

Gestion de flux dans le réseau

TD n ° 4

Modélisation mathématique

Q4

Sibylle Roux

Juliette Arazo

Nicolas Le Gallo

Tanguy Thomas

12 novembre 2017

Table des matières

I	Etude statistique des temps interarrivés	4
1	Etude statistique des temps interarrivés pour tous les serveurs	4
1.1	Indicateurs de position et de dispersion	4
1.2	Fonction de répartition	4
1.3	Histogramme	5
1.3.1	Histogramme avec classes isoamplitudes	5
1.3.2	Histogramme avec classes isofréquences	5
II	Etude statistique des temps de service	6
2	Indicateurs de position et de dispersion	6
3	Fonctions de répartition	7
3.1	Serveur 1	7
3.2	Serveur 2	8
3.3	Serveur 3	8
4	Histogrammes	9
4.1	Serveur 1	9
4.2	Serveur 2	10
4.3	Serveur 3	10
III	Ajustement graphique à des lois mathématiques	11
5	Estimation des paramètres	11
5.1	Loi normale	11
5.2	Loi uniforme	11
5.3	Loi exponentielle	11
6	Tous les serveurs	12
6.1	Superposition des fonctions de répartition	12
6.2	Superposition des fonctions de densité et de l'histogramme	12
7	Serveur 1	13
7.1	Superposition des fonctions de répartition	13
7.2	Superposition des fonctions de densité et de l'histogramme	13
8	Serveur 2	14
8.1	Superposition des fonctions de répartition	14
8.2	Superposition des fonctions de densité et de l'histogramme	14

9	Serveur 3	15
9.1	Superposition des fonctions de répartition	15
9.2	Superposition des fonctions de densité et de l'histogramme	15
A	Etude statistique des temps interarrivés	16
A.1	Indicateurs de position et de dispersion	16
A.2	Fonction de répartition	16
A.3	Histogramme	16
A.3.1	Histogramme avec classes isoamplitudes	16
A.3.2	Histogramme avec classes isofréquences	17
B	Etude statistique des temps de service	17
B.1	Indicateurs de position et de dispersion	17
B.1.1	Serveur 1	17
B.1.2	Serveur 2	18
B.1.3	Serveur 3	18
B.2	Fonctions de répartitions	19
B.2.1	Serveur 1	19
B.2.2	Serveur 2	19
B.2.3	Serveur 3	19
B.3	Histogrammes	20
B.3.1	Serveur 1	20
B.3.2	Serveur 2	20
B.3.3	Serveur 3	21
C	Ajustement graphique à des lois mathématiques	21
C.1	Tous les serveurs	21
C.1.1	Superposition des fonctions de répartitions	21
C.1.2	Superposition des fonctions de densité et de l'histogramme	22
C.2	Serveur 1	23
C.2.1	Superposition des fonctions de répartitions	23
C.2.2	Superposition des fonctions de densité et de l'histogramme	24
C.3	Serveur 2	25
C.3.1	Superposition des fonctions de répartitions	25
C.3.2	Superposition des fonctions de densité et de l'histogramme	26
C.4	Serveur 3	27
C.4.1	Superposition des fonctions de répartitions	27
C.4.2	Superposition des fonctions de densité et de l'histogramme	28

Première partie

Etude statistique des temps interarrivés

1 Etude statistique des temps interarrivés pour tous les serveurs

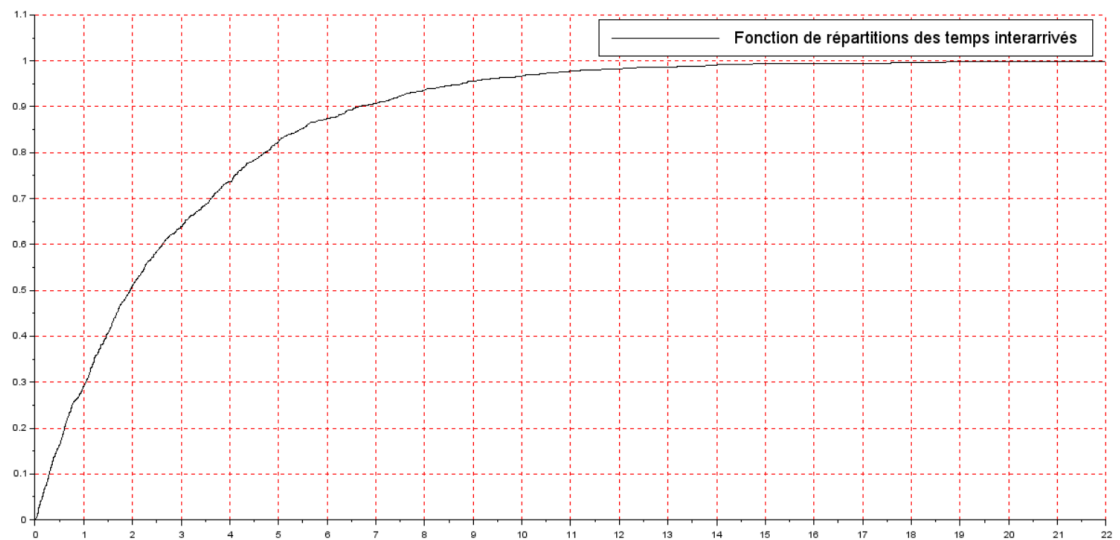
1.1 Indicateurs de position et de dispersion

