INSA – MS ESD Année 2019-2020
ROUFFIAC Jean-Eudes

STAT 1 : Sélection de variables et validation de modèles

Rapport de TP n°3

Titre : *Étude par simulation du niveau et de la puissance empirique de quelques tests non-paramétriques*

L'objectif de ce TP 3 est d'étudier le niveau empirique ainsi que la puissance empirique du test de conformité de la valeur d'un paramètre de la loi de Weibull standard.

On se place dans le cadre de la loi de Weibull standard de paramètre θ. On la notera $W(\theta)$. On dispose d'un échantillon $X_1, X_2, ..., X_n$ de variables aléatoires indépendantes de même loi $W(\theta)$. On souhaite construire un test de conformité pour le paramètre θ de la forme :

$$H_0 : "\theta = \theta_0" \text{ contre } H_1 : "\theta \neq \theta_0$$

où θ_0 est donné à priori.

On sait estimer le paramètre θ par l'estimateur du maximum de vraisemblance défini par :

$$T_n = arg \min_{z>0} G(z)$$

avec

$$G(z) = -n \log(z) - (z-1) \sum_{j=1}^{n} \log(X_j) + \sum_{j=1}^{n} X_j^z$$

Pour cet estimateur, on dispose du TLC suivant :

$$\sqrt{G''(T_n)}(T_n - \theta) \xrightarrow{n\to\infty} N(0,1)$$

avec

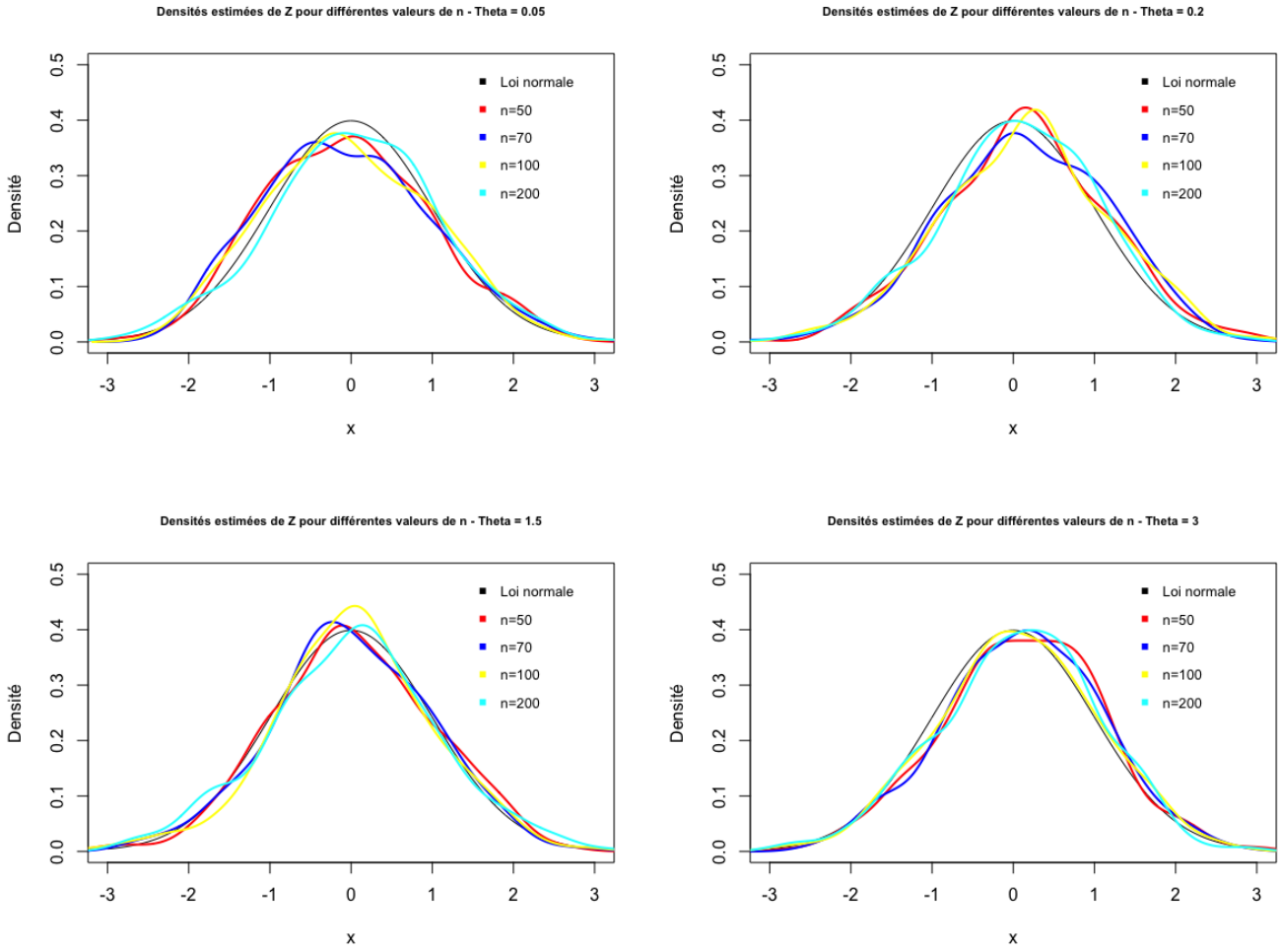
$$G''(z) = \frac{n}{z^2} + \sum_{j=1}^{n} (\log(X_j))^2 X_j^z$$

