



UNIVERSITÉ
DE LORRAINE



POLYTECH[®]
NANCY

*Modèles paramétriques
de signaux aléatoires*

Hugues GARNIER

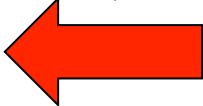
hugues.garnier@univ-lorraine.fr

Sommaire de l'EC de TdS

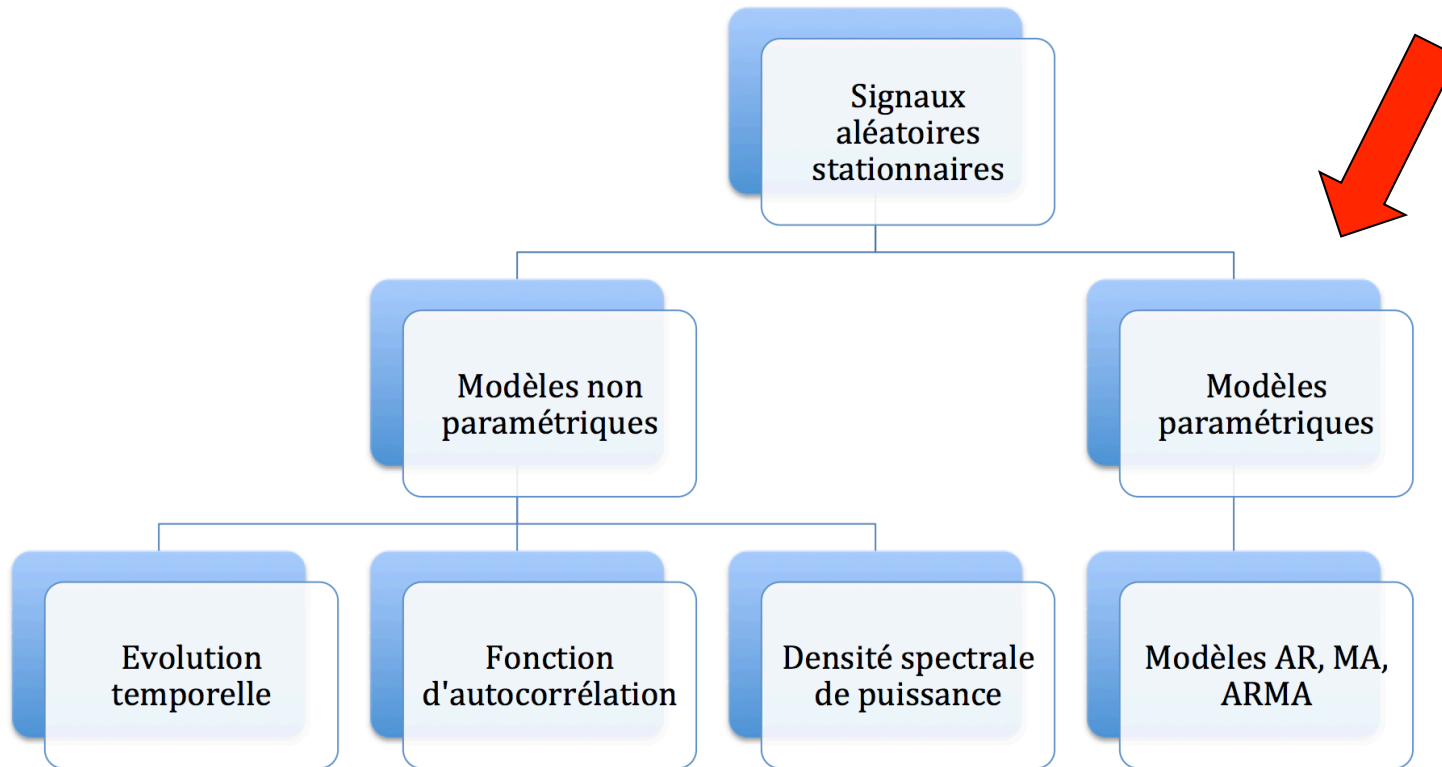
A. Analyse et traitement de signaux déterministes

- I. Rappels sur la théorie de l'échantillonnage
- II. Modélisation non paramétrique : *spectres*
- III. Méthodes d'estimation via la transformée de Fourier rapide (FFT)
- IV. Filtrage linéaire RIF et RII

B. Analyse et traitement de signaux aléatoires

- I. Signaux aléatoires
- II. Modélisation non paramétrique : *autocorrélation et spectre de puissance*
- III. Modélisation paramétrique : *modèles AR, MA, ARMA* 
- IV. Méthodes d'estimation : moindres carrés, ...

Modélisation des signaux aléatoires stationnaires



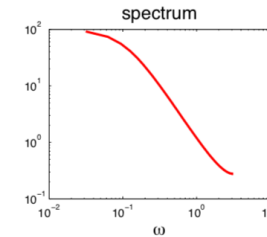
Factorisation spectrale

- Tout spectre de puissance non nul et d'amplitude finie peut être factorisé

$$\Phi_y(f) = |H(f)|^2 \times \sigma_e^2$$

$$\Phi_y(f) = H(f)H^*(f) \times \sigma_e^2$$

Factorisation spectrale



$$\Phi_y(z) = H(z)H(z^{-1}) \times \sigma_e^2 \quad \text{avec } z = e^{j2\pi \frac{f}{f_e}} \quad f \in \left[-\frac{f_e}{2}, \frac{f_e}{2} \right]$$

⇒ *Un signal aléatoire stationnaire corrélé peut être modélisé comme la sortie d'un filtre linéaire avec un bruit blanc à l'entrée : **modèle paramétrique***

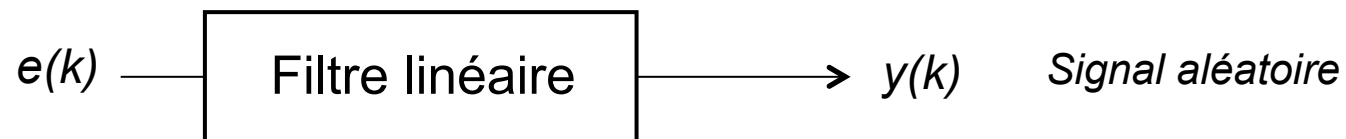


$$Y(z) = H(z) \times E(z)$$

$$y(k) = h(k) * e(k)$$

Modèles paramétriques de signaux aléatoires

- *Un signal aléatoire stationnaire ou non est souvent modélisé comme la réponse d'un filtre linéaire à un autre signal de caractéristiques plus simples*



Bruit blanc

$$e(k) \sim N(0, \sigma_e^2)$$

*Bruit de mesure, sons non voisés
comme f, v, s*

*Impulsion
de Kronecker*

$$e(k) = A \delta(k)$$

Sons explosifs comme b, d, k, t

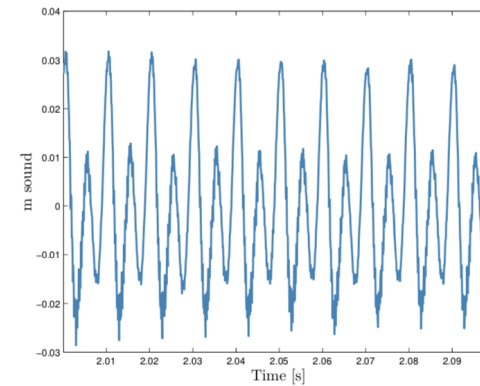
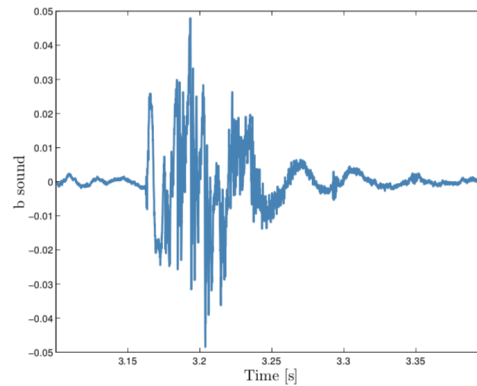
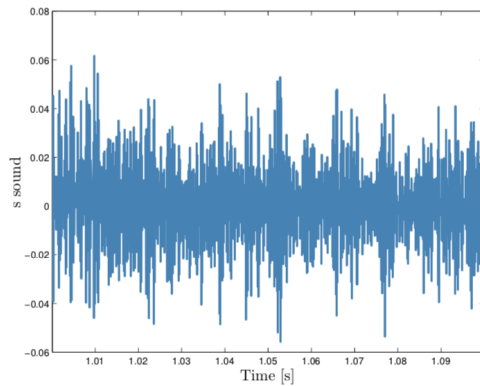
Train d'impulsions

$$e(k) = A \sum_{i=0}^n \delta(k-i)$$

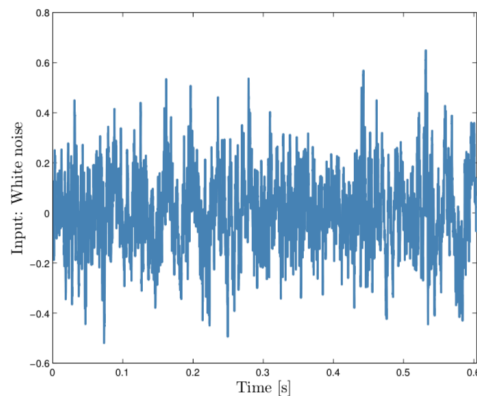
Sons voisés comme m, n, l

Types d'entrée standards : exemples

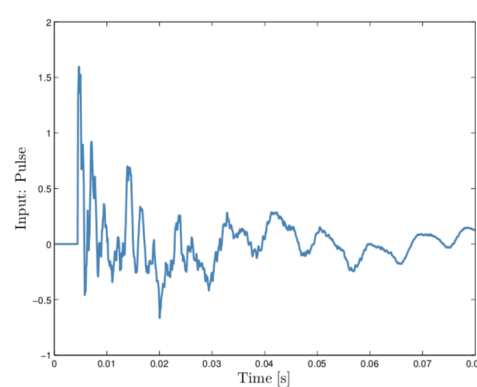
- Enregistrement des lettres s, b et m



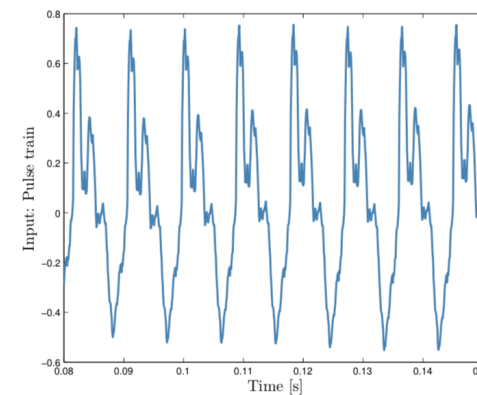
- Signal généré pour un même filtre pour les trois types d'entrée