Min	Max	Moyenne	Médiane
0.01	21.98	2.9	1.95

Variance	Ecart-type	Etendue
8.73	2.95	22

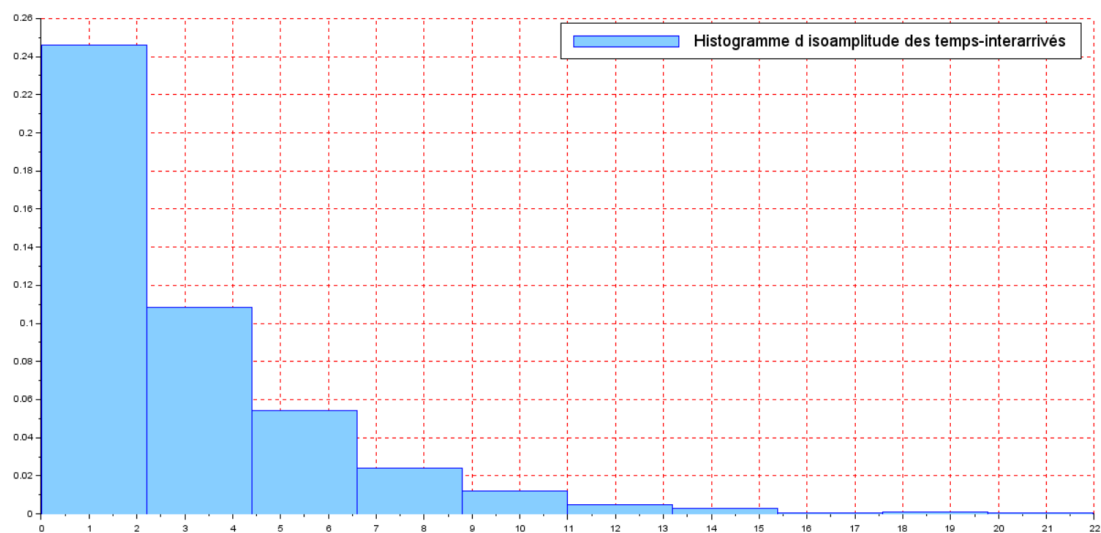
Q1	Q2	Interquartile
0.76	4.09	3.33

1.2 Fonction de répartition

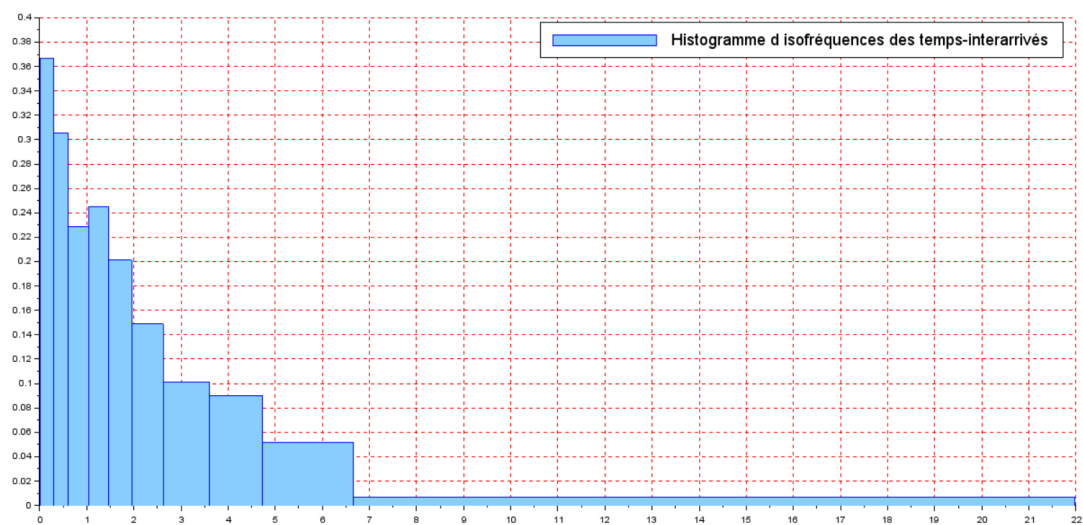


1.3 Histogramme

1.3.1 Histogramme avec classes isoamplitudes



1.3.2 Histogramme avec classes isofréquences



Deuxième partie

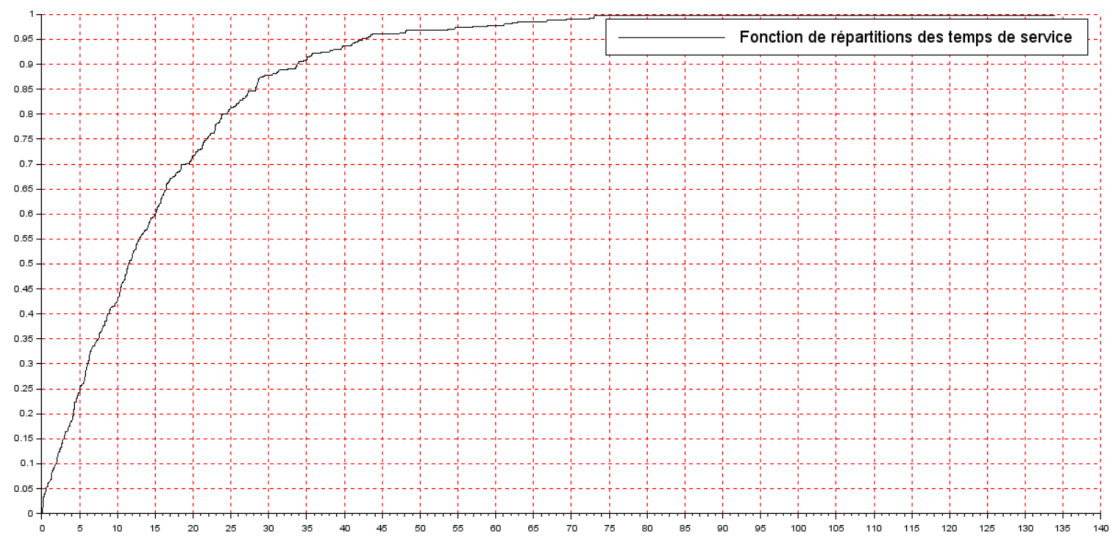
Etude statistique des temps de service

2 Indicateurs de position et de dispersion

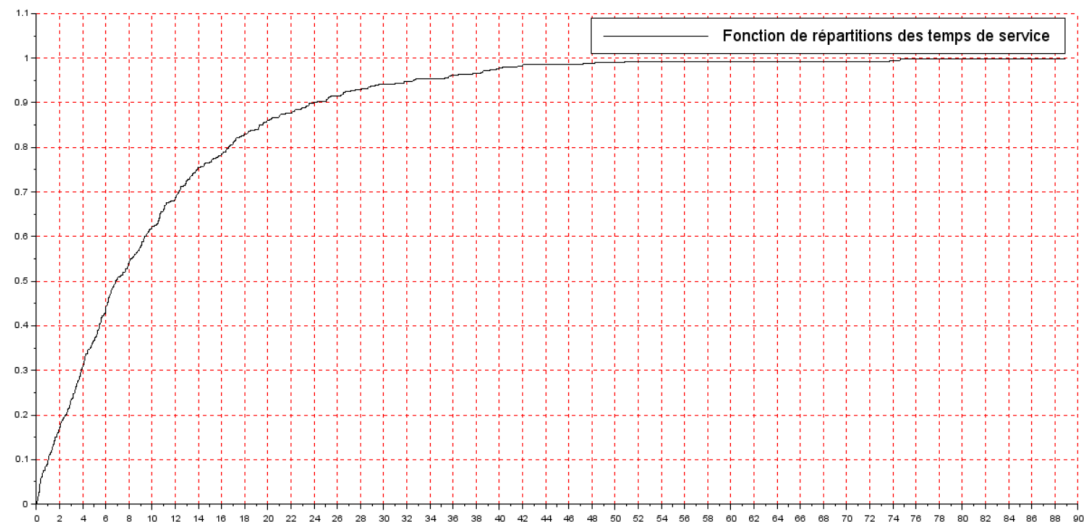
Indicateurs	Serveur 1	Serveur 2	Serveur 3
Minimum	0.01	0.04	0.01
Maximum	134	88.9	68.6
Etendue	134	88.8	68.6
Moyenne	15.5	10.6	6.27
Médiane	11.5	6.82	4.35
Q1	5.05	3.29	1.75
Q3	21.9	13.9	8.36
IQ	16.8	10.6	6.61
Ecart-Type	15	11.3	6.85
Variance	225	127	46.9