Il est donc possible pour n assez grand de construire un test de conformité pour le paramètre de la loi de Weibull.

Pour tester, au risque $\alpha \in]0,1[$, l'hypothèse nulle $H_0 : "\theta = \theta_0$ contre l'alternative $H_1 : "\theta \neq \theta_0$. On utilise donc la statistique de test Z définie par :

$$Z = \sqrt{G''(T_n)}(T_n - \theta)$$

qui suit approximativement sous H_0 une loi $N(0,1)$. Nous allons dans un premier temps étudier la qualité de cette approximation pour les valeurs de $n = 50, 70, 100, 200$ et dans le cas des valeurs $\theta_0 = 0.05, 0.2, 1.5$ et 3. On calcule donc la valeur z_{obs} de Z sur les données. On a $z_{obs} = \sqrt{g''(T_n)}(t_n - \theta)$ où t_n est l'estimation du paramètre θ fournie par l'algorithme de Newton-Raphson, et g est la version déterministe de G. Nous présenterons les résultats dans les 4 graphiques suivants où l'on superposera les densités de Z pour les différentes valeurs de n pour chaque valeur de θ_0.



On remarque pour les différentes valeurs de θ que les densités des statistiques Z pour les différentes valeurs de n sont proches de la densité de la loi normale. Notamment pour les grandes valeurs de n. Ce qui confirme que la statistique Z définie par $Z = \sqrt{G''(T_n)}(T_n - \theta)$ suit approximativement sous H_0 une loi normale $N(0,1)$ quand n devient grand.

La zone de rejet du test est de la forme : $\{|Z| > u_1 - \alpha/2\}$ où $u_1 - \alpha/2$ désigne le quantile d'ordre $(1 - \alpha/2)$ d'une loi $N(0,1)$. Si $|z_{obs}| \leq u_1 - \alpha/2$, alors on ne peut pas rejeter l'hypothèse nulle H_0 et on considère que le paramètre n'est pas significativement différent de la valeur θ_0. Sinon, on rejette l'hypothèse nulle H_0 au risque α, et on considère que le paramètre est significativement différent de θ_0. On va donc vérifier que le test est bien calibré sous l'hypothèse nulle H_0, c'est à dire qu'on est bien amené à ne rejeter H_0 à tort que dans 5% des cas (pour un risque α de 0.05) dans un premier temps, puis dans un second temps dans 1% des cas (pour un risque α de 0.01). On calculer alors pour chaque valeur de n le pourcentage de rejet sous H_0, et on résume cela dans les deux tableaux ci-dessous :

