Intelligence Artificielle & Machine Learning pour la modélisation de séries temporelles et de signaux

Séance 5 : Pré-traitements des séries temporelles

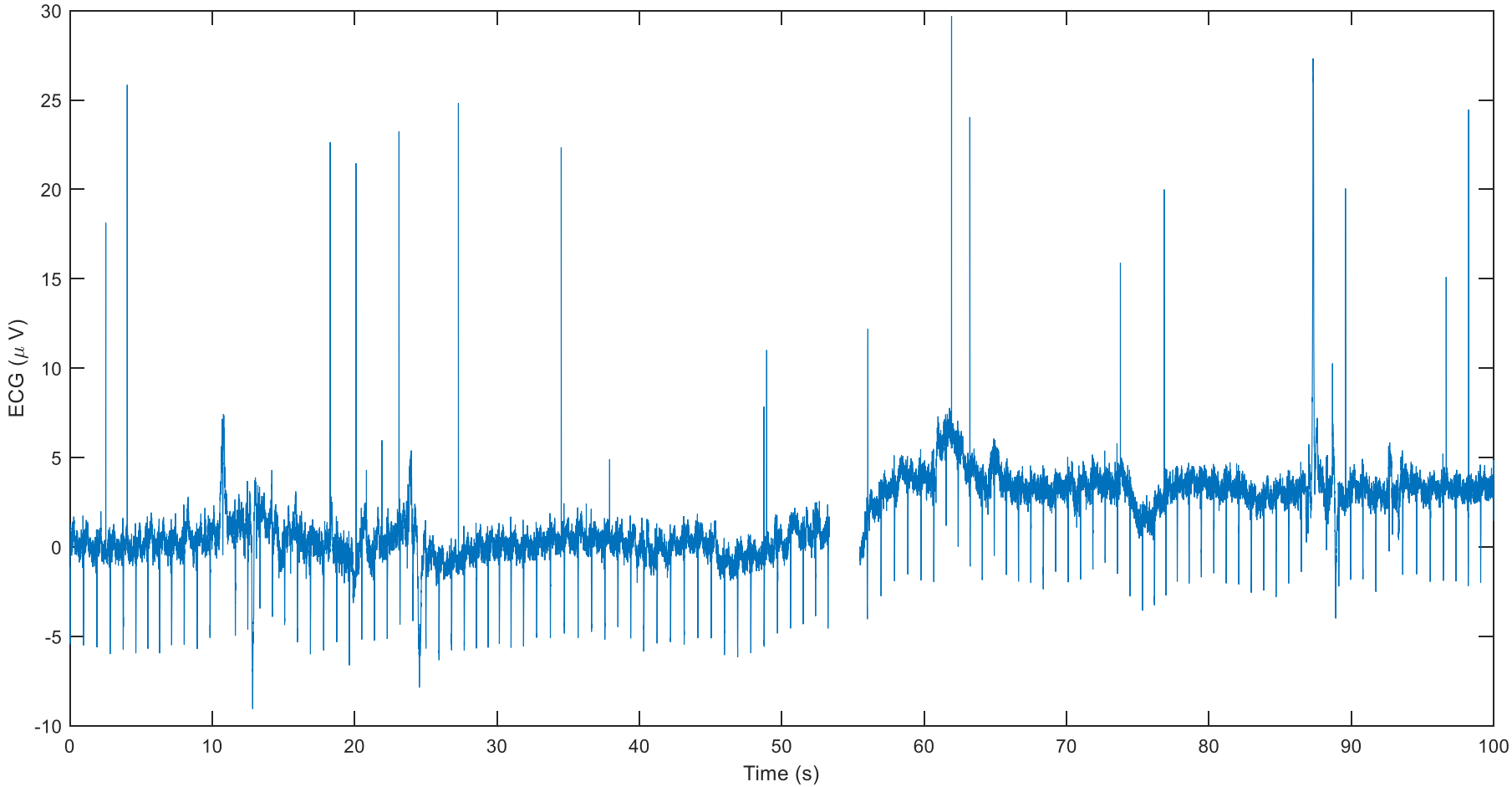
Laurent Oudre
laurent.oudre@ens-paris-saclay.fr

Diplôme ARIA
ENS Paris Saclay
2025-2026

Pourquoi pré-traiter les signaux ?

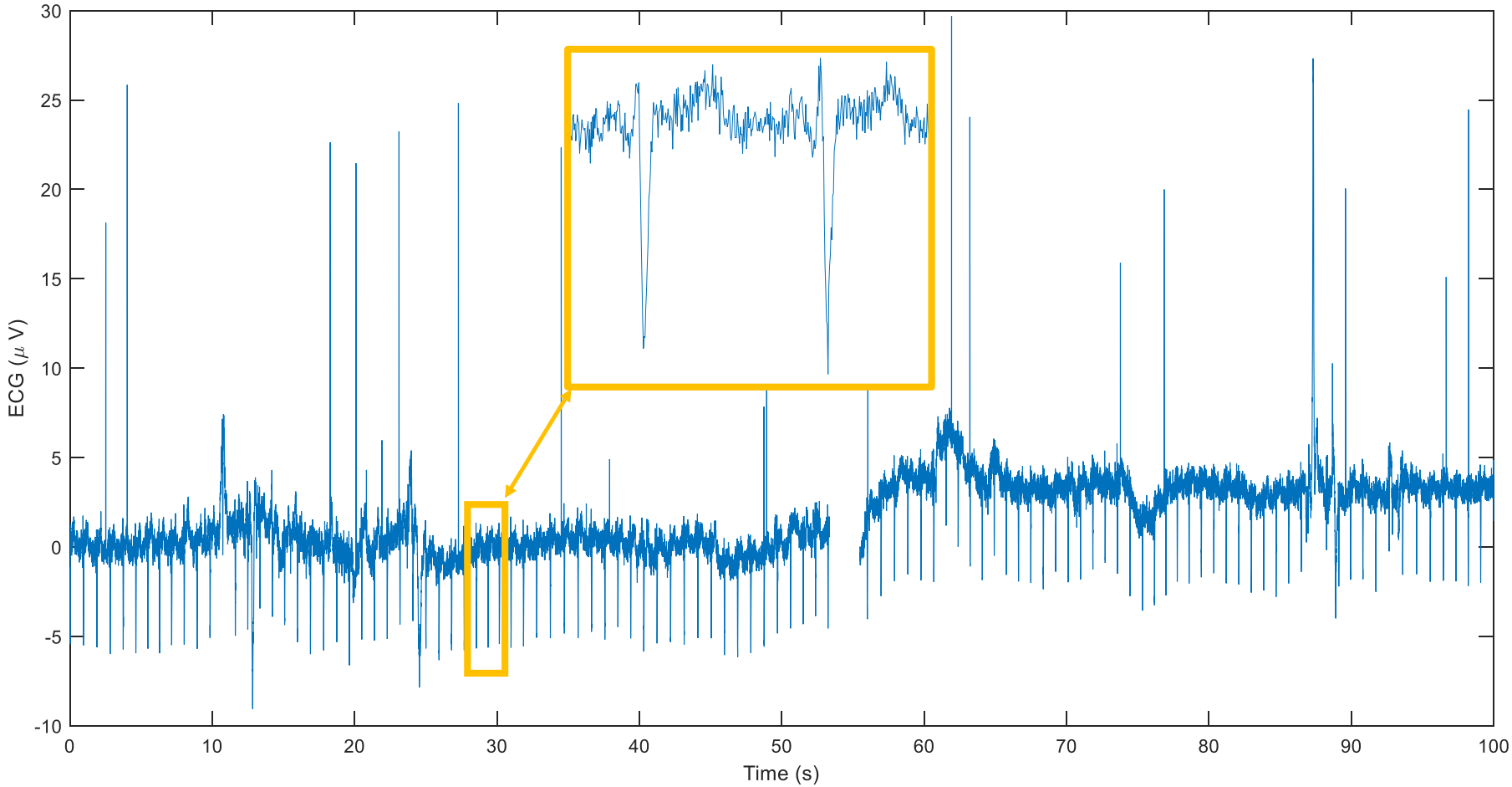
- Les signaux et séries temporelles sont souvent liés à la mesure d'un phénomène
- Cette mesure n'est jamais parfaite et introduit dans les données différents types d'artefacts qui rendent difficile ou empêchent totalement l'extraction d'information et l'apprentissage
- La première étape consiste toujours à nettoyer, normaliser et réparer les données, avant tout autre traitement
- Ces pré-traitements sont donc la base du data mining, et il est important de garder à l'esprit qu'ils ne sont pas anodins et peuvent également dégrader les données lorsqu'ils ne sont pas utilisés à bon escient

Exemple d'un signal ECG



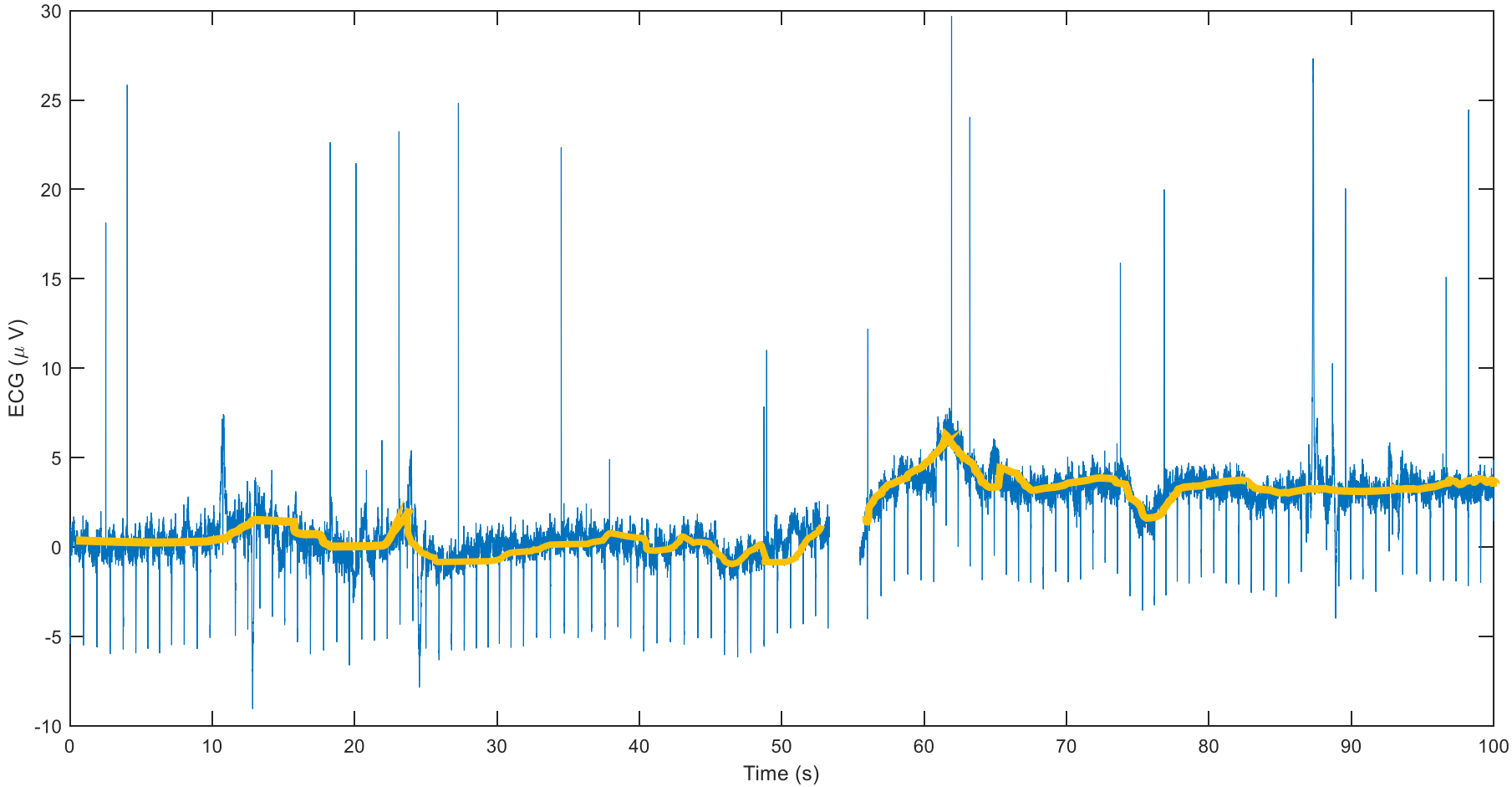
Signal ECG enregistré durant une anesthésie générale

Exemple d'un signal ECG



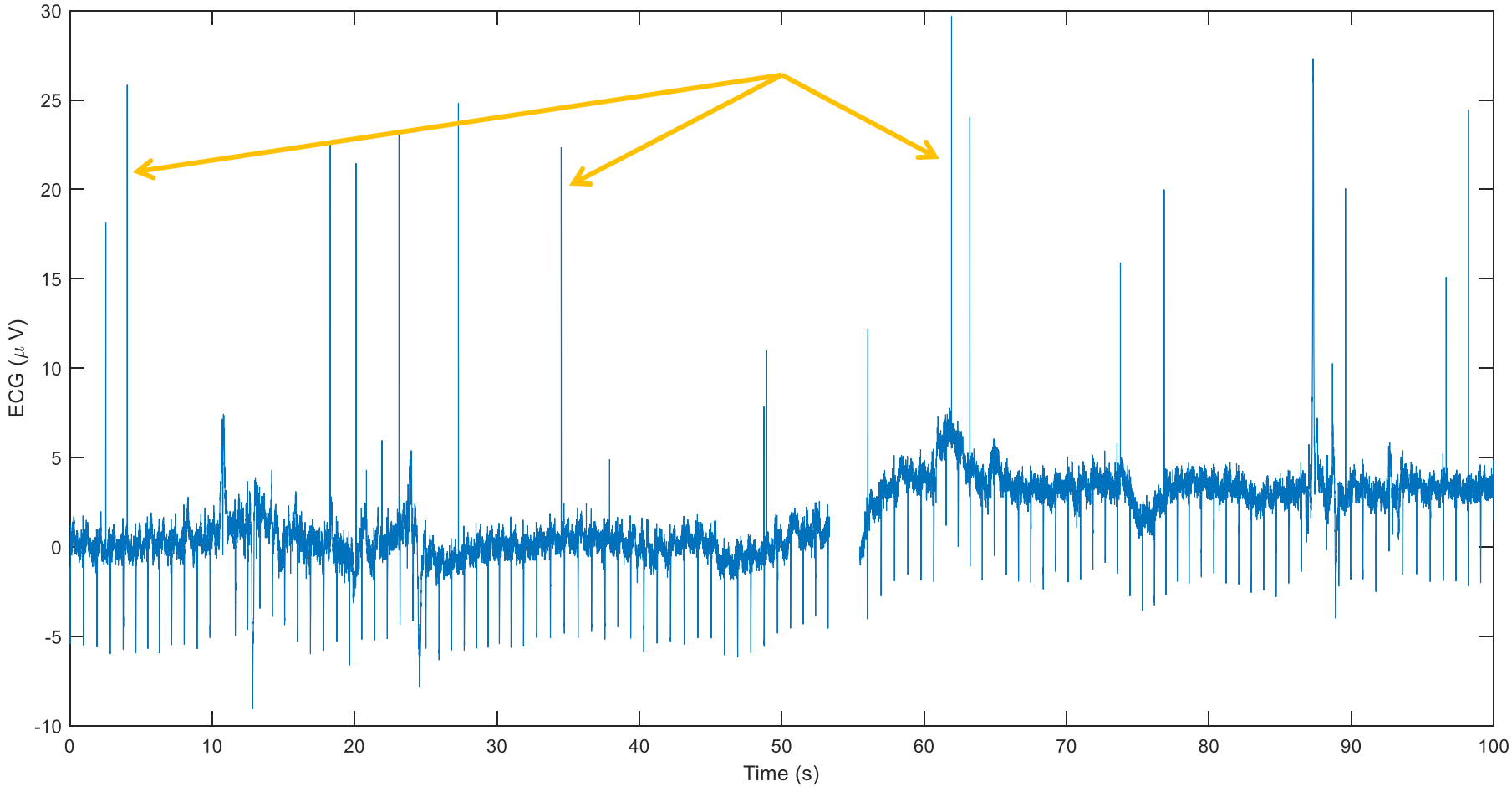
Présence d'un bruit de mesure $\rightarrow$ **Débruitage**

