Adaptive Sig. Proc. & Machine Intel.

Lecture 2 - Augmented Complex Statistics and Widely Linear Modelling

Danilo 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

Outline

Background on:

Linear stochastic models

- Autoregressive (ARE), moving average (AR), ARMA

Part 1: Complex Calculus

- Cauchy-Riemann equations do not work for real functions of complex variables!
- Solution: $\mathbb{CR}$-Calculus and gradients of cost functions (these are typically real valued)

Part 2: Complex Statistics

- Moving from real to complex data
- Key point 1: Circularity and widely linear estimation
- Covariance and pseudocovariance
- Key point 2: Widely linear autoregressive model $\looparrowright$ caters for both second order circular (proper) and non-circular (improper) data
- Practical examples

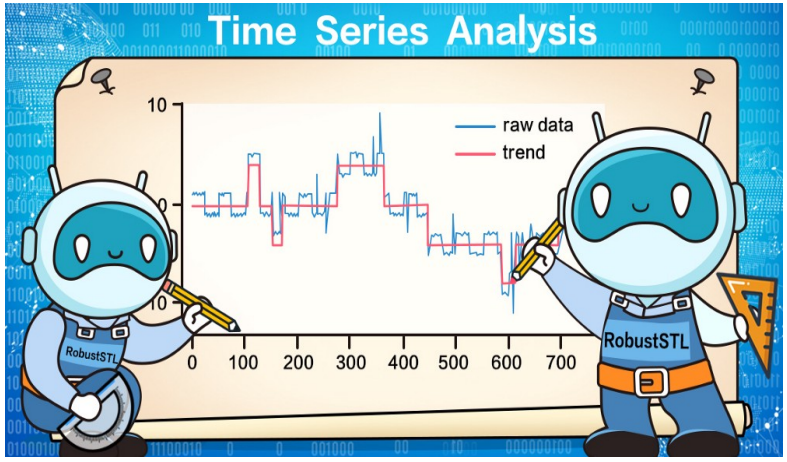
Background ↬ Linear stochastic models

Q: Have you ever considered what the following tasks

- Forecasting of financial data
- Supply-demand modelling (e.g. electricity or air-ticket pricing)
- Modelling of COVID-19 spread
- Weather forecasting and modelling in astronomy (e.g. sunspots)

have in common?

A: These are time series of which the signal generating mechanisms are largely unknown or untractable. We need to make sense from such data based on historical observations only – subject of **Time Series Analysis.**



Justification $\not\rightarrow$ Wold decomposition theorem (Existence theorem, also mentioned in your coursework)

Wold's decomposition theorem plays a central role in time series analysis, and explicitly proves that any covariance–stationary time series can be decomposed into two different parts: **deterministic** (such as a sinewave) and **stochastic** (filtered WGN).

Therefore, a general process can be written a sum of two processes

$$x[n] = x_p[n] + x_r[n] = x_p[n] + \sum_{j=1}^{q} b_j w[n-j] \qquad w \rightsquigarrow \text{white process}$$

$\Rightarrow x_r[n] \quad \not\rightarrow \quad$ regular random process

$\Rightarrow x_p[n] \quad \not\rightarrow \quad$ predictable process, with $x_r[n] \quad \perp x_p[n]$,

$$E\{x_r[m]x_p[n]\} = 0$$

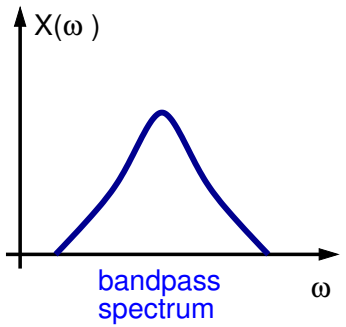
$\rightsquigarrow$ we can treat **separately** the **predictable** part (e.g. a deterministic sinusoidal signal) and the **random** signal.

Our focus will be on the modelling of the random component

NB: Recall the difference between shift–invariance and time–invariance

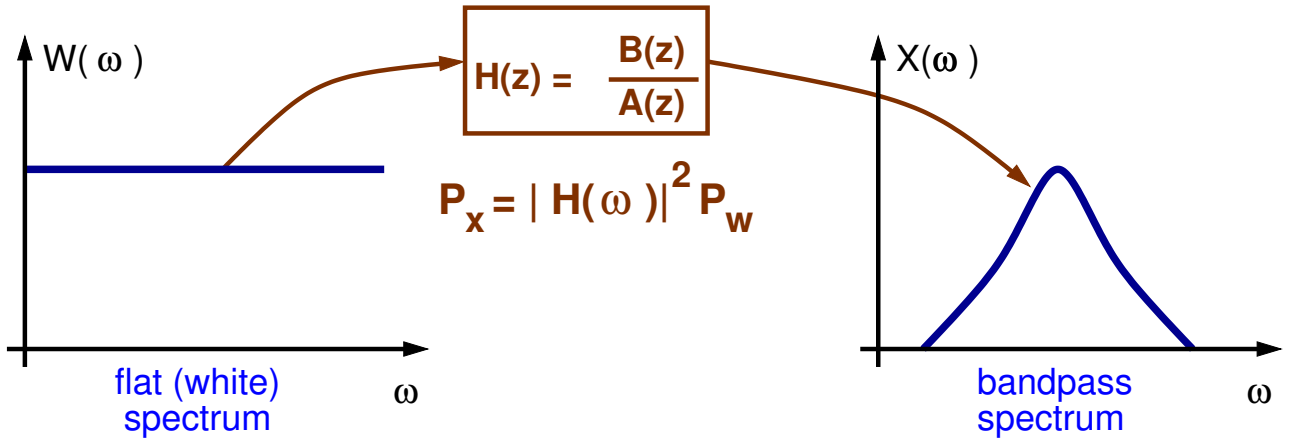
How do we model a real world signal?

Suppose the measured real world signal has e.g. a bandpass (any other) power spectrum



We desire to describe the whole long signal with very few parameters

1. Can we model first and second statistics of real world signal by shaping up the white noise spectrum using some transfer function?
2. Does this produce the same second order properties (mean, variance, ACF, spectrum) for any white noise input?



Can we use this linear stochastic model for prediction?

Towards linear stochastic processes

Wold's theorem implies that any purely non-deterministic covariance–stationary process can be arbitrarily well approximated by an ARMA process

Therefore, the general form for the power spectrum of a WSS process is

$$P_x(e^{\jmath\omega}) = \sum_{k=1}^{N} \alpha_k \delta(\omega - \omega_k) + P_{x_r}(e^{\jmath\omega})$$

We are interested in processes generated by **filtering white noise with a linear shift–invariant filter** that has a rational system function.

This class of digital filters includes the following system functions:

- Autoregressive (AR) $\rightarrow$ all pole system $\rightarrow \quad H(z) = 1/A(z)$
- Moving Average (MA) $\rightarrow$ all zero system $\rightarrow \quad H(z) = B(z)$
- Autoregressive Moving Average (ARMA) $\rightarrow$ poles and zeros $\rightarrow H(z) = B(z)/A(z)$

Definition: A covariance-stationary process $x[n]$ is called **(linearly) deterministic** if $p(x[n] \mid x[n-1], x[n-2], \ldots) = x[n]$.

$\rightarrow$ A stationary deterministic process, $x_p[n]$, can be predicted correctly (with zero error) using the entire past, $x_p[n-1], x_p[n-2], x[n-3], \ldots$

Spectrum of ARMA models (look also at Recap slides)

recall that two conjugate complex poles of $A(z)$ give one peak in the spectrum

$ACF \equiv PSD$ **in terms of the information available**

In ARMA modelling we filter white noise $w[n]$ **(so called driving input)** with a causal linear shift–invariant filter with the transfer function $H(z)$, a rational system function with p poles and q zeros given by

$$X(z) = H(z)W(z) \quad \looparrowright \quad H(z) = \frac{B_q(z)}{A_p(z)} = \frac{\sum_{k=0}^{q} b_k z^{-k}}{1 + \sum_{k=1}^{p} a_k z^{-k}}$$

For a stable $H(z)$, the **ARMA(p,q) stochastic process** $x[n]$ will be wide–sense stationary. For the **driving noise** power $P_w = \sigma_w^2$, the power of the stochastic process $x[n]$ is **(recall** $P_y = |H(z)|^2 P_x = H(z)H^*(z)P_x$**)**

$$P_x(z) = \sigma_w^2 \frac{B_q(z)B_q(z^{-1})}{A_p(z)A_p(z^{-1})} \quad \Rightarrow \quad P_z(e^{\jmath\theta}) = \sigma_w^2 \frac{\left|B_q(e^{\jmath\theta})\right|^2}{\left|A_p(e^{\jmath\theta})\right|^2} = \sigma_w^2 \frac{\left|B_q(\omega)\right|^2}{\left|A_p(\omega)\right|^2}$$

Notice that "$(\cdot)^*$" in analogue frequency corresponds to "z^{-1}" in "digital freq."

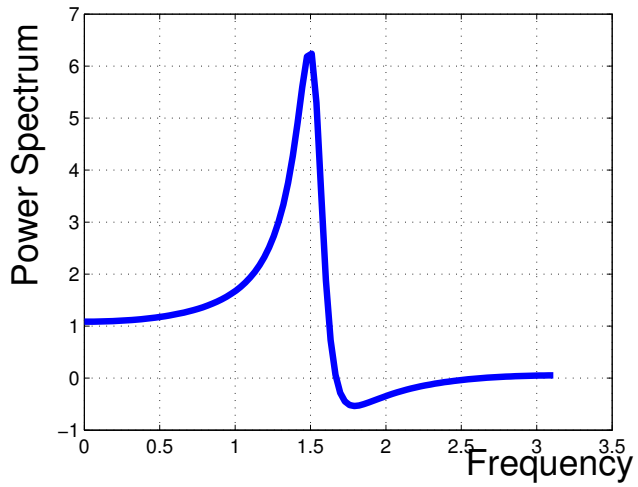
Example: Can the shape of power spectrum tell us about the order of the polynomials $B(z)$ and $A(z)$?

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)

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

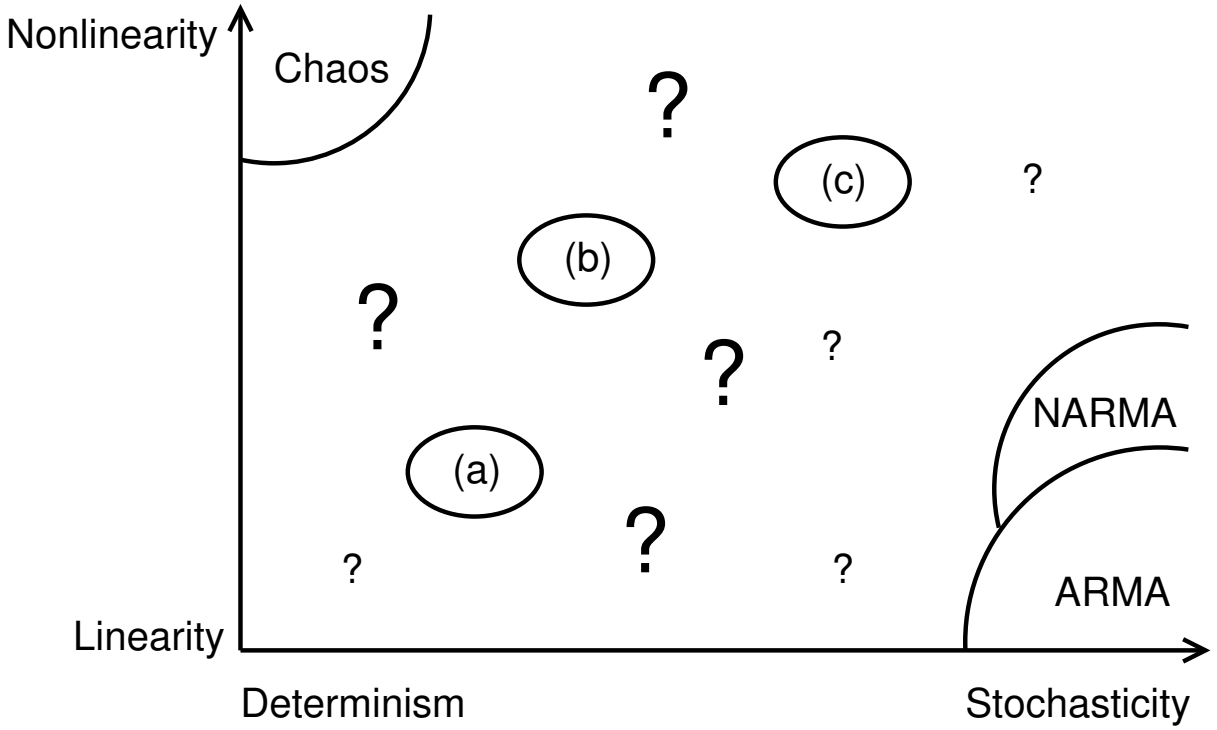


How can we categorise real–world measurements?

where would you place a DC level in WGN, $x[n] = A + w[n], \quad w \sim \mathcal{N}(0, \sigma_w^2)$

(a) Noisy oscillations, (b) Nonlinearity and noisy oscillations, (c) Random nonlinear process

(? left) Route to chaos, (? top) stochastic chaos, (? middle) mixture of sources



Our lecture is about ARMA models (linear stochastic)

How about observing the signal through a nonlinear sensor?

Difference equations $\nrightarrow$ the ACF follows the data model!

(for convenience, a slight abuse in notation from A(z) to the autoregressive part)

Since $X(z) = H(z)W(z)$, the random processes $x[n]$ and $w[n]$ are related by a linear difference equation with constant coefficients, given by

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

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

Autoregressive processes (pole–only)

A general $AR(p)$ process (autoregressive of order p) is given by

$$x[n] = a_1 x[n-1] + \cdots + a_p x[n-p] + w[n] = \sum_{i=1}^{p} a_i x[n-i] + w[n] = \mathbf{a}^T \mathbf{x}[n] + w[n]$$

Observe the auto–regression in $\{x[n]\} \looparrowright$ the past of x is used to generate the future

Duality between the AR and MA processes:

For example, the first order autoregressive process, AR(1),

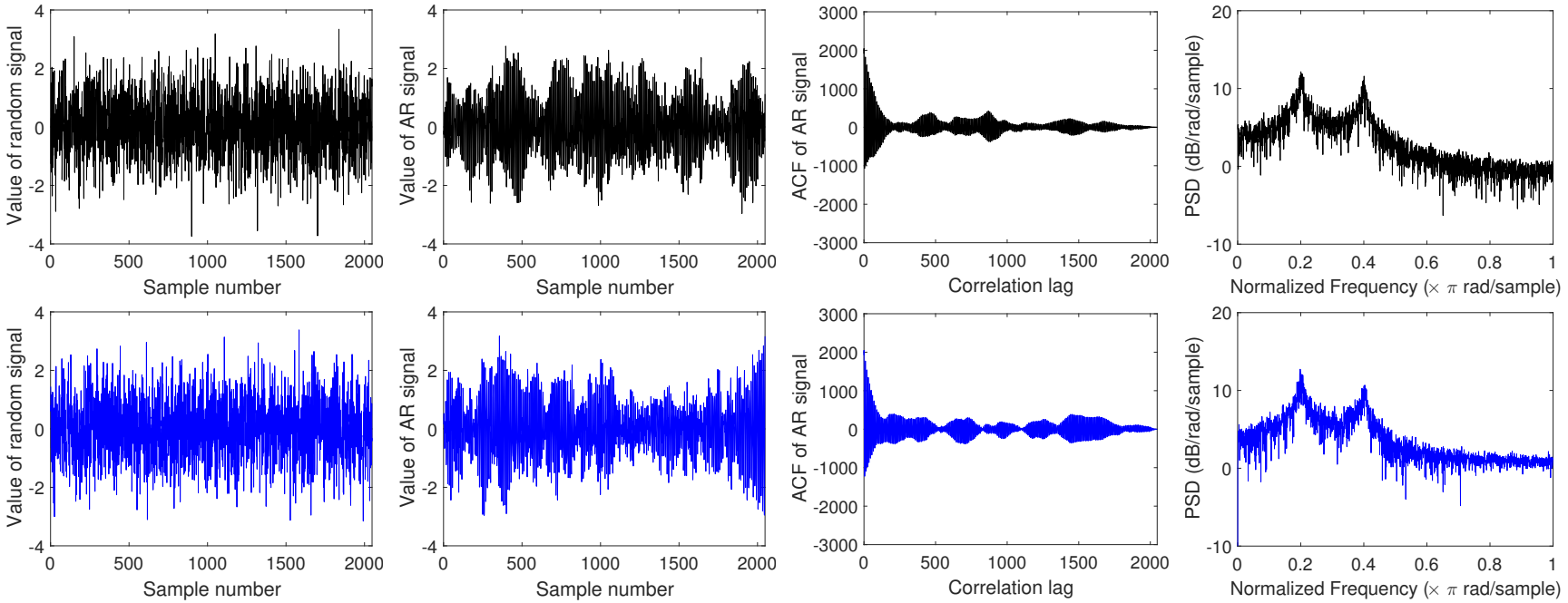
$$x[n] = a_1 x[n-1] + w[n] \quad \Leftrightarrow \quad \sum_{j=0}^{\infty} b_j w[n-j]$$

has an MA representation, too (right hand side above).

This follows from the duality between IIR and FIR filters.

Example: Statistical properties of AR processes

Drive the AR(4) model from Example 6 with two different WGN realisations $\sim \mathcal{N}(0,1)$

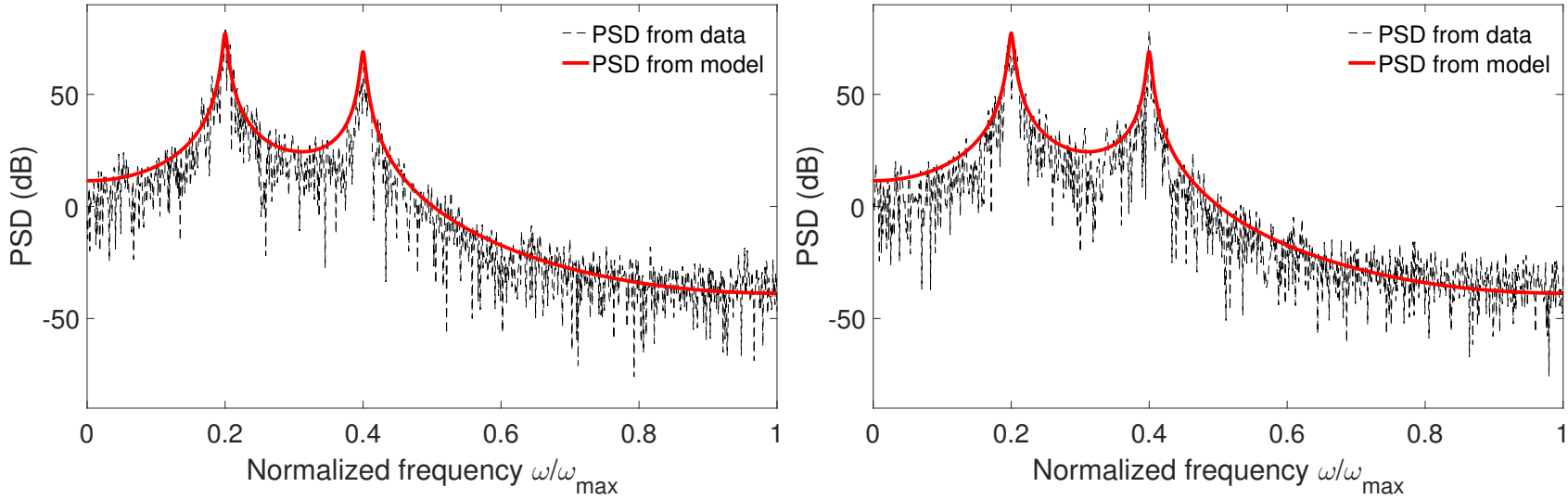



```
r = wgn(2048,1,1);
a = [2.2137, -2.9403, 2.1697, -0.9606];
a = [1 -a];
x = filter(1,a,r);
xacf = xcorr(x);
xpsd = abs(fftshift(fft(xacf)));
```

- The time domain random AR(4) processes look different
- The ACFs and PSDs are exactly the same (2nd-order stats)!
- **This signifies the importance of our statistical approach**

Example: Advantages of model-based analysis

Consider the PSD's for different realisations of the AR(4) process from Example 5



- The different realisations of the AR(4) process (based on different driving WGNs) lead to different Empirical PSD's (in thin black)
- The theoretical PSD from the model is consistent regardless of the data (in thick red)

```
N = 1024;
w = wgn(N,1,1);
a = [2.2137, -2.9403, 2.1697, -0.9606];      % Coefficients of AR(4) process
a = [1 -a];
x = filter(1,a,w);
xacf = xcorr(x);                             % Autocorrelation of AR(4) process
dft = fft(xacf);
EmpPSD = abs(dft/length(dft)).^ 2;           % Empirical PSD obtained from data
ThePSD = abs(freqz(1,a,N,1)).^ 2 ;           % Theoretical PSD obtained from model
```

Variance and spectrum of AR processes

Variance:

For $k = 0$, the contribution from the term $E\{x[n-k]w[n]\}$ is σ_w^2, and

$$r_{xx}(0) = a_1 r_{xx}(-1) + a_2 r_{xx}(-2) + \cdots + a_p r_{xx}(-p) + \sigma_w^2$$

Divide by $r_{xx}(0) = \sigma_x^2$ to obtain

$$\sigma_x^2 = \frac{\sigma_w^2}{1 - \rho_1 a_1 - \rho_2 a_2 - \cdots - \rho_p a_p}$$

Power spectrum: (recall that $P_{xx} = |H(z)|^2 P_{ww} = H(z)H^*(z)P_{ww}$, the expression for the output power of a linear system $\rightarrow$ see Appendix)

$$P_{xx}(f) = \frac{2\sigma_w^2}{|1 - a_1 e^{-\jmath 2\pi f} - \cdots - a_p e^{-\jmath 2\pi p f}|^2} \qquad 0 \leq f \leq 1/2$$

Fro more detail: *"Spectrum of Linear Systems"* from Lecture 1: Background

Key: Finding AR coefficients $\hookrightarrow$ the Yule–Walker eqns

(there are several similar forms – we follow the most concise one)

For $k = 1, 2, \ldots, p$ from the general AR(p) autocorrelation function, we obtain the 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**, $a_1, \ldots, a_p$, or $\mathbf{a} = [a_1, \ldots, a_p]^T$.

