MODEL IDENTIFICATION & DATA ANALYSIS

PART 1

by Yannick Giovanakis

April 20, 2018

Contents

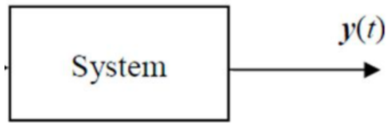
0 Introduction

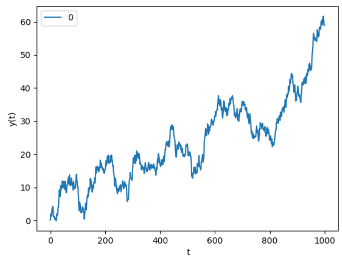
The course will deal with two types of situations :

1. Analysis and modelling of **Time-Series**
2. Analysis and modelling of **Input/Output Systems**

0.1 Time-Series

Time series consider vectors $\{y(1), y(2), ..., y(N)\}$ of **measured data** of cardinality N (large , 1000 - 10000).
Said vectors are considered in the **time-domain** : y(t) is a signal or **stochastic process** generated by the system whose output is than sampled.





0.1.1 TS Applications

TS are used for two problems :

1. **Prediction problem** : $\{y(1)...y(N)\} \rightarrow \hat{y}(N + K|N)$
   Given N measurements **estimate** the measurement K timesteps ahead
2. **Filtering problem** : $\{x_1(t)...x_N(t)\} \rightarrow \hat{x}(t|t)$
   Where $\{x_1(t)...x_N(t)\}$ are internal variables of the system

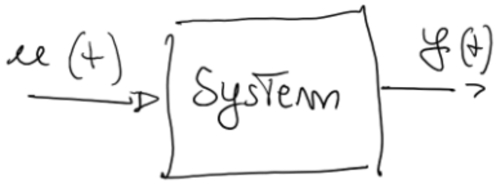
0.2 I/O Systems

I/O systems consider two measurements :

- **Input** : $\{u(1)...u(N)\}$
- **Output**: $\{y(1)...y(N)\}$

Resulting in two signals u(t) and y(t). Input signal $u(t)$ can be of two types:
–**Controllable** : can be affected (ex : voltage)

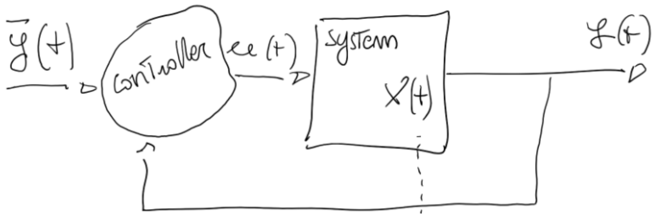
–**Uncontrollable** : cannot be affected (ex : rain)



0.2.1 I/O Applications

I/O systems are used for three problems :

1. **Prediction problem** : $\{y(1)...y(N)\} \rightarrow \hat{y}(N+K|N)$
   Given N measurements **estimate** the measurement K timesteps ahead
2. **Filtering problem** : $\{x_1(t)...x_N(t)\} \rightarrow \hat{x}(t|t)$
   Where $\{x_1(t)...x_N(t)\}$ are internal variables of the system
3. **System control problem** : given a desired output $\bar{y}(t)$, control $u(t)$ so that $y(t)$ is as close as possible to $\bar{y}(t)$



0.3 Time Series vs I/O Systems

In prediction and filtering problems both I/O systems and TS can be used. How to chose which one to use?

Ex.1

- **System** : Electric Motor
- **Input** : Current , temperature of motor,electromagnetic fields nearby..
- **Output** : Torque

We can say that our main input variable (current) is responsible for 90% of the output.The other variables only have slight effects on the torque so they are considered **noise**

The best model to choose is the **I/O**

Ex.1

- **System** : Macro-Economic System
- **Input** : Too many
- **Output** : Stock prices of FCA

There are many thousand variables affecting the output. Listing and measuring them all would make the model too complex . In this case all the input variables are considered **noise** : the best model to choose is the **Time Series**

Ex.3

- **System** : Environment
- **Input** : Rain, wind, heatings, cars , temperature, pressure...
- **Output** : PM10 levels

In this case some main inputs variables can be selected (ex :cars , heating and rain) while the others are modelled as noise. In this case **I/O** model should be used.

It is not wrong to consider all the inputs as noise and model the problem as **Time Series**.

General rule:

	Advantages
TS	Only y(t) must be measured
I/O	Better estimation

0.4 Modelling structures

Depending on the problem 2 modelling structures are used.
The TS are modelled with a **mathematical model** which outputs signal $y(t)$. An **imaginary** input $e(t)$ called **white noise** is considered as **standard input** and it is **part of the model**.
The I/O system is modelled by two **mathematical models** which output signal $y(t)$. As above **white noise** is considered as input of one of the two models. The other model has input $u(t)$ which is **not** part of the model.

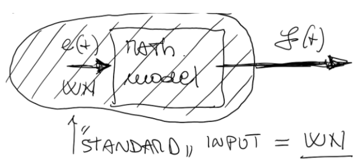


Figure 1: TS Model

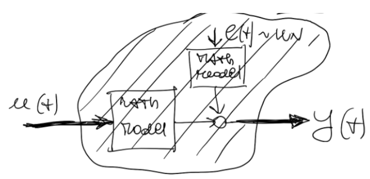


Figure 2: IO Model

All signals and systems are **time-discrete** . Analogue signals are converted to digital signals through **ADCs** .
Discrete time points are spaced evenly at pace ΔT = sampling time

0.5 Mathematical Models

The mathematical models used to elaborate output functions are either **white boxes** or **black - boxes**.

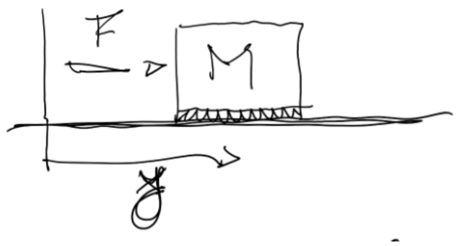


Figure 3: System to be modelled

0.5.1 White Box Models

Also called *first-principles models* assume that the parameters involved in the system are known and well defined. Using white box models we get a **physical interpretation of the model** which makes them useful if the aim is to design the system.
In the example we can derive laws that define our system's **transfer function** given as input a force $\vec{F}$ and output y :

$$\ddot{y}M = F - c\dot{y} \rightarrow Laplace \rightarrow s^2My = F - scy$$

$$(s^2M + sc)y = F$$
$$y = \frac{1}{s^2M + sc}F$$

0.5.2 Black Box Models

In black box models we don't know the internal parameters that influence the system. In our example , we only know that by changing the input $\vec{F}$ a corresponding change in output $y(t)$ can be measured . By measuring the data we can derive a model :

$$y(t) = \frac{b_0Z^2 + b_1Z + b_2}{a_0Z^2 + a_1Z + a_2}F(t)$$

where $a_0, ..., a_2, b_0, ..., b_2$ are the parameters.

0.5.3 White box vs Black box

Table 1: WB/BB Comparison

White Box	Black Box
-Get physical interpretation of the model and its parameters. -Useful for designing the system	-Very fast -Very accurate -Does not require know-how of the domain -Can be easily re-tuned

0.6 Stochastic Processes

Random variable RV:

$v(s)$ is completely defined by its probability distribution (Gaussian, Uniform...) which is related to its **probability density function** (PDF)

Stochastic Process:

is a sequence of **time-ordered random variables** defined at the same experiment S

$$v(1, S), v(2, S), ..., v(t, S)$$

where t is the time index. If the experiment is **fixed** $S = \bar{S}$, we get an instance , a **realisation** of the stochastic process :

$$v(1, \bar{S}), ..., v(t, \bar{S})$$

resulting in a set of samples $\{y(1), ..., y(N)\} = \{y(1, \bar{S}), ..., y(N, \bar{S})\}$

0.6.1 Characteristics

Mean value m(t):

expected value of a random variable v(t,S) at time t

$$m(t) = E[v(t, S)]$$

Covariance Function $\gamma(t_1, t_2)$:

expected value of the **product** of two **unbiased** random variables at time instants t1, t2 :

$$\gamma(t_1, t_2) : E[(v(t_1, S) - m(t_1))(v(t_2, S) - m(t_2))]$$

Removing the mean brings the signal closer to 0.
If t1 = t2 = t the covariance degenerates in **variance**:

$$\gamma(t) = E[(v(t, S) - m(t))^2]$$

0.6.2 Stationary Stochastic Processes

Has properties:

1. $m(t) = m, \forall t$

2. $\gamma(t_1, t_2)$ depends on $\tau = |t_1 - t_2|$
   This means that the covariance depends on the **distance in time** and not on specific considered samples.
   $\gamma(t_1, t_2) = \gamma(t_3, t_4) \rightarrow |t_1 - t_2| = |t_3 - t_4|$
   $\gamma(\tau) = E[(v(t) - m)(v(t - \tau) - m)]$ has properties :

   - $\gamma(0) = E[(v(t) - m)^2] \rightarrow$ **variance**
   - $|\gamma(\tau)| \leq \gamma(0)$
   - $\gamma(\tau) = \gamma(-\tau)$

SSP Equivalence

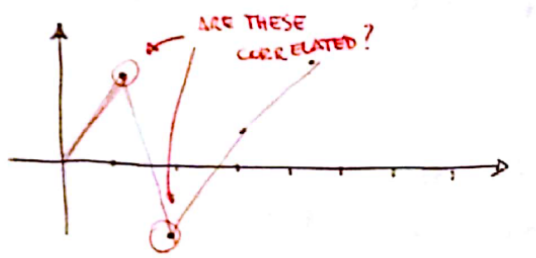
Two SSPs $y_1(t), y_2(t)$ are equivalent in a **weak sense** if:

- $m_{y1} = m_{y2}$
- $\gamma_{y1}(\tau) = \gamma_{y2}(\tau)$,$\forall \tau$

Correlation Function

If the $m = 0$ the $\gamma(\tau)$ function degenerates in the **correlation function** :

$$E[v(t)v(t-\tau)]$$



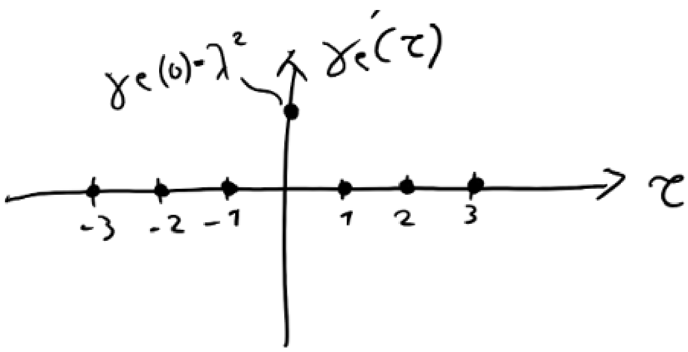
0.6.3 White Noise

e(t) is SSP called **white noise** and is written as

$$e(t) \rightarrow WN(\mu, \lambda^2)$$

Properties:

- Mean value : $E[e(t)] = \mu, \forall t$
- Variance : $\gamma_e(0) = E[(e(t) - \mu)^2] = \lambda^2$
- Covariance : $E[(e(t) - \mu)(e(t-\tau) - \mu)] = 0$, $\forall t, \forall \tau \neq 0$f



No covariance means that the samples are **not related**
Considering a **Gaussian Distribution** : $e(t) \rightarrow WGN(\mu, \lambda^2)$

0.7 Sample estimation of mean and covariance function

Dealing with samples it is useful to **estimate** the mean and covariance of the samples.
Output y(t) is a SSP : $\{y(1), ..., y(N)\}$ a particular realisation of $\bar{S}$ with :

- Mean $m = E[y(t)]$
- Covariance $\gamma(\tau) = E[(y(t) - m)(y(t - \tau) - m)]$

This seems trivial but the computation of the expected value cannot be done because the **distribution of the process** is **unkown**.
These two can be **estimated**

0.7.1 Sample Mean

The sample mean is a good estimator for the mean m :

$$\hat{m}_n = \frac{1}{N} \sum_{t=1}^{N} y(t)$$

Properties of the estimator :

1. $\hat{m}_n$ is **correct** if $E[\hat{m}_n] = m$
   Proof: $E[\hat{m}_n] = E[\frac{1}{N} \sum_{t=1}^{N} y(t, s)] = \frac{1}{N} \sum_{t=1}^{N} E[y(t, s)] = \frac{1}{N} \sum_{t=1}^{N} m =$**m**

   Example
   $y(t, S) = \bar{v}(s) \to WN(0, 1)$ and $S = \bar{S}, \{y(1, \bar{S}), ..., y(N, \bar{S}\}$ so :
   -$\hat{m}_n = \frac{1}{N} \sum_{t=1}^{N} y(t, \bar{S}) = \frac{1}{N} \sum_{t=1}^{N} \bar{v}(\bar{S}) = \frac{1}{N} N\bar{v}(\bar{S}) \neq 0 \to$ bad estimator
   -$\breve{m}_n = \frac{1}{N} \sum_{S=1}^{N} y(\bar{t}, S) = \frac{1}{N} v(S) \to 0 \to$ good estimator

2. $\hat{m}_n$ is **consistent** if $E[(\hat{m}_n - m)^2] \xrightarrow[N \to \infty]{} 0$
   The **error variance** approaches 0 for large values of N: this means that with a lot of data $N \to \infty$ we can estimate $\hat{m}_n$ more effectively.
   In general one can say that $\hat{m}_n$ is consistent if $\gamma(\tau) \xrightarrow[|\tau| \to \infty]{} 0$

Example

$y(t,S) = \bar{V}(S) \rightarrow WN(0,1)$
$\gamma(\tau) = E[(\gamma(\tau))(\gamma(t-\tau))] = E[\bar{V}(S)\bar{V}(S)] = E[\bar{V}(S)^2] = 1$

0.7.2 Sample Covariance

y(t) is a SSP with **zero mean**.
A good estimator for for the covariance is the **sample covariance**:

$$\hat{\gamma}_N(\tau) = \frac{1}{N-\tau}\sum_{t=1}^{N-\tau} y(t)y(t+\tau)$$

$$0 \leq \tau \leq N-1$$

It is important to notice that this approximation is good for $\tau << N$ because the accuracy of $\gamma_N(\tau)$ **decreases** with τ
Properties of the estimator :

1. $\hat{\gamma}_N(\tau)$ is **correct** if $E[\hat{\gamma}_N(\tau)] = \gamma(\tau)$

   Proof:
   $E[\hat{\gamma}_N(\tau)] = E[\frac{1}{N-\tau}\sum_{t=1}^{N-\tau} y(t)y(t+\tau)] = \frac{1}{N-\tau}\sum_{t=1}^{N-\tau} E[y(t)y(t+\tau)] = \frac{1}{N-\tau}\sum_{t=1}^{N-\tau}\gamma(\tau) = \gamma(\tau)$

2. $\hat{\gamma}_N(\tau)$ is **consistent** if $E[(\hat{\gamma}_N(\tau)-\gamma(\tau))^2] \xrightarrow[N\rightarrow\infty]{} 0$,
   True if $\gamma(\tau) \xrightarrow[|\tau|\rightarrow\infty]{} 0$

Observation 1:
$\hat{\gamma}_N(\tau) = \frac{1}{N-\tau}\sum_{t=1}^{N-\tau} y(t)y(t+\tau)$
$0 \leq \tau \leq N-1, \tau \geq 0$
but since y(t) is a SSP $\gamma(\tau) = \gamma(-\tau)$:

$$\hat{\gamma}_N(\tau) = \frac{1}{N-|\tau|}\sum_{t=1}^{N-|\tau|} y(t)y(t+|\tau|)$$

$$|\tau| \leq N-1$$

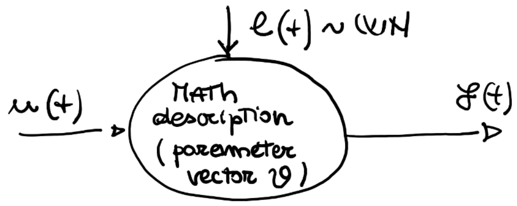
Observation 2:

$\hat{\gamma}'_N(\tau) = \frac{1}{N-|\tau|}\sum_{t=1}^{N-|\tau|} y(t)y(t+|\tau|) \rightarrow E[\hat{\gamma}'_N(\tau)] = ... = \frac{1}{N}\gamma(\tau)(N-|\tau|)$

As shown $\hat{\gamma}'_N(\tau)$ **doest not** satisfy the **correct** property.

However for N $\rightarrow \infty$ and $\tau << N$: $\hat{\gamma}'_N(\tau)$ is **asimptotically correct**

1 Chapter 1 : Model classes



$$\text{Mathematical model} = \begin{cases} u(t) & \text{input (I/O only)} \\ e(t) & \text{white noise} \\ y(t) & \text{output} \end{cases}$$

The mathematical model is described by **parametric parameter vector** θ that is found using a **parametric supervised** identification approach.
The models can be described with:

- **Differential Equations** in time domain
- **Transfer functions**

1.1 Time-Series model classes

The following processes are modelled with **differential equations**

1.1.1 Moving Average Models (MA)

A process y(t) **generated** by a WN e(t) is a moving average of order n **MA(n)** process if:

$$y(t) = c_0 e(t) + c_1 e(t-1) + ... + c_n e(t-n)$$

with parameter vector $\theta = \{c_0, ..., c_n\}$

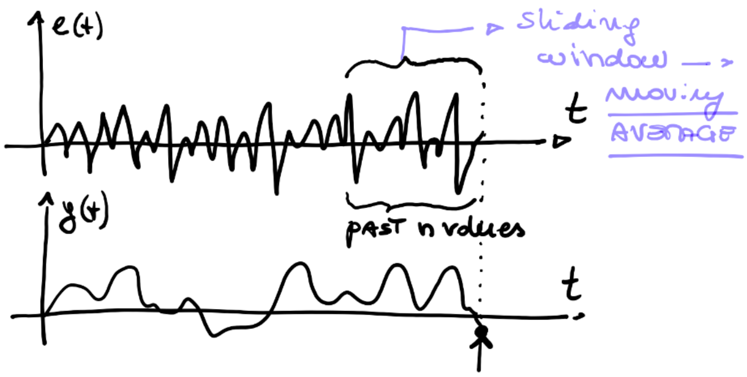


Figure 4: y(t) is linear combination of past n e(t) values

1.1.2 Autoregressive Models (AR)

A process y(t) **generated** by a WN e(t) is an autoregressive of order m **AR(m)** process if:

$$y(t) = a_1 y(t-1) + a_2 y(t-2) + ... + a_m y(t-m) + c_0 e(t)$$

with parameter vector $\theta = \{c_0, a_1, ..., a_m\}$

1.1.3 Autoregressive Moving Average Models (ARMA)

A process y(t) **generated** by a WN e(t) is an ARMA of order (n,m) **ARMA(n,m)** process if:

$$y(t) = a_1 y(t-1) + a_2 y(t-2) + ... + a_m y(t-m) + c_0 e(t) + c_1 e(t-1) + ... + c_n e(t-n)$$

with parameter vector $\theta = \{c_0, ..., c_n, a_1, ..., a_m\}$
ARMA(0,n) $\rightarrow$ MA(n) : MA(n) is **subclass** of ARMA
ARMA(m,0) $\rightarrow$ AR(m) : AR(m) is **subclass** of ARMA

1.2 Input/Output model classes

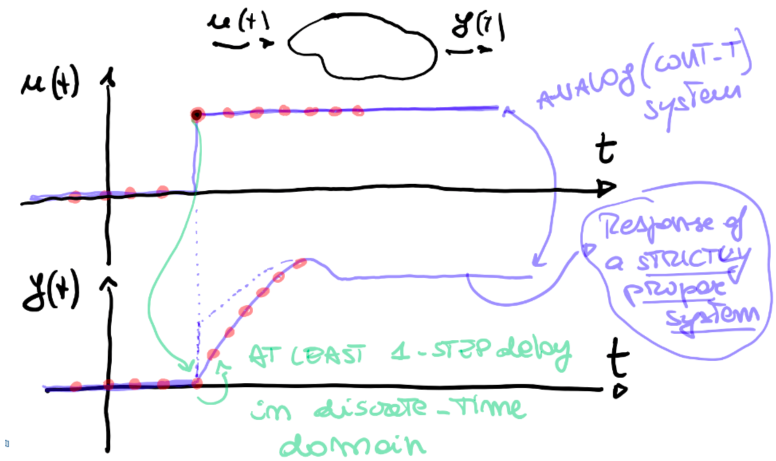
The following processes are modelled with **differential equations**

1.2.1 Autoregressive Moving Average Exogenous (ARMAX)

A process y(t) **generated** by a WN e(t) and **exogenous** signal u(t) is an ARMAX of order (n,m, p+k) process if:

$$y(t) = a_1 y(t-1) + ... + a_m y(t-m) + c_0 e(t) + ... + c_n e(t-n) + b_0 u(t-k) + ... + b_p u(t-k-p)$$

with parameter vector $\theta = \{c_0, ..., c_n, a_1, ..., a_m, b_0, ..., b_p\}$
$K \geq 1$ plays an important role : it represents the pure/intrinsic **delay** between y(t) and u(t). If u(t) is a step the corresponding output y(t) is shown in figure.



