Adaptive SP & Machine Intelligence
Lecture 4: Modern Spectral Estimation

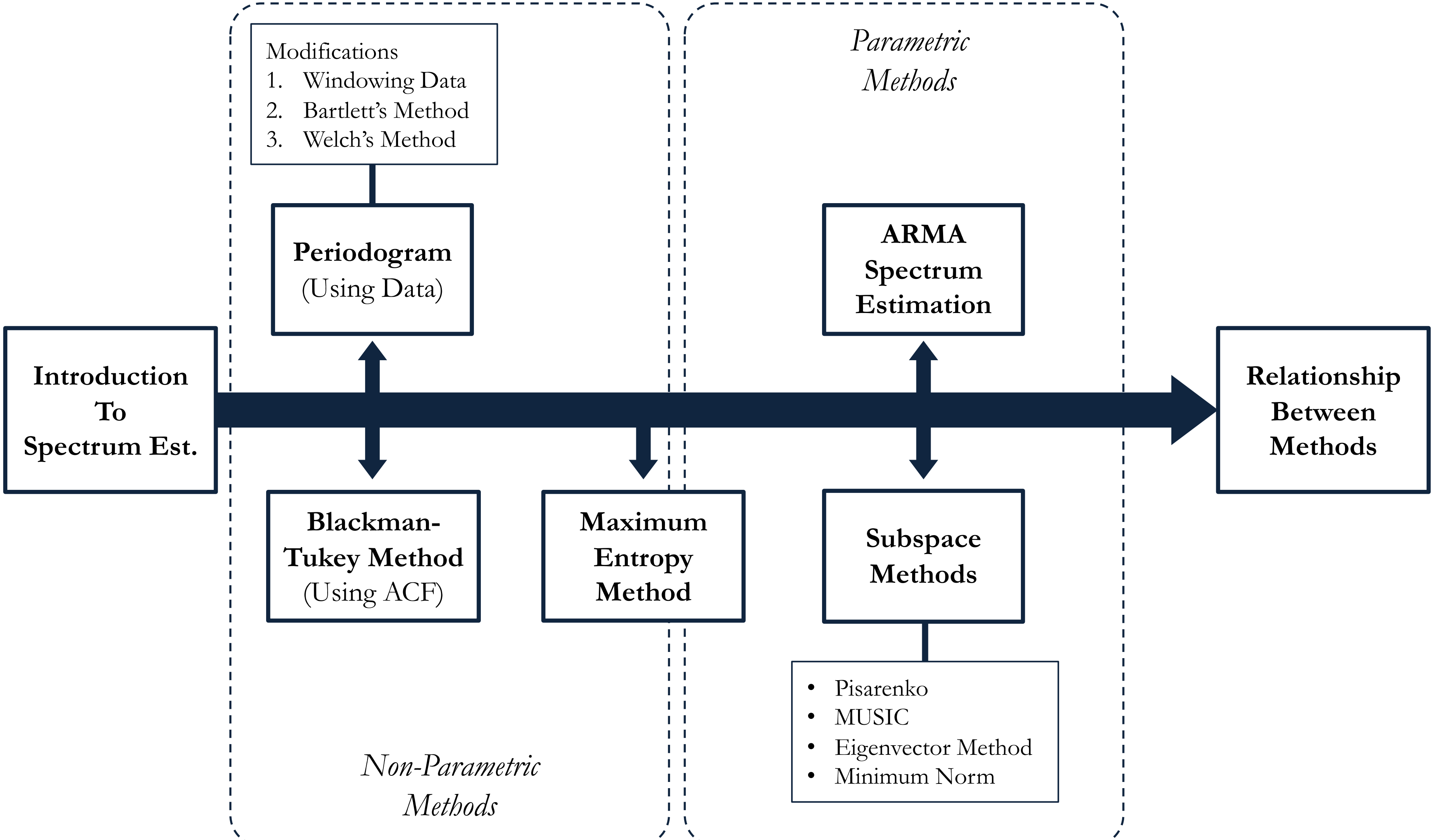
Danilo P. Mandic

Room: 813 Ext: 46271

Department of Electrical and Electronic Engineering
Imperial College London, UK

d.mandic@imperial.ac.uk URL: www.commsp.ee.ic.ac.uk/~mandic

Overview of Spectral Estimation Methods



Periodogram Based Methods

Periodogram

$$\hat{P}_{per}(\omega_m) = \frac{1}{N}\left|\sum_{k=0}^{N-1} x[k]e^{-\jmath\omega_m k}\right|^2$$

Windowing
Modified Periodogram

$$\hat{P}_{mod}(\omega_m) = \frac{1}{NU}\left|\sum_{k=0}^{N-1} w[k]x[k]e^{-\jmath\omega_m k}\right|^2$$

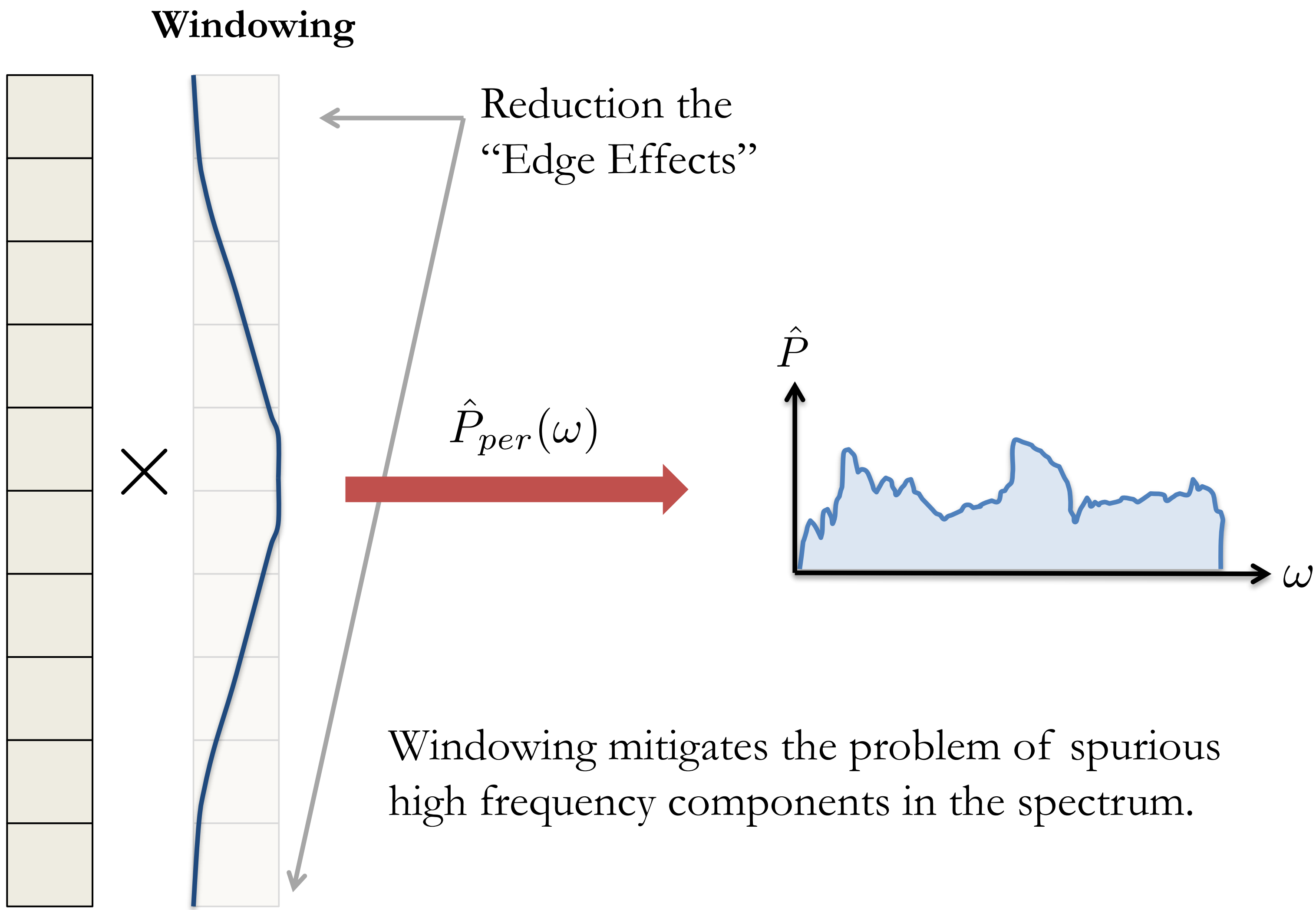
Averaging
Bartlett's Method

$$\hat{P}_{B}(\omega_m) = \frac{1}{N}\sum_{i=0}^{K-1}\left|\sum_{k=0}^{N-1} x[k+iL]e^{-\jmath\omega_m k}\right|^2$$

