Rapport des TPs d'Estimation Stochastique

James LE POIDEVIN, Clément SIGNOL | 06/12/2019

James LE POIDEVIN
Clément SIGNOL

TP n°1 : Introductions aux variables aléatoires

Présentation du TP

L'objectif de cette manipulation de Travaux Pratiques est d'illustrer sous MATLAB les notions théoriques vues en cours sur les variables aléatoires.

1. Variables aléatoires monodimensionnelles

1°) L'histogramme est semblable à une gaussienne mais si N augmente on retrouve la gaussienne lisse, plus on augmente le nombre de réalisations, plus la courbes est précise.

```
N = 1000;
%A modifier, Plus la valeur de N est grand plus l'histogramme
%resemble a une gaussien
X = randn ([N, 1]);
[counts , centers] = hist(X ,100);
bar(centers , counts/N/( centers (2)-centers (1)));
```

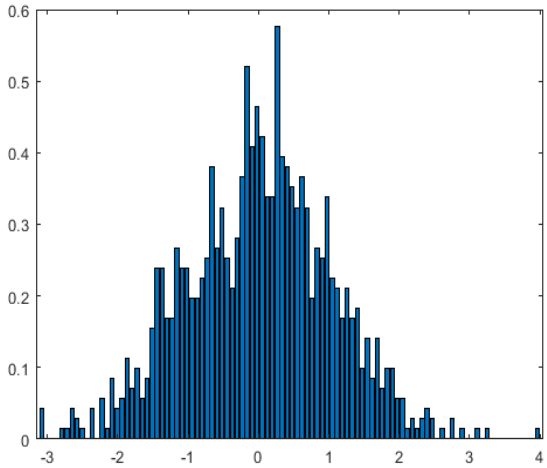


Figure 2 : Histogramme pour 1000 réalisations

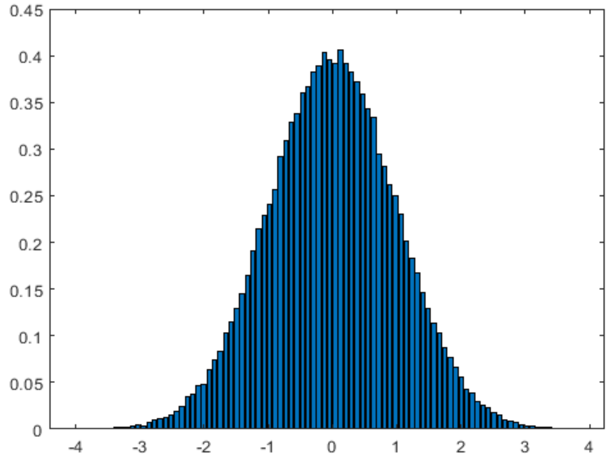


Figure 1 : Histogramme pour 100 000 réalisations

2°) Afin de vérifier l'adéquation avec la loi théorique Gaussienne, nous avons représenté sa densité de probabilité théorique en rouge. Nous constatons que la loi de densité suit la loi normale.

```
%Question2
sigma = 1; %variance (= ecart - type)
meanx = 0;
rangex = (min(X):0.1: max(X));
px = (1/( sqrt (2*pi)*sigma))*exp(-(rangex - meanx).*( rangex - meanx)/(2* sigma*sigma));
hold on
plot(rangex , px , 'r' )
hold off
```

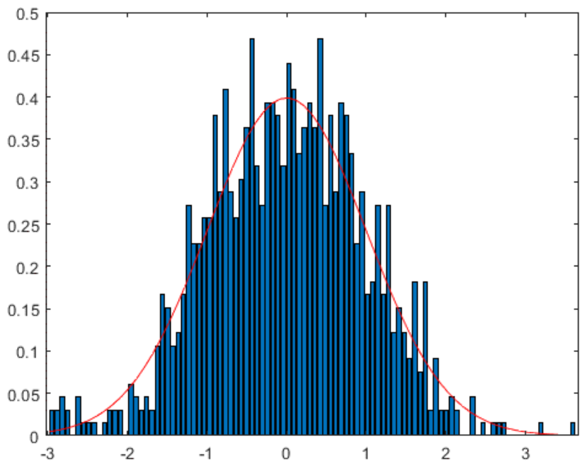


Figure 4 : Histogramme pour 1 000 réalisations avec densité de probabilité théorique