Sampling (red dots) gives a discrete approximation : when the input slope rises a sample is taken resulting in a high value. The corresponding output is still low : this causes a **1 step delay**

Example

$y(t) = \frac{1}{2}y(t-1) + \frac{1}{3}y(t-2) + e(t) + e(t-3) + u(t-2) + \frac{1}{2}u(t-4)$ The process is an ARMAX (2,3,2+2)

Observation : missing values as above can be present!

Remark

Armax models are the most general class models for **dynamic ,linear, time-invariant** systems.
Non-Linear N-ARMAX $y(t) = f(y(t-1), .., y(t-m), e(t), ..., e(t-n), u(t-k), ..., u(t-k-p)$
depend on **non-linear functions** : polynomials , splines ,NN ,Radial Basis Functions ,Fuzzy Sets.

1.3 Transfer function representation

The four models found above can be represented using **transfer functions**. To transform time domain equations into the equivalent transfer function representation the **Z operator** is introduced.

1.3.1 Z Operator

- The operator Z^{-1} is the **backward shift** operator :

$$Z^{-1}x(t) = x(t-1)$$

- The operator Z^{+1} is the **forward shift** operator :

$$Z^{+1}x(t) = x(t+1)$$

Both operators have properties :

- **Linearity** : $Z^{-1}(ax(t)+by(t)) = Z^{-1}ax(t)+Z^{-1}by(t) = ax(t-1)+by(t-1)$
- **Recursion** : $Z^{-1}(Z^{-1}...(Z^{-1}x(t))) = x(t-n) = Z^{-n}$

1.3.2 Time domain to Transfer Function

The Z operators are used to shift the equations of the time domain to be all at time **t**.
In case of a generic **ARMAX(m,n,p+k)** process

$y(t) = a_1y(t-1)+...+a_my(t-m)+c_0e(t)+...+c_ne(t-n)+b_0u(t-k)+...+b_pu(t-k-p)$

Applying the Z^{-1} operator:

$y(t) = a_1 Z^{-1} y(t) + ... + a_m Z^{-m} y(t) + c_0 e(t) + ... + c_n Z^{-n} e(t) + b_0 Z^{-k} u(t) + ... + b_p Z^{-k-p} u(t)$

Collecting :

$y(t)[1 - a_1 Z^{-1} + ... + a_m Z^{-m}] = [c_0 e + ... + c_n Z^{-n}] e(t) + [b_0 Z^{-k} + ... + b_p Z^{-k-p}] u(t)$

Dividing :

$$y(t) = \frac{[c_0 e + ... + c_n Z^{-n}]}{[1 - a_1 Z^{-1} + ... + a_m Z^{-m}]} e(t) + \frac{[b_0 + ... + b_p Z^{-p}]}{[1 - a_1 Z^{-1} + ... + a_m Z^{-m}]} u(t) Z^{-k}$$

Defining :

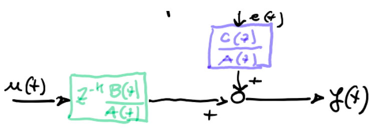
$$A(Z) = 1 - a_1 Z^{-1} + ... + a_m Z^{-m}$$

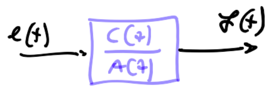
$$B(Z) = b_0 + ... + b_p Z^{-p}$$

$$C(Z) = c_0 e + ... + c_n Z^{-n}$$

The resulting process using TF representation is :

$$y(t) = \frac{C(Z)}{A(Z)} e(t) + \frac{B(Z)}{A(Z)} u(t) Z^{-k}$$





1.3.3 From Z^- to Z^+

The transfer functions can be written in negative, positive o mixed power of Z. The example explains how to get the positive power representation starting from a negative one :

$$y(t) = \frac{c_0 + c_1 Z^{-1} + ... + c_n Z^n}{1 - a_1 Z^{-1} - ... - a_m Z^{-m}} e(t)$$

If $m \geq n$ by multiplying by Z^{+m} :

$$y(t) = \frac{c_0 Z^m + c_1 Z^{m-1} + ... + c_n Z^{m-n}}{Z^m - a_1 Z^{m-1} - ... - a_m} e(t)$$

Observation

Even if feasible and correct it is better to **avoid** the mixed representation!

1.3.4 Importance of stationary property

Transformation **Time Domain** ↔ **Transfer Functions** are **feasible** if the **stationary property** holds because otherwise the Z operator is not applicable.

$$y(t) = \frac{Z + \frac{1}{2}}{Z - \frac{1}{3}} e(t), \text{e(t) - WN(0,1)}$$

In time domain

$$(Z - \frac{1}{3})y(t) = (Z + \frac{1}{2})e(t)$$
$$y(t+1) - \frac{1}{3}y(t) = e(t+1) + \frac{1}{2}e(t)$$
$$y(t+1) = \frac{1}{3}y(t) + e(t+1) + \frac{1}{2}e(t)$$

Time shift to start a time "t" (can be done in stationary conditions):

$$y(t) = \frac{1}{3}y(t-1) + e(t) + \frac{1}{2}e(t-1)$$

Back to TF

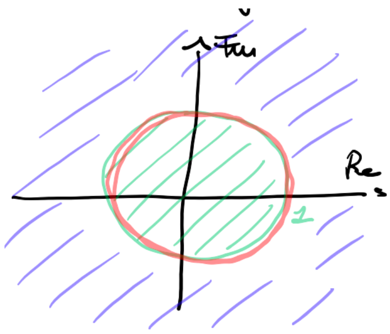
$$y(t) = \frac{1}{3}Z^{-1}y(t) + e(t) + \frac{1}{2}Z^{-1}e(t)$$
$$[1 - \frac{1}{3}Z^{-1}]y(t) = [1 + \frac{1}{2}Z^{-1}]e(t)$$
$$y(t) = \frac{[1 + \frac{1}{2}Z^{-1}]}{[1 - \frac{1}{3}Z^{-1}]}e(t)$$

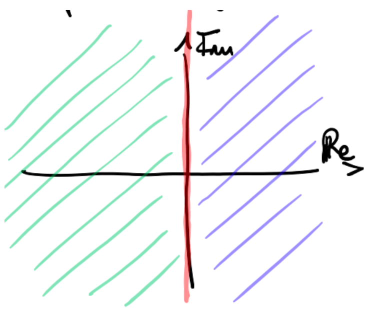
1.3.5 Pole,Zeros and Stability

Considering a process with signals e(t) , y(t) and a system W(Z) represented in **positive/null** power:

- **Poles** of W(Z) are the **roots** of the denominator
- **Zeros** of W(Z) are the **roots** of the nominator

A system is said to be **asymptotically stable** if and only if all the **poles** of W(Z) are **strictly inside** the unit circle (left graph).
Blue = unstable region
Red = simple stability region
Green = asymptotically stability region





Note: if we were dealing with **continuous** signals and processes instead of Z transformation we would apply **Laplace** . Also the stability region would change as seen on the right graph.

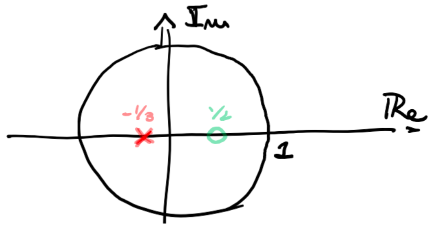
Example:

$$W(Z) = \frac{1 - \frac{1}{2}Z^{-1}}{1 + \frac{1}{3}Z^{-1}}$$

Move to positive power:

$$W(Z) = \frac{Z - \frac{1}{2}}{Z + \frac{1}{3}}$$

- **Pole** : $Z = -\frac{1}{3}$
- **Zero** : $Z = \frac{1}{2}$



The system is asymptotically stable since all poles are within the unit circle.

1.3.6 Stationary property and stability

In a stochastic process y(t) obtained as output of a system W(Z) fed with a stochastic process v(t) , y(t) is a **stationary process SSP** if and only if:

1. v(t) is a **stationary stochastic process**
2. W(Z) is **asymptotically stable**

Checking the stationary property is usually very long , instead these two properties make it easy : input v(t) is usually a **white noise** which is a **stationary stochastic process**.

1.3.7 Poles and Zeros in MA & AR processes

MA(1)

$$y(t) = e(t) + \frac{1}{2}e(t-1)$$

$$y(t) = (1 + \frac{1}{2}Z^{-1})e(t)$$

$$y(t) = (\frac{Z + \frac{1}{2}}{Z})e(t)$$

- **Zero** : $Z = -\frac{1}{2}$
- **Pole** : $Z = 0$

AR(1)

$$y(t) = \frac{1}{2}y(t-1) + 3e(t)$$

$$y(t) = (\frac{3Z}{Z - \frac{1}{2}})e(t)$$

- **Zero** : $Z = 0$
- **Pole** : $Z = \frac{1}{2}$

General conclusion

A **MA(n)** process is generated by a TF having :

- n **generic** zeros
- n poles **all in 0** $\rightarrow$ **always stationary!**

It is also called **All-Zeros** process

An **AR(m)** process i s generated by a TF having :

- m zero **all in 0**
- m **generic** poles It is also called **All-Poles** process

2 Chapter 2 : Analysis of Stochastic Processes

TS modelled with ARMA processes and I/O modelled with ARMAX models can be represented in 4 different ways :

- Time domain (Chap.1)
- Transfer function (Chap.1)
- **Probabilistic representation**
- **Frequency representation**

2.1 Probabilistic Representation

2.1.1 Probabilistic representation of MA(n)

Time domain representation : $y(t) = c_0 e(t) + ... + c_n e(t-n), e(t) \sim WN(0, \lambda^2)$.
The process is **stationary** as all the poles are in the origin.

- **Mean of y**

  $$m_y = E[y(t)] = E[c_o e(t) + ... + c_n e(t-n)] = c_0 E[e(t)] + ... + c_n E[e(t-n)]$$

  Because of stationary property $E[e(t)] = ... = E[e(t-n)] = 0$

  $$\boxed{m_y = 0}$$

- **Covariance of y**

  - $\tau = 0$
    $\gamma_y(0) = E[(y(t) - m_y)^2] = E[y(t)^2] = E[(c_0 e(t) + ... + c_n e(t-n))^2] = c_0^2 E[e(t)^2] + ... + c_n^2 E[e(t-n)^2] + 2c_0 c_1 E[e(t)e(t-1) + ... + 2c_{n-1} c_n E[e(t-n-1)e(t-n)]$
    where
    $E[e(t)^2] = E[e(t-1)^2] = ... = E[e(t-n)^2] = \lambda^2$
    $E[e(t)e(t-1)]... = 0$ because not correlated

    $$\boxed{\gamma_y(0) = \lambda^2 (c_0^2 + ... + c_n^2)}$$

- $\tau = 1$
  $\gamma_y(1) = E[(y(t) - m_y)(y(t-1) - m_y)] = E[y(t)y(t-1)]$
  $E[(c_0 e(t) + ... + e_n e(t-n))(c_0 e(t-1) + ... + c_n e(t-n-1)]$
  only terms at same time survive :
  $c_0 c_1 E[e(t-1)^2] + ... + c_{n-1} c_n E[e(t-n)^2]$
  where $E[e(t-i)^2] = \lambda^2$

$$\boxed{\gamma_y(1) = (c_0 c_1 + c_1 c_2 + ... + c_{n-1} c_n)\lambda^2}$$

- $\tau = 2$

$$\boxed{\gamma_y(2) = (c_0 c_2 + c_1 c_3 + ... + c_{n-2} c_n)\lambda^2}$$

- ...

- $\tau = n$

$$\boxed{\gamma_y(n) = c_0 c_n \lambda^2}$$

- $|\tau| > n$

$$\boxed{\gamma_y(\tau) = 0, \tau > n}$$

  Which means that **MA(n)** has a **finite memory** of n steps

2.1.2 Probabilistic representation of AR(1)

$y(t) = ay(t-1) + e(t), e(t) \sim WN(0, \lambda^2)$
Is $y(t)$ a **SSP**?
TF representation:
$y(t) = aZ^{-1} y(t) + e(t) \rightarrow y(t) = \frac{1}{1-aZ^{-1}} e(t) \rightarrow y(t) = \frac{Z}{Z-a} e(t)$
So $y(t)$ is a SSP if and only if $|a| < 1$

- **Mean of y**
  $m_y = E[y(t)] = E[ay(t-1) + e(t)] = am_y + m_e$
  $m_y(1-a) = m_e \rightarrow m_y = \frac{m_e}{1-a}$

$m_e = 0$ so :

$$\boxed{m_y = 0}$$

This hold only if the general input $v(t)$ has $m_v = 0$ and the system **W(Z)** is **asymptotically stable**.

- **Covariance of y**

  - $\tau = 0$
    $\gamma_y(0) = E[(y(t) - m_y)^2] = E[y(t)^2] = E[(ay(t-1) + e(t))^2]$
    $\gamma_y(0) = a^2 E[y(t-1)^2] + E[e(t)] + 2aE[e(t)y(t-1)]$
    $\gamma_y(0) = a^2\gamma_y(0) + \lambda^2 + 0$
    Observation: the fact that $E[e(t)y(t-1)] = 0$ is explained in 2.1.3!

$$\boxed{\gamma_y(0) = \frac{\lambda^2}{1-a^2}}$$

  - $\tau = 1$
    $\gamma_y(1) = E[(y(t) - m_y)(y(t-1) - m_y)] = E[(ay(t-1) + e(t))y(t-1)]$
    $\gamma_y(1) = E[ay(t-1)y(t-1)] + E[e(t)y(t-1)] = a\gamma_y(0) + 0$
    Observation: the fact that $E[e(t)y(t-1)] = 0$ is explained in 2.1.3!

$$\boxed{\gamma_y(1) = a\gamma_y(0)}$$

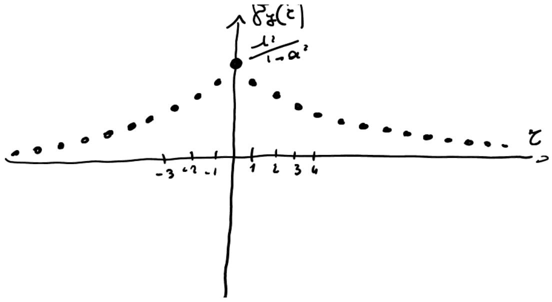
  - ...

  - $\tau \neq 0$

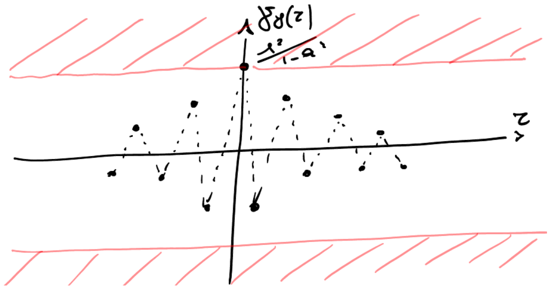
$$\boxed{a\gamma_y(\tau - 1)}$$

    Which means that **AR(1)** has an **infinite memory**.The formula is also known as **Yule-Walker formula of order 1**

1.Plot of $\gamma_y(\tau), 0 < a < 1$



2.Plot of $\gamma_y(\tau), -1 < a < 0$



2.1.3 AR/ARMA as MA(∞)

A general rule states that every AR/ARMA stationary stochastic process can be modelled as **MA(∞)**. Example with **AR(1)** as above:
$y(t) = \frac{1}{1-aZ^{-1}}e(t) \rightarrow y(t) = \sum\limits_{k=0}^{\infty}(aZ^{-1})^k e(t)$
A **geometric series of common ratio** aZ^{-1}
$y(t) = e(t)[1 + aZ^{-1} + a^2Z^{-2} + ...]$

$$\boxed{y(t) = e(t) + ae(t-1) + a^2e(t-2)...}$$

Which is the **MA(∞)** equivalent of AR(1) This formula is very useful to demonstrate that in an AR(1) $E[e(t)y(t-1)] = 0$ by expressing $y(t-1)$ in $MA(\infty)$:
$E[e(t)(e(t-1) + ae(t-2) + a^2e(t-2)...) = E[e(t)e(t-1)] + E[e(t)ae(t-2)... = 0$
Due to correlation all terms are equal to zero (WN property!).

2.2 Frequency Representation

The **power density / spectral density / spectrum** of a **SSP** y(t) :

$$\boxed{\Gamma_y(w) = \sum_{\tau=-\infty}^{\infty} \gamma_y(\tau) e^{-jw\tau}}$$

where $\Gamma_y(w)$ is the **Discrete Fourier Transform**.
Properties :

1. $\Gamma_y(w)$ is a **real** function of a **real** variable w which means that $Im\{\Gamma_y(w)\} = 0$
2. $\Gamma_y(w)$ is a **positive** function which means that $\Gamma_y(w) \geq 0, \forall w \in \Re$
3. $\Gamma_y(w)$ is an **even** function which means that $\Gamma_y(w) = \Gamma_y(-w)$
4. $\Gamma_y(w)$ is a **periodic** function of period 2π which means that $\Gamma_y(w) = \Gamma_y(w + k - 2\pi)$.

2.3 Inverse Fourier Transform

Fourier Transform :

$$\Gamma_y(w) = F\{\gamma_y(\tau)\} = \sum_{t=-\infty}^{\infty} \gamma_y(\tau) e^{-jw\tau}$$

Inverse Fourier Transform :

$$\gamma_y(\tau) = F^{-1}\{\Gamma_y(w)\} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Gamma_y(w) e^{jw\tau} dw$$

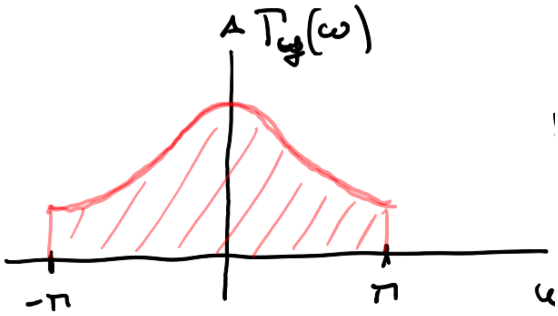
It is important to notice that $\Gamma_y(w)$ and $\gamma_y(\tau)$ contain the **same information** : passing from one to another does not result in **loss** or **gain** of information.

Special IFT : Computation of variance

A special case of IFT is the computation of the variance , when $\tau = 0$:

$$\boxed{\gamma_y(0) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Gamma_y(w) dw}$$

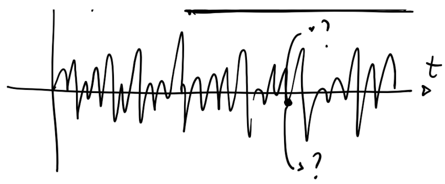
which is the **area below the spectrum** between $(-\pi, \pi)$ divided by 2π.



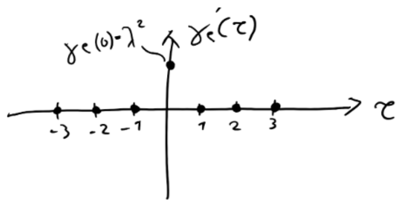
2.4 White Noise in the frequency domain

In case we are dealing with a WN : $e(t) = WN(0, \lambda^2)$ we can consider it in three different domains.

1. **Time domain**
   WN is clearly **unpredictable**

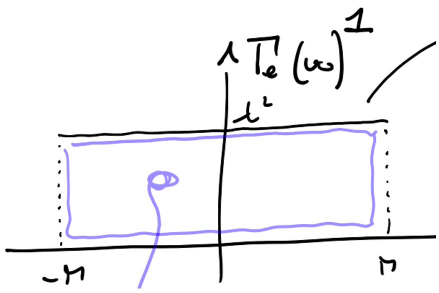


2. **Probabilistic domain**
   Considering the WN in the probabilistic domain and plotting its **variance** only $\gamma_e(0) \neq 0$: there is no **correlation** between $e(t)$ and $e(t \pm \tau)$



3. **Frequency domain**
   Since the definition of FT relies on the definition of **covariance** $\gamma_e(\tau)$, as seen in point 2 only for $\tau = 0 \rightarrow \gamma_e(\tau) \neq 0$:

$$\boxed{\Gamma_e(w) = \gamma_e(0)e^{jw0} = \gamma_e(0) = \lambda^2}$$



The area is $2\pi\lambda^2$ so the variance is $\frac{area}{2\pi} = \lambda^2 = \gamma_e(0)$
The **energy** of the WN is **uniformly distributed** over all frequencies.

2.5 Computation of the spectrum of a process generated as the output of a digital system

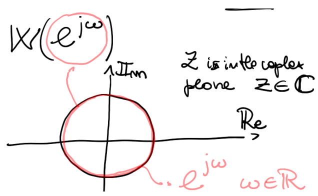
Problem : the **computation** of the $\Gamma_y(w)$ is quite difficult most of the times. A **simpler** alternative can be found using the notion of **Frequency Response**.

2.5.1 Frequency Response of a linear system

Given two signals (**SSP**) input v(t) and output y(t) , where input passes through W(Z) the system or digital filter , then the **frequency response** is

$$W(e^{jw})$$

which corresponds to the evaluation of the **transfer function** on the **unit circumference**



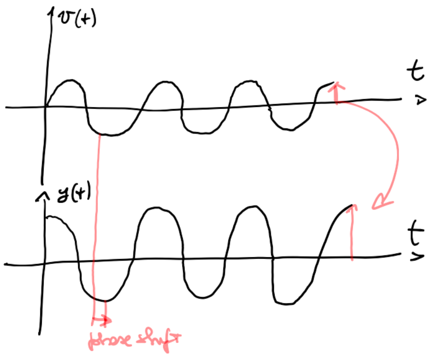
The frequency response is used in system theory in the **Frequency Response Theorem**

FR Th.