+ Overlapping windows
Welch's Method

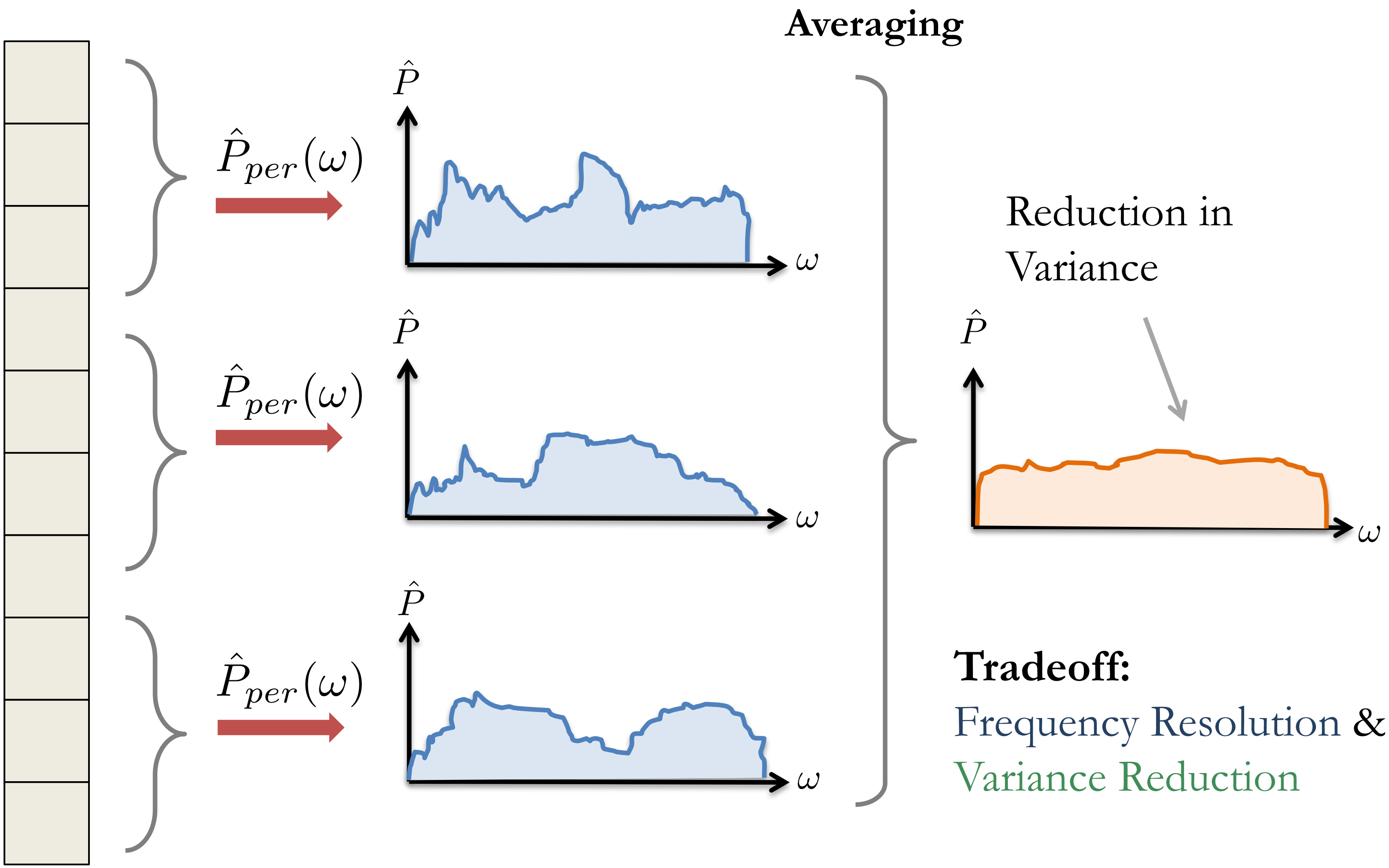
$$\hat{P}_{W}(\omega_m) = \frac{1}{KLU}\sum_{i=0}^{K-1}\left|\sum_{k=0}^{N-1} w[k]x[k+iD]e^{-\jmath\omega_m k}\right|^2$$

Modified Periodogram



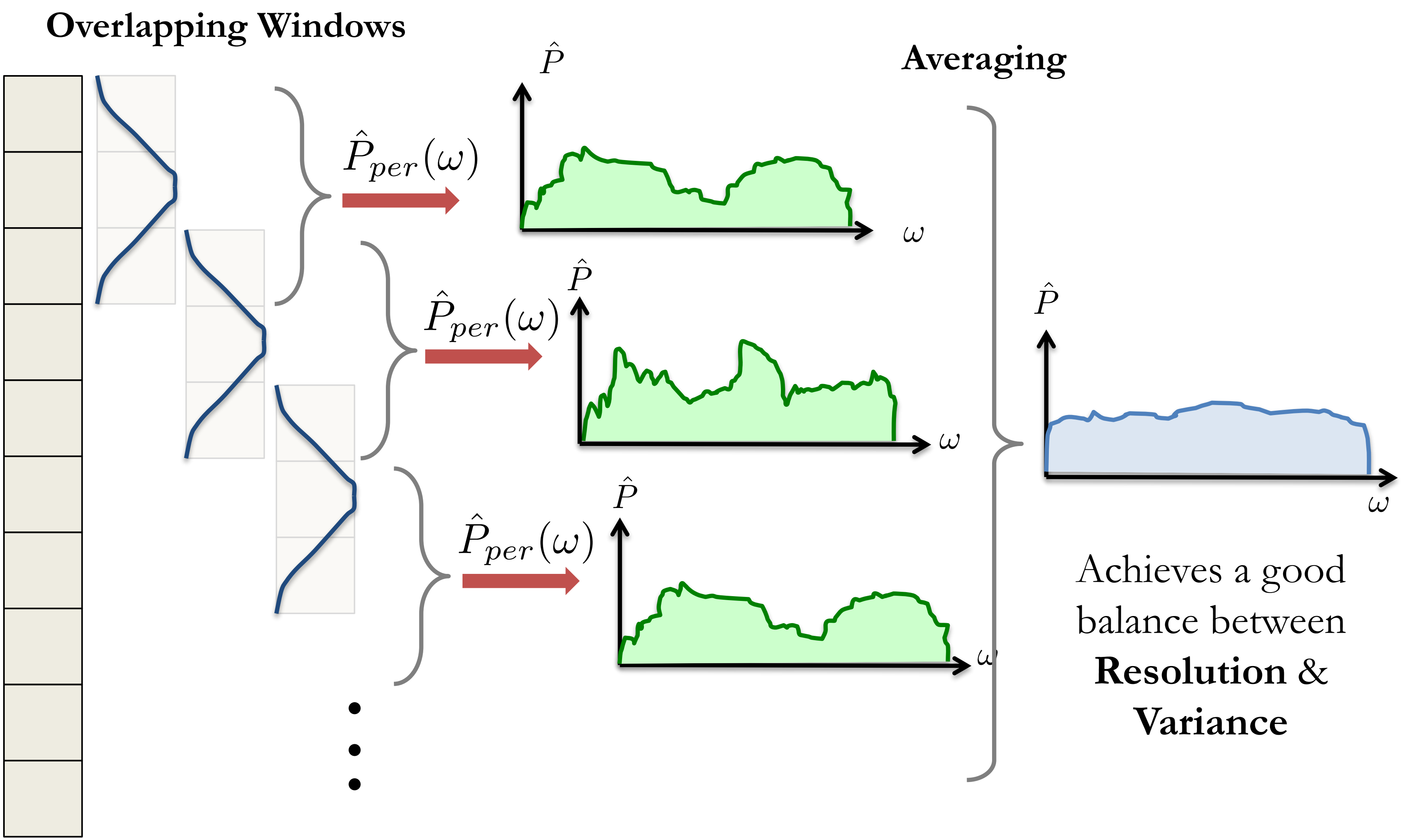
Windowing mitigates the problem of spurious high frequency components in the spectrum.

Bartlett's Method



Tradeoff:
Frequency Resolution &
Variance Reduction

Welch's Method



Achieves a good balance between **Resolution & Variance**

Blackman-Tukey Method

The Periodogram can also be expressed as:

$$\hat{P}_{per}(\omega_m) = \sum_{k=-N+1}^{N-1} \hat{\mathbf{r}}_{xx}[k] e^{-\jmath \omega_m k}$$

Autocorrelation Estimates at large lags are **unreliable**

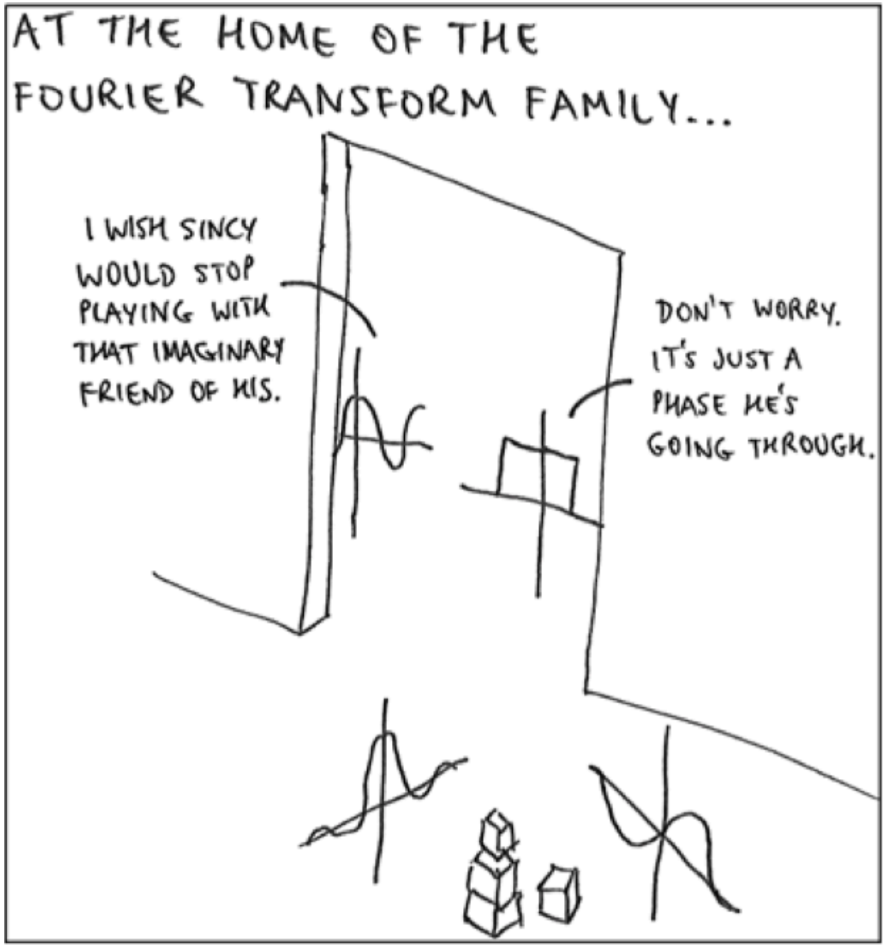
$$\hat{P}_{BT}(\omega_m) = \sum_{k=-M}^{M} w[k] \hat{\mathbf{r}}_{xx}[k] e^{-\jmath \omega_m k}$$

