TIAD_projet

Yu Zigong et Happy Hugo

26 fevrier 2020

```r
library('MASS')
```

1)

On fera le test suivant:

$$\text{H0: les log-returns sont centrés,}$$
$$\text{contre H1: les log-returns ne sont pas centrés.}$$

La Statistique de test : $Tn = \sqrt{n}\frac{\overline{X_n}}{\sqrt{V_n^*}}$. où $\overline{X_n}$ est la moyenne de SP, Vn^* est la variance empirique corrigée de SP.

Sous H0, Tn suit une loi normale $\mathcal{N}(0,1)$ asymptotiquement.

Sous H1, Tn est plus loin de 0 que sous H0.

On fera un test bilataire symétrique, donc la zone de rejet est de forme $\{|Tn|$ >= k$\}$, $k = \Phi^{-1}(0.995)$,où Φ est la fonction de répartition de la loi $\mathcal{N}(0,1)$.

```r
qnorm(0.995)  # On calcule k
```

```
## [1] 2.575829
```

La zone de rejet au niveau de 1% R=$\{|Tn|$ >= 2.576$\}$.

```r
SP <- SP500 # On stocke SP500 dans SP
n=length(SP)  # n est le longueur de SP
Tn = sqrt(n)*mean(SP)/sqrt(var(SP));Tn   # On calcule la statistique Tn
```

```
## [1] 2.545345
```

On a calculé $Tn = 2.545 \notin R$, donc on conserve H0 au niveau 1%, les log-returns sont bien centrés.

```r
(1-pnorm(abs(Tn),0,1))*2 # on calcule la p-valeur
```

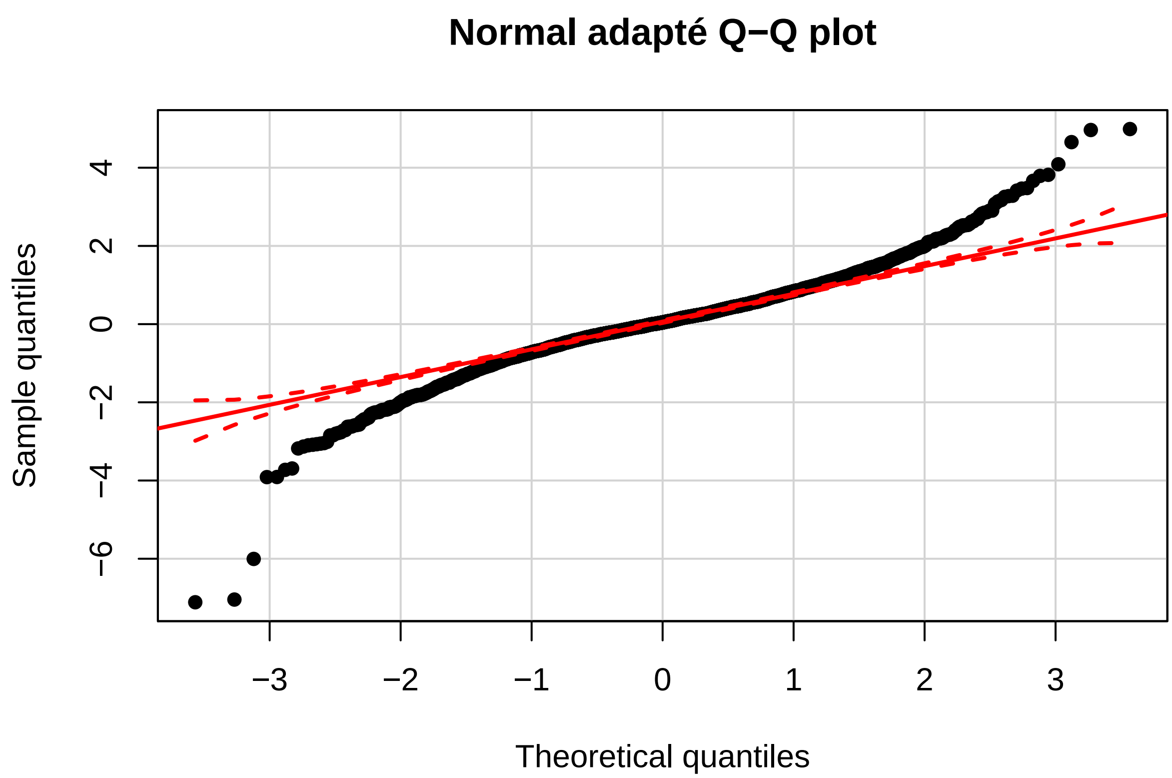
```
## [1] 0.01091698
```

p-valeur = 0.011.