Bruit blanc filtré



Impulsion de Kronecker filtré



train d'impulsions filtré

Forme polynomiale de l'équation aux différences Opérateur retard – Opérateur de transfert

- Soit une équation aux différences

$$y(k) + d_1 y(k-1) + \dots + d_{n_d} y(k-n_d) = e(k) + c_1 e(k-1) + \dots + c_{n_c} e(k-n_c)$$

- On définit *l'opérateur retard* q^{-1} $q^{-1}y(k) = y(k-1)$

$$y(k) + d_1 q^{-1}y(k) + \dots + d_{n_d} q^{-n_d}y(k) = e(k) + c_1 q^{-1}e(k) + \dots + c_{n_c} q^{-n_c}e(k)$$

$$\left(1 + d_1 q^{-1} + \dots + d_{n_d} q^{-n_d}\right) y(k) = \left(1 + c_1 q^{-1} + \dots + c_{n_c} q^{-n_c}\right) e(k)$$

- qui conduit à la *forme polynomiale* de l'équation aux différences

$D(q^{-1})y(k) = C(q^{-1})e(k)$

$$C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_{n_c} q^{-n_c}$$

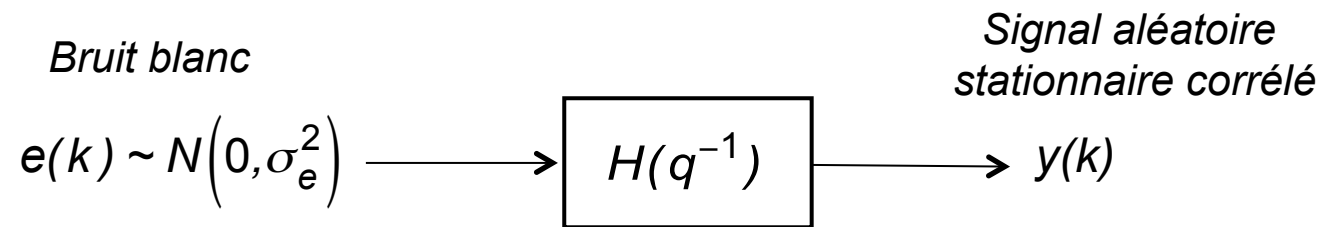
$$D(q^{-1}) = 1 + d_1 q^{-1} + \dots + d_{n_d} q^{-n_d}$$

- On définit *l'opérateur de transfert* $H(q^{-1})$ (*analogie avec* $H(z)$)

$y(k) = H(q^{-1})e(k) = \frac{C(q^{-1})}{D(q^{-1})} e(k)$

$$Y(z) = H(z)E(z) = \frac{C(z)}{D(z)} E(z)$$

Modèles paramétriques d'un signal aléatoire *stationnaire*

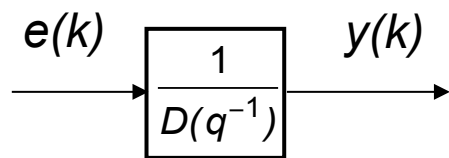


- Le filtre linéaire $H(q^{-1})$ peut avoir différentes formes qui correspondent à différents noms de modèles pour le signal aléatoire
- Il existe *trois modèles paramétriques essentiels* de signaux aléatoires stationnaires
 - **modèle AR** : modèle autorégressif
 - **modèle MA** : modèle à moyenne ajustée
 - **modèle ARMA** : modèle autorégressif à moyenne ajustée

Modèles paramétriques usuels de signaux aléatoires stationnaires

- Suivant la forme du filtre, on obtient différents types de modèles

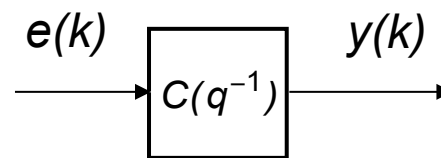
Modèle AR



$$y(k) = \frac{1}{D(q^{-1})} e(k)$$

$$H(q^{-1}) = \frac{1}{D(q^{-1})}$$

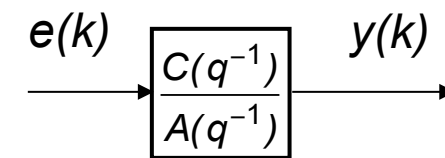
Modèle MA



$$y(k) = C(q^{-1}) e(k)$$

$$H(q^{-1}) = C(q^{-1})$$

Modèle ARMA



$$D(q^{-1})y(k) = C(q^{-1})e(k)$$

$$H(q^{-1}) = \frac{C(q^{-1})}{D(q^{-1})}$$

Modèles paramétriques usuels de signaux aléatoires stationnaires

- Déterminer la forme polynomiale des 3 équations aux différences ci-dessous et en déduire le type de modèles : AR, MA ou ARMA

$e(k)$: bruit blanc

$$y(k) = 1,5y(k-1) - 0,7y(k-2) + e(k)$$

$$y(k) = 1,5y(k-1) - 0,7y(k-2) + e(k)$$

$$y(k) = 1,5q^{-1}y(k) - 0,7q^{-2}y(k) + e(k)$$

$$(1 - 1,5q^{-1} + 0,7q^{-2})y(k) = e(k)$$

$$y(k) = \frac{1}{1 - 1,5q^{-1} + 0,7q^{-2}} e(k)$$

$$y(k) = e(k) + 0,01e(k-1) - 0,02e(k-2)$$

$$y(k) = 1,5y(k-1) - 0,7y(k-2) + e(k) + 0,01e(k-1) - 0,02e(k-2)$$

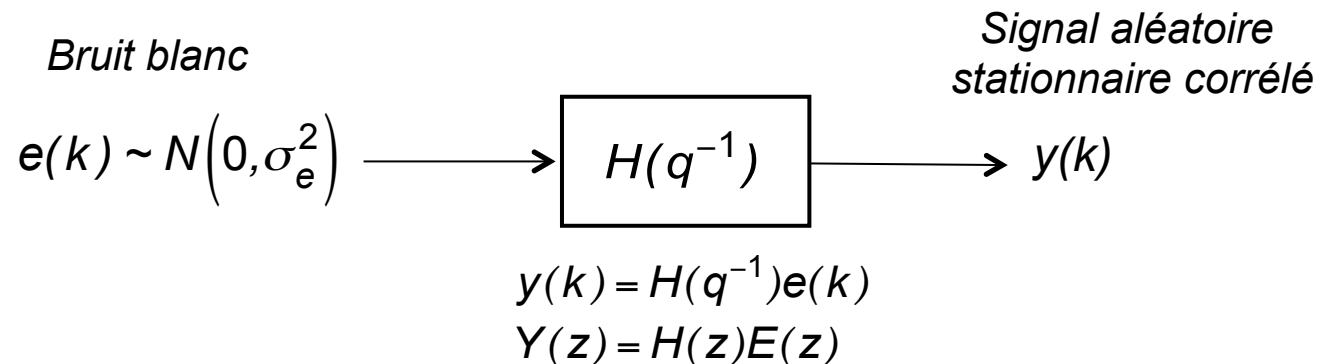
Estimation du spectre d'un signal aléatoire stationnaire - Bilan

Comment estimer le spectre de puissance d'un signal aléatoire à partir d'un nombre fini N de données ?