	Niveau empirique à 5% (on est sous H_0)			
n	$\theta = 0.05$	$\theta = 0.2$	$\theta = 1.5$	$\theta = 3$
50	5.25	5.50	4.50	5.00
70	3.75	4.50	4.00	4.50
100	4.00	6.25	5.50	3.50
500	6.00	4.75	7.00	4.00
1000	4.25	3.75	5.00	5.50

	Niveau empirique à 1% (on est sous H_0)			
n	$\theta = 0.05$	$\theta = 0.2$	$\theta = 1.5$	$\theta = 3$
50	0.75	1.50	1.25	0.75
70	0.25	1.00	1.25	0.25
100	0.25	1.25	1.50	0.75
500	1.00	0.75	1.50	1.50
1000	1.50	1.00	0.75	0.50

On remarque que pour toutes les valeurs de n et de θ, le test s'est comporté comme il le fallait. C'est à dire que dans 5% des cas, il s'est trompé. On est donc bien amené à ne rejeter H_0 à tort que dans 5% des cas. Il en est de même pour le risque 1%. Pour toutes les valeurs de n et de θ, on est amené à ne rejeter H_0 que dans 1% des cas. On pouvait se douter que le test se comporterait comme il doit se comporter au vue des graphiques précédents où l'on a pu observer la bonne approximation de la loi normale par la statistique Z pour différentes valeurs de θ et de n.

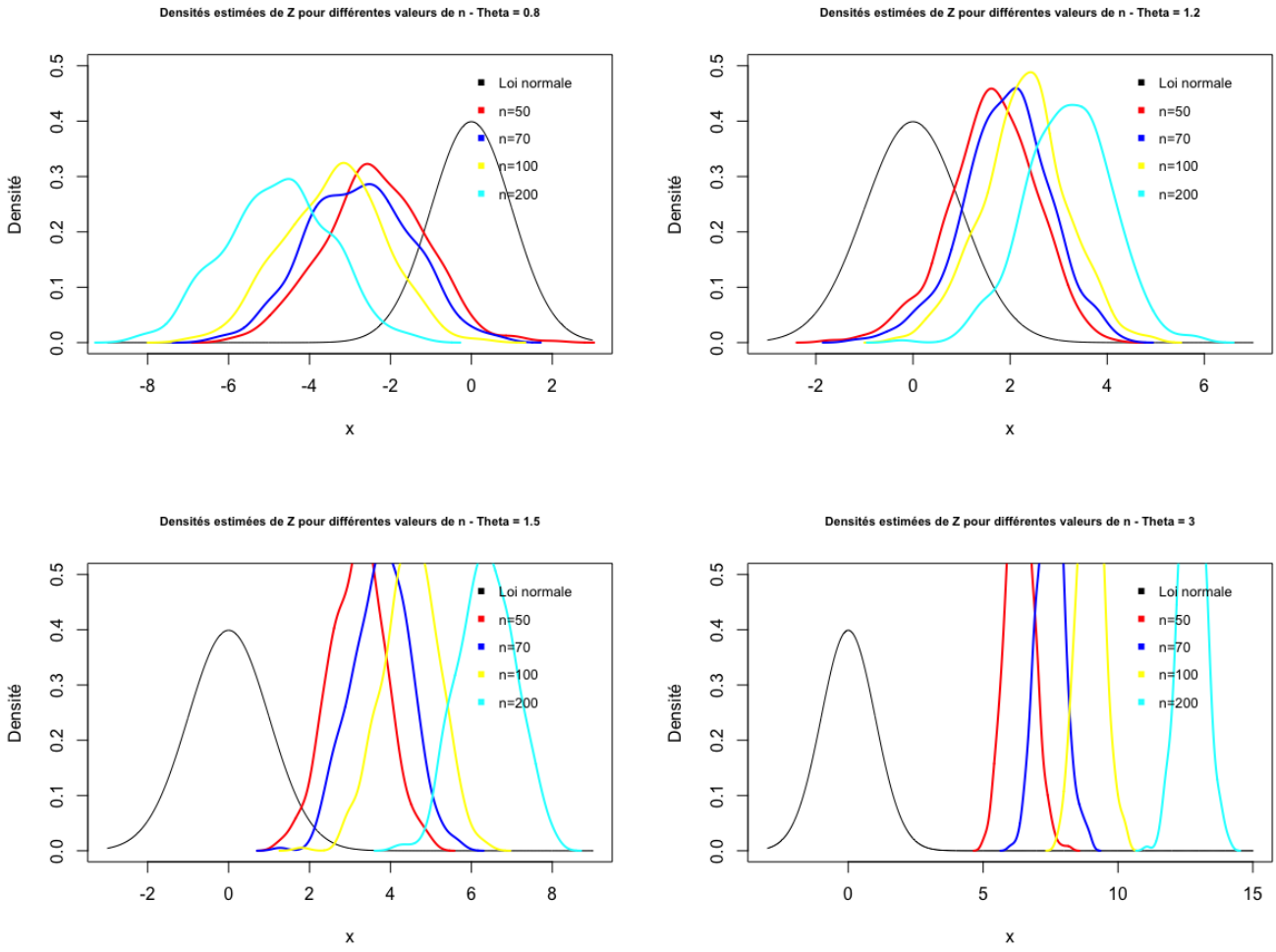
On veut maintenant étudier la puissance empirique, on se place dans le cas où $\theta_0 = 1$. On va alors se placer sous H_1 (en simulant des échantillons de la loi de Weibull de paramètre θ différents de θ_0), et on va comptabiliser le pourcentage de rejet (à raison) de H_0. On souhaite donc que ce pourcentage soit le plus élevé poossible et égal à 100% dans l'absolu. On prendra $\theta = 0.8, 1.2, 1.5$ et 3, avec $n = 50, 70, 100, 500, 1000$. On fait les tests au niveau théorique 5%.