2)

(a)

```r
# On fait le qq-plot avec la fonction qqPlot dans le package car!
library('car')
qqPlot( SP,distribution = "norm" ,main = "Normal adapté Q-Q plot",
        xlab = "Theoretical quantiles", ylab = "Sample quantiles",
        col.lines='red',pch=16,id=FALSE)
```




```r
# On voit que les queues de distribution sont plus dispersées.
```

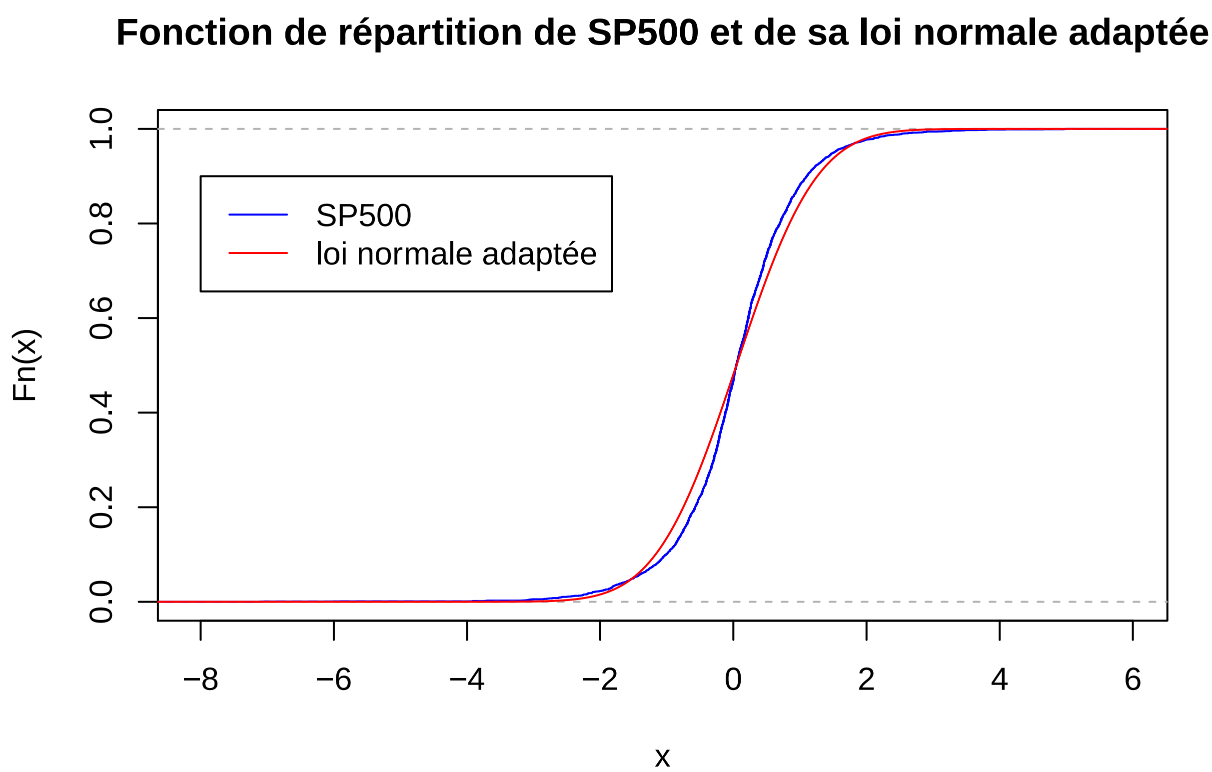
(b)

Ici, pour tracer la fonction de répartition de la loi normale adaptée de SP, on doit calculer les estimateurs des paramètres de loi normale $\mathcal{N}(m, \sigma^2)$, en plus par 1),SP est bien centré, donc m=0 et on estimera la variance de la loi normale.

On obtient les estimteurs de la loi $\mathcal{N}(0, \sigma^2)$: $\hat{\sigma_n^2} = Vn^*$, Vn^* est la variance empirique corrigée de SP.

```r
# On plot la fonction de répartition empirique de SP500 et sa loi
# normale adaptée.
plot(ecdf(SP),col='blue',
     main="Fonction de répartition de SP500 et de sa loi normale adaptée")
I <-seq(-10,7,length=1000)
```

```r
lines(I,pnorm(I,mean(SP),sqrt(var(SP))),col='red')
legend(-8,0.9, legend=c("SP500","loi normale adaptée"),
       col=c("blue","red"),lty=1,cex=1)
```




```r
library('nortest')
 # On fait le test lillie pour tester si SP500 suit loi normale.
lillie.test(SP)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  SP
## D = 0.064218, p-value < 2.2e-16
```

```r
# On fait le test Anderson-Darling pour tester si SP500 suit loi normale.
ad.test(SP)
```

```
##
##  Anderson-Darling normality test
##
## data:  SP
## A = 23.364, p-value < 2.2e-16
```

```r
# On fait le test Shapiro-Wilk pour tester si SP500 suit loi normale.
shapiro.test(SP)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  SP
## W = 0.95511, p-value < 2.2e-16
```