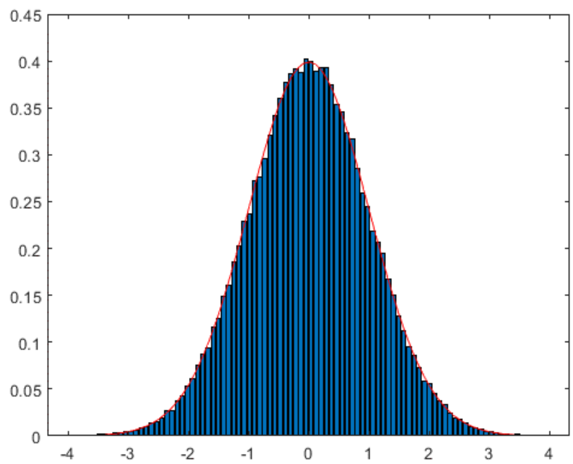


Figure 3 : Histogramme pour 100 000 réalisations avec densité de probabilité théorique

La courbe rouge épouse bien l'histogramme de la figure 4, un peu moins pour la figure 3.

3°) On trouve Moyenne=10 et Variance=2 => Y=moyenne+sqrt(variance)*x

```
%question3
Y = 10 + sqrt (2)*X;
mean_Y = mean2(Y)
variance = var(Y)
sigma = sqrt(variance)
gy = mean_Y + sigma*randn ([N, 1]);
```

```
mean_Y =

   10.0021

variance =

    1.9592

sigma =

    1.3997
```

mean_Y toujours autour de 10 et variance toujours autour de 10
Variance toujours 2 et sigma 1.4

2. Variables aléatoires bi-dimensionnelles

4°) Non, on ne peut retrouver ces moments à partir de l'histogramme, on ne pourra en donner que des estimations car on n'a un nombre d'expérience fini.

```
%Question4
figure
[counts , centers] = hist(gy ,100);
bar(centers , counts/N/( centers (2)-centers (1)));
rangey = (min(Y):0.1: max(Y));
py = (1/( sqrt (2*pi)*sigma))*exp(-(rangey - mean_Y).*( rangey -mean_Y)/(2* sigma*sigma));
hold on
plot(rangey , py , 'r' )
title('Question 4 : Histogramme de Y')
```

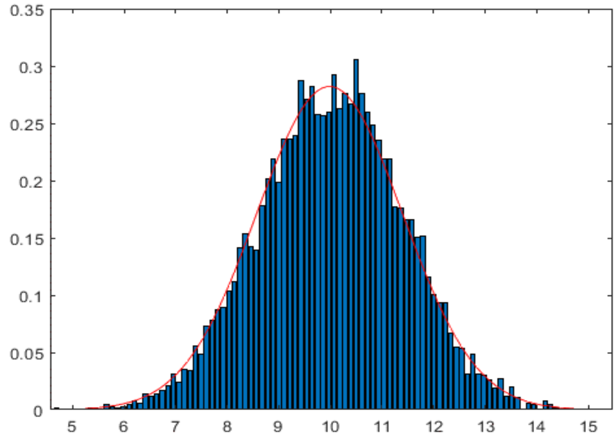


Figure 5 : Histogramme et densité de probabilité théorique pour Y

5°)

```
%Question5
N = 100000;
x11 = randn ([1, N]);
x12 = randn ([1, N]);
X1 = [x11; x12];
```

6°) C'est centré autour de (0,0). De plus, plus on s'écarte et moins il y a de points.

```
%Question6
plot(X1(1,:),X1(2,:), '.b' );
```

On plot le les deux Variable aléatoire en bleu avec des points.