If W(Z) is **asymptotically stable** and v(t) is $Asen(\Omega t + \phi$, where A is the amplitude an ϕ the phase of the sinusoid , the the output is a **pure sinusoid** with :

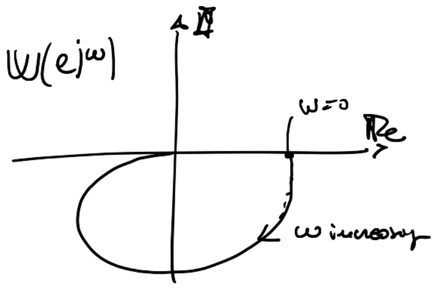
- the **same** angular speed Ω
- amplitude $A|W(e^{j\Omega})| \rightarrow$ **gain**
- phase $\phi + \angle W(e^{j\Omega}) \rightarrow$ **shift in phase**

$$\boxed{y(t) = A|W(e^{j\Omega})|sen(\Omega t + \phi + \angle W(e^{j\Omega})}$$



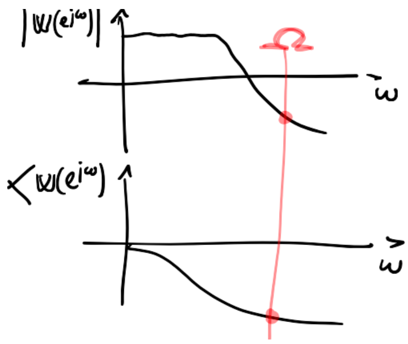
1.FR Nyquist plot

$W(e^{jw})$ is a complex function of a **real** variable.



2.FR Bode plot

Bode plot gives information about **magnitude** and **phase**



2.5.2 Spectrum computation with FR

If y(t) is output of a transfer function W(Z) which is **asymptotically stable** with input signal v(t) , then the spectrum is:

$$\boxed{\Gamma_y(w) = |W(e^{jw})|^2 \Gamma_v(w)}$$

The computation of $\Gamma_v(w)$ still remains but most of the time signal v(t) is a **white noise** $\sim (0, \lambda^2)$ which means that $\Gamma_v(w) = \lambda^2$

2.6 Equivalent representations of ARMA

An ARMA SSP can be represented in 4 different but **equivalent** ways with $e(t) \sim WN(0,1)$:

1. **Time domain** $y(t) = a_1 y(t-1) + ... + a_m y(t-m) + c_0 e(t) + ... + c_n e(t-n)$
2. **Transfer function** $y(t) = \frac{C(Z)}{A(Z)} e(t)$
3. **Probabilistic domain**:
$$\begin{cases} m_y & = E[y(t)] \\ \gamma_y(\tau) & = E[(y(t) - m_y)(y(t-\tau) - m_y)] \end{cases}$$
4. **Frequency domain**
$$\begin{cases} m_y & = E[y(t)] \\ \Gamma_y(w) & w \in \Re \end{cases}$$

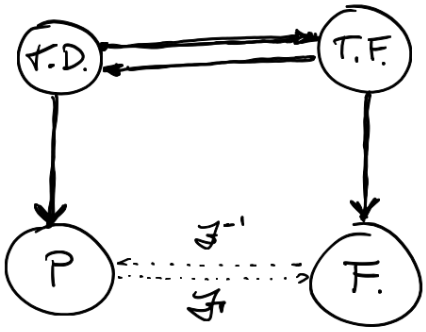
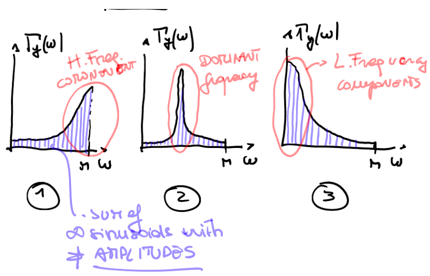


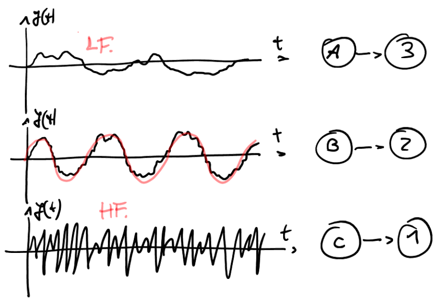
Figure 5: Bold : usual transformation , dotted : feasible but difficult

2.7 Example & Exercises

Example 1

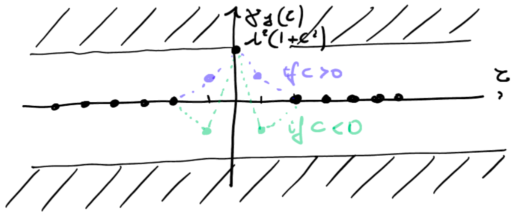
Given 3 output spectra match the corresponding time domain representation.





Example 2

Given a **MA(1)** process $y(t) = e(t) + ce(t-1), e \sim WN(0, \lambda^2), c \in \Re$.
- y(t) is **stationary** because MA(1) is always **asymptotically stable** -
$m_e = 0 \rightarrow m_y = 0$ - $\gamma_y(0) = \lambda^2(1+c^2)$ - $\gamma_y(1) = \lambda^2 c$ - $\gamma_y(\tau) = 0, |\tau| \geq 2$



1.Composition of $\Gamma_y(w)$ with $\lambda^2 = 1$:

- **From definition**

$$\Gamma_y(w) = \sum_{\tau=-\infty}^{\infty} \gamma_y(\tau) e^{-jw\tau}$$

- For $\tau = 0 : (1 + c^2)$
- For $\tau = 1 : ce^{-jw}$
- For $\tau = -1 : c + e^{+jw}$
- For $|\tau| \geq 2 : 0$

$$\Gamma_y(w) = 1 + c^2 + c(e^{-jw} + e^{+jw})$$

Recall : $e^{-jw} + e^{jw} = cosw - jsenw + cosw + jsenw = 2cosw$

$$\Gamma_y(w) = (1 + c^2) + 2ccosw$$

Which is **real, positive, even , periodic**

- **From frequency response**
The MA(1) transfer function is

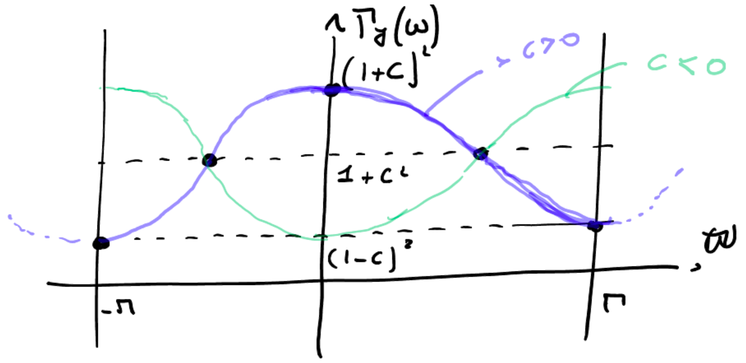
$$y(t) = (1 + cZ^{-1})e(t)$$

$$\Gamma_y(w) = |W(e^{jw}|^2 \Gamma_e(w) = |1 + ce^{-jw}|^2 \cdot 1$$

Recall : $|a + jb|^2 = Im[a + ib]^2 + Re[a + ib]^2 = a^2 + b^2 = (a + jb)(a - jb)$

$(1 + ce^{-jw})(1 - ce^{jw}) = 1 + c^2(e^{jw} \cdot e^{-jw}) + c(e^{-jw} + e^{jw}) = 1 + c^2 + 2ccosw$

2.Plotting of $\Gamma_y(w)$:
$\Gamma_y(0) = (1+c)^2$
$\Gamma_y(\frac{\pi}{2} = 1 + c^2)$
$\Gamma_y(\pi) = (1-c)^2$



3.Compute the variance $\gamma_y(0)$ given $\Gamma_y(w)$:

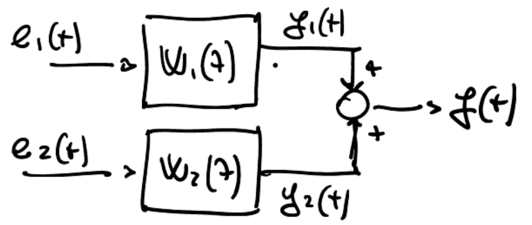
$$\gamma_y(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi}\Gamma_y(w)dw = \frac{1}{2\pi}\int_{-\pi}^{\pi}(1+c^2+2ccosw)dw$$

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}(1+c^2)dw + \frac{1}{2\pi}\int_{-\pi}^{\pi}2ccoswdw$$

$$\frac{1}{2\pi}[(1+c^2)[w]_{-\pi}^{\pi} + 2c[senw]_{-\pi}^{\pi}] = \frac{1}{2\pi}[(1+c^2)(2\pi)] =$$

$$1+c^2$$

Example 3



Consider the SSP y(t) generated by 2 inputs.
$W_1(t), W_2(t)$ are asymptotically stable.
$e_1(t) \sim W(0, \lambda_1^2), e_2(t) \sim W(0, \lambda_2^2)$
$e_1 \perp e_2 \rightarrow E[e_1(t)e_2(t-\tau)] = 0$
Calculate $\gamma_y(\tau)$ and $\Gamma_y(w)$

- $\gamma_y(\tau)$
  $\gamma_y(\tau) = E[y(t)y(t-\tau)] = E[(y_1(t) + y_2(t))(y_1(t-\tau) + y_2(t-\tau))]$
  $E[y_1(t)y_1(t-\tau)] + E[y_2(t)y_2(t-\tau)] + E[y_1(t)y_2(t-\tau)] + E[y_2(t)y_1(t-\tau)]$
  $\gamma_{y_1}(\tau) + \gamma_{y_2}(\tau) + 0 + 0$
  Term 3 and 4 are $= 0$ which is a result obtained by rewriting them as $MA(\infty)$ and exploiting the hypothesis that $e_1(t) \perp e_2(t)$.

$$\boxed{\gamma_y(t) = \gamma_{y_1}(t) + \gamma_{y_2}(t)}$$

- $\Gamma_y(t)$
  $\Gamma_y(t) = \sum_{\tau=-\infty}^{\infty} \gamma_y(\tau)e^{-jw\tau} = \sum_{\tau=-\infty}^{\infty} \gamma_{y_1}(\tau)e^{-jw\tau} + \sum_{\tau=-\infty}^{\infty} \gamma_{y_2}(\tau)e^{-jw\tau}$

$$\boxed{\Gamma_y(w) = \Gamma_{y_1}(w) + \Gamma_{y_2}(w)}$$

The result can be generalised to more than 2 inputs that are summed to form an SSP y(t):

$$\boxed{\gamma_y(t) = \gamma_{y_1}(t) + \gamma_{y_2}(t) + ... + \gamma_{y_k}(t)}$$

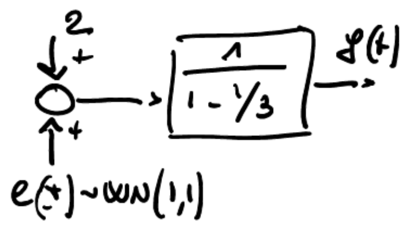
$$\boxed{\Gamma_y(w) = \Gamma_{y_1}(w) + \Gamma_{y_2}(w) + ... + \Gamma_{y_k}(t)}$$

The result hold if all $W_i(t)$ are asymptotically stable , all $v_i(t)$ are ssp and incorrelated

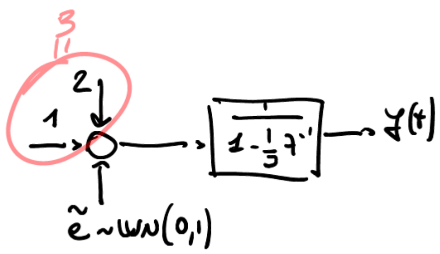
Example 4

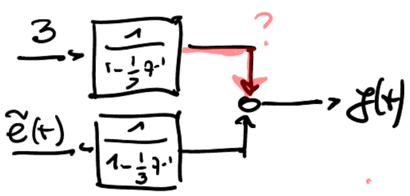
Consider the following AR(1) SSP $y(t) = \frac{1}{3}y(t-1) + e(t) + 2 \rightarrow e \sim WN(1,1)$ which has a asymptotically stable TF.
Calculate m_y and γ_y.

- **Mean of y**
  - **Method 1**
    $E[y(t)] = E[\frac{1}{3}y(t-1)e(t) + 2] \rightarrow (1 - \frac{1}{3})m_y = m_e + 2 \rightarrow m_y = \frac{9}{2}$
  - **Mehotd 2**



$e(t) = \tilde{e} + 1, \tilde{e} \sim WN(0,1)$
Using the superposition principle of LTI systems :





The constant value 3 can be seen as a **sinusoidal signal with w=0** so the **Frequency Response Theorem** can be applied:
$3(\frac{1}{1-\frac{1}{3}Z^{-1}})$calculated in$z = e^{j0}$
$3(\frac{1}{1-\frac{1}{3}}) = \frac{9}{2}$ And since $m_{\tilde{e}}$ is a zero mean signal $\rightarrow m_y = \frac{9}{2}$

- **Covariance of y**

  - **Method 1 : BAD**
    $E[(y(t)-\frac{9}{2})^2] = E[(\frac{1}{3}y(t-1)+e(t)+2-\frac{9}{2})^2]$
    $\gamma_y(0) = \frac{1}{9}E[y(t-1)^2] + E[e(t)^2] + \frac{25}{4} + \frac{2}{3}E[y(t-1)e(t)] - \frac{5}{3}E[y(t-1)] + 5E[e(t)]$

    Remark: $E[(e(t)-m_e) = \gamma_e(0) = E[e(t)^2] - 2E[e(t)m_e] + m_e^2$
    $E[e(t)] = \gamma_e(0) + m_e^2$
    Which can be generalised :

    $$\boxed{E[e(t)^2] = \gamma_e(0) + m_e^2}$$

    $$\boxed{E[y(t)^2] = \gamma_y(0) + m_y^2}$$

    $E[e(t)y(t-1)] = E[(e(t)-m_e)(y(t-1)-m_y)] + m_y m_e$

    $$\boxed{E[e(t)y(t-1)] = m_e m_y}$$

    As the **de-biased signals are incorellated!**

    $\gamma_y(0) = \frac{1}{9}(\gamma_y(0)+m_y^2)+(\gamma_e(0)+m_e^2)+\frac{25}{4}+\frac{2}{3}(m_e m_y)-\frac{5}{3}m_y-5m_e = \frac{9}{8}$
    Same computations for $\gamma_y(1), \gamma_y(2)$...
  - **Method 2: GOOD**
    Define two new processes:
    $\tilde{y}(t) = y(t) - \frac{9}{2} \rightarrow m_{\tilde{y}=0}$
    $\tilde{e}(t) = e(t) - 1 \rightarrow m_{\tilde{e}=0}$
    So $y(t) = \tilde{y}(t) - \frac{9}{2}$ and $e(t) = \tilde{e}(t) + 1$:
    $\tilde{y}(t) + \frac{9}{2} = \frac{1}{3}(\tilde{y}(t-1) + \frac{9}{2}) + (\tilde{e}+1) + 2$

    $$\boxed{\tilde{y}(t) = \frac{1}{3}\tilde{y}(t-1) + \tilde{e}(t), \tilde{e} \sim WN(0,1)}$$

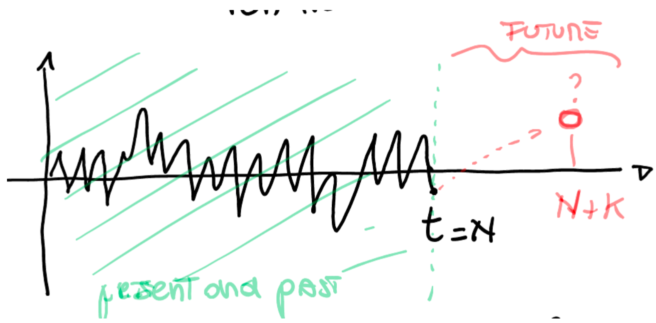
Where $\tilde{y}(t)$ is the **de-biased process**
$\gamma_{\tilde{y}(0)} = \frac{1}{1-\frac{1}{9}} = \frac{9}{8}$
$\gamma_{\tilde{y}(1)} = \frac{9}{8}\frac{1}{3} = \frac{3}{8}$
$\gamma_{\tilde{y}(2)} = \frac{3}{8}\frac{1}{3} = \frac{1}{8}$ ...
Now that we found $\gamma_{\tilde{y}(\tau)}$ we want to find $\gamma_y(\tau)$ $\gamma_y(\tau)$:
$E[(y(t) - \frac{9}{2})(y(t-\tau) - \frac{9}{2})] = E[\tilde{y}(t)\tilde{y}(t-\tau)] = \gamma_{\tilde{y}}(\tau)$
since $m_{\tilde{y}} = 0$ Which can be generalised :
If $y(t)$ and $\widetilde{y(t)}$ are two SSPs that **differ only from a constant value** $y(t) = \tilde{y}(t) + k$ then :

$$\boxed{\gamma_y(\tau) = \gamma_{\tilde{y}}, \forall \tau}$$

$$\boxed{\Gamma_y(w) = \Gamma_{\tilde{y}}, \forall w}$$

3 Chapter 3 : Prediction

The prediction problem is to find the **best possible value** for $\hat{y}(t+k|t)$ given the **measured data** up to time t $\{y(1), ..., y(N)\}$



To obtain the **optimal** prediction :

1. We have to make a mathematical model for $\{y(1), .., y(N)\}$
2. Using the model compute the optimal solution

To find the **best** mathematical model :

1. We select a class of models for time-series $y(t) = W(z, \theta)e(t)$ where $e(t)$ is a WN and θ a parameter vector.
2. We compute the prediction of $y(t)$ using the mathematical model :

$$\hat{y}(t+1|t;\theta)$$

3. $\hat{\theta} = argmin_\theta[\frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|t-1);\theta))^2]$
4. Find $y(t) = W(Z, \hat{\theta})e(t)$ which is the best model from prediction performance . Use this to compute $\hat{y}(N+K|N)$

To create a **predictor** from an ARMA/ARMAX we need to define 2 tools:

- **All-pass filter**
- **Canonical representation**

3.1 All-Pass Filter

An All-Pass Filter is a **first-order, linear, digital** filter with a special **constrained** structure:

$$\boxed{T(Z) = \frac{1}{a}\frac{Z+a}{Z+\frac{1}{a}}, a \in \Re}$$

that depends on only one parameter and has a **pole** in $z = -\frac{1}{a}$ and zero in $z = -a$
Properties :

- **Magnitude**
  $|T(e^{jw})|^2 = |\frac{1}{a}\frac{e^{jw}+a}{e^{jw}+\frac{1}{a}}|^2 = \frac{1}{a}(\frac{e^{jw}+a}{e^{jw}+\frac{1}{a}}) \cdot \frac{1}{a}(\frac{e^{-jw}+a}{e^{-jw}+\frac{1}{a}}) = \frac{1}{a^2}\frac{1+a^2+2acosw}{1+\frac{1}{a^2}+\frac{2cosw}{a}} = 1$
  An all-pass filter is characterized by a **frequency response** having **unitary magnitude** :

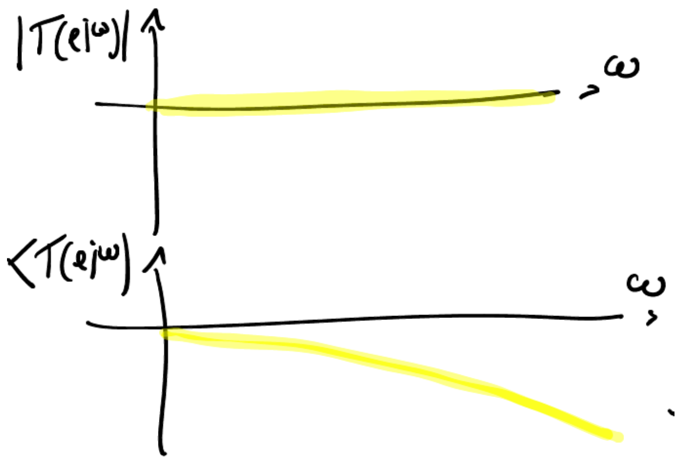
  $$\Gamma_y(w) = |T(e^{jw})|^2 T_v(w)$$

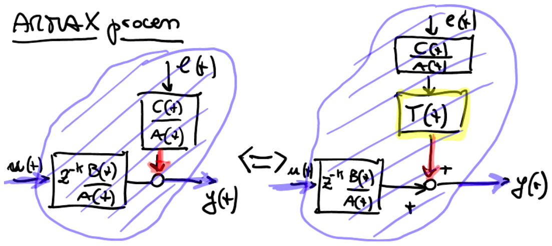
  An all-pass filter does not alter the **spectrum** of its input v(t). This does **not** mean that $y(t) = v(t)$ but that they're **statistically equivalent** :

  $$\boxed{\Gamma_y(w) = \Gamma_v(w)}$$

  $$\boxed{\gamma_y(w) = \gamma_v(w)}$$

  Input and output are **not** identical because although there is no change in amplitude ,an all-pass filter makes **a distortion in phase**.





The two representations of the ARMAX process are **equivalent**. The phase distortion added to signal e(t) is **not relevant**. On the other hand , adding a T(Z) all-pass filter between $u(t)$ and $y(t)$ alters the behaviour of the system **critically**!!