- Il existe différentes méthodes. Toutes ont leurs avantages et inconvénients. Elles peuvent être rassemblées dans deux grandes familles de méthodes :
 - *Méthodes basées sur un modèle non paramétrique*
 - Aucune hypothèse sur le processus n'est effectuée. On va chercher à estimer directement le spectre de puissance à partir du nombre fini N de données : périodogramme et corrélogramme
 - *Méthodes basées sur un modèle paramétrique*
 - Formulation de l'hypothèse que le signal peut être représenté par un des 3 modèles paramétriques (AR , MA , $ARMA$) dont les paramètres sont estimés à partir du nombre fini N de données. On en déduit alors par calcul le spectre de puissance

Spectre de puissance d'un signal aléatoire à partir d'un modèle paramétrique

Un signal aléatoire stationnaire corrélé peut être modélisé comme la sortie d'un filtre linéaire avec un bruit blanc à l'entrée

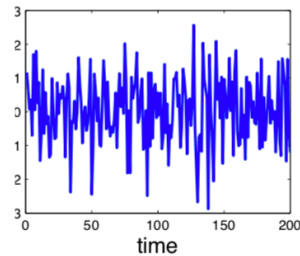


Si $H(z)$ est connu, le spectre de puissance de $y(k)$ est (*factorisation spectrale*)

$$\Phi_y(z) = H(z)H(z^{-1}) \times \sigma_e^2 \quad \text{avec } z = e^{j2\pi \frac{f}{f_e}}$$

Modèles paramétriques de signaux aléatoires et leurs modèles non paramétriques associés - Exemple

Modèles non paramétriques



$e(k) \sim N(0, \sigma_e^2)$
 bruit blanc

Modèle paramétrique

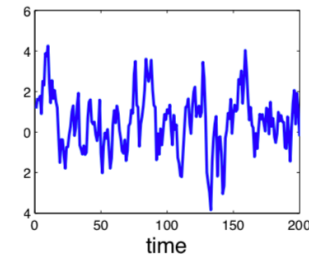
$$H(q^{-1}) = \frac{1}{1 - 0,9q^{-1}} = \frac{q}{q - 0,9} \longrightarrow y(k)$$

$$y(k) = 0,9y(k-1) + e(k)$$

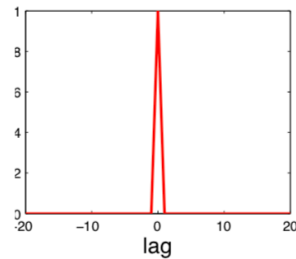
$$y(k) = \frac{1}{1 - 0,9q^{-1}} e(k)$$

$$\hat{\Phi}_x(f) = \left| \hat{H} \left(z = e^{j2\pi \frac{f}{f_e}} \right) \right|^2 \times \sigma_e^2$$

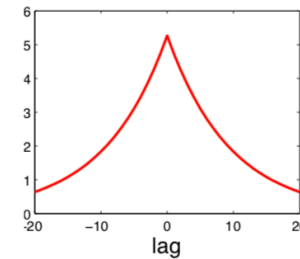
Modèles non paramétriques



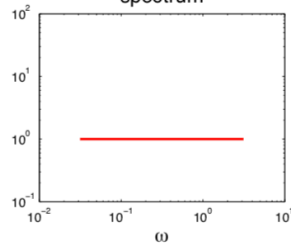
covariance



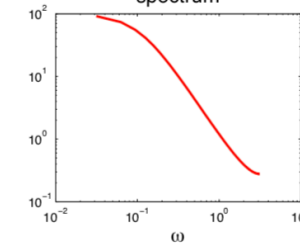
covariance



spectrum

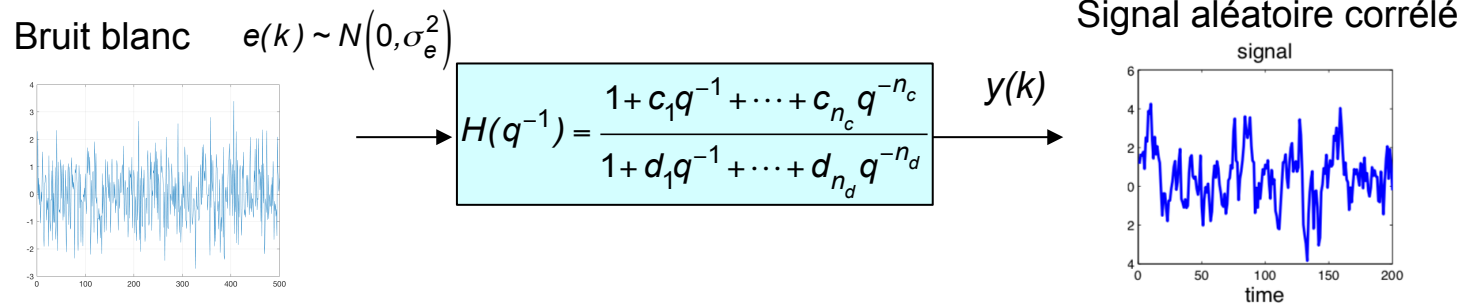


spectrum



Estimation du spectre de puissance Méthode basée sur un modèle paramétrique

- Hypothèse que le signal aléatoire peut être représenté par un des 3 modèles paramétriques : *AR, MA ou ARMA*



- Estimation du modèle paramétrique à partir d'un nombre fini de données puis calcul du spectre de puissance

$$\hat{H}(z) = \frac{\hat{C}(z)}{\hat{D}(z)} \quad \text{avec } z = e^{j2\pi \frac{f}{f_e}} \Rightarrow \quad \hat{\Phi}_y(f) = \left| \hat{H}\left(z = e^{j2\pi \frac{f}{f_e}}\right) \right|^2 \times \sigma_e^2$$

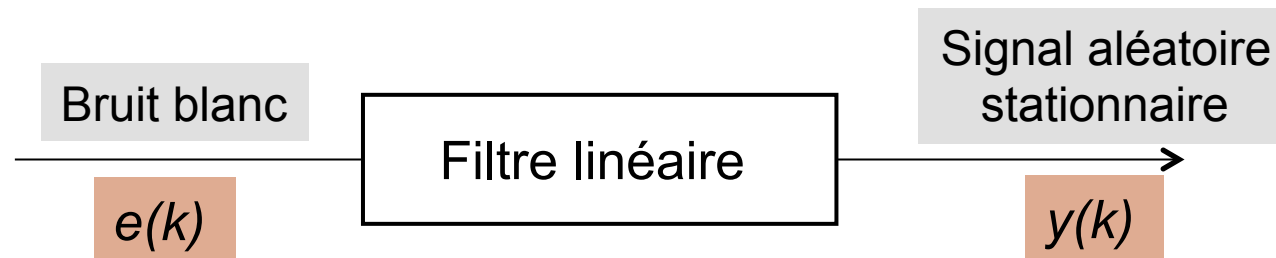
A partir de la mesure de N échantillons d'un signal aléatoire $x(k)$, comment estimer les paramètres de $H(q^{-1})$ (ou $H(z)$) ?

Sommaire de la suite consacrée à l'estimation paramétrique

- Principe de prévision
- Estimation par minimisation de l'erreur de prévision
- Estimateur fondé sur la régression linéaire (*pour modèle AR*)
- Estimateur fondé sur l'optimisation non-linéaire (*pour modèle MA et ARMA*)
- Aspects pratiques : *choix de l'ordre, ...*

Estimation paramétrique de modèles de signaux aléatoires stationnaires

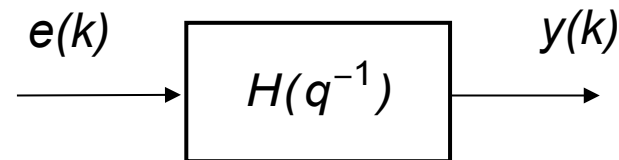
Objectif : déterminer la forme et les paramètres d'un modèle paramétrique d'un signal aléatoire stationnaire



Plusieurs choix doivent être effectués :

- quel type de modèle (AR, MA, ARMA) pour le filtre linéaire ?
- comment estimer/apprendre les paramètres du modèle choisi ?
- comment choisir l'ordre du modèle choisi

Modèle paramétrique d'un signal aléatoire stationnaire



$$y(k) = H(q^{-1})e(k)$$

$e(k)$ est un bruit blanc gaussien à temps discret

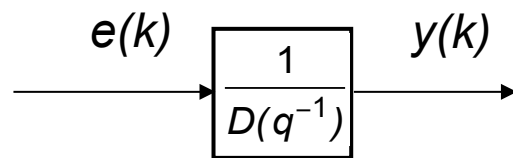
Sa densité de probabilité est définie par ses 2 premiers moments (moyenne et variance)

$$e(k) \sim N(0, \sigma_e^2)$$

Modèles paramétriques usuels de signaux aléatoires stationnaires - Rappel

- Suivant la forme du filtre, on obtient différentes structures de modèles

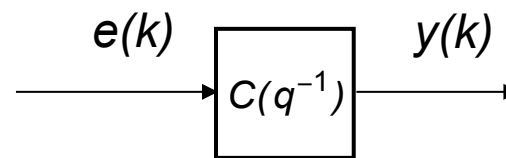
Modèle AR



$$y(k) = \frac{1}{D(q^{-1})} e(k)$$

$$H(q^{-1}) = \frac{1}{D(q^{-1})}$$

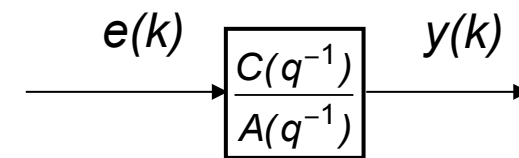
Modèle MA



$$y(k) = C(q^{-1})e(k)$$

$$H(q^{-1}) = C(q^{-1})$$

Modèle ARMA

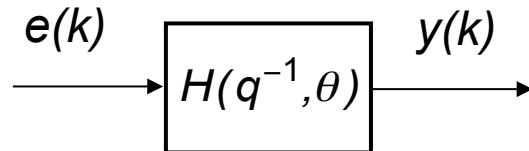


$$D(q^{-1})y(k) = C(q^{-1})e(k)$$

$$H(q^{-1}) = \frac{C(q^{-1})}{D(q^{-1})}$$

- peuvent modéliser des signaux réels comme les signaux de parole, les EEG, ...