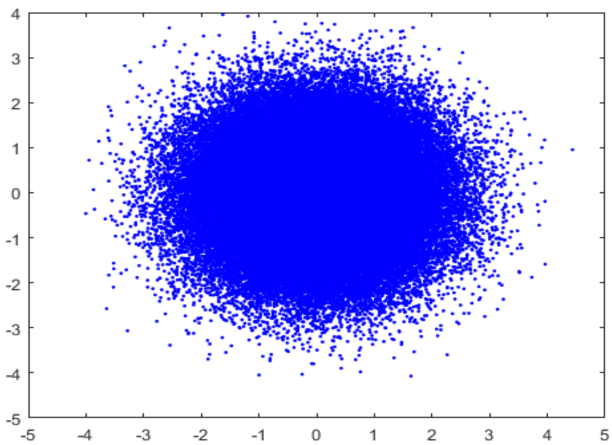


Figure 6 : Nuage de points de x11 et x12

7°) Nous constatons que notre nuage de point est centré en (10,2) soit la moyenne de x21 et x22. On voit aussi que la variance a un impact sur l'étalement des points.

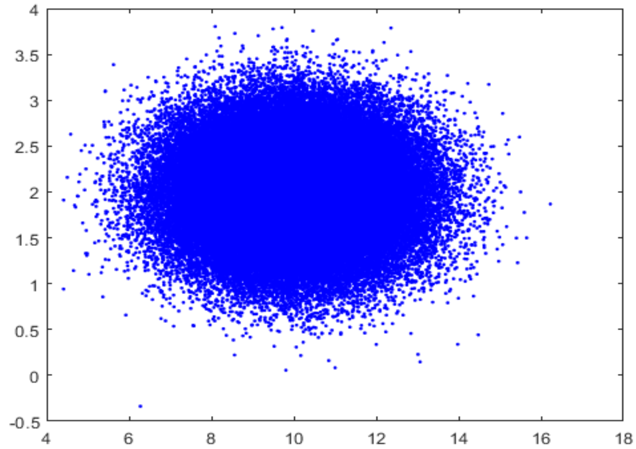


Figure 7 : Nuage de points X21 X22

```
%Question7 %figure3
var21 = 2;
var22 = 0.2;
moy21 = 10;
moy22 = 2;

x21 = 10 + sqrt(var21)*randn ([1, N]);
x22 = 2 + sqrt(var22)*randn ([1, N]);
X2 = [x21; x22];
figure;
plot(X2(1,:),X2(2,:), '.b' );
```

8°) Quand on passe de 1 à 5, Arnaud te gueule dessus parce que tu ne désamorces pas la bombe, la moyenne est à peu près identique mais l'écart-type est beaucoup plus grand pour 5.

```
%Question8
x31 = x11;
x321 = x12 + 1*x11;
x325 = x12 + 5*x11;
X31 = [x31; x321 ];
X35 = [x31; x325 ];
```

X321 correspond à la Variable aléatoire x32 avec a qui prend la valeur 1.

X325 correspond à la Variable aléatoire x32 avec a qui prend la valeur 5.

9°) Nous avons maintenant : x31 = x11, x32 = x12 + a*x11, X3 = [x31 ; x32]

Quand on passe « a » de 1 à 5, la moyenne est à peu près identique mais l'écart-type est beaucoup plus grand pour 5.

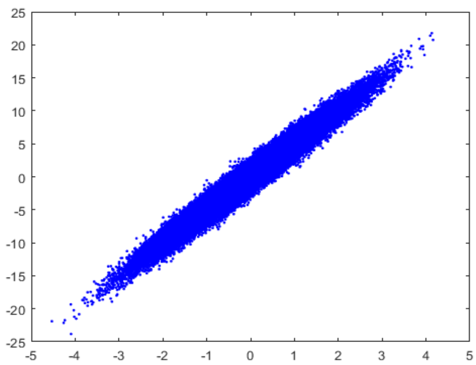


Figure 9 : a=5

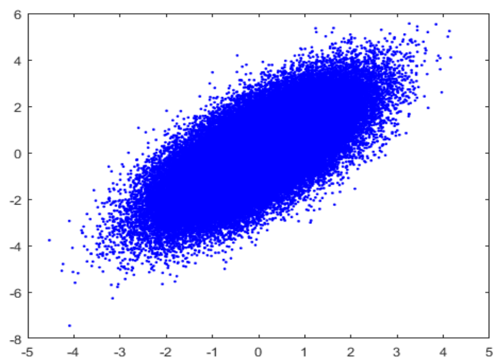


Figure 8 : a=1

10°) m3,1 m3,2 = m2,1 +am1,1

3. Somme de variables aléatoires

11°)

```
%question11
N = 10000;
K3 = 3;
X3 = randn ([K3 , N]);
Y3 = sum(X3);

K6 = 6;
X6 = randn ([K6 , N]);
Y6 = sum(X6);
```