3 Fonctions de répartition

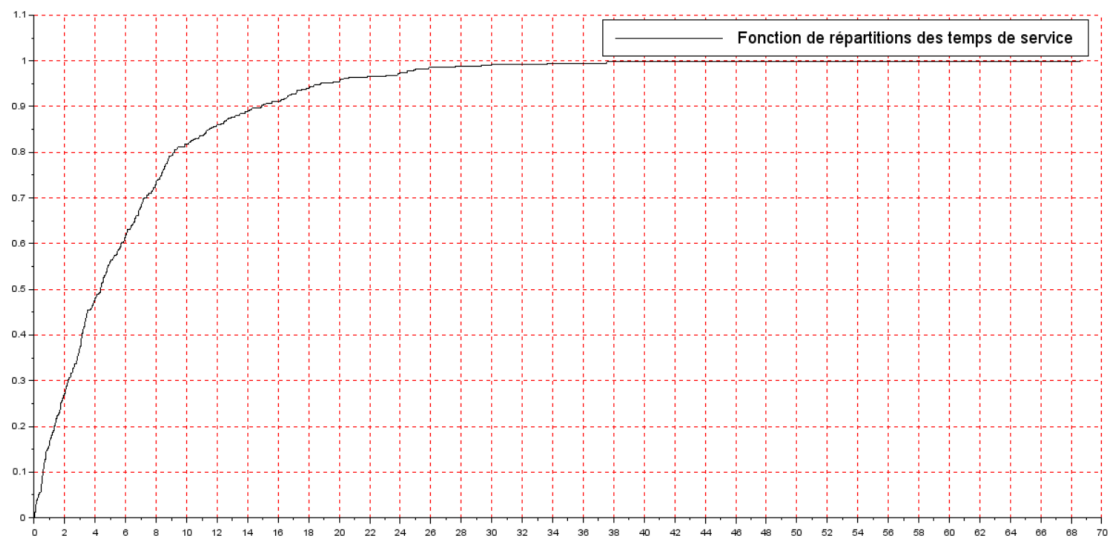
3.1 Serveur 1



3.2 Serveur 2

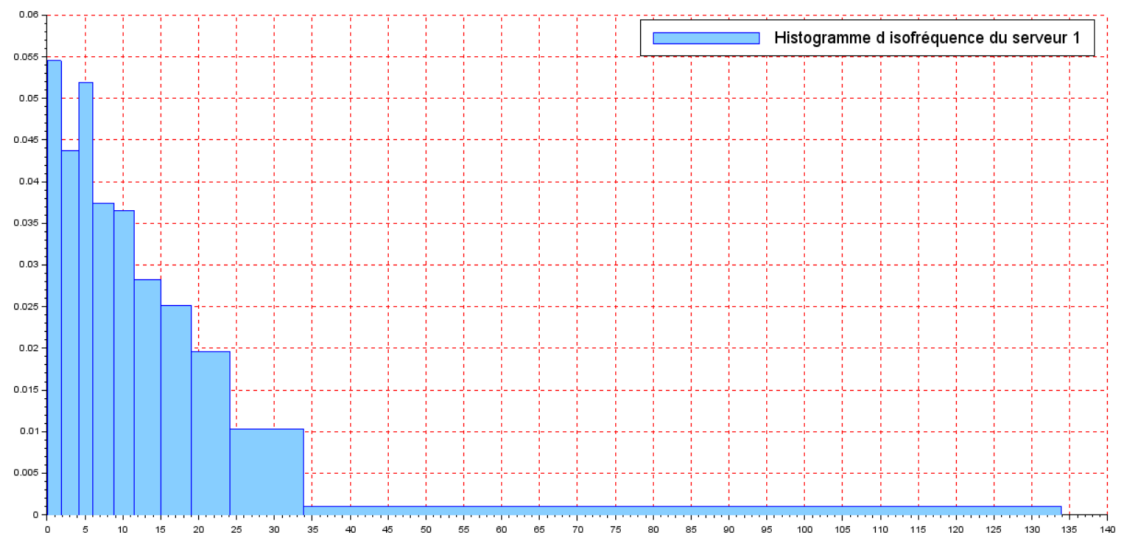


3.3 Serveur 3

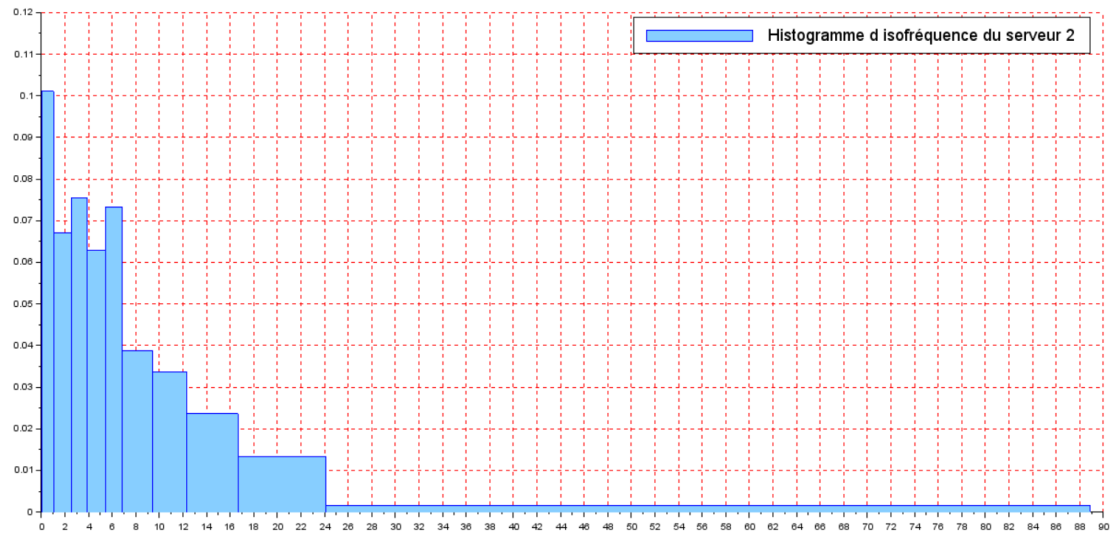


4 Histogrammes

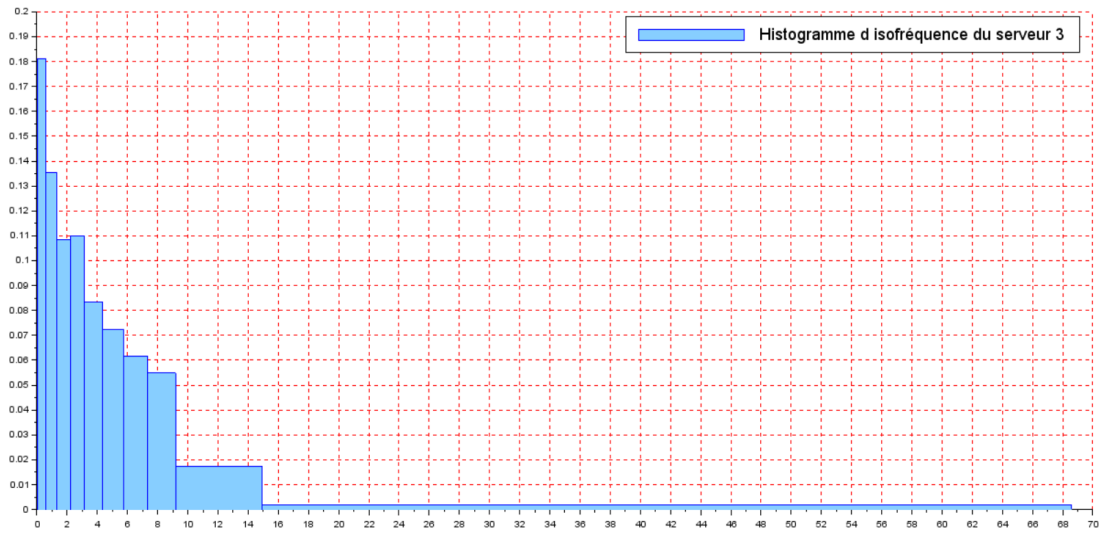
4.1 Serveur 1



4.2 Serveur 2



4.3 Serveur 3



Troisième partie

Ajustement graphique à des lois mathématiques

5 Estimation des paramètres

5.1 Loi normale

La loi normale $X \sim N(\mu, \sigma)$ s'exprime en fonction de l'espérance μ ainsi que de l'écart-type σ . On associe donc l'espérance à la moyenne des temps d'attentes/temps interarrivés.

5.2 Loi uniforme