3.2 Canonical Representation

An ARMA process can have ∞ **equivalent** representations (there is no way to represent it in a unique way).
There is a special representation called **Canonical Representation**:
Given a SSP y(t) that can be modellded as an ARMA process:

$$y(t) = \frac{C(Z)}{A(Z)}e(t)$$

,

$$\frac{C(Z)}{A(Z)}$$

is the **canonical representation** if:

1. C(Z) and A(Z) have **same degree** (relative degree is 0)
2. C(Z) and A(Z) are **coprime** (no common factors)
3. C(Z) and A(Z) are **monic** (coefficient of max degree of both C(Z) and A(Z) is 1)
4. All roots of C(Z) and A(Z) are **strictly inside** the unit circle

Example

$y(t) = \frac{1+3Z^{-1}}{2+Z^{-1}}e(t-2), e(t) \sim WN(0,1)$

- **Type and order**
  ARMA type process of order 1,3 = ARMA(1,3)
- **Canonical form**

$$y(t) = \frac{Z^{-2} + 3Z^{-3}}{2 + Z^{-1}} e(t)$$

1. Degree of C(Z) is 2 , degree of A(Z) is 0 $\rightarrow$ X
2. No common factors $\rightarrow$ OK
3. C(Z) is monic , A(Z) is not monic $\rightarrow$ X
4. Zero in -3 $\rightarrow$ X

By collecting and using an **All-Pass Filter**:

$$y(t) = \frac{Z^{-2}(1 + 3Z^{-1})}{2(1 + \frac{1}{2}Z^{-1})} \cdot 3\frac{1 + \frac{1}{3}Z^{-1}}{1 + 3Z^{-1}} e(t)$$

$$y(t) = \frac{z^{-2}\cancel{(1+3z^{-1})}}{2(1+\frac{1}{2}z^{-1})} \underbrace{\boxed{3\frac{1+\frac{1}{3}z^{-1}}{\cancel{1+3z^{-1}}}}}_{T(z)} e(t)$$

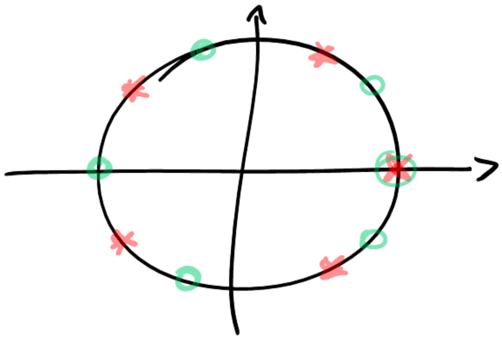
Defining $\theta = \frac{3}{2}e(t-2), \theta \sim ?$
The variance of $\theta = E[\theta(t)^2] = E[(\frac{3}{2}e(t-2))^2] = \frac{9}{4} \cdot 1$
$\theta \sim WN(0, \frac{9}{4})? \rightarrow \Gamma_\theta(w) = |\frac{3}{2}e^{-2jw}|^2 \cdot 1 = \frac{9}{4}$ which is constant value so
$\theta \sim WN(0, \frac{9}{4})$ Finally we obtain :

$$\boxed{y(t) = \frac{1 + \frac{1}{3}Z^{-1}}{1 + \frac{1}{2}Z^{-1}} \theta(t), \theta \sim WN(0, \frac{9}{4})}$$

Which is an **ARMA(1,1)** $\rightarrow$ the **canonical representation** is the representation with **minimum order!**

Remark

Does the canonical representation **always** exist?
Not always : the problem lies in the 4^{th} condition . It is possible that the system has poles or zeros **on** the unitary circle.



If $C(Z)$ has **zeros on the u.c** $\rightarrow$ prediction from data is **not asymptotically stable**
If $A(Z)$ has **roots in +1** $\rightarrow$ **ARIMA models**:

$$\boxed{y(t) = \frac{C(Z)}{(Z-1)^d A(Z)} e(t) \rightarrow ARIMA(m,d,n)}$$

[Autoregressive Integrated Moving average]
A special case of ARIMA $\rightarrow$ **ARIMA(0,1,0)**:

$$\boxed{y(t) = \frac{1}{1-Z^{-1}} e(t)}$$

$$\boxed{y(t) = y(t-1) + e(t), e(t) \sim WN(0, \lambda^2)}$$

Process y(t) is a **Random Walk** that uses as TF $\frac{1}{1-Z^{-1}}$ which is an **integrator in discrete time**

ARIMA processes have an **asymptotically stable** predictor that can be used to model **not strictly stationary processes!!**

3.3 Predictor

The predictor at time t+k , given the data up to time t is:

$$\hat{y}(t+k|t)$$

The **prediction error** is:

$$\epsilon(t+k) = y(t+k) - \hat{y}(t+k|t)$$

So the **real value** is predictor + error:

$$y(t+k|t) = \hat{y}(t+k|t) + \epsilon(t+k)$$

$$y(t) = \hat{y}(t|t-k) + \epsilon(t)$$

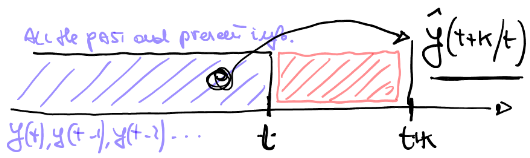
The formulas as equivalent because as per hypothesis $y(t)$ is **stationary**.

3.3.1 Optimality

The predictor $\hat{y}(t+k|t)$ is **optimal** if:

1. $E[\hat{y}(t+k|t) \cdot \epsilon(t+k)] = 0$, predictor and error must be **incorrelated**

2. $E[y(t) \cdot \epsilon(t+k)] = E[y(t-1) \cdot \epsilon(t+k)] ... = 0$



The red part shows the **unpredictable** part of y(t+k) , which is the error $\epsilon(t+k)$. If error and predictor were **correlated** than some useful unused information about $\hat{y}(t+k|t)$ would be in $\epsilon(t+k)$ which means that the predictor is not optimal. The same goes for $y(t), y(t-1)...$ of point 2): the error cannot contain information about the past/present information.

3.3.2 1-step ahead prediction of MA(n)

$$y(t) = e(t) + c_1 e(t-1) + ... + c_n e(t-n), e(t) \sim WN(0, \lambda^2)$$

We assume the the MA(n) is represented in the **canonical representation** : we must make assumptions about the 4^{th} property.
Given :
- **Present time** : t-1 $\rightarrow c_1 e(t-1) + ... + c_n e(t-n)$
- **Future** : t $\rightarrow e(t)$

Predictor from noise

The **optimal predictor** from **noise** is :

$$\boxed{\hat{y}(t|t-1) = c_1 e(t-1) + ... + c_n e(t-n)}$$

with error

$$\boxed{\epsilon(t) = y(t) - \hat{y}(t|t-1) = e(t)}$$

Optimality :

- $E[\hat{y}(t|t-1)\epsilon(t)] = E[(c_1 e(t-1) + ... + c_n e(t-n))(e(t))] = 0$
- $E[y(t-1)\epsilon(t)] = E[(e(t-1)+c_1 e(t-2)+...+c_n e(t-n-1))(e(t))] = ... = 0$

Verified because of incorrelation of white noise.
Since WN is **unknown** and cannot be measured , a better predictor has to be chosen from **measurable data**.

Predictor from data

TF:

$$y(t) = (1 + c_1 Z^{-1} + ... + c_n Z^{-n})e(t)$$

Inverse TF (**Whitening Filter**):

$$e(t) = \frac{1}{1 + c_1 Z^{-1} + ... + c_n Z^{-n}} y(t)$$

$$\hat{y}(t|t-1) = (1+c_1 Z^{-1}+...+c_n Z^{-n})e(t) \rightarrow \hat{y}(t|t-1) = \frac{c_1 Z^{-1} + ... + c_n Z^{-n}}{1 + c_1 Z^{-1} + ... + c_n Z^{-n}} y(t)$$

Collecting Z^{-1} :

$$\boxed{\hat{y}(t|t-1) = \frac{c_1 + ... + c_n Z^{-n+1}}{1 + c_1 Z^{-1} + ... + c_n Z^{-n}} y(t-1)}$$

$$\hat{y}(t/t-1) = \overbrace{-c_1 \hat{y}(t-1/t-2) - c_2 \hat{y}(t-2/t-3) \cdots - c_n \hat{y}(t-n/t-n-1)}^{\text{PAST PREDICTIONS}} + \underbrace{c_1 y(t-1) + c_2 y(t-2) + \cdots c_n y(t-n)}_{\text{PRESENT AND PAST MEASUREMENT}}$$

As seen in time-domain representation the prediction makes use of **present and past** data as well as **past predictions**.

3.3.3 K-steps ahead predictor of MA(n)

$$y(t) = e(t) + c_1 e(t-1) + ... + c_{k-1} e(t-k+1) + c_k e(t-k) + ... + c_n e(t-n)$$

Given:
-**Present time**: k $\rightarrow c_k e(t-k) + ... + c_n e(t-n)$
-**Future** : t $\rightarrow e(t) + ... + c_{k-1} e(t-k+1)$

Predictor from noise

$$\boxed{\hat{y}(t|t-k) = c_k e(t-k) + ... + c_n e(t-n)}$$

with error:

$$\boxed{\epsilon(t) = e(t) + ... + c_{k-1} e(t-k+1)}$$

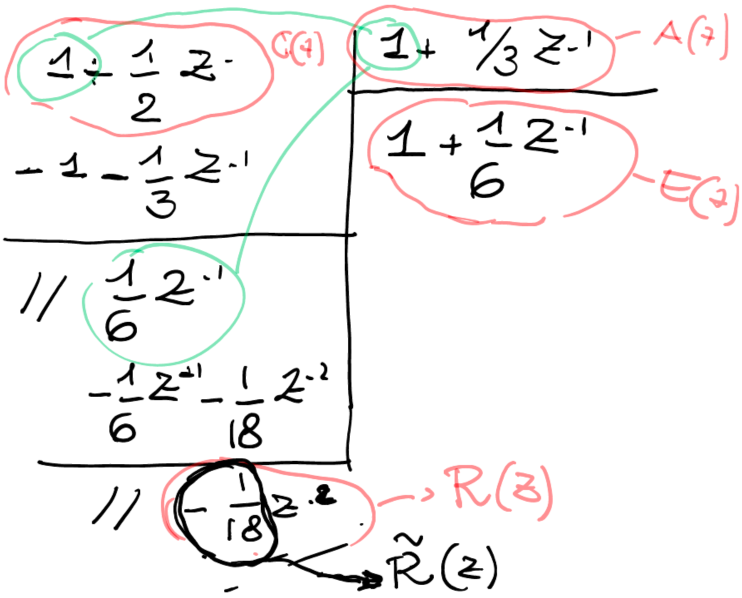
Predictor from data

$$\boxed{\hat{y}(t|t-k) = \frac{c_k + c_{k+1} Z^{-1} + ... + c_n Z^{-n+k}}{1 + c_1 Z^{-1} + ... + c_n Z^{-n}} y(t-k)}$$

3.3.4 K-steps ahead predictor of general ARMA(m,n)

$$y(t) = \frac{C(Z)}{A(Z)} e(t), e(t) \sim WN(0, \lambda^2)$$

(Under the hypothesis of **canonical representation**)
The AR(m) part presents a recursion : need to introduce k-steps **polynomial division** between C(Z) and A(Z) obtaining :
- **E(Z)** $\rightarrow$ result (quotient)
- **R(Z)** $\rightarrow$ residual (remainder)



$$C(Z) = E(Z)A(Z) + R(Z)$$

$$\boxed{\frac{C(Z)}{A(Z)} = E(Z) + \frac{R(Z)}{A(Z)}}$$

Noting that in k-steps division R(Z) can be rewritten by collecting Z^{-k}:

$$R(Z) = Z^{-k}\tilde{R}(Z)$$

$$\boxed{\frac{C(Z)}{A(Z)} = E(Z) + \frac{Z^{-k}\tilde{R}(Z))}{A(Z)}}$$

The new transfer function is:

$$y(t) = [E(Z) + \frac{Z^{-k}\tilde{R}(Z))}{A(Z)}]e(t)$$

$$y(t) = E(Z)e(t) + \frac{\tilde{R}(Z))}{A(Z)}e(t-k)$$

Where $E(Z)e(t)$ is the **unpredictable part** as it depends on $e(t), ..., e(t-k+1)$

Predictor from noise

$$\boxed{\hat{y}(t|t-k) = \frac{\tilde{R}(Z)}{A(Z)}e(t-k)}$$

with error:

$$\boxed{\epsilon(t) = E(Z)e(t)}$$

Predictor from data

$$y(t) = \frac{C(Z)}{A(Z)}e(t) \xrightarrow[\text{Whitening}]{} e(t) = \frac{A(Z)}{C(Z)}y(t)$$

$$\hat{y}(t|t+k) = \frac{\tilde{R}(Z)Z^{-k}}{A(Z)} \cdot \frac{A(Z)}{C(Z)}y(t)$$

$$\boxed{\hat{y}(t|t-k) = \frac{\tilde{R}(Z)}{C(Z)}y(t-k)}$$

Remark 1

Both the predictor from noise and data work under the assumption of SSP . The stationary property is satisfied if both A(Z) and C(Z) have all roots (poles) strictly inside the unitary circle. But this is satisfied by the 4^{th} condition of the canonical representation hypothesis.

Remark 2

$\epsilon(t) = y(t) - \hat{y}(t|t-k) = E(Z)e(t)$ where $E(Z)$ is a SSP of type **MA(k-1)**

Remark 3

In the case of **K=1** the polynomial division result in :
-**E(Z)** = 1 as both C(Z),A(Z) are monic and have same degree
-**R(Z)** = C(Z)-A(Z)
which results in

$$\boxed{\hat{y}(t|t-k) = \frac{C(Z) - A(Z)}{A(Z)}e(t)}$$

$$\boxed{\hat{y}(t|t-k) = \frac{C(Z) - A(Z)}{C(Z)}y(t)}$$

$$\boxed{\epsilon(t) = e(t)}$$

Instead of having term R(Z) , the formula is now C(Z)-A(Z). As $R(Z) = \tilde{R}(Z)Z^{-1}$ there is a hidden Z^{-1} in C(Z)-A(Z).

3.3.5 K-steps ahead prediction of ARMAX(m,n,k+p)

$$y(t) = \frac{B(Z)}{A(Z)}u(t-k) + \frac{C(Z)}{A(Z)}e(t), e(t) \sim WN(0, \lambda^2)$$

Where:

$$A(Z) = 1 + a_1 Z^{-1} + ... + a_m Z^{-m}$$

$$B(Z) = b_0 + b_1 Z^{-1} + ... + b_p Z^{-p}$$

$$C(Z) = 1 + c_1 Z^{-1} + ... + c_n Z^{-n}$$

In the hypothesis that $\frac{C(Z)}{A(Z)}$ is in **canonical representation** and keeping in mind that for $\frac{B(Z)}{A(Z)}u(t-k)$ no **spectral equivalence** modifications can be made.
In an ARMAX(m,n,k+p) process the most interesting prediction that can be made is the **delay** between u(t) and y(t) → **k** so we'll deal only with k-steps predictions.

Predictor from noise

Separate predictable from unpredictable part in $\frac{C(Z)}{A(Z)}e(t)$
K-steps division $\frac{C(Z)}{A(Z)} \to E(Z) + \frac{R(Z)}{A(Z)}$

$$y(t) = \frac{B(Z)}{A(Z)}u(t-k) + E(Z)e(t) + \frac{\tilde{R}(Z)}{A(Z)}e(t-k)$$

where

$$\frac{B(Z)}{A(Z)}u(t-k) \to \text{depends on } u(t-k), ..., u(t-k-p) \to \text{predictable}$$

$$E(Z)e(t) \to \text{depends on } e(t), e(t-1), ..., e(t-k+1) \to \text{unpredictable}$$

$$\frac{\tilde{R}(Z)}{A(Z)}e(t-k) \to \text{depends on } e(t-k), ..., e(t-k-p) \to \text{predictable}$$

so

$$\boxed{\hat{y}(t|t-k) = \frac{B(Z)}{A(Z)}u(t-k) + \frac{R(Z)}{A(Z)}e(t)}$$

$$\boxed{\epsilon(t) = y(t) - \hat{y}(t|t-k) = E(t)e(t)}$$

Which is optimal if

- $\epsilon(t) \perp \hat{y}(t|t-k)$
- $\epsilon(t) \perp y(t-k), y(t-k-1)...$

Predictor from data

$$e(t) = \frac{A(Z)}{C(Z)}y(t) - \frac{B(Z)}{A(Z)}u(t-k)$$

$$\hat{y}(t|t-k) = \frac{B(Z)}{A(Z)}u(t-k) + \frac{R(Z)}{A(Z)}[\frac{A(Z)}{C(Z)}y(t) - \frac{B(Z)}{A(Z)}u(t-k)]$$

$$\hat{y}(t|t-k) = \frac{R(Z)}{C(Z)}y(t) + [\frac{B(Z)}{A(Z)} - \frac{R(Z)B(Z)}{A(Z)C(Z)}]u(t-k)$$

$$\hat{y}(t|t-k) = \frac{B(Z)}{C(Z)}y(t) + [\frac{B(Z)(C(Z)-R(Z))}{A(Z)C(Z)}]u(t-k)$$

Knowing that $C(Z) = A(Z)E(Z) + R(Z) \rightarrow C(Z) - R(Z) = A(Z)E(Z)$

$$\boxed{\hat{y}(t|t-k) = \frac{B(Z)E(Z)}{C(Z)}u(t-k) + \frac{\tilde{R}(Z)}{C(Z)}y(t-k)}$$

$$\boxed{\epsilon = E(Z)e(t)}$$

Note that $\frac{\tilde{R}(Z)}{C(Z)}y(t-k)$ is the exact ARMA predictor.

The prediction error is the same as in the **ARMA** process: the **exogenous** part does not add any **additional uncertainty**.

Remark : Special case k=1

$$y(t) = \frac{B(Z)}{A(Z)}u(t-1) + \frac{C(Z)}{A(Z)}e(t)$$

$E(Z) = 1$ and $R(Z) = C(Z) - A(Z)$

$$\hat{y}(t|t-1) = \frac{B(Z)}{C(Z)}u(t-1) + \frac{C(Z)-A(Z)}{C(Z)}y(t)$$

3.4 Examples & Exercises

3.4.1 Example 1

Given a process

$$y(t) = \frac{Z+3}{2Z+1}e(t-1), e(t) \sim WN(0,1)$$

Since the pole of the TF is $z = -\frac{1}{2}$ inside the unitary circle , W(Z) is asymptotically stable $\rightarrow$ y(t) is **stationary**.