12°)

```
%Question12
[counts , centers] = hist(Y3 ,100);
bar(centers , counts/N/( centers (2)-centers (1)));
```

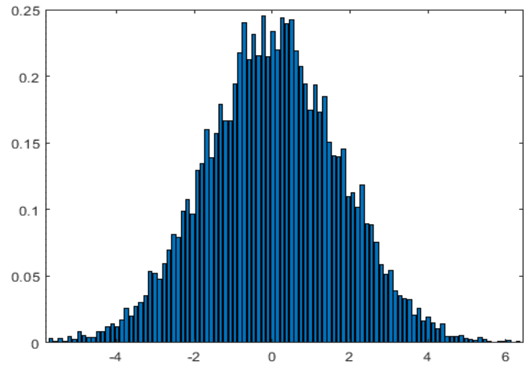


Figure 10 : Histogramme des réalisations de Y (K=3)

13°)

```
%Question 13
sigma = sqrt(((1/N)*sum(Y3*Y3')))
meanx3 = 0;
rangey3 = (min(Y3):0.1: max(Y3));
py3 = (1/( sqrt (2*pi)*sigma))*exp(-(rangey3 - meanx3).*(rangey3 - meanx3)/(2* sigma*sigma));
hold on;
plot(rangey3 , py3 , 'r' )
hold off
```

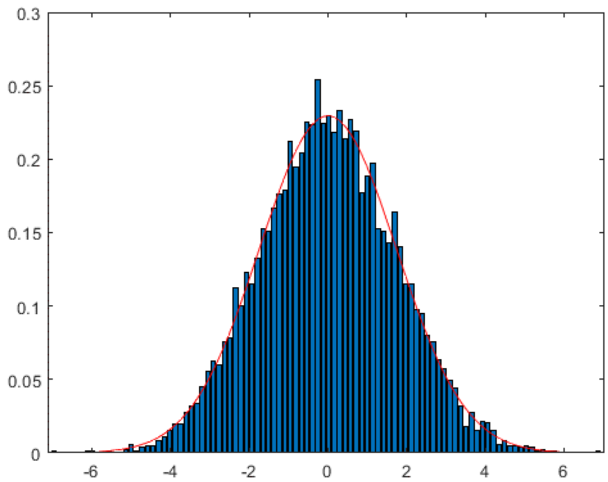


Figure 11 : Histogramme de réalisation de Y (K=3) qui suit une loi Gaussienne de moyenne 0 et de varience unité

14°)

```
X_chi2 =X3 .* X3;
z= sum ( X_chi2 );
[ counts , centers ] = hist (z ,100) ;
bar ( centers , counts /N/( centers (2) -centers (1) ));
```

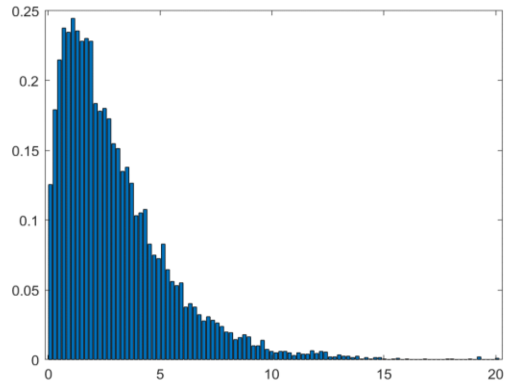


Figure 12 : Histogramme somme des K variables aléatoires gaussiennes Xk^2 + densité de probabilité avec K = 3