La loi uniforme dépend de 2 paramètres : a et b qui correspondent à l'intervalle $[a, b]$ sur laquelle est définie la loi uniforme. On associe donc a au minimum et b au maximum des temps d'attentes/temps interarrivés.

5.3 Loi exponentielle

La loi exponentielle dépend d'un seul paramètre : λ . Or on sait que pour une loi exponentielle X :

$$E(X) = \frac{1}{\lambda} \quad (1)$$

$$V(X) = \frac{1}{\lambda^2} \quad (2)$$

Donc

$$\lambda = \frac{1}{E(X)} \quad (3)$$

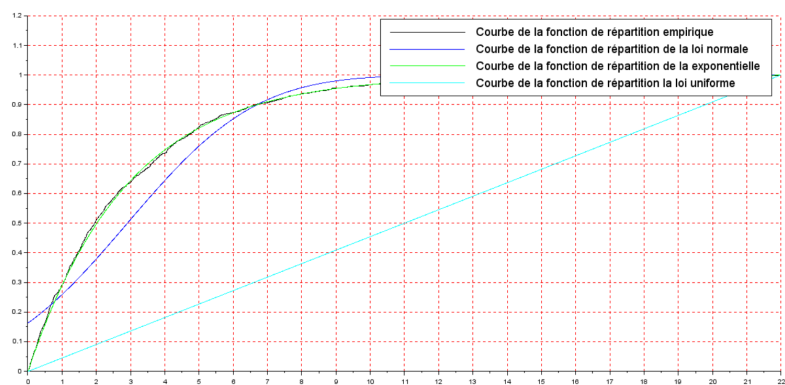
$$\lambda = \frac{1}{\sqrt{V(X)}} \quad (4)$$

$$= \frac{1}{\sigma} \quad (5)$$

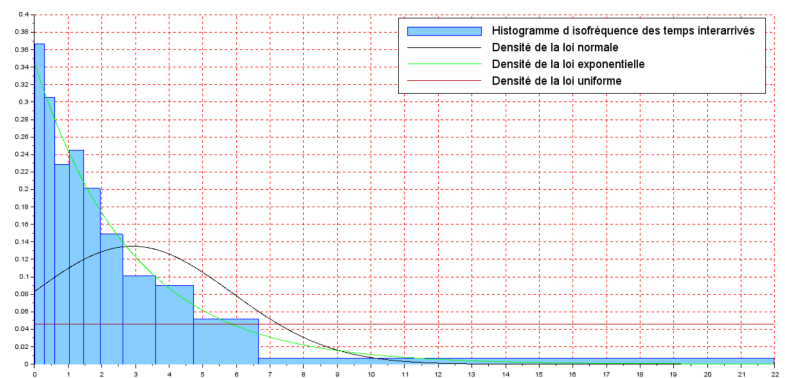
On en déduit déjà que pour qu'une la moyenne doit être égale à l'écart type pour que les temps de services soient les réalisations d'une loi exponentielle.

