ADOBE

DEBA | JC| HF

10/30/2024

Application de la théorie des valeurs extrême en finance de marché: Cas des rendements journaliers des action Adobe.

Définition du Contexte

Créée en décembre 1982 et cotée au NASDAQ depuis 1986, Adobe est un leader mondial dans l'édition de logiciels de création graphique, de montage audio et vidéo. Après une forte expansion initiale, l'entreprise a vu sa croissance freinée en 1994 par une interdiction d'acquérir des logiciels concurrents, ce qui l'a poussée à concentrer ses efforts sur ses activités existantes. Cette restriction levée en 2005, Adobe a relancé sa stratégie d'expansion et, au cours de la dernière décennie, a renforcé sa présence sur le marché européen en absorbant plusieurs concurrents. En 2022, Adobe a tenté d'acquérir l'éditeur d'images vectorielles Figma pour 20 milliards de dollars, mais cet achat a été bloqué par les autorités de régulation européennes et britanniques [**?**].

Dans un contexte d'évolution rapide des technologies depuis les années 80, accéléré de manière exponentielle au cours des deux dernières décennies, il est pertinent d'analyser comment les actions d'Adobe ont évolué dans cet environnement en perpétuelle transformation. Étudier la volatilité des rendements de ses actions s'avère crucial pour un investisseur potentiel, désireux de comprendre les risques associés à cette entreprise technologique dynamique. Une telle analyse permettra d'anticiper les pertes potentielles dans des situations de marché inattendues.

Données de Rendement

Dans le cadre cette analyse, nous travaillons sur les données de prix de clôture des actions d'Adobe sur la période 2005-2020. Ces données sont accessible sur yahoo finance depuis la library R *quantmod.* Les rendements logarithmiques quotidiens pour cette série sont calculés par:

$$R_t = \ln \frac{P_t}{P_{t-1}}$$

où P_t est le prix de clôture de l'action au jour t. Pour les besoins de la modélisation, nous travaillerons sur les pertes en considérant les rendements négatifs $L_t = -R_t$.

```r
library("quantmod")
```

```
## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: TTR

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

Import data

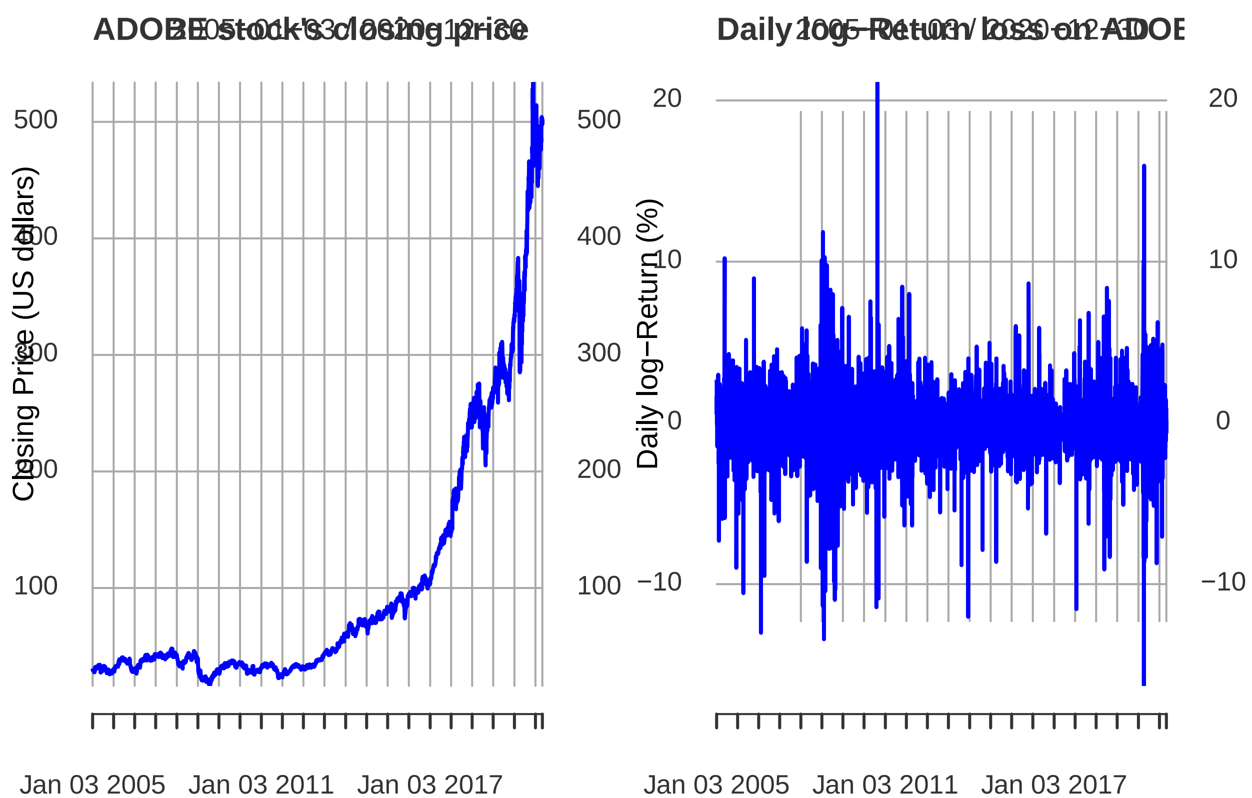
```
# Get historical data for Bitcoin from Yahoo Finance
data <- getSymbols("ADBE", src = "yahoo", from="2005-01-01", to = "2020-12-31", auto.assign = FALSE)
data <- na.omit(data)
data_cl <- Cl(data)