15°)

```
hold on
range_z =( min(z) :0.1: max(z));
pz =( range_z .^(( K3 /2) -1) .* exp(- range_z /2) ) /(2^( K3 /2) * gamma (K3 /2));
plot ( range_z ,pz ,'r');
```

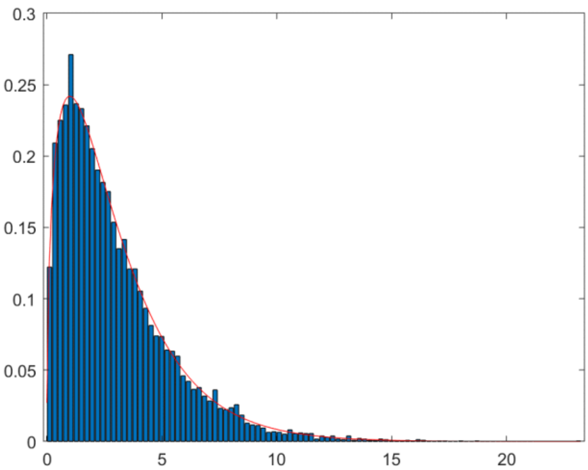


Figure 14 : K = 3

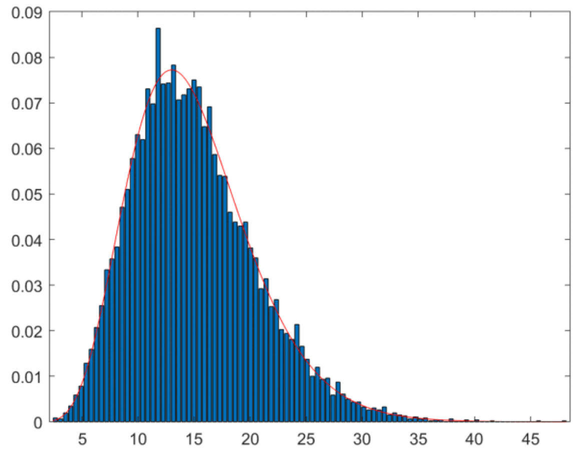


Figure 13 : K=15

16°)

```
%Question 16
var_z = var (z)
mean_z = mean (z)
```

```
var_z =

    6.0242


mean_z =

    3.0071
```

17)