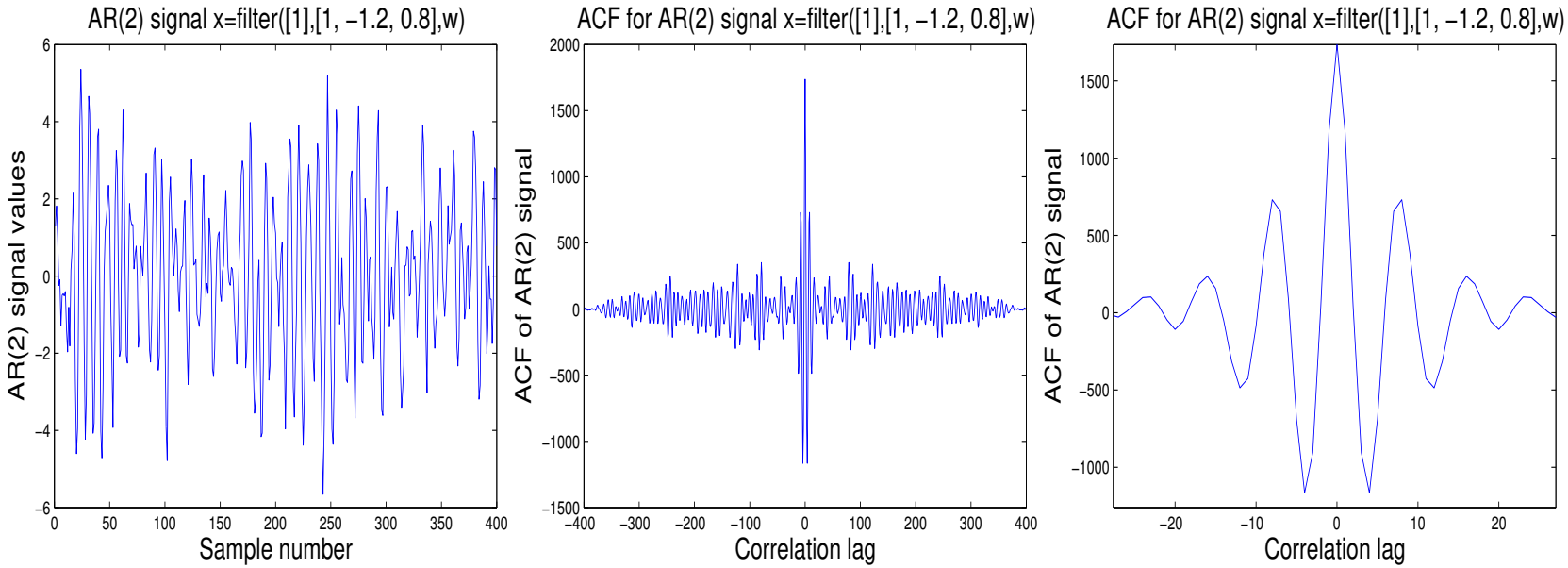
The above equations can be expressed in a compact vector–matrix form as

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

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

Example: Find the parameters of an AR(2) process, $x(n)$, generated by $x[n] = 1.2x[n-1] - 0.8x[n-2] + w[n]$

Coursework: comment on the shape of the ACF for large lags



Matlab: `for i=1:6; [a,e]=aryule(x,i); display(a);end`

$\mathbf{a}^{(1)} = [0.6689]$ $\mathbf{a}^{(2)} = [1.2046, -0.8008]$

$\mathbf{a}^{(3)} = [1.1759, -0.7576, -0.0358]$

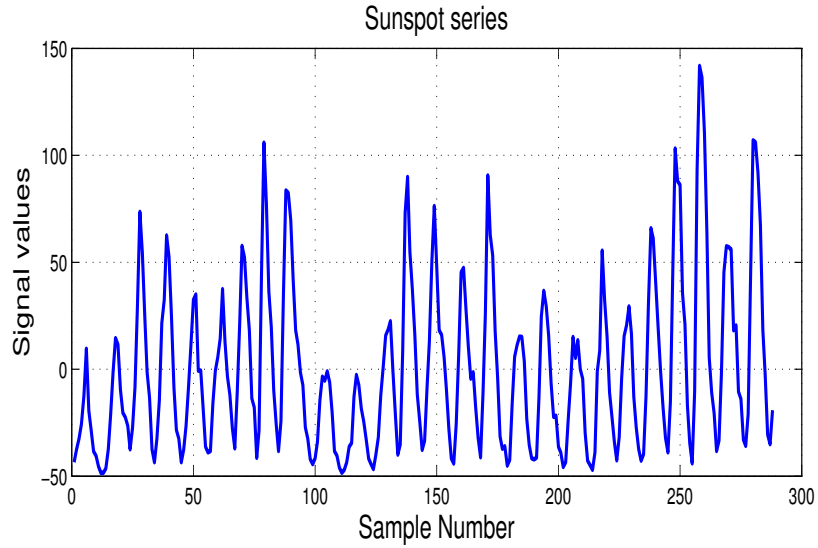
$\mathbf{a}^{(4)} = [1.1762, -0.7513, -0.0456, 0.0083]$

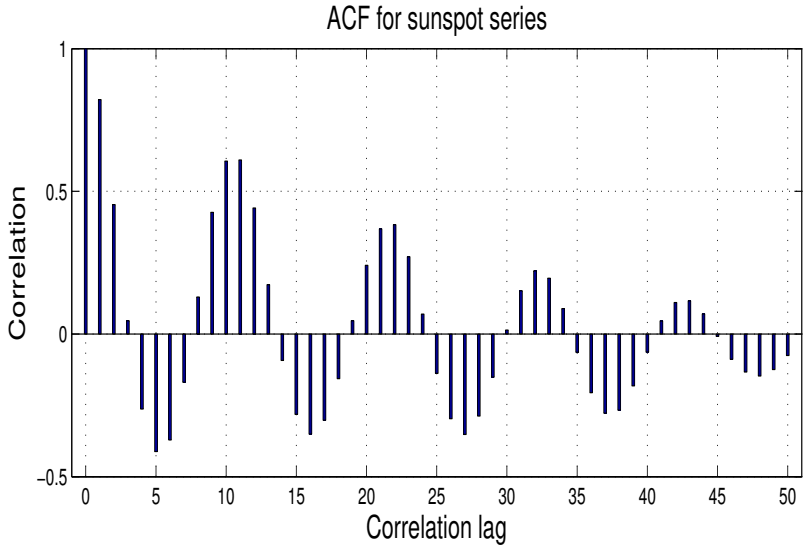
$\mathbf{a}^{(5)} = [1.1763, -0.7520, -0.0562, 0.0248, -0.0140]$

$\mathbf{a}^{(6)} = [1.1762, -0.7518, -0.0565, 0.0198, -0.0062, -0.0067]$

Example: Work by Yule $\looparrowright$ model of sunspot numbers

Recorded for $>$ 300 years. To study them in 1927 Yule invented the $AR(2)$ model





We first center the data, as we do not wish to model the DC offset (determinisic component), but the stochastic component (AR model driven by white noise)!

Using the Y-W equations we obtain:

$\mathbf{a}_1 = [0.9295]$

$\mathbf{a}_2 = [1.4740, -0.5857]$

$\mathbf{a}_3 = [1.5492, -0.7750, 0.1284]$

$\mathbf{a}_4$=[1.5167,-0.5788,-0.2638,0.2532]

$\mathbf{a}_5$=[1.4773,-0.5377,-0.1739, 0.0174,0.1555]

$\mathbf{a}_6$=[1.4373, -0.5422, -0.1291, 0.1558, -0.2248, 0.2574]

Partial autocorrelation function: Motivation

Notice: ACF of $AR(p)$ infinite in duration, **but** can by be described in terms of p nonzero functions ACFs.

Denote by a_{kj} the jth coefficient in an autoregressive representation of order k, so that a_{kk} is the last coefficient. Then

$$\rho_j = a_{kj}\rho_{j-1} + \cdots + a_{k(k-1)}\rho_{j-k+1} + a_{kk}\rho_{j-k} \qquad j = 1, 2, \ldots, k$$

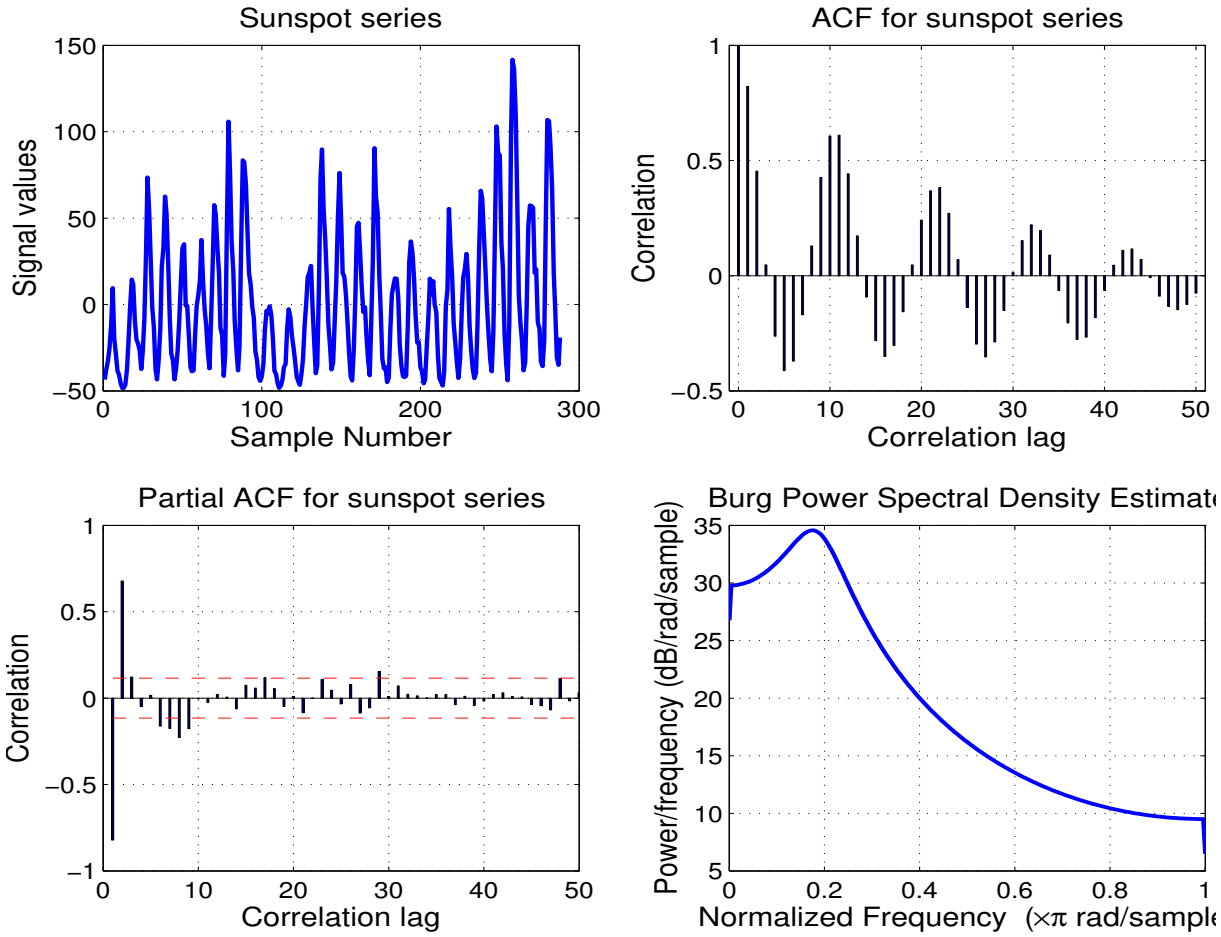
leading to the Yule–Walker equations, which can be written as

$$\begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{k-1} \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{k-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_{k1} \\ a_{k2} \\ \vdots \\ a_{kk} \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_k \end{bmatrix}$$

The only difference from the standard Y-W equations is the use of the symbols a_{ki} to denote the AR coefficient $a_i \not\mapsto k$ indicating the model order

Example (contd.): Model order for sunspot numbers

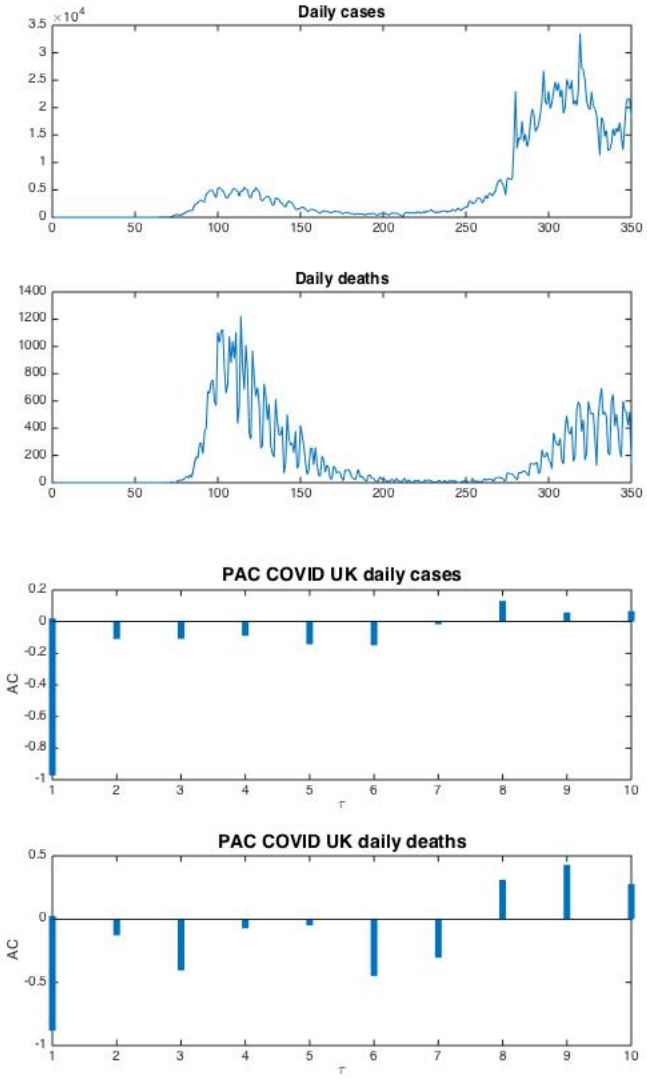
After $k = 2$ the partial correlation function (PAC) is very small, indicating $p = 2$



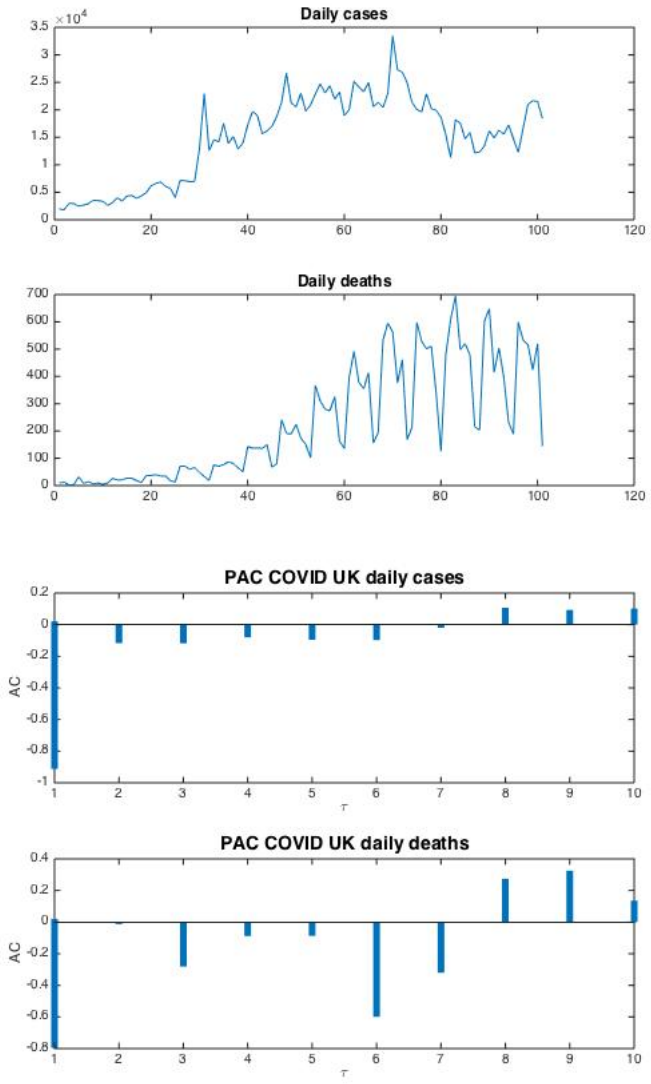
The broken red lines denote the 95% confidence interval which has the value $\pm 1.96/\sqrt{N}$, and where $PAC \approx 0$

Example: ARMA(p,q) modelling of COVID-19 data?

COVID-19 time series in the UK



Second wave, UK COVID-19



AR model based prediction: Importance of model order

For a zero mean process $x[n]$, the best **linear predictor**, in the **mean square error** sense, of $x[n]$ based on $x[n-1], x[n-2], \ldots$ is

$$\hat{x}[n] = a_{k-1,1}x[n-1] + a_{k-1,2}x[n-2] + \cdots + a_{k-1,k-1}x[n-k+1]$$

(apply the $E\{\cdot\}$ operator to the general $AR(p)$ model expression, and recall that $E\{w[n]\} = 0$)

(Hint:
$E\{x[n]\} = \hat{x}[n] = E\left\{a_{k-1,1}x[n-1] + \cdots + a_{k-1,k-1}x[n-k+1] + w[n]\right\} = a_{k-1,1}x[n-1] + \cdots + a_{k-1,k-1}x[n-k+1])$)

whether the process is an AR or not

In MATLAB, check the function:

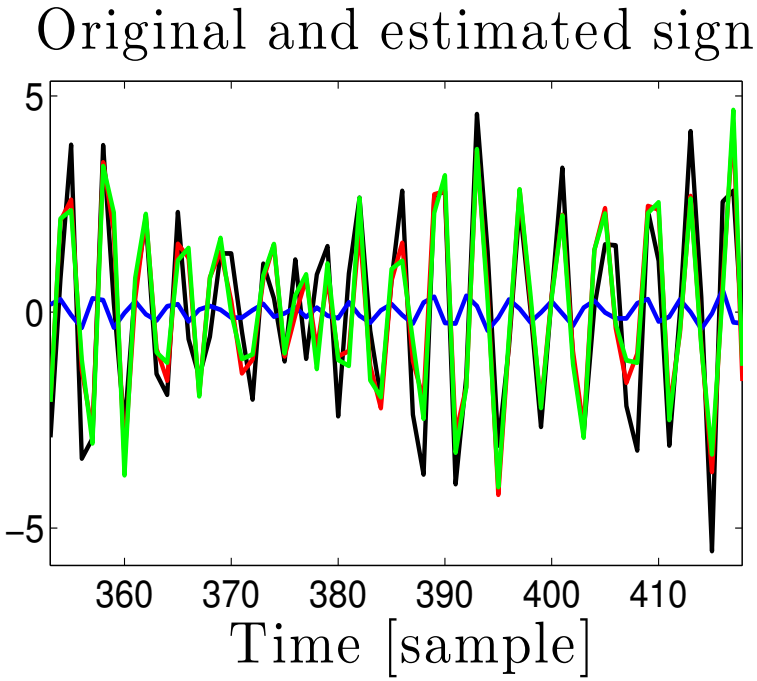
`ARYULE`

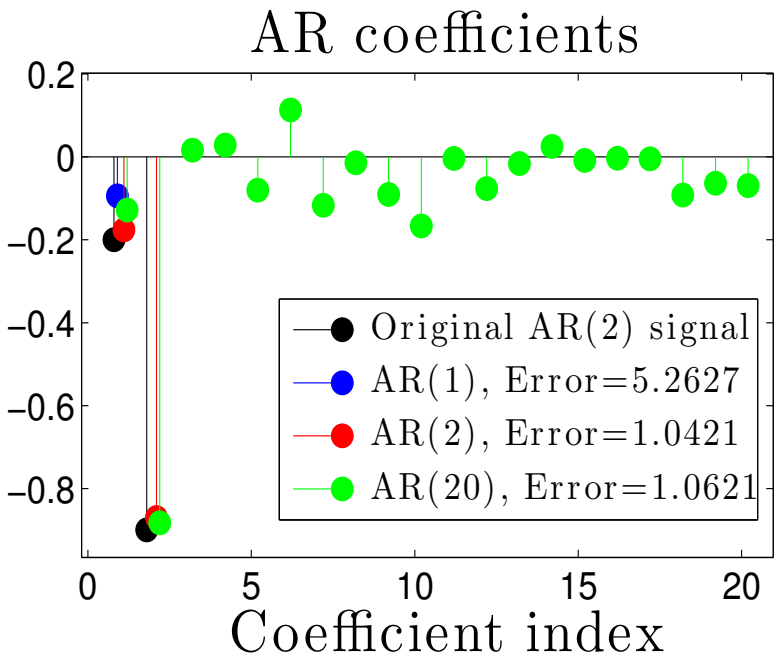
and functions

`PYULEAR, ARMCOV, ARBURG, ARCOV, LPC, PRONY`

Example: Under– vs Over–fitting a model ↬ Estimation of the parameters of an AR(2) process

Original AR(2) process $x[n] = -0.2x[n-1] - 0.9x[n-2] + w[n]$, $w[n] \sim \mathcal{N}(0,1)$, estimated using AR(1), AR(2) and AR(20) models:





The *higher order* coefficients of the AR(20) model are close to zero and therefore do not contribute significantly to the estimate, while the AR(1) does not have sufficient degrees of freedom. (see also Appendix 3)

Model order selection $\looparrowright$ practical issues

In practice: the greater the model order the greater accuracy & complexity

Q: When do we stop? What is the optimal model order?