	Niveau empirique à 5% (on est sous H_0)	Puissance empirique (on est sous H_1			
n	$\theta_0 = 1$	$\theta = 0.8$	$\theta = 1.2$	$\theta = 1.5$	$\theta = 3$
50	5.50	63.50	36.75	96.25	100.00
70	4.50	73.75	50.25	99.75	100.00
100	7.25	88.50	66.25	99.75	100.00
500	4.75	100.00	100.00	100.00	100.00
1000	5.50	100.00	100.00	100.00	100.00

On observe que le test se comporte très bien pour $\theta = 3$ puisque dès $n = 50$, le pourcentage de rejet est à 100%. Pour $\theta = 1.5$, le test se comporte aussi très bien puisque le pourcentage de rejet est à 96% pour $n = 50$, puis 100% dès $n = 70$. En revanche, le test semble avoir plus de mal pour $\theta = 0.8$ et $\theta = 1.2$. En effet, le pourcentage de rejet est à 100% seulement pour un très grand nombre de données (plusieurs centaines). Pour un faible nombre de données, le pourcentage de rejet est de 63% pour $\theta = 0.8$ et 37% pour $\theta = 1.2$. Donc plus θ est proche de $\theta_0 = 1$, plus il faut un grand nombre de données pour

que le test soit efficace.

On peut d'ailleurs se rendre compte de cela sur les graphiques ci-dessous. On a représenté les densités des statistiques Z sous H_1 ainsi que la densité de la loi normale pour les différentes valeurs de n et de θ. On voit alors très distinctement que pour un θ proche de θ_0, plus les valeurs de n sont petites, plus la densité de Z "chevauche" celle de la loi normale, ce qui implique la difficulté du test à séparer pour des petites valeurs de n et un θ proche de θ_0.



Tout le code R de cette partie se trouve en annexe, ainsi que dans un fichier .R joint au compte rendu.