# Log-rendements en pourcentage
# Nous passons les pertes en positifs

data_cl$`return` <- - Delt(data_cl, k = 1, type = 'log') * 100
```

Analyse descriptive

```
# Visualiser les rendements return
par(mfrow=c(1,2))
plot(data_cl$`ADBE.Close`, main = "ADOBE stock's closing price", ylab = "Closing Price (US dollars)", c
plot(data_cl$`return`, main = "Daily log-Return loss on ADOBE stock", ylab = "Daily log-Return (%)", col
```



Au cours de la dernière decennie, le cours des actions Adobe a connu une évolution exponentielle. Coté à moins de 100$ avant 2009, le cours est passé au dessus de la barre de 200$ en 2015 et fluctue en fin 2020 autour de 400$.

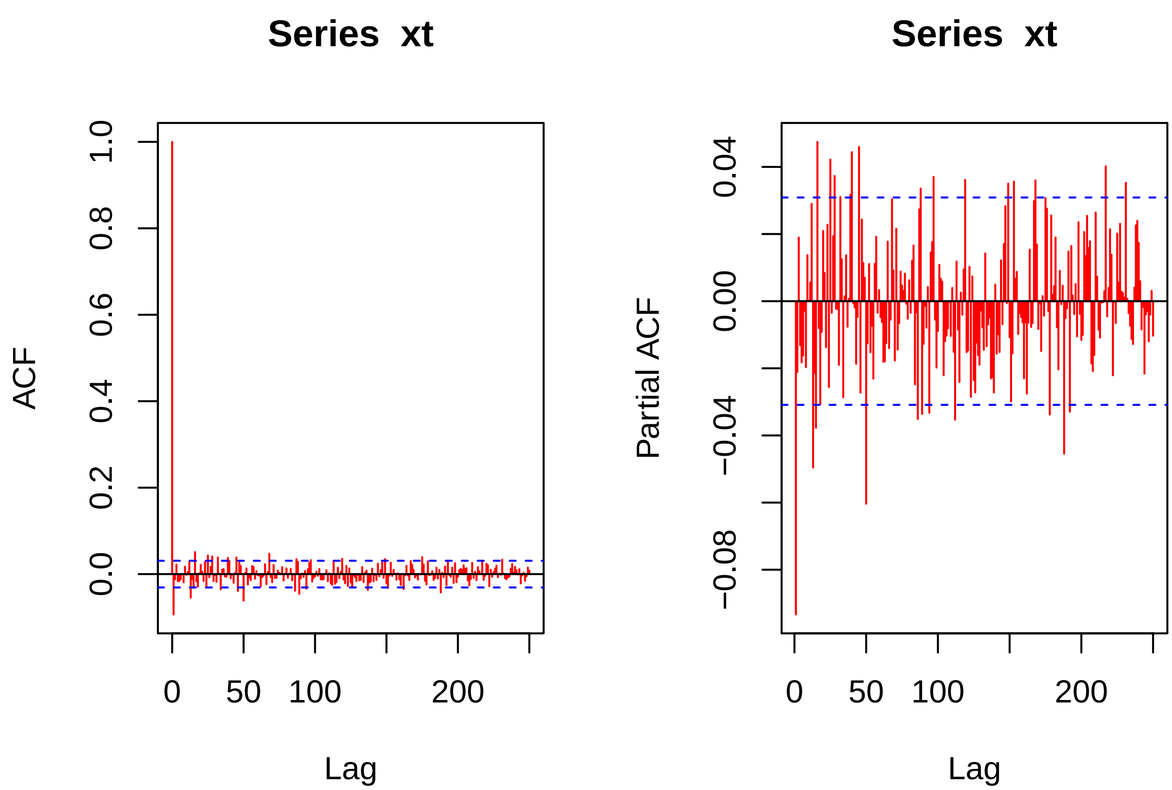
Deux pertes majeurs se dégage sur la période d'étude: une en 2009 et une autre en 2018. Il serait intéressant d'explorer les cause de ces volatilités atypiques.

Indépendance des observations

Nous étudions ici les corrélations entre les rendements observés dans le but de nous assurer du respect des hypothèses indépendance et d'indentique distribution des données faite en amont du modèle à estimer.