Solution: To establish a trade–off between computational complexity and model accuracy, we introduce "penalty" for a high model order. Such criteria for model order selection are:

MDL: The minimum description length criterion (MDL) (by Rissanen),

AIC: The Akaike information criterion (AIC)

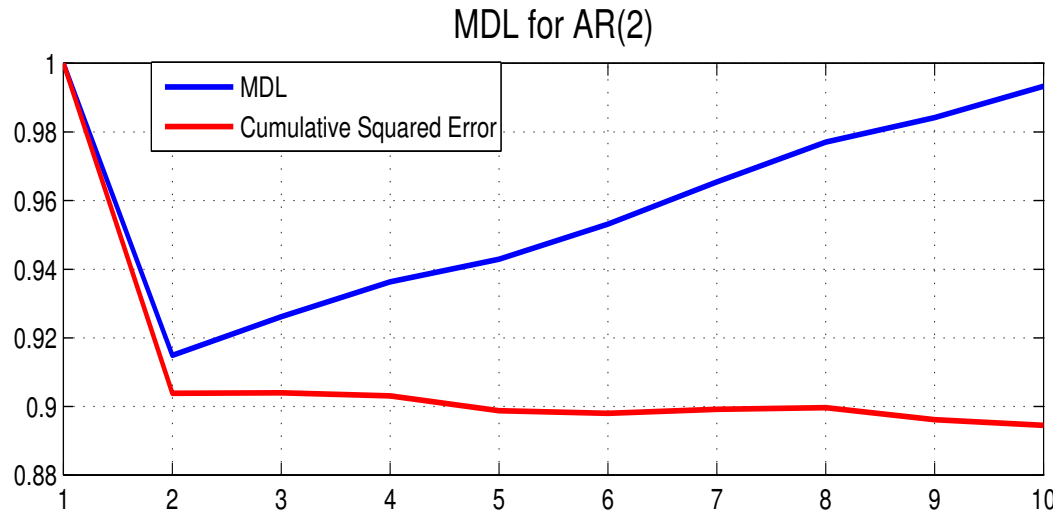
$$\textbf{MDL} \quad p_{opt} = \min_{p} \left[log(E) + \frac{p * log(N)}{N} \right]$$

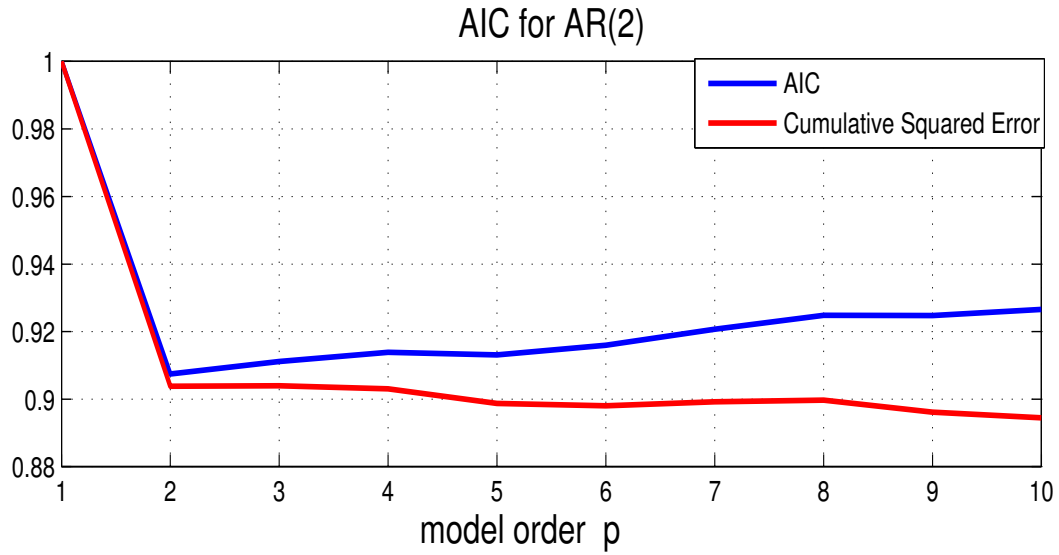
$$\textbf{AIC} \quad p_{opt} = \min_{p} \left[log(E) + 2p/N \right]$$

$E \rightsquigarrow$ the loss function (typically cumulative squared error),
$p \rightsquigarrow$ the number of estimated parameters (model order),
$N \rightsquigarrow$ the number of available data points.

Example: Model order selection $\looparrowright$ MDL vs AIC

MDL and AIC criteria for an AR(2) model with $a_1 = 0.5 \qquad a_2 = -0.3$



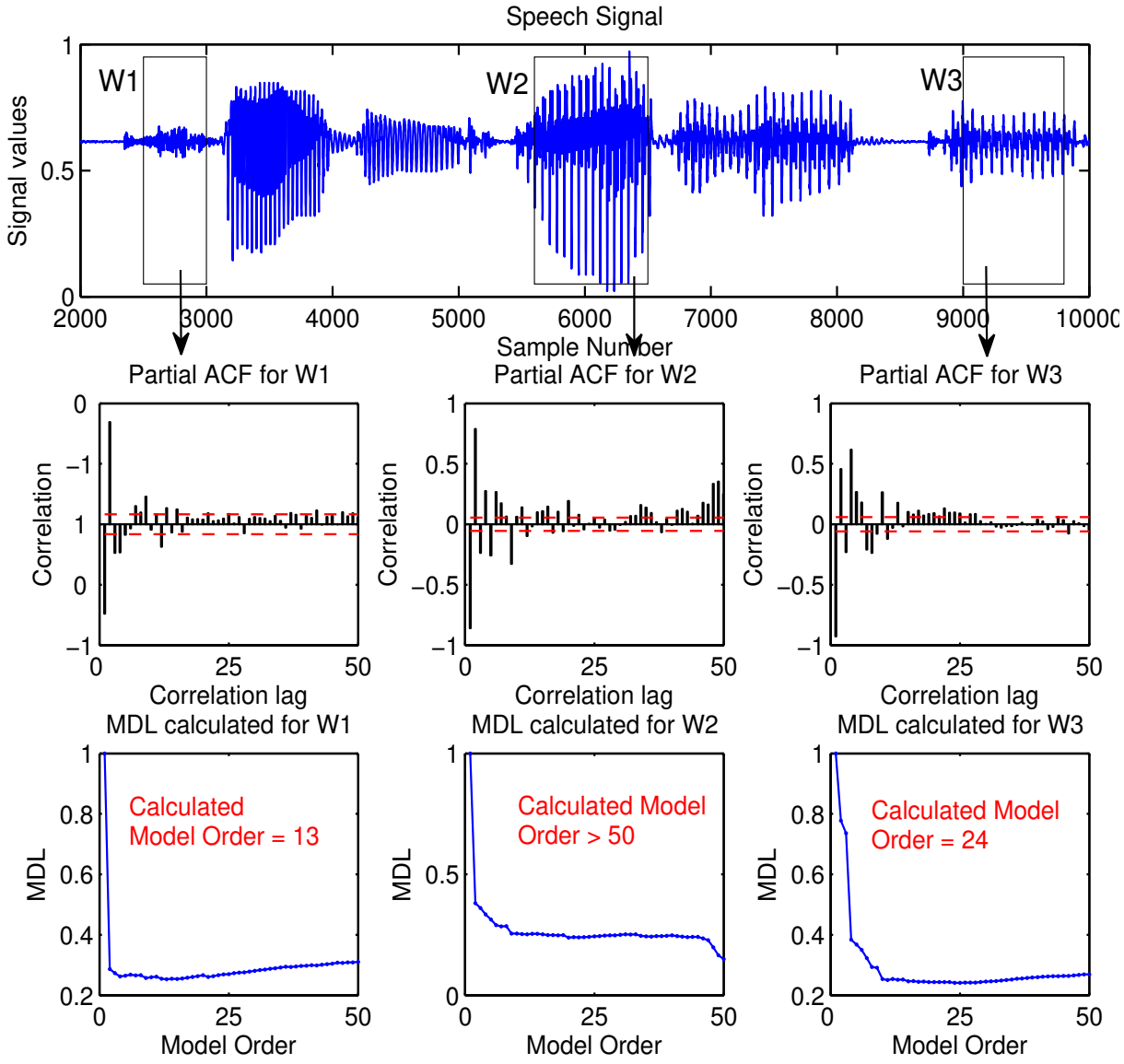


The graphs on the left show the **(prediction error)^2** (vertical axis) versus the **model order** p (horizontal axis). Notice that $p_{opt} = 2$.

The curves are **convex**, i.e. a monotonically decreasing **error^2** with an increasing **penalty term** (MDL or AIC correction).

Hence, we have a unique minimum at $p = 2$, reflecting the correct model order (no over-modelling)

Analysis of nonstationary signals



- Consider a real–world speech signal, and thee different segments with different statistical properties
- **Different AR model orders required for different segments of speech ↬ opportunity for content analysis!**
- To deal with nonstationarity we need short sliding data windows

Part 1: Motivation for complex-domain Data Analytics

Motivation for modelling in $\mathbb{C}$

Much more convenient in a number of applications

- Magnetic source imaging (fMRI, MRI, MEG) are recorded in the Fourier domain, that is, the data are inherently complex-valued
- Interferometric radar - high coherence in order to obtain both the altitude and amplitude introduces speckles
- Array signal processing, antennas, direction of arrival (DoA)
- Transform domain signal processing (DCT, DFT, wavelet)
- Mobile communications (equalisation, I/Q mismatch, nonlinearities)
- Complex-valued neural networks $\not\mapsto$ image processing
- Optics and seismics - reflection, refraction $\not\mapsto$ phase information
- Mixture models in ML, elliptical models, characteristic functions
- Much work still to be done – **great opportunity for future research!**

Fundamental theorem of algebra (FTA)

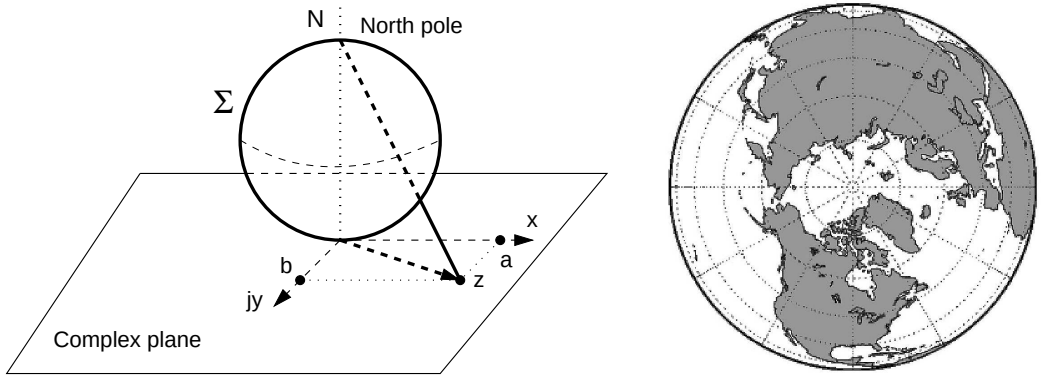
- Initial work by Albert Girard in 1629

'there are n roots to an n-th order polynomial'

He also introduced the abbreviations $\sin, \cos, \tan$ in 1626.

- Descartes in the 1630s **'For every equation of degree n we can imagine roots which do not correspond to any real quantity'**
- In 1749 Euler proved the FTA

Every n-th order polynomial in $\mathbb{R}$ has exactly n roots in $\mathbb{C}$



(a) Riemann sphere

(b) Earth projection from South pole

Stereographic projection and Riemann sphere

- Cauchy → **'conjugate'**, Hankel → **'direction'**, Weierstrass → **'absolute value'**

History of mathematical notation

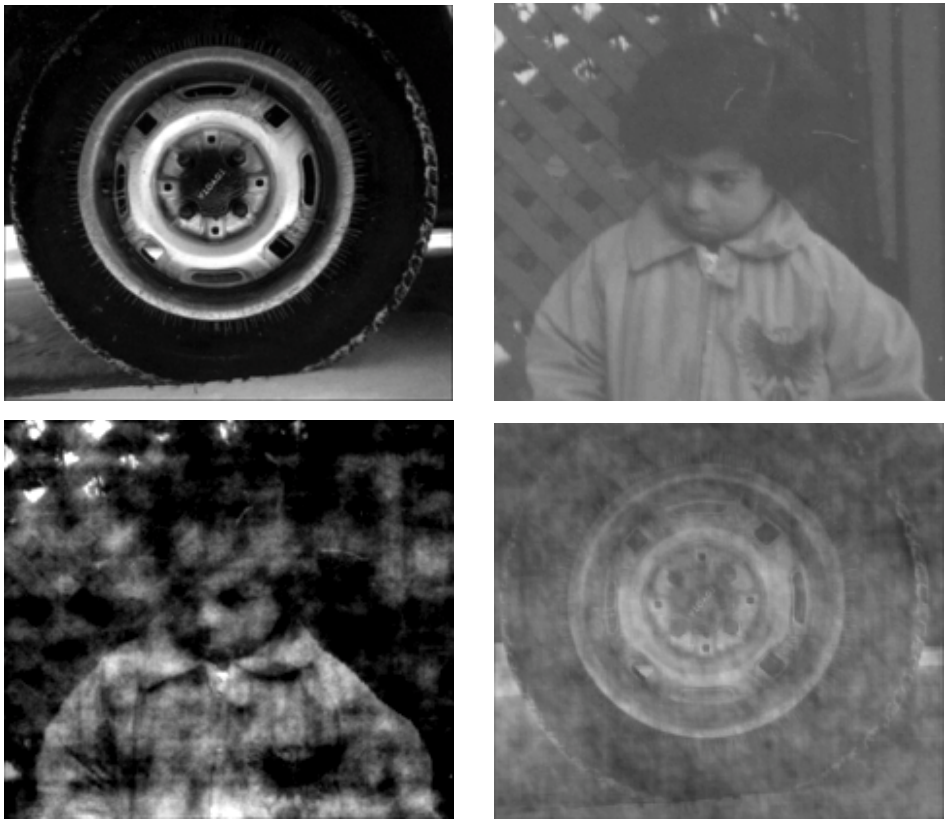
Did you know?

- $9th$ century Al Kwarizimi's *Algebra* - solutions descriptive rather than in form of equations
- $16th$ century - G. Cardano *Ars Magna* - unknowns denoted by single roman letters
- Descartes (1630-s) established general rules
  - lowercase italic letters at the beginning of the alphabet for unknown constants a, b, c, d
  - lowercase italic letters at the end of the alphabet for unknown variables x, y, z
- $\sqrt{-1} = \imath$ – Gauss 1830s, boldface letters for vectors $\mathbf{x}, \mathbf{v}$ - Oliver Heaviside
- Hence $ax^2 + by + cz = 0$

More detail: F. Cajori, *History of Mathematical Notations*, 1929

Example 1: Human visual system

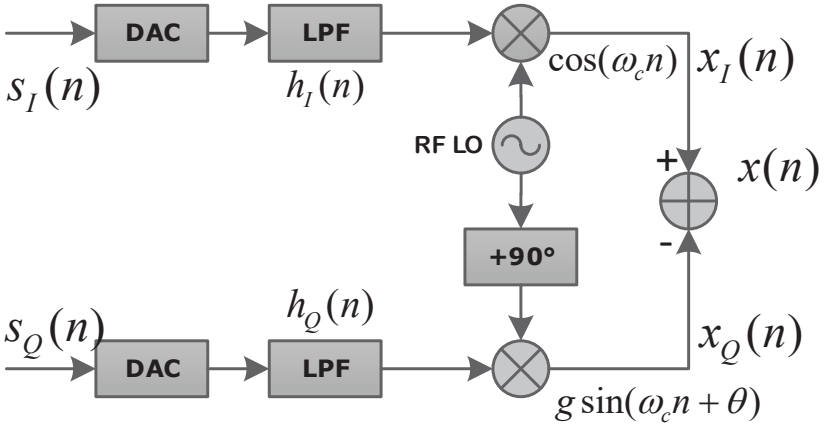
Importance of phase information

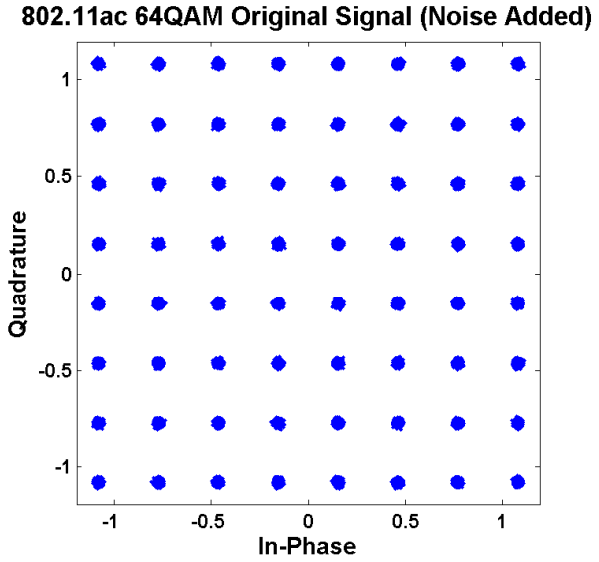


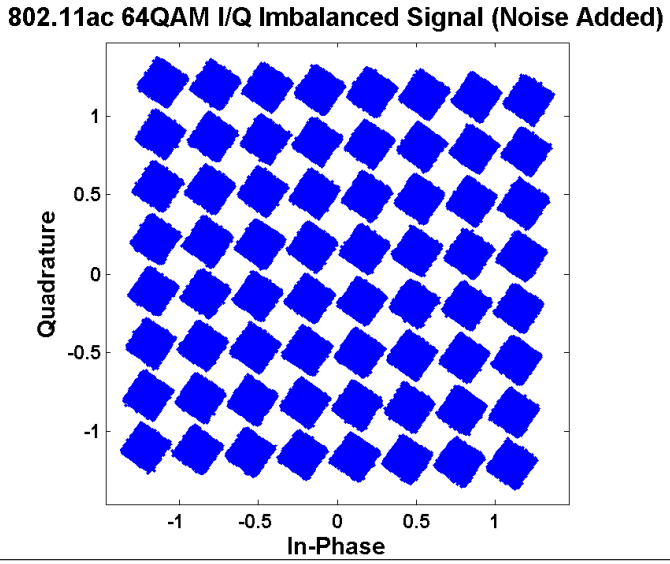
Surrogate images. *Top:* Original images I_1 and I_2; *Bottom:* Images $\hat{I}_1$ and $\hat{I}_2$ generated by exchanging the amplitude and phase spectra of the original images.