Exemple d'un signal ECG



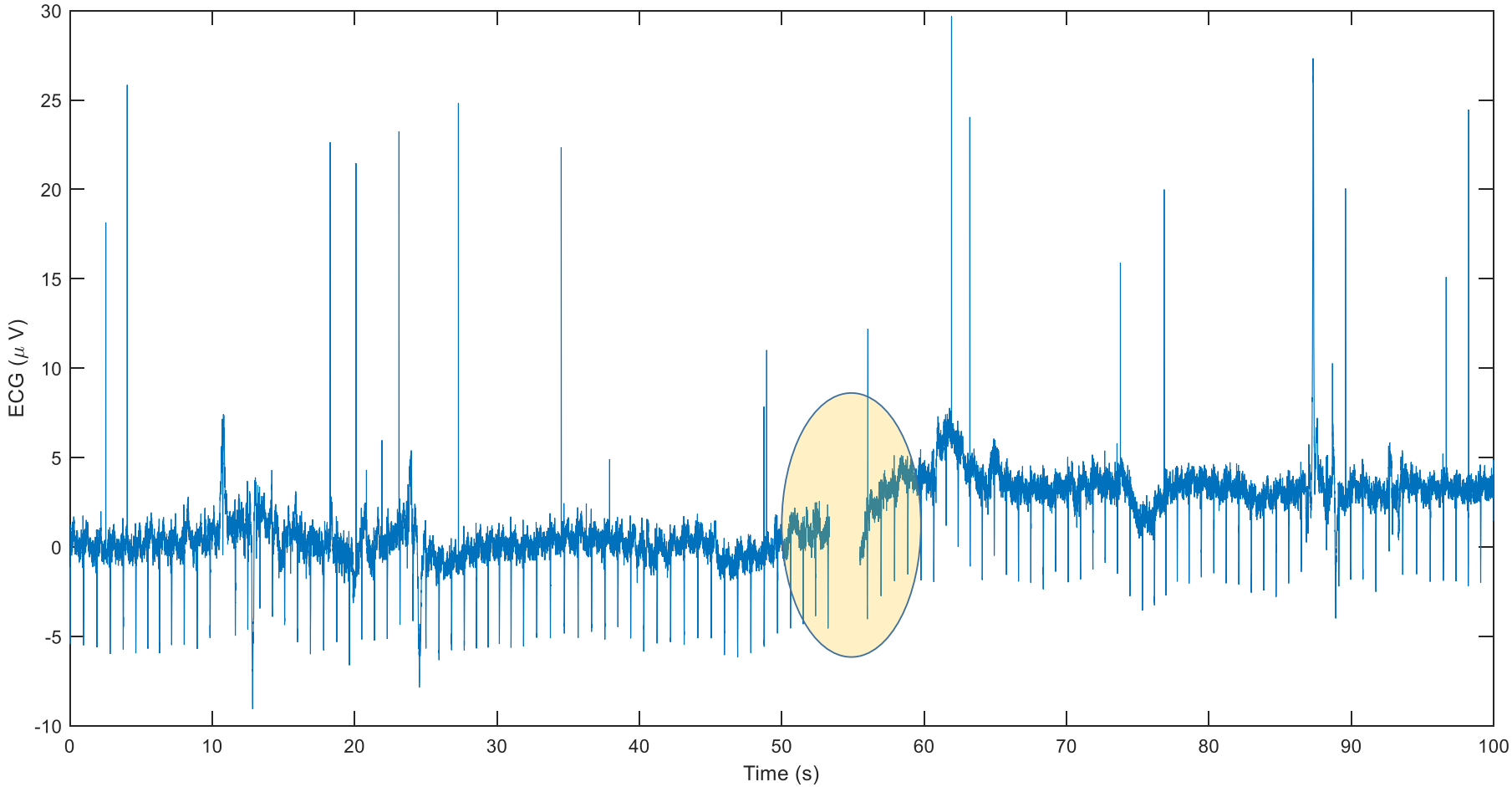
Présence d'une tendance $\rightarrow$ **Detrending**

Exemple d'un signal ECG



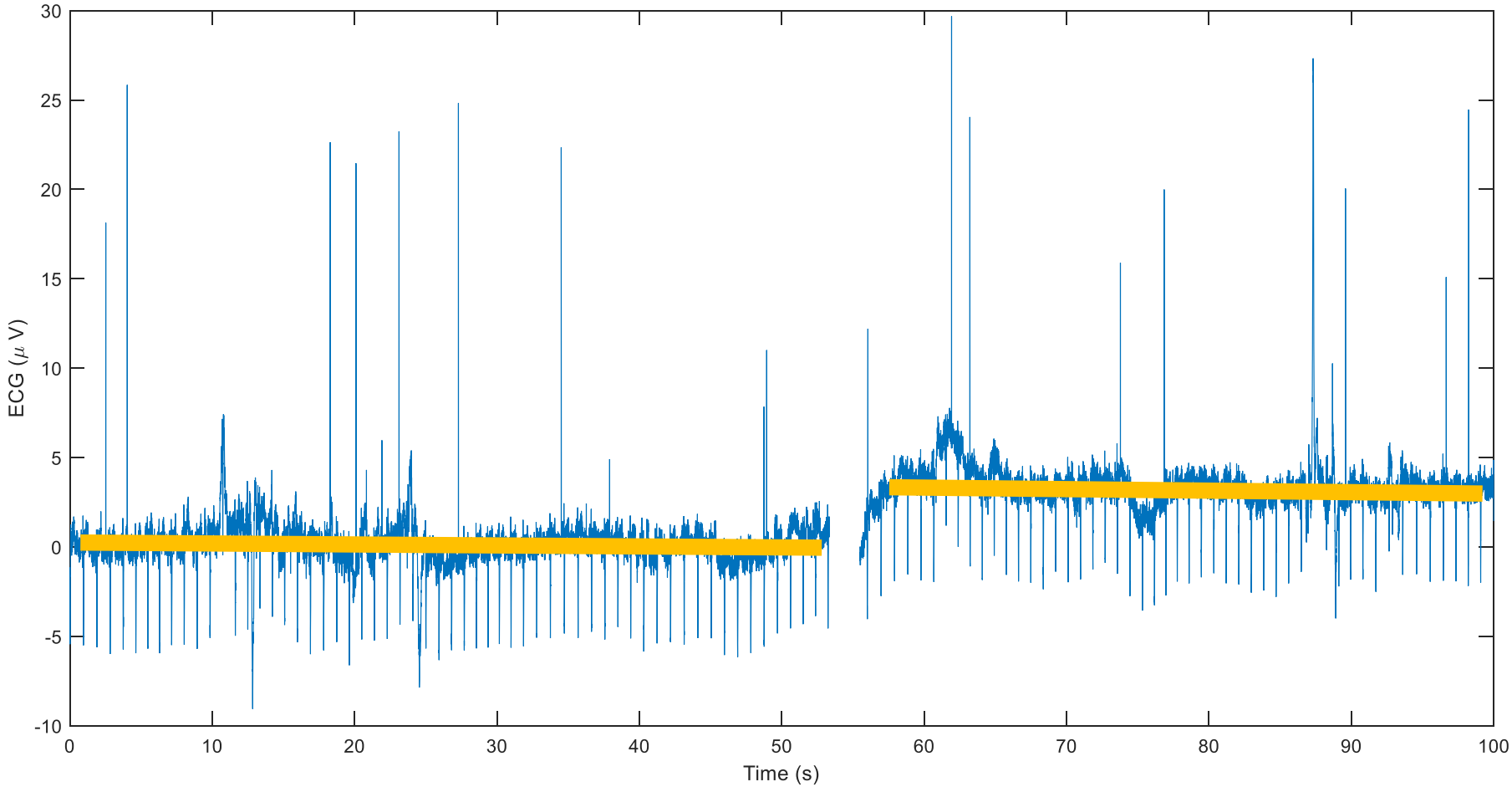
Valeurs aberrantes (outliers) $\rightarrow$ **Suppression du bruit impulsionnel**

Exemple d'un signal ECG



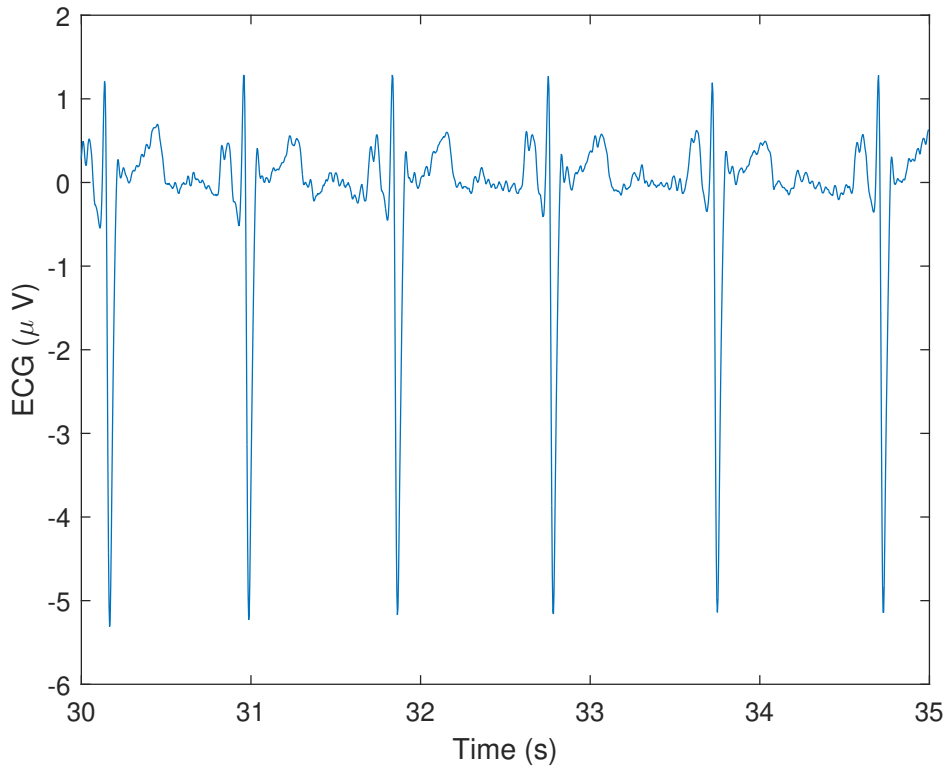
Perte de données : échantillons manquants $\rightarrow$ **Interpolation**

Exemple d'un signal ECG



Rupture de stationnarité $\rightarrow$ **Détection de ruptures** (voir Séance 8)

Exemple d'un signal ECG



Résultat après pré-traitements

On retrouve les battements cardiaques : permet le suivi du patient et le calcul du rythme cardiaque

Programme de la séance

- Connaître les techniques afin de nettoyer les données temporelles avant de les utiliser en apprentissage

Session 5 : Pre-processing

Plan du cours

Plan du cours

Principe du débruitage

- Le modèle le plus courant pour modéliser une erreur de mesure est le modèle signal+bruit déjà évoqué au cours précédent. Etant donné un signal $x[n]$ déterministe corrompu par un bruit $b[n]$, le signal bruité $y[n]$ s'écrit :

$$y[n] = x[n] + b[n]$$

- Une des tâches fondamentales en traitement du signal consiste à débruiter le signal mesuré, c'est à dire à retrouver le signal $x[n]$ à partir du signal $y[n]$
- C'est une tâche impossible à réaliser de façon parfaite, car par définition on ne connaît pas $b[n]$
- Néanmoins, si on connait quelques propriétés du bruit et/ou du signal, on peut tout de même améliorer la qualité du signal

Hypothèses

$$y[n] = x[n] + b[n]$$

- On suppose souvent que $b[n]$ est un bruit blanc ce qui implique :
  - $b[n]$ est à moyenne statistique nulle et tous les échantillons de b sont indépendants
  - $x[n]$ et $b[n]$ sont décorrélés
  - La DSP $\Gamma_b[k]$ est une constante égale à σ^2 (variance du bruit) : le bruit affecte donc toutes les fréquences entre $-\frac{F_e}{2}$ et $+\frac{F_e}{2}$
- On peut également avoir des hypothèses sur $x[n]$, qui sont souvent liées à l'occupation spectrale. En effet, les signaux présents dans le monde réel ont souvent une occupation spectrale dans une certaine bande de fréquence $B = [f_{min}, f_{max}]$

Notion de Bruit Blanc Additif Gaussien (BBAG)

Un bruit blanc additif gaussien $b[n]$ est :

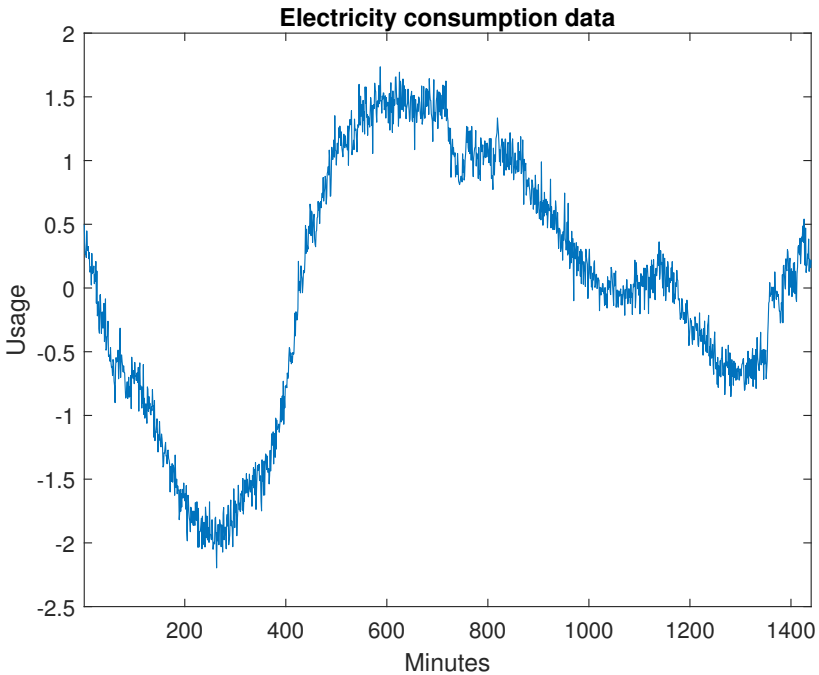
- **Additif** : le bruit corrompt tous les échantillons
- **Blanc** : processus stationnaire à moyenne nulle et où tous les échantillons sont non-corrélés deux à deux

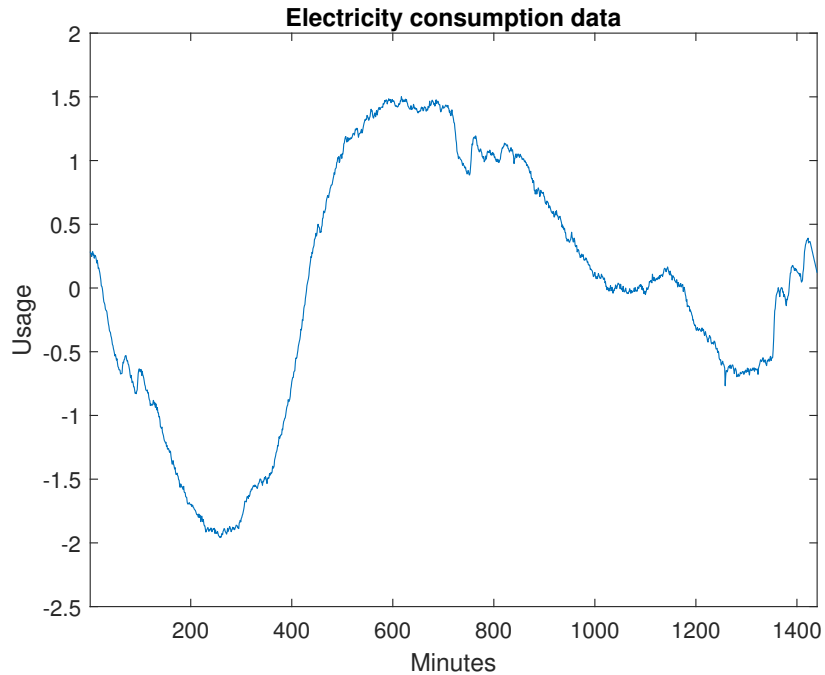
$$\gamma_b[m] = \begin{cases} \sigma^2 & m = 0 \\ 0 & \text{sinon} \end{cases}$$

- **Gaussien** : tous les échantillons sont indépendants et identiquement distribués (i.i.d.) selon la loi

$$b[n] \sim \mathcal{N}(0, \sigma^2)$$

Exemple





Comment supprimer le bruit ?