Le test de Lilliefors est moins performant lorsque les queues de distribution empiriques s'loignent des queues théoriques.Le test d'Anderson-Darling lui donne plus d'importance aux queues de distributions.Le test de Shapiro-Wilk est plus performant pour les faibles effectifs.

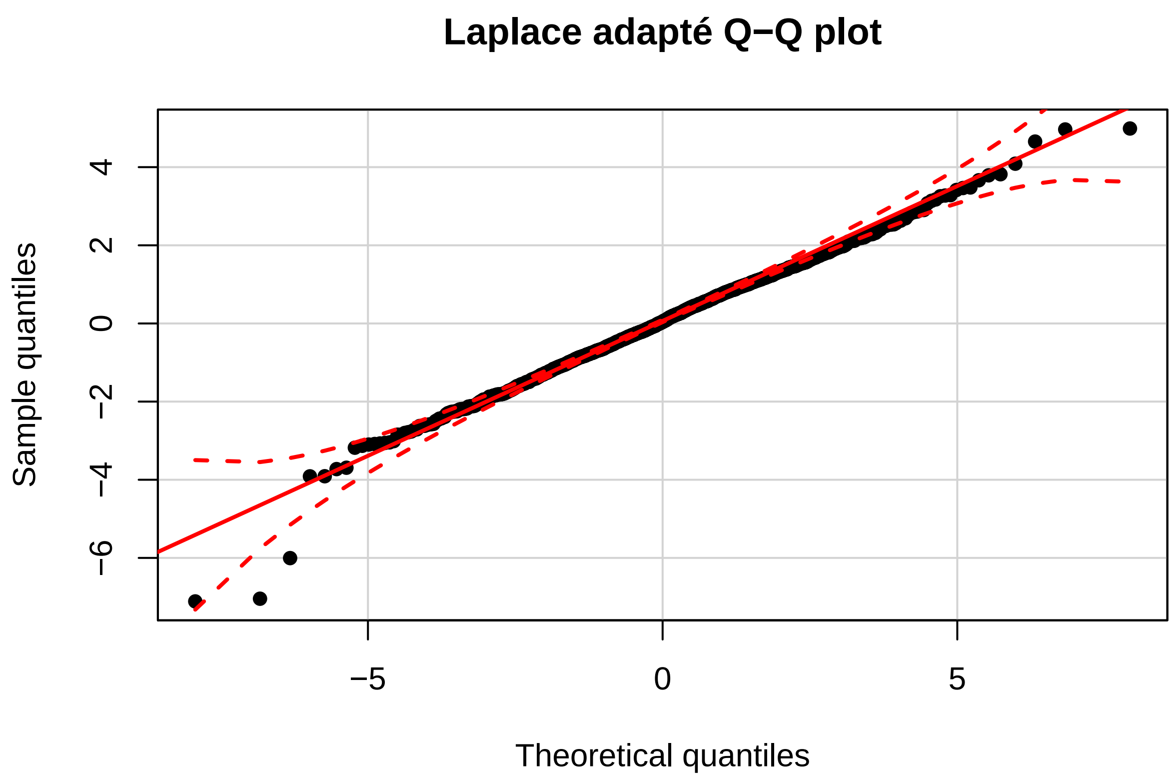
Ici, on a des queues de distribution empiriques qui s'éloignent de la théorie et un fort effectif (2780 données). Donc le test d'Anderson-Darling semble être le plus pertinent.

Cependant,les trois tests donnent la même p-valeur nulle, donc, on rejette H0 et déside que SP500 ne suit pas une loi normale.

3)

(a)

```r
#On utilise le package LaplacesDemon.
library('LaplacesDemon' )
qqPlot( SP,distribution = "laplace" ,main = "Laplace adapté Q-Q plot",
       xlab = "Theoretical quantiles", ylab = "Sample quantiles",
       col.lines='red',pch=16,id=FALSE)
```




```r
#On voit que les queues de distribution sont moins dispersées.
```

(b)
$|X|$ ~Exp(λ)
preuve:

La densité de X est $f(x) = \frac{\lambda}{2}e^{-\lambda|x|}$.