Example 2: Noncircularity arising from I/Q imbalance

One of the key issues in the future 5G networks







Consider the baseband discrete-time input signal, $s(n)$, which is complex circular, e.g., 64-QAM. After passing through an I/Q imbalanced modulator, the output $x(n)$ becomes noncircular, that is (WL model)

$$x(n) = \mu(n) * s(n) + \nu(n) * s^*(n)$$

where

$$\mu(n) = 1/2[h_I(n) + gh_Q(n)e^{-j\theta}]$$

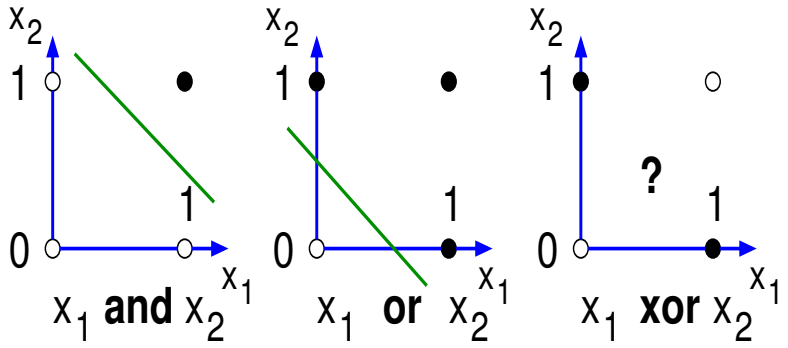
$$\nu(n) = 1/2[h_I(n) - gh_Q(n)e^{-j\theta}]$$

Modern complex estimation: Numerous opportunities

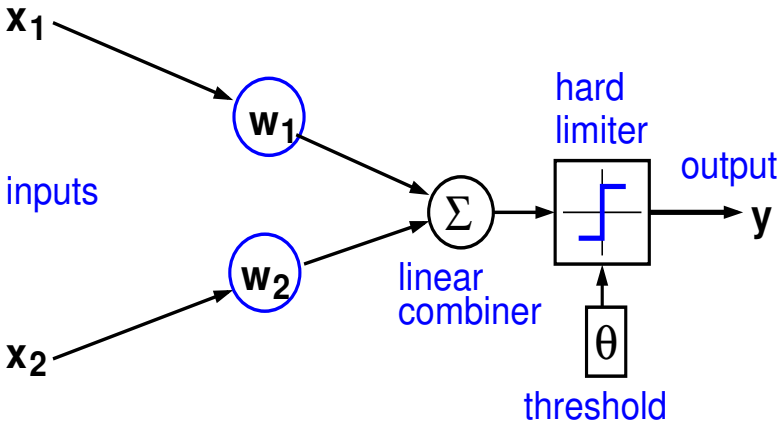
- Complex signals by design (communications, analytic signals, equivalent baseband represenation to eliminate spectral redundancy)
- By convenience of representation (radar, sonar, wind field), direction of arrival related problems
- **Problem:** More powerful algebra than $\mathbb{R}^2$ but no ordering (operator "$\leq$" makes no sense!) and the notion of pdf has to be induced from $\mathbb{R}^2$
- **Problem:** Special form of nonlinearity (the only continuously differentiable function in $\mathbb{C}$ is a constant (Liouville theorem)
- **Solution:** Special 'augmented' statistics – (started in maths in 1992) – more degrees of freedom and physically meaningful matrix structures
- We can differentiate between several kinds of noises (doubly white circular with various distributions $n_r \perp n_i$ & $\sigma^2_{n_r} = \sigma^2_{n_i}$, doubly white noncircular $n_r \perp n_i$ & $\sigma^2_{n_r} > \sigma^2_{n_i}$, noncircular noise)

Jumping ahead a little (Recall the human–monkey scatter plot)

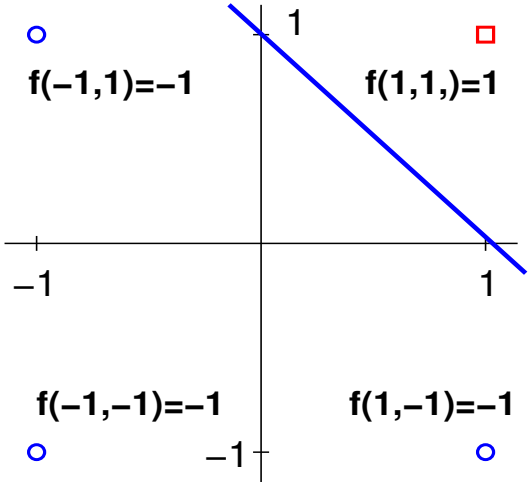
(Non–)Linear separability and learning machines



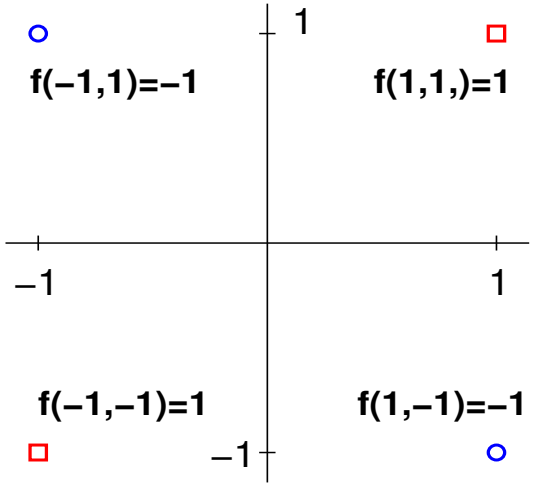
Logic functions and linear separability



Perceptron–type neural network



AND function is linearly separable



How about the XOR

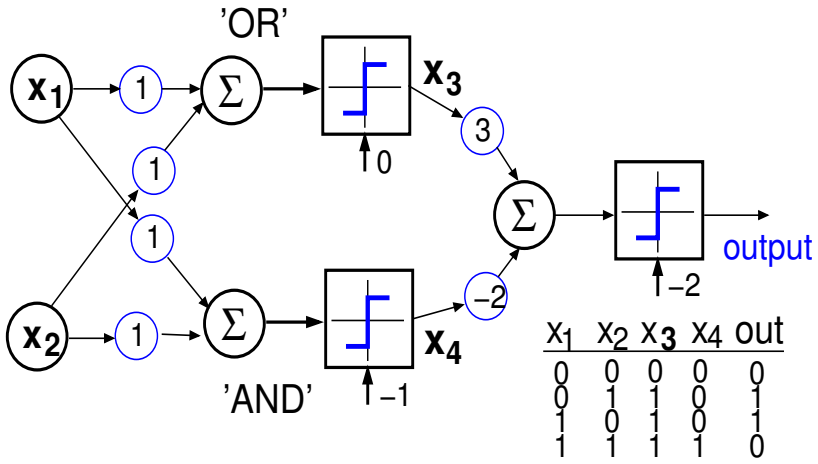
Usefulness of complex numbers in machine intelligence

Example: A single complex neuron is worth as much as a real–valued MLP

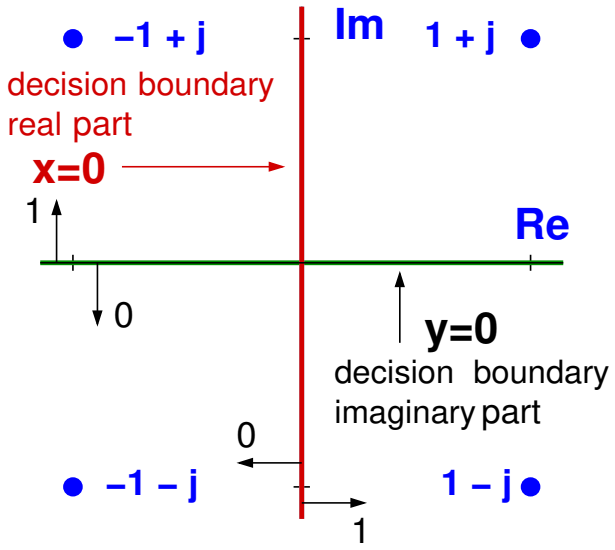
x_1	x_2	z	$P(z) =$ XOR
1	1	$1+\jmath$	1
1	-1	$1-\jmath$	-1
-1	1	$-1+\jmath$	-1
-1	-1	$-1-\jmath$	1

$$P(z) = \begin{cases} 1, & \arg(z) \text{ in 1st or 3rd quadrants} \\ -1, & \arg(z) \text{ in 2nd or 4th quadrants.} \end{cases}$$

In $\mathbb{R}$ we need multiple layers, the so called **multilayer perceptron** shown below



But a single neuron in $\mathbb{C}$ performs the XOR and is amplitude independent



Part 2: Complex Calculus

We will now introduce a modern perspective on complex calculus, the so-called **CR calculus** which offers much more flexibility in the differentiation of complex functions, and is indispensable in learning systems where the objective (cost) functions are typically real-valued functions of complex variables.

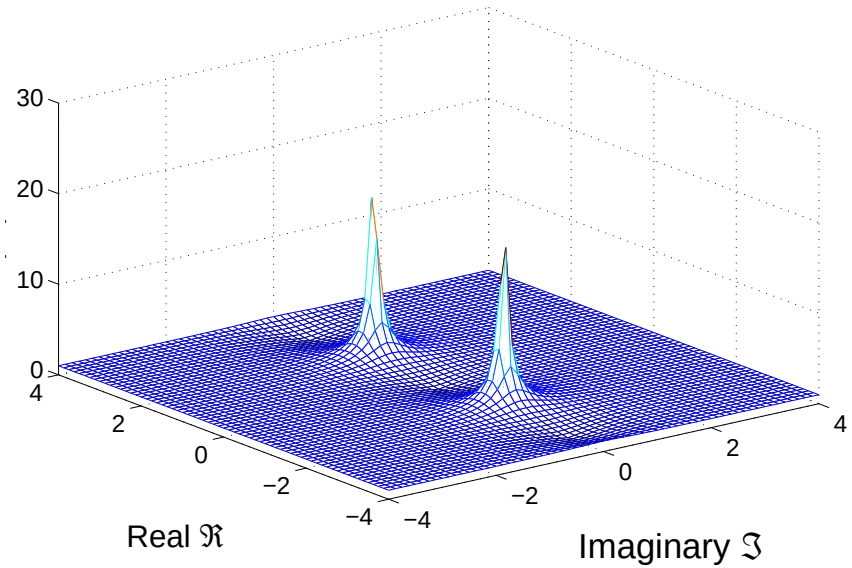
Such functions are not differentiable using the standard complex differentiation (Cauchy-Riemann), yet gradient based learning schemes require such derivatives.

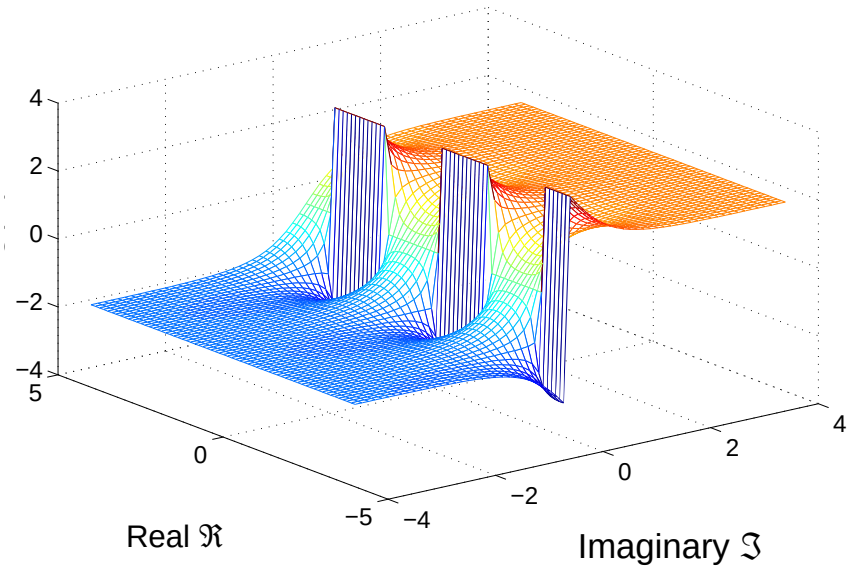
We show that the CR-calculus applies both to the holomorphic (complex analytic) and non-holomorphic functions of complex variable, and will elucidate the use of the so-called 'pseudo-gradient'.

A simple example of complex neural networks

Observe the difference between complex-valued and real activation functions

Consider the magnitude and phase for the function $f(z) = \tanh(\cdot)$





Singularities: Isolated singularities (removable singularities, poles, essential singularities), branch points, singularities at ∞.

In gradient based learning, we seek a coefficient vector $\mathbf{w}$ using the so called **pseudo-gradient** of the cost function $J = E\{|e|^2\} = E\{ee^*\}$,

$$\nabla_{\mathbf{w}} J(e, e^*) = \frac{\partial J}{\partial \mathbf{w}_r} + \jmath \frac{\partial J}{\partial \mathbf{w}_i} \qquad \text{but where does this come from}$$

Cauchy–Riemann derivatives are very restrictive!

Recall: $f(z) = u(x,y) + jv(x,y) \rightarrow f'(z) = \partial u(x,y)/\partial x + j\partial v(x,y)/\partial x$

$$\frac{\partial u(x,y)}{\partial x} = \frac{\partial v(x,y)}{\partial y}, \qquad \frac{\partial v(x,y)}{\partial x} = -\frac{\partial u(x,y)}{\partial y}$$

Intuition: The Jacobian matrix of $f(z) = u + jv$, is given by

$$\mathbf{J} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{bmatrix} \quad \Leftrightarrow \quad \begin{bmatrix} '1' & '1' \\ '-1' & '1' \end{bmatrix}$$

$\rightarrow$ **Thus, e.g. $f(z) = z^*$ is not analytic**, as its Jacobian $\mathbf{J} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$.

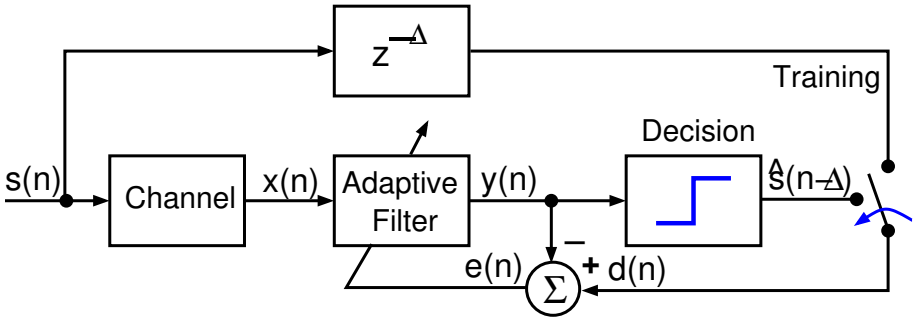
Functions which depend on both $z = x + jy$ and $z^* = x - jy$ are not analytic, for example

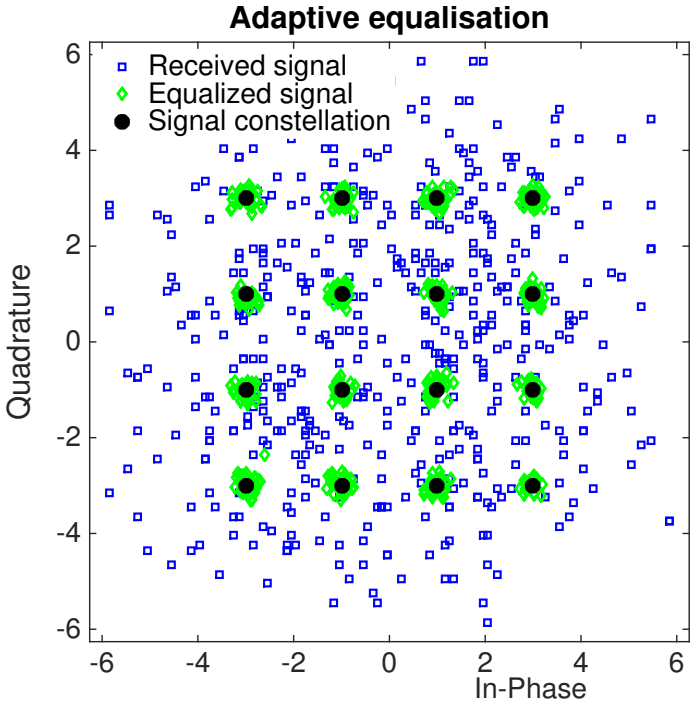
$$J(z, z^*) = zz^* = x^2 + y^2 \quad \Rightarrow \quad \mathbf{J} = \begin{bmatrix} 2x & 2y \\ 0 & 0 \end{bmatrix} \quad \Leftrightarrow \quad \frac{\partial u}{\partial x} \neq \frac{\partial v}{\partial y} \quad \frac{\partial v}{\partial x} \neq -\frac{\partial u}{\partial y}$$

Another typical example is the cost function $J = \frac{1}{2}e(k)e^*(k) = \frac{1}{2}|e(k)|^2$

Example 3: Complex-valued optimisation (how does it work?)

Channel equalisation in wireless comms (reversing distortion of transmitted signals)





Objective: To minimize the cost function

$$J(\mathbf{w}) = |e(k)|^2 = e(k)e^*(k)$$
$$= e_r^2 + e_i^2 = u(k) + jv(k)$$

Here, $u(k) = e_r^2 + e_i^2$ and $v(k) = 0$

But, $\frac{\partial u}{\partial e_r} = 2e_r \neq \frac{\partial v}{\partial e_i} = 0$, $\frac{\partial u}{\partial e_i} = 2e_i \neq -\frac{\partial v}{\partial e_r} = 0$

↬ **Our $J(\mathbf{w})$ is not differentiable in the Cauchy–Riemann sense** (here, error power is a real function of complex variables $e(k)$ and $e^*(k)$) **but the equaliser has to work!**

Isomorphism between $\mathbb{C}$ and $\mathbb{R}^2$

Moving from real-valued to complex-valued data

$$z \rightarrow z^a \quad \leftrightarrow \quad \begin{bmatrix} z \\ z^* \end{bmatrix} = \begin{bmatrix} 1 & \jmath \\ 1 & -\jmath \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

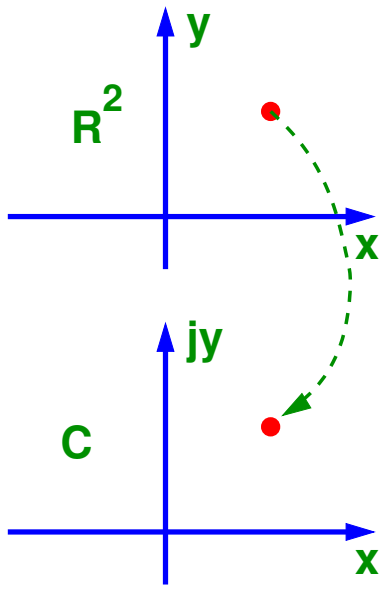
whereas in the case of complex–valued signals, we have

$$\mathbf{z} \rightarrow \mathbf{z}^a \quad \leftrightarrow \quad \begin{bmatrix} \mathbf{z} \\ \mathbf{z}^* \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \jmath\mathbf{I} \\ \mathbf{I} & -\jmath\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$$

For convenience, the "augmented" complex vector $\mathbf{v} \in \mathbb{C}^{2N\times 1}$ can be introduced as

$$\begin{aligned} \mathbf{v} &= [z_1, z_1^*, \ldots, z_N, z_N^*]^T \\ \mathbf{v} &= \mathbf{A}\mathbf{w}, \qquad \mathbf{w} = [x_1, y_1, \ldots, x_N, y_N]^T \end{aligned}$$

where matrix $\mathbf{A} = diag(\mathbf{J}, \ldots, \mathbf{J}) \in \mathbb{C}^{2N\times 2N}$ is block diagonal and transforms the **composite** real vector $\mathbf{w}$ into the **augmented complex vector** $\mathbf{v}$.



The key: The complex-real, or in short $\mathbb{CR}$-derivatives

So, the pseudogradient = the $\mathbb{R}^*$-derivative!

Goal: Find the derivative of a complex function $f(z)$ w.r.t. $z = x + \jmath y$.

In standard Multivariate Calculus in $\mathbb{R}^{N\times 1}$ the derivative of a function $g(\mathbf{x})$, $\mathbf{x} = [x_1, x_2, \ldots, x_N]$ is defined as $\frac{\partial g}{\partial \mathbf{x}} = \left[\frac{\partial g}{\partial x_1}, \ldots, \frac{\partial g}{\partial x_N}\right]^T$

- **Step 1:** Define the vector $\mathbf{x} = [x, \jmath y]^T$, hence $z = \mathbf{1}^T\mathbf{x}$.
- **Step 2:** Express the derivative of f with respect to "real" vector $\mathbf{x}$ i.e $\frac{\partial f}{\partial \mathbf{x}} = \left[\frac{\partial f}{\partial x} \quad \frac{\partial f}{\jmath \partial y}\right]^T$ (see the Appendix 3 for vector-valued derivatives)
- **Step 3:** Transform the derivative vector in Step 2 back into $\mathbb{C}$

$$\frac{\partial f}{\partial z} = \mathbf{1}^T \frac{\partial f}{\partial \mathbf{x}} = \frac{\partial f}{\partial x} + \frac{\partial f}{\jmath \partial y} = \frac{\partial f}{\partial x} - \jmath \frac{\partial f}{\partial y}$$

- **Step 4:** Normalise the derivative since f is "differentiated twice", to give the $\mathbb{R}-$derivatives (*cf.* differentiate wrt z^* for $\mathbb{R}^*-$ derivatives)

$\mathbb{R}-\text{der}: \frac{\partial f}{\partial z} = \frac{1}{2}\left[\frac{\partial f}{\partial x} - \jmath \frac{\partial f}{\partial y}\right]$. Similarly, $\mathbb{R}^*-\text{der}: \frac{\partial f}{\partial z^*} = \frac{1}{2}\left[\frac{\partial f}{\partial x} + \jmath \frac{\partial f}{\partial y}\right]$

$\mathbb{CR}$-derivatives of holomorphic functions

$\mathbb{CR}$-derivatives vs. standard $\mathbb{C}$-derivatives for $f(z) = z$ & $f(z) = z^*$

If a function $f = f(z, z^*) = u(x, y) + \jmath v(x, y)$ is holomorphic, then the Cauchy–Riemann conditions are satisfied, that is

$$\frac{\partial u(x,y)}{\partial x} = \frac{\partial v(x,y)}{\partial y} \quad \text{and} \quad \frac{\partial v(x,y)}{\partial x} = -\frac{\partial u(x,y)}{\partial y}$$

Therefore the $\mathbb{R}-$ and $\mathbb{R}^*-$derivatives are

$$\mathbb{R} - \text{der.} : \frac{\partial f}{\partial z}\Big|_{z^*=\text{const.}} = \frac{1}{2}\left[\frac{\partial f}{\partial x} - \jmath\frac{\partial f}{\partial y}\right] = \frac{1}{2}[2\frac{\partial u}{\partial x} + 2\jmath\frac{\partial v}{\partial x}] = f'(z)$$

$$\mathbb{R}^* - \text{der.} : \frac{\partial f}{\partial z^*}\Big|_{z=\text{const.}} = \frac{1}{2}\left[\frac{\partial f}{\partial x} + \jmath\frac{\partial f}{\partial y}\right] = 0$$

↬ **For holomorphic functions the $\mathbb{R}^*$-derivative vanishes and the $\mathbb{R}$-derivative is equivalent to the standard complex derivative**

Example: (i) $f(z) = z = x + \jmath y \quad \Rightarrow \quad \mathbb{R} - \text{der} = 1 \quad \& \quad \mathbb{R}^* - \text{der} = 0,$
(ii) $f(z) = z^* = x - \jmath y \Rightarrow \mathbb{R} - \text{der} = 0 \quad \& \quad \mathbb{R}^* - der = 1$

Example 4: The $\mathbb{CR}$–calculus for non–holomorphic functions, like the standard cost function $J = E\{|e|^2\}$

Consider a real function of complex variable $f(z) = |z|^2 = zz^* = x^2 + y^2$, where $z = x + \jmath y$ (clearly non–holomorphic). Assuming $z \perp z^*$, the $\mathbb{R}$-derivative and the conjugate $\mathbb{R}^*$-derivative are

$$\frac{\partial f}{\partial z} = \frac{\partial (zz^*)}{\partial z} = z^* \quad \text{and} \quad \frac{\partial f}{\partial z^*} = \frac{\partial (zz^*)}{\partial z^*} = z$$

To verify, start from

$$f(z) = f\big(u(x,y) + jv(x,y)\big) = f(u,v) = x^2 + y^2$$

where $u(x,y) = x^2 + y^2$ and $v(x,y) = 0$. Therefore,

$$\mathrm{R-der:} \qquad \frac{\partial f}{\partial z} = \frac{1}{2}\Big[\frac{\partial (x^2+y^2)}{\partial x} - \jmath \frac{\partial (x^2+y^2)}{\partial y}\Big] = \frac{1}{2}\big[2x - \jmath 2y\big] = x - \jmath y = z^*$$

$$\mathrm{R^*-der:} \qquad \frac{\partial f}{\partial z^*} = \frac{1}{2}\Big[\frac{\partial (x^2+y^2)}{\partial x} + \jmath \frac{\partial (x^2+y^2)}{\partial y}\Big] = \frac{1}{2}\big[2x + \jmath 2y\big] = x + \jmath y = z$$

Example 5: Some typical $\mathbb{CR}$-derivatives

Prove these from the definitions of the $\mathbb{R}$ and $\mathbb{R}^*$ derivatives

For the $\mathbb{R}-$ derivative, the function is partially differentiated w.r.t z while keeping z^* constant, and vice versa for the $\mathbb{R}^*-$ derivative.