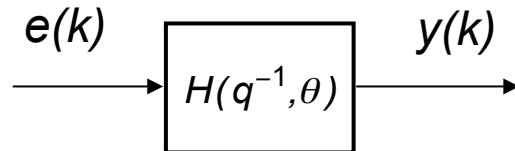
Modèles paramétriques - Vocabulaire



$$y(k) = H(q^{-1}, \theta)e(k)$$

- Le vecteur des paramètres θ rassemble les coefficients de $H(q^{-1}, \theta)$ à estimer
- *L'estimée* de θ sera notée $\hat{\theta}$ et dépendra de l'estimateur utilisé : moindres carrés, ...
- Le but de l'estimation est de fournir la valeur numérique des paramètres

Paramétrisation des opérateurs de transfert



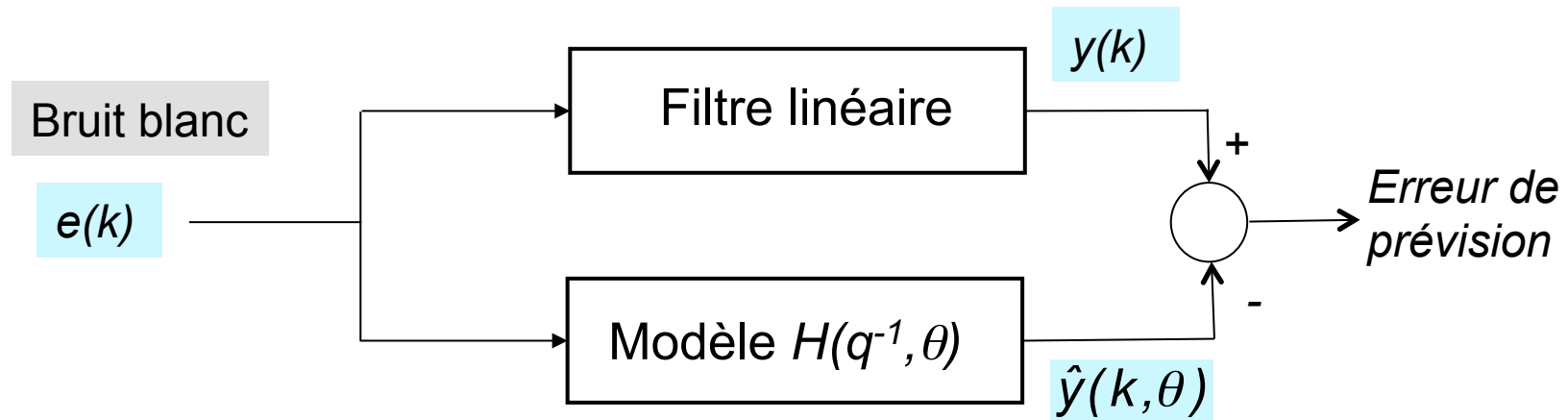
$$y(k) = H(q^{-1}, \theta)e(k)$$

- L'opérateur de transfert $H(q^{-1}, \theta)$ est paramétré de la manière suivante (forme utilisée au sein de la boîte à outils *System Identification de Matlab*)

$$H(q^{-1}, \theta) = \frac{C(q^{-1})}{D(q^{-1})} = \frac{1 + c_1 q^{-1} + \dots + c_{n_c} q^{-n_c}}{1 + d_1 q^{-1} + \dots + d_{n_d} q^{-n_d}}$$

- où le vecteur des paramètres θ rassemble les coefficients c_i et d_i
- n_c et n_d représentent le nombre de paramètres à estimer des polynômes $C(q^{-1}, \theta)$ et $D(q^{-1}, \theta)$

Principe de la méthode de l'erreur de prévision



- Pour un vecteur de paramètres donné θ , le modèle permet de calculer la prévision du signal notée $\hat{y}(k, \theta)$
- Il faut alors déterminer θ tel que $\hat{y}(k, \theta)$ soit « *le plus proche possible* » de $y(k)$

Estimation par minimisation de l'erreur de prévision

- Pour résoudre ce problème d'estimation, il faut remarquer :
 1. La prévision de la sortie $\hat{y}(k, \theta)$ dépend de la structure de modèle choisie : AR, MA ou ARMA

2. Le concept « *doit être la plus proche possible* » doit être formulé mathématiquement. Plusieurs choix existent dont

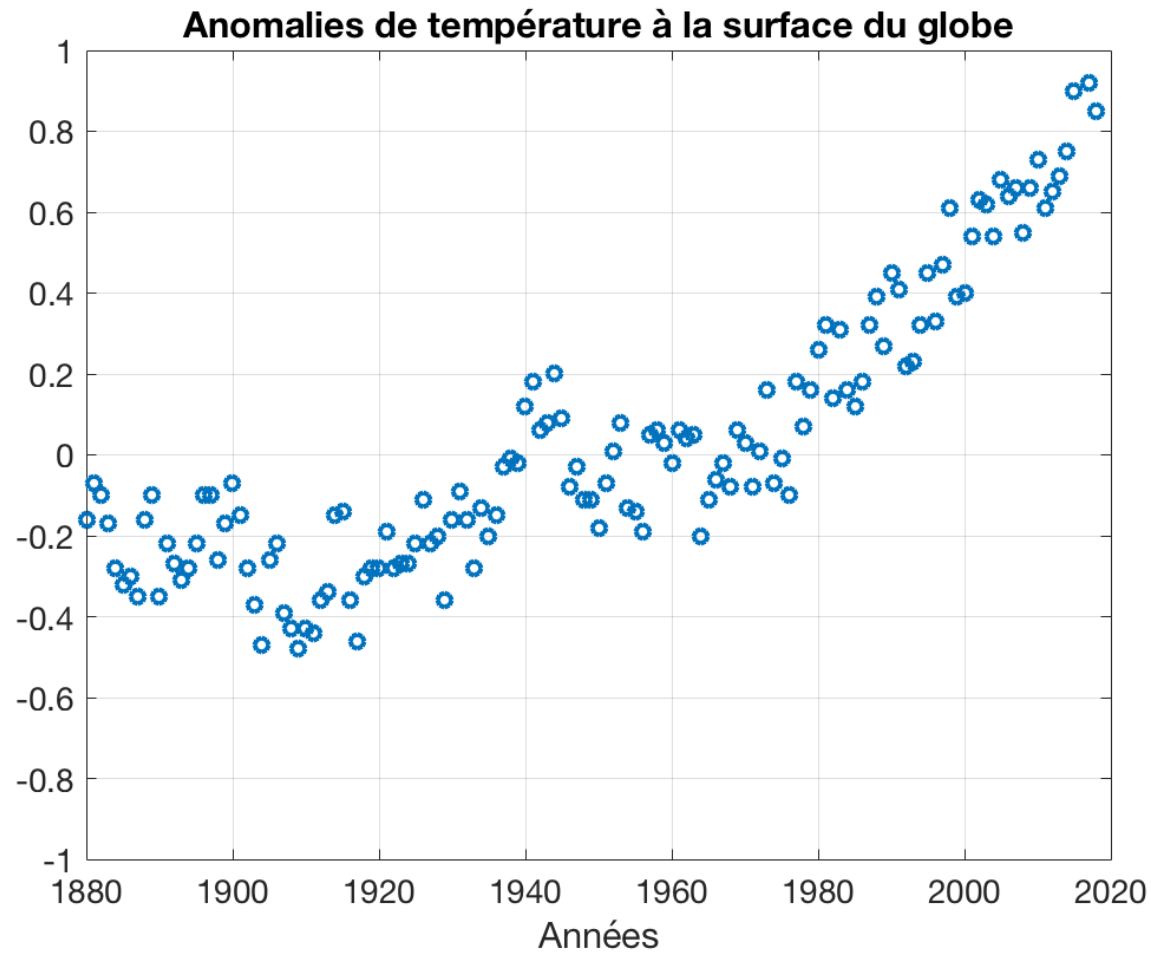
- Critère des moindres valeurs absolues
$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N |y(k) - \hat{y}(k, \theta)|$$

- ♦ Méthode robuste aux éventuelles données aberrantes

- Critère des moindres carrés
$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N (y(k) - \hat{y}(k, \theta))^2$$

- ♦ Méthode peu robuste aux éventuelles données aberrantes
- ♦ Si le modèle s'écrit sous la forme de régression linéaire, la méthode des moindres carrés conduit à une solution analytique ! **Youppie**

Rappels - Méthode des moindres carrés
 Illustration pour déterminer un modèle de
 tendance des anomalies de température



Estimation d'un modèle linéaire par moindres carrés

$$\theta = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N (y(t_k) - (\alpha \times t_k + \beta))^2$$

Ce critère est une fonction quadratique de θ

- Au minimum du critère, sa dérivée première par rapport à θ est nulle :

$$\begin{aligned} \frac{\partial V(\theta, Z^N)}{\partial \alpha} &= \frac{2}{N} \sum_{k=1}^N -t_k (y(t_k) - (\alpha \times t_k + \beta)) = 0 \\ \frac{\partial V(\theta, Z^N)}{\partial \beta} &= \frac{2}{N} \sum_{k=1}^N -(y(t_k) - (\alpha \times t_k + \beta)) = 0 \end{aligned} \quad \begin{bmatrix} \sum_{k=1}^N t_k^2 & \sum_{k=1}^N t_k \\ \sum_{k=1}^N t_k & \sum_{k=1}^N N \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^N t_k y(t_k) \\ \sum_{k=1}^N y(t_k) \end{bmatrix}$$

- **L'estimateur des moindres carrés** est donné par :