```
xt = na.omit(ts(data=data_cl$return))

par(mfrow=c(1,2))
acf(xt, lag.max =250, col='red' )
pacf(xt, lag.max =250, col='red' )
```



L'autocorrélogramme suggère une absence d'autocorrélation entre les rendements, laissant penser que ceux-ci pourraient être indépendants dans le temps.

```
library(chron)
```

Modélisation des Valeurs Extrêmes

Afin de pourvoir extimer les pertes extrêmes quotidiennes enregistrées sur les actions adobe, nous allons mettre en application la théorie des valeurs extrêmes. Des deux méthodes couramment utilisées dans le cadre des valeurs extrêmes, nous allons nous allons implémenter la méthode de la distribution des valeurs extrême (GEV). Cette méthode consiste à ajuster les valeurs maximales d'échantillons de rendements (L_t) extrêmes à une loi des valeurs extrêmes généralisée. Il est nécessaire pour ce faire de choisir une fréquence pour les blocs d'analyse. Nous considérons des blocs mensuels (20 jours), et de calculer les valeurs maximales de rendement pour chaque bloc. Ces valeurs maximales suivent une distribution qui sera ajustée pour estimer la queue de la distribution des extrêmes.

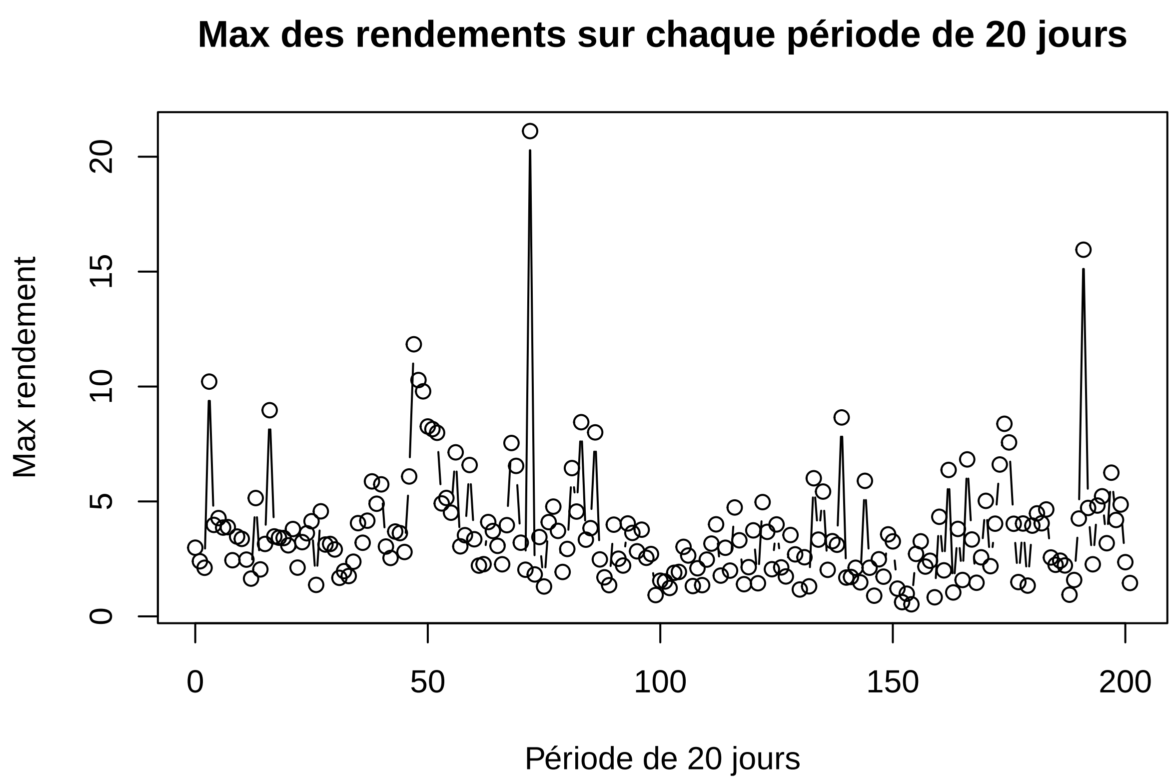
Echantillon retenue pour l'étude

Comme annocé plus haut, pour chaque mois on retient la perte maximale enregistrée. Chaque mois est un bloc de 20 jours correspondant aux jours d'ouverture de la bourse.

```
# Créer un identifiant de période de 10 jours
data_cl$period <- as.integer((1:nrow(data_cl) - 1) / 20)

# Calculer le maximum pour chaque trimestre
max_ten_day <- aggregate(return ~ period, data = data_cl, FUN = max)

# Tracer le résultat
plot(max_ten_day$period, max_ten_day$return, type = "b",
    main = "Max des rendements sur chaque période de 20 jours",
    xlab = "Période de 20 jours", ylab = "Max rendement")
```



L'échantillon obtenue par l'approche par block est constitué de … observations.

Estimation d'une GEV