1. **Compute** $\gamma_y(0)$

   NB.: To calculate the variance it is not important fort the system to be in canonical representation

   $y(t) = \frac{C(Z)}{A(Z)}e(t-1)$ is **not canonical** since it has

   - Z = -3 not inside unitary circle
   - 2Z in the A(Z) term
   - $Z^{-1}e(t)$

   Using an **All-Pass Filter**:

$$y(t) = \frac{Z+3}{2(Z+\frac{1}{2})}Z^{-1} \cdot 3\frac{Z+\frac{1}{3}}{Z+3}e(t)$$

$$\eta = \frac{3}{2}Z^{-1}e(t) \sim WN(0, \frac{9}{4}$$

$$y(t) = \frac{Z+\frac{1}{3}}{Z+\frac{1}{2}}\eta(t)$$

   Passing in time domain :

$$y(t) = -\frac{1}{2}y(t-1) + \eta(t) + \frac{1}{3}\eta(t-1)$$

   -$m_y = E[y(t)] = -\frac{1}{2}E[y(t-1)] + \frac{4}{3}m_e \to 0$

   -$\gamma_y(0) = E[y(t)^2] = E[(-\frac{1}{2}y(t-1) + \eta(t) + \frac{1}{3}\eta(t-1))^2]$

   $\gamma_y(0) = \frac{1}{4}\gamma_y(0) + \frac{9}{4} + \frac{1}{9}\frac{9}{4} - \frac{1}{3}E[y(t-1)\eta(t-1)]$

   $\frac{3}{4}\gamma_y(0) = \frac{10}{4} - \frac{1}{3}E[(-\frac{1}{2}y(t-2) + \eta(t-1) + \frac{1}{3}\eta(t-2))\eta(t-1)]$

   $\frac{3}{4}\gamma_y(0) = \frac{10}{4} - \frac{1}{3}E[\eta(t-1)^2] \to \frac{3}{4}\gamma_y(0) = \frac{10}{4} - \frac{1}{3}\frac{9}{4}$

$$\boxed{\gamma_y(0) = \frac{7}{3}}$$

2. **Prediction for k=1**

   Using the canonical negative power representation

$$y(t) = \frac{1+\frac{1}{3}Z^{-1}}{1+\frac{1}{2}Z^{-1}}\eta(t)$$

   Applying the k=1 prediction formula $\hat{y}(t|t-1) = \frac{C(Z)-A(Z)}{C(Z)}y(t)$:

$$\hat{y}(t|t-1) = \frac{1+\frac{1}{3}Z^{-1} - 1 - \frac{1}{2}Z^{-1}}{1+\frac{1}{3}Z^{-1}}y(t)$$

$$\hat{y}(t|t-1) = \frac{-\frac{1}{6}}{1+\frac{1}{3}Z^{-1}} y(t-1)$$

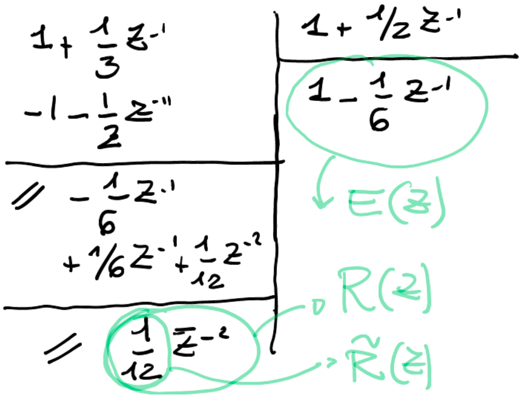
In time domain :

$$\boxed{\hat{y}(t|t-1) = -\frac{1}{3}\hat{y}(t-1|t-2) - \frac{1}{6}y(t-1)}$$

$$\boxed{\epsilon(t) = y(t) - \hat{y}(t|t-1) = E(Z)\eta(t) = \eta(t)}$$

$$\boxed{var[y(t) - \hat{y}(t|t-1)] = var[\eta(t)] = \frac{9}{4}}$$

3. **Prediction for k=2**



$$\hat{y}(t|t-2) = \frac{R(Z)}{C(Z)}y(t) = \frac{\tilde{R}(Z)}{C(Z)}y(t-2)$$

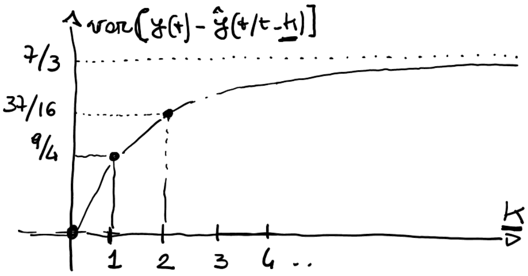
$$\hat{y}(t|t-2) = \frac{\frac{1}{12}}{1+\frac{1}{3}Z^{-1}}y(t-2)$$

$$\boxed{\hat{y}(t|t-2) = -\frac{1}{3}\hat{y}(t-1|t-3) + \frac{1}{12}y(t-2)}$$

$$\boxed{\epsilon(t) = y(t) - \hat{y}(t|t-2) = E(Z)\eta(t) = (1-\frac{1}{6}Z^{-1})\eta(t)}$$

$$\boxed{var[y(t) - \hat{y}(t|t-2)] = var[(1-\frac{1}{6}Z^{-1})\eta(t)] = \frac{37}{16}}$$

4. **Properties of $var[\epsilon(t)]$ as function of k**



- $k = 0 \rightarrow var[y(t) - \hat{y}(t|t-k)] = 0$
- $k = 1 \rightarrow var[y(t) - \hat{y}(t|t-k)] = \lambda^2$
- $k \rightarrow \infty \rightarrow var[y(t) - \hat{y}(t|t-k)] = \gamma_y(0)$ because when $k \rightarrow \infty$ the prediction goes to zero!
- $var[y(t) - \hat{y}(t|t-k)]$ is a **monotonic (not strictly) increasing** function

5. **Prediction goodness**

The ESR is a useful prediction measure:

$$ESR(k) = \frac{var[y(t) - \hat{y}(t|t-k)]}{var[y(t)]}$$

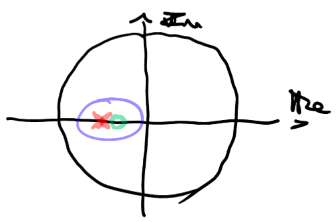
For k=1

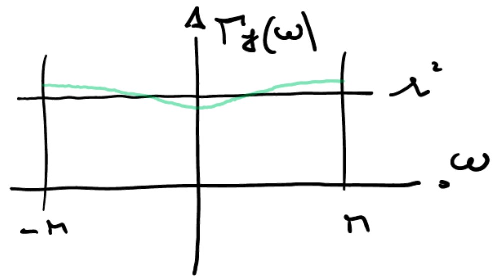
$$ESR(1) = \frac{\frac{9}{4}}{\frac{7}{3}} = 0.97$$

Which is a very bad prediction. The most trivial prediction that can be done is

$$\hat{y}(t|t-k) = m_y$$

(predicting the mean) which has **ESR(k)=1**.
For k=1 we only have a 3% better prediction than the trivial one.
The predictor for k=1 is **optimal** which means that **no better** prediction can be made : the bad prediction is an intrinsic property of the process y(t).





By analysing the poles and zeros, it is easy to see that they're so close together that they almost cancel each other out.
The **spectrum** $\Gamma_y(w)$ in green is very close to that of the **white noise**: this is the reason y(t) is hard to predict

3.4.2 Example 2 - Practical

We have measured 5 data points of a signal:
$y(1) = 1, y(2) = \frac{1}{2}, y(3) = -\frac{1}{2}, y(4) = 0, y(5) = -\frac{1}{2}$
With t=5 represent the present time , make a prediction of $\hat{y}(6|5)$. To solve the problem we must make a mathematical modelling assumption. Since we're still not able to do this we need some interpretations models for this signal
Model A

$$y(t) = \frac{1}{2}y(t-1) + \frac{1}{4}y(t-2) + e(t), e(t) \sim WN(0, \lambda^2)$$

Model B

$$y(t) = e(t) + \frac{1}{2}e(t-1), e(t) \sim WN(0, \lambda^2)$$

To determine which model is better we compute the **optimal** model assuming the chosen model is right.

- **Assuming Model A right**

$$y(t) = \frac{1}{1 - \frac{1}{2}Z^{-1} - \frac{1}{4}Z^{-2}}e(t)$$

is an AR(2) process in canonical representation (check it always!).
Since we're dealing with a k=1 prediction :

$$\hat{y}(t|t-1) = \frac{C(Z) - A(Z)}{C(Z)}y(t)$$

$$\hat{y}(t|t-1) = \frac{1 - 1 + \frac{1}{2}Z^{-1} + \frac{1}{4}Z^{-2}}{1} y(t)$$
$$\hat{y}(t|t-1) = \frac{1}{2}y(t-1) + \frac{1}{4}y(t-2)$$

Substituting the data points :

$$\hat{y}(6|5) = \frac{1}{2}y(5) + \frac{1}{4}y(4) = -\frac{1}{4}$$

- **Assuming Model B right**

$$y(t) = (1 + \frac{1}{2}Z^{-1})e(t)$$

is an MA(1) process in canonical representation.
Since we're dealing with a k=1 prediction :

$$\hat{y}(t|t-1) = \frac{C(Z) - A(Z)}{C(Z)} y(t)$$

$$\hat{y}(t|t-1) = \frac{1 + \frac{1}{2}Z^{-1} - 1}{1 + \frac{1}{2}Z^{-1}} y(t)$$

$$\hat{y}(t|t-1) = -\frac{1}{2}\hat{y}(t-1|t-2) + \frac{1}{2}y(t-1)$$

Substituting the data points :

$$\hat{y}(6|5) = -\frac{1}{2}\hat{y}(5|4) + \frac{1}{2}y(5) = -\frac{1}{2}\hat{y}(5|4) - \frac{1}{4}$$

To compute $-\frac{1}{2}\hat{y}(5|4)$ we need to go back to the **initial condition** to compute all terms up to time =5:
-$\hat{y}(2|1) = -\frac{1}{2}\hat{y}(1|0) + \frac{1}{2}y(1)$ by making the assumption that $-\frac{1}{2}\hat{y}(1|0) = m_y \rightarrow \frac{1}{2}$
-$\hat{y}(3|2) = -\frac{1}{2}\hat{y}(2|1) + \frac{1}{2}y(2) = 0$
-$\hat{y}(4|3) = -\frac{1}{2}\hat{y}(3|2) + \frac{1}{2}y(3) = -\frac{1}{4}$
-$\hat{y}(5|4) = -\frac{1}{2}\hat{y}(4|3) + \frac{1}{2}y(4) = \frac{1}{8}$
-$\hat{y}(6|5) = -\frac{1}{2}\hat{y}(5|4) + \frac{1}{2}y(5) = -\frac{5}{16}$
Our final prediction for model B is $\hat{y}(6|5) = -\frac{5}{16}$ which depends on the initial condition made assuming that $-\frac{1}{2}\hat{y}(1|0) = m_y$. It the choice of the initial condition important? **If the system is asymptotically stable, and N is big the initial condition is not important as it will vanish.**

3.4.3 Example 3 - ARMAX & ARX

$$y(t) = (Z + 6Z^{-1})u(t-2) + \frac{2}{3 + \frac{3}{2}Z{-1}}\eta(t-1), \eta \sim WN(0,1)$$

Find predictor from data and the corresponding error with its variance.
u(t-2) $\rightarrow$ k=2
Canonical form for ARMA part

$$\frac{2}{3(1+\frac{1}{2}Z^{-1})}Z^{-1}\eta(t)$$

So

$$e(t) = \frac{2}{3}\eta(t-1), e(t) \sim WN(0, \frac{4}{9}$$

$$\frac{1}{1+\frac{1}{2}Z^{-1}}e(t)$$

Substituting in the original process:

$$y(t) = (Z + 6Z^{-1})u(t-2) + \frac{1}{1+\frac{1}{2}Z^{-1}}e(t), e(t) \sim WN(0, \frac{4}{9})$$

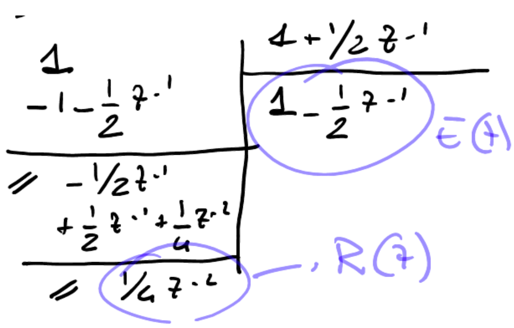
Is the term $(Z + 6Z^{-1})u(t-2)$ also in canonical representation? Wrong question, there is nothing we can do about it!
We need the form :

$$y(t) = \frac{B(Z)}{A(Z)}u(t-k) + \frac{C(Z)}{A(Z)}e(t)$$

So rewriting :

$$y(t) = \frac{(2+6Z{-1})(1+\frac{1}{2}Z^{-1})}{(1+\frac{1}{2}Z^{-1})}u(t-2) + \frac{1}{1+\frac{1}{2}}e(t)$$

Using a k-steps long division $\frac{C(Z)}{A(Z)}$:



$$\hat{y}(t|t-2) = \frac{(2+6Z^{-1})(1+\frac{1}{2}Z^{-1})(1-\frac{1}{2}Z-1)}{1}u(t-2) + \frac{\frac{1}{4}Z^{-2}}{1}y(t)$$

$$\boxed{\hat{y}(t|t-2) = 2u(t-2) + 6u(t-3) - \frac{1}{2}u(t-4) - \frac{3}{2}u(t-5) + \frac{1}{4}y(t-2)}$$

No old prediction is used → process is **ARMAX(1,0,2+2)** → **ARX(1,4)** model.

$$\boxed{\epsilon = E(Z)e(t) = (1 - \frac{1}{2})Z^{-1}e(t)}$$

The variance of ϵ :

$$var[\epsilon(t)] = (1 + \frac{1}{4}) \cdot \frac{4}{9} = \frac{5}{4}\frac{4}{9} = \frac{5}{9}$$

3.4.4 Example ARMA with non-zero mean

$$y(t) = e(t) + 4e(t-1), e \sim WN(1,1)$$

Compute $\hat{y}(t|t-1)$ and $\hat{y}(t|t-2)$ from data.
Canonical form representation

$$y(t) = (1+4Z^{-1})e(t) \rightarrow y(t) = (1+4Z^{-1})[4 \cdot \frac{1+\frac{1}{4}Z^{-1}}{1+4Z^{-1}}]e(t)$$

Getting the new $\eta(t)$:

$$\eta(t) = 4e(t)$$

$$m_\eta = E[\eta(t)] = E[4e(t)] = 4$$

$$var[\eta] = E[(\eta(t)-4)^2] = E[(4e(t)-4)^2] = 16E[(e(t)-1)^2] = 16$$

Canonical form :

$$\boxed{y(t) = (1 + \frac{1}{4}Z^{-1})\eta(t)}$$

$$\boxed{\eta(t) \sim WN(4,16)}$$

Method 1

De-biasing technique:

$$\tilde{y}(t) = y(t) - m_y$$

$$\tilde{\eta}(t) = \eta(t) - m_\eta$$

Mean of y:

$$E[y(t)] = E[(\eta(t) + \frac{1}{4}\eta(t-1))] \rightarrow m_y = \frac{5}{4}m_\eta = 5$$

So:

$$\tilde{y}(t) = y(t) - 5$$

$$\tilde{\eta}(t) = \eta(t) - 4 \rightarrow \tilde{\eta} \sim WN(0, 16)$$

Obtaining:

$$\tilde{y} + 5 = (\tilde{\eta}(t) + 4) + \frac{1}{4}(\tilde{\eta}(t-1) + 4)$$

$$\boxed{\tilde{y} = \tilde{\eta}(t) + \frac{1}{4}\tilde{\eta}(t-1)}$$

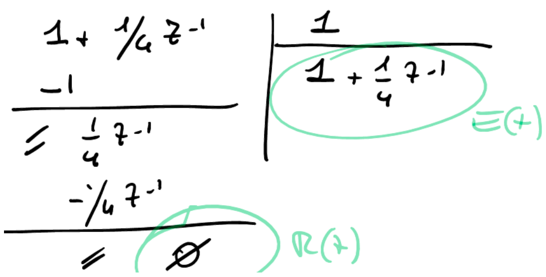
Now we can compute the predictions for $\tilde{y}$ for k=1:

$$\hat{\tilde{y}}(t|t-1) = \frac{(1 + \frac{1}{4}Z^{-1}) - 1}{(1 + \frac{1}{4}Z{-}1)}\tilde{y}(t)$$

$$\boxed{\hat{\tilde{y}}(t|t-1) = \frac{\frac{1}{4}}{(1 + \frac{1}{4}Z{-}1)}\tilde{y}(t-1)}$$

$$\boxed{\epsilon(t) = \tilde{\eta}(t) = 16}$$

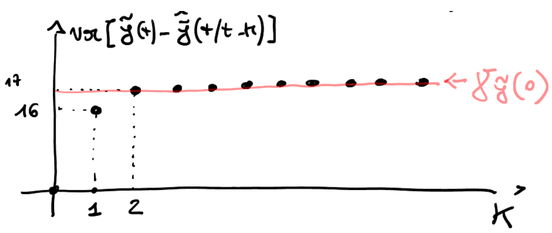
Now we can compute the prediction for $\tilde{y}$ for k=2:



Which means that

$$\boxed{\hat{\tilde{y}}(t|t-2) = 0}$$

Because MA(1) process has a **finite memory** of 1-step only! So $\tilde{\epsilon}(t) = \tilde{y}(t) - \hat{\tilde{y}}(t|t-2) = \tilde{y})(t)$ so the $var[\tilde{\epsilon}(t)] = var[\tilde{y}(t)] = (1 + \frac{1}{16}) \cdot 16 = 17$



We need to go back to the original process because $\hat{y}(t|t-1) \neq \hat{\tilde{y}}(t|t-1)$:
- for **k=1**

$$\hat{y}(t|t-1) - 5 = \frac{\frac{1}{4}}{1 + \frac{1}{4}Z^{-1}}(y(t-1) - 5)$$

$$\hat{y}(t|t-1) = \frac{\frac{1}{4}}{1 + \frac{1}{4}Z^{-1}}y(t-1) + 5 - 5\frac{\frac{1}{4}}{1 + \frac{1}{4}Z^{-1}}$$

The term $5\frac{\frac{1}{4}}{1+\frac{1}{4}Z^{-1}}$ can be resolved by applying the **frequency response theorem** by taking in account that 5 is a sinusoid with frequency $w = 0$:

$$\frac{\frac{1}{4}}{1 + \frac{1}{4}e^{0j}} \cdot 5 = 1$$

so :

$$\boxed{\hat{y}(t|t-1) = \frac{\frac{1}{4}}{1 + \frac{1}{4}Z^{-1}}y(t-1) + 4}$$

- for **k=2**

$$\hat{\tilde{y}}(t|t-2) = 0 \rightarrow \hat{y}(t|t-2) - 5 = 0$$

$$\boxed{\hat{y}(t|t-2) = 5}$$

$$\boxed{var[\tilde{\eta}(t)] = var[\eta(t)]}$$

Method 2

De-bias technique only on $\eta(t)$:

$$\tilde{\eta}(t) = \eta(t) - 4$$

$$y(t) = \tilde{\eta}(t) + 4 + \frac{1}{4}(\tilde{\eta}(t) + 4)$$

$$\boxed{y(t) = \tilde{\eta}(t) + \frac{1}{4}\tilde{\eta}(t-1) + 5}$$

Which can be considered as an **ARMAX** process.

-for k=1

$$y(t) = u(t-1) + (1 + \frac{1}{4}Z^{-1}\tilde{\eta}(t)), u(t) = 5\forall t$$

$u(t-1)$ is chosen arbitrarily because we need to compute $\hat{y}(t|t-1)$:

- k =1
- $B(Z) = 1$
- $C(Z) = 1 + \frac{1}{4}Z^{-1}$
- $A(Z) = 1$

$$\hat{y}(t|t-1) = \frac{1 \cdot 1}{1 + \frac{1}{4}Z^{-1}}u(t-1) + \frac{(1 + \frac{1}{4}Z^{-1}) - 1}{1 + \frac{1}{4}Z^{-1}}y(t)$$

As $u(t-1) = 5$ the system taking it as input can be simplified again using FR. Theorem :

$$\boxed{\hat{y}(t|t-1) = \frac{\frac{1}{4}}{1 + \frac{1}{4}Z^{-1}}y(t-1) + 4}$$

Which is the same result as for the first method.

-for k=2

$$y(t) = u(t-2) + (1 + \frac{1}{4}Z^{-1})\tilde{\eta}(t), u(t) = 5\forall t$$

Making a 2 step long division $\frac{C(Z)}{A(Z)}$:
-$E(Z) = 1 + \frac{1}{4}Z^{-1}$ -$R(Z) = 0$

$$\hat{y}(t|t-2) = \frac{1 \cdot (1 + \frac{1}{4}Z^{-1})}{1 + \frac{1}{4}Z^{-1}}u(t-2) + 0$$

$$\boxed{\hat{y}(t|t-2) = 5}$$

Which again is the same result as in the first method.

4 Chapter 4 : Identification

The focus of MIDA 1 are **parametric** identification or learning techniques. They are the most used and popular identification techniques but many non-parametric techniques are essential for identification (ex: state-space identification ,spectrum estimation ,unsupervised learning...)
Any **parametric identification technique** is based on a **five step approach**:

1. **Experiment design & data collection**
   This step deals with **designing** the experiment, selecting the **length N** of the dataset and **data pre-processing**.

2. **Selection of a class of parametric models**
   This steps deals with the selection of **class** of parametric models $m(\theta)$ where θ is the unknown parameter vector. Our focus will be on :
   -discrete time
   -dynamic
   -linear
   -time-invariant
   systems. As already seen **ARMAX & ARMA** are the most general class of models for these systems.

3. **Selection of a performance index**
   A function $J(\theta) \geq 0$ that tells the **ordering** of different models. The performance index assesses the **quality** of a model.
   The **prediction error method** is the choice for our performance index:

$$\boxed{J_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|t-1,\theta))^2}$$

   that represents the **sample variance** of the prediction error computed on the available dataset of length N.
   The P.E.M assumes that the ability of a model to make a good prediction of the future is a good **quality index** for the model.

4. **Optimization**
Optimization consists in **minimizing** $J(\theta)$ with respect to θ :

$$\hat{\theta} = argmin_{\theta}\{J(\theta)\}$$

so that

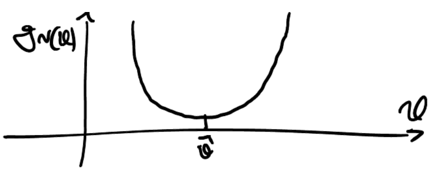
$$m(\hat{\theta})$$

is the **optimal model** on the class of models $m(\theta)$.

$$J_N(\theta) = R^{n_\theta} \to R^+$$

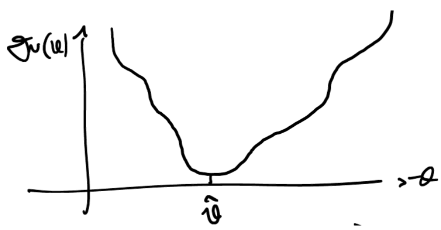
In optimisation 3 different situations can be found:

- $J(\theta)$ is **quadratic** function of θ



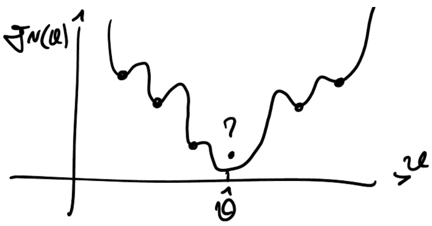
  J_N is a **quadratic** function of θ : in this case it's usually easy to find the **global minimum** explicitly.
  AR & ARX models are of this kind.

- $J(\theta)$ is **not** a quadratic function , **no local minima**



  In this case the function has no local minima so the **unique solution** is guaranteed to be found using an **iterative algorithm**.

- $J(\theta)$ is **not** quadratic, **with local minima**



In this case the function has local minima so using an **iterative algorithm** is the best way to find the unique solution which is **not guaranteed** to be found.
ARMAX & ARMA models are of this kind.