Lags: $M < N - 1$

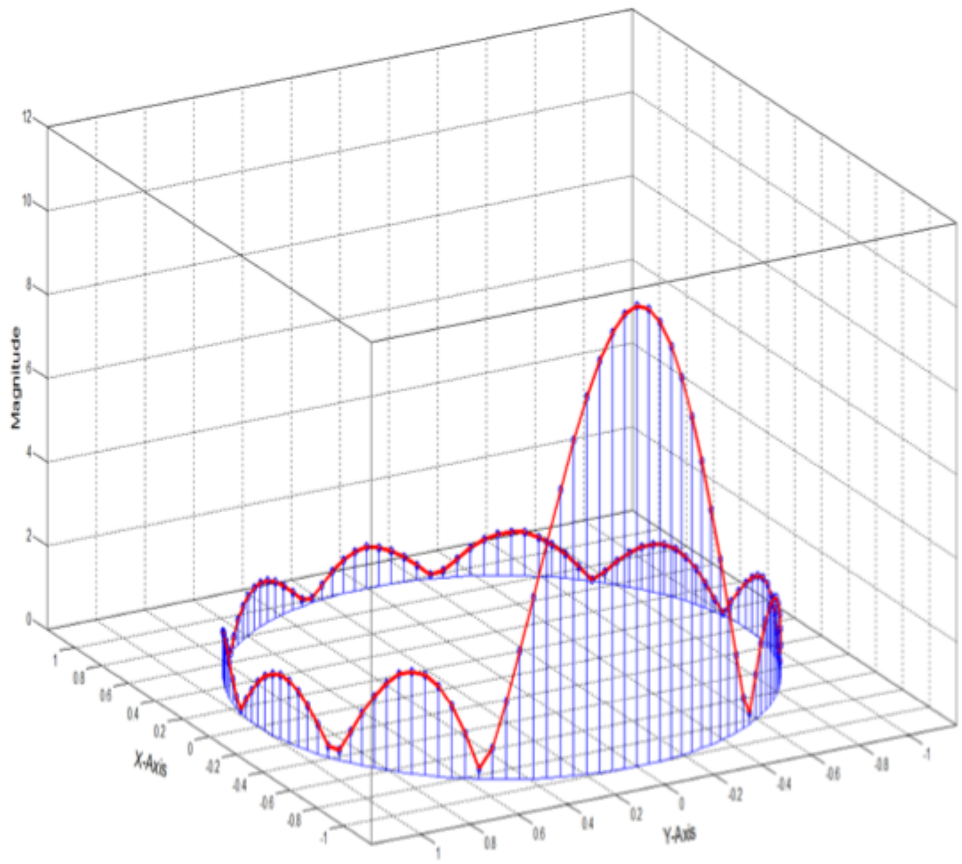
Windowing

Next: Can we **extrapolate the autocorrelation** estimates for lags $k > M$?

But, the main problem remains the same …

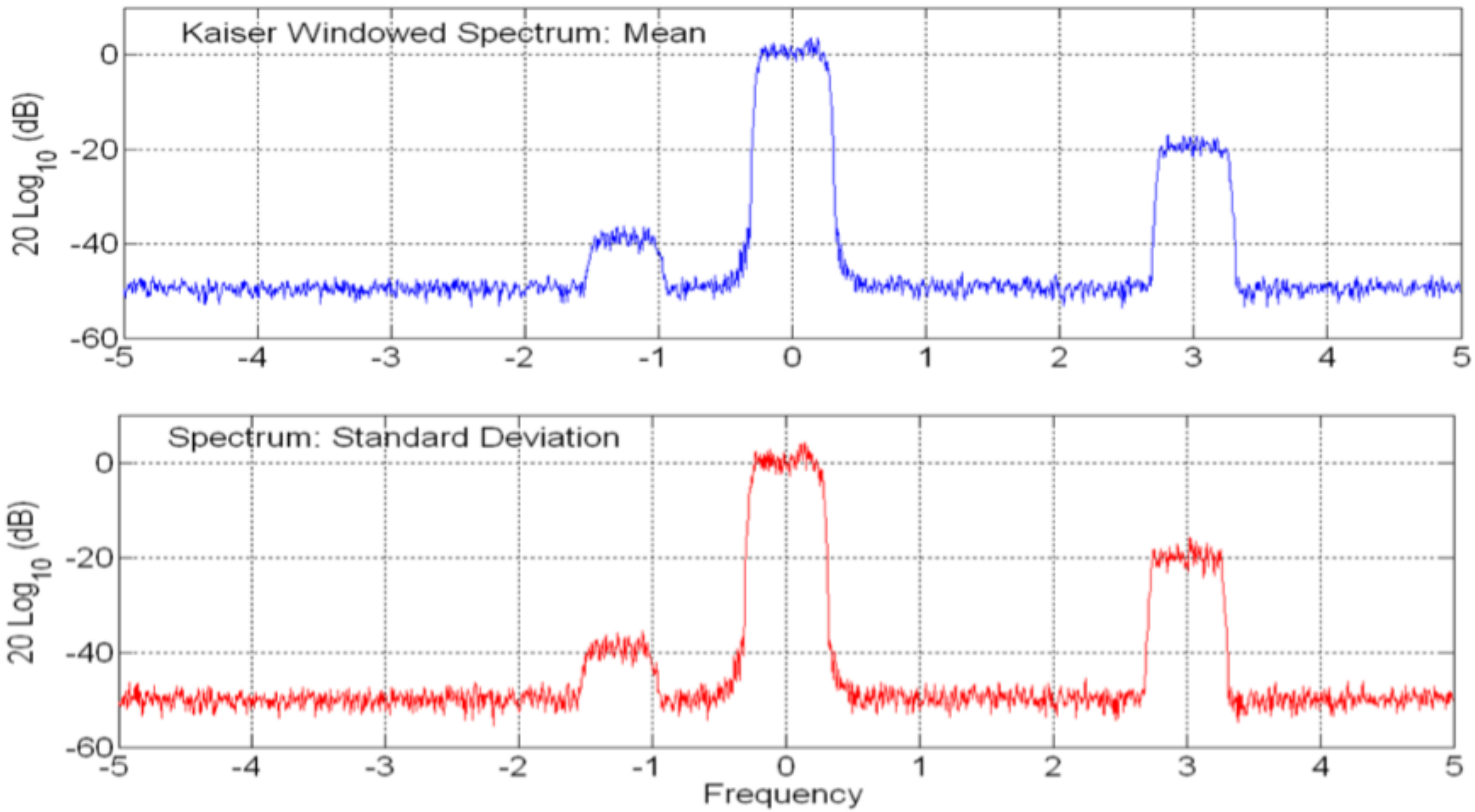


(credit, Fred Harris)