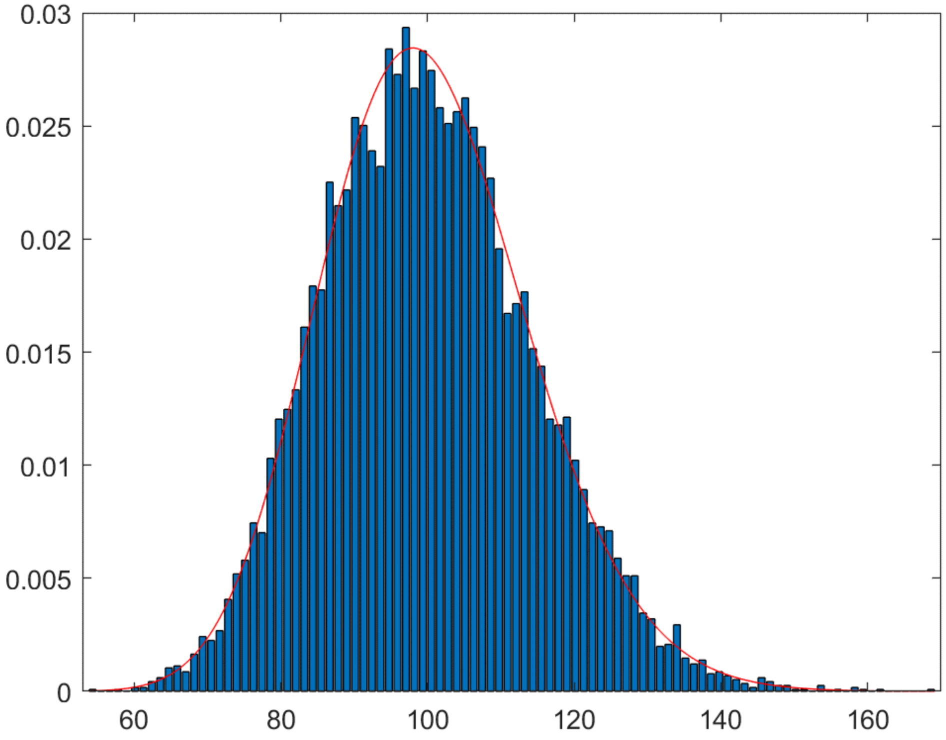


Figure 15 : Histogramme somme des K variables aléatoires gaussiennes Xk^2 + densité de probabilité avec K = 100

James LE POIDEVIN
Clément SIGNOL

TP n°2 : Estimation par maximum de vraisemblance moindres carrés

Présentation du TP

L'objectif de cette manipulation de Travaux Pratiques est d'illustrer sous MATLAB la synthèse et l'évaluation d'un estimateur par maximum de vraisemblance sur un modèle linéaire Gaussien.

1. Modélisation

1°) Dans de cas où notre entrée est représentée par une impulsion discrète tel que $x[n]=\delta[n]$ où $\delta[0]=1$ *et* $\delta[n]=0$ pour tout n, notre relation devient : $Y[n]=h[n]+B[n]$

Avec $B[n]$ le bruit qui suit une loi Gaussienne $N(0, \rho^2)$ tel que $\rho^2 = 0.4$. Donc Y suit une loi Gaussienne tel que $Y \sim N(h, \rho^2)$.

Ainsi nous pouvons déduire un estimateur de la réponse impulsionnelle avec $\hat{Y} \sim N(h,\rho^2)$.

Calcul du biais de $\hat{Y}$: $b\hat{Y}(h)=E[\hat{Y}-h]=E[\hat{Y}]-h=E[h-B]-h=h-0-h=0$

Calcul de la variance :

$Var(\hat{Y})=E[(\hat{h}-E[\hat{h}])^2]=E[(Y-E[Y])^2]=E[(h-B-h)^2]=E[B^2]=Var(B)=0.4$

Donc notre fonction estimateur $\hat{Y}$ aura comme moyenne 0 car non biaisé et une variance égale à 0.4.

2°) On a le modèle suivant : $y_m[n] = \sum_{k=0}^{M-1} \mathrm{h}[\mathrm{k}]x[n-k]$

On constate déjà qu'il s'agit d'une forme linéaire.

$$y_m[n] = (h[0] \ldots \ h[M-1]) \begin{pmatrix} x[n] \\ \ldots \\ x[n-(M-1)] \end{pmatrix}$$

Donc on peut l'écrire sous la forme y = constante * paramètres inconnus.

Donc y_m est bien une forme linéaire avec des paramètres inconnus.

3°) On a la relation $Y = Ah + B$ avec $x[n]$ inconnu.

4°) Comme nous sommes dans le cas d'un modèle d'observation linéaire avec perturbations additives Gaussiennes alors le critère que doit maximiser l'estimé $\hat{h}$ du maximum de vraisemblance de h est :

$\hat{h}_{mle} = (A^t R^{-1} A)^{-1} A^t R^{-1} Y$ Avec R = ρ^2 = 0.4

D'après les résultats théoriques de notre cours nous déduisons que le biais de $\hat{h}mle$=0.

Que sa matrice de covariance est $Cov\ \hat{h}_{mle} = (A^t R^{-1} A)^{-1}$.

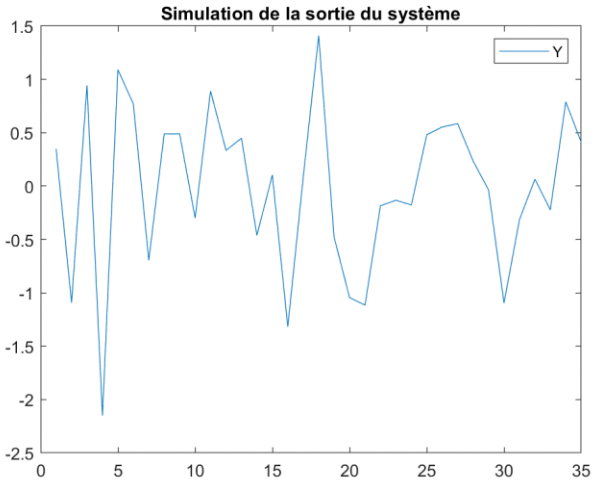
Et que l'erreur d'estimation est $EQM\ \hat{h}_{mle} = trace(A^t R^{-1} A)^{-1}$.

