OD DataSIM, 2021-2022

MSTAT
Travaux pratiques (Matlab)

1 Variables aléatoires : quelques distributions

On considère une variable aléatoire scalaire X centrée réduite ; on considère les 3 cas suivants :
- X suit une loi de Gauss ;
- X suit une loi uniforme ;
- X suit une loi de mélange de 2 lois normales (composantes équiprobables et de même variance, moyenne des composantes $\pm m$, avec $m = 0.95$; nécessairement, la variance de chaque composante est alors $1 - m^2$).

Dans les 3 cas, en prenant $N = 100$ puis $N = 4000$:

a) Simuler N réalisations indépendantes (fonctions `rand, randn`). [1]

b) Tracer l'histogramme normalisé (`hist, bar`) à partir de ces réalisations [2] et tracer, sur le même graphique, la densité de probabilité théorique correspondante.

c) Estimer moyenne et écart-type (`mean, std`).

d) Visualiser ces N réalisations (nous visualisons donc une réalisation d'un signal stationnaire indépendant).

2 Loi normale bidimensionnelle

On considère la variable aléatoire normale centrée $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$, où X_1 et X_2 sont de variances respectives σ_1^2, σ_2^2, et de coefficient de corrélation ρ.
Simuler $N = 200$ réalisations de $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$, et tracer les couples obtenus dans le plan (x_1, x_2), ainsi que les ellipses de confiance à 91 % calculées pour les valeurs exactes et estimées de la moyenne et de la variance (`mean, cov`).
On prendra $\sigma_1 = 2$, $\sigma_2 = 5$, $\rho = 0,9$.

3 Autocorrélation et analyse spectrale. Cas d'un signal indépendant

Pour un signal stationnaire centré $X = (X[t])_{t\in\mathbb{Z}}$, la fonction d'autocorrélation $(c[\tau])_{\tau\in\mathbb{Z}}$ définie par $c[\tau] = E(X[\tau]\, X[t+\tau])$ mesure la covariance entre le signal à l'instant t et le signal à l'instant $t + \tau$.
La densité spectrale (ou spectre, PSD, power spectral density) de puissance S en est la transformée de Fourier, c'est une décomposition fréquentielle du signal.
Pour un signal indépendant, la fonction d'autocorrélation est une impulsion en 0, la PSD est constante.

a) Simuler et visualiser un signal indépendant gaussien centré de 1024 points, de variance 1.

b) Noter les valeurs de la moyenne et de la variance estimées. Estimer (fonction `xcorr`, option `'biased'`) et visualiser la fonction d'autocorrélation $c[\tau]$ pour $\tau \in \{-200, \ldots, 200\}$. Noter la valeur du maximum.

c) Calculer la PSD exacte et la comparer à ses estimations par la méthode du périodogramme (avec un pas fréquentiel de `1/Nfft`, où `Nfft=1024`), calculées :
- avec une fenêtre rectangulaire (`boxcar`) de 1024 points :
  `[S,F] = pwelch (signal, boxcar(1024), 0, Nfft, Fe, 'twosided'); plot(F,S)`
- avec des fenêtres rectangulaires de 64 points.

---

1. Pour simuler N réalisations indépendantes d'une variable centrée réduite :
Gauss : `x = randn(N,1);`
uniforme : `x = 2*sqrt(3)*(rand(N,1)-0.5);`
mélange : `m=0.95; x=randn(N,1)*sqrt(1-m*m)+m; k=find(rand(N,1)>0.5); x(k) = x(k)-2*m;`
2. Pour tracer un histogramme de n bâtons du vecteur x : `[nx,bx] = hist(x,n); nx = nx/(bx(2)-bx(1))/sum(nx); bar(bx,nx,1)`

4 Modélisation AR

Un modèle auto-régressif d'ordre p (AR(p)) d'un signal consiste à le considérer comme la sortie d'un système de fonction de transfert $\frac{1}{1+a_1 z^{-1}+\ldots+a_\mathrm{p} z^{-p}}$ excité par un signal indépendant de variance σ^2.