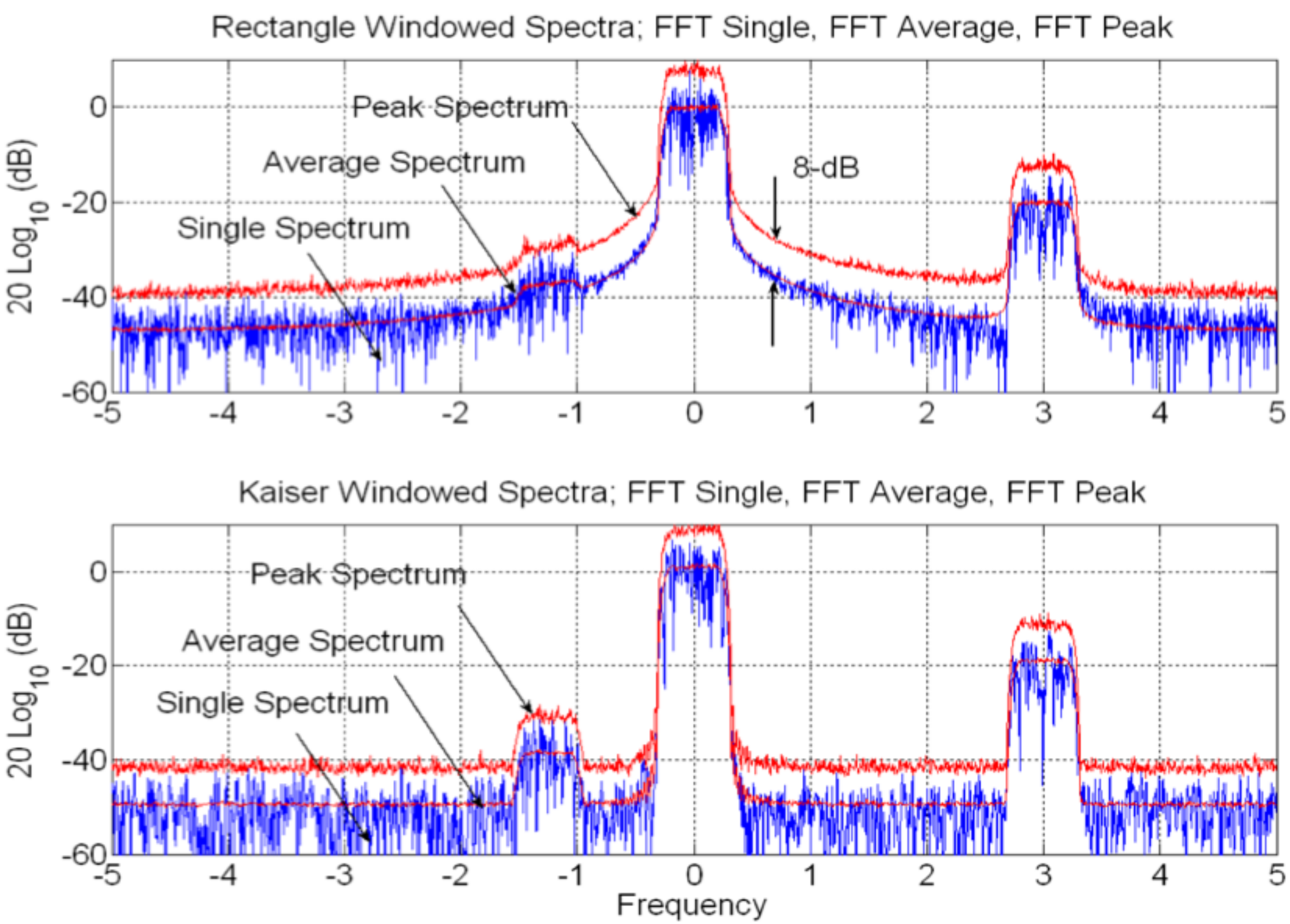


Sinc on a unit circle

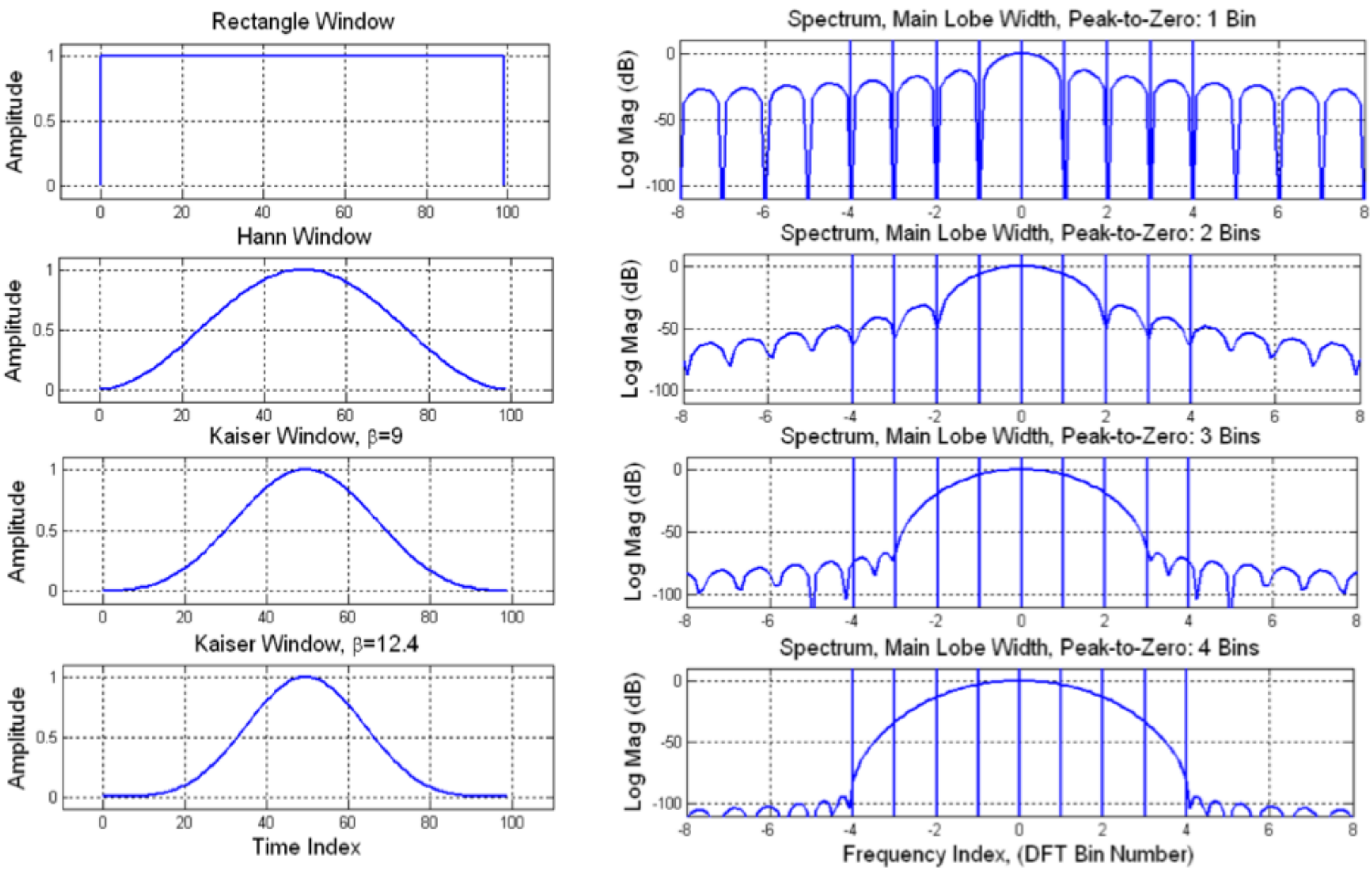
Fourier methods yield inconsistent estimates



Can we somehow improve 'peak to average'



Are we effectively amplifying noise by windowing



Maximum Entropy Method

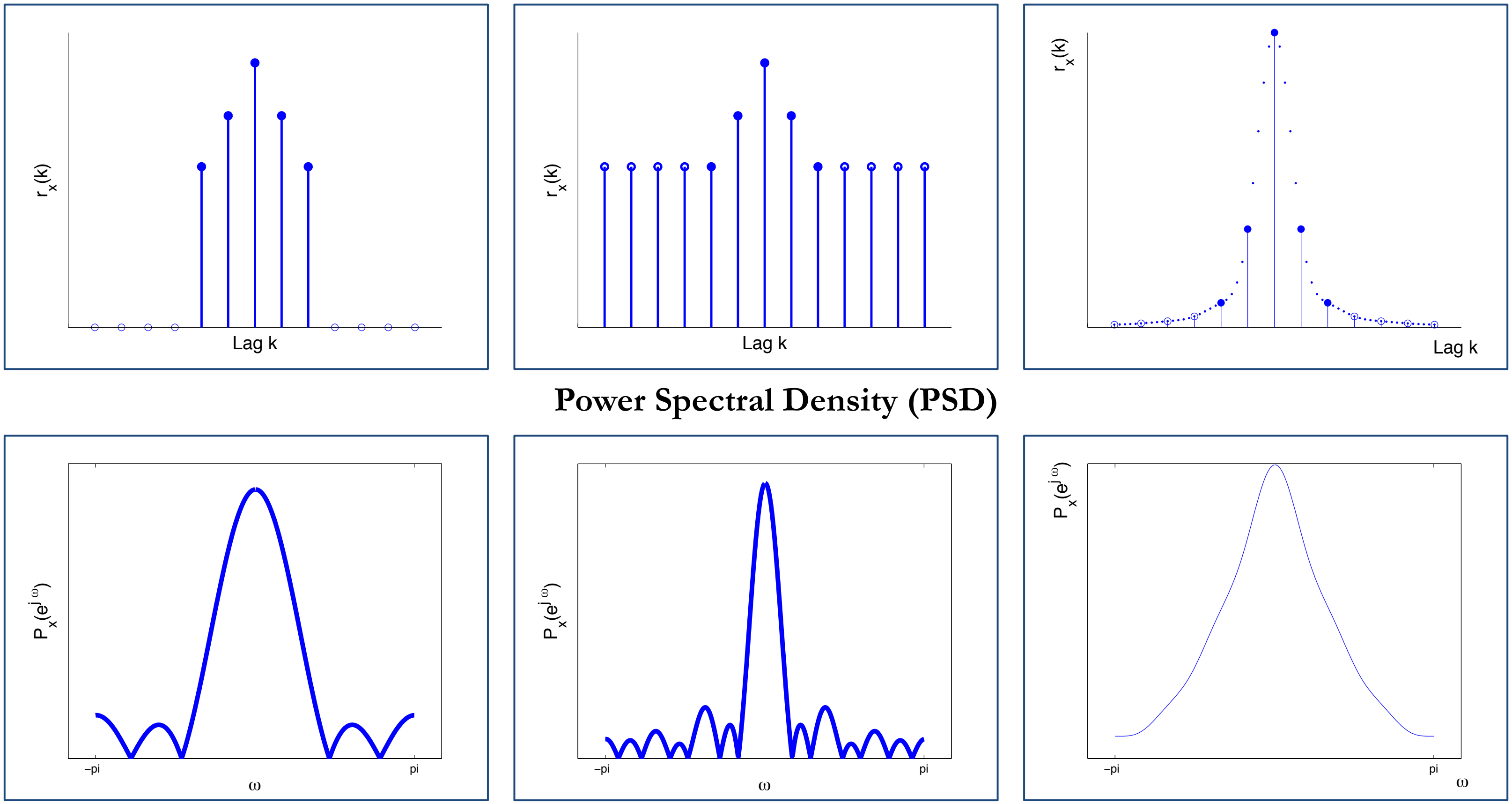
How can we **extrapolate the autocorrelation** estimates with imposing the **least amount of structure on the data**?

$\Longrightarrow$ Maximize the randomness $\Longrightarrow$ **Maximize Entropy**

Which one has the "flattest" PSD?

Autocorrelation Sequences

Power Spectral Density (PSD)



Maximum Entropy Method (MEM)

Entropy of Gaussian random process $x(n)$ with PSD $P_{xx}(\omega)$:

$$H(x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \ln P_{xx}(\omega)\, d\omega$$

Goal: Find extrapolated autocorrelation values $r_e(k)$ to maximize the entropy:

$$\frac{\partial H(x)}{\partial r_e^*(k)} = 0, \quad \text{for: } |k| > p$$

**Refer to handout for the full derivation*

$$\boxed{\hat{P}_{mem}(\omega) = \frac{\sigma_w^2}{\left|1 + \sum_{k=1}^{p} \hat{a}_k e^{-\jmath k\omega}\right|^2}}$$

Estimated using the Yule-Walker Method

The MEM method is **identical to the all-pole AR(p) spectrum** although **no assumptions were made** about the model of the data (except Gaussianity).

MEM, derivation

$\Rightarrow$ for a Gaussian process with a given autocorr. sequence $r_x(k)$ for $|k| \leq p$ the Maximum Entropy Power Spectrum minimises entropy $H(x)$

subject to the constraint that the inverse DFT of $P_{xx}(\omega)$ equals the given set of autocorrelations for $|k| \leq p$, that is

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} P_{xx}(\omega) e^{\jmath k\omega} d\omega = r_x(k) \qquad |k| \leq p$$

The values of $r_e(k)$ that maximize the entropy may be found by setting the derivative of $H(x)$ wrt $r_e^*(k)$ equal to zero:

$$\frac{\partial H(x)}{\partial r_e^*(k)} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1}{P_{xx}(\omega)} \frac{\partial P_{xx}(\omega)}{\partial r_e^*} d\omega = 0 \qquad |k| > p$$

Notice that $\frac{\partial P_{xx}(\omega)}{\partial r_e^*} = e^{\jmath k\omega} \quad \Rightarrow \quad \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1}{P_{xx}(\omega)} e^{\jmath k\omega} d\omega = 0, \quad |k| > p.$

MEM, spectrum

Therefore:

$$Q_{xx}(\omega) = \frac{1}{P_{xx}(\omega)} = \sum_{k=-p}^{p} q_{xx}(k) e^{-\jmath k\omega}$$

$\Rightarrow \hat{P}_{mem}$ is an all–pole spectrum, given by

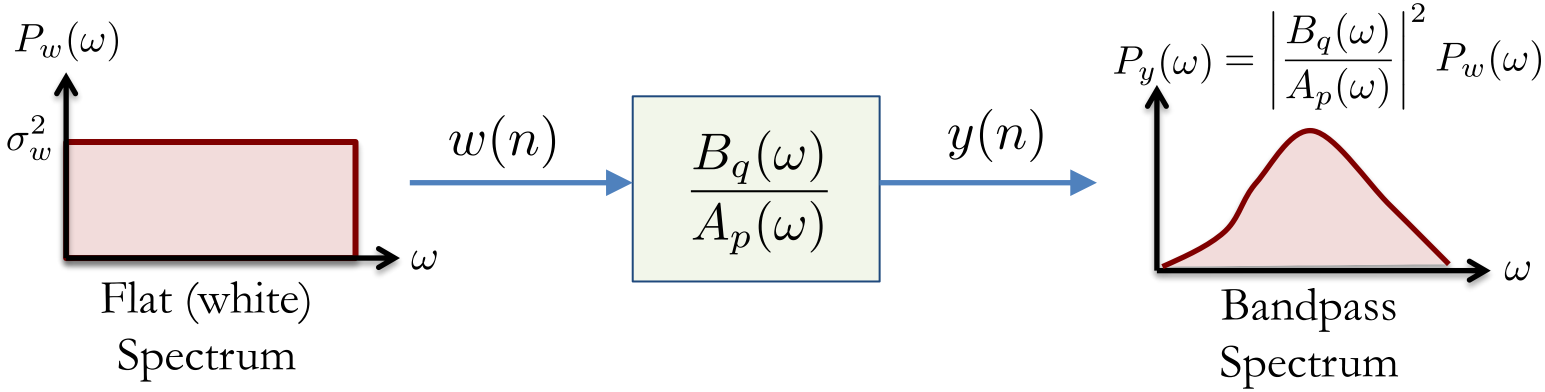
$$\hat{P}_{mem}(\omega) = \frac{|b(0)|^2}{A_p(\omega)A_p^*(\omega)} = \frac{|b(0)|^2}{|1 + \sum_{k=1}^{p} a_p(k)e^{-\jmath k\omega}|^2}$$

Alternatively

$$\hat{P}_{mem}(\omega) = \frac{|b(0)|^2}{|\mathbf{e}^H \mathbf{a}_p|^2}$$

Coefficients $\mathbf{a}[1, a_p(1), \ldots, a_p(p)]^T$ and $b(0)$ are found from the normal equations (Yule–Walker).