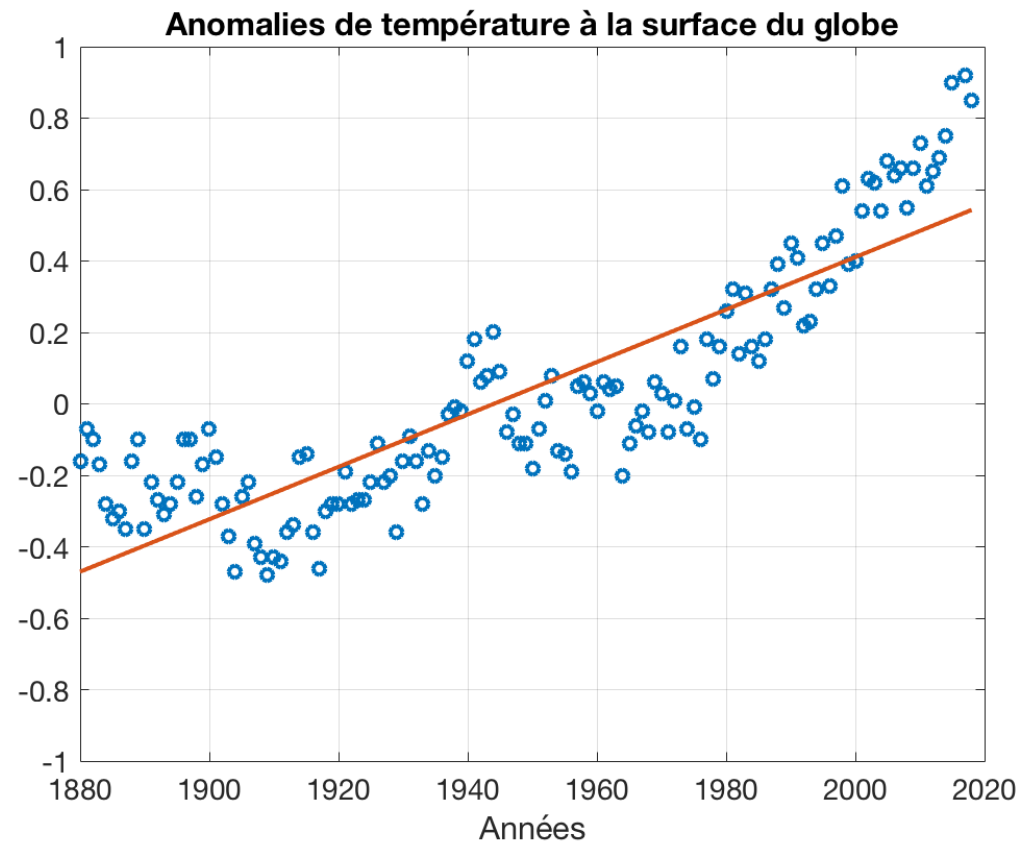
$$\begin{bmatrix} \hat{\alpha} \\ \hat{\beta} \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^N t_k^2 & \sum_{k=1}^N t_k \\ \sum_{k=1}^N t_k & N \end{bmatrix}^{-1} \begin{bmatrix} \sum_{k=1}^N t_k y(t_k) \\ \sum_{k=1}^N y(t_k) \end{bmatrix}$$

Sous Matlab

```
theta_hat = inv([sum(years.^2) sum(years);
sum(years) N])*...
[sum(years.*T);sum(T)]
T_hat = theta(1)* years + theta(2);
plot(years,T,'o',years,T_hat)
```

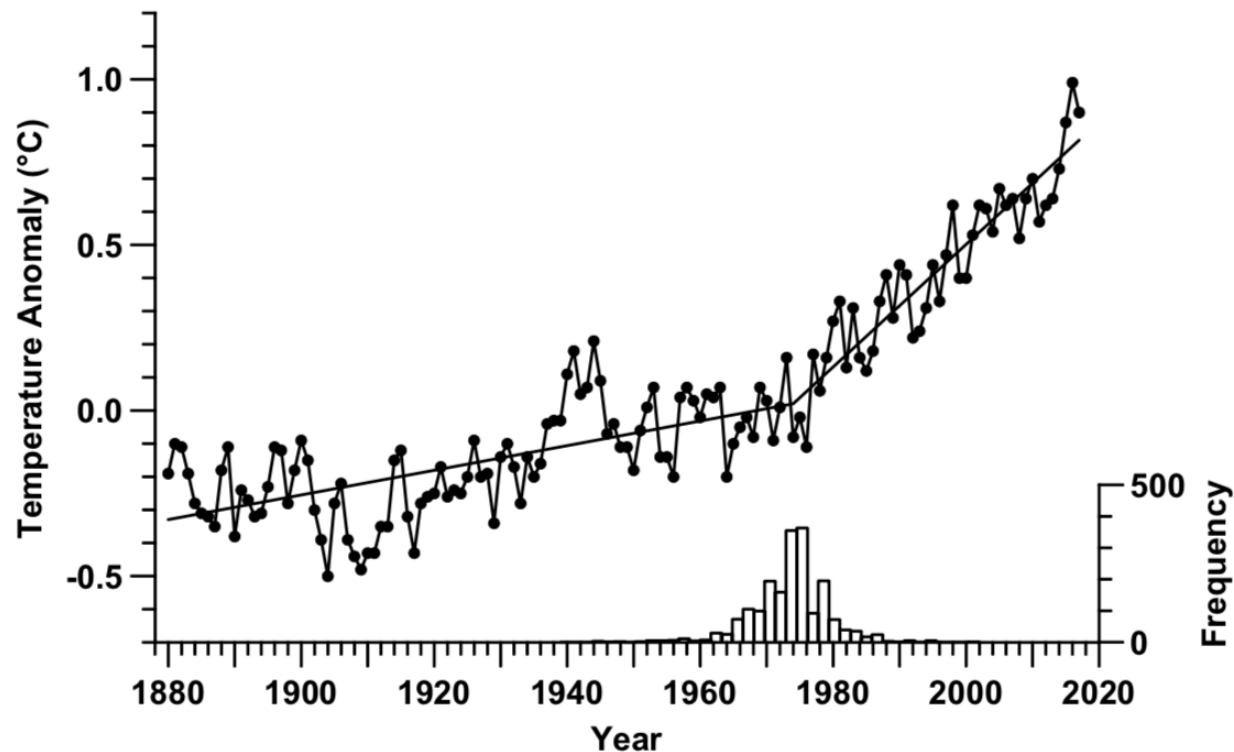
Estimation d'un modèle linéaire par moindres carrés

$$V\left(\begin{bmatrix} \alpha \\ \beta \end{bmatrix}, Z^N\right) = \frac{1}{N} \sum_{k=1}^N \left(y(t_k) - (\alpha \times t_k + \beta)\right)^2$$



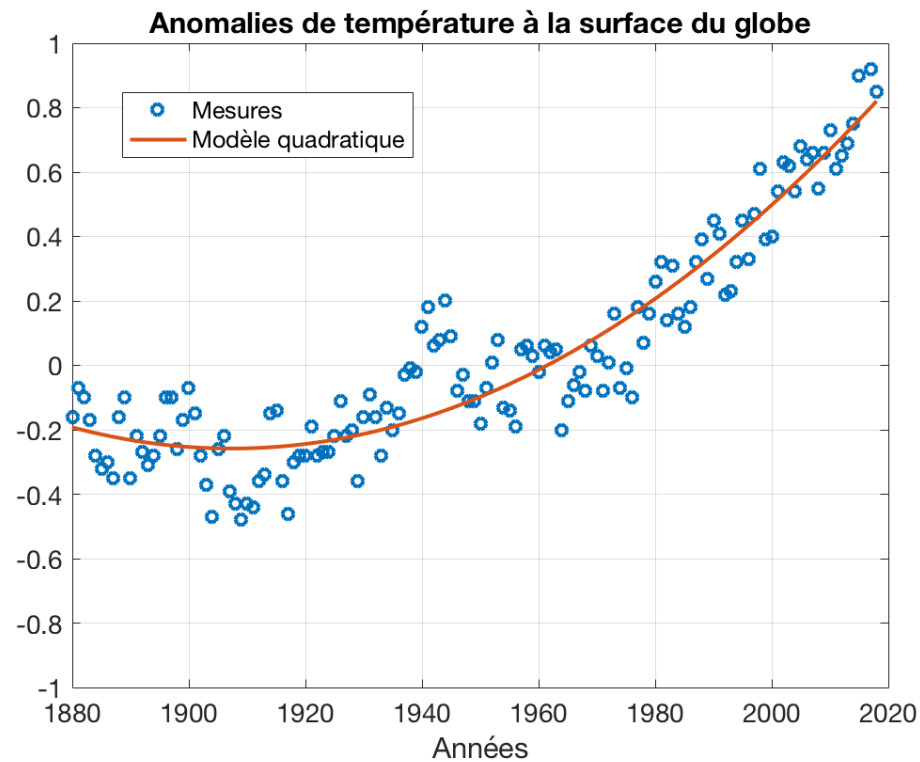
Estimation de 2 modèles linéaires par morceaux par moindres carrés

$$V\left(\begin{bmatrix} \alpha_1 \\ \beta_1 \end{bmatrix}, Z^{N_1}\right) = \frac{1}{N} \sum_{k=1}^{N_1} (y(t_k) - (\alpha_1 \times t_k + \beta_1))^2 \quad \left(\begin{bmatrix} \alpha_2 \\ \beta_2 \end{bmatrix}, Z^{N_2-N_1-1}\right) = \frac{1}{N} \sum_{k=N_1+1}^{N_2} (y(t_k) - (\alpha_2 \times t_k + \beta_2))^2$$



Estimation d'un modèle quadratique par moindres carrés

$$V\left(\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}, Z^N\right) = \frac{1}{N} \sum_{k=1}^N \left(y(t_k) - (\alpha t_k^2 + \beta t_k + \gamma) \right)^2$$



Estimation paramétrique par minimisation de l'erreur de prévision - **Cas général**

Soit un ensemble Z^N d'échantillons du signal aléatoire

$$Z^N = \{y(1), y(2), \dots, y(N)\}$$

1. Calculer la prévision de la sortie du modèle choisi à partir des échantillons jusqu'à l'instant $k-1$ $\hat{y}(k, \theta)$

2. Former l'erreur de prévision $\varepsilon(k) = y(k) - \hat{y}(k, \theta)$

3. Construire la fonction de coût (ou critère)

$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N \varepsilon^2(k) = \frac{1}{N} \sum_{k=1}^N (y(k) - \hat{y}(k, \theta))^2$$

Les paramètres optimaux sont ceux qui minimisent la fonction de coût

$$\hat{\theta} = \arg \min_{\theta} V(\theta, Z^N)$$

Prévision de la sortie – Cas général

- Soit le modèle général

$$y(k) = H(q^{-1})e(k)$$

$e(k)$ est un bruit blanc de moyenne nulle et variance σ^2

Aucune hypothèse sur sa densité de probabilité (gaussienne, uniforme, ...)