Cas d'école

- Un signal $x[n]$ échantillonné à $F_e = 100$ Hz, dont on ne connaît pas la TFD ou la DSP mais dont on sait qu'il est en bande de base avec une largeur de bande de $B = 10$ Hz
- On sait que le bruit blanc additif va corrompre toutes les fréquences jusqu'à 50 Hz
- Il y a deux phénomènes :
  - Entre 0 et 10 Hz, il y aura du signal et du bruit : si l'on agit dans cette bande de fréquence, on risque donc d'agir également sur le signal
  - Entre 10 et 50 Hz, on sait qu'il n'y aura a priori que du bruit : on peut donc supprimer ce contenu
- Dans ce cas, on peut débruiter le signal $y[n]$ grâce à un filtre passe-bas de fréquence de coupure $f_c = 10$ Hz

Filtrage linéaire

- La première solution est directement liée aux notions vues en traitement du signal
- Sachant que $\gamma_x[m] = \mathbb{E}[x[n]x[n+m]]$ et en utilisant le fait que $x[n]$ et $b[n]$ sont non corrélés, on obtient que

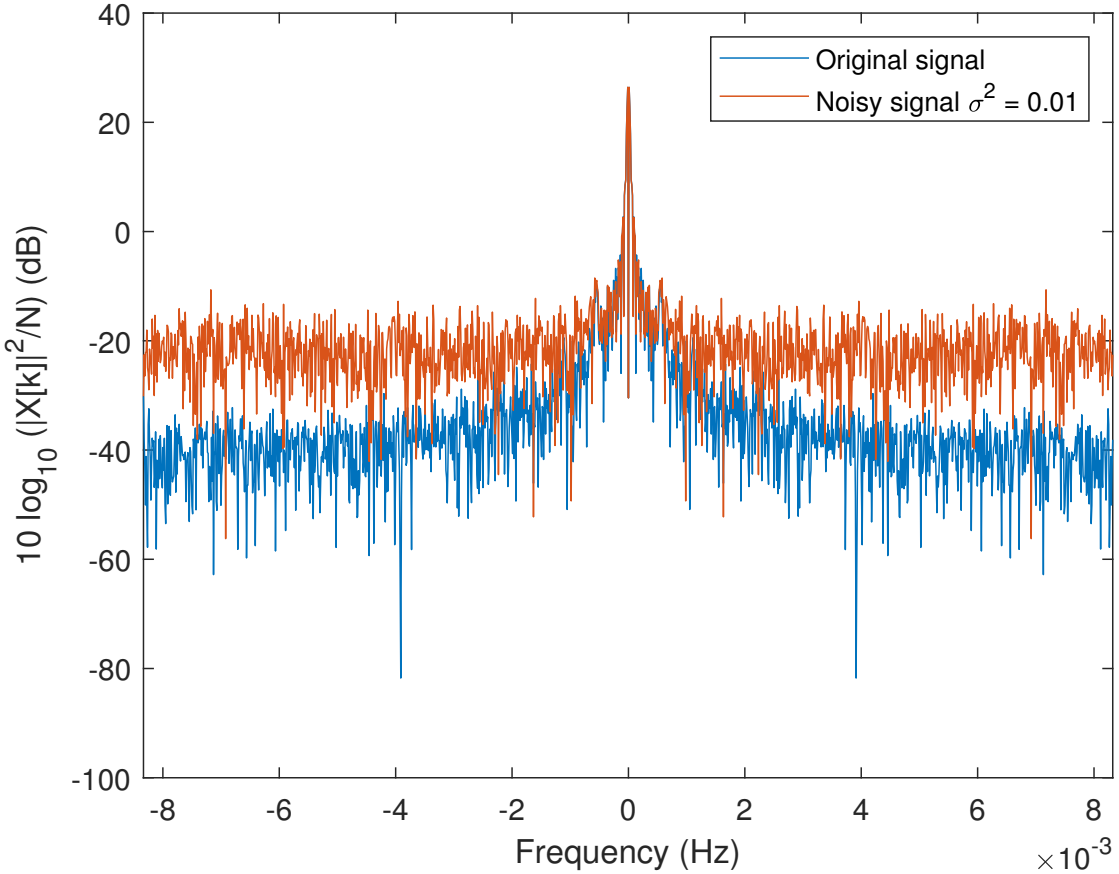
$$\gamma_y[m] = \gamma_x[m] + \gamma_b[m]$$

- En calculant la TFD de cette équation, on a

$$|Y[k]|^2 = |X[k]|^2 + N\sigma^2$$

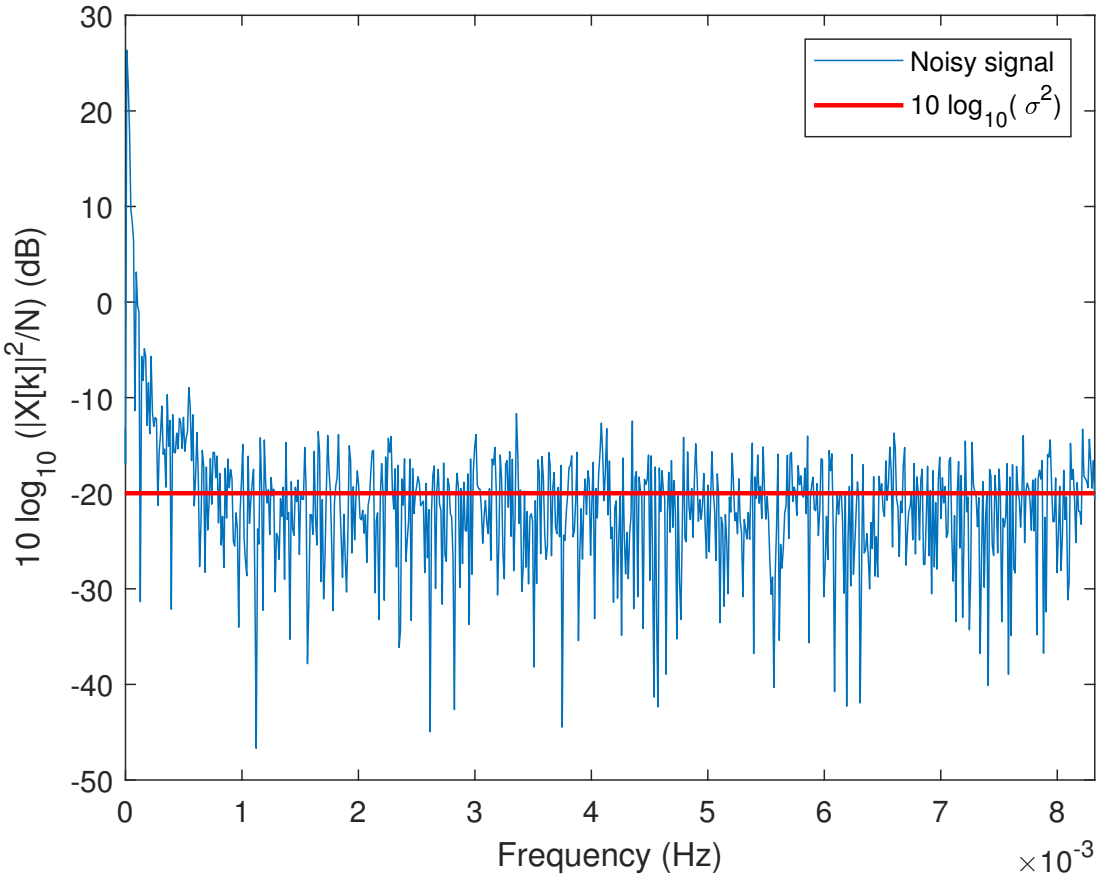
- Ajouter un BBAG revient à ajouter une constante à la TFD du signal

Exemple



Ajouter un BBAG de variance $\sigma^2 = 0.01$ donne une constante de $10\log_{10}(0.01) = 20dB$ sur le log-spectre $10\log_{10}\left(\frac{|X[k]|^2}{N}\right)$ lorsqu'on est en dehors de la largeur de bande du signal

Exemple



En traçant le log-spectre du signal bruité, on peut repérer une constante hors de la largeur de bande, égale à $10\log_{10}(\sigma^2)$ et on peut en conclure que toutes les fréquences supérieures à 0.001 Hz (par exemple) sont susceptibles de ne contenir que du bruit.

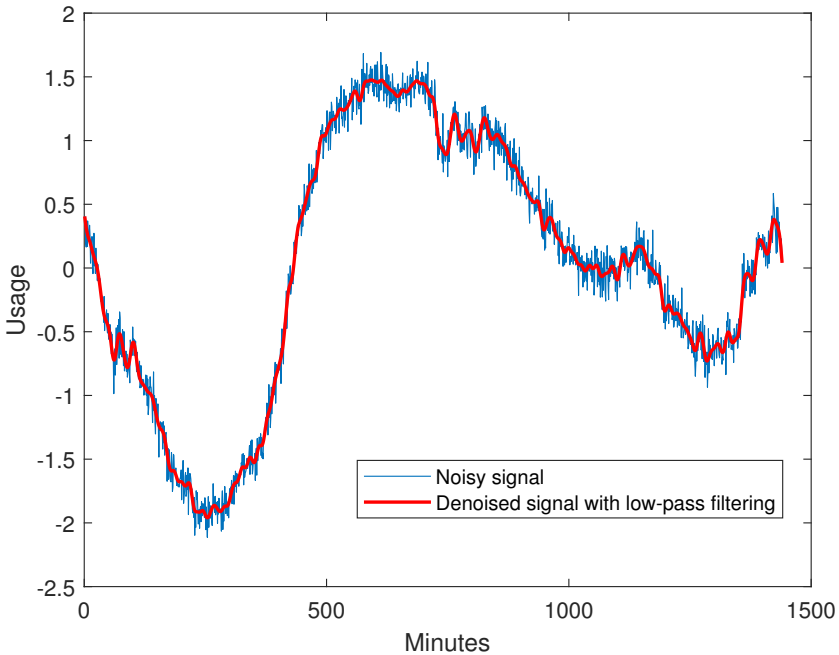
Conception du filtre

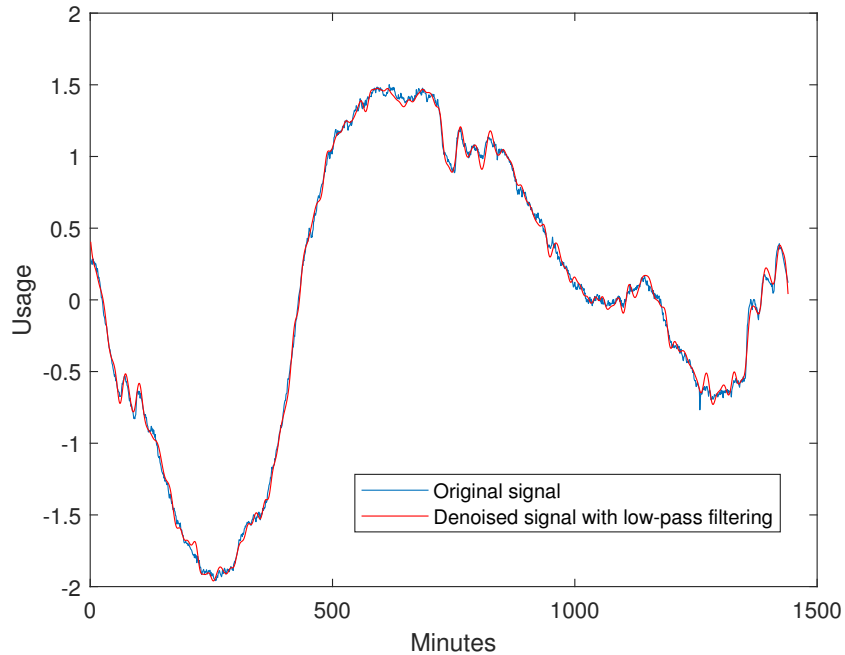
- En observant le log-spectre du signal bruité et en utilisant une connaissance préalable soit sur la variance du bruit, soit sur la largeur du bande du signal d'intérêt, on peut déterminer le type de filtre ainsi que la fréquence de coupure associée
- Il s'agit ensuite d'utiliser un filtre numérique. Deux filtres utiles :
  - **Filtre à moyenne glissante L :**

$$\hat{x}[n] = \frac{1}{L} \sum_{k=1}^{L-1} y[n-k]$$

    Filtre passe-bas avec une fréquence de coupure $f_c \approx \frac{0.442947 \times F_e}{\sqrt{L^2-1}}$
  - **Filtres de Butterworth :** peuvent être passe-bas, passe-bande, etc...