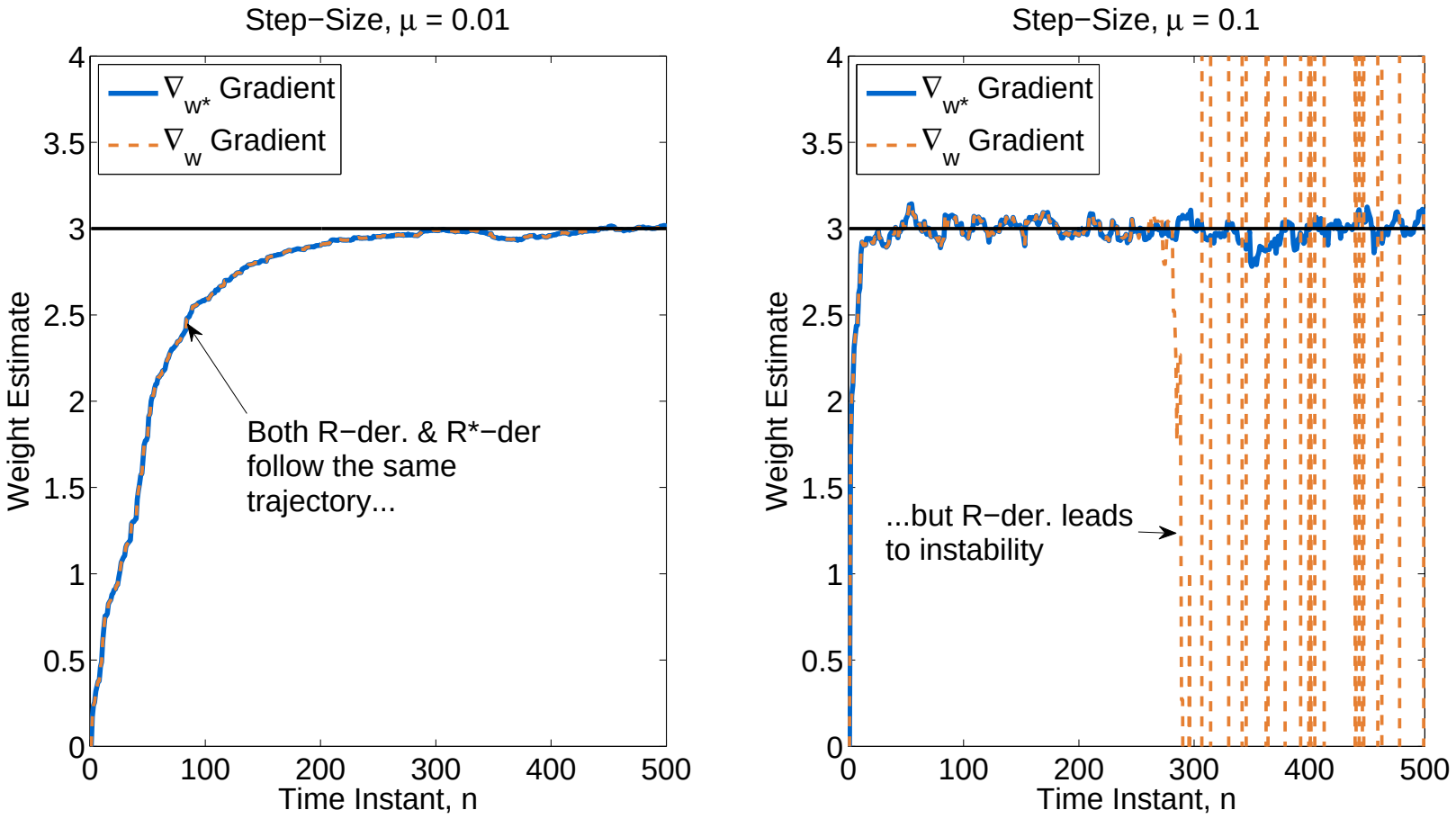
$f(z, z^*)$	$\mathbb{R}-$der	$\mathbb{R}^*-$der	$\mathbb{C}-$der
z	1	0	1
z^*	0	1	Undefined
$\lvert z\rvert^2 = zz^*$	z^*	z	Undefined
z^2z^*	$2\lvert z\rvert^2$	z^2	Undefined
e^z	e^z	0	e^z

$\rightarrow$ If $f(z, z^*)$ is independent of z^*, then the $\mathbb{R}$-derivative of $f(z)$ is equivalent to the standard $\mathbb{C}$-derivative (Cauchy–Riemann);

Which derivative to we choose to compute the gradient?

An example from learning systems: $\mathbb{R}$-der vs. $\mathbb{R}^*$-der? (more later)

Simulation for the CLMS derived using $\mathbb{R}$-der. and $\mathbb{R}^*$-der. $(\mathbf{w}_o = 3)$



Part 3: Complex Statistics

Now that we have familiarised ourselves with the concept of (non-)circularity, we will examine how to use the concept in the domain of second-order statistics and how to design so-called **widely linear** estimators which are second-order optimal for both second-order circular (proper) and second-order noncircular (improper) data.

Back again to the isomorphism between $\mathbb{C}$ and $\mathbb{R}^2$

Moving from real-valued to complex-valued data

$$z \rightarrow z^a \quad \leftrightarrow \quad \begin{bmatrix} z \\ z^* \end{bmatrix} = \begin{bmatrix} 1 & \jmath \\ 1 & -\jmath \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

whereas in the case of complex–valued signals, we have

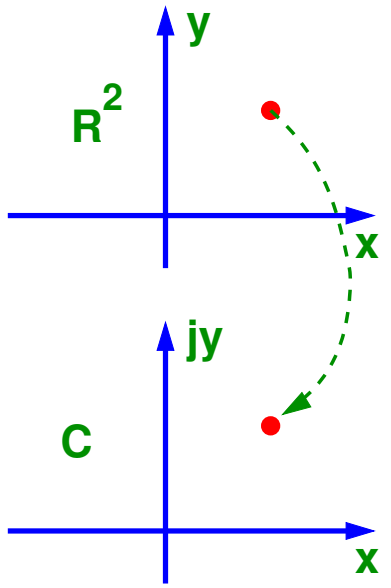
$$\mathbf{z} \rightarrow \mathbf{z}^a \quad \leftrightarrow \quad \begin{bmatrix} \mathbf{z} \\ \mathbf{z}^* \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \jmath\mathbf{I} \\ \mathbf{I} & -\jmath\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$$

For convenience, the "augmented" complex vector $\mathbf{v} \in \mathbb{C}^{2N\times 1}$ can be introduced as

$$\mathbf{v} = [z_1, z_1^*, \ldots, z_N, z_N^*]^T$$

$$\mathbf{v} = \mathbf{A}\mathbf{w}, \qquad \mathbf{w} = [x_1, y_1, \ldots, x_N, y_N]^T$$

where matrix $\mathbf{A} = diag(\mathbf{J}, \ldots, \mathbf{J}) \in \mathbb{C}^{2N\times 2N}$ is block diagonal and transforms the **composite** real vector $\mathbf{w}$ into the **augmented complex vector** $\mathbf{v}$.



The multivariate complex normal distribution

We cannot introduce a CDF $\not\hookrightarrow$ pdf introduced via duality with $\mathbb{R}$

Recall, the relationships like "$<$" or "$\geq$" make no sense in $\mathbb{C}$.

$$\mathbf{V} = cov(\mathbf{v}) = E[\mathbf{v}\mathbf{v}^H] = \mathbf{A}\mathbf{W}\mathbf{A}^H \qquad \text{where} \quad \mathbf{W} = \mathbf{w}\mathbf{w}^H$$

Using the result by Vanden Bos from 1995

$$\begin{aligned} \mathbf{w} &= \mathbf{A}^{-1}\mathbf{v} = \frac{1}{2}\mathbf{A}^H\mathbf{v} \\ det(\mathbf{W}) &= \left(\frac{1}{2}\right)^{2N} det(\mathbf{V}) \\ \mathbf{w}^T\mathbf{W}^{-1}\mathbf{w} &= \mathbf{v}^H\mathbf{V}^{-1}\mathbf{v} \end{aligned}$$

The multivariate *generalised complex normal distribution* (GCND) can now be expressed as

$$f(\mathbf{v}) = \frac{1}{\pi^N\sqrt{det(\mathbf{V})}}\, e^{-\frac{1}{2}\mathbf{v}^H\mathbf{V}^{-1}\mathbf{v}}$$

and has been derived without any restriction.

Circular complex random variables

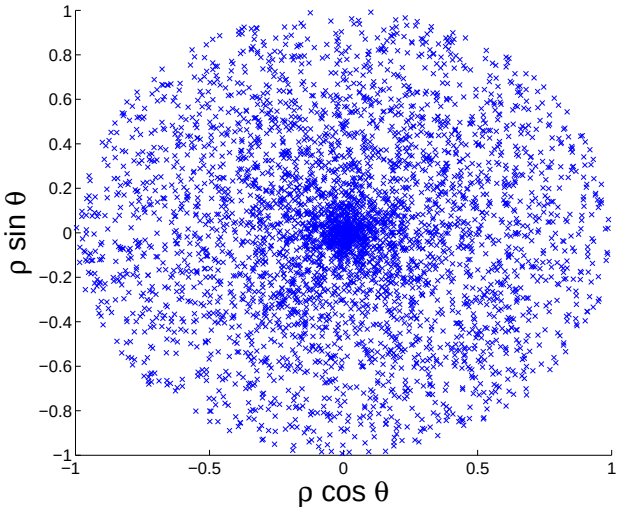
Try to generate complex ran. var. from various distrib. in MATLAB

Circularity $\not\hookrightarrow$ Rotation invariant distrib.

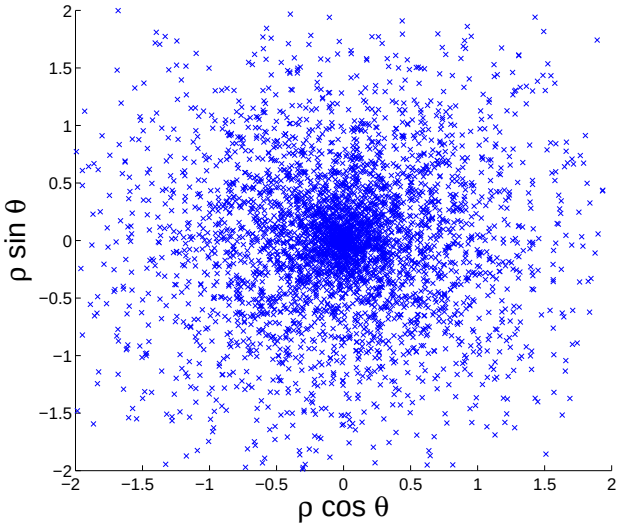
$$p(\rho, \theta) = p(\rho, \theta - \phi)$$

1. The name of the distribution takes after the distribution of the real–valued random variable ρ with a pdf $p(\rho)$;
2. It can be Gaussian, uniform, etc.
3. Take another real–valued random variable θ, which must be uniformly distributed on $[0, 2\pi]$ and independent of ρ;
4. Construct the complex random variable $Z = X + jY$ as

$$X = \rho\cos(\theta), \qquad Y = \rho\sin(\theta)$$



(g) Uniform circular



(h) Gaussian circular

Complex circularity

Definition: A complex-valued random is called **circular** if its probability distribution is not dependent on the angle, that is, the distribution is "**rotation invariant**".

For simplicity, we consider univariate complex-valued random variables; the concepts are readily extended to the multivariate case.

Recall that for an iid complex-valued random variable $Z = X + \jmath Y$, the pdf

$$\mathcal{P}_Z(z) = \mathcal{P}_X(x)\mathcal{P}_Y(y)$$

On the other hand, in the case of a **rotation invariant** $\mathcal{P}_Z(z)$, its pdf is only be dependent of the **Euclidean distance** from the origin in the complex domain. Therefore, if the random variable Z is circular, we have

$$g(r) = \mathcal{P}_Z(z) = \mathcal{P}_X(x)\mathcal{P}_Y(y)$$

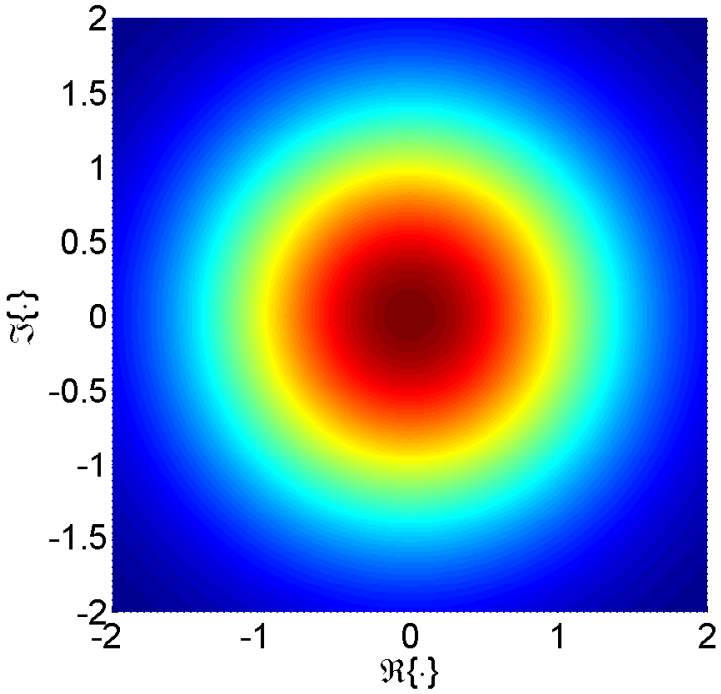
where $r = \sqrt{x^2 + y^2}$ and $g(\cdot)$ is a general function.

Circularity (inherently assumed in standard processing in $\mathbb{C}$)

Some circular distributions

Circular complex-valued random variables

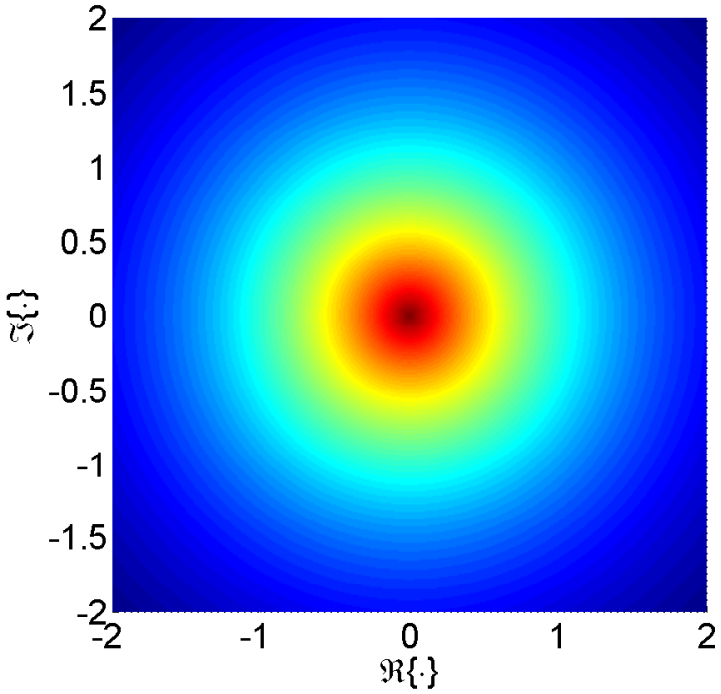
The distribution of R is Rayleigh. Thus, the distributions of the real and imaginary parts are Gaussian.



circular Rayleigh distribution

The distribution of R is exponential

$$\mathcal{P}_R(r) = \lambda e^{-\lambda r},\ \lambda = 1$$



circular exponential distribution

More on circularity
A noncircular distribution

Independent real & imaginary distributions but not circular!

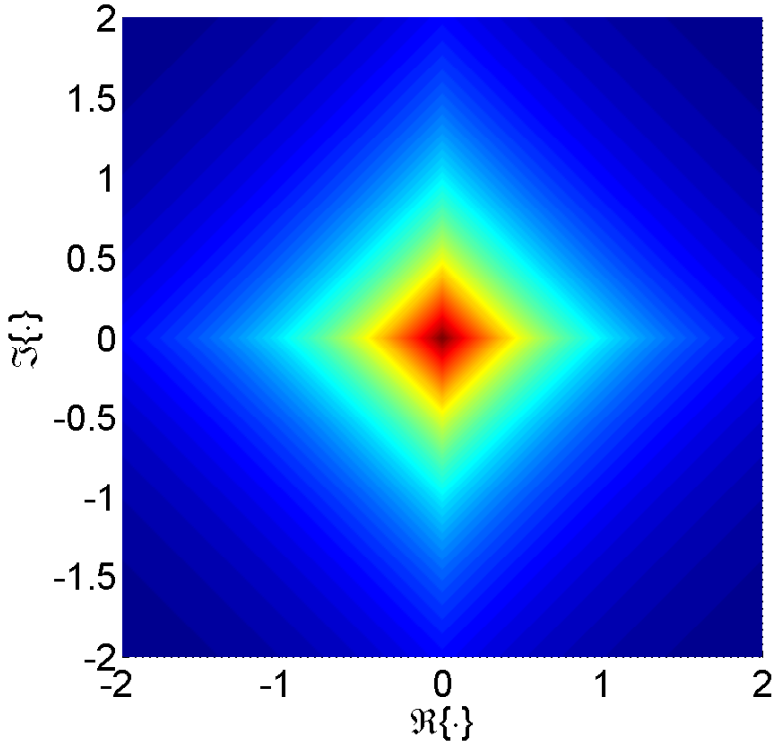
Distributions of the real and imaginary part are **independent Laplace distributions**

$$\mathcal{P}_X(x) = \frac{1}{2}e^{-|x|} \text{ and } \frac{1}{2}\mathcal{P}_Y(y) = \frac{1}{2}e^{-|y|}$$

Thus,

$$\mathcal{P}_Z(z = x + \jmath y) = \frac{1}{4}e^{-(|x|+|y|)}$$

Although the distributions on the real and imaginary axes are independent and hence uncorrelated, the resulting distribution is not rotation invariant, that is, it is non-circular.



Other definitions of circularity

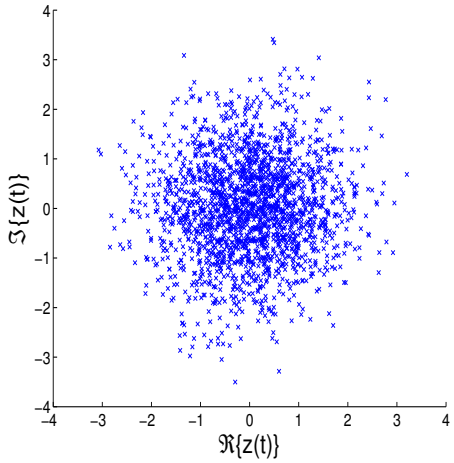
Via Probability density function, Characteristic Function, Cumulants

- *Probability density function.* A complex random variable Z is circular if its pdf is a function of only the product zz^*, that is[1]

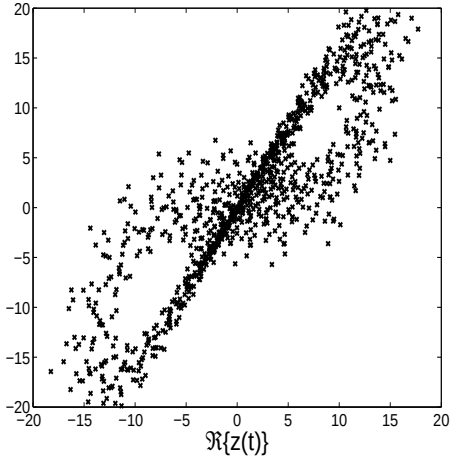
$$p_{Z,Z^*}(z, z^*) = p_{Z_\phi, Z_\phi^*}(z_\phi, z_\phi^*)$$

and for for Gaussian CCRVs we have

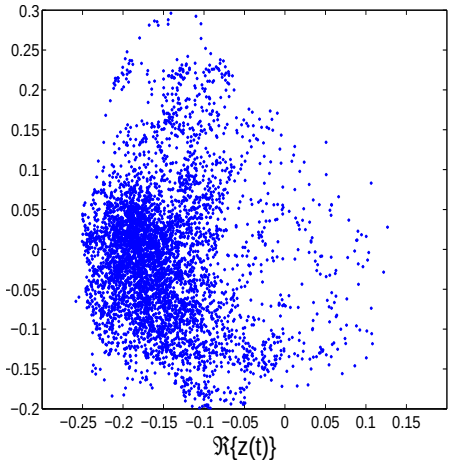
$$p_{Z,Z^*}(z, z^*) = \frac{1}{\pi\sigma^2} e^{-zz^*/\sigma^2}$$



(i) Complex AR(4)



(j) Complex Lorenz



(k) Complex wind

[1] The pdf of a circular complex random variable is function of only the modulus of z, **and not of z^*.**

Key: What are we doing wrong $\hookrightarrow$ Widely Linear Model

Consider the MSE estimator of a signal y in terms of another observation x

$$\hat{y} = E[y|x]$$

For zero mean, jointly normal y and x, the solution is

$$\hat{y} = \mathbf{h}^T \mathbf{x}$$

In standard MSE in the complex domain $\hat{y} = \mathbf{h}^H \mathbf{x}$, however

$$\hat{y}_r = E[y_r|x_r, x_i] \quad \& \quad \hat{y}_i = E[y_i|x_r, x_i]$$

$$thus \qquad \hat{y} = E[y_r|x_r, x_i] + \jmath E[y_i|x_r, x_i]$$

Upon employing the identities $x_r = (x + x^*)/2$ and $x_i = (x - x^*)/2\jmath$

$$\hat{y} = E[y_r|x, x^*] + \jmath E[y_i|x, x^*]$$

and thus arrive at the **widely linear** estimator for general complex signals

$$y = \mathbf{h}^H \mathbf{x} + \mathbf{g}^H \mathbf{x}^*$$

We can now process general (noncircular) complex signals!