- Multiplions par $H^{-1}(q^{-1})$

$$H^{-1}(q^{-1})y(k) = e(k)$$

$$y(k) - y(k) + H^{-1}(q^{-1})y(k) = e(k)$$

$$y(k) = \left(1 - H^{-1}(q^{-1})\right) y(k) + e(k)$$

- Puisque $e(k)$ est un bruit blanc ($E(e(k))=0$), la meilleure prévision est obtenue en supposant $e(k)=0$ dans l'expression ci-dessous qui définit la prévision de la sortie dans le cas général (*quel que soit $H(q^{-1})$*)

$$\hat{y}(k, \theta) = \left(1 - H^{-1}(q^{-1})\right) y(k)$$

Prévision de la sortie – *Cas d'un modèle AR*

- Soit le modèle AR

$$y(k) = \frac{1}{D(q^{-1})} e(k)$$

- La prévision du signal s'écrit :

$$\hat{y}(k, \theta) = \left(1 - H^{-1}(q^{-1})\right) y(k) = \left(1 - D(q^{-1})\right) y(k) \quad H(q^{-1}) = \frac{1}{D(q^{-1})}; \quad H^{-1}(q^{-1}) = D(q^{-1})$$

$$\hat{y}(t_k, \theta) = -d_1 y(k-1) - \dots - d_{n_d} y(k-n_d)$$

$$\hat{y}(k, \theta) = \begin{bmatrix} -y(k-1) & \dots & -y(k-n_d) \end{bmatrix} \begin{bmatrix} d_1 \\ \vdots \\ d_{n_d} \end{bmatrix}$$

$$\hat{y}(k, \theta) = \phi^T(k) \theta$$

$$\text{avec } \phi^T(k) = \begin{bmatrix} -y(k-1) & \dots & -y(k-n_d) \end{bmatrix} \quad \theta = \begin{bmatrix} d_1 & \dots & d_{n_d} \end{bmatrix}^T$$

- La prévision du signal est donc une **fonction linéaire en les paramètres**
- Le problème est celui d'une **régression linéaire** et les paramètres pourront être calculés par **moindres carrés simples**

Estimation paramétrique par minimisation de l'erreur de prévision - Cas d'un modèle AR

Soit un ensemble Z^N d'échantillons du signal aléatoire

$$Z^N = \{y(1), y(2), \dots, y(N)\}$$

1. Calculer la prévision du signal à partir des données jusqu'à l'instant $k-1$ (*modèle de régression linéaire*)

$$\hat{y}(k, \theta) = \phi^T(k) \theta \quad \phi^T(k) = \begin{bmatrix} -y(k-1) & \dots & -y(k-n_d) \end{bmatrix}$$

$$\theta = \begin{bmatrix} d_1 & \dots & d_{n_d} \end{bmatrix}^T$$

2. Former l'erreur de prévision

$$\varepsilon(k) = y(k) - \hat{y}(k, \theta)$$

3. Construire la fonction de coût

$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N \varepsilon^2(k) = \frac{1}{N} \sum_{k=1}^N (y(k) - \hat{y}(k, \theta))^2$$

Estimation d'un modèle AR par moindres carrés

$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N \varepsilon^2(k) = \frac{1}{N} \sum_{k=1}^N (y(k) - \hat{y}(k, \theta))^2$$

Ce critère est une fonction quadratique de θ

- Au minimum du critère, sa dérivée première par rapport à θ est nulle :

$$\frac{\partial V(\theta, Z^N)}{\partial \theta} = \frac{2}{N} \sum_{k=1}^N \phi(k) (y(k) - \phi^T(k)\theta) = 0$$

$$\frac{2}{N} \sum_{k=1}^N (\phi(k)y(k) - \phi(k)\phi^T(k)\theta) = 0$$

- **L'estimateur des moindres carrés simples** est donné par :

$$\hat{\theta}_{mc} = \left[\sum_{k=1}^N \phi(k)\phi^T(k) \right]^{-1} \sum_{k=1}^N \phi(k)y(k)$$

à condition que l'inverse de la matrice existe

Estimation d'un modèle AR par moindres carrés

Formulation matricielle

- On peut également exprimer l'erreur de prédiction sous forme matricielle

$$\begin{bmatrix} \varepsilon(1) \\ \vdots \\ \varepsilon(N) \end{bmatrix} = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix} - \begin{bmatrix} \phi^T(1) \\ \vdots \\ \phi^T(N) \end{bmatrix} \theta$$

$$\varepsilon_N = Y_N - \Phi_N \theta$$

- Le critère s'écrit : $V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N \varepsilon^2(k) = \frac{1}{N} \varepsilon_N^T \varepsilon_N = \frac{1}{N} (Y_N - \Phi_N \theta)^T (Y_N - \Phi_N \theta)$

- La condition de gradient nul conduit $\frac{\partial V(\theta, Z^N)}{\partial \theta} = -Y_N^T \Phi_N + \theta^T (\Phi_N^T \Phi_N) = 0$

aux équations normales : $(\Phi_N^T \Phi_N) \theta = \Phi_N^T Y_N$

- L'estimateur des moindres carrés peut aussi s'écrire sous la forme matricielle :

$$\hat{\theta}_{mc} = (\Phi_N^T \Phi_N)^{-1} \Phi_N^T Y_N$$

Estimateur des moindres carrés

Aspects numériques

$$\hat{\theta}_{mc} = \left(\Phi_N^T \Phi_N \right)^{-1} \Phi_N^T Y_N = \left[\sum_{k=1}^N \phi(k) \phi^T(k) \right]^{-1} \sum_{k=1}^N \phi(k) y(k)$$

- Sauf pour les cas d'école, il vaut mieux éviter d'utiliser les formes ci-dessus qui nécessitent une inversion de matrice, ce qui peut s'avérer désastreux sur le plan numérique
- Alternatives : éviter l'inversion de matrice
 - Décomposition en valeurs singulières
 - Décomposition QR
 - Beaucoup mieux adapté pour les problèmes mal conditionnés, c'est-à-dire ceux où la matrice à inverser est numériquement mal conditionnée
- Remarque : sous Matlab

$$\hat{\theta}_{mc} = \Phi_N \backslash Y_N$$

Exemple : estimation d'un modèle AR d'ordre 2

- Soit le modèle AR d'ordre 2

$$y(k) = -d_1 y(k-1) - d_2 y(k-2) + e(k)$$

$$y(k) = \begin{bmatrix} -y(k-1) & -y(k-2) \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} + e(k)$$

$$y(k) = \phi^T(k) \theta + e(k) \quad \text{avec } \phi^T(k) = \begin{bmatrix} -y(k-1) & -y(k-2) \end{bmatrix} \quad \theta = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$

$$\begin{bmatrix} y(3) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} -y(2) & -y(1) \\ \vdots & \vdots \\ -y(N-1) & -y(N-2) \end{bmatrix} \theta + \begin{bmatrix} e(3) \\ \vdots \\ e(N) \end{bmatrix}$$

$$Y_N = \Phi_N \theta + E_N$$

$$\hat{\theta}_{mc} = \left(\Phi_N^T \Phi_N \right)^{-1} \Phi_N^T Y_N$$

- Sous Matlab

```
Phi_N = [-y(2:N-1) -y(1:N-2)];  
Y_N = y(3:N);  
theta = Phi_N \ Y_N
```

Exemple : estimation d'un modèle AR d'ordre 2

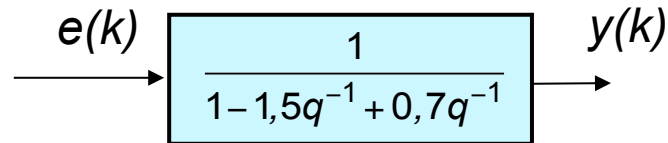
$$\hat{\theta}_{mc} = \left(\Phi_N^T \Phi_N \right)^{-1} \Phi_N^T Y_N = \left[\sum_{k=1}^N \phi(k) \phi^T(k) \right]^{-1} \sum_{k=1}^N \phi(k) y(k)$$

$$\Phi_N^T \Phi_N = \begin{bmatrix} -y(2) & \dots & -y(N-1) \\ -y(1) & \dots & -y(N-2) \end{bmatrix} \begin{bmatrix} -y(2) & -y(1) \\ \vdots & \vdots \\ -y(N-1) & -y(N-2) \end{bmatrix}$$

$$\Phi_N^T Y_N = \begin{bmatrix} -y(2) & \dots & -y(N-1) \\ -y(1) & \dots & -y(N-2) \end{bmatrix} \begin{bmatrix} y(3) \\ \vdots \\ y(N) \end{bmatrix}$$

$$\hat{\theta}_{mc} = \begin{pmatrix} \hat{d}_1 \\ \hat{d}_2 \end{pmatrix} = \begin{pmatrix} \sum_{k=1}^N y^2(k) & \sum_{k=1}^N y(k)y(k-1) \\ \sum_{k=1}^N y(k)y(k-1) & \sum_{k=1}^N y^2(k) \end{pmatrix}^{-1} \begin{pmatrix} -\sum_{k=1}^N y(k)y(k-1) \\ -\sum_{k=1}^N y(k)y(k-2) \end{pmatrix}$$