Annexe

```
### Partie 4 : Etude puissance test de conformit  loi de Weibull


g <- function(x, theta){
  n = length(x)
  return(-n*log(theta) + n * log(theta) - (theta - 1) * sum(log(x
     )) + sum((x)^theta))
}

g_prime <- function(z, x){
  n = length(x)
  return(-(n/z)- sum(log(x)) + sum(log(x)*x^z))
}

g_seconde <- function(z, x){
  n = length(x)
  return((n/(z*z)) + sum(log(x)*log(x)*x^z))
}

est_teta <- function(x){
  z = 1
  k = 0
  z1 = z - (g_prime(z, x) / g_seconde(z, x))
  while ((abs(z1 - z)) > 0.0001  && k < 100)  {
    z = z1
    z1 = z - (g_prime(z, x) / g_seconde(z, x))
    k = k + 1
  }
  #print(k)
  return(z1)
}

variable_Z <- function(theta,theta_theorique, x){
  n = length(x)
  return(sqrt(g_seconde(theta,x))*(theta-theta_theorique))
}

set.seed(20)

valn <- c(50,70,100,200,500,1000)
valtheta <- c(0.05,0.2,1.5,3)

theta1 = theta2 = theta3 = theta4  = matrix(NA, nrow=6, ncol=B)
res1 = res2 = res3 = res4 = matrix(NA, nrow=6, ncol=B)

l = 1

for (theta_theorique in valtheta) {
```

```
  B <- 400 ; N <- 1000
  dataW <- matrix(NA, ncol=B, nrow=N)
  for(b in 1:B){
    dataW[,b] = rweibull(N,theta_theorique,1)
  }

    for(j in 1:B){
      k = 1

      for(i in valn) {
        if (l == 1){
          theta1[k, j] = est_teta(dataW[1:i,j])
          res1[k,j] = variable_Z(theta1[k,j],theta_theorique,
             dataW[1:i,j])
          k = k + 1
        }

        if (l == 2){
          theta2[k, j] = est_teta(dataW[1:i,j])
          res2[k,j] = variable_Z(theta2[k,j],theta_theorique,
             dataW[1:i,j])
          k = k + 1
        }

        if (l == 3){
          theta3[k, j] = est_teta(dataW[1:i,j])
          res3[k,j] = variable_Z(theta3[k,j],theta_theorique,
             dataW[1:i,j])
          k = k + 1
        }

        if (l == 4){
          theta4[k, j] = est_teta(dataW[1:i,j])
          res4[k,j] = variable_Z(theta4[k,j],theta_theorique,
             dataW[1:i,j])
          k = k + 1
        }

      }
    }

  l =l + 1
}

x = seq(-3,3, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s␣estim es␣de␣Z␣pour␣diff rentes␣valeurs␣de␣n
   ␣-␣Theta␣=␣0.05", xlab = "x", ylab = "Densit ")
lines(density(res1[1,]), lwd=2, col = 'red')
lines(density(res1[2,]), lwd=2, col = 'blue')
```

```
lines(density(res1[3,]), lwd=2, col = 'yellow')
lines(density(res1[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)

x = seq(-3,3, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s estim es de Z pour diff rentes valeurs de n
   - Theta = 0.2", xlab = "x", ylab = "Densit ")
lines(density(res2[1,]), lwd=2, col = 'red')
lines(density(res2[2,]), lwd=2, col = 'blue')
lines(density(res2[3,]), lwd=2, col = 'yellow')
lines(density(res2[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)

x = seq(-3,3, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s estim es de Z pour diff rentes valeurs de n
   - Theta = 1.5", xlab = "x", ylab = "Densit ")
lines(density(res3[1,]), lwd=2, col = 'red')
lines(density(res3[2,]), lwd=2, col = 'blue')
lines(density(res3[3,]), lwd=2, col = 'yellow')