Power spectra of ARMA processes



$$y(n) = \underbrace{-\sum_{k=1}^{p} a_k y(n-k)}_{\substack{\textbf{Autoregressive} \\ \text{AR(p)}}} + \underbrace{\sum_{k=0}^{q} b_k w(n-k)}_{\substack{\textbf{Moving Average} \\ \text{MA(q)}}}$$

$$\hat{P}_{ARMA}(\omega) = \frac{\left|\sum_{k=0}^{q} \hat{b}_k e^{-\jmath k\omega}\right|^2}{\left|1 + \sum_{k=1}^{p} \hat{a}_k e^{-\jmath k\omega}\right|^2}$$

Recap: ARMA processes

Random processes $x[n]$ and $w[n]$ are related by a linear difference equation with constant coefficients, given by

$$H(z) = \frac{X(z)}{W(z)} = \frac{B(z)}{A(z)} \quad \leftrightarrow \text{ARMA(p,q)} \leftrightarrow \quad x[n] = \underbrace{\sum_{l=1}^{p} a_l x[n-l]}_{autoregressive} + \underbrace{\sum_{l=0}^{q} b_l w[n-l]}_{moving\ average}$$

Notice that the autocorrelation function of $x[n]$ and crosscorrelation between the **stochastic process** $x[n]$ and **the driving input** $w[n]$ follow the same difference equation, i.e. if we multiply both sides of the above equation by $x[n-k]$ and take the statistical expectation, we have

$$r_{xx}(k) = \underbrace{\sum_{l=1}^{p} a_l\, r_{xx}(k-l)}_{\text{easy to calculate}} + \underbrace{\sum_{l=0}^{q} b_l\, r_{xw}(k-l)}_{\text{can be complicated}}$$

Since x is WSS, it follows that $x[n]$ and $w[n]$ are jointly WSS

Recap: Yule-Walker equations

For $k = 1, 2, \ldots, p$ from the general autocorrelation function, we obtain a set of equations:

$$
\begin{aligned}
r_{xx}(1) &= a_1 r_{xx}(0) + a_2 r_{xx}(1) + \cdots + a_p r_{xx}(p-1) \\
r_{xx}(2) &= a_1 r_{xx}(1) + a_2 r_{xx}(0) + \cdots + a_p r_{xx}(p-2) \\
\vdots &= \vdots \\
r_{xx}(p) &= a_1 r_{xx}(p-1) + a_2 r_{xx}(p-2) + \cdots + a_p r_{xx}(0)
\end{aligned}
$$

These equations are called the **Yule–Walker or normal equations.**

Their solution gives us the set of **autoregressive parameters** $\mathbf{a} = [a_1, \ldots, a_p]^T$.

The above can be expressed in a vector–matrix form as

$$\mathbf{r}_{xx} = \mathbf{R}_{xx}\mathbf{a} \quad \Rightarrow \quad \mathbf{a} = \mathbf{R}_{xx}^{-1}\mathbf{r}_{xx}$$

The ACF matrix $\mathbf{R}_{xx}$ is positive definite (Toeplitz) which guarantees matrix inversion

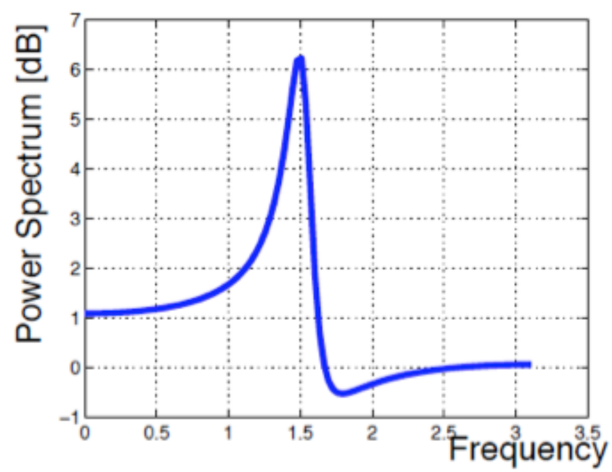
Recap: ARMA processes

Plot the power spectrum of an ARMA(2,2) process for which

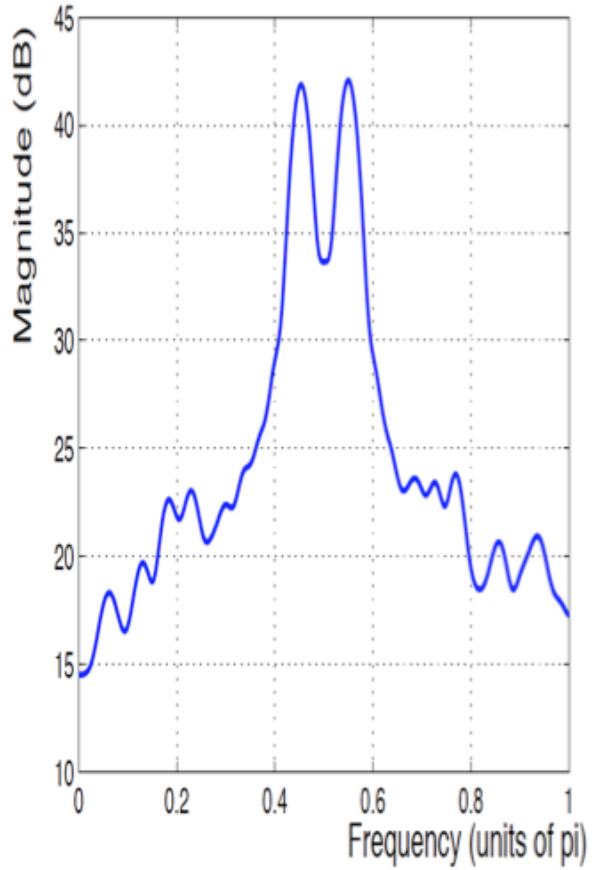
- the zeros of $H(z)$ are $z = 0.95e^{\pm \jmath\pi/2}$
- poles are at $z = 0.9e^{\pm \jmath 2\pi/5}$

Solution: The system function is (poles and zeros – resonance & sink)

$$H(z) = \frac{1 + 0.9025z^{-2}}{1 - 0.5562z^{-1} + 0.81z^{-2}}$$



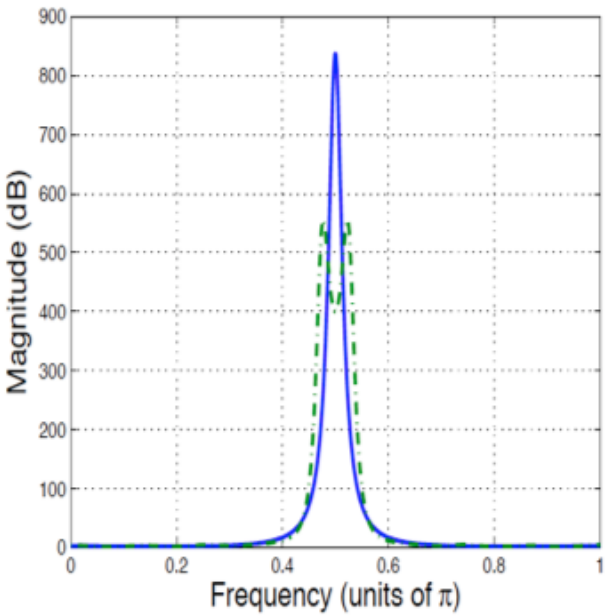
ARMA spectrum estimation



All–pole model capable of estimating two sinewaves in noise!

Blackman-Tukey and MA: Smooth spectra $\hookrightarrow$ not suitable here

Consider an AR(2) signal $x(n) = -0.9x(n-2) + w(n)$ with $w \sim \mathcal{N}(0,1)$.
Consider $N = 64$ data samples, and model orders $p = 4$ and $p = 12$.



Notice that this is an AR(2) model!

Although the true spectrum has a single spectral peak at $\omega - \pi/2$ (blue), when overmodelling using $p = 12$ this peak is split into two peaks (green).

Complex AR modelling (from Lecture 2)

Standard AR model of order n is given by

$$z(k) = a_1 z(k-1) + \cdots + a_n z(k-n) + q(k) = \mathbf{a}^T \mathbf{z}(k) + q(k),$$

Using the Yule-Walker equations the AR coefficients are found from

$$\mathbf{a}^* = \mathcal{C}^{-1}\mathbf{c}$$

$$\begin{bmatrix} a_1^* \\ a_2^* \\ \vdots \\ a_n^* \end{bmatrix} = \begin{bmatrix} c(0) & c^*(1) & \dots & c^*(n-1) \\ c(1) & c(0) & \dots & c^*(n-2) \\ \vdots & \vdots & \ddots & \vdots \\ c(n-1) & c(n-2) & \dots & c(0) \end{bmatrix}^{-1} \begin{bmatrix} c(1) \\ c(2) \\ \vdots \\ c(n) \end{bmatrix}$$

where $\mathbf{c} = [c(1), c(2), \dots, c(n)]^T$ is the time shifted correlation vector.