Pour une modélisation AR(1), la fonction d'autocorrélation vaut $c[\tau] = \frac{\sigma^2 (-a)^{|\tau|}}{1-a^2}$.
Si le vecteur Matlab `A` contient les coefficients du modèle ($A = [1, a_1, a_2, \ldots, a_\mathrm{p}]$), et la variance du bruit d'entrée vaut σ^2, on peut calculer la densité spectrale théorique à l'aide de la fonction `fft` ; soit `nfft` le nombre de fréquences par période de la densité spectrale, `F` le vecteur des fréquences, S le vecteur de la densité spectrale :
`S=sigma2./(abs(fft(A,nfft))).^2; F=[0:nfft-1]/nfft; plot(F,S)`

a) Pour $a = -0,8$, puis $a = -0,99$, simuler et visualiser un signal AR(1) de 1024 échantillons, l'excitation est gaussienne, de variance telle que la variance du signal simulé soit 1 (`randn, filter`).
b) Estimer moyenne et variance de ces 2 signaux.
c) Calculer et visualiser leurs fonctions d'autocorrélation exacte $c_{\mathrm{th}}[\tau]$ et estimée $c_{\mathrm{es}}[\tau]$ pour $\tau \in \{-100, \ldots, 100\}$.
d) Visualiser leur PSD exacte.
e) Visualiser leur PSD estimée par le périodogramme (longueur FFT = longueur fenêtre = 128).
f) Visualiser la PSD analytique à partir du modèle AR d'ordre 1 identifié par la méthode de covariance (en notant la valeur du paramètre estimé ainsi que la variance de la pseudo innovation : `[A,sigma2] = arcov(signal,1)`.

5 Illustration du théorème de la limite centrale

On considère M variables aléatoires indépendantes $(X_i)_{1\leqslant i\leqslant M}$ suivant une loi uniforme centrée de variance ν_x. Soit Z la variable aléatoire définie par :

$$Z = \frac{1}{M}\sum_{i=1}^{M} X_i$$

a) Calculer la variance théorique ν_z de Z.
b) On prend $\nu_x = 1$. Pour $M = 2$ et $M = 6$, tracer l'histogramme normalisé obtenu à partir de $N = 4000$ réalisations de Z ; le comparer à la densité de probabilité de la loi gaussienne centrée de variance ν_z ; estimer ν_z à partir de ces N échantillons.

6 Filtrage de Kalman

Un moteur à courant continu est commandé par la tension d'induit $u(t)$. La position angulaire du rotor $\theta(t)$ est mesurée par un codeur incrémental à $L = 512$ lignes par tour, fournissant une mesure $y(t)$ de $\theta(t)$. $\Omega(t)$ est la vitesse de rotation ($\Omega(t) = \dot{\theta}(t)$). En vue d'asservir la vitesse, on cherche à estimer en ligne $\theta(t)$ et $\Omega(t)$, connaissant $u(t)$ et $y(t)$.

6.1 Synthèse de l'entrée du système

La tension d'alimentation $u(t)$ est un créneau centré de période $\Delta = 100$ ms, d'amplitude crête à crête $A = 0{,}1$ V. On échantillonne ce signal à la période $T_\mathrm{e} = 1$ ms. Programmer une fonction `MatLab` permettant de synthétiser cette entrée échantillonnée pendant une durée D, sous la forme `u = entree(D,A,Delta,Te)`, où `u` est un vecteur colonne contenant le signal échantillonné (`square`).

6.2 Modélisation et simulation du système

En définissant le vecteur d'état $x(t) = \begin{bmatrix} \theta(t) \\ \Omega(t) \end{bmatrix}$, on obtient la représentation dans l'espace d'état suivante :

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{T} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{G}{T} \end{bmatrix} u \\ \theta = \begin{bmatrix} 1 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \end{bmatrix} u \end{cases}$$

Les signaux d'entrée-sortie du système sont échantillonnés à la période T_e. On suppose que l'entrée $u(t)$ est constante entre deux instants d'échantillonnage. La représentation d'état ci-dessus peut donc être échantillonnée sans approximation par la méthode de l'invariance indicielle [3] (bloqueur d'ordre 0, fonction `MatLab c2dm`). Pour tout n, et toute fonction f, on notera $f_n = f(n\,T_\mathrm{e})$.
La mesure y_n fournie par le codeur incrémental quantifie la position angulaire réelle θ_n. [4]

a) Programmer le simulateur (c'est-à-dire une fonction de la forme : `[y,x] = simule(u,G,T,Te,L,x1)`, où `y` est une matrice colonne de même dimension que `u` contenant l'évolution de la mesure y_n, `x` est une matrice à 2 colonnes contenant l'évolution du vecteur d'état, et `x1` est l'initialisation du vecteur d'état).

b) Tester le simulateur en prenant $G = 50$ rad.s^{-1}.V^{-1} et $T = 20$ ms.