2. Implémentation et analyse sous MATLAB

5°)

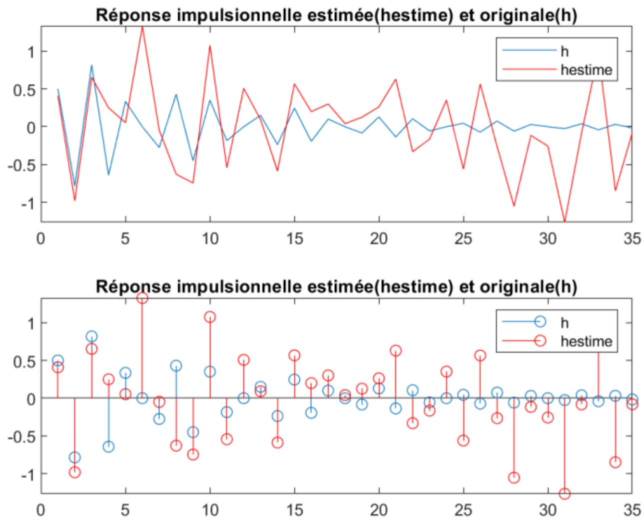
```
%Question5
moyB=0;
varB=0.4;

%Dirac
dir = zeros( length(h), 1);
dir(1,1) = 1;
Y=syst(dir,h,varB);
figure;
plot(Y);
title('Simulation de la sortie du système')
legend('Y');
hestime=Y(1 :length(h));
```



6°) En bleu on peut voir la réponse impulsionnelle original et en rouge l'estimé. On peut dire que cette estimation n'est pas très satisfaisante vu le la figure ci-contre :

```
%Question 6
figure;
subplot 211;
plot(h);
hold on;
plot(hestime,'r');
title('Réponse impulsionnelle estimée(hestime) et originale(h)')
legend('h','hestime');
subplot 212;
stem(h);
hold on;
stem(hestime,'r');
title('Réponse impulsionnelle estimée(hestime) et originale(h)')
legend('h','hestime');
%Distance quadratique dQN=
dQN=(1/length(h))*norm(hestime-h)^2
```



Distance Quadratique = 0.3317

```
%Question 7 : Réalisation impulsionnelle pour 1000 réalisations
Nr = 1000;
moyE = 0;
varE = 0;
for i = 1 : Nr
Y = syst(dir,h,varB);
hestime=Y(1 :length(h));
moyE = moyE + mean(hestime);
varE = varE + var(hestime);
end
moyE=moyE/Nr
varE=varE/Nr
```

7°) **Moy E** = moyenne empirique=0.0031

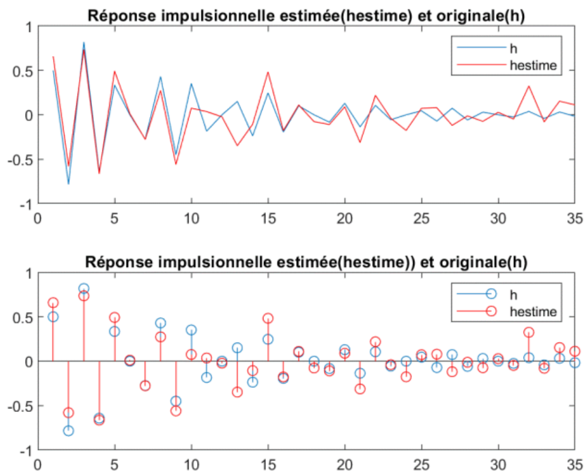
Var E = variance empirique=0.4926

8°)

```
% Question 8
Y=syst(x,h,varB);
A=toeplitz(x,zeros(1,length(h)));
hmle=inv(A'*inv(varB)*A)*A'*inv(varB)*Y;
```