lines(density(res3[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)

x = seq(-3,3, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s estim es de Z pour diff rentes valeurs de n
   - Theta = 3", xlab = "x", ylab = "Densit ")
lines(density(res4[1,]), lwd=2, col = 'red')
lines(density(res4[2,]), lwd=2, col = 'blue')
lines(density(res4[3,]), lwd=2, col = 'yellow')
lines(density(res4[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)



resultat1 = resultat2 = resultat3 = resultat4  = matrix(NA, nrow=
   6, ncol=2)

niveau1 <- 0.05
niveau2 <- 0.01

for (i in 1:6){
```

```
  nrejet1 = nrejet2= nrejet3= nrejet4= nrejet5= nrejet6= nrejet7=
      nrejet8 = 0
  for(j in 1:B){
    if(abs(res1[i,j]) > qnorm(1-niveau1/2)){nrejet1 <- nrejet1 +
       1}
    if(abs(res1[i,j]) > qnorm(1-niveau2/2)){nrejet2 <- nrejet2 +
       1}
    if(abs(res2[i,j]) > qnorm(1-niveau1/2)){nrejet3 <- nrejet3 +
       1}
    if(abs(res2[i,j]) > qnorm(1-niveau2/2)){nrejet4 <- nrejet4 +
       1}
    if(abs(res3[i,j]) > qnorm(1-niveau1/2)){nrejet5 <- nrejet5 +
       1}
    if(abs(res3[i,j]) > qnorm(1-niveau2/2)){nrejet6 <- nrejet6 +
       1}
    if(abs(res4[i,j]) > qnorm(1-niveau1/2)){nrejet7 <- nrejet7 +
       1}
    if(abs(res4[i,j]) > qnorm(1-niveau2/2)){nrejet8 <- nrejet8 +
       1}

  }
  resultat1[i,1]=nrejet1
  resultat1[i,2]=nrejet2
  resultat2[i,1]=nrejet3
  resultat2[i,2]=nrejet4
  resultat3[i,1]=nrejet5
  resultat3[i,2]=nrejet6
  resultat4[i,1]=nrejet7
  resultat4[i,2]=nrejet8
}

resultat1.pourcent = 100*resultat1/B
resultat2.pourcent = 100*resultat2/B
resultat3.pourcent = 100*resultat3/B
resultat4.pourcent = 100*resultat4/B
```

```
#Pouvoir s parateur du test


dataR1 <- dataR2 <- dataR3 <- dataR4 <- dataR5 <-matrix(NA, ncol=
   B, nrow=N)
for(j in 1:B){
  dataR1[,j] = rweibull(N,1,1)
  dataR2[,j] = rweibull(N,0.8,1)
  dataR3[,j] = rweibull(N,1.2,1)
```

```
  dataR4[,j] = rweibull(N,1.5,1)
  dataR5[,j] = rweibull(N,3,1)
}


set.seed(20)

valn <- c(50,70,100,200,500,1000)
valtheta <- c(1,0.8,1.2,1.5,3)

theta5 = theta6 = theta7 = theta8  = theta9 = matrix(NA, nrow=6,
   ncol=B)
res5 = res6 = res7 = res8 = res9 = matrix(NA, nrow=6, ncol=B)


theta0 = 1

for(j in 1:B){
    k = 1

    for(i in valn) {
        theta5[k, j] = est_teta(dataR1[1:i,j])
        res5[k,j] = variable_Z(theta5[k,j],theta0, dataR1[1:i,j])

        theta6[k, j] = est_teta(dataR2[1:i,j])
        res6[k,j] = variable_Z(theta6[k,j],theta0, dataR2[1:i,j])

        theta7[k, j] = est_teta(dataR3[1:i,j])
        res7[k,j] = variable_Z(theta7[k,j],theta0, dataR3[1:i,j])

        theta8[k, j] = est_teta(dataR4[1:i,j])
        res8[k,j] = variable_Z(theta8[k,j],theta0, dataR4[1:i,j])

        theta9[k, j] = est_teta(dataR5[1:i,j])
        res9[k,j] = variable_Z(theta9[k,j],theta0, dataR5[1:i,j])

        k = k +1
    }
  }


resultat5 = resultat6 = resultat7 = resultat8 = resultat9  =
   matrix(NA, nrow=6, ncol=1)

niveau1 <- 0.05

for (i in 1:6){
  nrejet5 = nrejet6 = nrejet7 = nrejet8 = nrejet9 = 0