Key: Dealing with Complex Statistics

Provides us with a tremendous amount of structure

For $\mathbf{z} = \mathbf{x} + \jmath\mathbf{y}$, the 'augmented' weight and input vectors are

$$\mathbf{w}^a = [\mathbf{h}^T, \mathbf{g}^T]^T \text{ and } \mathbf{z}^a = [\mathbf{z}^T, \mathbf{z}^H]^T$$

so that

$$y = \mathbf{w}^{aH}\mathbf{z}^a$$

and the 'augmented' covariance matrix then becomes

$$\mathbf{C}_{zz}^a = E\left[\mathbf{z}^a\mathbf{z}^{aH}\right] = E\left[\begin{array}{c}\mathbf{z}\\ \mathbf{z}^*\end{array}\right]\left[\mathbf{z}^H\mathbf{z}^T\right] = \left[\begin{array}{cc}\mathbf{C}_{zz} & \mathbf{P}_{zz}\\ \mathbf{P}_{zz}^* & \mathbf{C}_{zz}^*\end{array}\right]$$

Remark #1: In general, the standard covariance matrix

$$\mathbf{C}_{zz} = E\{\mathbf{z}\mathbf{z}^H\}$$

does not completely describe the second order statistics of $\mathbf{z}$

Remark #2: The **pseudocovariance** or **complementary covariance** matrix

$$\mathbf{P}_{zz} = E\{\mathbf{z}\mathbf{z}^T\}$$

also needs to be taken into account.

Second order (non-)circularity $\rightsquigarrow$ (im-)properness

Remark #3: For second-order circular, the pseudocovariance matrix

$$\mathbf{P}_{zz} = E\{\mathbf{z}\mathbf{z}^T\} = \mathbf{0}$$

vanishes and such signals are called **second order circular** or **proper**.

For illustration, consider the scalar case

$$E\{z \times z^T\} = E\{x^2\} - E\{y^2\} + 2\jmath E\{xy\} = \sigma_x^2 - \sigma_y^2 + 2\jmath\rho_{xy}$$

Clearly, for circular (proper data), the powers in the real and imaginary channels are equal (and so $\sigma_x^2 = \sigma_y^2$), and the real and imaginary channel are not correlated, so that $\rho_{xy} = 0$. Therefore, $pcov = 0$.

Remark #4: General complex random processes are second-order noncircular or ***improper***.

'Properness/improperness' are second order statistical properties and 'circularity/noncircularity' are properties of the probability density function.

Measuring improperness $\hookrightarrow$ an intuitive example

For scalar data, the covariance $c = E[|z|^2] = E[zz^*]$, and pseudocovariance $p = E[z^2]$

Q: Consider the estimation of a zero-mean complex r.v. $z \in \mathbb{C}$ from its conjugate, that is

$$\hat{z} = hz^*$$

Solution: Find an estimate of h that minimises

$$J_{\text{MSE}} = E[|e|^2] = E[|z^* - \hat{z}^*|^2]$$

The Wiener solution is then

$$h_{\text{opt}} = E[zz^*]^{-1}E[zz] = \frac{p}{c} = \rho_z$$

where ρ_z is referred to as the **circularity quotient**. We now have

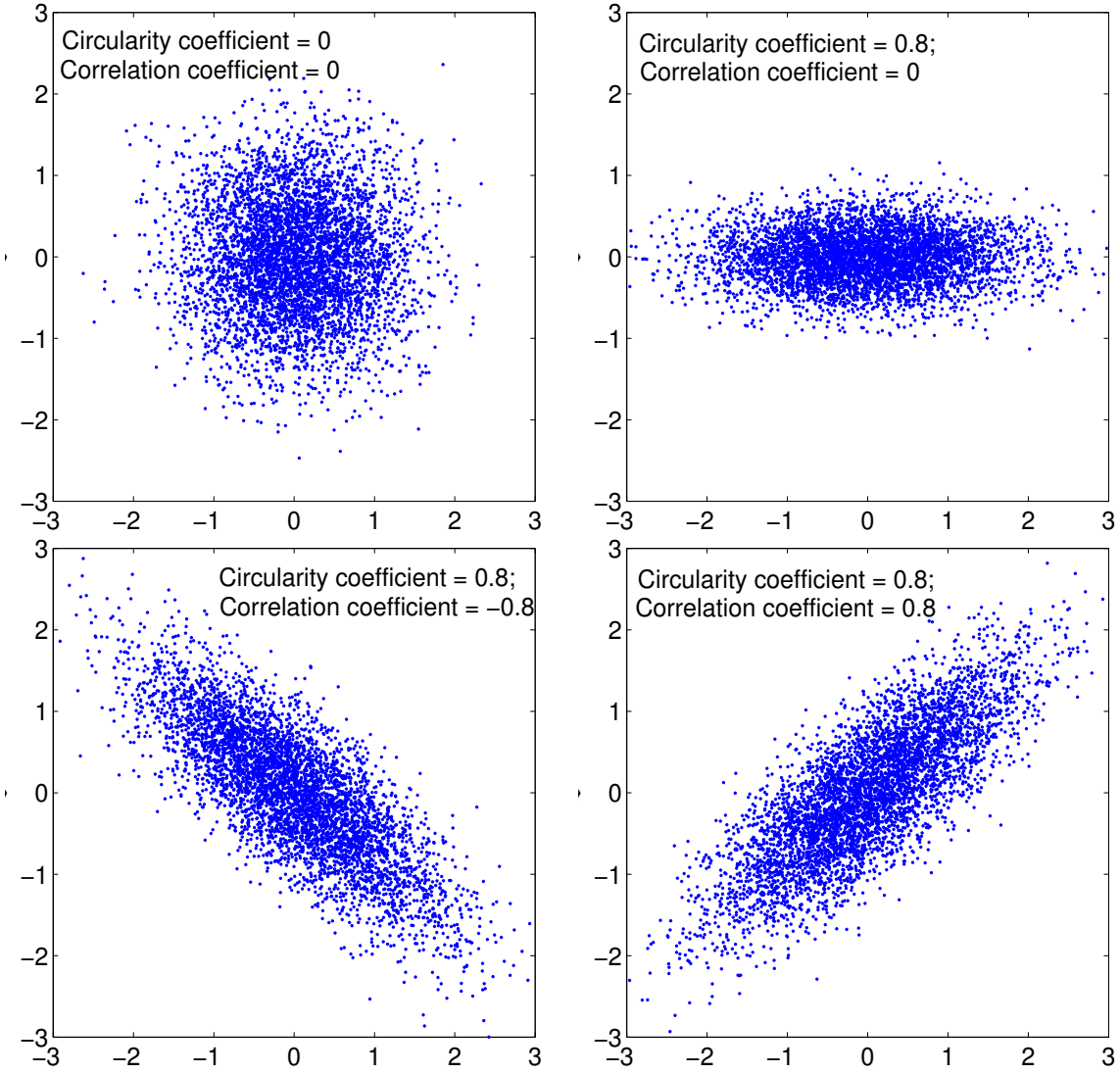
$$\rho_z = \frac{1}{c}\left(\sigma_x^2 - \sigma_y^2 + 2\jmath c_{xy}\right)$$

where the real part of ρ_z gives the power difference between the real and imaginary parts while the imaginary part of ρ_z models their correlation (both normalized by total power).

$$\text{Now, the } \textbf{circularity coefficient} \qquad \eta = \frac{|p|}{c} \qquad 0 \leq \eta \leq 1$$

Different kinds of noncircularity

'Noncircular' and 'Improper' used interchangeably, but these are not identical



So, **the degree of circularity** can be used as a **fingerprint** of a signal, allowing us enormous additional freedom in estimation, compared with standard **strictly linear** systems.

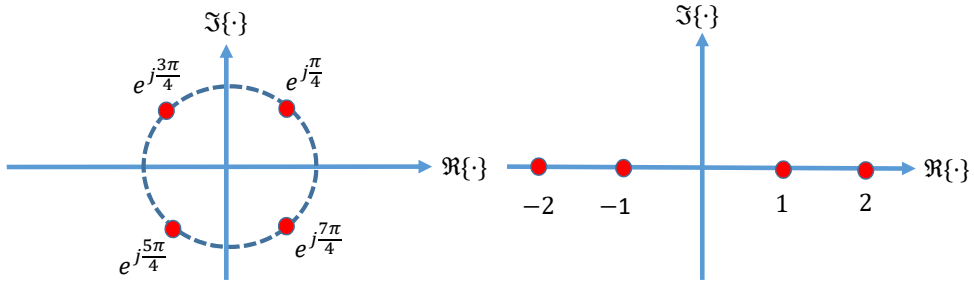
For instance, we can now differentiate between different Gaussian signals!

Recall: Real valued ICA cannot separate two Gaussian signals.

Circularity in data communications
Constellations in communications, 4 symbols

Consider a communication system with 4 complex-valued symbols.

The most widely used modulation schemes are quadrature phase shift keying (**QPSK**) and amplitude shift keying (**ASK**).



QPSK **ASK**

Although these constellations are arranged so that the distances of each point to its nearest neighbour is equal in both cases, the **QPSK is more compact**.

QPSK second-order statistics:

$$\text{covariance}: \quad c = E[zz^*] = 1$$
$$\text{pseudocov.}: \quad p = E[zz] = 0$$

ASK second-order statistics:

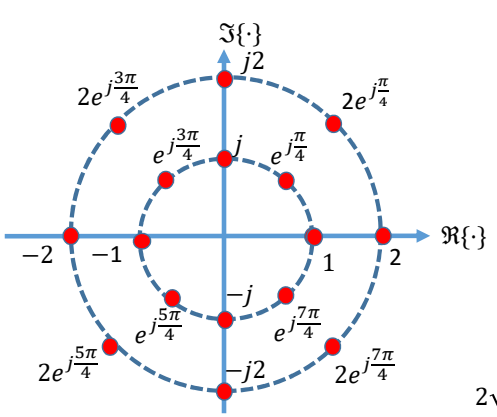
$$\text{covariance}: \quad c = E[zz^*] = 2.5$$
$$\text{pseudocov.}: \quad p = E[zz] = 2.5$$

In the case of the **QPSK** there is no power difference or correlation between the real and imaginary components, resulting in the impropriety measure of $\rho = 0$.
In the case of the **ASK** all the information is on the real axis, resulting in the impropriety measure of $\rho = 1$ (real-valued signals are maximally non-circular).

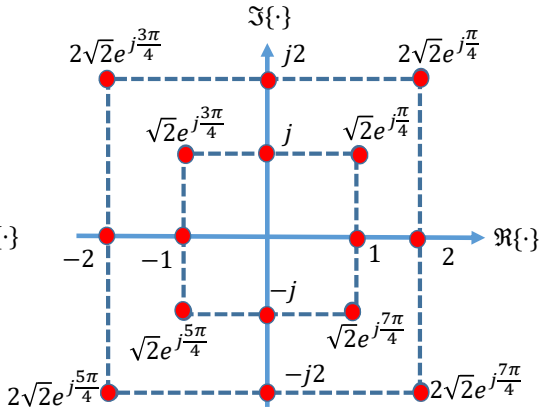
Circularity in communications
Constellations in communications, 16 symbols

Now, consider a communication system with 16 complex-valued symbols.

The most widely used modulation schemes are the amplitude and quadrature phase shift keying **APSK**) and quadrature amplitude modulation **QAM**).
Note that the constellation for **16-APSK** is more **compact** than that of the **16-QAM**.



16-APSK



16-QAM

16-APSK second-order statistics:

$$c = E[zz^*] = 2.5$$
$$p = E[zz] = 0$$

16-QAM second-order statistics:

$$c = E[zz^*] = 3.75$$
$$p = E[zz] = 0$$

Although **both methods are proper**, **only the 16-APSK is circular** (losely speaking).
Note that **circular** constellations offer better **energy efficiency**, whereas **non-circular constellations are more resilient to noise**, especially when using widely-linear processing.

Autoregressive Modelling in $\mathbb{C}$

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 AR model (WLAR)

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

Widely linear normal equations

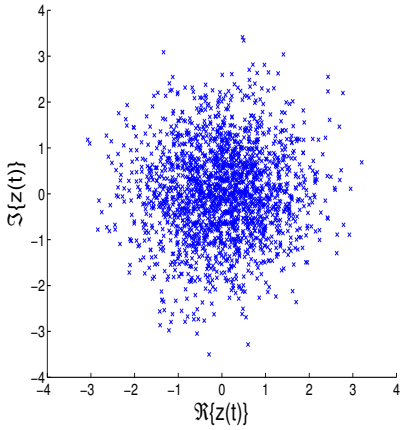
$$\begin{bmatrix} \mathbf{h}^* \\ \mathbf{g}^* \end{bmatrix} = \begin{bmatrix} \mathbf{C} & \mathbf{P} \\ \mathbf{P}^* & \mathbf{C}^* \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{c} \\ \mathbf{p}^* \end{bmatrix}$$

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

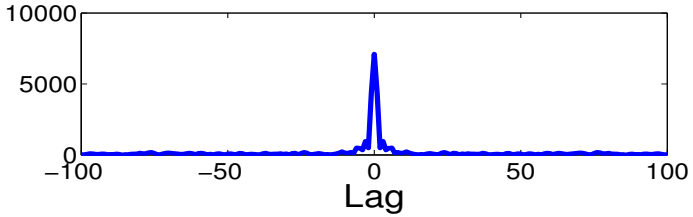
Example 6: Pseudocovariance $\leftrightsquigarrow$ properness

Real-world data are rarely circular (if nothing else $\not\hookrightarrow$ short length, artefacts)?

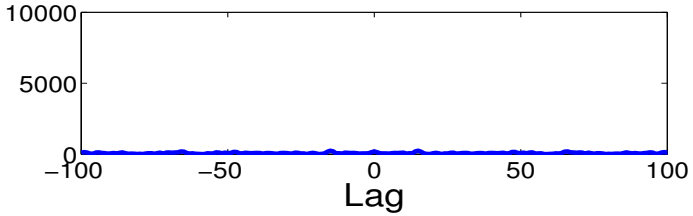
Complex AR(4) process (circular)



Covariance of the AR(4) signal

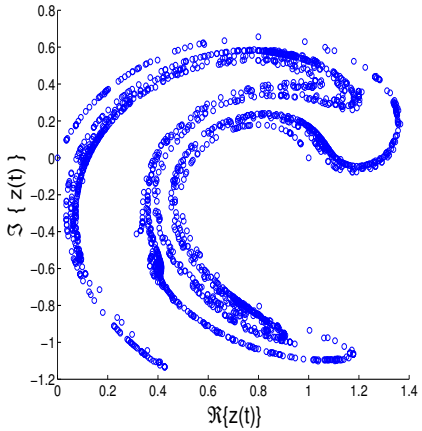


Pseudocovariance of the AR(4) signal

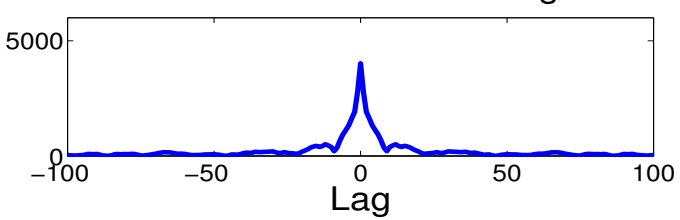


Complex AR(4) process (proper)

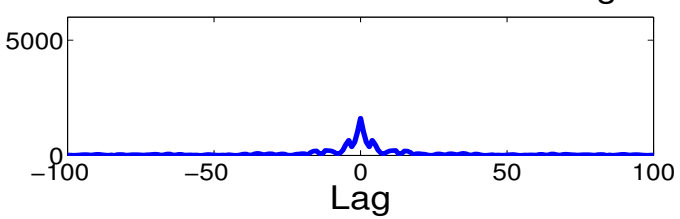
Complex Ikeda map (noncircular)



Covariance of the Ikeda signal



Pseudocovariance of the Ikeda signal

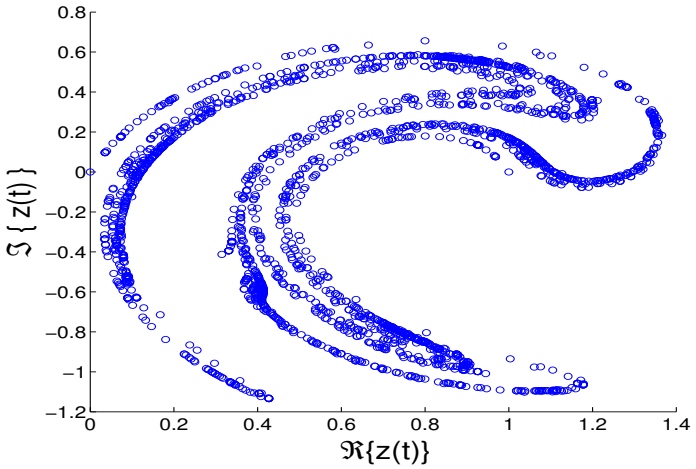


Complex Ikeda map (improper)

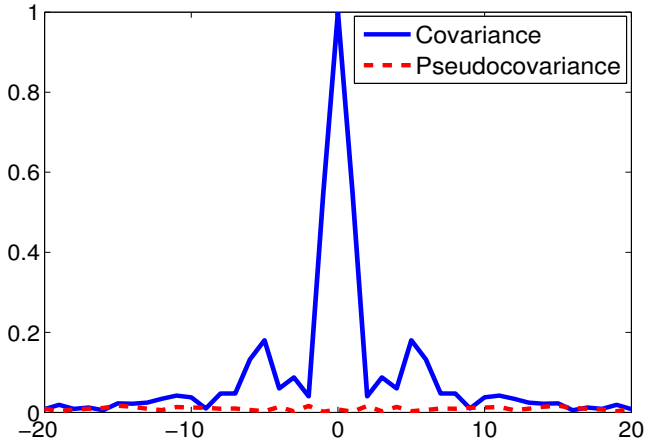
Key: WL model caters for both proper and improper data

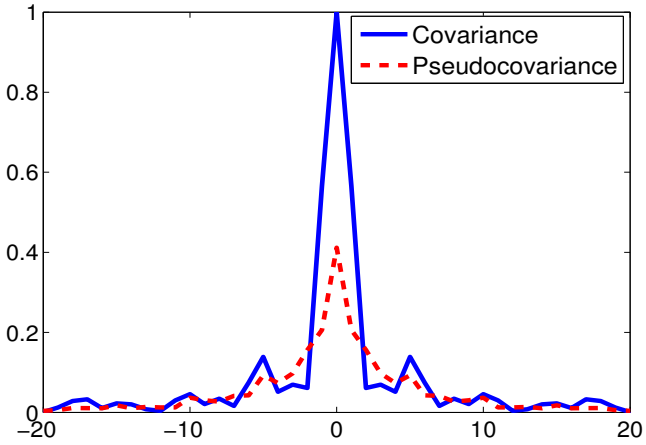
This is a rigorous way to model general complex signals!

Circularity for Ikeda map

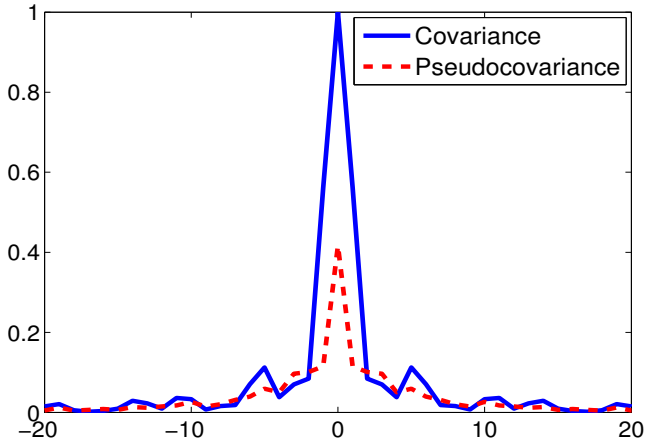


AR model of Ikeda signal





Covariances: Original Ikeda map



Widely linear AR model of Ikeda map

How does non-circularity influence estimation in $\mathbb{C}$?