```
## La librairie pour l'EVT
library(evd)
```

```
fit <- fgev(max_ten_day$return)
fit
```

```
##
## Call: fgev(x = max_ten_day$return)
## Deviance: 811.6442
##
## Estimates
##     loc   scale   shape
## 2.5310  1.3671  0.2074
##
## Standard Errors
##      loc    scale    shape
## 0.10913  0.08696  0.05760
##
## Optimization Information
##   Convergence: successful
##   Function Evaluations: 44
##   Gradient Evaluations: 13
```

Estimation et signifiactifité des paramètres

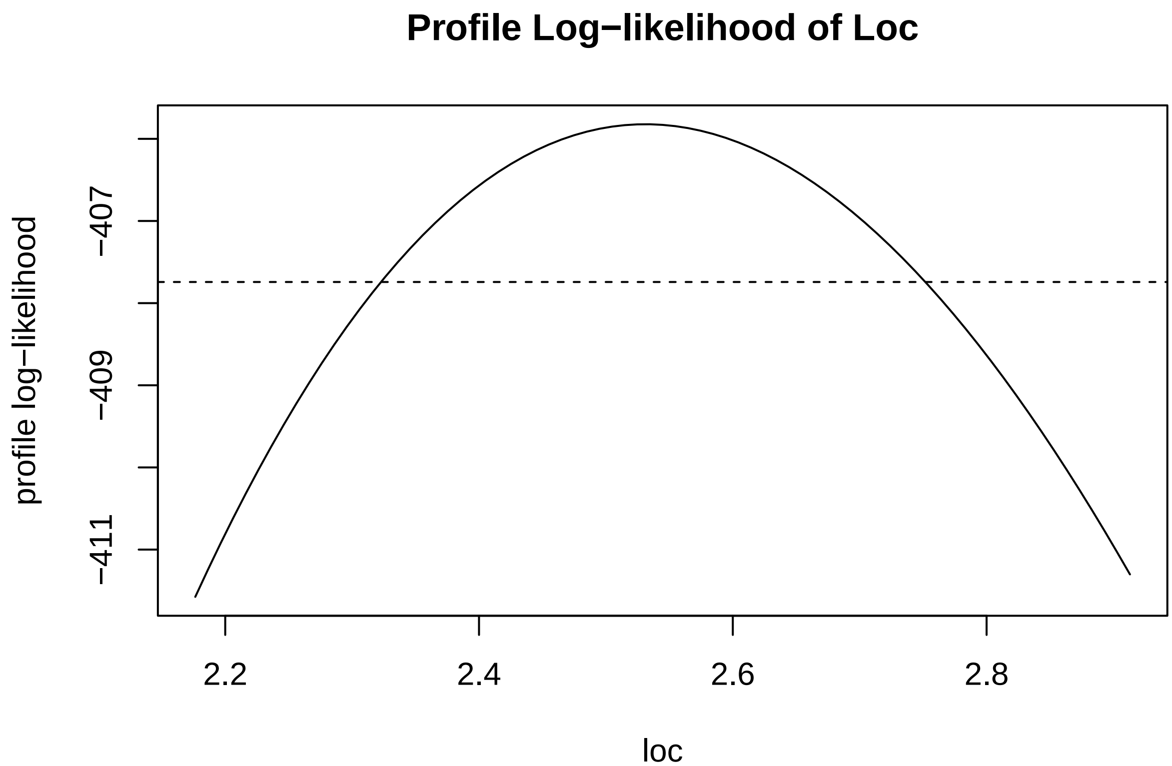
```
## Intervalles de confiance
confint(fit)##à la Wald