  for(j in 1:B){
    if(abs(res5[i,j]) > qnorm(1-niveau1/2)){nrejet5 <- nrejet5 +
       1}
```

```
    if(abs(res6[i,j]) > qnorm(1-niveau1/2)){nrejet6 <- nrejet6 +
       1}
    if(abs(res7[i,j]) > qnorm(1-niveau1/2)){nrejet7 <- nrejet7 +
       1}
    if(abs(res8[i,j]) > qnorm(1-niveau1/2)){nrejet8 <- nrejet8 +
       1}
    if(abs(res9[i,j]) > qnorm(1-niveau1/2)){nrejet9 <- nrejet9 +
       1}

  }
  resultat5[i,1]=nrejet5
  resultat6[i,1]=nrejet6
  resultat7[i,1]=nrejet7
  resultat8[i,1]=nrejet8
  resultat9[i,1]=nrejet9

}

resultat5.pourcent = 100*resultat5/B
resultat6.pourcent = 100*resultat6/B
resultat7.pourcent = 100*resultat7/B
resultat8.pourcent = 100*resultat8/B
resultat9.pourcent = 100*resultat9/B




x = seq(-3,3, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s␣estim es␣de␣Z␣pour␣diff rentes␣valeurs␣de␣n
   ␣-␣Theta␣=␣0.05", xlab = "x", ylab = "Densit ")
lines(density(res5[1,]), lwd=2, col = 'red')
lines(density(res5[2,]), lwd=2, col = 'blue')
lines(density(res5[3,]), lwd=2, col = 'yellow')
lines(density(res5[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi␣normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)

x = seq(-9,3, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s␣estim es␣de␣Z␣pour␣diff rentes␣valeurs␣de␣n
   ␣-␣Theta␣=␣0.8", xlab = "x", ylab = "Densit ")
lines(density(res6[1,]), lwd=2, col = 'red')
lines(density(res6[2,]), lwd=2, col = 'blue')
lines(density(res6[3,]), lwd=2, col = 'yellow')
lines(density(res6[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi␣normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)
```

```
x = seq(-3,7, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s␣estim es␣de␣Z␣pour␣diff rentes␣valeurs␣de␣n
   ␣-␣Theta␣=␣1.2", xlab = "x", ylab = "Densit ")
lines(density(res7[1,]), lwd=2, col = 'red')
lines(density(res7[2,]), lwd=2, col = 'blue')
lines(density(res7[3,]), lwd=2, col = 'yellow')
lines(density(res7[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi␣normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)

x = seq(-3,9, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s␣estim es␣de␣Z␣pour␣diff rentes␣valeurs␣de␣n
   ␣-␣Theta␣=␣1.5", xlab = "x", ylab = "Densit ")
lines(density(res8[1,]), lwd=2, col = 'red')
lines(density(res8[2,]), lwd=2, col = 'blue')
lines(density(res8[3,]), lwd=2, col = 'yellow')
lines(density(res8[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi␣normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)


x = seq(-3,15, by = 0.01)
plot(x, dnorm(x,0,1), ylim = c(0, 0.5) , type="l", cex.main = 0.7
   , main="Densit s␣estim es␣de␣Z␣pour␣diff rentes␣valeurs␣de␣n
   ␣-␣Theta␣=␣3", xlab = "x", ylab = "Densit ")
lines(density(res9[1,]), lwd=2, col = 'red')
lines(density(res9[2,]), lwd=2, col = 'blue')
lines(density(res9[3,]), lwd=2, col = 'yellow')
lines(density(res9[4,]), lwd=2, col = 'cyan')
legend("topright",legend=c("Loi␣normale","n=50", "n=70", "n=100",
    "n=200"), col=c("black","red", "blue", "yellow", "cyan"),pch=1
   5, bty="n",cex=0.8)
```