Pour $\forall t < 0, \mathbb{P}_\lambda(|X| \leq t) = 0$.

Pour $\forall t \geq 0, \mathbb{P}_\lambda(|X| \leq t) = \mathbb{P}_\lambda(-t \leq X \leq t) = \int_{-t}^{t} f(x)dx = \int_{-t}^{t} \frac{\lambda}{2}e^{-\lambda|x|}dx = \int_0^t \lambda e^{-\lambda x}dx = 1 - e^{-\lambda x}$.

$|X|$ suit une loi $\mathcal{E}(\lambda)$.

CQFM

(c)

Ici, pour tracer la fonction de répartition de la loi normale adaptée de SP, on doit calculer les estimateurs des paramètres de loi Laplace $Laplace(0, \lambda)$.

On calcule les estimteurs de la loi $Laplace(0, \lambda)$ par EMV:

La vraisemblance $\mathcal{L}(\underline{x}, \lambda) = \prod_{i=1}^{n} f(x_i) = \prod_{i=1}^{n} \frac{\lambda}{2} e^{-\lambda |x_i|}$.

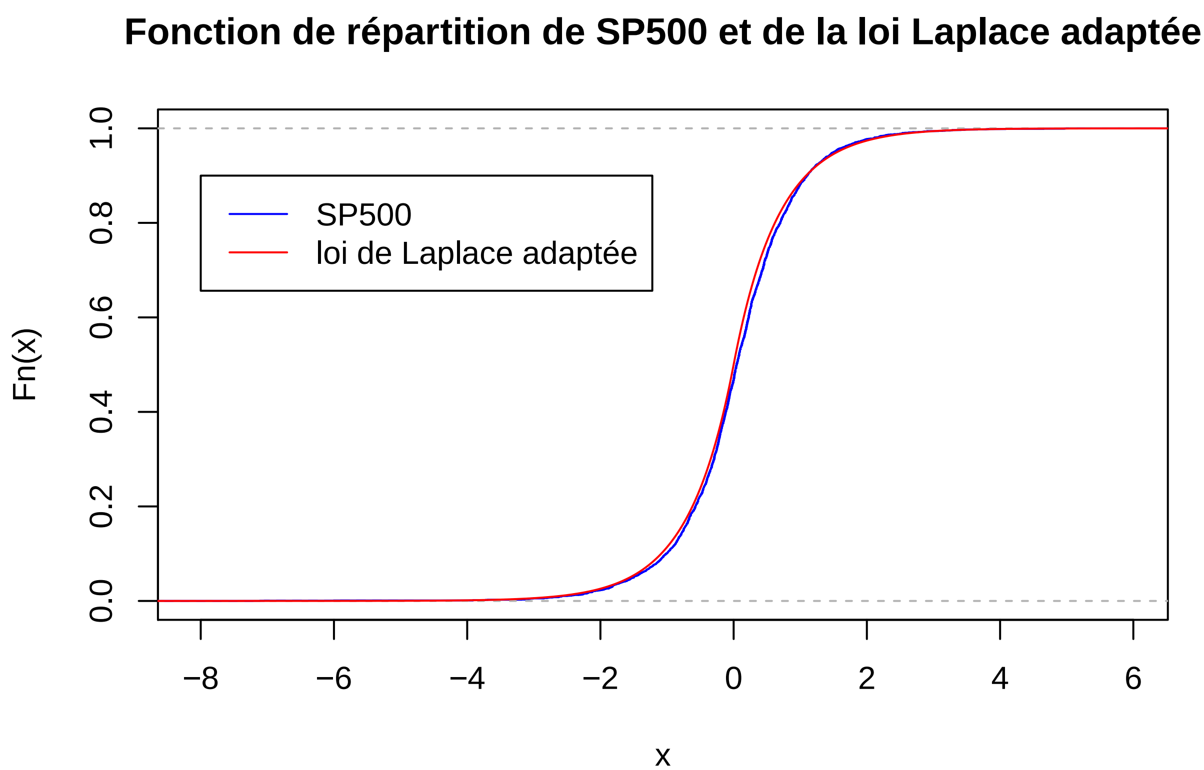
La log-vraisemblance $L(\underline{x}, \lambda) = \log(\mathcal{L}(\underline{x}, \lambda)) = n \log(\frac{\lambda}{2}) - \lambda \sum_{i=1}^{n} |x_i|$.

$\frac{dL}{dx} = \frac{n}{\lambda} - \sum_{i=1}^{n} |x_i| = 0 \Rightarrow \lambda = \frac{1}{n} \sum_{i=1}^{n} |x_i|$.