Real-world example: Estimation in the Smart Grid

Three-phase voltages can be represented as a single-channel complex signal by first using the **Clarke Transform**,

$$\begin{bmatrix} v_0(k) \\ v_\alpha(k) \\ v_\beta(k) \end{bmatrix} = \underbrace{\sqrt{\frac{2}{3}} \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}}_{\text{Clarke Matrix}} \underbrace{\begin{bmatrix} V_a(k)\cos(\omega nT + \phi_a) \\ V_b(k)\cos(\omega nT + \phi_b - \frac{2\pi}{3}) \\ V_c(k)\cos(\omega nT + \phi_c + \frac{2\pi}{3}) \end{bmatrix}}_{\text{Three-phase voltage}}$$

Then by forming the complex-valued $\alpha\beta$ voltage: $v(k) = v_\alpha(k) + \jmath v_\beta(k)$:

$$v(k) = v_\alpha(k) + \jmath v_\beta(k) = A(k)e^{\jmath\omega kT} + B(k)e^{-\jmath\omega kT}$$

$$A(k) = \frac{\sqrt{6}}{6}\left[V_a(k)e^{\jmath\phi_a} + V_b(k)e^{\jmath\phi_b} + V_c(k)e^{\jmath\phi_c}\right],$$

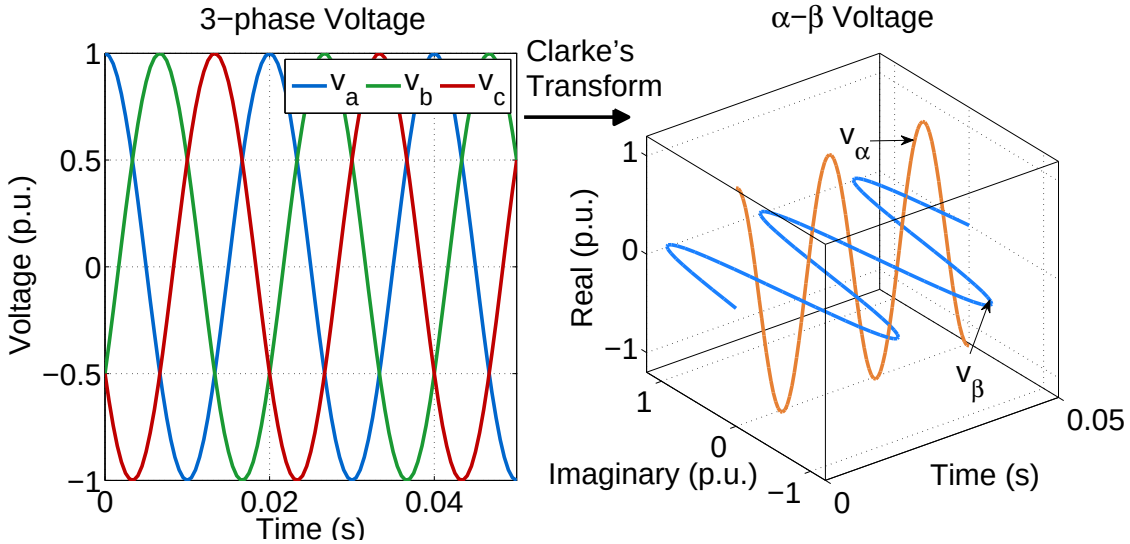
$$B(k) = \frac{\sqrt{6}}{6}\left[V_a(k)e^{-\jmath\phi_a} + V_b(k)e^{-\jmath\left(\phi_b+\frac{2\pi}{3}\right)} + V_c(k)e^{-\jmath\left(\phi_c-\frac{2\pi}{3}\right)}\right]$$

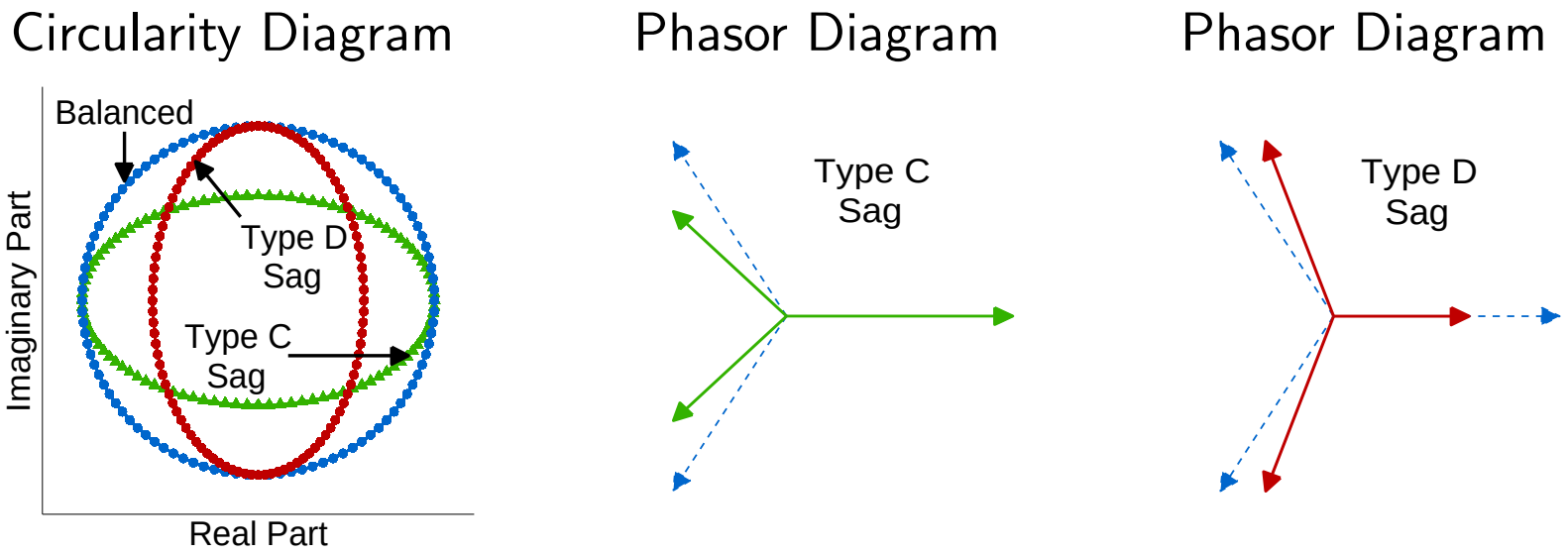
For balanced systems i.e. $V_a(k) = V_b(k) = V_c(k)$ and $\phi_a = \phi_b = \phi_c$,

$$B(k) = 0$$

Example 7: Noncircularity, a fault signature in power grid

Visualising the Clarke transform and noncircular voltage sags



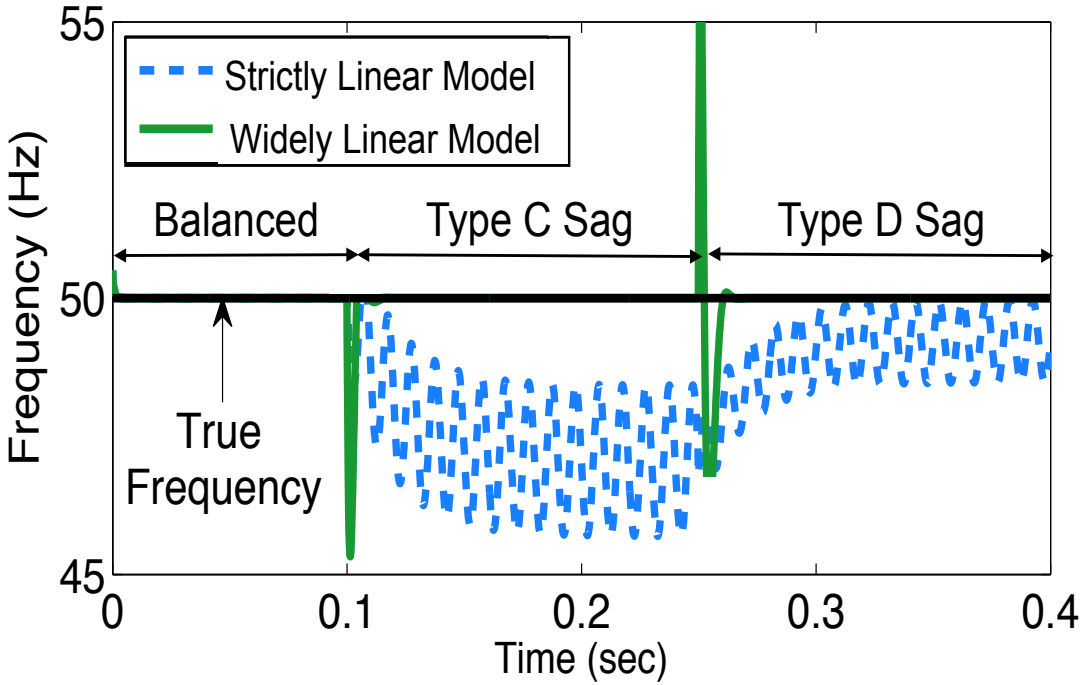


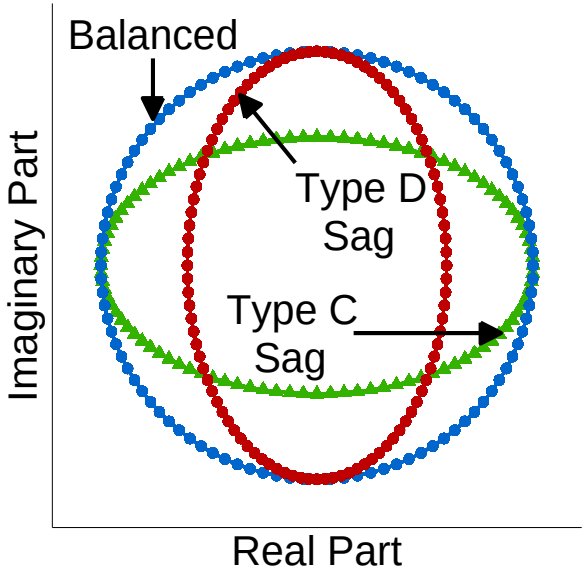
Example 7: Enhanced degrees of freedom using WLM

WLM for frequency estimation in power grid

◦ We are jumping a little ahead, here the idea is just to show that by the additional degree of freedom in WLM, we can greatly improve the performance of any signal processing algorithm in $\mathbb{C}$

↳ Figure below: The widely linear model was able to estimate the frequency for both **circular** (balanced) and **noncircular** (unbalanced) voltages.





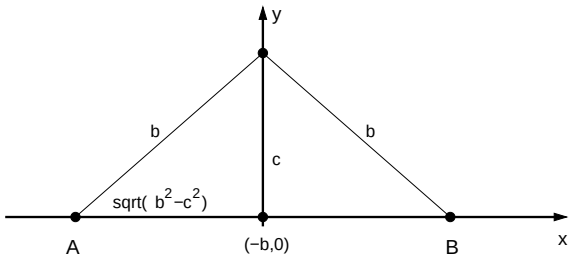
Lecture summary

- We have demystified several basic concepts in complex calculus
- Problems with the Cauchy-Riemann derivatives
- The CR-calculus deals with both analytic and non-analytic functions
- This has made possible optimisation of real functions of complex variables (typical minimum error power cost functions)
- Complex noncircularity $\not\hookrightarrow$ a mathematical microscope into data behaviour, and a signature of many conditions
- Circularity $\not\hookrightarrow$ property of a probability distribution, properness is a second order statistical property (pseudocovariance vs covariance)
- Widely linear modelling $\not\hookrightarrow$ deals with both proper and improper signals
- Examples in communications and smart grid

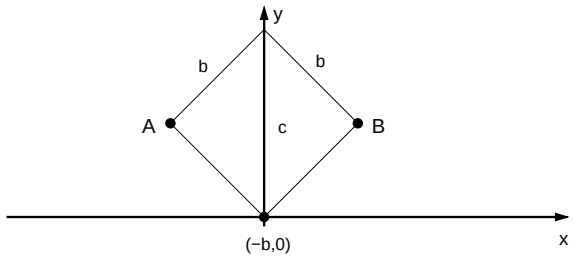
Some trivia: The role of geometry

◦ Complex numbers were only accepted after they had a geometric interpretation, but it was initially only possible for $b^2 - c^2 \geq 0$.

◦ Wallis - complex number a point on the plane (solutions A & B)



Real solution



Complex solution

◦ In 1732 Leonhard Euler, $x^n - 1 = 0 \;\rightarrow\; \cos\theta + \sqrt{-1}\sin\theta$

◦ Abraham de Moivre (1730) and again Euler (1748), introduced the famous formulas

$$(\cos\theta + \jmath\sin\theta)^n = \cos n\theta + \jmath\sin n\theta$$

$$\cos\theta + \jmath\sin\theta = e^{\jmath\theta}$$

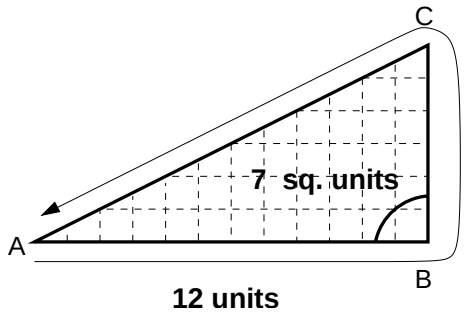
◦ In 1806 Argand interpreted $\jmath = \sqrt{-1}$ as a rotation by 90^o and introduced Argand diagram, $z = x + \jmath y$, and the modulus $\sqrt{x^2 + y^2}$.

◦ In 1831 Karl Friedrich Gauss introduced $\imath = \sqrt{-1}$ and complex algebra.

Some trivia: History of complex numbers

Find a triangle of Area = 7 and Perimeter = 12

- Heron of Alexandria (60 AD)



To solve this, let the side $|AB| = x$, and the height $|BC| = h$, to yield

$$\begin{aligned} area &= \frac{1}{2} x\ h \\ perimeter &= x + h + \sqrt{x^2 + h^2} \end{aligned}$$

In order to solve for x we need to find the roots of

$$6x^2 - 43x + 84 = 0$$

However, this equation does not have real roots.

Recap: What is a derivative?

we need to understand where the pseudo-gradient comes from

The definition of derivative for $f(x) \in \mathbb{R}$:

$$f'(x) = \lim_{\Delta_x \to 0} \frac{f(x+\Delta_x)-f(x)}{\Delta_x}$$

For a complex function

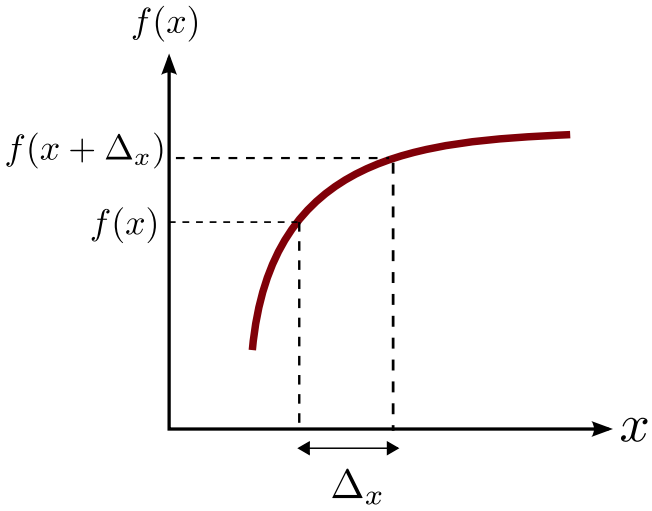
$$f(z) = u(x,y) + \jmath v(x,y)$$

to be differentiable at $z = x + \jmath y$, the limit must converge to a unique complex number no matter how $\Delta z = \Delta_x + \jmath \Delta_y \to 0$.

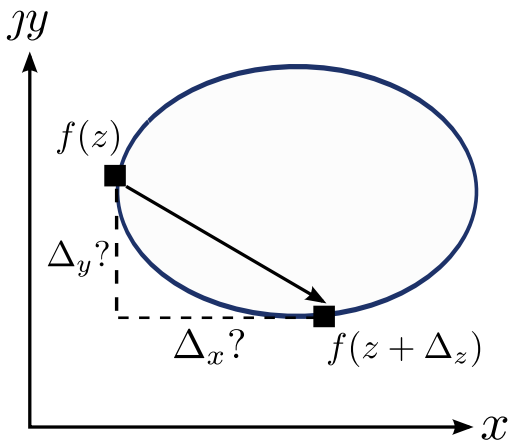
$$f'(z) = \lim_{\Delta_z \to 0} \frac{f(z+\Delta_z)-f(z)}{\Delta_z}$$

So, the complex derivative is only defined for analytic functions.

Real-Domain:



Complex-Domain:



Recap: Complex derivatives, Cauchy-Riemann conditions

Conditions for the derivative to exist in $\mathbb{C}$

For $f(z)$ to be analytic, a unique limit must exist regardless of how Δz approaches zero

$$f'(z) = \lim_{\substack{\Delta_x \to 0 \\ \Delta_y \to 0}} \frac{\left[u\left(x+\Delta_x, y+\Delta_y\right) + \jmath v\left(x+\Delta_x, y+\Delta_y\right)\right] - \left[u(x,y) + \jmath v(x,y)\right]}{\Delta_x + \jmath \Delta_y}$$

must exist regardless of how Δz approaches zero. It is convenient to consider the two following cases for the $\mathbb{C}-$ derivatives:

Case 1: $\Delta_y = 0$ and $\Delta_x \to 0$, which yields

$$\begin{aligned} f'(z) &= \lim_{\Delta_x \to 0} \frac{\left[u(x+\Delta_x,y) + \jmath v(x+\Delta_x,y)\right] - \left[u(x,y) + \jmath v(x,y)\right]}{\Delta_x} \\ &= \lim_{\Delta_x \to 0} \frac{u(x+\Delta_x,y) - u(x,y)}{\Delta_x} + \jmath \frac{v(x+\Delta_x,y) - v(x,y)}{\Delta_x} \\ &= \frac{\partial u(x,y)}{\partial x} + \jmath \frac{\partial v(x,y)}{\partial x} \end{aligned}$$

Recap: Complex derivatives, Cauchy-Riemann conditions

Conditions for the derivative to exist in $\mathbb{C}$

Case 2: $\Delta_x = 0$ and $\Delta_y \to 0$, which yields

$$\begin{aligned} f'(z) &= \lim_{\Delta_y \to 0} \frac{\left[u\big(x,y+\Delta_y\big)+\jmath v\big(x,y+\Delta_y\big)\right]-\left[u(x,y)+\jmath v(x,y)\right]}{\jmath\Delta_y} \\ &= \lim_{\Delta_y \to 0} \frac{u\big(x,y+\Delta_y\big)-u(x,y)}{\jmath\Delta_y} + \frac{v\big(x,y+\Delta_y\big)-v(x,y)}{\Delta_y} \\ &= \frac{\partial v(x,y)}{\partial y} - \jmath\frac{\partial u(x,y)}{\partial y} \end{aligned}$$

For continuity, the limits from Case 1 and Case 2 must be identical, which yields **the Cauchy-Riemann equations**

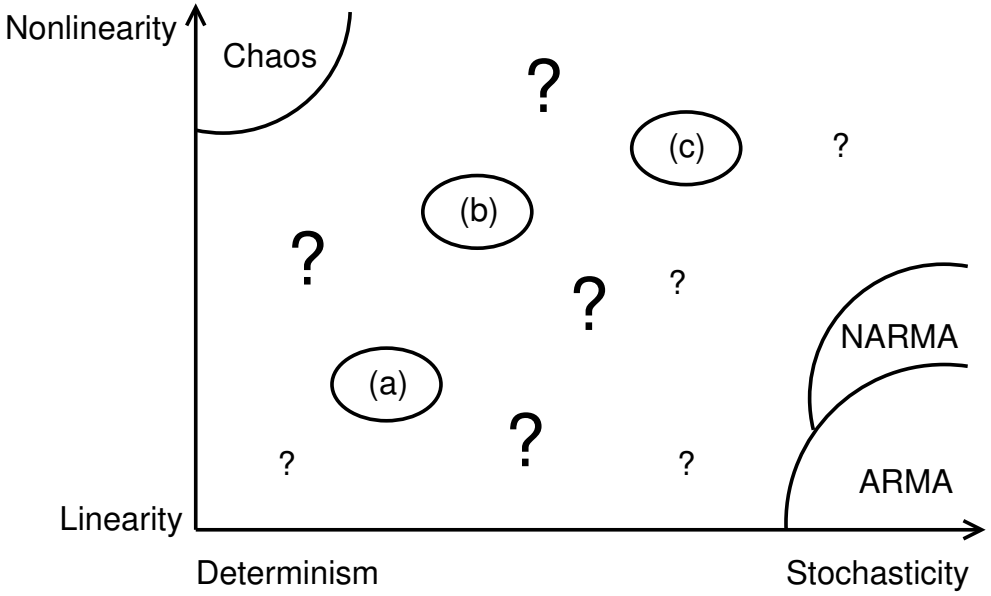
$$\frac{\partial u(x,y)}{\partial x} = \frac{\partial v(x,y)}{\partial y}, \qquad \frac{\partial v(x,y)}{\partial x} = -\frac{\partial u(x,y)}{\partial y}$$

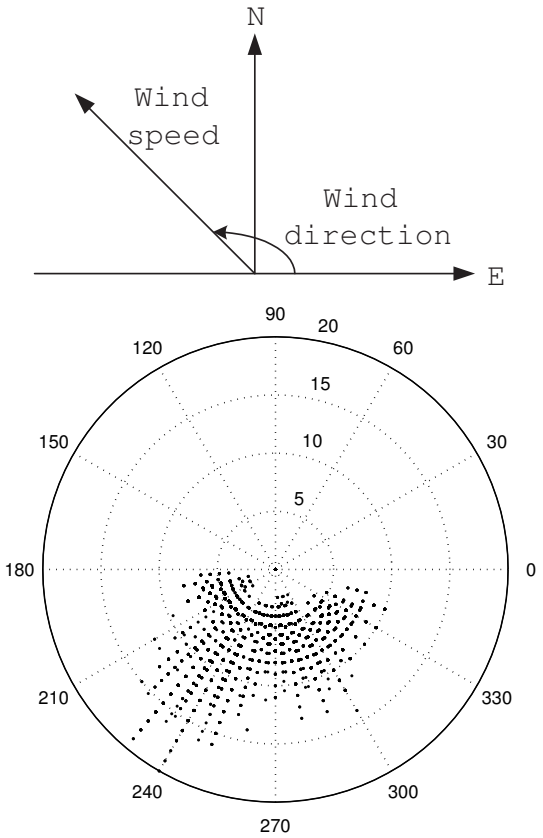
This introduces a tremendous amount of structure (restrictions) in the calculus, as shown in an intuitive example on the next two slides.