Exemple



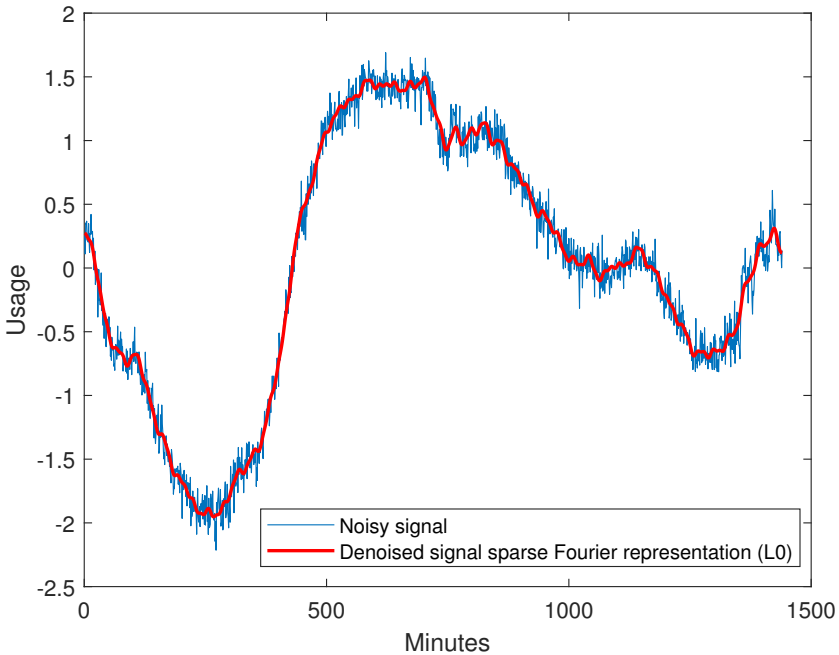


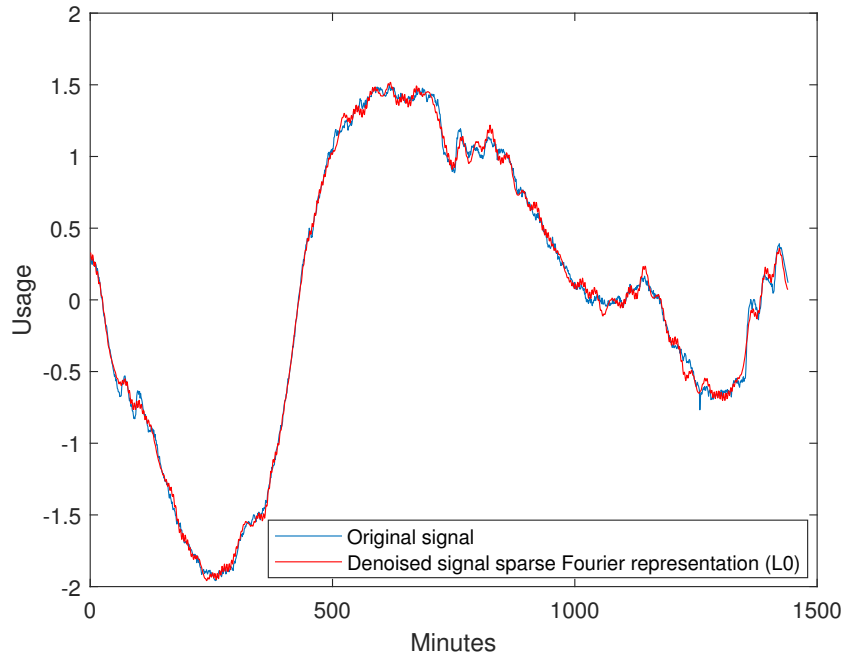
Filtrage passe-bas (Butterworth d'ordre 4) avec $f_c = 0.001$ Hz

Débruitage par dictionnaire

- Une autre solution consiste à utiliser le codage parcimonieux vu dans la séance précédente
- L'idée est la suivante : en forçant une décomposition parcimonieuse du signal, les composantes *signal* seront capturées, mais pas celles liées au bruit
- Pour le dictionnaire on peut utiliser soit des dictionnaires connus (Fourier, ondelettes, polynomes...), soit faire une étape d'apprentissage de dictionnaire sur des trames du signal

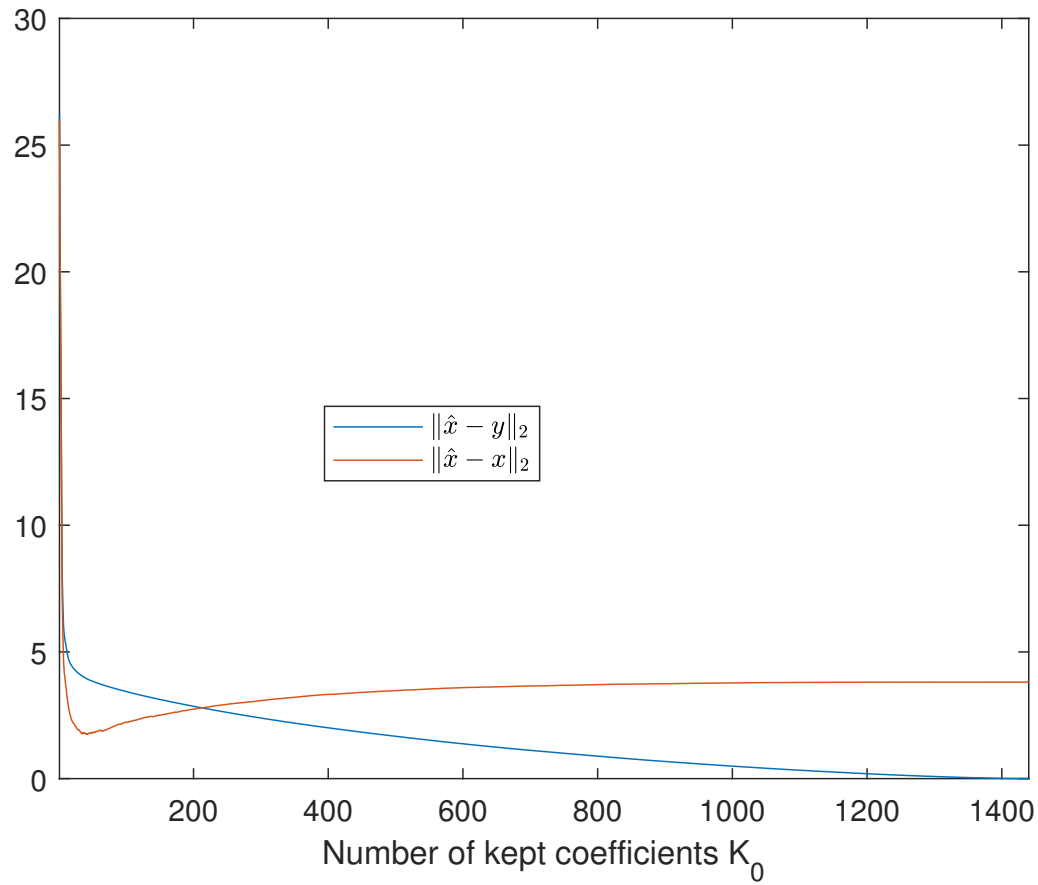
Débruitage par dictionnaire fixe





Matching pursuit avec un dictionnaire de Fourier et $K_0 = 40$

Débruitage par dictionnaire fixe



Influence du paramètre K_0 sur les performances
Bleu : distance au signal bruité, rouge : distance au signal original

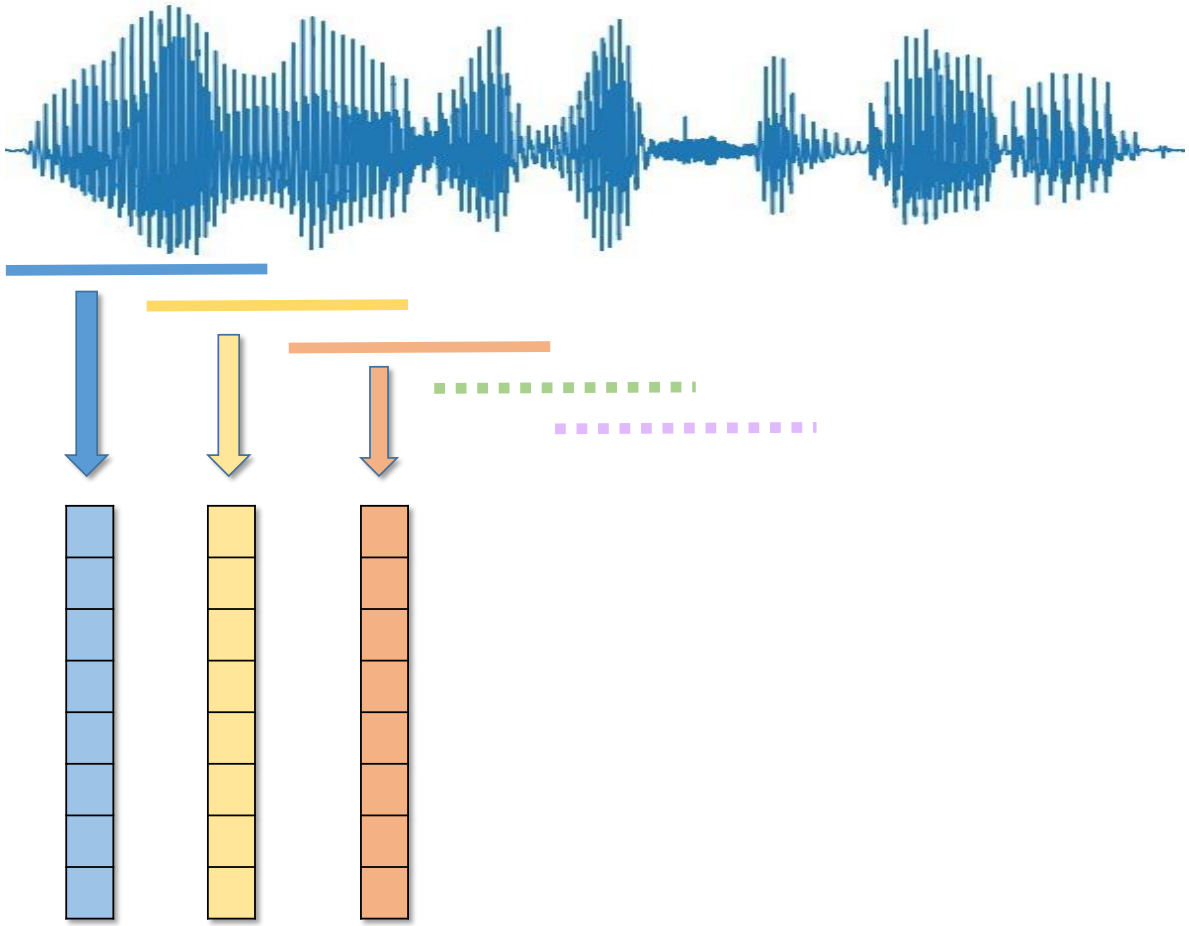
Comment fixer les paramètres K_0 ou λ ?

- Les paramètres dépendent forcément du dictionnaire choisi
- On peut par exemple tracer la RMSE entre le signal courant et le signal initial pour différentes valeurs de K_0 et choisir la valeur qui permet d'observer un coude sur la courbe
- Certaines fonctions Python proposent automatiquement de renvoyer les activations pour plusieurs valeurs de λ, permettant ainsi de choisir la valeur qui convient le mieux

Débruitage par dictionnaire adaptatif

- Au lieu d'utiliser des dictionnaires sur étagères, on peut apprendre une représentation directement à partir du signal grâce à des algorithmes d'**apprentissage de dictionnaire** (voir Séance précédente)
- On formera pour cela une **matrice de trajectoire X** qui permettra de stocker des trames du signal

Matrice de trajectoire



N_w : taille de fenêtre, N_o : taille de recouvrement

N_w lignes, $N_f = \lfloor \frac{N-N_w}{N_w-N_o} \rfloor + 1$ colonnes

Apprentissage de dictionnaire adaptatif

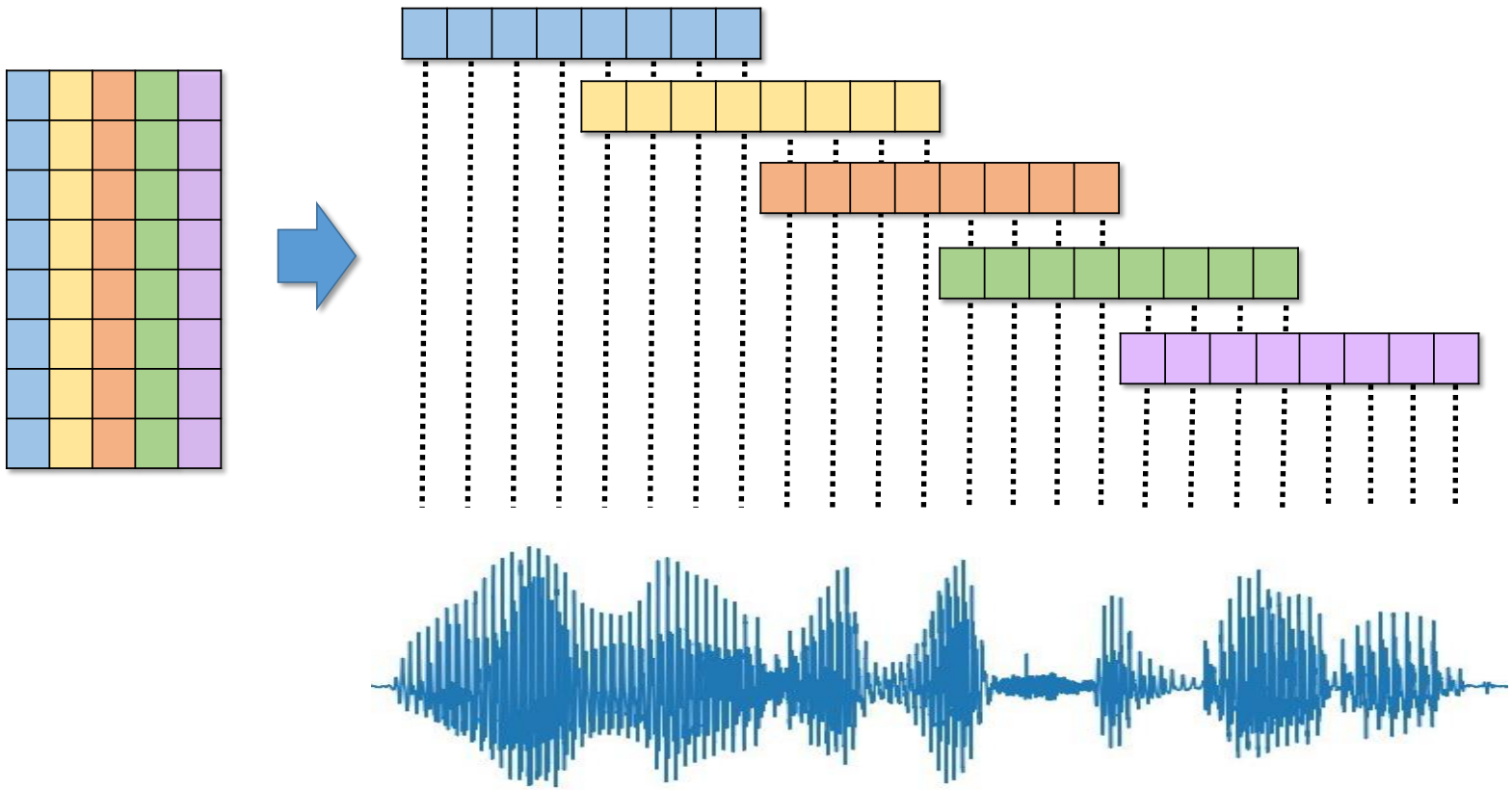
Grâce aux algorithmes décrits dans la Séance précédente, on peut calculer une approximation de la matrice de trajectoire

$$\hat{\mathbf{X}} = \mathbf{DZ} \approx \mathbf{X}$$

- $\mathbf{D} \in \mathbb{R}^{N_w \times K}$: dictionnaire composé de K atomes
- $\mathbf{Z} \in \mathbb{R}^{K \times N_f}$: activations parcimonieuses (degré de parcimonie spécifié par K_0 ou λ)