Estimateur des MC – Implantation sous Matlab



$$y(k) = \frac{1}{1 - 1,5q^{-1} + 0,7q^{-1}} e(k)$$

`% Simulation du modèle AR(2)`

`C=1;`

`D=[1 -1.5 0.7];`

`N=500;`

`e= randn(N,1);`

`% bruit blanc gaussien`

`y=filter(C,D,e);`

`% simulation du modèle AR(2)`

`% Estimation des paramètres par MC`

`Phi = [-y(2:N-1) -y(1:N-2)];` `% Matrice de régression`

`Y = y(3:N);` `% Régresseur`

`theta_mc=(Phi\Y)'` `% Estimation par MC`

`ans -1.5024 0.7078` `% On estime bien les 2 paramètres`

`% voir aussi la fonction ar de la SID`

`data=iddata(y,[]); idplot(data);`

`Mar=ar(data,2,'ls')`

Analyse des performances de l'estimateur des MC par simulation de Monte Carlo

```
C=1;
D=[1 -1.5 0.7];
N=500;           % Nombre d'échantillons
Matrice_param_ar=[];
for i=1:200      % 200 réalisations de bruit
    randn('state',sum(100*clock));
    e=randn(N,1);
    y=filter(C,D,e); % simulation du signal aléatoire
    data= iddata(y,[]); % Objet de données ayant y comme sortie
    Mar=ar(data,2,'ls'); % Estimation par MC des paramètres du modèle AR
    Matrice_param_ar=[Matrice_param_ar; Mar.A(2:end)];
end
% Valeur moyenne des paramètres estimés par MC
mean(Matrice_param_ar)
-1.4995    0.6984
% Ecart-type des paramètres estimés par MC
std(Matrice_param_ar)
0.0325    0.0326
```

Méthode de l'erreur de prévision *Cas des modèles AR*

La prévision du signal s'exprime comme une fonction *linéaire en les paramètres* et s'écrit sous forme de régression linéaire

$$\hat{y}(k, \theta) = \phi^T(k) \theta$$

- Une solution analytique existe ! ***Youppie !***
- Le vecteur de paramètres estimés par moindres carrés est obtenu **analytiquement** à partir des mesures, *c.a.d.* sans problème d'initialisation. La solution est donnée par la formule

$$\hat{\theta}_{mc} = \left(\Phi_N^T \Phi_N \right)^{-1} \Phi_N^T Y_N$$

Méthode de l'erreur de prévision *Cas des modèles MA et ARMA*

La prévision du signal *ne s'exprime pas comme une fonction linéaire en les paramètre* et ne s'écrit pas sous forme de régression linéaire

$$\hat{\theta} = \arg \min_{\theta} V(\theta, Z^N)$$

$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N \varepsilon^2(k, \theta)$$

- Une ***solution analytique n'existe pas*** ! La minimisation du critère est effectuée à l'aide d'algorithmes *d'optimisation numérique itératifs* comme celui de *Gauss-Newton*

$$\hat{\theta}^{(i+1)} = \hat{\theta}^{(i)} - \mu R_N^{-1}(\hat{\theta}^{(i)}) f_N(\hat{\theta}^{(i)})$$

- μ : pas
- $f_N(\cdot)$: approximation du gradient $\frac{dV(\theta, Z^N)}{d\theta}$
- $R_N(\cdot)$: approximation du Hessien $\frac{d^2V(\theta, Z^N)}{d\theta^2}$

Estimation de *modèles MA et ARMA* *Algorithme d'optimisation numérique*

$\hat{\theta}$ est un minimum (au moins localement) si son *gradient* est nul et son *Hessien* est défini semi-positif

$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=1}^N \varepsilon^2(k, \theta)$$

$$\frac{dV(\theta, Z^N)}{d\theta} = - \underbrace{\frac{2}{N} \sum_{k=1}^N \varepsilon(k, \theta) \psi(k, \theta)}_{f_N(\theta)}$$

$$\frac{d^2V(\theta, Z^N)}{d\theta^2} = \underbrace{\frac{2}{N} \sum_{k=1}^N \psi(k, \theta) \psi^T(k, \theta)}_{R_N(\theta)} - \frac{2}{N} \sum_{k=1}^N \varepsilon(k, \theta) \frac{d\psi(k, \theta)}{d\theta}$$

$$\psi(k, \theta) = - \frac{d\varepsilon(k, \theta)}{d\theta}$$

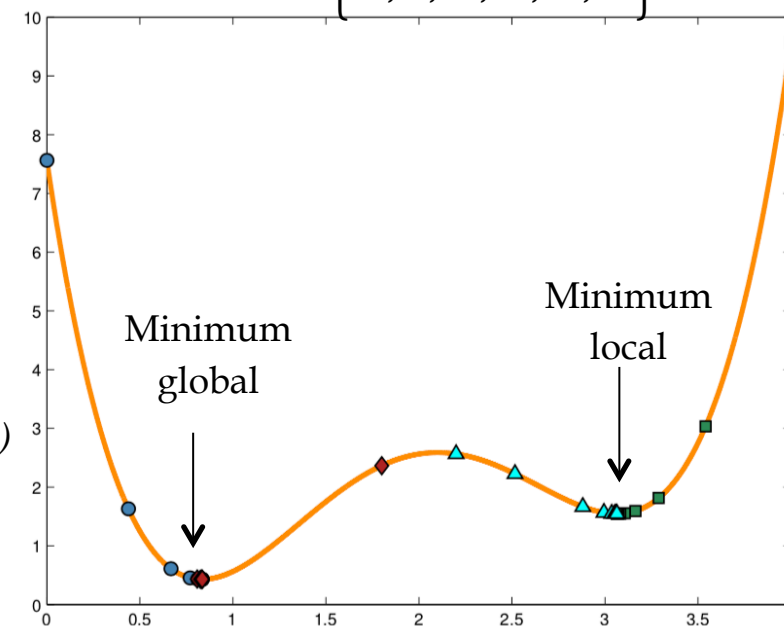
Risque de convergence vers un minimum local Exemple

$$f(\theta) = \theta^4 - 8\theta^3 + \frac{43}{2}\theta^2 - \frac{43}{2}\theta + \frac{121}{16}$$

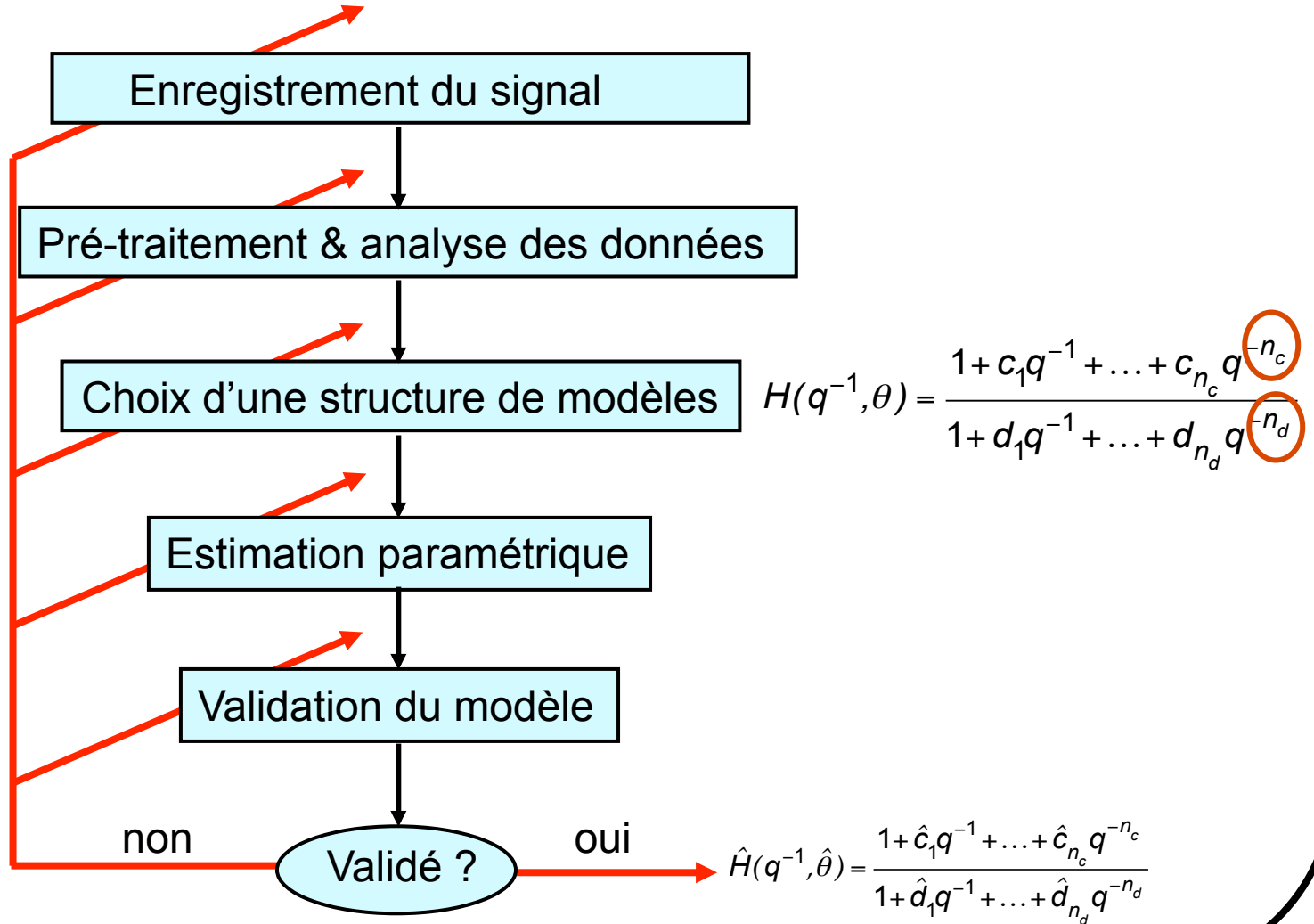
- La fonction possède un minimum local et un minimum global
- Plusieurs valeurs initiales testées pour démarrer l'algorithme de Gauss Newton

$$\hat{\theta}^{(0)} = \{0 ; 1,8 ; 2,2 ; 4\}$$

- La convergence peut être très sensible à l'initialisation de l'algorithme
- Si le critère possède plusieurs minima, l'algorithme peut converger vers un minimum local et non vers le minimum global en fonction du choix de $\hat{\theta}^{(0)}$
- Condition d'arrêt $\|\hat{\theta}^{(i+1)} - \hat{\theta}^{(i)}\| < \delta_\theta$