6 Tous les serveurs

6.1 Superposition des fonctions de répartition

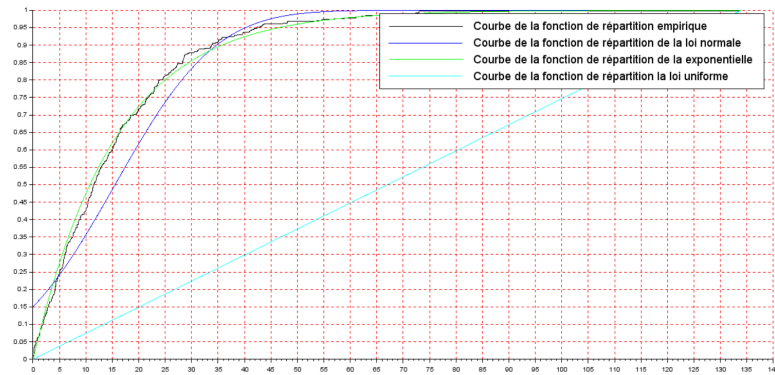


6.2 Superposition des fonctions de densité et de l'histogramme

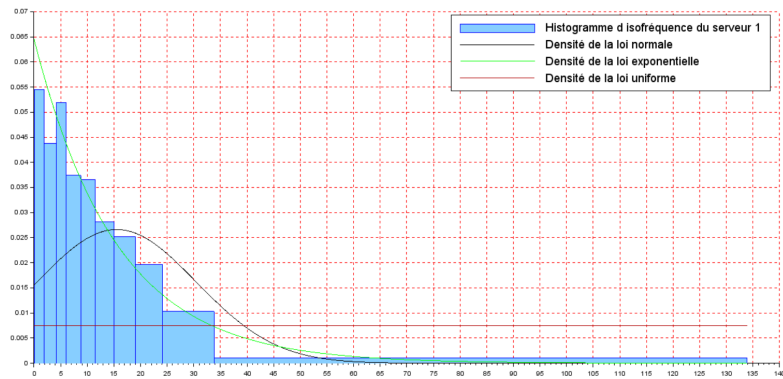


7 Serveur 1

7.1 Superposition des fonctions de répartition

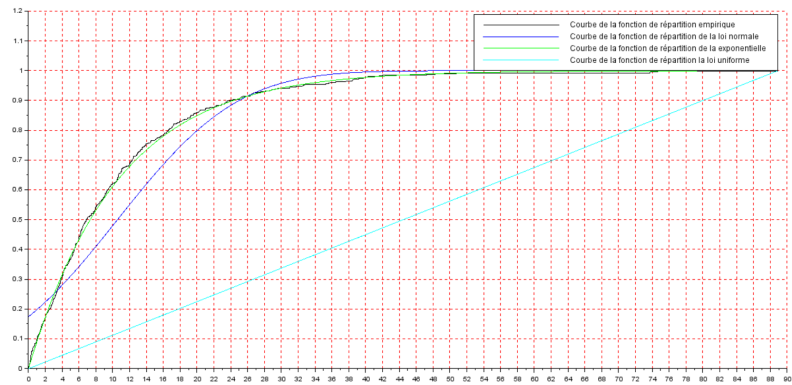


7.2 Superposition des fonctions de densité et de l'histogramme

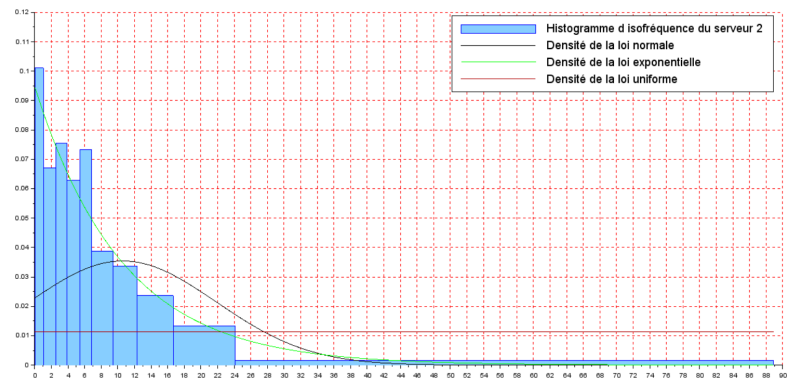


8 Serveur 2

8.1 Superposition des fonctions de répartition

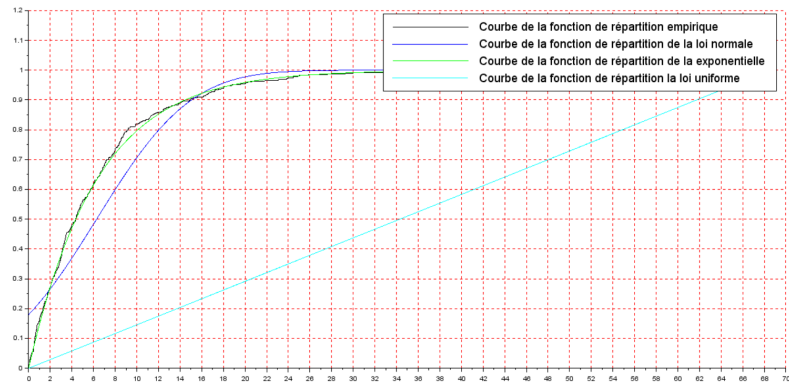


8.2 Superposition des fonctions de densité et de l'histogramme

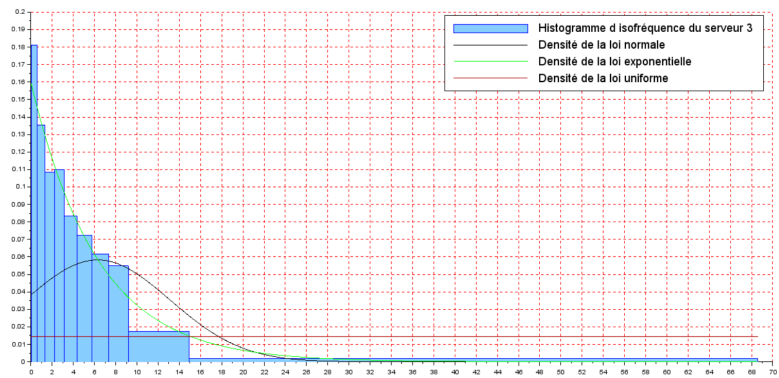


9 Serveur 3

9.1 Superposition des fonctions de répartition



9.2 Superposition des fonctions de densité et de l'histogramme



A Etude statistique des temps interarrivés

A.1 Indicateurs de position et de dispersion

```
// Extraction des temps inter-arrivées
t_ia = data(2:$, 2) - data(1:1237, 2);

extremes = [min(t_ia), max(t_ia)] // calcul du min et du max
moyenne = mean(t_ia) // calcul de la moyenne
mediane = perctl(t_ia,50) // calcul de la mediane

// calcul de la variance et de l'écart-type
v = variance(t_ia)
s = stdev(t_ia)

// calcul de l'étendue
etendue = extremes(2) - extremes(1)

Q1 = perctl(t_ia, 25) // premier quartile
Q3 = perctl(t_ia, 75) // troisième quartile
IQ = Q3(1) - Q1(1) // intervalle interquartile
```

A.2 Fonction de répartition

```
// Extraction des temps inter-arrivées
t_ia = data(2:$, 2) - data(1:1237, 2);

tab = tabul(t_ia, 'i'); // construction du tableau des effectifs
tab(:,2) = tab(:,2)/length(t_ia); // calcul des fréquences
F = cumsum(tab(:,2)); // calcul des fréquences cumulées
plot2d2(tab(:,1),F)
legend("Fonction de répartitions des temps interarrivés")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];
```

A.3 Histogramme

A.3.1 Histogramme avec classes isoamplitudes

```
// Extraction des temps inter-arrivées
t_ia = data(2:$, 2) - data(1:1237, 2);

C = linspace(min(t_ia), max(t_ia), 11) // calcul des classes
```



```
histplot(C, t_ia, style=2) // dessine l'histogramme
legend("Histogramme d isoamplitude des temps-interarrivés")
```

A.3.2 Histogramme avec classes isofréquences

```
// Extraction des temps inter-arrivées
t_ia = data(2:$, 2) - data(1:1237, 2);

deciles=perctl(t_ia,10:10:90) // Calcul des déciles
// Affectations d'isofréquences comme bornes de classes
for i=2:10
    ClassesDeciles(i)=deciles(i-1)
end