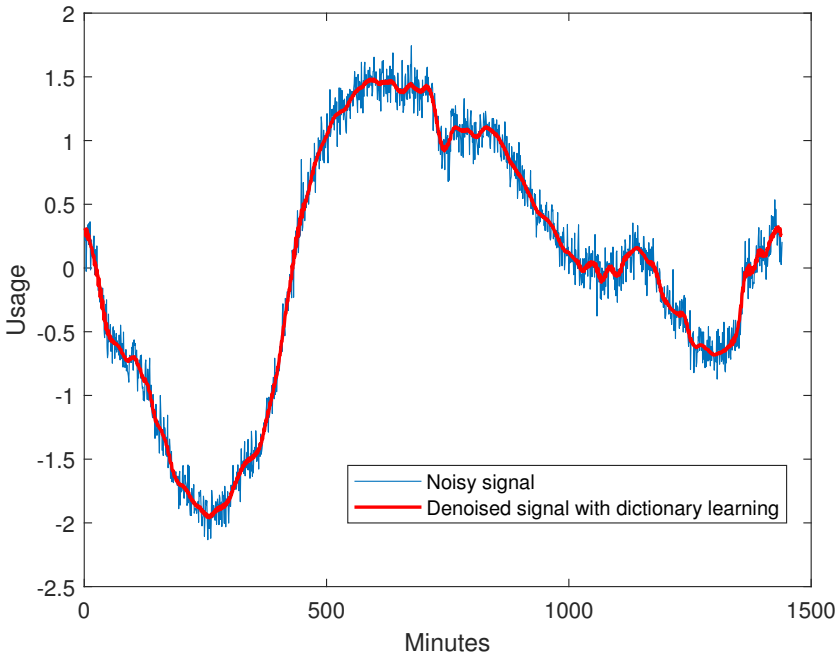
Chaque trame est approximée comme une combinaison linéaire d'un petit nombre d'atomes appris

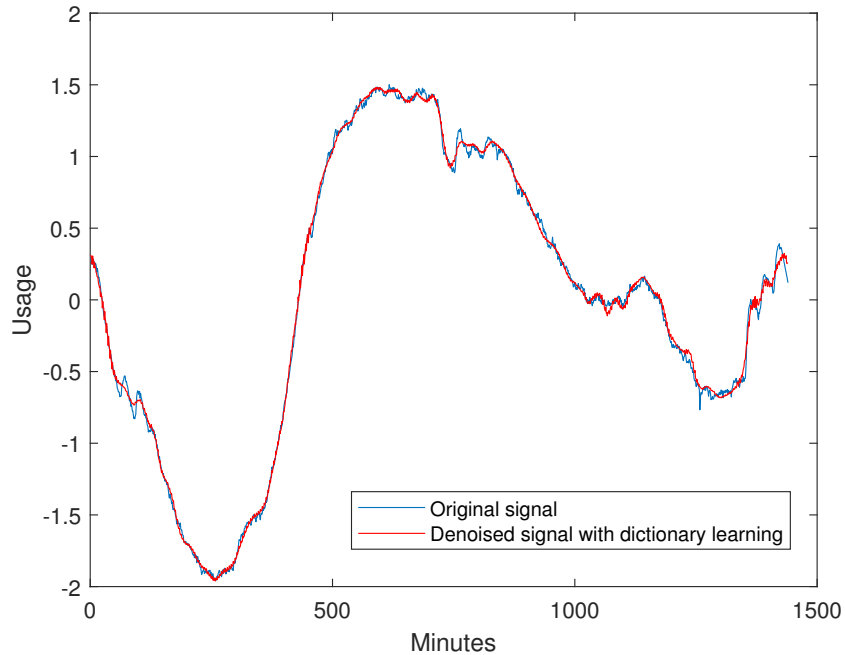
Reconstruction à partir de la matrice de trajectoire



Dépliement de la matrice de trajectoire et moyennage le long des trames en recouvrement

Exemple





Apprentissage de dictionnaire avec $K = 5$, $K_0 = 2$, $N_w = 32$, $N_o = 28$

Plan du cours

Notion de tendance

- Contrairement à la notion de bruit blanc qui est une notion définie proprement du point de vue statistique, la notion de tendance (baseline) est beaucoup plus floue
- On appelle en général tendance toutes les variations *lentes* par rapport à la dynamique du signal
- Pour étudier le signal il faut donc le décomposer en deux sous-signaux : un signal qui donne les tendances générales (évolutions lentes) et un signal qui conserve uniquement les variations (saisonnalité)
- Selon les cas, le signal d'intêret sera soit la tendance soit le résidu

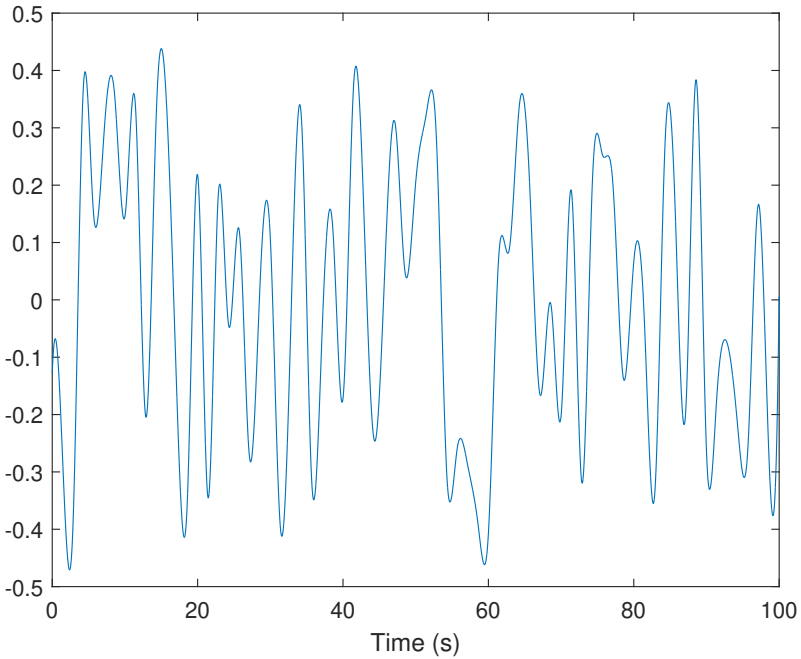
Modèle associé

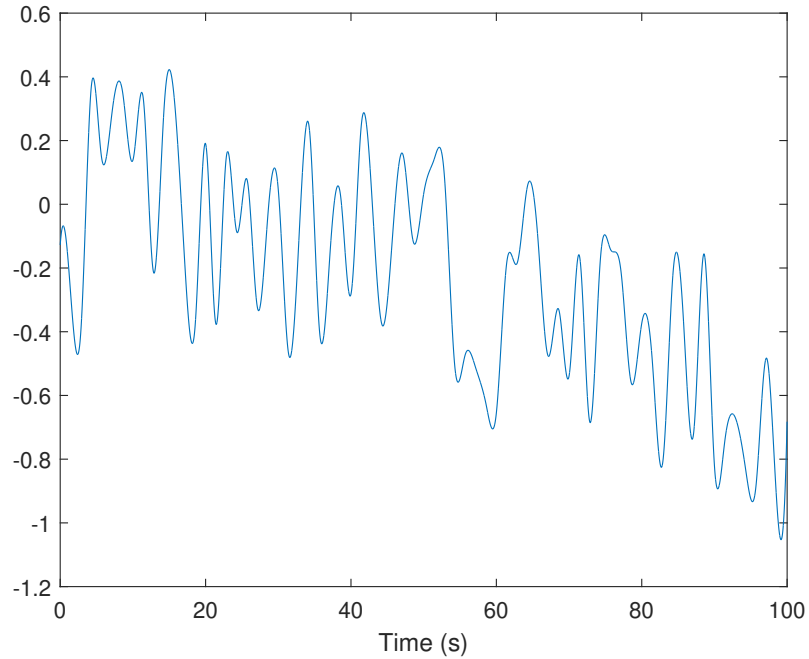
$$x[n] = x^{tendance}[n] + x^{saisonnalite}[n]$$

Plusieurs stratégies :

- Le signal de tendance est constitué de basses fréquences : filtrage passe-bas avec les filtres déjà vus
- Approximation du signal de tendance par des fonctions à variations douces (constante, linéaire, quadratique...) : voir Séance 3