L croissant quand $\lambda \in]0, \frac{1}{n} \sum_{i=1}^{n} |x_i|$ *et décroissant quand* $\lambda \in [\frac{1}{n} \sum_{i=1}^{n} |x_i|, +\infty[$.

Donc l'estimateur $\hat{\lambda}_n = \frac{1}{n} \sum_{i=1}^{n} |x_i|$.

```
plot(ecdf(SP),col='blue',
     main="Fonction de répartition de SP500 et de la loi Laplace adaptée")
I <-seq(-10,7,length=1000)
lines(I,plaplace(I,0,mean(abs(SP))),col='red')
legend(-8,0.9, legend=c("SP500","loi de Laplace adaptée"),
       col=c("blue","red"),lty=1,cex=1)
```



On a vu que les deux courbes se sont bien prochées.

Maintenant, pour tester si SP suit une loi de Laplace, on fait le test d'Anderson-Darling par rapport à la loi exponentielle puis que $X \sim Laplace(\lambda) \Leftrightarrow |X| \sim \mathcal{E}(\lambda)$.

C'est le test suivant:(On pose X=|SP|)

$$\text{H0: X suit } \mathcal{E}(\theta),\ \theta > 0,$$
$$\text{contre H1: X ne suit pas } \mathcal{E}(\theta),\ \theta > 0.$$

On considère la statistique $W_{A,n} = n\int_{-\infty}^{+\infty} \frac{(F_n(x)-F_{\hat{\theta_n}}(x))^2}{F_{\hat{\theta_n}}(x)(1-F_{\hat{\theta_n}}(x))} dF_{\hat{\theta_n}}$, où F_n est la fonction de répartition empirique de X. Après du calcul d'un exercice en cours, en posant $Z_i = F_{\hat{\theta_n}}(X_{(i)})$, On a :

$$W_{A,n} = \sum_{i=1}^{n}(-ln(Z_i)\frac{2i-1}{Z_i} - ln(1-Z_i)\frac{2(n-i)+1}{n}) - n$$

Dans ce cas là, il y aura les échantillons nuls. Donc on les supprime pour éviter le calcul de ln(0).

La statistique de test est plus grand sous H1 que sous H0. Donc la zone de rejet R est de forme $\{W_{A,n} \geq k\}$. Dans notre cours, on a montré que sous H0 la loi de $W_{A,n}$ ne dépend de θ puisque l'on a $F_{\hat{\theta_n}}(X_i) = 1 - e^{\frac{X_i}{\overline{X_n}}} = 1 - e^{\frac{U_i}{\overline{U_n}}}$, où $U_i \sim \mathcal{E}(1)$, donc on peut bien calculer la zone de rejet au niveau 1%.

```
W_An_exp <- function(n){
  x <- rexp(n,1)   # simule n variables de loi E(1)
  y <- sort(x)        # on range x
  v <- c(1:n)
  y=1-exp(-y/mean(x))
  W_An=sum(-log(y)*(2*v-1)/n-log(1-y)*(2*(n-v)+1)/n)-n
  return(W_An)
}

pAD_exp <- function(n,N,a){
  W_An_N <- numeric(N)    # on crée un vecteur de taille N
  for (i in 1:N) {
    W_An_N[i]=W_An_exp(n)       #on fait N fois le calcule de W_An
  }
  S <-outer(W_An_N,a,"<=")   # on calcule et renvoie la proba empirique
  V <-colSums(S)
  return(V/N)
}

I <- seq(1,3,length=2001 )
#On calcule 3 fois k (car k est varié chaque fois )
1+min(which(1-pAD_exp(2778,1000,I)<=0.01))/1000
```

```
## [1] 1.836
```

```
1+min(which(1-pAD_exp(2778,1000,I)<=0.01))/1000
```

```
## [1] 1.868
```

```
1+min(which(1-pAD_exp(2778,1000,I)<=0.01))/1000
```

```
## [1] 2.039
```

La zone de rejet R au niveau 1% est $\{W_{A,n} \geq k\}$, où k est entre 1.8 et 2.2.

```
# Maintenant on calcule la sttatistique:
x=abs(SP)
y0 <- sort(x)       # on range x
y0=1-exp(-y0/mean(x))
# Dans ce cas là, il y aura les échantillons nuls. On ne les
# teste pas.
y0=y0[y0!=0]
n=length(y0)
v <- c(1:n)
W_An=sum(-log(y0)*(2*v-1)/n-log(1-y0)*(2*(n-v)+1)/n)-n;W_An
```

```
## [1] 3.181648
```

La statitique $W_An = 3.182 \in R$ en ne considérant pas les valuers nulles dans SP, dans ce cas là, on rejette H0 et déside que SP ne suit pas une loi exponentielle.

```
# On calcule la p-valeur
1-pAD_exp(2778,100000,W_An)
```

```
## [1] 0.00057
```

p-valeur=0.00054=0.054%,donc on rejette H0, et déside que |SP| ne suit pas une loi exponentielle, c'est à dire que SP ne suit pas une loi de Laplace, mais la p-valeur de ce test est plus elevée que le test d'AD de normalité.