ClassesDeciles(1)=min(t_ia)
ClassesDeciles(11)=max(t_ia)

histplot(ClassesDeciles,t_ia,style=2) // dessine l'histogramme
legend("Histogramme d isofréquences des temps-interarrivés")
```

B Etude statistique des temps de service

B.1 Indicateurs de position et de dispersion

B.1.1 Serveur 1

```
// Extraction des temps de service
index_bool = ( data(:, 3) = 1 )
tabS1 = data(index_bool, :)
t_s1 = tabS1(1:$,4)

extremesS1 = [min(t_s1), max(t_s1)] // calcul du min et du max
moyenneS1 = mean(t_s1) // calcul de la moyenne
medianeS1 = perctl(t_s1,50) // calcul de la mediane

// calcul de la variance et de l'écart-type
vS1 = variance(t_s1)
sS1 = stdev(t_s1)

// calcul de l'étendue
etendueS1 = extremesS1(2) - extremesS1(1)

Q1S1 = perctl(t_s1, 25) // premier quartile
Q3S1 = perctl(t_s1, 75) // troisième quartile
IQS1 = Q3S1(1) - Q1S1(1) // intervalle interquartile
```

B.1.2 Serveur 2

```
// Extraction des temps de service
index_bool = ( data(:, 3) = 2 )
tabs2 = data(index_bool, :)
t_s2 = tabs2(1:$,4)

extremesS2 = [min(t_s2), max(t_s2)] // calcul du min et du max
moyenneS2 = mean(t_s2) // calcul de la moyenne
medianeS2 = perctl(t_s2,50) // calcul de la mediane

// calcul de la variance et de l'écart-type
vS2 = variance(t_s2)
sS2 = stdev(t_s2)

// calcul de l'étendue
etendueS2 = extremesS2(2) - extremesS2(1)

Q1S2 = perctl(t_s2, 25) // premier quartile
Q3S2 = perctl(t_s2, 75) // troisième quartile
IQS2 = Q3S2(1) - Q1S2(1) // intervalle interquartile
```

B.1.3 Serveur 3

```
// Extraction des temps de service
index_bool = ( data(:, 3) = 3 )
tabS3 = data(index_bool, :)
t_s3 = tabS3(1:$,4)

extremesS3 = [min(t_s3), max(t_s3)] // calcul du min et du max
moyenneS3 = mean(t_s3) // calcul de la moyenne
medianeS3 = perctl(t_s3,50) // calcul de la mediane

// calcul de la variance et de l'écart-type
vS3 = variance(t_s3)
sS3 = stdev(t_s3)

// calcul de l'étendue
etendueS3 = extremesS3(2) - extremesS3(1)

Q1S3 = perctl(t_s3, 25) // premier quartile
Q3S3 = perctl(t_s3, 75) // troisième quartile
IQS3 = Q3S3(1) - Q1S3(1) // intervalle interquartile
```

B.2 Fonctions de répartitions

B.2.1 Serveur 1

```
// Extraction des temps de service

index_bool = ( data(:, 3) == 1 )
tabS1 = data(index_bool, :)
t_s1 = tabS1(1:$,4);

tab = tabul(t_s1,'i')
tab(:,2) = tab(:,2)/length(t_s1)
F = cumsum(tab(:,2))

plot2d2(tab(:,1),F)
legend("Fonction de répartitions des temps de service")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];
```