Widely linear model

$$y(k) = \mathbf{h}^T(k)\mathbf{x}(k) + \mathbf{g}^T(k)\mathbf{x}^*(k) + q(k)$$

Widely linear normal equations

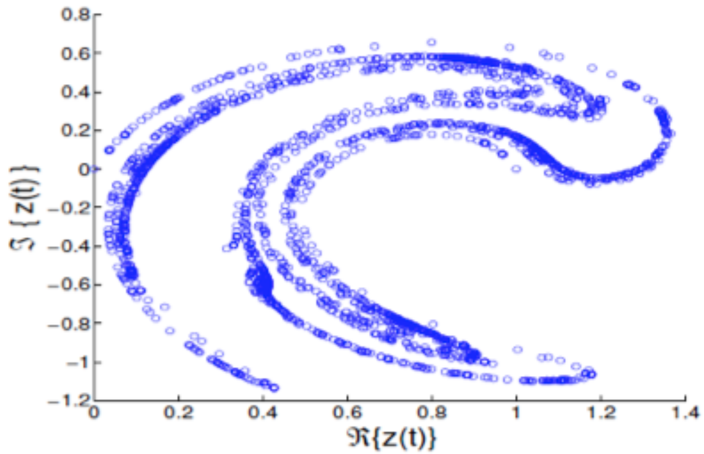
$$\begin{bmatrix} \mathbf{h}^* \\ \mathbf{g}^* \end{bmatrix} = \begin{bmatrix} \mathcal{C} & \mathcal{P} \\ \mathcal{P}^* & \mathcal{C}^* \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{c} \\ \mathbf{p}^* \end{bmatrix}$$

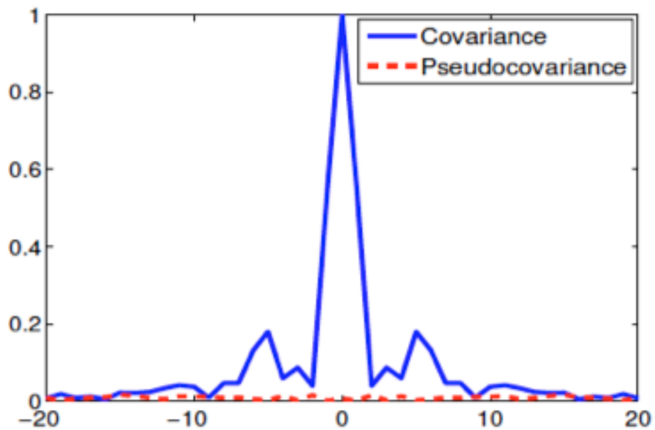
where $\mathbf{h}$ and $\mathbf{g}$ are coefficient vectors and $\mathbf{x}$ the regressor vector.

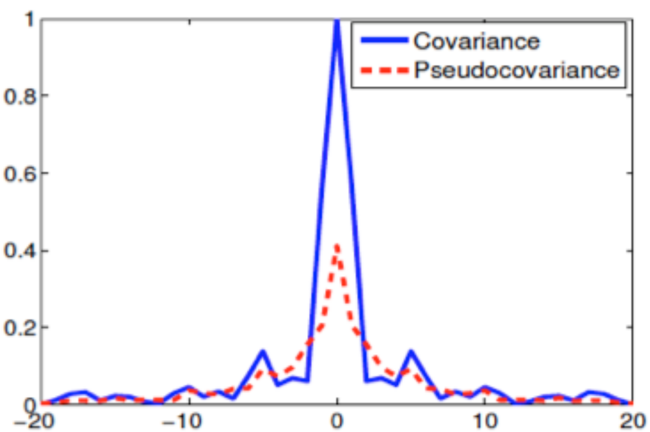
Complex AR modelling, simulations

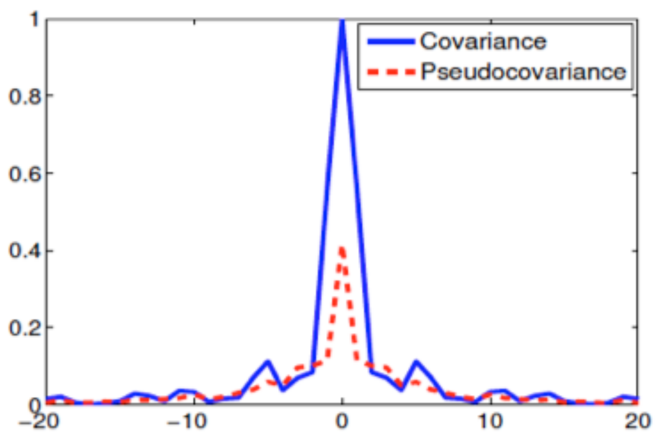
Circularity for Ikeda map

AR model of Ikeda signal







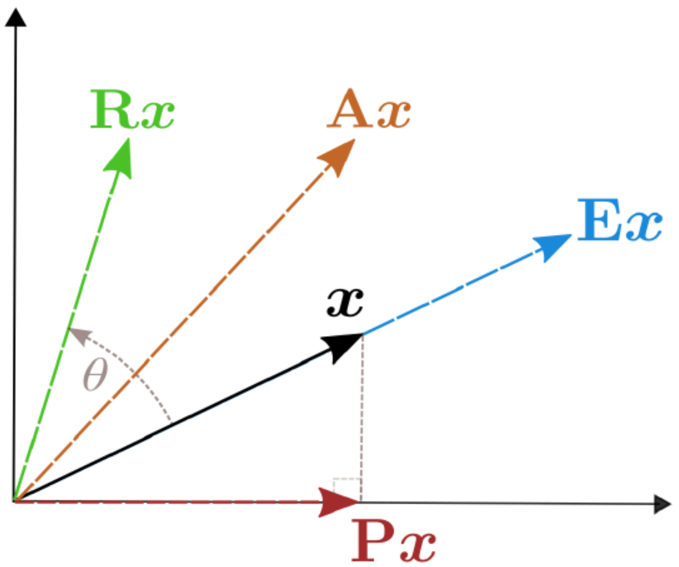


Covariances: Original Ikeda

Widely linear AR of Ikeda

Subspace Methods: Introduction

What is that a matrix does to a vector?



$\mathbf{A} \rightsquigarrow$ any general matrix

$\mathbf{R} \rightsquigarrow$ a rotation matrix ($\mathbf{R}^T = \mathbf{R}^{-1}$ and $\det \mathbf{R} = 1$)

$\mathbf{E}\mathbf{x} = \lambda \mathbf{x} \rightsquigarrow$ eigenanalysis

$\mathbf{P} \rightsquigarrow$ projection matrix

An example of a rotation matrix

$$\mathbf{R} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

Ampli-twist

A matrix $\mathbf{A}$ which multiplies a vector $\mathbf{x}$

(i) stretches or shortents the vector

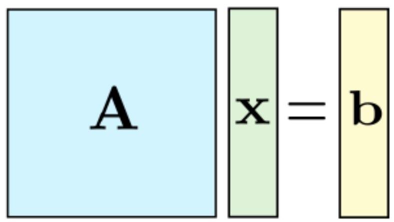
(ii) rotates the vector

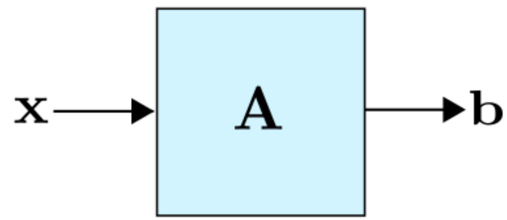
What can we say about the properties of the matrix $\mathbf{A}$, matrix $\mathbf{E}$ and the projection matrix $\mathbf{P}$ (rank, invertibility, ...)?

Is the projection matrix invertible?

The meaning of eigenanalysis

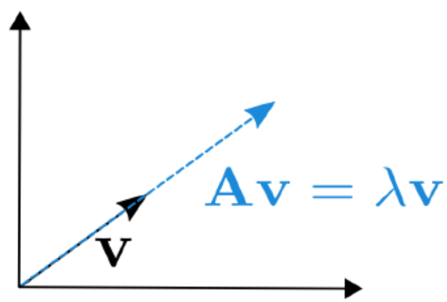
Let $\mathbf{A}$ be an $n \times n$ matrix, where $\mathbf{A}$ is a linear operator on vectors in $\mathbb{R}^n$, such that $\mathbf{A}\,\mathbf{x} = \mathbf{b}$

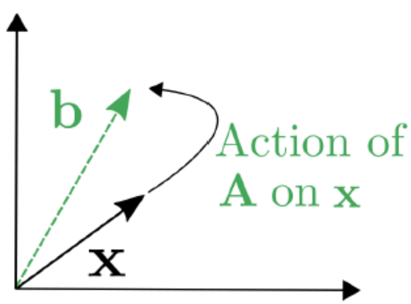




An **eigenvector** of $\mathbf{A}$ is a vector $\mathbf{v} \in \mathbb{R}^n$ such that $\mathbf{A}\,\mathbf{v} = \lambda \mathbf{v}$, where λ is called the corresponding eigenvalue.

Matrix $\mathbf{A}$ only changes the length of $\mathbf{v}$, not its direction!





Equation $\mathbf{A}\,\mathbf{v} = \lambda \mathbf{v}$ **Equation $\mathbf{A}\,\mathbf{x} = \mathbf{b}$.**

Principal Component Analysis (PCA)

- Many signal processing, control and machine learning tasks employ multivariate data which often exhibit dependencies and redundancies.
- For example, it is often useful to reduce the dimensionality of a signal while maintaining the useful information.
- This reduces the computational complexity of any algorithm while preserving the physical meaning of the data.
- Besides dimensionality reduction, we often would like to transform the multi-channel data such each channel is orthogonal to each other (the data covariance matrix is diagonal)
- We use the PCA to accomlish this goal $\rightarrow$ The PCA has been called one of the most valuable results from applied linear algebra.

PCA – derivation

- Consider a general data vector, $\mathbf{x}_k \in \mathbb{C}^{M \times 1}$, with the empirical (sample) covariance matrix defined as

$$\mathrm{cov}(\mathbf{x}_k) \stackrel{\text{def}}{=} \mathbf{R}_\mathbf{x} = \frac{1}{N} \sum_{i=0}^{N-1} \mathbf{x}_k \mathbf{x}_k^H .$$

- Also, if we define a matrix $\mathbf{X} \in \mathbb{C}^{N \times M}$:

$$\mathbf{X}^T = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_N \end{bmatrix} \implies \mathbf{R}_\mathbf{x} = \tfrac{1}{N} \mathbf{X}^H \mathbf{X}$$

- The symmetric covariance matrix $\mathbf{R}_\mathbf{x}$ admits the following eigenvalue decomposition: $\mathbf{Q}^H \mathbf{R}_\mathbf{x} \mathbf{Q} = \mathbf{\Lambda}$

- The diagonal eigenvalue matrix, $\mathbf{\Lambda} = \mathrm{diag}\{\lambda_1, \lambda_2, \dots, \lambda_M\}$, indicates the power of each component of $\mathbf{x}_k$.

- The matrix of eigenvectors, $\mathbf{Q}_r = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_M]$, designates the principal directions of the data.