5. **Validation**
The validation step checks if $m(\hat{\theta})$ can be considered a **valid** model for our purposes. Usually a technique called **cross-validation** is used.

4.1 Identification of ARX models

Given an available dataset of length N :

$$\{u(1), u(2), ..., u(N)\}$$

$$\{y(1), y(2), ..., y(N)\}$$

An the model class **ARX(m,p+1)**:

$$y(t) = \frac{B(Z)}{A(Z)}u(t-1) + \frac{1}{A(Z)}e(t), e(t) \sim WN(0, \lambda^2)$$

where $\theta = \left[a_1...a_m b_0...b_p\right]^T$ is the **parameter vector** of dimension $n_\theta = m+p+1$.

Remark

Using **k=1** is not a restriction but the **most general** case of an ARX. If the system has $k > 1$ we will find out during the identification process.

4.1.1 Loss function: Least Squares

The loss function for the ARX models is the **P.E.M.**:

$$\boxed{J_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|t-1, \theta))^2}$$

The predictor for the model , deriving it from the general ARMAX model, is :

$$\hat{y}(t|t-1;\theta) = \frac{B(Z)}{1} \cdot 1 \cdot u(t-1) + (1 - A(Z))y(t)$$

$$\boxed{\hat{y}(t|t-1;\theta) = b_0 u(t-1) + ... + b_p u(t-p-1) - a_1 y(t-1) - ... - a_m y(t-m)}$$

where we can define the **data vector**:

$$\phi = \left[-y(t-1), ... - y(t-m), u(t-1), ...u(t-p-1)\right]^T$$

so $\hat{y}(t|t-1) = \phi(t)^T\theta$, a linear function of θ. Substituting in the loss function:

$$J_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \phi(t)\theta)^2$$

A **quadratic** function is obtained so the **unique solution** can be found explicitly using a minimization method.To find the minimum we differentiate wrt to the parameter vector θ :

$$\frac{\partial J_N(\theta)}{\partial \theta} = 0$$

$$\frac{\partial J_N(\theta)}{\partial \theta} = \frac{2}{N}\sum_{t=1}^{N} \phi(t)(y(t) - \phi(t)^T\theta)$$

$$(\sum_{t=1}^{N} \phi(t)\phi(t)^T)\theta = \sum_{t=1}^{N} y(t)\phi(t)$$

Assuming that the n_θ x n_θ matrix $\sum\limits_{t=1}^{N} \phi(t)\phi(t)^T$ matrix is **non singular** and thus **invertible**:

$$\boxed{\hat{\theta}_N = (\sum_{t=1}^{N} \phi(t)\phi(t)^T)^{-1}(\sum_{t=1}^{N} y(t)\phi(t))}$$

This is the **explicit** solution of the ARX identification problem also known as **Least Squares**

4.1.2 Example

Consider a dataset of length N=10 and

$$y(t) = \frac{b}{1+aZ^{-1}}u(t-1) + \frac{1}{1+aZ^{-1}}e(t), e(t) \sim WN(0,\lambda^2)$$

an ARX(1,1) general model class. Assuming that the process is in canonical representation ($|a| < 1$ must hold). The predictor of the model is :

$$\hat{y}(t|t-1) = \frac{B(Z)}{1}u(t-1) + \frac{1-A(Z)}{1}e(t)$$

$$\boxed{\hat{y}(t|t-1) = bu(t-1) - ay(t-1)}$$

and $\theta = [a,b]^T$.

Method 1

The loss function is :

$$J_{10}(\theta) = \frac{1}{10}\sum_{t=1}^{10}(y(t) - bu(t-1) + ay(t-1))^2$$

Remark

Since we don't have data points for t=0 , starting at time t=1 doesn't allow us to compute $-bu(0) + ay(0)$ so a modified version of the performance index is used:

$$\boxed{J_N(\theta) = \frac{1}{N-h}\sum_{t=h+1}^{N}(y(t) - \hat{y}(t|t-1))^2}$$

where $h = max\{m, p+1\}$

In our case $h = max(1,1) = 1$ so

$$J_{10}(\theta) = \frac{1}{9}\sum_{t=2}^{10}(y(t) - bu(t-1) + ay(t-1))^2$$

To obtain the best parameter vector $\rightarrow \frac{\partial J_{10}(\theta)}{\partial \theta} = 0$ where $\theta = [a,b]^T$:

$$\frac{\partial J_{10}(\theta)}{\partial \theta} = \begin{cases} \frac{\partial J_{10}(\theta)}{\partial a} = \frac{2}{9}\sum_{t=2}^{10}(y(t) - bu(t-1) + ay(t-1)) \cdot y(t-1) = 0 \\ \frac{\partial J_{10}(\theta)}{\partial b} = \frac{2}{9}\sum_{t=2}^{10}(y(t) - bu(t-1) + ay(t-1)) \cdot (-u(t-1)) = 0 \end{cases}$$

Which can be rewritten as :

$$\begin{bmatrix} \sum_{t=2}^{10} y(t-1)^2 & -\sum_{t=2}^{10} y(t-1)u(t-1) \\ -\sum_{t=2}^{10} y(t-1)u(t-1) & \sum_{t=2}^{10} u(t-1)^2 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} -\sum_{t=2}^{10} y(t-1)y(t) \\ \sum_{t=2}^{10} y(t)u(t-1) \end{bmatrix}$$

so

$$\begin{bmatrix} \hat{a}_{10} \\ \hat{b}_{10} \end{bmatrix} = \begin{bmatrix} \sum_{t=2}^{10} y(t-1)^2 & -\sum_{t=2}^{10} y(t-1)u(t-1) \\ -\sum_{t=2}^{10} y(t-1)u(t-1) & \sum_{t=2}^{10} u(t-1)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_{t=2}^{10} y(t-1)y(t) \\ \sum_{t=2}^{10} y(t)u(t-1) \end{bmatrix}$$

Method 2

Consider the predictor $\hat{y}(t|t-1) = bu(t-1) + ay(t-1)$ and the available data. Assuming that the predictor makes the **perfect prediction** on the measured data :

$$\begin{cases} -ay(1) + bu(1) = y(2) \\ -ay(2) + bu(2) = y(3) \\ \ldots \\ -ay(9) + bu(9) = y(10) \end{cases}$$

Which can be separated into two matrices:

$$\Phi = \begin{bmatrix} -y(1) & u(1) \\ -y(2) & u(2) \\ \ldots & \ldots \\ -y(9) & u(9) \end{bmatrix} \quad Y = \begin{bmatrix} y(2) \\ y(3) \\ \ldots \\ y(10) \end{bmatrix}$$

Obtaining a **linear** system of 9 equations and 2 unknowns :

$$\boxed{\Phi \cdot \theta = Y}$$

Remark

- **Undetermined linear system**
  Number of unknowns ¿ number of equations → **infinite solutions**
- **Square linear system**
  Number of unknowns = number of equations → **1! solution**
- **Over determined linear system**
  Number of unknowns ¡ number of equations → **No solutions!**

Unfortunately our case is the last one with no solutions! In this case a **least squares** (**approximate**) solution can be found using the square matrix :
$\Phi\theta = Y \rightarrow \Phi^T\Phi\theta = \Phi^T Y$

$$\boxed{\hat{\theta} = [\Phi^T\Phi]^{-1}\Phi^T Y}$$

where $[\Phi^T\Phi]^{-1}\Phi^T = \Phi^+$ is the **pseudo inverse** of Φ. By creating the matrices above we obtain the **same** result for $\hat{\theta}$

4.1.3 Example 2

Assume a dataset of 5 points collected from an SSP with zero mean. $y(1) = \frac{1}{2}, y(2) = 0, y(3) = -1, y(4) = -\frac{1}{2}, y(5) = \frac{1}{4}$ and we want to make a prediction $\hat{y}(6|5)$

1.Build the model

We selected a class of model **AR(1)** (reason discussed further on).

$$y(t) = ay(t-1) + e(t)e(t) \sim WN(0, \lambda^2)$$

is our $m(\theta)$. Transforming $\rightarrow y(t) = \frac{1}{1-aZ^{-1}}e(t)$ which is in canonical representation if $|a < 1|$. The corresponding k=1 predictor is

$$\hat{y}(t|t-1) = \frac{C(Z) - A(Z)}{C(Z)}y(t) \rightarrow \hat{y}(t|t-1) = ay(t-1)$$

with **performance index**:

$$J_5(a) = \frac{1}{4}\sum_{t=2}^{5}(y(t) - ay(t-1))^2$$

After some easy computation we find that $J_5(a) = \frac{1}{4}[\frac{3}{2}a^2 - \frac{3}{4}a + \frac{21}{16}]$ which is the measure to be **minimized**:

$$\frac{\partial J_5(a)}{\partial a} = \frac{1}{4}[3a - \frac{3}{4}] = 0$$

Which has solution

$$\hat{a}_5 = \frac{1}{4}$$

So best model identified in AR(1) class is :

$$\boxed{y(t) = \frac{1}{1 - \frac{1}{4}Z^{-1}}e(t)}$$

2.Compute prediction

$\hat{y}(6|5) = \frac{1}{4}y(5) = \frac{1}{16}$

3. First remark

Is $\hat{a} = \frac{1}{4}$ fine?Yes , it hold the condition that $|a| < 1$.

What if $\hat{a} = 4$? In that case we have to find the **canonical form** of the system.

4. Second remark

The variance of the white noise λ^2 is not required for the model and prediction estimation. If we wanted to estimate also λ^2 and not only a (which still is the most important parameter to estimate):

$$\lambda^2 = var[e(t)] \rightarrow var[\epsilon(t)]$$

Using an **approximate estimation** using the 5 available data points :

$$\hat{\lambda_5^2} = J_5(\hat{a}_5)$$

4.2 Identification of ARMAX models

While ARX model estimation relies on *least squared method* , ARMAX model estimation relies on **maximum likelihood**.
Given the available N-long data vector **u** and **y** , the model class is $m(\theta) : y(t) = \frac{B(Z)}{A(Z)}u(t-1) + \frac{C(Z)}{A(Z)}e(t)$ where

$$A(Z) = 1 + ... + a_m Z^{-m}$$

$$B(Z) = b_0 + ... + b_p Z^{-p}$$

$$C(Z) = 1 + ... + c_n Z^{-n}$$

and working under the hypothesis that $\frac{C(Z)}{A(Z)}$ is in **canonical representation** $\rightarrow$ **ARMAX(m,n,p+1)** where k=1 is **not** a restriction but indeed the **most general case**.
Using the P.E.M. approach :

$$\hat{\theta} = argmin_\theta = \{J_N(\theta)\}$$

with performance index

$$J_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|t-1,\theta))^2$$

where the parameter vector has dimension $n_\theta = m + n + p + 1$:

$$\theta = [a_1, ..., a_m, b_0, ...b_p, c_1, ..., c_m]^T$$

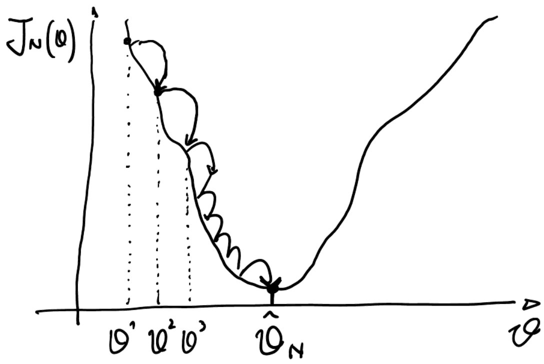
Since k=1 $\epsilon(t) = e(t) = \frac{A(Z)}{C(Z)}y(t) - \frac{B(Z)}{C(Z)}u(t-1)$ so the performance index

$$J_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(\frac{A(Z)}{c(Z)}y(t) - \frac{B(Z)}{C(Z)}u(t-1))^2$$

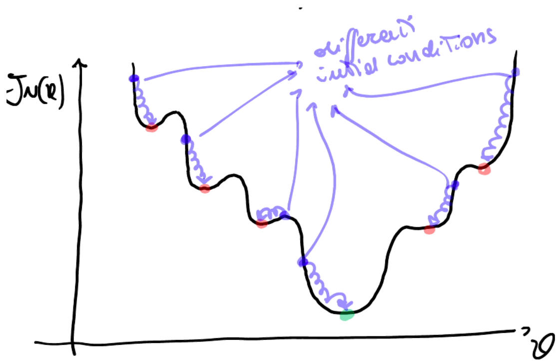
This performance index is now an issue as it kills the optimisation : C(Z) is a **non-linear** function of θ so it is not a quadratic function of θ. So the **minimization** requires an **iterative approach**.

4.2.1 Loss function :Maximum Likelihood Method

The **iterative** loss function of ARMAX models starts from an initial condition $\theta^1 \to \theta^2 ... \to \hat{\theta}_N$ until a final solution is reached



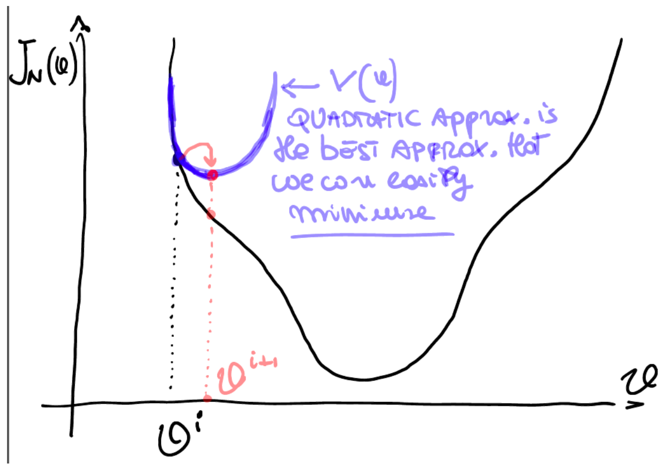
Unfortunately usually $J_N(\theta)$ has lots of **local minima**:



To attenuate this disturbance many **different** initial conditions θ^1 should be used , each time trying to reach the best solution. The problem is that there is no **theoretical guarantee** that solution converges to the global minimum.
The key problem for iterative methods is building the **step** :

$$\theta^i \to \theta^{i+1}$$

The basic idea is to compute the **local quadratic approximation** of $J_N(\theta)$ around the point θ^i and **minimize the local approximation** using the **Newton method**



Using **Taylor's expansion** around θ^i:

$$\gamma(\theta) = J_N(\theta^i) + \left[\frac{\partial J_N(\theta)}{\partial \theta}\bigg|_{\theta^i} \right] (\theta - \theta^i) + \frac{1}{2}(\theta - \theta^i)^T \left[\frac{\partial^2 J_N(\theta)}{\partial^2 \theta}\bigg|_{\theta^i} \right] (\theta - \theta^i)...$$

Minimizing the function $\gamma(\theta) \rightarrow \frac{\partial \gamma(\theta)}{\partial \theta} = 0$:

$$\left[\frac{\partial J_N(\theta)}{\partial \theta}\bigg|_{\theta^i} \right] + \frac{1}{2} \cdot 2 \left[\frac{\partial^2 J_N(\theta)}{\partial^2 \theta}\bigg|_{\theta^i} \right] (\theta - \theta^i) = 0$$

$$\boxed{\theta = \theta^i - \left[\frac{\partial^2 J_N(\theta)}{\partial^2 \theta}\bigg|_{\theta^i} \right]^{-1} \cdot \left[\frac{\partial J_N(\theta)}{\partial \theta}\bigg|_{\theta^i} \right]}$$

Where the first is matrix is the inverse of the second order derivative (**Hessian Matrix**) of $J_N(\theta)$ around θ^i and the second matrix is the first order (**Gradient Vector**) of $J_N(\theta)$ around θ^i.

- **Gradient vector**

$$\frac{\partial J_N(\theta)}{\partial \theta} = \frac{2}{N} \sum_{t=1}^{N} \epsilon(t) \cdot \frac{\partial \epsilon(t)}{\partial \theta}$$

- **Hessian matrix**

$$\frac{\partial^2 J_N(\theta)}{\partial^2 \theta} = \frac{2}{N} \sum_{t=1}^{N} \frac{\partial \epsilon(t)}{\partial \theta} \cdot \frac{\partial \epsilon(t)}{\partial \theta}^T + \frac{2}{N} \sum_{t=1}^{N} \epsilon(t) \frac{\partial^2 \epsilon(t)}{\partial \theta^2}$$

Which can be approximated to :

$$\frac{\partial^2 J_N(\theta)}{\partial^2 \theta} = \frac{2}{N} \sum_{t=1}^{N} \frac{\partial \epsilon(t)}{\partial \theta} \cdot \frac{\partial \epsilon(t)}{\partial \theta}^T$$

The approximation holds because of three reasons:

1. **Reason**

   We can compute the Hessian using only $\frac{\partial \epsilon(t)}{\partial \theta}$ without the burden of computing $\frac{\partial^2 \epsilon(t)}{\partial \theta^2}$

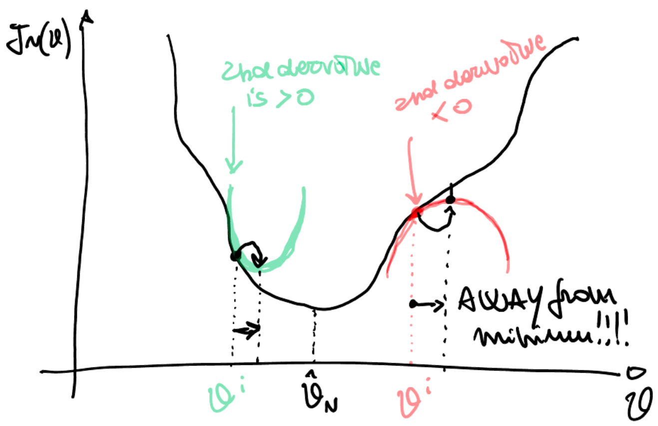
2. **Reason**

   Since $\epsilon(t, \theta) = y(t) - \hat{y}(t|t-1;\theta)$ notice that $\frac{\partial^2 \epsilon(t)}{\partial \theta^2}$ does **not** depend on y(t) but only on $\hat{y}(t|t-1;\theta) \rightarrow y(t-1), y(t-2), ...$

   Moreover if θ^i is close to the **optimum** , $\epsilon(t;\theta) \approx e(t)$. So under these assumptions $\epsilon(t;\theta^i)$ and $\left.\frac{\partial^2 \epsilon(t)}{\partial \theta^2}\right|_{\theta^i}$ are **uncorrelated** (orthogonal).

3. **Reason**

   By neglecting the second Hessian term we can guarantee that the approximation is **semi-definite positive** (≥ 0)



In conclusion the approximation **guarantees** that the updating step always goes in the **right direction!**

The final updating rule is :

$$\boxed{\theta^{i+1} = \theta^i - \left[\sum_{t=1}^{N} \frac{\partial \epsilon(t;\theta^i)}{\partial \theta} \cdot \frac{\partial \epsilon(t;\theta^i)^T}{\partial \theta} \right]^{-1} \cdot \left[\sum_{t=1}^{N} \epsilon(t;\theta) \cdot \frac{\partial \epsilon(t;\theta^i)}{\partial \theta} \right]}$$

How to calculate $\frac{\partial\epsilon(t;\theta)}{\partial\theta}$?

$$\epsilon(t,\theta) = \frac{A(Z)}{C(Z)}y(t) - \frac{B(Z)}{C(Z)}u(t-1)$$

$$\epsilon(t,\theta) = \frac{1 + a_1 Z^{-1} + ... + a_m Z^{-m}}{1 + c_1 Z^{-1} + ... + c_n Z^{-n}}y(t) - \frac{b_0 + ... + b_p Z^{-p}}{1 + c_1 Z^{-1} + ... + c_n Z^{-n}}u(t-1)$$

where $\theta = [a_1, ..., a_m, b_0, ..., b_p, c_1, ..., c_n]^T$. Deriving for each element in the parameter vector:

Parameters a

$$\frac{\partial\epsilon(t,\theta)}{\partial a_1} = \frac{Z^{-1}}{C(Z)}y(t) = \frac{1}{C(Z)}y(t-1) = \alpha(t-1)\text{defined!}$$

$$...$$

$$\frac{\partial\epsilon(t,\theta)}{\partial a_m} = \frac{Z^{-m}}{C(Z)}y(t) = \frac{1}{C(Z)}y(t-m) = \alpha(t-m)\text{defined!}$$

Parameters b

$$\frac{\partial\epsilon(t,\theta)}{\partial b_0} = -\frac{1}{C(Z)}u(t-1) = \beta(t-1)\text{defined!}$$

$$...$$

$$\frac{\partial\epsilon(t,\theta)}{\partial b_p} = \frac{Z^{-p}}{C(Z)}u(t-1) = \frac{1}{C(Z)}u(t-p-1) = \beta(t-p-1)\text{defined!}$$

Parameters c

$$(1 + c_1 Z^{-1} + ... + c_n Z^{-n})\epsilon(t,\theta) = A(Z)y(t) - B(Z)u(t-1)$$

$$Z^{-1}\epsilon(t,\theta) + C(Z)\frac{\partial\epsilon(t,\theta)}{\partial c_1} = 0$$

$$\frac{\partial\epsilon(t,\theta)}{\partial c_1} = -\frac{1}{C(Z)}\epsilon(t-1,\theta) = \gamma(t-1)\text{defined!}$$

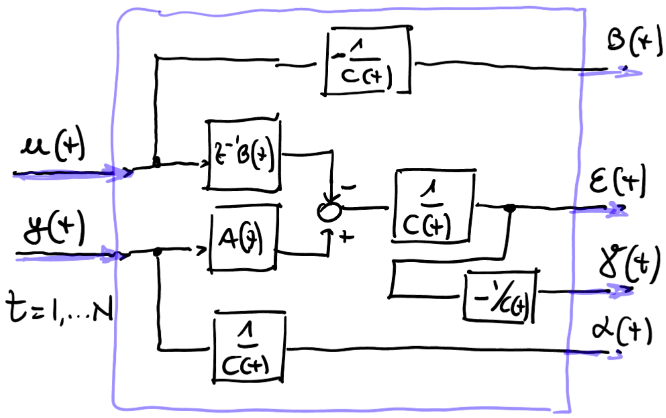
$$...$$

$$\frac{\partial\epsilon(t,\theta)}{\partial c_n} = -\frac{1}{C(Z)}\epsilon(t-n,\theta) = \gamma(t-n)\text{defined!}$$

In conclusion :

$$\boxed{\frac{\partial \epsilon(t,\theta)}{\partial \theta} = [\alpha(t-1), ..., \alpha(t-m), \beta(t-1), ..., \beta(t-p-1), \gamma(t-1), ..., \gamma(t-n)]^T}$$

We obtain a vector m+n+p+1 of **defined signals**.
A **filtering scheme** can used to compute the vector:



4.2.2 Possible issues

1. The Hessian approximation might not be **invertible** → solution : add term δI

2. The polynomial $C(Z)^i$ might have roots **outside** the unit circle. The resulting filtering is not asymptotically stable → solution : canonical form

4.2.3 Possible updating rules

There are different **updating rules** which have all the same general expression :

$$\boxed{\theta^{i+1} = \theta^i - \bigcirc \cdot \left[\frac{\partial J_N(\theta)}{\partial \theta}\right]}$$

- **Gradient method**

$$\bigcirc = \mu$$

  μ is a scalar number called **step**. Characteristics :
  + simplest method

+ correct direction **guaranteed**
- slow when near to minimum
- sensitive to choice of μ (too small = slow, too big = instability)
This method is the backpropagation rule in NN.