B.2.2 Serveur 2

```
// Extraction des temps de service

index_bool = ( data(:, 3) == 2 )
tabS1 = data(index_bool, :)
t_s1 = tabS1(1:$,4);

tab = tabul(t_s1,'i')
tab(:,2) = tab(:,2)/length(t_s1)
F = cumsum(tab(:,2))

plot2d2(tab(:,1),F)
legend("Fonction de répartitions des temps de service")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];
```

B.2.3 Serveur 3

```
// Extraction des temps de service

index_bool = ( data(:, 3) == 3 )
```

```

tabS1 = data(index_bool, :)
t_s1 = tabS1(1:$,4);

tab = tabul(t_s1,'i')
tab(:,2) = tab(:,2)/length(t_s1)
F = cumsum(tab(:,2))

plot2d2(tab(:,1),F)
legend("Fonction de répartitions des temps de service")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

```

B.3 Histogrammes

B.3.1 Serveur 1

```

// Extraction des temps de service
index_bool = ( data(:, 3) == 1 )
tabS1 = data(index_bool, :)
t_s1 = tabS1(1:$,4);

deciles=perctl(t_s1,10:10:90);
for i=2:10
    ClassesDeciles(i)=deciles(i-1)
end
ClassesDeciles(1)=min(t_s1)
ClassesDeciles(11)=max(t_s1)

histplot(ClassesDeciles,t_s1,style=2)
legend("Histogramme d isofréquence du serveur 1")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

```

B.3.2 Serveur 2

```

// Extraction des temps de service
index_bool = ( data(:, 3) == 2 )
tabS2 = data(index_bool, :)
t_s2 = tabS2(1:$,4);

```

```

deciles=perctl(t_s2,10:10:90);
for i=2:10
    ClassesDeciles(i)=deciles(i-1)
end
ClassesDeciles(1)=min(t_s2)
ClassesDeciles(11)=max(t_s2)

histplot(ClassesDeciles,t_s2,style=2)
legend("Histogramme d isofréquence du serveur 2")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

```

B.3.3 Serveur 3

```

// Extraction des temps de service
index_bool = ( data(:, 3) == 3 )
tabS3 = data(index_bool, :)
t_s3 = tabS3(1:$,4);

deciles=perctl(t_s3,10:10:90);
for i=2:10
    ClassesDeciles(i)=deciles(i-1)
end
ClassesDeciles(1)=min(t_s3)
ClassesDeciles(11)=max(t_s3)

histplot(ClassesDeciles,t_s3,style=2)
legend("Histogramme d isofréquence du serveur 3")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

```

C Ajustement graphique à des lois mathématiques

C.1 Tous les serveurs

C.1.1 Superposition des fonctions de répartitions

```

// Extraction des temps inter-arrivées
t_ia = data(2:$, 2) - data(1:1237, 2);

```

```

tab = tabul(t_ia, 'i'); // construction du tableau des effectifs
tab(:,2) = tab(:,2)/length(t_ia); // calcul des fréquences
F = cumsum(tab(:,2)); // calcul des fréquences cumulées
plot2d2(tab(:,1),F)
legend("Fonction de répartitions des temps interarrivés")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

// Répartition loi normale
a=min(t_ia):0.01:max(t_ia)
m=ones(a)*mean(t_ia)
s=ones(a)*stdev(t_ia)
[P,Q]=cdfnor("PQ",a,m,s)
plot2d2(a,P,style=2)

// Repartition loi exponentielle
lambda=1/mean(t_ia)
b=1-exp(-lambda*a)
plot2d2(a,b,style=3)

// Repartition loi uniforme
c=(a-min(t_ia))/(max(t_ia)-min(t_ia))
plot2d2(a,c,style=4)

legend("Courbe de la fonction de répartition empirique","Courbe de la fonction
de répartition de la loi normale","Courbe de la fonction de répartition de la
exponentielle","Courbe de la fonction de répartition la loi uniforme")

```

C.1.2 Superposition des fonctions de densité et de l'histogramme

```

// Extraction des temps inter-arrivées
t_ia = data(2:$, 2) - data(1:1237, 2);

// Histogramme
deciles=perctl(t_ia,10:10:90);
for i=2:10
    ClassesDeciles(i)=deciles(i-1)
end
ClassesDeciles(1)=min(t_ia)
ClassesDeciles(11)=max(t_ia)
histplot(ClassesDeciles,t_ia,style=2)

// Densité de la loi normale

```

```

a=min(t_ia):0.01:max(t_ia)
m=mean(t_ia)
v=stdev(t_ia)
b=(1/(v*sqrt(2*pi)))*exp((-1/2)*((a-m)/v)^2))
plot2d2(a,b,style=1)

// Densité de la loi exponentielle
lambda=1/mean(t_ia)
b=lambda*exp(-lambda*a)
plot2d2(a,b,style=3)

// Densité de la loi uniforme
h=1/(max(t_ia)-min(t_ia))
b=ones(a)*h
plot2d2(a,b,style=20)

legend("Histogramme d isofréquence des temps interarrivés","Densité de la loi
normale","Densité de la loi exponentielle","Densité de la loi uniforme")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

```

C.2 Serveur 1

C.2.1 Superposition des fonctions de répartitions

```

// Extraction des temps de service

index_bool = ( data(:, 3) == 1 )
tabS1 = data(index_bool, :)
t_s1 = tabS1(1:$,4);

// Repartition empirique
tab = tabul(t_s1,'i')
tab(:,2) = tab(:,2)/length(t_s1)
F = cumsum(tab(:,2))

plot2d2(tab(:,1),F)

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

```

```

// Répartition loi normale
a=min(t_s1):0.01:max(t_s1)
m=ones(a)*mean(t_s1)
s=ones(a)*stdev(t_s1)
[P,Q]=cdfnor("PQ",a,m,s)
plot2d2(a,P,style=2)

// Repartition loi exponentielle
lambda=1/mean(t_s1)
b=1-exp(-lambda*a)
plot2d2(a,b,style=3)

// Repartition loi uniforme
c=(a-min(t_s1))/(max(t_s1)-min(t_s1))
plot2d2(a,c,style=4)

legend("Courbe de la fonction de répartition empirique","Courbe de la fonction
de répartition de la loi normale","Courbe de la fonction de répartition de la
exponentielle","Courbe de la fonction de répartition la loi uniforme")