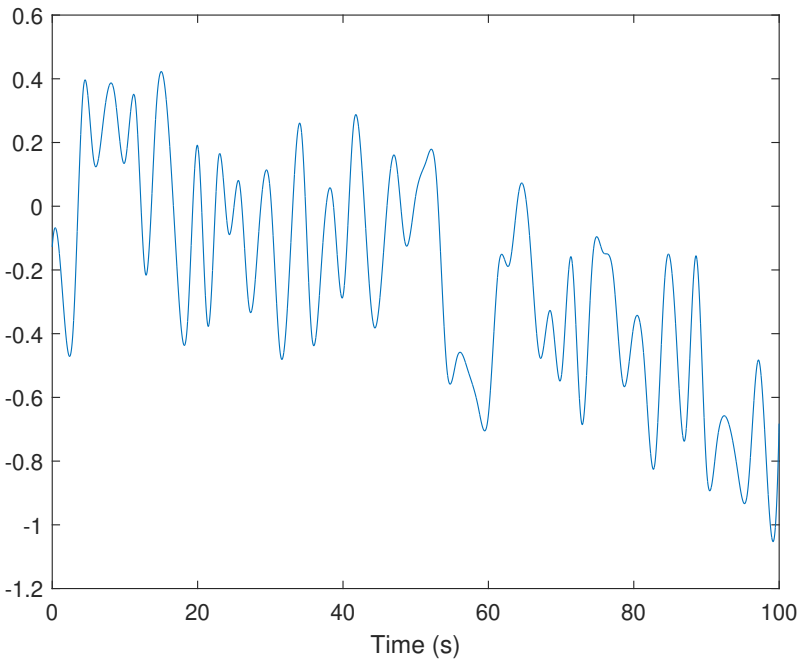
Exemple

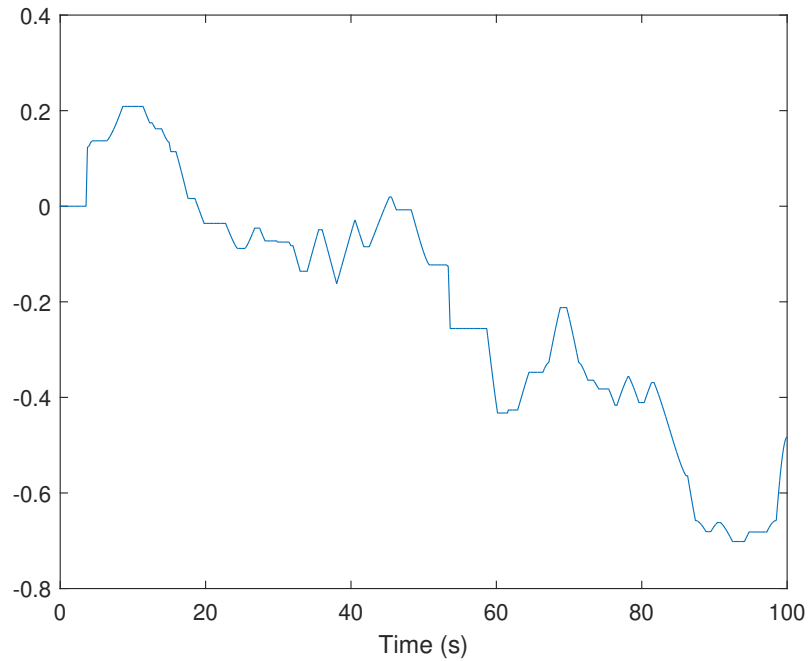




Signal sans tendance/avec tendance

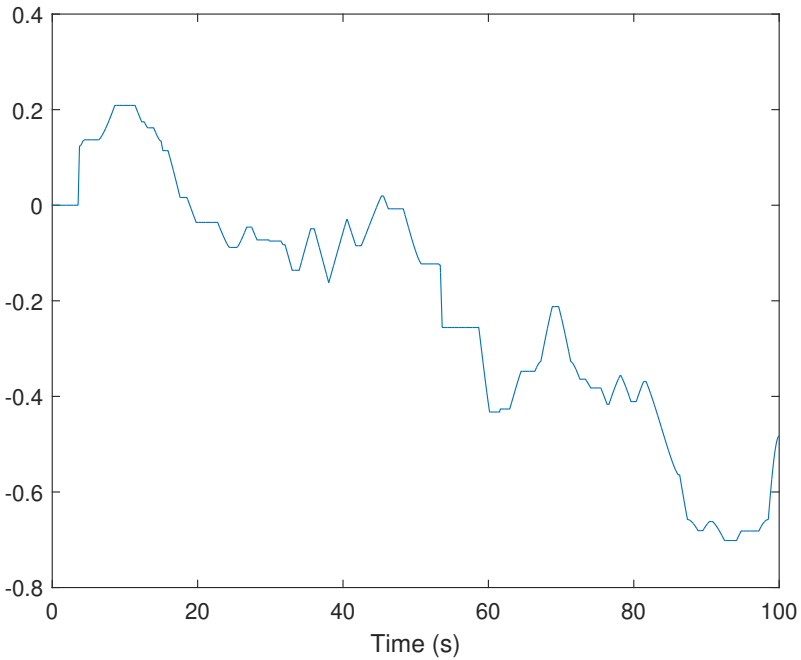
Exemple

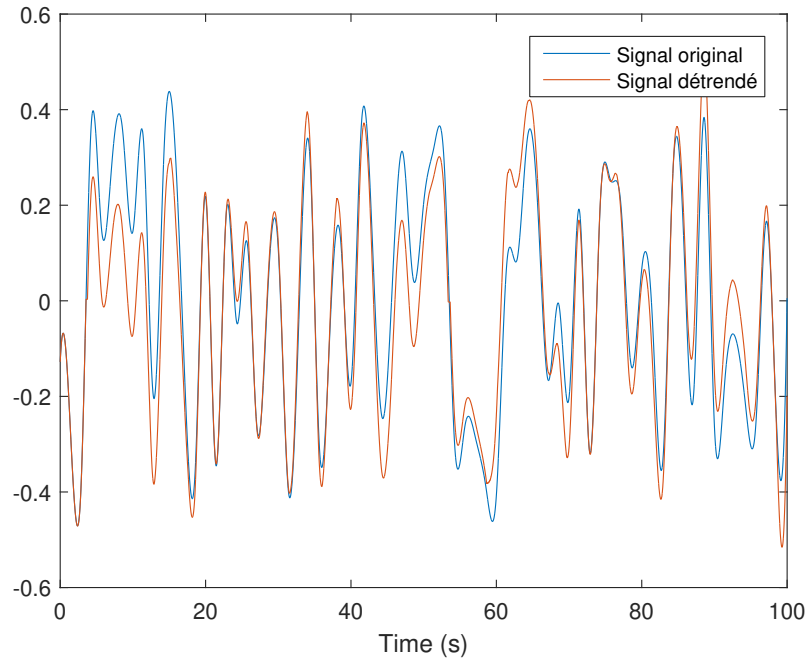




Utilisation d'un filtre médian sur une fenêtre de 15 secondes

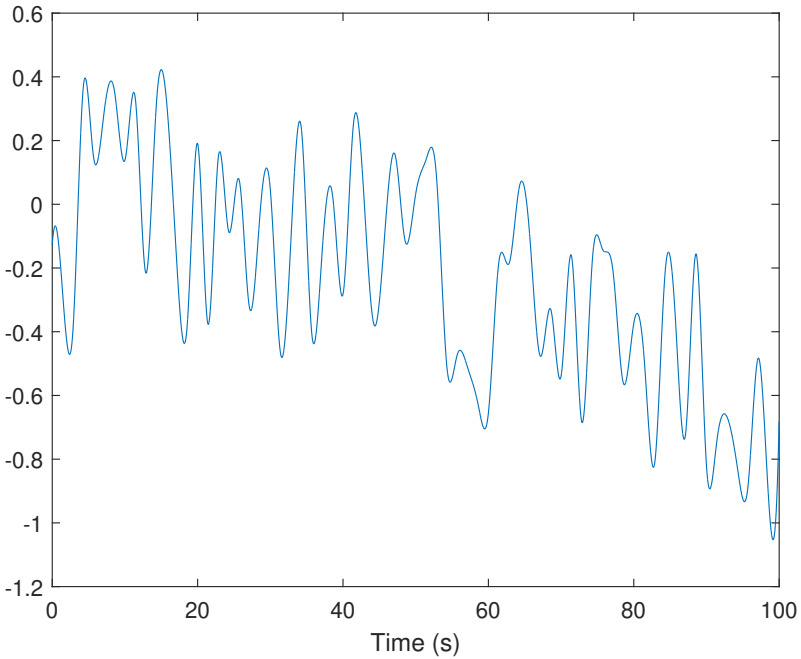
Exemple

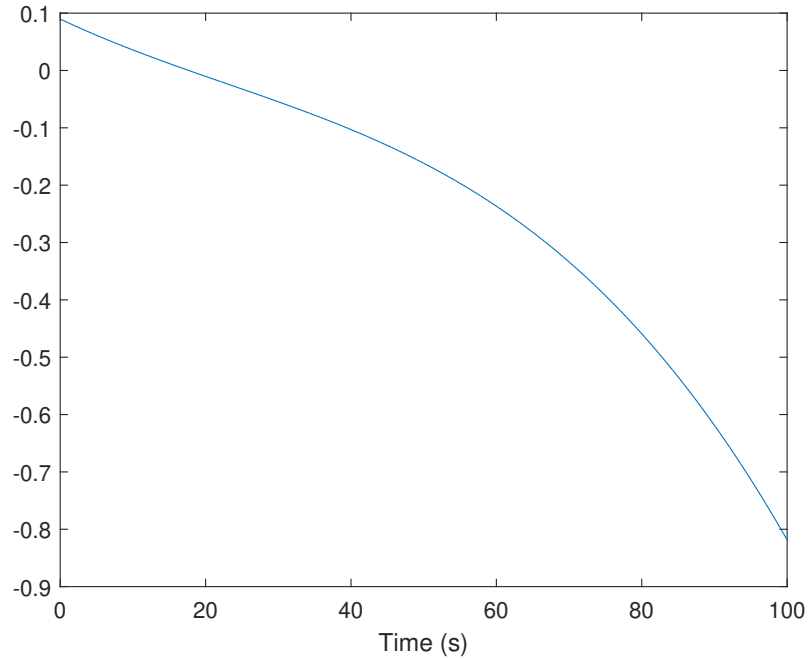




Utilisation d'un filtre médian sur une fenêtre de 15 secondes

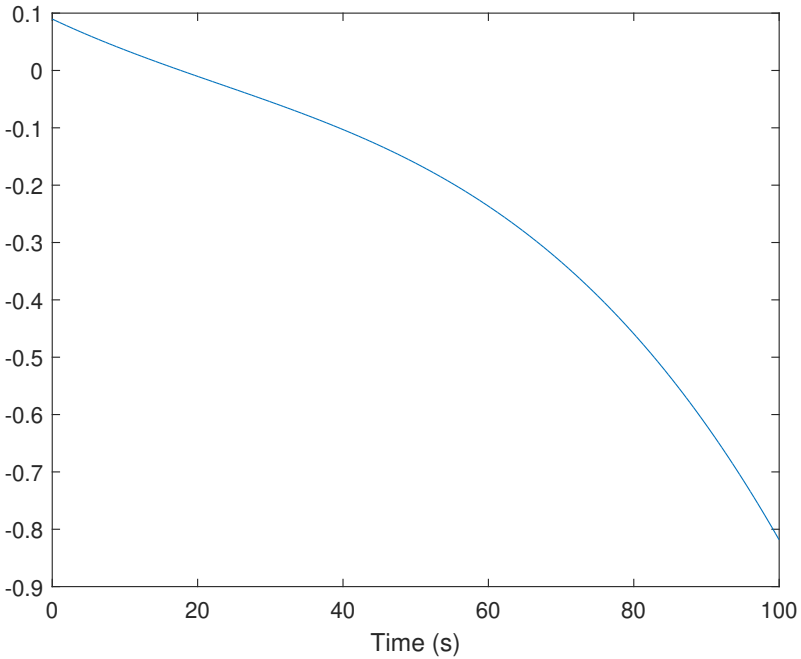
Exemple

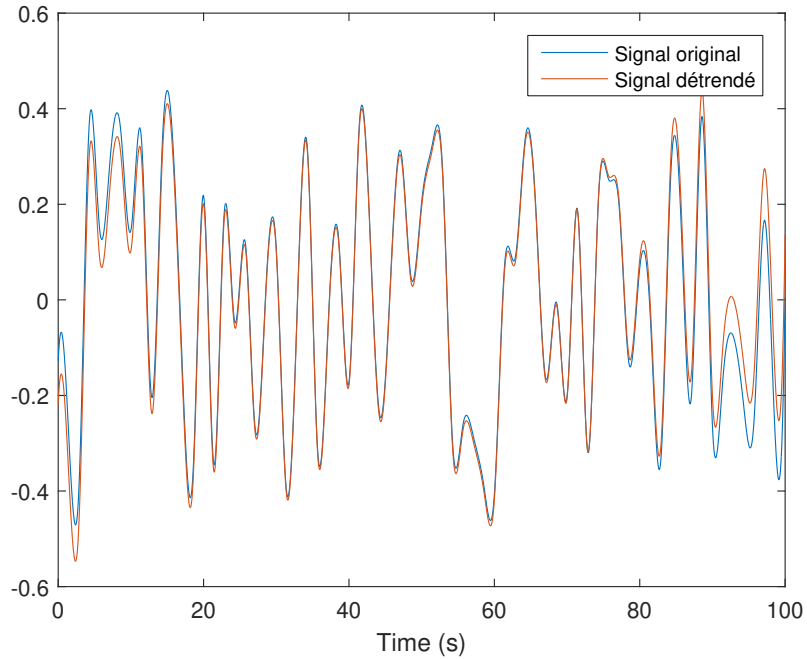




Regression sur les fonctions cst, t, t^2 et t^3

Exemple





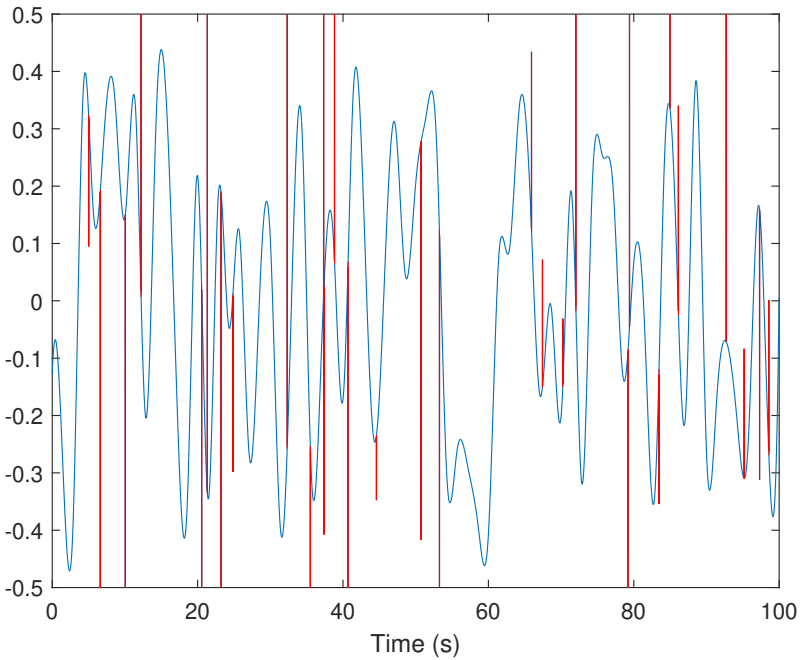
Regression sur les fonctions cst, t, t^2 et t^3

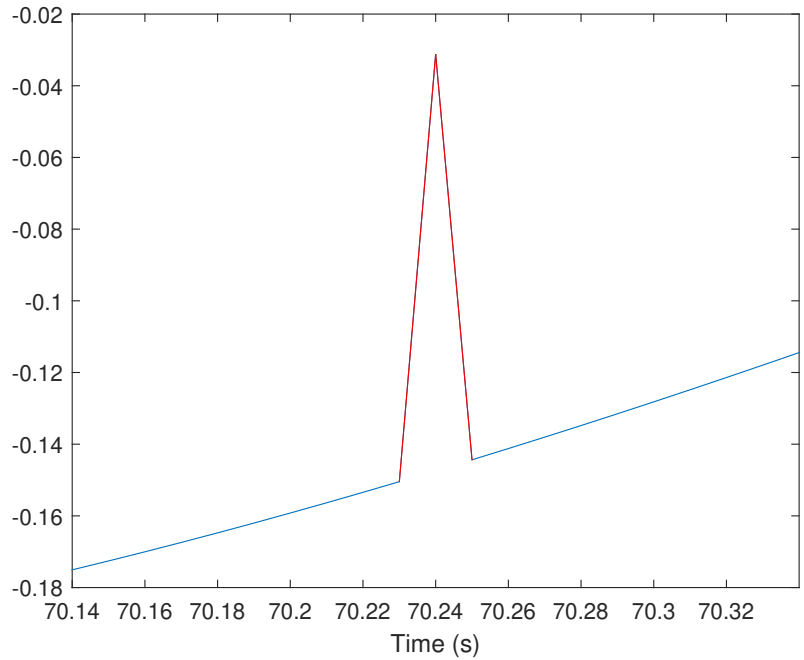
Plan du cours

Bruit impulsionnel

- A l'inverse du bruit blanc additif qui corrompt tous les échantillons, le **bruit impulsionnel** est un phénomène local qui n'affecte qu'un petit nombre d'échantillons dans le signal
- Il s'agit dans la plupart des cas de mesures qui ont raté sur un petit intervalle de temps
- Ce bruit crée des hautes fréquences artificielles qui peuvent fausser les traitements et les interprétations
- Selon la gravité du phénomène, plusieurs approches peuvent être utilisées

Cas d'échantillons isolés



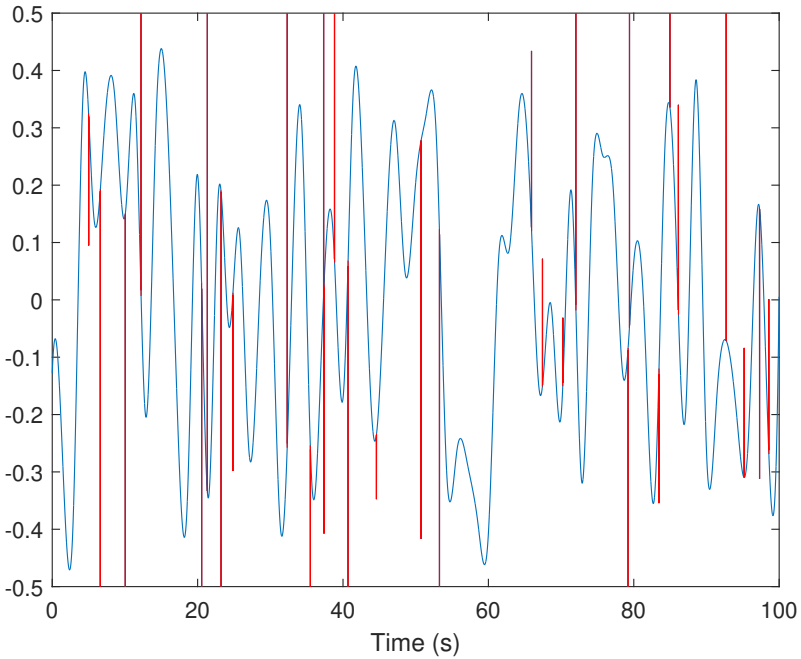


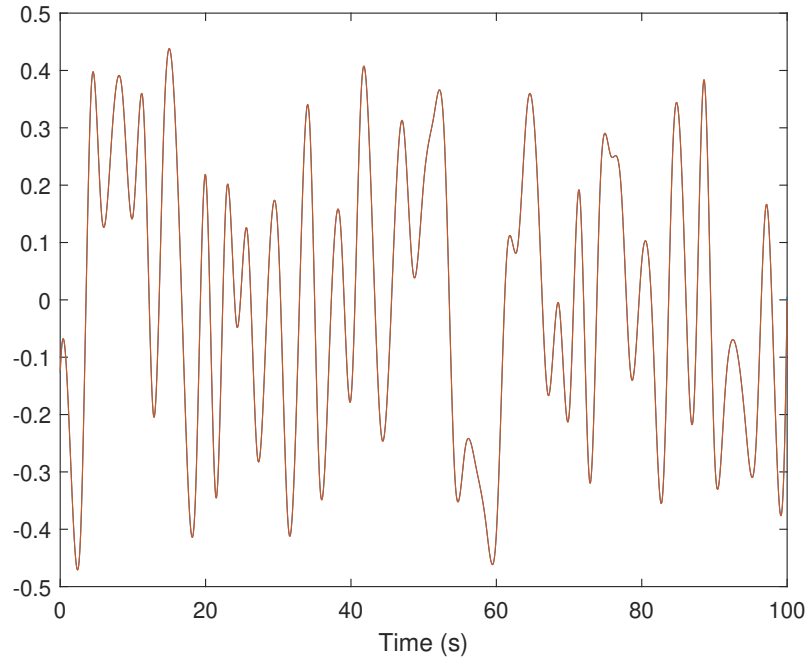
Le bruit affecte des échantillons isolés

Solutions

- Si les valeurs prises par le bruit sont particulièrement grandes par rapport à celles du signal, on peut les repérer en observant l'histogramme des valeurs prises par le signal : ceci est similaire à la détection d'outliers dans des données statistiques
- Si ce n'est pas le cas, on peut débruiter le signal grâce à un filtre médian avec une fenêtre temporelle de 3 échantillons
  Exemple : signal original $[0.3 \;\; 0.4 \;\; 0.45]$ et signal bruité $[0.3 \;\; 0.9 \;\; 0.45]$
  - Si filtre moyenneur, on remplace la valeur 0.9 par la moyenne des trois échantillons : 0.55
  - Si filtre médian, on remplace cette valeur par la médiane : 0.375

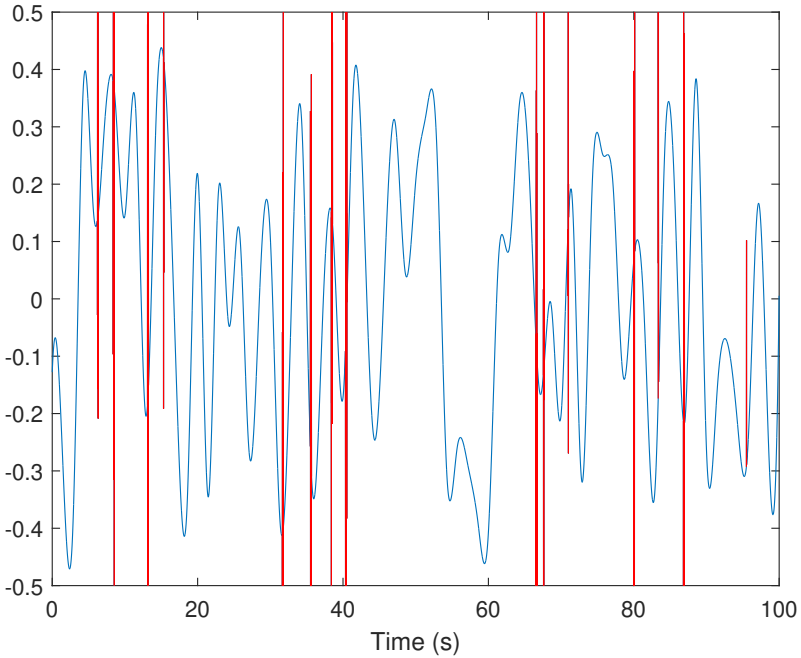
Cas d'échantillons isolés

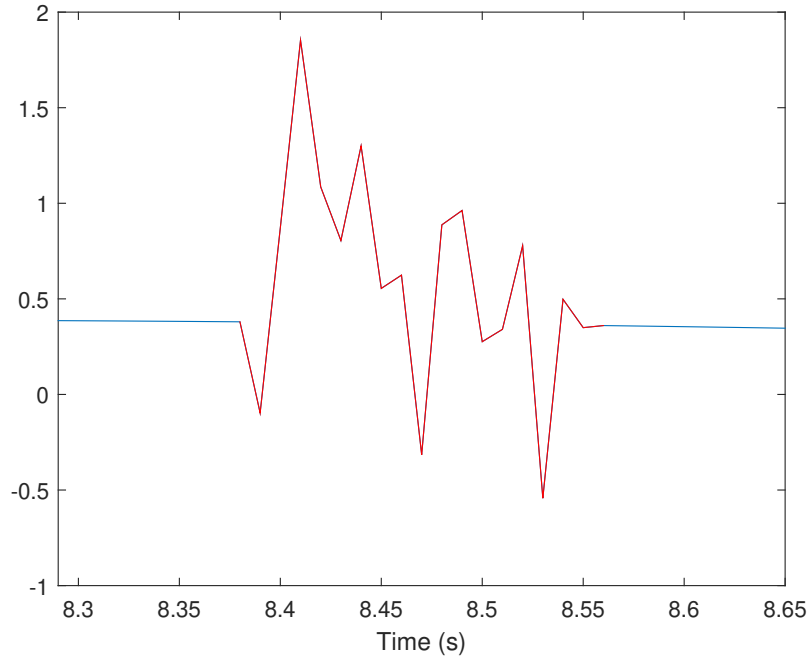




Reconstruction parfaite grâce à un filtre médian (3 échantillons)