Méthodologie itérative d'estimation d'un modèle paramétrique d'un signal aléatoire stationnaire



Aspects pratiques

Pré-traitement et validation de modèles

- Suppression de la valeur moyenne et de la tendance linéaire du signal

detrend

- Décimation des données si le signal est sur-échantillonné
 - Règle : $f_e \approx 10 f_B$

decimate

- Pré-filtrage des données
 - supprimer les perturbations ou les composantes fréquentielles qui ne doivent pas être modélisées

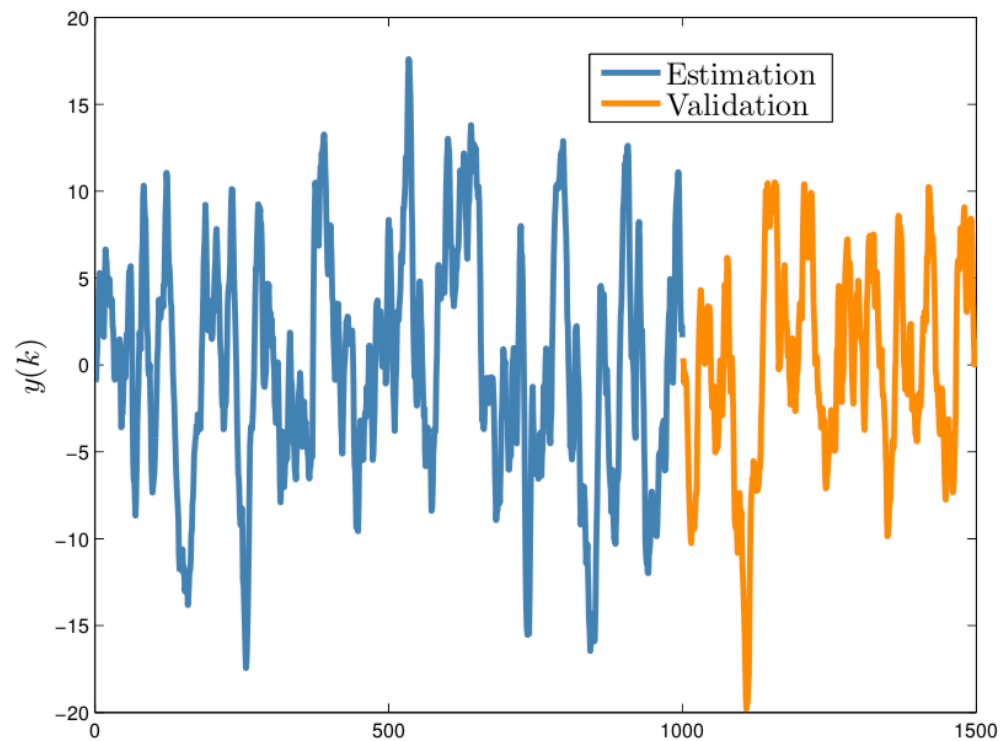
butter, filtfilt

- Constitution d'un jeu de données d'estimation et d'un jeu de données de validation
 - Couper les données en 2 parties :
 - 2/3 de N pour l'estimation ; 1/3 de N pour la validation

Aspects pratiques

Jeu de données pour l'estimation et la validation

- $N = 1500$. Déterminons le « meilleur » ordre pour le modèle AR
 - Coupons les données en 2 parties :
 - $2/3$ de N (=1000) échantillons pour l'estimation
 - $1/3$ de N (=500) pour la validation



Sélection de l'ordre d'un modèle AR

- Pb délicat de la méthodologie d'estimation

$$H(q^{-1}, \theta) = \frac{1}{1 + d_1 q^{-1} + \dots + d_{n_d} q^{-n_d}}$$

- *Méthodologie*

- Estimer plusieurs modèles $AR(n_d)$ pour une plage d'ordres possibles
- Sélection à l'aide de critères d'aide au choix de l'ordre

- Critère d'information d'Akaike (AIC) $AIC(n_d) = V(\hat{\theta}_{mc}) \left(1 + \frac{2n_d}{N} \right)$

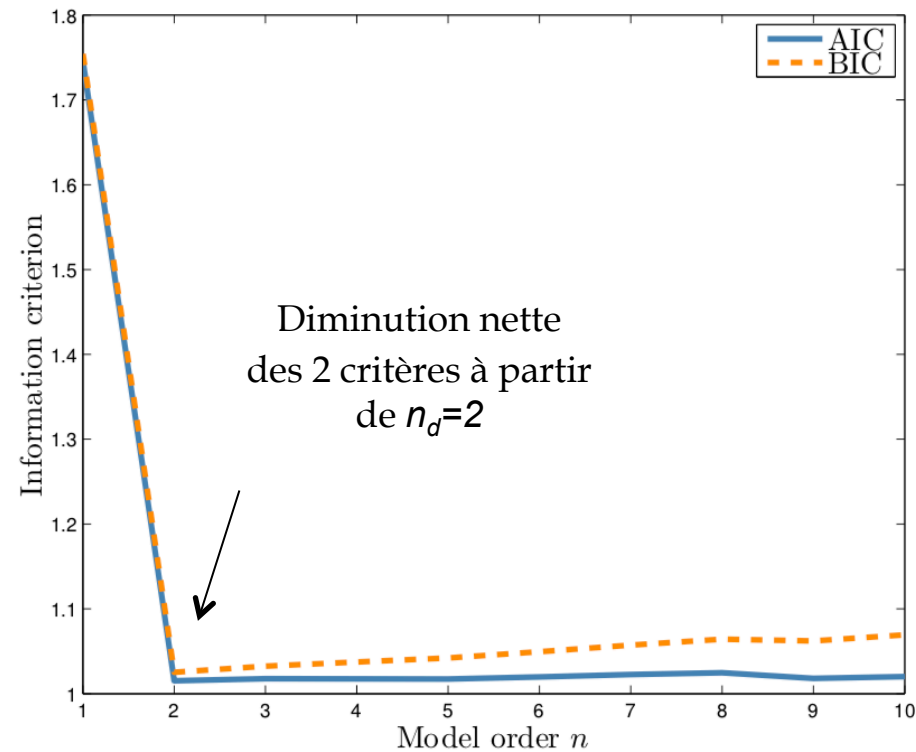
- Critère d'information bayésien (BIC) $BIC(n_d) = V(\hat{\theta}_{mc}) \left(1 + \frac{n_d \log(N)}{N} \right)$

- Choisir l'ordre pour lequel ces 2 critères présentent une valeur suffisamment faible
- Attention, ces deux critères peuvent conduire à une sur-estimation de l'ordre du modèle

Aspects pratiques

Recherche de l'ordre d'un modèle AR - Exemple

- $N = 1500$.
- Estimation de modèles AR d'ordre 1 à 10 et tracé de l'évolution des critères AIC et BIC pour différents ordres du modèle



Aspects pratiques

Recherche de l'ordre d'un modèle AR - Exemple

- Valeurs des 2 critères très proches pour $n_d = 2$, $n_d = 3$ et $n_d = 4$
- Comment choisir entre ces 3 modèles ?
 - Idée : comparer les paramètres et leur écart-type estimés
 - Matlab : `present(ar(data,nd,'ls'))`

$\hat{a}_1^{[2]} = -1.50 \pm 0.026$	$\hat{a}_1^{[3]} = -1.49 \pm 0.032$	$\hat{a}_1^{[4]} = -1.49 \pm 0.032$
$\hat{a}_2^{[2]} = 0.58 \pm 0.026$	$\hat{a}_2^{[3]} = 0.56 \pm 0.054$	$\hat{a}_2^{[4]} = 0.56 \pm 0.057$
	$\hat{a}_3^{[3]} = 0.016 \pm 0.032$	$\hat{a}_3^{[4]} = 0.014 \pm 0.057$
		$\hat{a}_4^{[4]} = 0.001 \pm 0.032$

- $\hat{a}_3^{[3]}$, $\hat{a}_3^{[4]}$ et $\hat{a}_4^{[4]}$ sont quasiment nuls
- Cela indique que $n_d = 3$ et $n_d = 4$ sont des modèles d'ordre trop grand

Règle du rasoir d'Occam

Règle générale proposée par le philosophe Occam, qui permet d'éliminer, de « raser » des explications improbables d'un phénomène

Pourquoi faire compliquer, quand on peut faire simple ?

**“Among models with similar
performance,
select the model that describes
data with fewest parameters”**

Ce n'est pas fini !

- Nombreuses extensions des méthodes de base pour :
 - *Machine learning et IA*
 - EC au S8 et S9 spécialité I2S
 - Voir, par exemple, la chaine youtube science4all
 - ♦ <https://www.youtube.com/watch?v=DrijkPVf7Bw&t=3s>
 - *Maintenance prédictive*
 - EC au S8 et S9 spécialité M3
 - *Apprentissage de modèles dynamiques pour le contrôle*
 - EC au S8 et S9 parcours CSS spécialité I2S
 - *Modélisation de séries temporelles*
 - Voir, par exemple, article sur les Prix Nobel d'économie pour la prévision de séries temporelles en macroéconométrie
 - ♦ <http://images.math.cnrs.fr/Nobel-2011-d-economie.html>