```

C.2.2 Superposition des fonctions de densité et de l'histogramme

```

// Extraction des temps de service
index_bool = ( data(:, 3) == 1 )
tabS1 = data(index_bool, :)
t_s1 = tabS1(1:$,4);

deciles=perctl(t_s1,10:10:90);
for i=2:10
    ClassesDeciles(i)=deciles(i-1)
end
ClassesDeciles(1)=min(t_s1)
ClassesDeciles(11)=max(t_s1)

histplot(ClassesDeciles,t_s1,style=2)

// Densité de la loi normale
a=min(t_s1):0.01:max(t_s1)
m=mean(t_s1)
v=stdev(t_s1)
b=(1/(v*sqrt(2*pi)))*exp((-1/2)*((a-m)/v)^2)
plot2d2(a,b,style=1)

// Densité de la loi exponentielle
lambda=1/mean(t_s1)
b=lambda*exp(-lambda*a)

```



```

plot2d2(a,b,style=3)

// Densité de la loi uniforme
h=1/(max(t_s1)-min(t_s1))
b=ones(a)*h
plot2d2(a,b,style=20)

legend("Histogramme d isofréquence du serveur 1","Densité de la loi
normale","Densité de la loi exponentielle","Densité de la loi uniforme")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

```

C.3 Serveur 2

C.3.1 Superposition des fonctions de répartitions

```

// Extraction des temps de service

index_bool = ( data(:, 3) == 2 )
tabS2 = data(index_bool, :)
t_s2 = tabS2(1:$,4);

// Répartition empirique
tab = tabul(t_s2,'i')
tab(:,2) = tab(:,2)/length(t_s2)
F = cumsum(tab(:,2))

plot2d2(tab(:,1),F)

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];

// Répartition loi normale
a=min(t_s2):0.01:max(t_s2)
m=ones(a)*mean(t_s2)
s=ones(a)*stdev(t_s2)
[P,Q]=cdfnor("PQ",a,m,s)
plot2d2(a,P,style=2)

// Répartition loi exponentielle
lambda=1/mean(t_s2)

```

```

b=1-exp(-lambda*a)
plot2d2(a,b,style=3)

// Repartition loi uniforme
c=(a-min(t_s2))/(max(t_s2)-min(t_s2))
plot2d2(a,c,style=4)

legend("Courbe de la fonction de répartition empirique","Courbe de la fonction
de répartition de la loi normale","Courbe de la fonction de répartition de la
exponentielle","Courbe de la fonction de répartition la loi uniforme")

```

C.3.2 Superposition des fonctions de densité et de l'histogramme

```

// Extraction des temps de service
index_bool = ( data(:, 3) == 2 )
tabS2 = data(index_bool, :)
t_s2 = tabS2(1:$,4);

deciles=perctl(t_s2,10:10:90);
for i=2:10
    ClassesDeciles(i)=deciles(i-1)
end
ClassesDeciles(1)=min(t_s2)
ClassesDeciles(11)=max(t_s2)

histplot(ClassesDeciles,t_s2,style=2)

// Densité de la loi normale
a=min(t_s2):0.01:max(t_s2)
m=mean(t_s2)
v=stdev(t_s2)
b=(1/(v*sqrt(2*pi)))*exp((-1/2)*((a-m)/v)^2))
plot2d2(a,b,style=1)

// Densité de la loi exponentielle
lambda=1/mean(t_s2)
b=lambda*exp(-lambda*a)
plot2d2(a,b,style=3)

// Densité de la loi uniforme
h=1/(max(t_s2)-min(t_s2))
b=ones(a)*h
plot2d2(a,b,style=20)

legend("Histogramme d isofréquence du serveur 2","Densité de la loi
normale","Densité de la loi exponentielle","Densité de la loi uniforme")

```

```
// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];
```

C.4 Serveur 3

C.4.1 Superposition des fonctions de répartitions

```
// Extraction des temps de service
```

```
index_bool = ( data(:, 3) == 3 )
tabS3 = data(index_bool, :)
t_s3 = tabS3(1:$,4);
```

```
// Repartition empirique
tab = tabul(t_s3,'i')
tab(:,2) = tab(:,2)/length(t_s3)
F = cumsum(tab(:,2))
```

```
plot2d2(tab(:,1),F)
```

```
// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];
```

```
// Répartition loi normale
a=min(t_s3):0.01:max(t_s3)
m=ones(a)*mean(t_s3)
s=ones(a)*stdev(t_s3)
[P,Q]=cdfnor("PQ",a,m,s)
plot2d2(a,P,style=2)
```

```
// Repartition loi exponentielle
lambda=1/mean(t_s3)
b=1-exp(-lambda*a)
plot2d2(a,b,style=3)
```

```
// Repartition loi uniforme
c=(a-min(t_s3))/(max(t_s3)-min(t_s3))
plot2d2(a,c,style=4)
```

```
legend("Courbe de la fonction de répartition empirique","Courbe de la fonction
de répartition de la loi normale","Courbe de la fonction de répartition de la
```

```
exponentielle","Courbe de la fonction de répartition la loi uniforme")
```

C.4.2 Superposition des fonctions de densité et de l'histogramme

```
// Extraction des temps de service
index_bool = ( data(:, 3) == 3 )
tabS3 = data(index_bool, :)
t_s3 = tabS3(1:$,4);

deciles=perctl(t_s3,10:10:90);
for i=2:10
    ClassesDeciles(i)=deciles(i-1)
end
ClassesDeciles(1)=min(t_s3)
ClassesDeciles(11)=max(t_s3)

histplot(ClassesDeciles,t_s3,style=2)

// Densité de la loi normale
a=min(t_s3):0.01:max(t_s3)
m=mean(t_s3)
v=stdev(t_s3)
b=(1/(v*sqrt(2*pi)))*exp((-1/2)*((a-m)/v)^2))
plot2d2(a,b,style=1)

// Densité de la loi exponentielle
lambda=1/mean(t_s3)
b=lambda*exp(-lambda*a)
plot2d2(a,b,style=3)

// Densité de la loi uniforme
h=1/(max(t_s3)-min(t_s3))
b=ones(a)*h
plot2d2(a,b,style=20)

legend("Histogramme d isofréquence du serveur 3","Densité de la loi
normale","Densité de la loi exponentielle","Densité de la loi uniforme")

// Définition des paramètres d'affichages
a=gca();
a.x_location = "origin";
a.grid=[5,5];
```