9°)

```
%Question 9
figure;
subplot 211;
plot(h);
hold on;
plot(hmle,'r');
title('Réponse impulsionnelle estimée(hestime) et originale(h)')
legend('h','hestime');
subplot 212;
stem(h);
hold on;
stem(hmle,'r');
title('Réponse impulsionnelle estimée(hestime)) et originale(h)')
legend('h','hestime');
%Distance quadratique dQN=
dQN1=(1/length(h))*norm(hmle-h)^2
```



Distance Quadratique = 0.0244

10°)

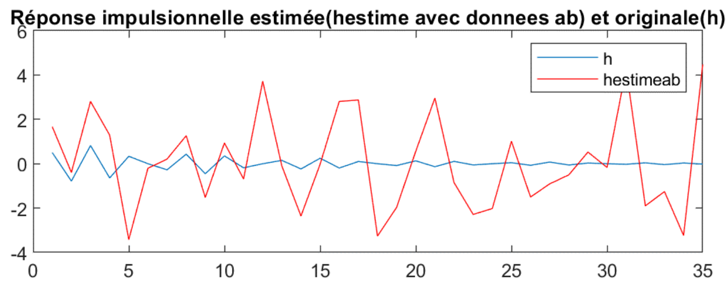
```
% Question 10 : Réalisation impulsionnelle pour 1000 réalisations
Nr = 1000;
moyE = 0;
varE = 0;
for i = 1 : Nr
Y = syst(x,h,varB);
A=toeplitz(x,zeros(1,length(h)));
hmle=inv(A'*inv(varB)*A)*A'*inv(varB)*Y;
moyE = moyE + mean(hmle);
varE = varE + var(hmle);
end
moyE1=moyE/Nr
varE1=varE/Nr
```

Moy E1 = moyenne empirique= 0.0040

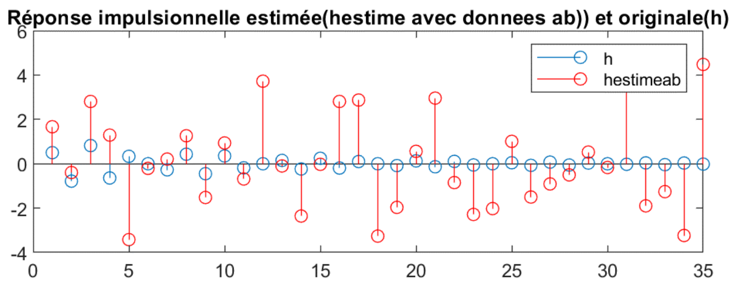
Var E1 = variance empirique = 0.1170

11°)

```
%% Question 11 : Données aberrantes
Y=syst(x,h,varB);
Yab=Y;
for i=1:5
Yab(randi(length(Yab)))=50;
end
A=toeplitz(x,zeros(1,length(h)));
hab=inv(A'*inv(varB)*A)*A'*inv(varB)*Yab;
figure;
subplot 211;
plot(h);
hold on;
plot(hab,'r');
title('Réponse impulsionnelle estimée(hestime avec donnees ab) et originale(h)')
legend('h','hestimeab');
subplot 212;
stem(h);
hold on;
stem(hab,'r');
title('Réponse impulsionnelle estimée(hestime avec donnees ab)) et originale(h)')
legend('h','hestimeab');
%Distance quadratique dQN=
dQN2=(1/length(h))*norm(hab-h)^2
```



dQN2=Distance Quadratique=4.4092



12°)

```
%% Question 12
Y = syst(x,h,varB);
A=toeplitz(x,zeros(1,length(h)));
hmle=inv(A'*inv(varB)*A)*A'*inv(varB)*Y;
convN=conv(hmle,x);
convA=conv(hab,x);
figure
subplot 211;
stem(Y-convN(1:length(x)));
title('Erreur sans les données aberrantes');
subplot 212;
stem(Yab-convA(1:length(x)));
title('Erreur avec les données aberrantes');
```