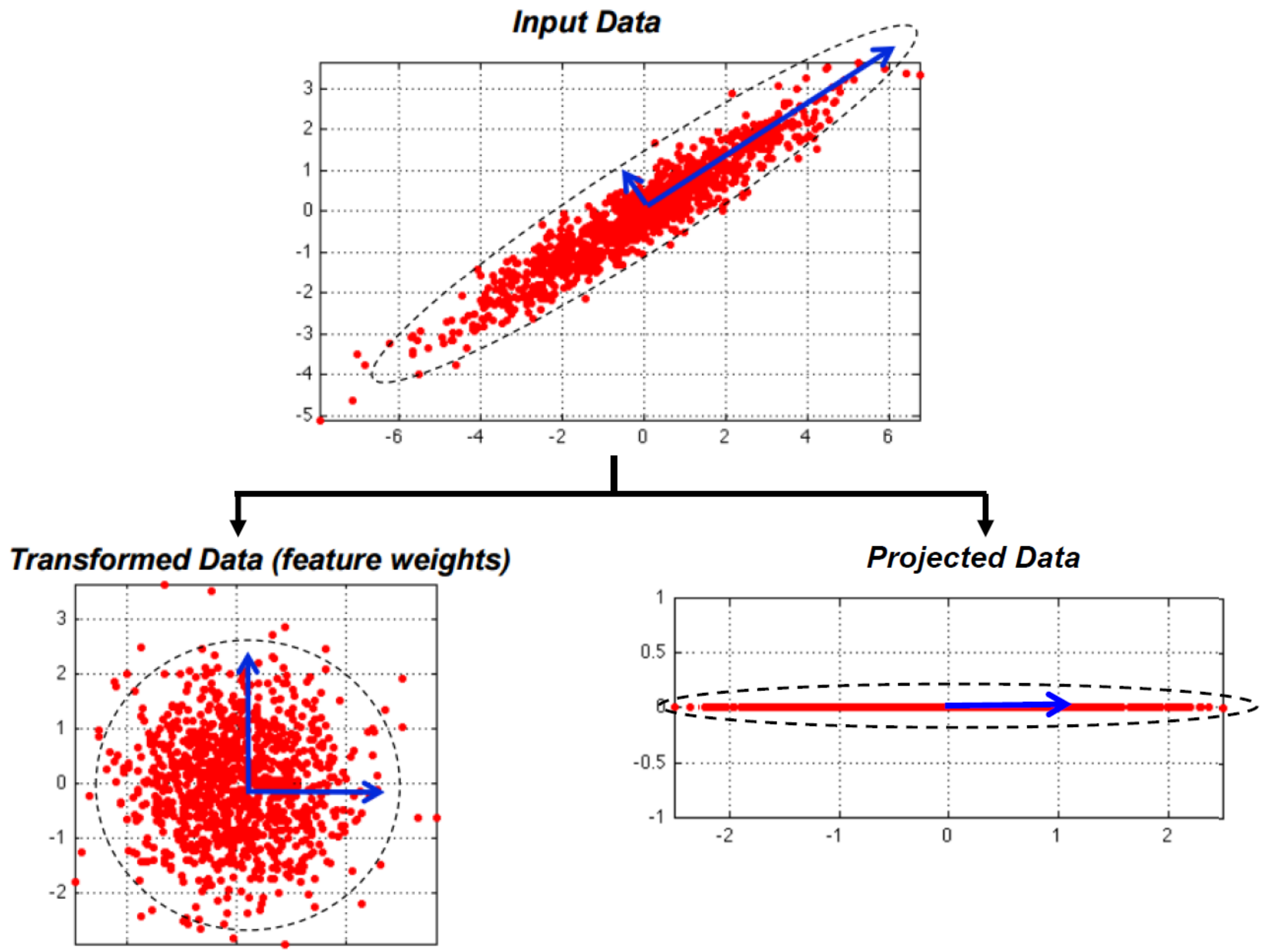
PCA – dimensionality reduction

- PCA selects the **directions in which the data expresses the maximal variance**, that is, the directions of the principal eigenvectors of the data matrix.
- If we choose the r-leading principal components, $r < M$, we can reduce the dimensionality of the data onto the axes which exhibit the r-largest variances.
- To perform dimensionality reduction, the PCA can be applied to obtain the transformed data vector $\mathbf{u}_{r,k} \in \mathbb{C}^r$ with dimensions $r < M$ as

$$\mathbf{u}_{r,k} = \mathbf{W}_r \mathbf{x}_k = \boldsymbol{\Lambda}_{1:r}^{-\frac{1}{2}} \mathbf{Q}_{1:r}^T \mathbf{x}_k$$

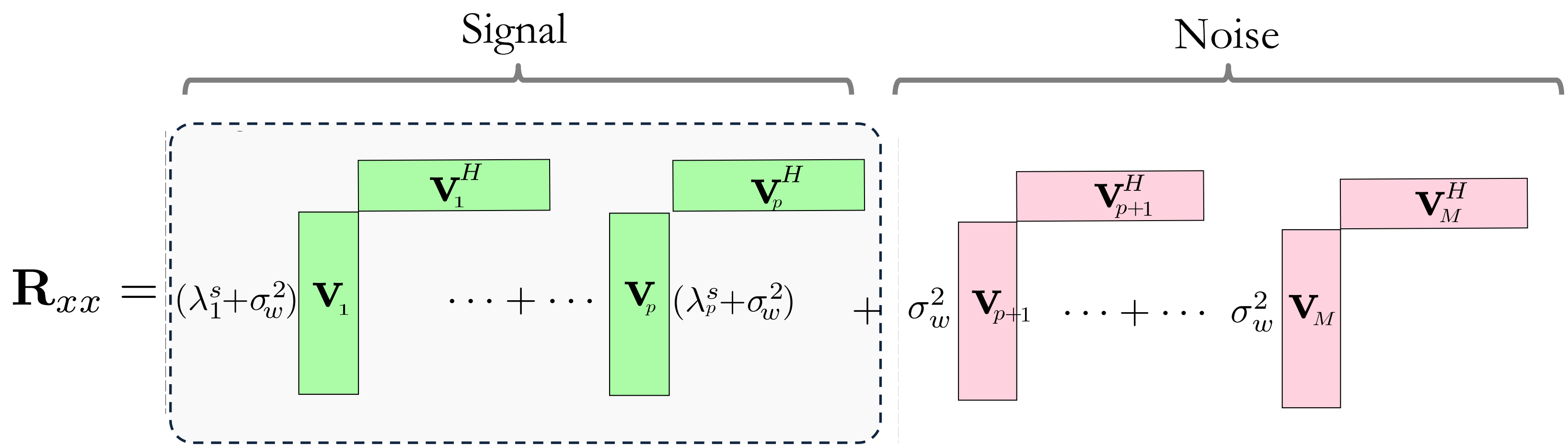
- The PCA matrix $\mathbf{W}_r$ can be interpreted as a projection matrix as we are unable to recover $\mathbf{x}_k$ from the "reduced" data vector $\mathbf{u}_{r,k}$.

PCA – Geometric Interpretation



Principal Components Spectral Estimation

- PCA can be applied to the **autocorrelation matrix** of a univariate time series to find orthogonal directions of maximal variance, also known as **Singular Spectrum Analysis**.
- PCA **can also be used with Blackman–Tukey, maximum entropy method** and **AR spectrum estimation.**



$$\hat{\mathbf{R}}_{xx} \approx \hat{\mathbf{R}}_s = \sum_{i=1}^{p} \lambda_i \mathbf{v}_i \mathbf{v}_i^H$$

Linear Algebra terms: **We impose a rank p constraint** on $\mathbf{R}_{xx}$

Can we de-noise the signal by discarding the noise eigenvectors? $[\mathbf{v}_{p+1}, \ldots, \mathbf{v}_M]$

Subspace Methods for Spectral Estimation

- Eigenanalysis can be applied to the **autocorrelation matrix** of a univariate time series to find orthogonal directions of maximal variance, also known as **Singular Spectrum Analysis (SSA)**.

- Consider $x(n) = A_1 e^{\jmath n\omega_1} + w(n)$

$$\boxed{A_1 = |A_1|e^{\jmath\Phi} \qquad w(n) \sim \mathcal{N}(0, \sigma_w^2)}$$

- On vector notation: $\boxed{\mathbf{x} = A_1\mathbf{e}_1 + \mathbf{w}}$

$$\mathbf{x} = [x(0),\, x(1),\, \ldots,\, x(M-1)]^T$$
$$\mathbf{e}_1 = [1,\, e^{\jmath\omega_1},\, \ldots,\, e^{\jmath\omega_1(M-1)}]^T$$

Autocorrelation: Matrix

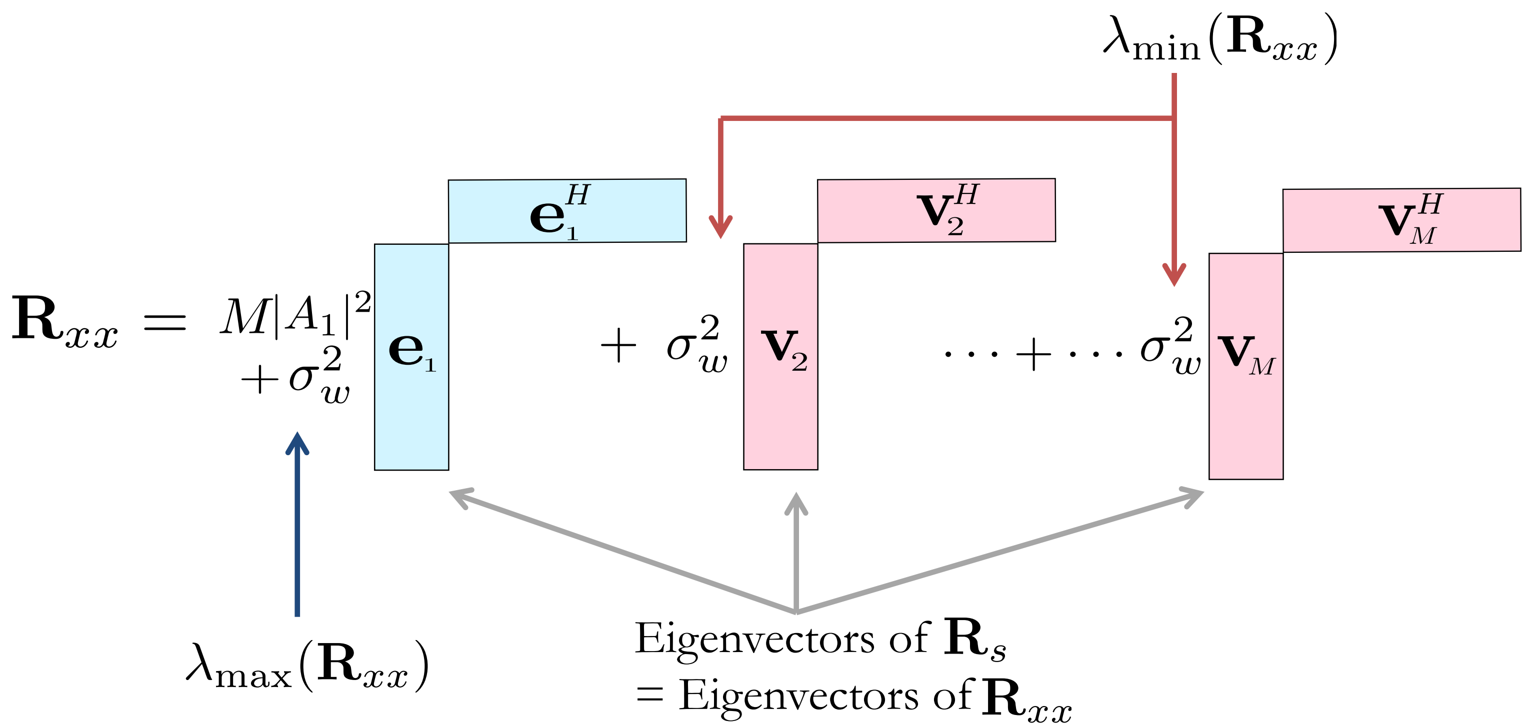
$$E(\mathbf{x}\mathbf{x}^H) = \mathbf{R}_{xx} = \underbrace{|A_1|^2\mathbf{e}_1\mathbf{e}_1^H}_{\mathbf{R}_s} + \underbrace{\sigma_w^2\mathbf{I}}_{\mathbf{R}_n}$$