Cas d'échantillons contigus





Le bruit affecte des groupes d'échantillons contigus

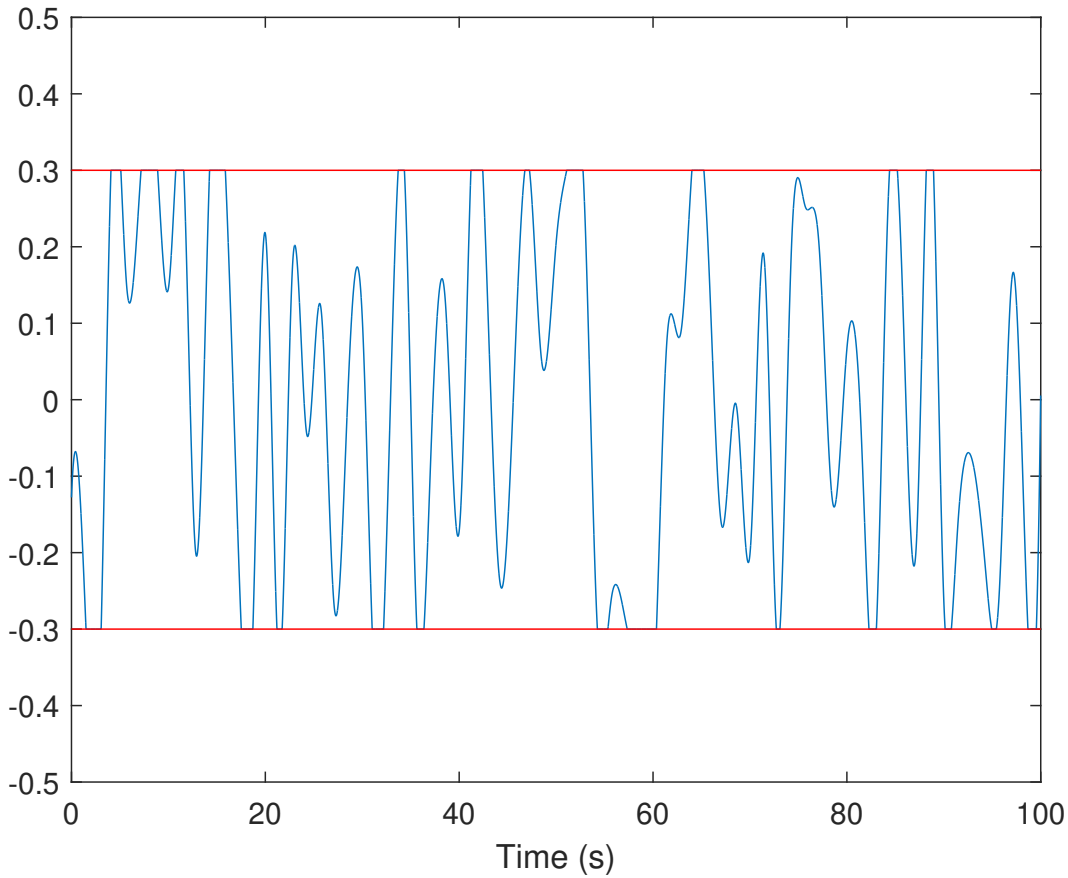
Cas d'échantillons contigus

- Si le bruit concerne des groupes d'échantillons contigus, un simple filtrage n'est pas suffisant
- Le débruitage s'effectue dans ce cas en deux étapes :
  - Une étape de **détection**, qui consiste à repérer quels sont les échantillons bruités

$$\mathcal{T} \in 0 \dots N-1$$

  - Une étape de **reconstruction** ou **interpolation** qui consiste à reconstruire ces échantillons

Cas particulier : clipping



Phénomène de clipping : seuillage du à une mauvaise calibration
Ici : détection aisée

Détection du bruit impulsionnel

- Intuitivement, les échantillons corrompus suivent un modèle qui n'est pas celui du signal
- On peut donc se baser sur un modèle : les échantillons qui dévieront de façon significative de ce modèle seront détectés comme étant corrompus
- Procédure :
  1. Choix d'un modèle réaliste pour le signal
  2. Estimation des paramètres à partir des échantillons
  3. Détection des échantillons déviants du modèle

Cas du modèle AR

$$x[n] = -\sum_{i=1}^{p} a_i x[n-i] + b[n]$$

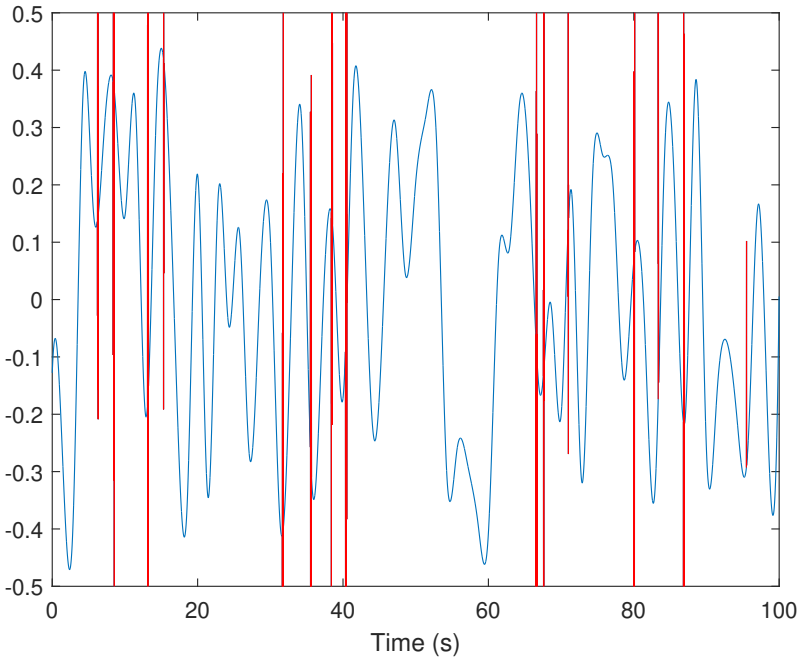
- Choix d'un ordre p et estimation des paramètres $\hat{\mathbf{a}}$ grâce à la fonction d'autocorrélation du signal (voir cours 3)
- En théorie, si l'estimation des paramètres est correcte et si la variance du bruit n'est pas très élevée, la quantité

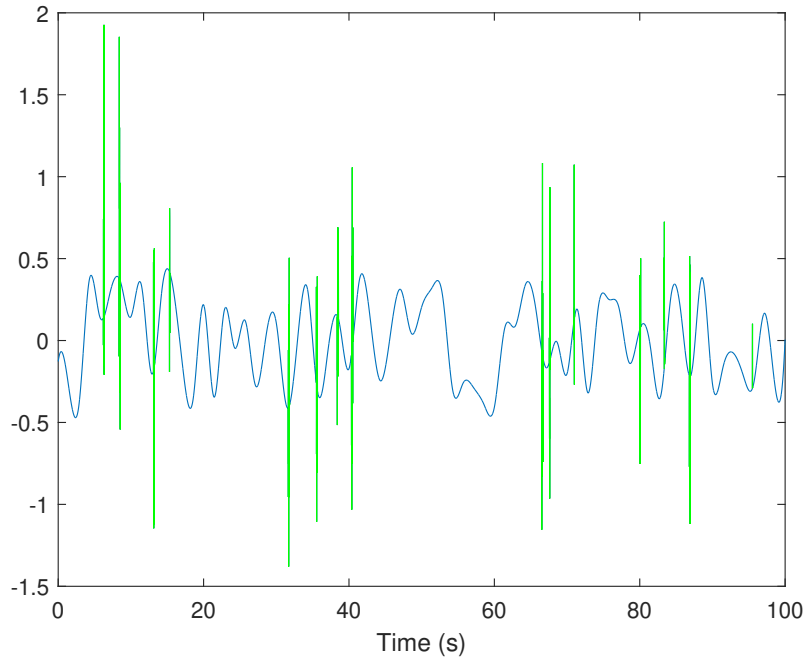
$$d[n] = x[n] + \sum_{i=1}^{p} \hat{a}_i x[n-i]$$

  doit être relativement petite
- On va donc considérer un seuil λ et détecter comme corrompus tous les échantillons n pour lesquels la quantité $|d[n]|$ est supérieure au seuil

$$\mathcal{T} = \{n \text{ tq } |d[n]| > \lambda\}$$

Cas d'échantillons contigus





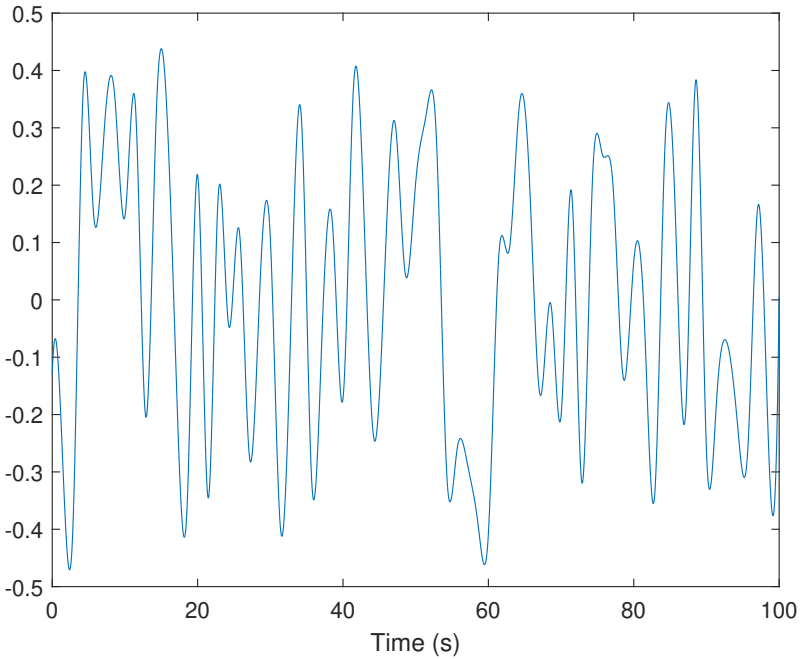
Détection avec un modèle $AR(10)$

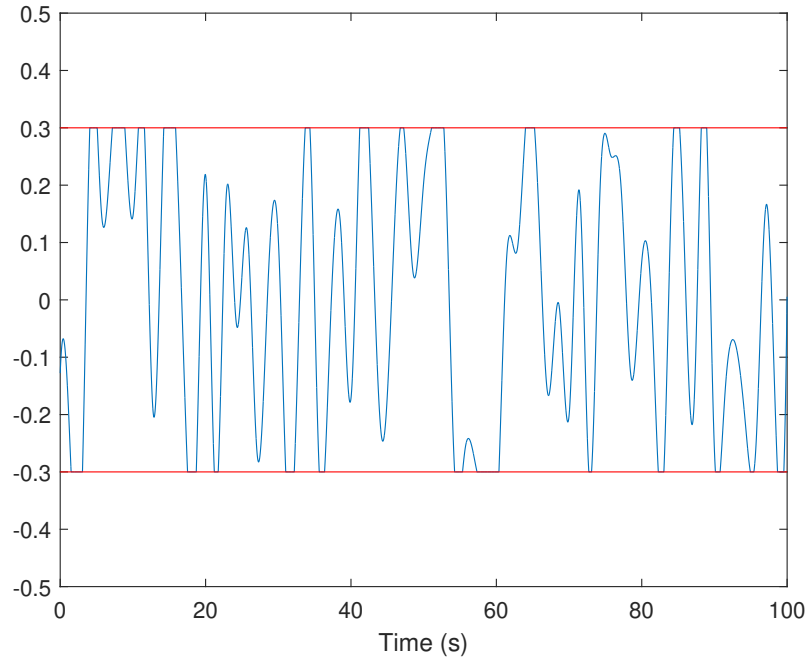
Plan du cours

Intérêt de l'interpolation

- L'**interpolation**, également appelée reconstruction ou inpainting, consiste à remplacer des échantillons inconnus ou que l'on sait corrompus, par des échantillons cohérents avec le reste du signal observé
- Il s'agit d'une tâche courante en apprentissage et en statistiques (complétion de valeurs manquantes) et de nombreux outils issus du ML peuvent être utilisés et adaptés : en revanche ils ne prendront pas en compte l'aspect temporel des données
- Les méthodes d'interpolations prennent en entrée deux éléments : le signal $x[n]$ qui comporte des échantillons manquants/corrompus et l'ensemble $\mathcal{T}$ des échantillons manquants