- **Newton method**

$$\bigcirc = \text{Inverse of hessian matrix of performance index}$$

- **Quasi- Newton method**

$$\bigcirc = \text{Inverse of} \geq \text{approximation of the Hessian}$$

This is the one seen in 4.2.1.
Has all the positive aspects of Newton method (variable step tuned to the specific point of optimisation)but can guarantee the right direction of the step.
To avoid the singularity of matrix $\sum_{t=1}^{N} \frac{\partial\epsilon(t;\theta^i)}{\partial\theta} \cdot \frac{\partial\epsilon(t;\theta^i)^T}{\partial\theta}$ usually δI is added where δ is a small number and I the identity matrix.

5 Identification analysis and complements

5.1 Asymptotic analysis of P.E.M

The system identification procedure:

1. Collect data set **u** and **y**
2. Select class of parametric models $m(\theta)$
3. Find the best in-class model $m(\hat{\theta}) : \hat{\theta} = argmin_{\theta}\{J_N(\theta\})$

Is the model $m(\hat{\theta})$ a **good** model?
To make a clean theoretical analysis we move into asymptotic quantities :

$$J_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}\epsilon(t,\theta)^2 \xrightarrow{N\to\infty} \bar{J}(\theta) = E[\epsilon(t,\theta)^2]$$

$J_N(\theta)$ is the **real** performance index which makes an average over **time**. $\bar{J}(\theta)$ is the **asymptotic** performance index which makes an average over **events**.
We can assume that $J_N(\theta) \to \bar{J}(\theta)$ if $\epsilon(t,\theta)$ is an **ergodic process** , a process where we can compute expected values over events using expected values over time.
Consider $\bar{J}(\theta)$ and assume that it has a unique global minimum :

$$\bar{\theta} : \bar{J}(\theta) \geq \bar{J}(\bar{\theta}) \forall\theta \in \Re^{n_\theta}$$

If $J_N(\theta) \to \bar{J}(\theta)$ we can assume that $\hat{\theta}_N \to \bar{\theta}$ Now lets assume that the **real system S** that has generated the dataset is within the model class : $\mathbf{S} \in m(\theta) \to a\theta^0$ exists so that $m(\theta^0) = \mathbf{S}$.

$$\text{Is } \theta^0 = \bar{\theta}?$$

In other words , is the P.E.M performance index able to select the **true** parameter θ^0.

Proof

Consider the prediction error $\epsilon(t,\theta) = y(t) - \hat{y}(t|t-1,\theta)$. Add on both sides $-\hat{y}(t|t-1,\theta^0)$:

$$\epsilon(t,\theta) - \hat{y}(t|t-1,\theta^0) = y(t) - \hat{y}(t|t-1,\theta) - \hat{y}(t|t-1,\theta^0)$$

Where $y(t) - \hat{y}(t|t-1, \theta^0)$ is the **white noise** e(t) of the true system **S**.

$$\epsilon(t, \theta) = e(t) - (\hat{y}(t|t-1, \theta^0) - \hat{y}(t|t-1, \theta))$$

Square and apply expected value:

$$E[\epsilon(t, \theta)^2] = E[e(t)^2] + E[(\hat{y}(t|t-1, \theta^0) - \hat{y}(t|t-1, \theta))^2] + 2E[e(t)(\hat{y}(t|t-1, \theta^0) - \hat{y}(t|t-1, \theta))]$$

The last term is =0 because e(t) cannot be correlated with $\hat{y}(t|t-1, \theta)$ or $\hat{y}(t|t-1, \theta^0)$. Remembering that $E[\epsilon(t, \theta)^2] = \bar{J}(\theta)$ and $E[e(t)^2] = var[e(t)] = \lambda^2$:

$$\bar{J}(\theta) = \lambda^2 + E[(\hat{y}(t|t-1, \theta^0) - \hat{y}(t|t-1, \theta))^2]$$

$$E[(\hat{y}(t|t-1, \theta^0) - \hat{y}(t|t-1, \theta))^2] = \begin{cases} \geq 0 & \text{if } \theta \neq \theta^0 \\ 0 & \text{if } \theta = \theta^0 \end{cases}$$

So $\bar{J}(\theta) \geq \lambda^2 = \bar{J}(\theta^0)$ which means that θ^0 is the global minimum of $\bar{J}(\theta)$

$$\bar{\theta} = \theta^0$$

The P.E.M provides the **true model** if $\mathbf{S} \in m(\theta)$

Remark 1

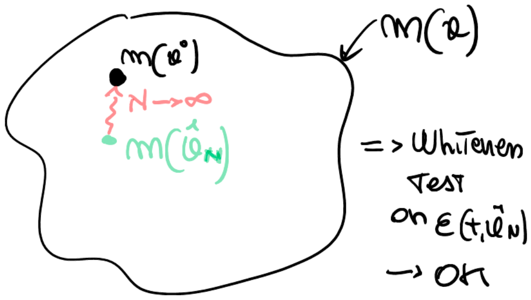
If $\mathbf{S} \in m(\theta)$ then $\epsilon(t, \theta^0) \approx \epsilon(t, \hat{\theta}_N)$ =WN.
We can use this result to check (**a-posteriori**) if the estimated model $m(\hat{\theta}_N)$ is the true model by performing a **whiteness test** on the signal:

$$\epsilon(t, \hat{\theta}), t = 1, 2, ..., N$$

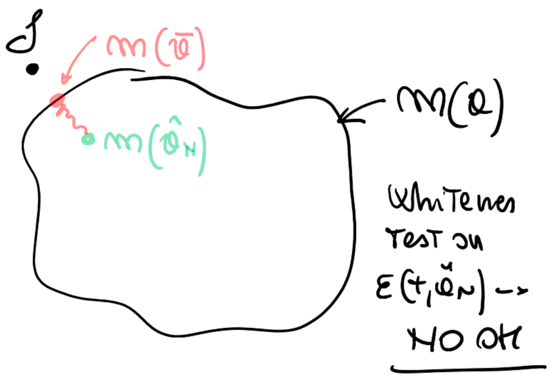
This is a practical way to make a final **validation** of the identification procedure.

Remark 2

1. $\mathbf{S} \in m(\theta) \to \bar{\theta} = \theta^0$



2. $\mathbf{S} \notin m(\theta) \to \bar{\theta} \neq \theta^0$



Best you can do is get as close as possible to S within model class $m(\theta)$

5.2 Model-order selection

In the system identification procedure we make 2 critical choices :

1. ARMA or ARMAX ?
2. Model order ? (ARMA(m,n)/ARMAX(m,n,p))

For ARMA the total order is $n = m + n$ and represents a **2-D** order problem, while for ARMAX the total order is $n = m + n + p$ and represents a **3-D** order

problem.
To simplify the problem to a **1-D** problem we make the assumption of using **balanced models** ($m \approx n \approx p$).
What is the best **global** order of n_θ?
Intuitively ,select $n_\theta \to$ find $\hat{\theta}_N$; compute $J_N(\hat{\theta}_N; n_\theta)$ (use the n_θ that provides the minimum $J_N(\hat{\theta}_N)$ which is a **totally wrong** approach because:

$$J_N(\hat{\theta}_N) \xrightarrow{n_\theta \to \infty} 0$$

Even stricter:

$$J_N(\hat{\theta}_N) \xrightarrow{n_\theta \to N} 0$$

So if the number of parameters is equal to the number of data we obtain error = 0, which is bad because we will never achieve **generalisation**. There are 3 main approaches to find the best n_θ under the following assumptions:

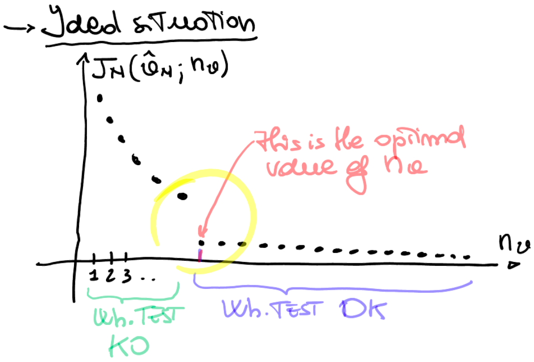
- A batch of N measured data is available
- We test increasingly model orders $n_\theta = 1, 2, 3...$
- $J_N(\hat{\theta}_N; n_\theta)$ is the performance index computed on its best parameter vector which is dependent on n_θ.

5.2.1 Discontinuity search

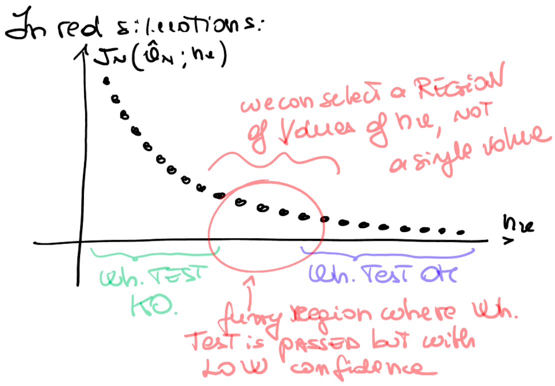
Algorithm :

- Select a value of n_θ
- Find $\hat{\theta}_N$
- Compute $J_N(\hat{\theta}_N)$
- Make **whiteness test** on $\epsilon(t, \hat{\theta}_N) = y(t) - \hat{y}(t|t-1; \hat{\theta}_N)$
- Repeat procedure for $n_\theta = 1, 2, 3...$

Ideally , by plotting the graph of performance index and model-order a **discontinuity** should be visible. The **optimal** value for n_θ is the smallest after said discontinuity.



What happens **really** is that there is no real discontinuity rather than a **region of values** which hold the optimal solution.In that region the whiteness test is still passed but with **low confidence** levels.



5.2.2 Cross validation

Suppose we have a set of N dataset points. We divide this data set in two subsets

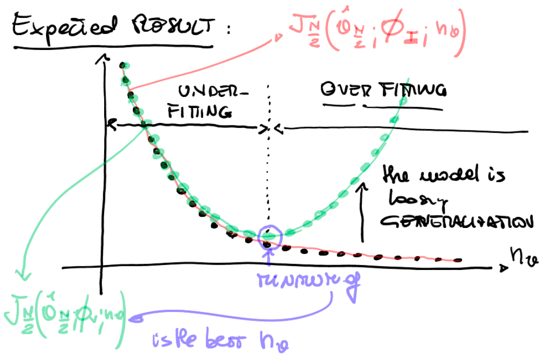
- **Identification/Learning** dataset
  $\Phi_i = 1 \rightarrow \frac{N}{2}$
- **Validation** dataset
  $\Phi_v = 1\frac{N}{2} + 1 \rightarrow N$

Procedure :

1. Define a model order n_θ
2. Find $\hat{\theta}_{\frac{N}{2}}$ by minimizing $J_{\frac{N}{2}}(\theta; \Phi_i; n_\theta)$

3. Compute $J_{\frac{N}{2}}(\hat{\theta}_{\frac{N}{2}}; \Phi_i; n_\theta)$ and $J_{\frac{N}{2}}(\hat{\theta}_{\frac{N}{2}}; \Phi_v; n_\theta)$

4. Repeat for $n_\theta = 1, 2, 3...$



In the over-fitting region the estimated model fits **not only** the specific dynamics of the system but also the **noise** which results in loosing **generality**.

Drawbacks of crossvalidation

We are forced to used just $\frac{N}{2}$ data points instead of N data points. This is not an issue for N large ($\sim$ 100.000) but is an issue for N small ($\sim$ 500)

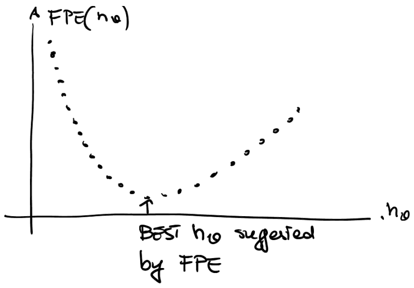
5.2.3 Estimation criteria

1. **Find prediction error (FPE)**

$$\boxed{\text{FPE}(n_\theta) = \frac{N + n_\theta}{N - n_\theta} J_N(\hat{\theta}_N, n_\theta)}$$

The first part $\frac{N+n_\theta}{N-n_\theta}$ is an **increasing function** while the second part $J_N(\hat{\theta}_N, n_\theta)$ is a **decreasing** function.

The FPE is a modification of the original performance index and has a **minimum**.



2. **Akaike Information Criterion (AIC)**

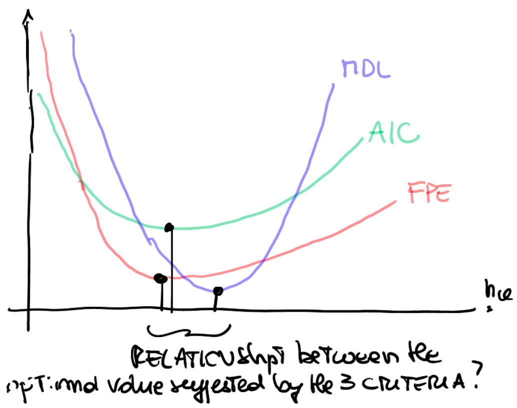
$$\boxed{AIC(n_\theta) = 2\frac{n_\theta}{N} + ln(J_N(\hat{\theta}, n_\theta))}$$

Again the first part is increasing while the second is decreasing.

3. **Minimum description length (MDL)**

$$\boxed{MDL(n_\theta = ln(N)\frac{n_\theta}{N} + ln(J_N(\hat{\theta}, n_\theta))}$$

Again the first part is increasing while the second is decreasing.



5.2.4 Comparison between FPE and AIC

$$ln(FPE) = ln(\frac{N + n_\theta}{N - n_\theta} J_N(\hat{\theta}_N, n_\theta))$$

$$ln(FPE) = ln(\frac{1 + \frac{n_\theta}{N}}{1 - \frac{n_\theta}{N}} J_N(\hat{\theta}_N, n_\theta))$$

Remark

Remind that $ln(1 + x) \approx x$ when $x = 0$

If $n_\theta << N \rightarrow \frac{n_\theta}{N} \approx 0$ so :

$$ln(1 + \frac{n_\theta}{N}) - ln(1 - \frac{n_\theta}{N}) + ln(J_N(\hat{\theta}; n_\theta)) \approx \frac{n_\theta}{N} - (-\frac{n_\theta}{N}) + ln(J_N(\hat{\theta}; n_\theta))$$

$$2\frac{n_\theta}{N} + ln(J_N(\hat{\theta}; n_\theta)) = AIC(n_\theta)$$

So if $n << N$ (always true in predicted applications):

$$\boxed{ln(FPE) = AIC}$$

Notice that if f(x) has a minimum in x_0 then also $ln(f(x))$ has a minimum in $x_0 \rightarrow \frac{d}{dx}(ln(f(x))) = \frac{1}{f(x)}f'(x)$:

$$\boxed{argmin_\theta\{FPE(n_\theta\} = argmin_\theta\{AIC(n_\theta\}}$$

So FPE and AIC provide the **same optimal** value for n_θ

5.2.5 Comparison between AIC and MDL

$$AIC(n_\theta) = 2\frac{n_\theta}{N} + ln(J_N(\hat{\theta}, n_\theta))$$

$$MDL(n_\theta = ln(N)\frac{n_\theta}{N} + ln(J_N(\hat{\theta}, n_\theta))$$

Only difference in is $2\frac{n_\theta}{N}$ vs $ln(N)\frac{n_\theta}{N}$ AIC and MDL provide **slightly** different results , if $ln(N) > 2$ AICF(FPE) suggest a **bigger** solution than MDL.

- **AR/ARX**
  If **S** the real system is an AR/ARX system , **MDL** is theoretically the correct indication of **best value**

- **ARMA/ARMAX**
  If **S** is an ARMA/ARMAX , **AIC(FPE)** should be used.

5.3 Design of experiment

Given data sets **u** and **y** and considering a generic **ARX(m,p+1)** model class we can use the P.E.M approach to solve the identification problem using Least Squares :

$$\hat{\theta}_N = (\sum_{t=1}^{N} \phi(t)\phi^T(t))^{-1}(\sum_{t=1}^{N} y(t)\phi(t))$$

We have a unique solution if $(\sum_{t=1}^{N} \phi(t)\phi^T(t))^{-1}$ is **invertible**. When is it invertible? Focusing on condition $u(t)$, we can in some situations **design** the input signal **u**= $\{u(1), ..., u(N)\}$.Define:

$$S(N) = \sum_{t=1}^{N} \phi(t)\phi^T(t)$$

$$R(N) = \frac{1}{N}S(N)$$

So :

$$\hat{\theta}_N = R(N)^{-1}(\sum_{t=1}^{N} y(t)\phi(t))$$

We will focus on the **asymptotic** value of $R(N) \xrightarrow{N\to\infty} \bar{R}$. The problem is now : when is $\bar{R}$ **invertible**?
In a generic ARX(m,p+1) $\bar{R}$ has the following structure:

$$\bar{R} = \begin{array}{c|c} \bar{R}_y & -\bar{R}_{y\mu} \\ \hline -\bar{R}_{\mu y} & \bar{R}_\mu \end{array}$$

- $\bar{R}_y$
  Is an m x m **covariance** matrix of order m-1 of the signal y(t):

$$\bar{R}_y = \begin{bmatrix} \gamma_y(0) & \gamma_y(1) & ... & \gamma_y(m-1) \\ \gamma_y(1) & \gamma_y(0)\gamma_y(1) & ... & \gamma_y(m-2) \\ ... & ... & ... & ... \\ \gamma_y(m-1) & ... & ... & \gamma_y(0) \end{bmatrix}$$

- $\bar{R}_u$
  Is a $(p+1)$ x $(p+1)$ **covariance** matrix of order p of signal u(t):

$$\bar{R}_u = \begin{bmatrix} \gamma_u(0) & \gamma_u(1) & ... & \gamma_u(p) \\ \gamma_u(1) & \gamma_u(0)\gamma_u(1) & ... & \gamma_u(p-1) \\ ... & ... & ... & ... \\ \gamma_u(p) & ... & ... & \gamma_u(0) \end{bmatrix}$$

- $\bar{R}_{yu}$
  Is a m x $(p+1)$ **cross-variance** matrix between y(t) and u(t):

$$\bar{R}_{yu} = \begin{bmatrix} \gamma_{yu}(0) & \gamma_{yu}(1) & ... & \gamma_{yu}(p) \\ \gamma_{yu}(1) & \gamma_{yu}(0)\gamma_{yu}(1) & ... & \gamma_{yu}(p-1) \\ ... & ... & ... & ... \\ \gamma_{yu}(m-1) & ... & ... & \gamma_{yu}(0) \end{bmatrix}$$

- $\bar{R}_{uy}$
  Is $\bar{R}_{uy} = \bar{R}_{yu}^T$

Remark : Lemma di Schur

Given a **block matrix**

$$M = \begin{array}{c|c} F & K \\ \hline K^T & H \end{array}$$

where F,H are **square** and **symmetric** matrices then $M > 0$ if and only if:

- $\mathbf{H > 0}$
- $\mathbf{F\text{-}KH^{-1}K^T} > 0$

The important part from Schur's Lemma is that $H > 0$ which refers to the part of $\bar{R}_u$ containing only u(t). The condition that must hold for $\bar{R}$ to be **invertible** is

$$\boxed{\bar{R}_u > 0}$$

Let's define the covariance matrix of $u(t)$ of order i (i x i matrix) :

$$\bar{R}_u^{(i)} = \begin{bmatrix} \gamma_u(0) & \gamma_u(1) & ... & \gamma_u(i-1) \\ \gamma_u(1) & \gamma_u(0) & ... & \gamma_u(i-2) \\ ... & ... & ... & ... \\ \gamma_u(i-1) & ... & ... & \gamma_u(0) \end{bmatrix}$$

A signal $u(t)$ is **persistently exciting** of order n if:

$$\boxed{\bar{R}_u^{(1)} > 0, \bar{R}_u^{(2)} > 0, ..., \bar{R}_u^{(n)} > 0, \bar{R}_u^{(n+1)} \geq 0, ...}$$

The maximum order is n so , n is the order of $\bar{R}_u^{(i)}$ such that this matrix is **invertible**.
In Conclusion:

A necessary condition for the identification process of an ARX(m,p+1) model is that the input signal u(t) must be **persistently exciting** of order at least **p+1**

Remark 1: WN

Notice that if $u(t) \approx WN(0, \lambda^2)$:

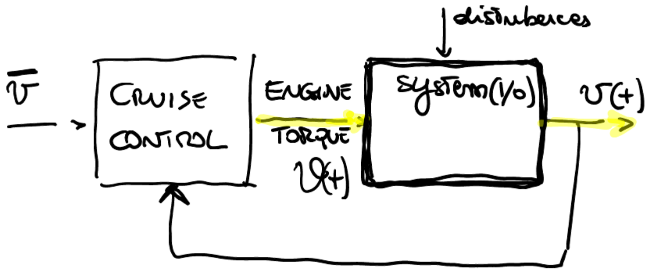
$$\bar{R}_u^{(i)} = \begin{bmatrix} \lambda^2 & 0 & ... & 0 \\ 0 & \lambda^2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & ... & ... & \lambda^2 \end{bmatrix} = \lambda^2 I^{(i)}$$

A WN is **persistently exciting** signal of order ∞

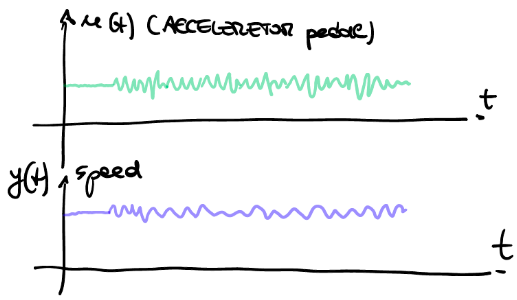
Remark 2: Sinusoid

$u(t) = Asen(wt)$ is a **persistently exciting signal** of order **2**. If possible the best choice for $u(t)$ is a WN!