4)
(a)
On trace la densté estimée en utilisant notre fonctions dans TD3.

```
X=SP #On stocke SP dans X
Xmax = max(abs(X))
y =seq(-2*Xmax,2*Xmax,length= 1001)

est <- function(h,X,y){
  M3=outer(y,X,"-")
  f=rowMeans(1/h*K(M3/h))
  return(f)
}
# On choisit le noyau d'Epachnikov et le crée
K <-function(x){
  return ((3/4)*(1-x^2)*(abs(x)<=1))
}
#On crée une fonction pour calculer l'intégrale
integrale <- function(X,pas)
{
  resultat=1/2*(sum(X[2:length(X)]*pas)+sum(X[1:length(X)-1]*pas))
  return(resultat)
}
#On va minimiser risque pour trouver une meilleure fenêtre
risque <- function(h,X,y)
{
  n <- length(X)
  Gh <- mean((est(h,X,X)-(1/h*K(0))/n)*n/(n-1))
  pas = y[2]-y[1]
  CV = integrale(est(h,X,y)^2,pas)-2*Gh
  return(CV)
}
#On fait varier h entre 0 et 1 pour trouver le meilleur h à 0.1 près
CV_h <- numeric(10)
for (i in 1:10 ) {
  CV_h[i]=risque(i/10,X,y)
}
which.min(CV_h)/10
```

```
## [1] 0.3
```

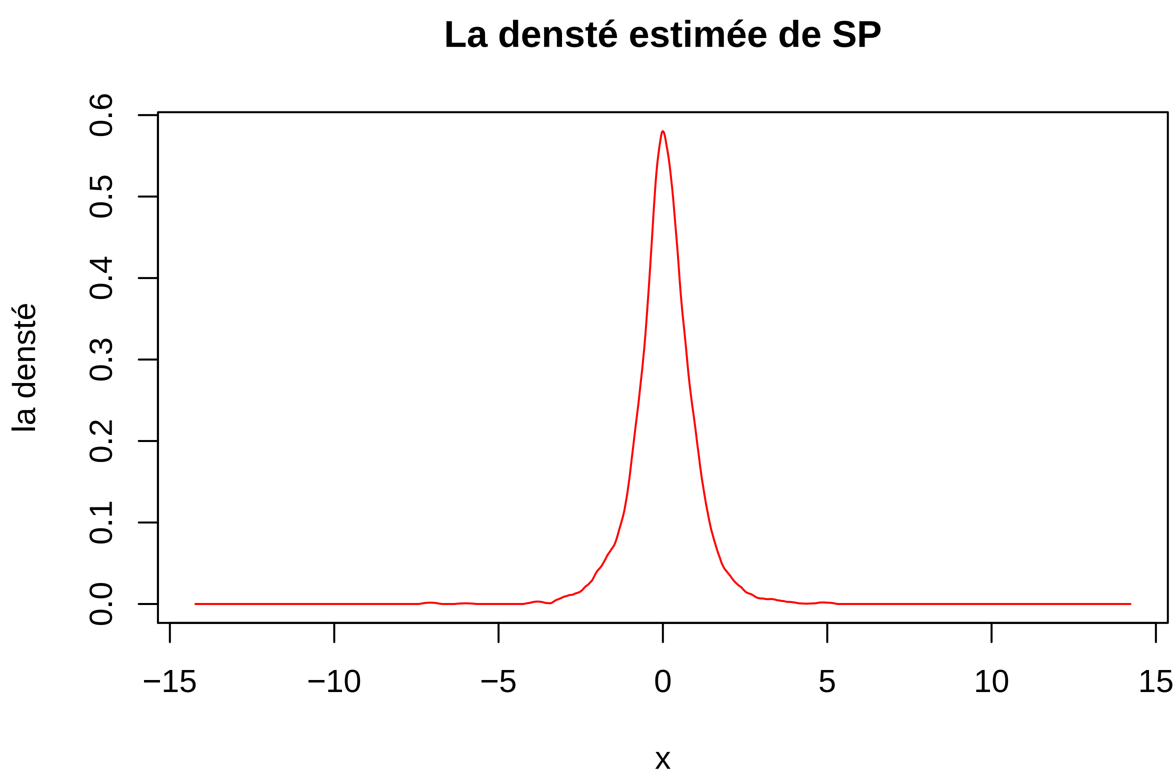
La meilleure fenêtre à 0.1 près est 0.3

```r
#On fait varier h entre 0.2 et 0.4 pour trouver le meilleur h à 0.01 près
CV_h1 <- numeric(20)
for (i in 1:20 ) {
  CV_h1[i]=risque(which.min(CV_h)/10-0.1+i/100,X,y)
}

h_min=which.min(CV_h)/10-0.1+which.min(CV_h1)/100;h_min
```

```
## [1] 0.33
```