```

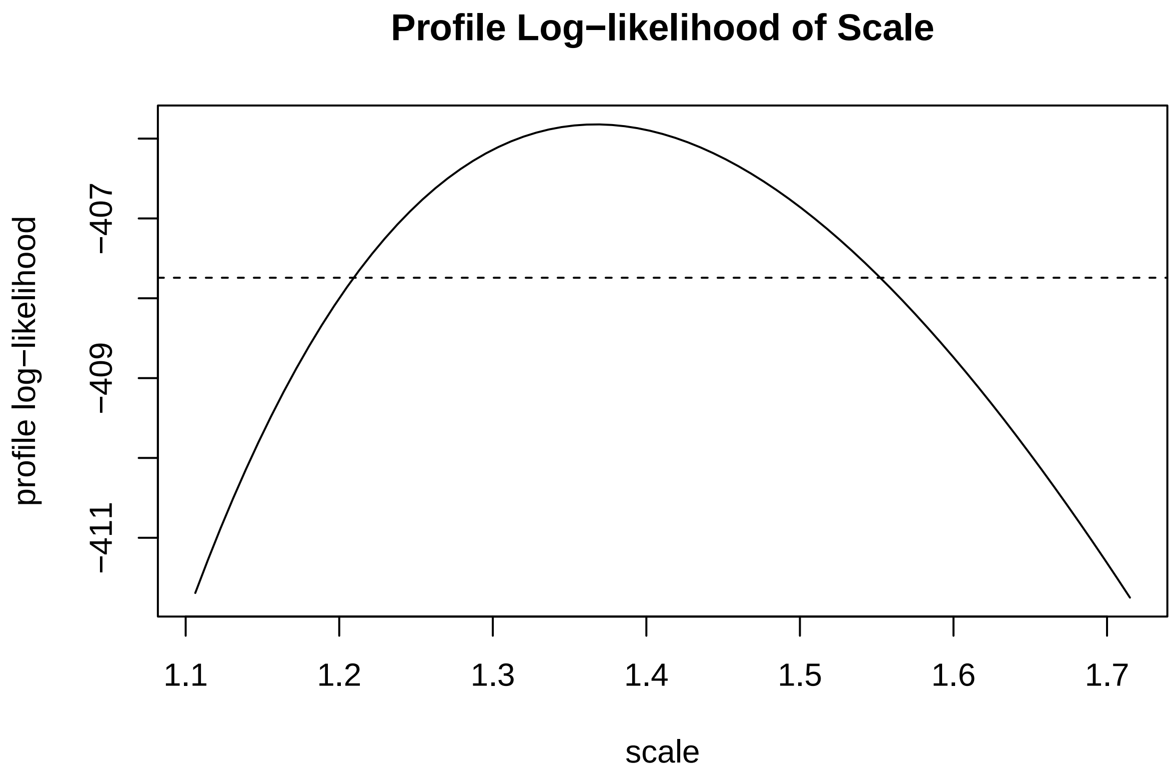
```
##            2.5 %    97.5 %
## loc   2.31715355 2.7449275
## scale 1.19667726 1.5375348
## shape 0.09453213 0.3203315
```

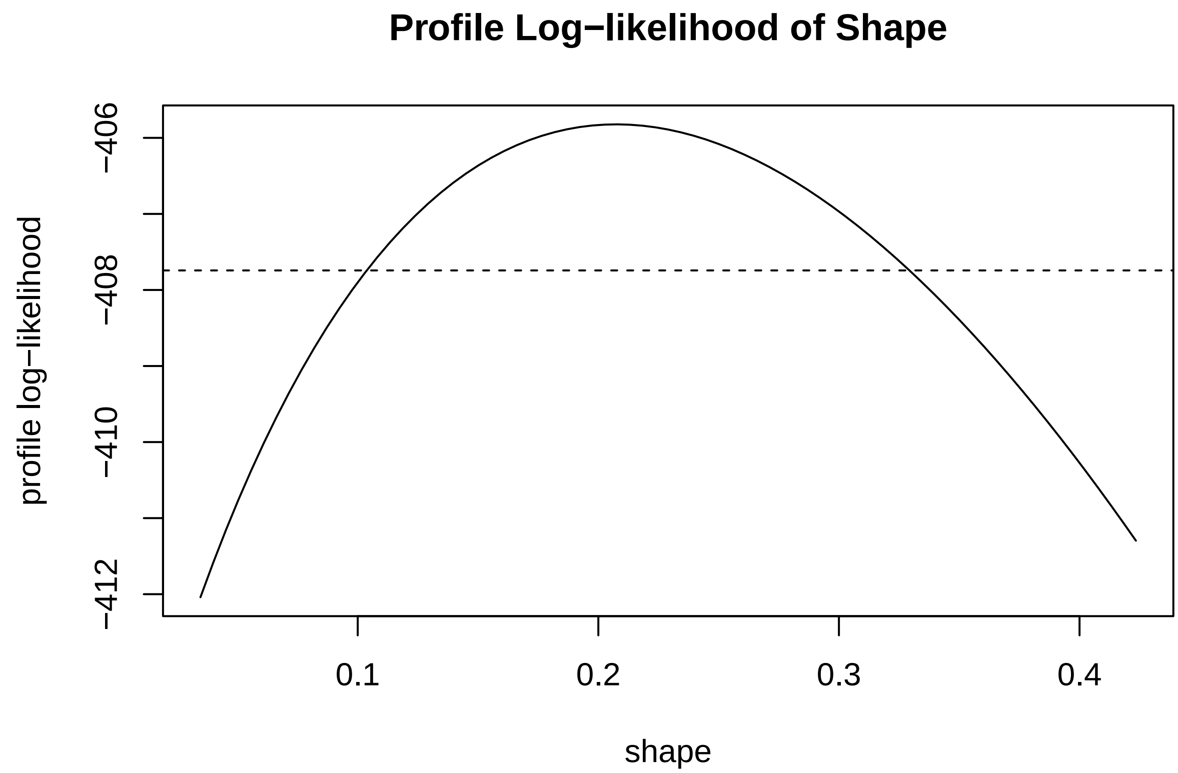
```
prof <- profile(fit)## via le profil de la vraisemblance
```