Remark 3 : Practical application WN or BLWN?



Problem : design a block box model of the system, starting from an **excitation** experiment.How to design the experiment?



In principle $u(t)$ corresponding to the input of the acceleration pedal can be modelled as a WN (allows complete excitation). This is **not** a good solution though because a lot of excitation energy is used on a **useless** bandwidth. Assuming that the bandwidth of the cruise control system is about 2Hz using the **full-bandwidth** up to $w = \pi$ is **useless**. A better choice would be a **band limited white noise (BLWN)** which focuses its attention of the system identification procedure on the **relevant frequency range** $[0, \text{just beyond control bandwith}]$.

5.4 Uncertainty evaluation of a parametric identification algorithm

Assume that $\mathbf{S} \in m(\theta)$ then there exists a single parameter vector which represents the **true** system:

$$\theta^0 = \bar{\theta}$$

In practice we can only use a **finite number** of data points (**N**) which means that $\hat{\theta}_N$ depends on N.Notice that $\hat{\theta}_N$ is also a vector of **random variables** such that

$$E[\hat{\theta}_N] = \theta^0$$

But is it also true that

$$var[\hat{\theta}_N] =? = E[(\hat{\theta}_N - \theta^0)(\hat{\theta}_N - \theta^0)^T]$$

The covariance matrix of $\hat{\theta}_N$ provides an estimation of the **uncertainty** in the estimation of θ^0 using $\hat{\theta}_N$.
It can be proven that :

$$\boxed{var[\hat{\theta}_N] = \frac{1}{N}\lambda^2 \bar{C}^{-1}}$$

where

$$\lambda^2 = var[e(t)] = var[y(t) - \hat{y}(t|t-1, \theta^0)]$$
$$\bar{C} = E\left[\left.\frac{\partial \epsilon(t,\theta)}{\partial \theta}\right|_{\theta=\theta^0} \cdot \left.\frac{\partial \epsilon(t,\theta)}{\partial \theta}\right|_{\theta=\theta^0}^T \right]$$

Practical computation of $var[\hat{\theta}_N]$:

$$\lambda^2 = E[e(t)^2] \approx \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|t-1, \hat{\theta}_N)$$

$$\bar{C} \approx \frac{1}{N}\sum_{t=1}^{N}\left(\left.\frac{\partial \epsilon(t,\theta)}{\partial \theta}\right|_{\theta=\hat{\theta}_N} \cdot \left.\frac{\partial \epsilon(t,\theta)}{\partial \theta}\right|_{\theta=\hat{\theta}_N}^T \right)$$

First approximation : instead of using expected value $\rightarrow$ average over **time**.
Second approximation : instead of using $\theta^0 \rightarrow \hat{\theta}_N$

5.4.1 Interpretation of $\bar{C}$

$$\bar{J}(\theta) = E[\epsilon(t,\theta)^2]$$

$$\frac{\partial \bar{J}(\theta}{\partial \theta} = E\left[2\epsilon(t,\theta)\frac{\partial \epsilon(t,\theta))}{\partial \theta}\right]$$

$$\frac{\partial^2 \bar{J}(\theta)}{\partial \theta^2} = E\left[2\frac{\partial \epsilon(t,\theta)}{\partial \theta}\frac{\partial \epsilon(t,\theta)}{\partial \theta}^T + 2\epsilon(t,\theta)\frac{\partial^2 \epsilon(t,\theta)}{\partial \theta^2}\right]$$

If $\hat{\theta}_N = \theta^0 \rightarrow \epsilon(t,\theta) = e(t)$:

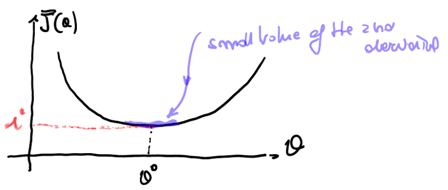
$$\left.\frac{\partial^2 \bar{J}(\theta)}{\partial \theta^2}\right|_{\theta=\theta^0} = E\left[2\left.\frac{\partial \epsilon(t,\theta)}{\partial \theta}\right|_{\theta=\theta^0}\left.\frac{\partial \epsilon(t,\theta)}{\partial \theta}\right|_{\theta=\theta^0}^T + 2e(t)\left.\frac{\partial^2 \epsilon(t,\theta)}{\partial \theta^2}\right|_{\theta=\theta^0}\right]$$

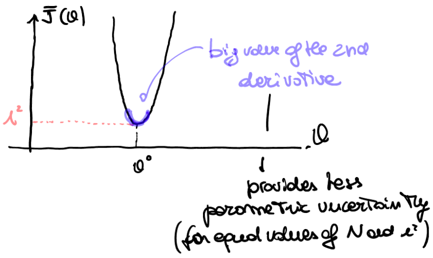
Due to **non** correlation :

$$\left.\frac{\partial^2 \bar{J}(\theta)}{\partial \theta^2}\right|_{\theta=\theta^0} = E\left[2\left.\frac{\partial \epsilon(t,\theta)}{\partial \theta}\right|_{\theta=\theta^0}\left.\frac{\partial \epsilon(t,\theta)}{\partial \theta}\right|_{\theta=\theta^0}^T\right] = 2\bar{C}$$

$$\boxed{\bar{C} = \frac{1}{2}\left.\frac{\partial^2 \bar{J}(\theta)}{\partial \theta^2}\right|_{\theta=\theta^0}}$$

- The estimation error has a **larger** variance for **larger** values of λ^2 (size of of the WN at the input of the system)
- The estimation error has a **larger** variance for **smaller** values of N.
- The estimation error has a **larger** variance if the **second derivative** of the performance index around the global minimum θ^0 is **smaller**





With this formula we can understand if the number of data is **enough** to provide the requested level of uncertainty in parameter estimation.

6 Pre-Processing

Assume that we have a time-series system identification problem and we have **raw data**:

$$\{y(1), y(2), ..., y(N)\}$$

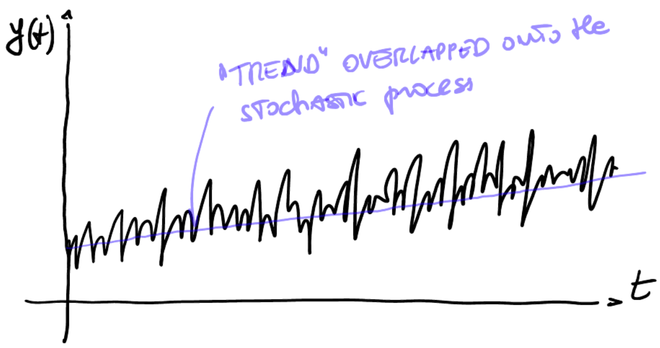
Before applying system identification techniques it is useful to check some **basic properties**:

- $E[y(t)] = m_y$ is **invariant** in time (not time-dependent).
- $\gamma_y(\tau) \xrightarrow{\tau \to \infty} 0 \to$ the process is **ergodic** so we can use **time-average** to compute **probabilistic** properties.
- No data points are **missing**

These properties are usually obtained by applying the following **pre-processing** techniques:

- **De-trend**
- **Seasonal behaviour removal**
- **Replacement of missing data**

6.1 Removing a linear trend



Model assumption :

$$y(t) = \tilde{y}(t) + kt + m$$

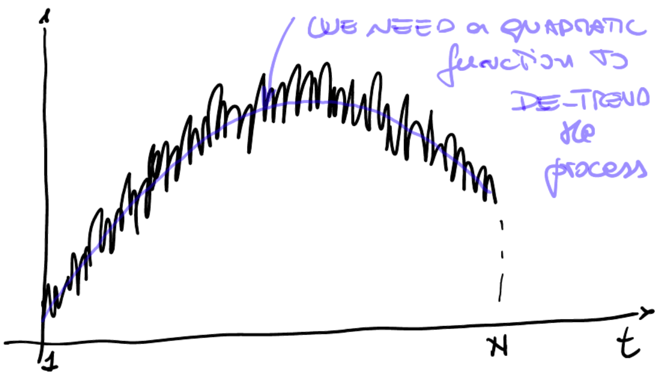
where

$$kt + m \text{ is a deterministic function of time } (\textbf{trend})$$

$$\tilde{y}(t) = \frac{C(Z)}{A(Z)} e(t) \text{ , zero-mean SSP}$$

Remark

- k=0 $\rightarrow$ trend is simply a **bias** (trend of order 0)
- Higher order trends like $k_3 t^3 + k_2 t^2 + k_1 t + m$ (cubic trend)



Assuming that the **linear trend** assumption is enough:

$$y(t) = \tilde{y}(t) + kt + m$$

$$E[y(t)] = E[\tilde{y}(t)] + E[kt + m]$$

$$\begin{cases} k \cdot 1 + m = y(1) \\ k \cdot 2 + m = y(2) \\ ... \\ k \cdot N + m = y(3) \end{cases}$$

A system with N equations and 2 variables (**over-determined)**.
By solving the problem using a **least squares** approach :

$$\phi = \begin{bmatrix} 1 & 1 \\ 2 & 1 \\ .. & .. \\ N & 1 \end{bmatrix}, \theta = \begin{bmatrix} k \\ m \end{bmatrix}, Y = \begin{bmatrix} y(1) \\ y(2) \\ ... \\ y(N) \end{bmatrix}$$

$$\phi\theta = Y \rightarrow \phi^T\phi\theta = \phi^T Y$$

$$\boxed{\hat{\theta} = \begin{bmatrix} \hat{k} \\ \hat{m} \end{bmatrix} = (\phi\phi^T)^{-1}\phi Y}$$

So:

$$\phi^T\phi = \begin{bmatrix} 1 & 2 & ... & N \\ 1 & 1 & ... & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 2 & 1 \\ ... & ... \\ N & 1 \end{bmatrix} = \begin{bmatrix} \sum_{t=1}^{N} t^2 & \sum_{t=1}^{N} t \\ \sum_{t=1}^{N} t & N \end{bmatrix}$$

$$\phi^T Y = \begin{bmatrix} 1 & 2 & ... & N \\ 1 & 1 & ... & 1 \end{bmatrix} \cdot \begin{bmatrix} y(1) \\ y(2) \\ ... \\ y(N) \end{bmatrix} = \begin{bmatrix} \sum_{t=1}^{N} t \cdot y(t) \\ \sum_{t=1}^{N} y(t) \end{bmatrix}$$

$$\boxed{\hat{\theta} = \begin{bmatrix} \hat{k} \\ \hat{m} \end{bmatrix} = \begin{bmatrix} \sum_{t=1}^{N} t^2 & \sum_{t=1}^{N} t \\ \sum_{t=1}^{N} t & N \end{bmatrix}^{-1} \begin{bmatrix} \sum_{t=1}^{N} t \cdot y(t) \\ \sum_{t=1}^{N} y(t) \end{bmatrix}}$$

The same result can be found using:

$$\begin{bmatrix} \hat{k} \\ \hat{m} \end{bmatrix} argmin_{(k,m)} = \left\{ \frac{1}{N} \sum_{t=1}^{N} (y(t) - kt - m)^2 \right\}$$

Once $\hat{k}, \hat{m}$ are found we can **de-trend** the data:

$$\boxed{\tilde{y(t)} = y(t) - \hat{k}t - \hat{m}}$$

If we wish to predict y(t):

$$\hat{y}(t+1|t) = \hat{\tilde{y(t)}}(t+1|t) + \hat{k}(t+1) + \hat{m}$$

6.2 Removing seasonal behaviour

In many practical applications some **seasonal** behaviour is overlapped to the main system dynamics. We assume a model of the **raw** dataset as follows:

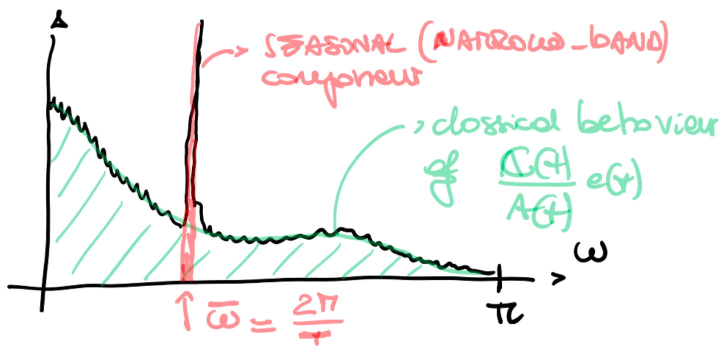
$$y(t) = \tilde{y}(t) + s(t), \text{ t=1,2,...N}$$

$$\tilde{y}(t) = \frac{C(Z)}{A(Z)}e(t)$$

$$s(t) \text{ is a periodic signal with period T} : s(t + kT) = s(t)$$

Remark

- T is usually **a-priori known** (a day, a week ,a year...)
- It is possible to have multiple seasonal behaviour overlapped, where each component can be dealt with **independently**.
- If T is **not** known a-priori, it is easy to **detect** it by a simple **FFT** of the raw signal



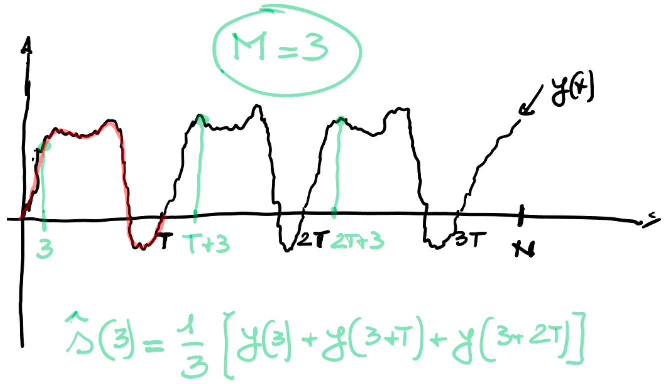
- s(t) is **not a trend** $\rightarrow$ the process (raw data) y(t) can have **both** a trend and a seasonal behaviour. **FIRST** remove the trend **THEN** the seasonal behaviour.
- If we don't remove a seasonal behaviour we end up with an **ARMA** model having a pair of **complex conjugate poles** at:

$$e^{\pm j\Omega}, \Omega = \frac{2\pi}{T}$$

The seasonal component of raw data can be estimated :

$$t = 1, 2, 3...N \ , \ M \cdot T \leq N$$

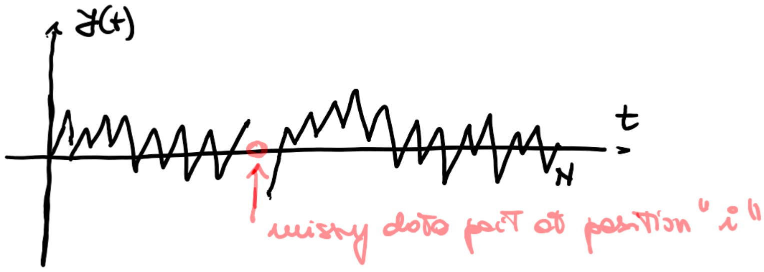
$$\boxed{\frac{1}{M} \sum_{h=0}^{M-1} y(t + hT) \text{ , t=1,2,3...N}}$$



Once $\hat{s}$ is computed we can remove it from y(t).
After modelling $\tilde{y}(t)$ with an ARMA model the prediction will be:

$$\boxed{\hat{y}(t+1|t) = \hat{\tilde{y}}(t+1|t) + \hat{s}(|t+1|_T)}$$

6.3 Missing data



Missing values are missing data points or **outliers** that are removed. We need to fill in the data point $y(i)$ with an **estimated** one $\tilde{y}(i)$.

6.3.1 Linear interpolation

$$\boxed{\tilde{y}(i) = \frac{y(i-1) + y(i+1}{2}}$$

+: simple
-: does not work if we have **batch** of missing data points.

6.3.2 Model estimation

1. compute $\tilde{y}^{(1)}(i)$ using the linear interpolation method

2. using the complete dataset (**including** $\tilde{y}^{(1)}(i)$ estimate a model for this dataset $\rightarrow m(\hat{\theta}_N^{(1)})$

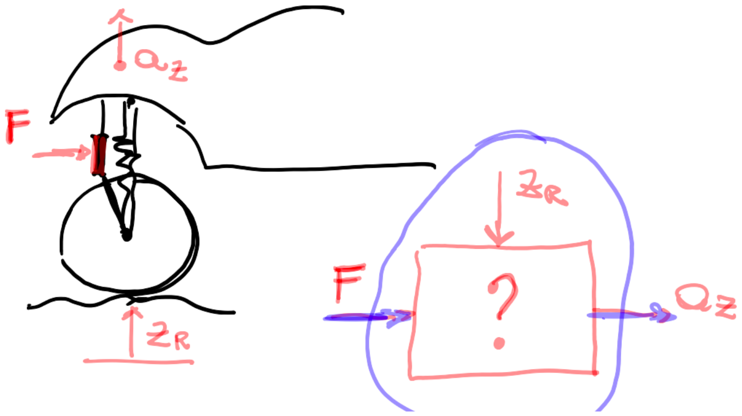
3. using the found model estimate $\tilde{y}(i)$ as

$$\boxed{\tilde{y}(i) = \hat{y}(i|i-1; \hat{\theta}_N^{(1)})}$$

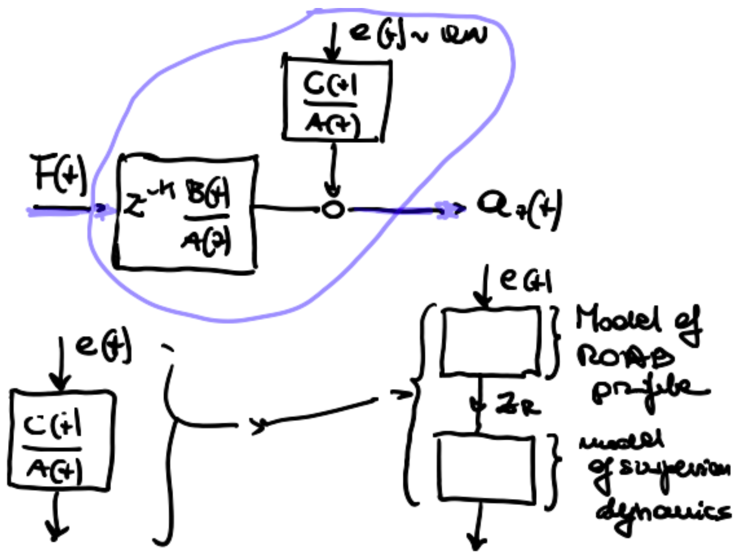
+:works also with a **batch** of missing data
-:more complex

Remark/Example

I/O systems are the most typical situations where system identification techniques are used:



In this example $\mathbf{F}= F(1), ..., F(N)$ is the input signal and $\mathbf{a}= a_z(1), ..., a_z(N)$ the output signal. We can estimate an **ARMAX** model:



Neglecting the term e(t) (model of road profile and dynamics of suspension in this case) by taking in account only the input signal we end up with the **wrong** model which leads to the **wrong** description of the dynamics.