La meilleure fenêtre à 0.01 près que l'on a obtenue est 0.33

```r
fm=est(h_min,X,y)

plot(y,fm,col='red',type = 'l',xlab = "x",ylab="la densté",
     main="La densté estimée de SP")
```



(b)

```r
plot(y,fm,type = 'l',xlim=c(-5,5))
x=SP
```

```r
d_epa = density(x,kernel = "epanechnikov",bw='nrd');d_epa$bw
```

```
## [1] 0.1549821
```

```r
d_epa0 = density(x,kernel = "epanechnikov",bw='nrd0');d_epa0$bw
```

```
## [1] 0.1315886
```

```r
d_epa_ucv = density(x,kernel = "epanechnikov",bw='ucv');d_epa_ucv$bw
```

```
## [1] 0.130009
```

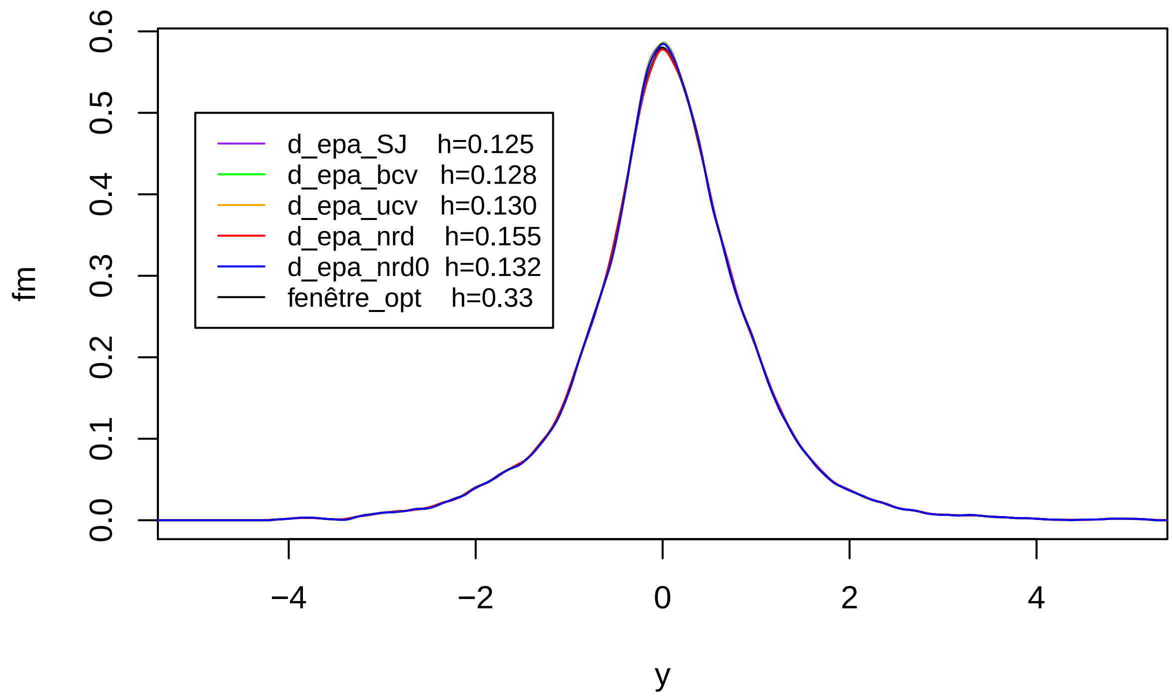
```r
d_epa_bcv = density(x,kernel = "epanechnikov",bw='bcv');d_epa_bcv$bw
```

```
## [1] 0.1279333
```

```r
d_epa_SJ = density(x,kernel = "epanechnikov",bw='SJ');d_epa_SJ$bw
```