6.3 Estimation par filtrage de Kalman

On appelle w_n le bruit de mesure supposé blanc du codeur incrémental (dû à l'erreur de quantification), r sa variance ($y_n = \theta_n + w_n$). Pour tenir compte des erreurs de modélisation éventuelles, on suppose que l'entrée réelle du système est $u_n + v_n$, où v_n est un bruit blanc de variance q, indépendant de w_n.
En vue de réaliser l'asservissement en vitesse du rotor, on cherche à estimer au mieux X_n à l'aide d'un filtre de Kalman. On sait que le moteur est non alimenté et à l'arrêt avant le premier instant d'échantillonnage, mais on n'a aucune information sur la position angulaire initiale.

a) Écrire la représentation d'état stochastique du modèle du système d'entrée u_n et de sortie y_n.

b) Proposer une valeur de r fondée sur des considérations physiques concernant le codeur incrémental.

c) Proposer une initialisation de la prédiction de l'état $\hat{X}_{1/0}$ et de la variance de l'erreur de prédiction $P_{1/0}$, d'après les connaissances *a priori.*

d) Programmer les filtres de Kalman optimal (dont le gain dépend du temps), et stationnaire (à gain constant, `dlqe`), c'est-à-dire deux fonctions du type `xe = kal(y,u,G,T,Te,L,x1_0,P1_0,q)`, où `xe` est une matrice à 2 colonnes contenant l'évolution de l'estimation du vecteur d'état, et où `x1_0` et `P1_0` désignent respectivement l'initialisation de la prédiction de l'état $\hat{X}_{1/0}$ et de la variance de l'erreur de prédiction $P_{1/0}$.

3. Étant donné un système à temps continu régi par la représentation d'état suivante :

$$\begin{cases} \dot{x}(t) = A\,x(t) + B\,u(t) \\ y(t) = C\,x(t) + D\,u(t) \end{cases}$$

l'échantillonnage par la méthode de l'invariance indicielle à la période T_e donne :

$$\begin{cases} x_{n+1} = \tilde{A}\,x_n + \tilde{B}\,u_n \\ y_n = C\,x_n + D\,u_n \end{cases} \text{avec} \begin{cases} \tilde{A} = e^{A\,T_\mathrm{e}} \\ \tilde{B} = \int_0^{T_\mathrm{e}} e^{A\,\tau}\,B\,d\tau \end{cases}$$

4. Un codeur incrémental à L lignes par tour a une précision de $2\,\pi/L$ rad. Sous `MatLab` : `y = round(teta*L/2/pi)*2*pi/L`

6.4 Simulations

Comparer sur l'estimation de la position et de la vitesse les performances des deux filtres pour différentes valeurs de q (se placer dans le cas $\hat{\theta}_{1/0} - \theta_1 = \pm 0,05$), dans les cas suivants :

— le système est parfaitement modélisé : $\begin{cases} G_{\text{exact}} = G_{\text{filtre}} = 50 \text{ rad.s}^{-1}.\text{V}^{-1} \\ T_{\text{exact}} = T_{\text{filtre}} = 20 \text{ ms} \end{cases}$

— le modèle du système est imprécis : $\begin{cases} G_{\text{exact}} = G_{\text{filtre}} = 50 \text{ rad.s}^{-1}.\text{V}^{-1} \\ T_{\text{exact}} = 20 \text{ ms} \\ T_{\text{filtre}} = 25 \text{ ms} \end{cases}$

L'indice « exact » indique la valeur utilisée pour la simulation, l'indice « filtre » la valeur utilisée pour l'estimation.