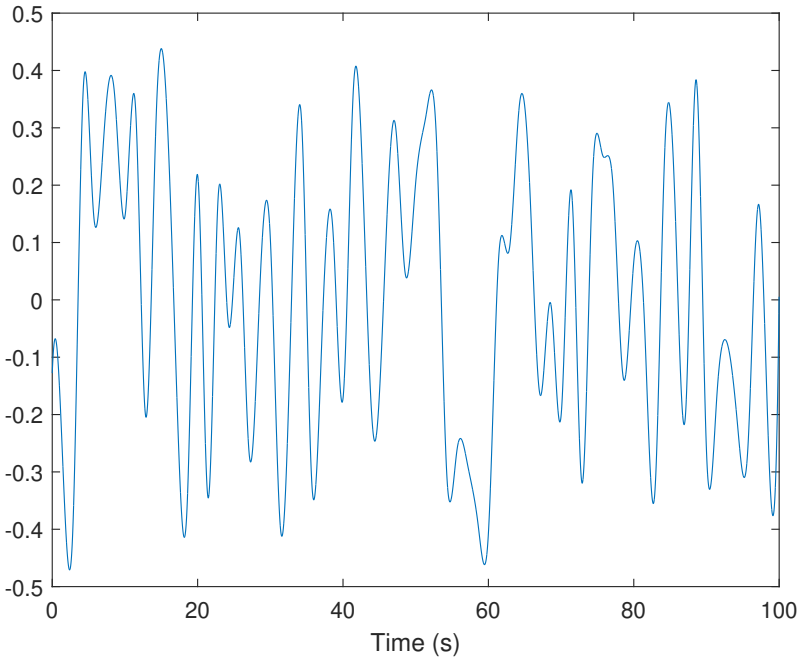
Exemple

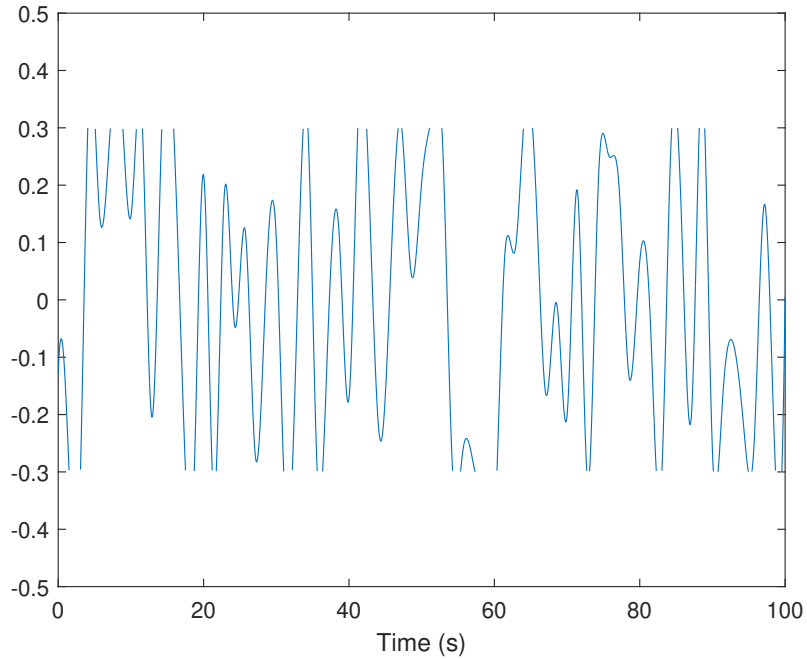




Phénomène de clipping

Exemple





Les échantillons clippés sont considérés comme manquants

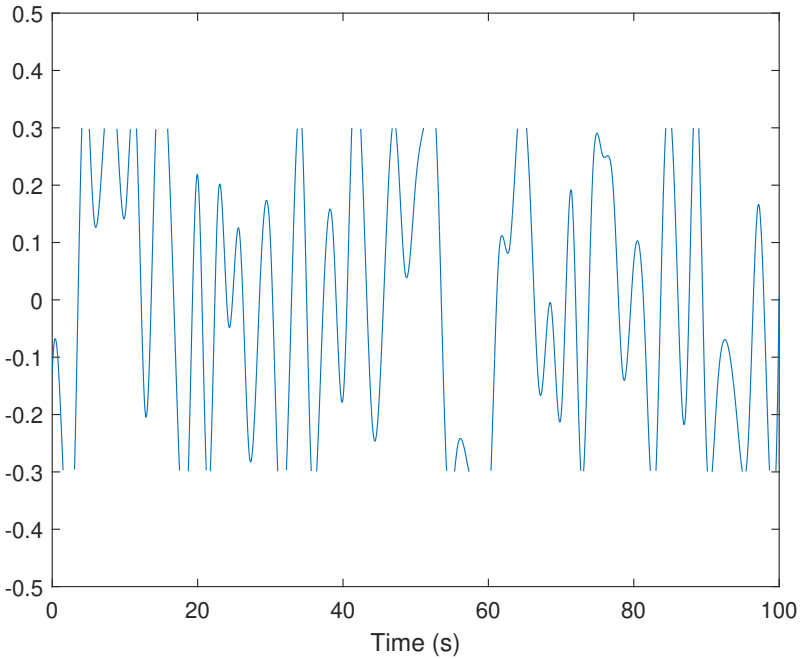
Approximation polynomiale

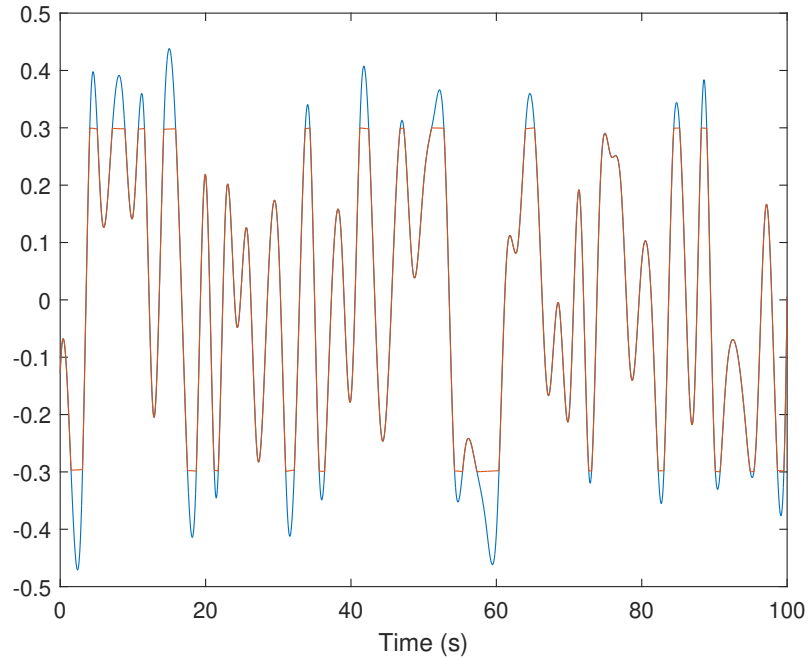
- La façon la plus simple pour reconstruire des échantillons manquants revient à supposer que le signal suit un modèle polynomial
- Deux modèles sont souvent utilisés :
  - Interpolation linéaire : on approxime la portion de signal manquante par une droite

    Condition sur les valeurs prises par la droite au début et à la fin de la portion manquante
  - Interpolation par spline cubique : on approxime la portion de signal manquante par un polynome de degré 3

    Condition sur les valeurs prises par le polynome et sa dérivée au début et à la fin de la portion manquante
- Ces approximations fournissent des résultats acceptables pour des portions de petite taille (quelques échantillons)

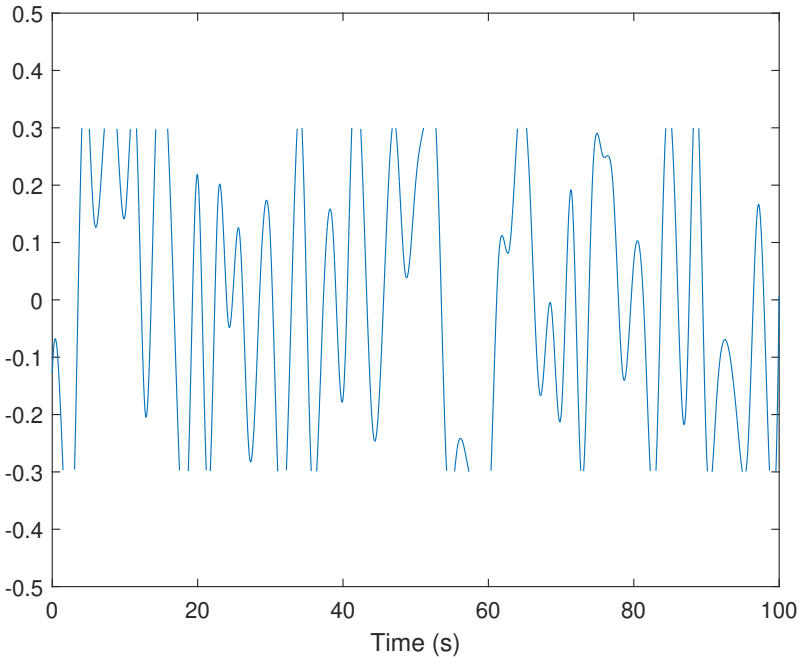
Exemple

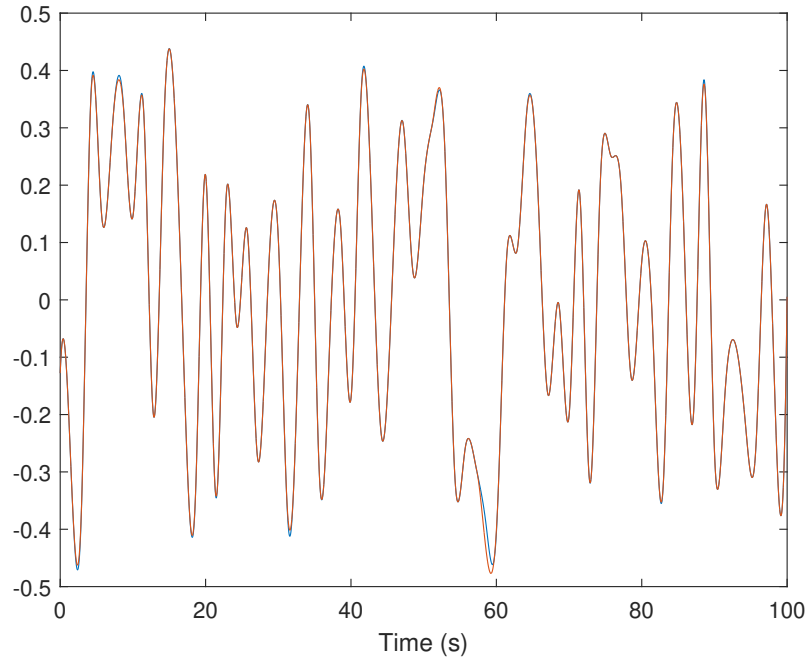




Interpolation linéaire

Exemple





Interpolation par spline cubique

Approximation par modèle

- Dans le cas général de portions d'échantillons manquants relativement grandes, on peut se baser une nouvelle fois sur un modèle statistique pour les données
- Procédure :
  1. Choix d'un modèle réaliste pour le signal
  2. Estimation des paramètres à partir des échantillons non manquants
  3. Remplissage des échantillons manquants par des valeurs cohérentes avec le modèle

Approximation par modèle

- Problème : comment estimer les paramètres sur une série temporelle qui contient des valeurs manquantes ?
- Solution : démarche itérative
  1. On commence par mettre tous les échantillons manquants à une valeur par défaut (zero, recopie du dernier échantillon, interpolation linéaire...)
  2. On apprend les paramètres du modèle
  3. On reconstruit les échantillons manquants en adéquation avec le modèle
  4. On réitère le processus jusqu'à ce que le modèle soit stable

Cas du modèle AR

- Dans le cas du modèle AR, étant donnée une approximation des paramètres $\hat{\mathbf{a}}$, on peut obtenir une reconstruction du signal en supposant que on a

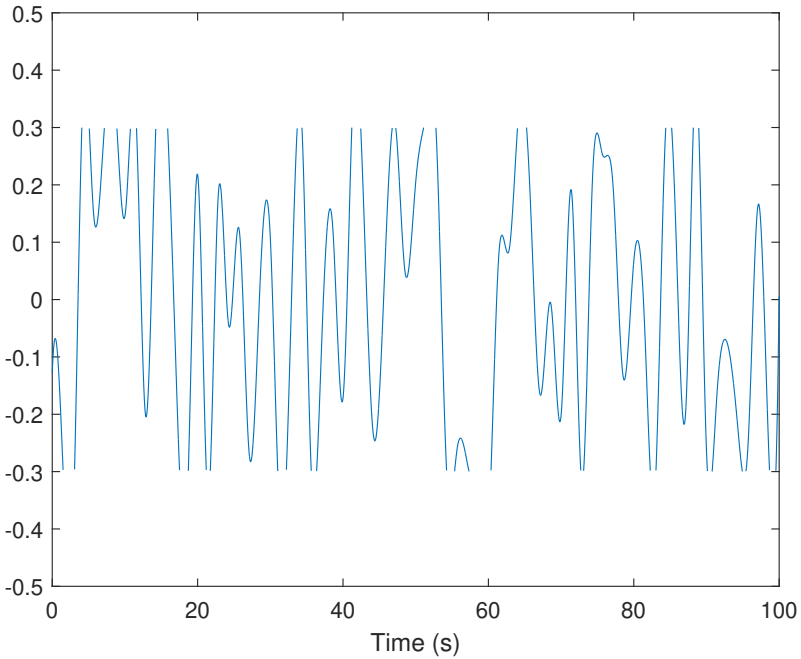
$$x[n] \approx -\sum_{i=1}^{p} \hat{a}_i x[n-i]$$

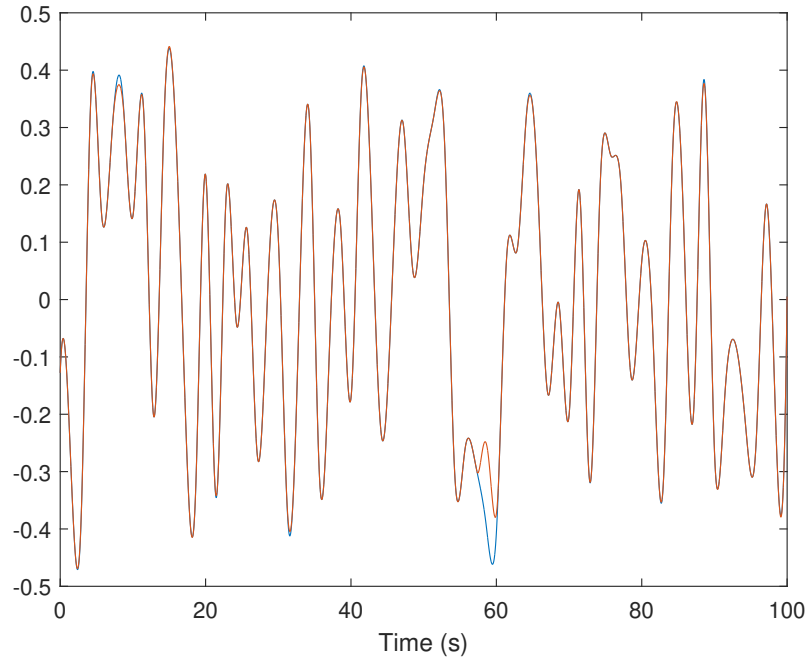
- La reconstruction se fait par minimisation d'un critère :

$$\sum_{n=p+1}^{N-1} \left| x[n] + \sum_{i=1}^{p} \hat{a}_i x[n-i] \right|^2$$

qui se résout grâce à un système d'équations linéaires

Exemple





Interpolation par modèle $AR(10)$

Références

- A. Prochazka, Signal analysis and prediction. Springer Science & Business Media, 2013.
- N. Wiener. Extrapolation, interpolation, and smoothing of stationary time series, vol. 2. (1949).
- M. Watson "Univariate detrending methods with stochastic trends." Journal of monetary economics 18.1 (1986) : 49-75.
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.406.8198
- J. Kantelhardt et al. "Multifractal detrended fluctuation analysis of nonstationary time series." Physica A : Statistical Mechanics and its Applications 316.1-4 (2002) : 87-114.
https://arxiv.org/pdf/physics/0202070.pdf
- A. Adler et al. "Audio inpainting." IEEE Transactions on Audio, Speech, and Language Processing 20.3 (2011) : 922-932.
https://hal.inria.fr/inria-00577079/document
- L. Oudre. Automatic detection and removal of impulsive noise in audio signals. Image Processing On Line, 5 :267-281, 2015.
http://dx.doi.org/10.5201/ipol.2015.64
- C. Truong, L. Oudre, N. Vayatis. Selective review of offline change point detection methods. Signal Processing, 2019
http://www.laurentoudre.fr/publis/TOG-SP-19.pdf
- M. Basseville, I. V. Nikiforov. Detection of abrupt changes : theory and application. Vol. 104. Englewood Cliffs : Prentice Hall, 1993.