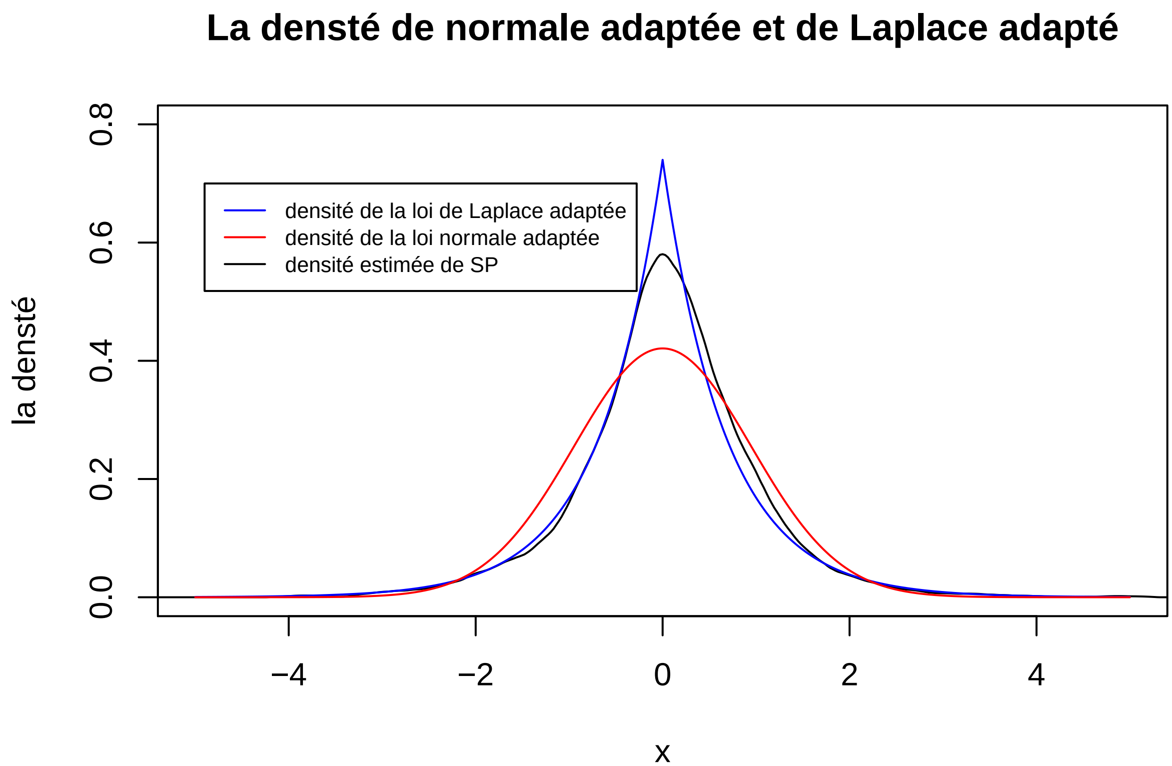
```
## [1] 0.1249875
```

```r
lines(d_epa_SJ,col='purple')
lines(d_epa_bcv,col='green')
lines(d_epa_ucv,col='orange')
lines(d_epa,col='red')
lines(d_epa0,col='blue')
legend(-5,0.5, legend=c("d_epa_SJ    h=0.125","d_epa_bcv   h=0.128",
                        "d_epa_ucv   h=0.130","d_epa_nrd    h=0.155",
                        "d_epa_nrd0  h=0.132","fenêtre_opt    h=0.33"),
       col=c("purple", "green","orange","red","blue","black"),lty=1,cex=0.8)
```



On voit que les densités estimées par les façons différentes sont bien coïncidentes grâce à la grande taille d'observations.

```
I=seq(-5,5,0.001)
plot(y,fm,type = 'l',xlim=c(-5,5),ylim=c(0,0.8),
     xlab = "x",ylab="la densté",
     main="La densté de normale adaptée et de Laplace adapté")
lines(I,dlaplace(I,0,mean(abs(SP500))),col='blue')
lines(I,dnorm(I,0,sqrt(var(SP500))),col='red')
legend(-4.9,0.7, legend=c("densité de la loi de Laplace adaptée",
                          "densité de la loi normale adaptée",
                          "densité estimée de SP"),
       col=c("blue","red","black"),lty=1,cex=0.7)
```



On voit que la densité estimée de SP est plus proche de la densité de la loi de Laplace adaptée que de la densité de la loi normale adaptée. Mais exigeantement, SP ne suit pas une loi de Laplace, on doit chercher l'autre loi pour une estimation plus précise.