Signal modality – So why are complex signals different?

(many expressions are conformal $\not\hookrightarrow$ but dangerous to directly apply real tools!)

Deterministic vs. Stochastic nature
Linear vs. Nonlinear nature





Change in signal modality can indicate e.g. health hazard (fMRI, HRV)

Real world signals are denoted by '???'

- $\exists$ a unique signature of complex signals?
- $\not\hookrightarrow$ **degree of noncircularity**

Appendix 1: Noncircularity and I/Q imbalance $\hookrightarrow$ A proof

Derivation:

The modulated passband signal $x_p(n)$ is given by

$$\begin{aligned} x_p(n) &= [s_I(n) * h_I(n)] \cos \omega_c n - [s_Q(n) * h_Q(n)] g \sin(\omega_c n + \varphi) \\ &= [\underbrace{s_I(n) * h_I(n) + g \sin \varphi s_Q(n) * h_Q(n)}_{x_I(n)}] \cos \omega_c n - \underbrace{g \cos \varphi}_{x_Q(n)} \sin \omega_c n \end{aligned}$$

Upon extracting the baseband signal from $x_p(n)$, and taking the in-phase and quadrature branches as the real and imaginary parts of $x(n)$, we have

$$\begin{aligned} x(n) &= x_I(n) + j x_Q(n) \\ &= \underbrace{\frac{1}{2}[h_I(n) + g e^{-j\varphi} h_Q(n)]}_{\mu(n)} * s(n) + \underbrace{\frac{1}{2}[h_I(n) - g e^{-j\varphi} h_Q(n)]}_{\nu(n)} * s^*(n) \end{aligned}$$

where $s(n) = s_I(n) + j s_Q(n)$

In a narrow-band scenario, the I/Q imbalance becomes frequency-independent, that is, $h_I(n) = h_Q(n) \approx \delta(n)$, and so

$$x(n) = \underbrace{\frac{1}{2}[1 + g e^{-j\varphi}]}_{\mu} s(n) + \underbrace{\frac{1}{2}[1 - g e^{-j\varphi}]}_{\nu} s^*(n)$$

Appendix 2: The depressed cubic (so called 'cubic formula') implicitly uses complex numbers

◦ In the 16th century Niccolo Tartaglia and G. Cardano considered closed formulas for the roots of third- and fourth-order polynomials.

◦ Cardano first introduced complex numbers in his book *Ars Magna* in 1545, as a tool for finding roots of the 'depressed cubic' $x^3 + ax + b = 0$.

$$ay^3 + by^2 + cy + d = 0 \quad \text{substitute} \quad y = x - \frac{1}{3}b \quad \Rightarrow \quad x^3 + \beta x + \gamma = 0$$

◦ Scipione del Ferro of Bologna and Tartaglia showed that the depressed cubic can be solved as

$$x = \sqrt[3]{-\frac{\gamma}{2} + \sqrt{\frac{\gamma^2}{4} + \frac{\beta^3}{27}}} + \sqrt[3]{-\frac{\gamma}{2} - \sqrt{\frac{\gamma^2}{4} + \frac{\beta^3}{27}}}$$

Tartaglia's formula for the roots of $x^3 - x = 0$ is $\frac{1}{\sqrt{3}}\left((\sqrt{-1})^{\frac{1}{3}} + \frac{1}{(\sqrt{-1})^{\frac{1}{3}}}\right)$.

◦ In 1572, in his *Algebra*, while solving for $x^3 - 15x - 4 = 0$, R. Bombelli arrived at $\left(2 + \sqrt{-1}\right) + \left(2 - \sqrt{-1}\right) = 4$ and introduced the symbol $\sqrt{-1}$.

◦ In 1673 John Wallis realised that the general solution for the form $x^2 + 2bx + c^2 = 0$ is

$$x = -b \pm \sqrt{b^2 - c^2}$$

Appendix 3: Derivatives of a multivariate function

$$f(\mathbf{x}) = f(x_1, \ldots, x_N)$$

$$\text{Gradient } \nabla_x f(\mathbf{x})) = \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} \\ \frac{\partial f(\mathbf{x})}{\partial x_2} \\ \vdots \\ \frac{\partial f(\mathbf{x})}{\partial x_N} \end{bmatrix} = \mathbf{0} \text{ and the Hessian matrix } \mathbf{H}_x > \mathbf{0}.$$

where the elements of the Hessian matrix are $\{H_x\}_{i,j} = \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j}$

Theorem: If $f(\mathbf{z}, \mathbf{z}^*)$ is a real-valued function of the complex vectors $\mathbf{z}$ and $\mathbf{z}^*$, the vector pointing in the direction of the maximum rate of change of $f(\mathbf{z}, \mathbf{z}^*)$ is $\nabla_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*)$, the derivative of $f(\mathbf{z}, \mathbf{z}^*)$ wrt $\mathbf{z}^*$. [Hayes 1996].

Thus, the turning points of $f(\mathbf{z}, \mathbf{z}^*)$ are solutions to $\nabla_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*) = \mathbf{0}$,

$$\text{where } \nabla_{\mathbf{z}^*} = \frac{1}{2} \begin{bmatrix} \frac{\partial}{\partial x_1} + \jmath \frac{\partial}{\partial y_1} \\ \vdots \\ \frac{\partial}{\partial x_n} + \jmath \frac{\partial}{\partial y_n} \end{bmatrix} \quad , \nabla_{\mathbf{z}} \mathbf{a}^H \mathbf{z} = \mathbf{a}^*, \quad , \nabla_{\mathbf{z}^*} \mathbf{a}^H \mathbf{z} = \mathbf{0}$$

Appendix 4: Some useful examples from $\mathbb{CR}$-calculus

For proofs see lecture supplement

$$\text{Linear Form:} \quad \frac{\partial}{\partial \mathbf{x}^*}\left\{\mathbf{x}^T\mathbf{a}\right\} = \mathbf{0}$$

$$\text{Linear Form:} \quad \frac{\partial}{\partial \mathbf{x}^*}\left\{\mathbf{x}^H\mathbf{a}\right\} = \mathbf{a}$$

$$\text{Quadratic Form:} \quad \frac{\partial}{\partial \mathbf{x}^*}\left\{\mathbf{x}^H\mathbf{C}\mathbf{x}\right\} = \mathbf{C}\mathbf{x}$$

$$\text{Quadratic Form:} \quad \frac{\partial}{\partial \mathbf{x}^*}\left\{\mathbf{x}^T\mathbf{C}\mathbf{x}^*\right\} = \mathbf{C}^T\mathbf{x}$$

$$\text{Vector Form:} \quad \mathbf{y} = \mathbf{A}\mathbf{x}, \ \frac{\partial \mathbf{y}^H}{\partial \mathbf{x}^*} = \mathbf{A}^H$$

Appendix 4: Some useful examples from $\mathbb{CR}$-calculus

Chain Rule

$$\text{Linear Form:} \quad \frac{\partial}{\partial \mathbf{z}^*}\left\{\mathbf{x}^H\mathbf{a}\right\} = \frac{\partial \mathbf{x}^H}{\partial \mathbf{z}^*}\mathbf{a} + \frac{\partial \mathbf{a}^T}{\partial \mathbf{z}^*}\mathbf{x}^*$$

$$\text{Quadratic Form:} \quad \frac{\partial}{\partial \mathbf{z}^*}\left\{\mathbf{x}^H\mathbf{C}\mathbf{x}\right\} = \frac{\partial \mathbf{x}^H}{\partial \mathbf{z}^*}\mathbf{C}\mathbf{x} + \frac{\partial \mathbf{x}^T}{\partial \mathbf{z}^*}\mathbf{C}^T\mathbf{x}^*$$

$$\text{Vector Form:} \quad \mathbf{y} = \mathbf{A}\mathbf{x}, \quad \frac{\partial \mathbf{y}^H}{\partial \mathbf{z}^*} = \frac{\partial \mathbf{x}^H}{\partial \mathbf{z}^*}\mathbf{A}^H, \quad \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}^*} = \frac{\partial \mathbf{x}^T}{\partial \mathbf{z}^*}\mathbf{A}^T$$

Matrix Derivatives

$$\text{Linear Form:} \quad \frac{\partial}{\partial \mathbf{B}^*}\left\{\mathrm{Tr}\,\mathbf{B}^*\mathbf{C}\right\} = \mathbf{C}^T$$

$$\text{Quadratic Form:} \quad \frac{\partial}{\partial \mathbf{A}^*}\left\{\mathrm{Tr}\,\mathbf{A}\mathbf{C}\mathbf{A}^H\right\} = \mathbf{A}\mathbf{C}$$

Appendix 6: CR calculus and learning alg. (more later)

The derivative of the cost function $\frac{1}{2}e(k)e^*(k)$ and CLMS

As $\mathbb{C}$-derivatives are not defined for real functions of complex variable

$$\mathbb{R}-\text{der:}\quad \frac{\partial}{\partial \mathbf{z}}=\frac{1}{2}\left[\frac{\partial}{\partial \mathbf{x}}-\jmath\frac{\partial}{\partial \mathbf{y}}\right] \qquad \mathbb{R}^*-\text{der:}\quad \frac{\partial}{\partial \mathbf{z}^*}=\frac{1}{2}\left[\frac{\partial}{\partial \mathbf{x}}+\jmath\frac{\partial}{\partial \mathbf{y}}\right]$$

and the gradient

$$\nabla_{\mathbf{w}}J=\frac{\partial J(e,e^*)}{\partial \mathbf{w}}=\left[\frac{\partial J(e,e^*)}{\partial w_1},\ldots,\frac{\partial J(e,e^*)}{\partial w_N}\right]^T=2\frac{\partial J}{\partial \mathbf{w}^*}=\underbrace{\frac{\partial J}{\partial \mathbf{w}^r}+\jmath\frac{\partial J}{\partial \mathbf{w}^i}}_{pseudogradient}$$

The standard Complex Least Mean Square (CLMS) (Widrow *et al.* 1975)

$$\begin{aligned} y(k) &= \mathbf{w}^H(k)\mathbf{x}(k) \\ e(k)=d(k)-\mathbf{w}^H(k)\mathbf{x}(k) &\qquad e^*(k)=d^*(k)-\mathbf{x}^H(k)\mathbf{w}(k) \\ &\text{and}\quad \nabla_{\mathbf{w}}J=\nabla_{\mathbf{w}^*}J \\ \mathbf{w}(k+1) &= \mathbf{w}(k)-\mu\frac{\partial \frac{1}{2}e(k)e^*(k)}{\partial \mathbf{w}^*(k)}=\mathbf{w}(k)+\mu e^*(k)\mathbf{x}(k) \end{aligned}$$

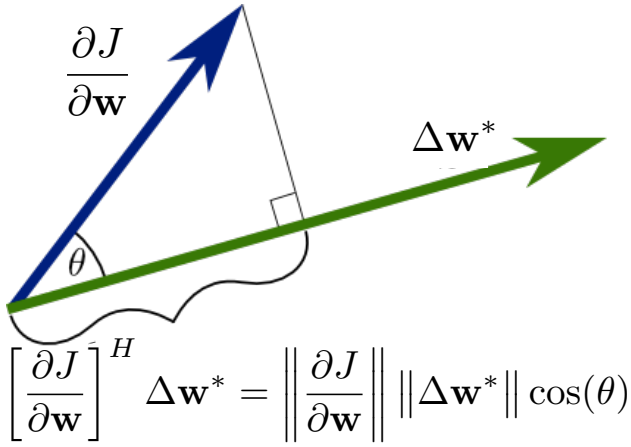
Thus, no tedious computations $\rightsquigarrow$ the CLMS is derived in one line.

App 7: Stochastic gradient optimis. $\hookrightarrow$ complex gradient

Cost function $J(e, e^*) = |e|^2 = ee^*$, where $e(k) = d(k) - \mathbf{w}^H(k)\mathbf{x}(k)$

Gradient: $\nabla_{\mathbf{w}} J = \dfrac{\partial J}{\partial \mathbf{w}} = \left[\dfrac{\partial J}{\partial w_1}, \ldots, \dfrac{\partial J}{\partial w_N}\right]^T$

For the minima : $\dfrac{\partial J}{\partial \mathbf{w}} = \mathbf{0}$ and $\dfrac{\partial J}{\partial \mathbf{w}^*} = \mathbf{0}$



The first term of Taylor series expansion becomes (since $J(e, e^*)$ is real):

$$\Delta J(e, e^*) = \left[\frac{\partial J}{\partial \mathbf{w}}\right]^T \Delta \mathbf{w} + \left[\frac{\partial J}{\partial \mathbf{w}^*}\right]^T \Delta \mathbf{w}^* = 2\Re\left\{\left[\frac{\partial J}{\partial \mathbf{w}}\right]^H \Delta \mathbf{w}^*\right\} = 2\Re\left\{\left[\frac{\partial J}{\partial \mathbf{w}^*}\right]^T \Delta \mathbf{w}^*\right\}$$

Therefore, the scalar product

$$< \partial J/\partial \mathbf{w}, \Delta \mathbf{w}^* > = \left[\frac{\partial J}{\partial \mathbf{w}}\right]^H \Delta \mathbf{w}^* = \| \partial J/\partial \mathbf{w} \| \| \Delta \mathbf{w}^* \| \cos \angle(\partial J/\partial \mathbf{w}, \Delta \mathbf{w}^*)$$

achieves its maximum value when $\frac{\partial J}{\partial \mathbf{w}} \parallel \Delta \mathbf{w}^*$, that is, for $\nabla_{\mathbf{w}} J = \nabla_{\mathbf{w}^*} J$.

The maximum change of the gradient of the cost function is in the direction of the conjugate weight vector (R^*-derivative) $\rightsquigarrow$ equivalent to pseudogradient .

Appendix 8: Performance advantage of widely linear modelling in $\mathbb{C}$

The MSE of the widely linear and the strictly linear estimator of a variable y are respectively given by

$$E[|e_{\text{WL}}|^2] = E[|y - y_{\text{WL}}|^2] = E[|y|^2] - (\mathbf{h}^H\mathbf{c} + \mathbf{g}^H\mathbf{p}^*)$$
$$E[|e_{\text{SL}}|^2] = E[|y - y_{\text{SL}}|^2] = E[|y|^2] - \mathbf{c}^H\mathcal{C}^{-1}\mathbf{c}$$

The performance advantage of using the widely linear model in $\mathbb{C}$ is then

$$\triangle\text{MSE} = E[|e_{\text{SL}}|^2] - E[|e_{\text{WL}}|^2] = \underbrace{[\mathbf{p} - \mathcal{P}\mathcal{C}^{-*}\mathbf{c}^*]^H[\mathcal{C} - \mathcal{P}\mathcal{C}^{-*}\mathcal{P}^*]^{-1}[\mathbf{p} - \mathcal{P}\mathcal{C}^{-*}\mathbf{c}^*]}$$

Is this term always nonnegative?

A joint diagonalisation of $\mathcal{C}$ and $\mathcal{P}$ can be achieved by using the strong uncorrelating transform to give $\mathcal{C} = \mathcal{Q}\mathbf{I}\mathcal{Q}^H$ and $\mathcal{P} = \mathcal{Q}\mathbf{\Lambda}\mathcal{Q}^T$, where $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$ and $1 \geq \lambda_1 \geq \cdots \geq \lambda_N \geq 0$. Upon such joint diagonalisation, we have

$$\triangle\text{MSE} = \underbrace{[\mathcal{Q}^{-1}\mathbf{p} - \mathbf{\Lambda}\mathcal{Q}^{-*}\mathbf{c}^*]^H}_{\mathbf{b}^H} \frac{\mathbf{I}}{\mathbf{I} - \mathbf{\Lambda}^2} \underbrace{[\mathcal{Q}^{-1}\mathbf{p} - \mathbf{\Lambda}\mathcal{Q}^{-*}\mathbf{c}^*]}_{\mathbf{b}} = \sum_{n=1}^{N} \frac{|b_n|^2}{1 - \lambda_n^2} \geq 0$$

Appendix 9: Data model $\not\hookrightarrow$ Gaussianity

starting from real-valued data

Why Gaussian? **Justification: Central Limit Theorem**

If we form a sum of independent measurements

$\Rightarrow$ the distribution of the sum tends to a Gaussian distribution

$$p(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} e^{-\frac{(x-\mu_x)^2}{2\sigma_x^2}} \qquad x \sim \mathcal{N}(\mu_x, \sigma_x^2)$$

$\Rightarrow$ **distribution defined by its mean and variance!!!**

If $x \sim \mathcal{N}(0, \sigma_x^2)$ then $E\{x^{2n-1}\} = 1, 3, \ldots, (2n-1)\sigma_x^{2n}, \quad \forall n$

In the vector case (N Gaussian random variables)

$$p\big(x[0], x[1], \ldots, x[N-1]\big) = \frac{1}{(2\pi)^{N/2} det(\mathbf{C}_{xx})^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_x)^T \mathbf{C}_{xx}^{-1}(\mathbf{x}-\boldsymbol{\mu}_x)}$$

where $\mathbf{C}_{xx} = E\{(\mathbf{x}-\boldsymbol{\mu}_x)(\mathbf{x}-\boldsymbol{\mu}_x)^T\}$ is the **covariance matrix.**

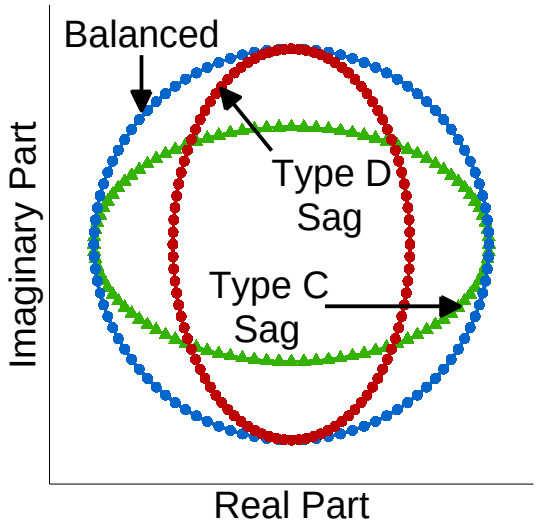
App. 10: Does degree of circularity influence estimation in $\mathbb{C}$?

Voltage sag: A magnitude and/or phase imbalance

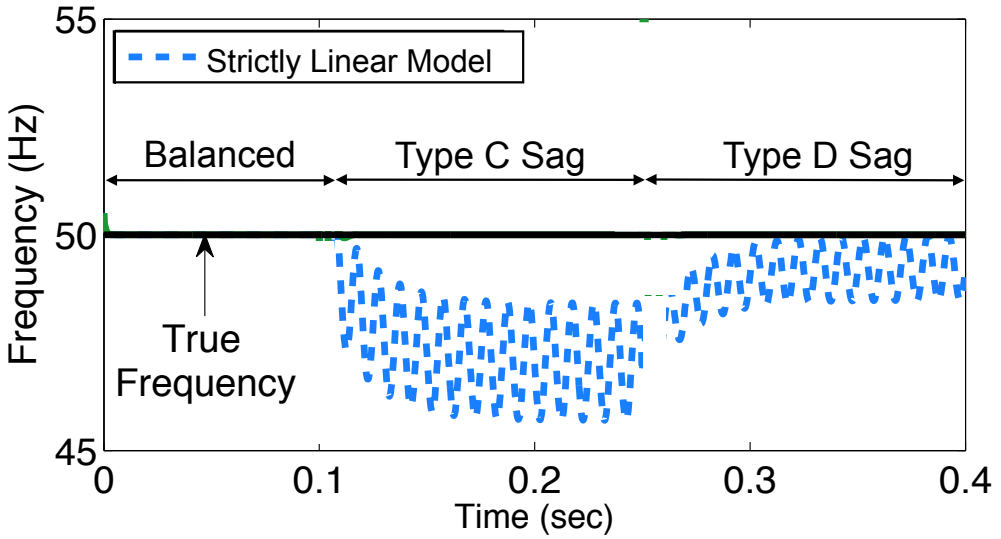
- For balanced systems, $v(k) = A(k)e^{\jmath\omega k\Delta T} \rightarrow$ circular trajectory.
- Unbalanced systems, $v(k) = A(k)e^{\jmath\omega k\Delta T} + B(k)\mathbf{e}^{-\jmath\omega\mathbf{k}\boldsymbol{\Delta}\mathbf{T}}$ are influenced by the "conjugate" component.

↬ **We need the complex conjugate of the signal too.**

Circularity Diagram



The strictly linear model, $\hat{v} = f(v)$, yields biased estimates when the system is unbalanced



Notes:

○

Notes:

○

Notes:

○

Notes:

○