Signal Autocorrelation
Rank 1
Single non-zero Eigenvalue $= M|A_1|^2$

Noise Autocorrelation
Rank M
All Eigenvalues $= \sigma_w^2$

Decomposing the Autocorrelation Matrix

- $\mathbf{R}_s = |A_1|^2 \mathbf{e}_1 \mathbf{e}_1^H$ is Hermitian.
- Remaining M-1 eigenvectors are orthogonal to $\mathbf{e}_1$ $\rightsquigarrow$ $\mathbf{e}_1^H \mathbf{v}_i = 0, \;\; i = 2, \dots M$



Can we use the idea that $\mathbf{e}_1^H \mathbf{v}_i = 0$,to somehow estimate the power spectrum?

Multiple Sinusoids

Consider the signal $x(n) = A_1 e^{\jmath n\omega_1} + A_2 e^{\jmath n\omega_2} + w(n)$

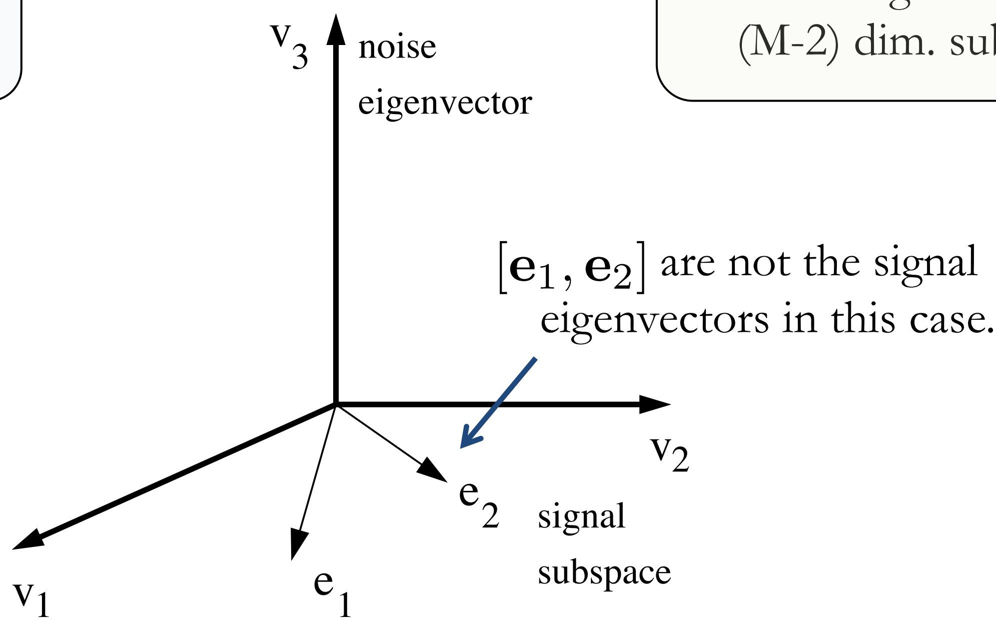
$$\mathbf{R}_{xx} = \mathbf{E}\mathbf{P}\mathbf{E}^H + \sigma_w^2 \mathbf{I} \qquad \mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2]\,,\ \mathbf{P} = \mathsf{diag}(|A_1|^2, |A_2|^2)$$

Rank 2

The first 2 eigenvalues of $\mathbf{R}_{xx}$ are $\lambda_i^s + \sigma_w^2$
The remaining are σ_w^2

- Signal eigenvectors span a 2D subspace
- Noise eigevectos span a (M-2) dim. subspace

The signal and noise subspaces are orthogonal



Subspace Methods

Extending to p sinusoids: $\mathbf{R}_{xx} = \mathbf{E}\mathbf{P}\mathbf{E}^H + \sigma_w^2\mathbf{I}$

$$\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_p]\,,\ \mathbf{P} = \mathrm{diag}(|A_1|^2, \ldots, |A_p|^2)$$

$$\text{Using}\quad \mathbf{e}_i^H\mathbf{v}_k = 0 \quad \begin{cases} i = 1, \ldots, p \\ k = p+1, \ldots, M \end{cases}$$

$\Longrightarrow$ PSD estimation can be performed as: $\hat{P}_{sub}(\omega) = \dfrac{1}{\sum_{i=p+1}^{M} \alpha_i \left|\mathbf{e}^H\mathbf{v}_i\right|^2}$

Pisarenko Harmonic Decomposition

$$\boxed{\hat{P}_{PHD}(\omega) = \frac{1}{\left|\mathbf{e}^H\mathbf{v}_{\min}\right|^2}}$$

MUltiple Signal Classification (MUSIC)

$$\boxed{\hat{P}_{MU}(\omega) = \frac{1}{\sum_{i=p+1}^{M}\left|\mathbf{e}^H\mathbf{v}_i\right|^2}}$$

EigenVector Method

$$\boxed{\hat{P}_{EV}(\omega) = \frac{1}{\sum_{i=p+1}^{M}\frac{1}{\lambda_i}\left|\mathbf{e}^H\mathbf{v}_i\right|^2}}$$

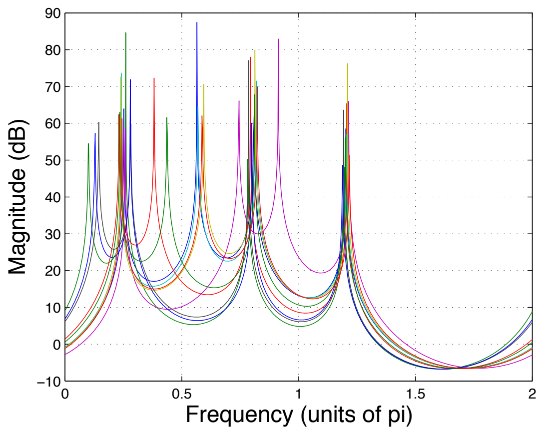
Minimum Norm Method

$$\boxed{\hat{P}_{MN}(\omega) = \frac{1}{\left|\mathbf{e}^H\mathbf{a}\right|^2}}$$

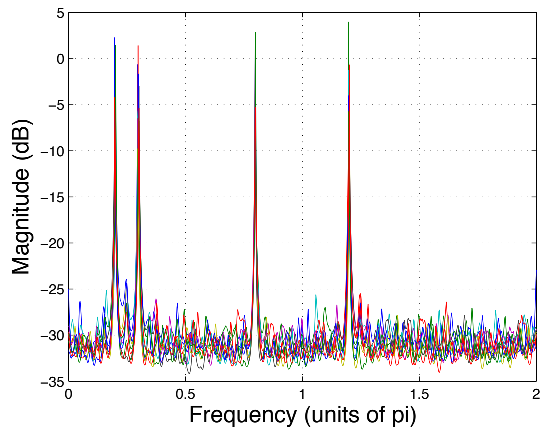
$\mathbf{a}$ $\in$Noise Subspace & has min. norm

Comparison of the 4 Subspace Methods

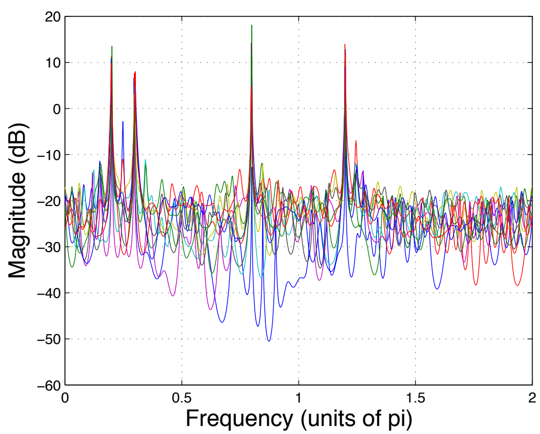
Pisarenko



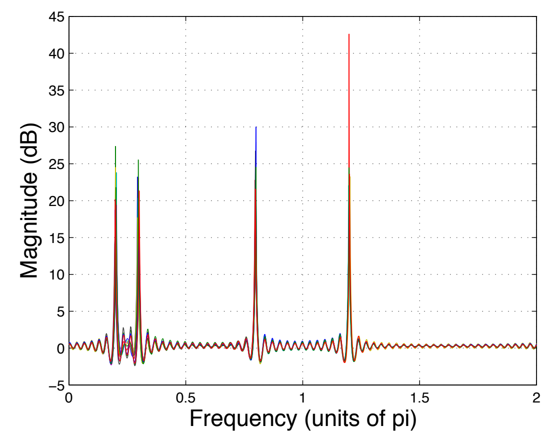
MUSIC



EigenVector



Minimum Norm



Overlay of 10 different realizations of 4 complex sinusoids in white noise.

Pisarenko only needs a 5 x 5 correlation matrix

A 64 x 64 correlation matrix was used for other methods

Except for Pisarenko's method, all other estimates are correct!

Summary of the Different Methods