L. Ljung, "On asymptotic properties of hyperparameter estimators for kernel-based regularization methods," in *Automatica*, May. 2018.
-  D. Needell, J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples., " *Applied and computational harmonic analysis* , 2009.



References III

-  G. Pillonetto, F. Dinuzzo, T. Chen, G. D Nicolao, L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, Feb. 2014.
-  G. Pillonetto, G. D. Nicolao, "A new kernel-based approach for linear system identification," *Automatica*, 2010.
-  S. Rangan, "Generalized Approximate Message Passing for Estimation with Random Linear Mixing," in *IEEE Intl. Sympo. Info. Theo.*, 2011.
-  S. Rangan, A. K. Fletcher, P. Schniter, U. S. Kamilov, "Inference for Generalized Linear Models via Alternating Directions and Bethe Free Energy Minimization," in *IEEE Trans. Inf. Theory*, Jan. 2017.
-  S. Rangan, P. Schniter, A. K. Fletcher, "Vector Approximate Message Passing," in *IEEE Trans. Inf. Theory*, Oct. 2019.
-  C. D. Richmond and L. L. Horowitz, "Parameter bounds on estimation accuracy under model misspecification," in *IEEE Trans. on Sig. Process.*, May. 2015.
-  E. Riegler, G. E. Kirkelund, C. N. Manchón, B. H. Fleury, "Merging Belief Propagation and the Mean Field Approximation: a Free Energy Approach," in *IEEE Trans. on Info. Theo.*, Jan. 2013.
-  P. Rusmevichtong, B. Van Roy, "An analysis of belief propagation on the turbo decoding graph with Gaussian densities," *IEEE Trans. Inf. Theory*, 2001.
-  M. Al-Shoukairi, P. Schniter, B. D. Rao, "GAMP-Based Low Complexity Sparse Bayesian Learning Algorithm," in *IEEE Trans. on Sig. Process.*, Jan. 2018.
-  D. Shutin, T. Buchgraber, S. R. Kulkarni, H. V. Poor, "Fast Variational Sparse Bayesian Learning With Automatic Relevance Determination for Superimposed Signals," in *IEEE Trans. Sig. Process.*, Dec. 2011.
-  V. Šmídl, A. Quinn "The Variational Bayes Method in Signal Processing," in *Springer Series on Sig. and Comm. Tech.*, 2005.
-  K. Takeuchi, "Rigorous Dynamics of Expectation- Propagation-Based Signal Recovery from Unitarily Invariant Measurements," in *IEEE Trans. Inf. Theory*, Jan. 2020.
-  X. Tan, J. Li, "Computationally Efficient Sparse Bayesian Learning via Belief Propagation," in *IEEE Trans. on Sig. Proc.*, Apr. 2010.

References IV

-  C. K. Thomas, K. Gopala, D. Slock, "Sparse Bayesian learning for a bilinear calibration model and mismatched CRB," in *EUSIPCO*, 2019.
-  C. K. Thomas, D. Slock, "SAVE - space alternating variational estimation for sparse Bayesian learning," in *IEEE Data Sci. Wkshp.*, Jun. 2018.
-  C. K. Thomas, D. Slock, "Space Alternating Variational Estimation and Kronecker Structured Dictionary Learning," in *IEEE ICASSP*, 2019.
-  C. K. Thomas and D. Slock, "Convergence Analysis Of Sparse Bayesian Learning under Approximate Inference Techniques," in *Asilomar Conf. on Sig., Sys., and Comp.*, Nov. 2019.
-  C. K. Thomas and D. Slock, "Low Complexity Static and Dynamic Sparse Bayesian Learning Combining BP, VB and EP Message Passing," in *IEEE Asilomar*, Nov. 2019.
-  R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. Roy. Statist. Soc. Series B (Methodol.)*, 1996.
-  M. E. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," *J. Mach. Learn. Res.*, 2001.
-  M. E. Tipping, A. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *AISTATS*, 2003.
-  J. A. Tropp, A. C. Gilbert, "Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit," *IEEE Trans. Info. Theo.*, 2007.
-  D. G. Tzikas, A. C. Likas, N. P. Galatsanos, "The variational approximation for Bayesian inference," in *IEEE Sig. Process. Mag.*, Nov. 2008.
-  S. Wagner, R. Couillet, M. Debbah, D. Slock, "Large System Analysis of Linear Precoding in MISO Broadcast Channels with Limited Feedback," in *IEEE Trans. Inf. Theory*, July. 2012.
-  Y. Weiss, W. T. Freeman, "Correctness of belief propagation in Gaussian graphical models of arbitrary topology," *NIPS*, 2000.
-  D. P. Wipf, B. D. Rao, "Sparse Bayesian Learning for Basis Selection," *IEEE Trans. Sig. Process.*, Aug 2004.
-  J. S. Yedidia, W. T. Freeman, Y. Weiss, "Constructing free-energy approximations and generalized belief propagation algorithms," in *IEEE Trans. on Info. Theo.*, Jun. 2005.
-  D. Zachariah, P. Stoica, "Online Hyperparameter-Free Sparse Estimation Method," in *IEEE Trans. on Sig. Proc.*, July. 2015.