```
## [1] "profiling loc"
## [1] "profiling scale"
## [1] "profiling shape"
```

```
plot(prof)
```


Profile Log−likelihood of Loc
profile log−likelihood
−407
−409
−411
2.2
2.4
2.6
2.8
loc


Profile Log−likelihood of Scale
profile log−likelihood
−407
−409
−411
1.1
1.2
1.3
1.4
1.5
1.6
1.7
scale



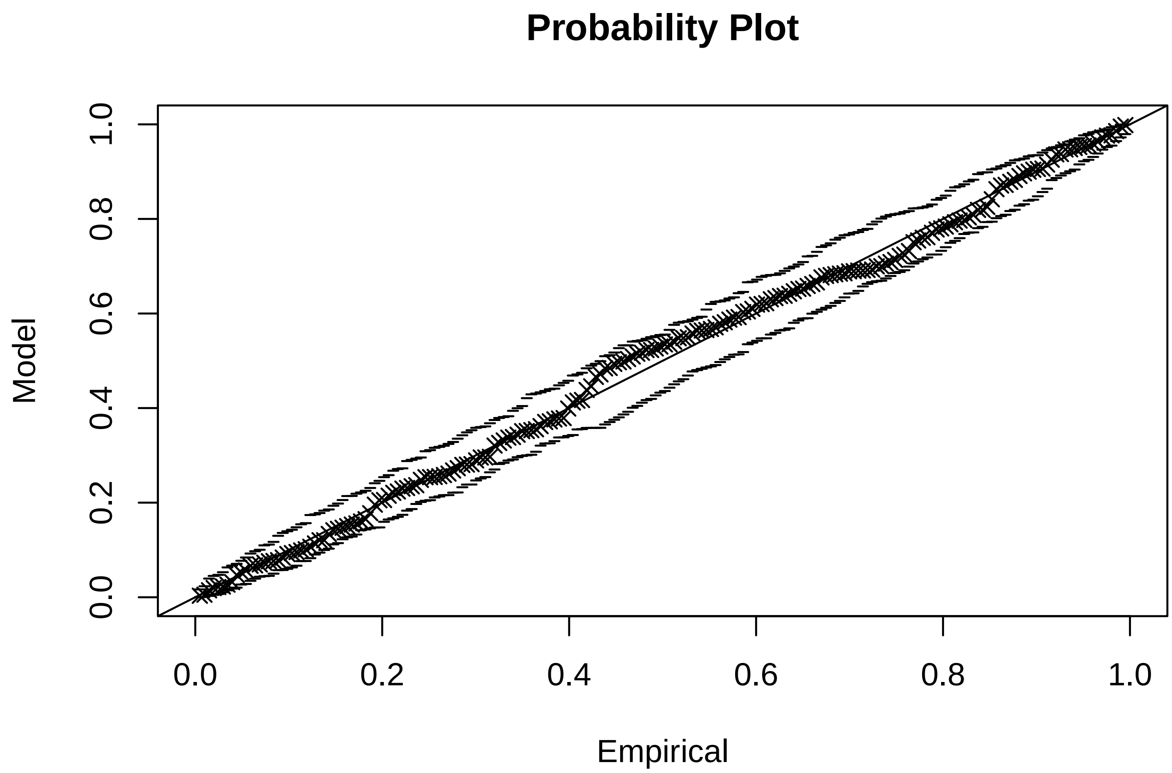

```
confint(prof)
```

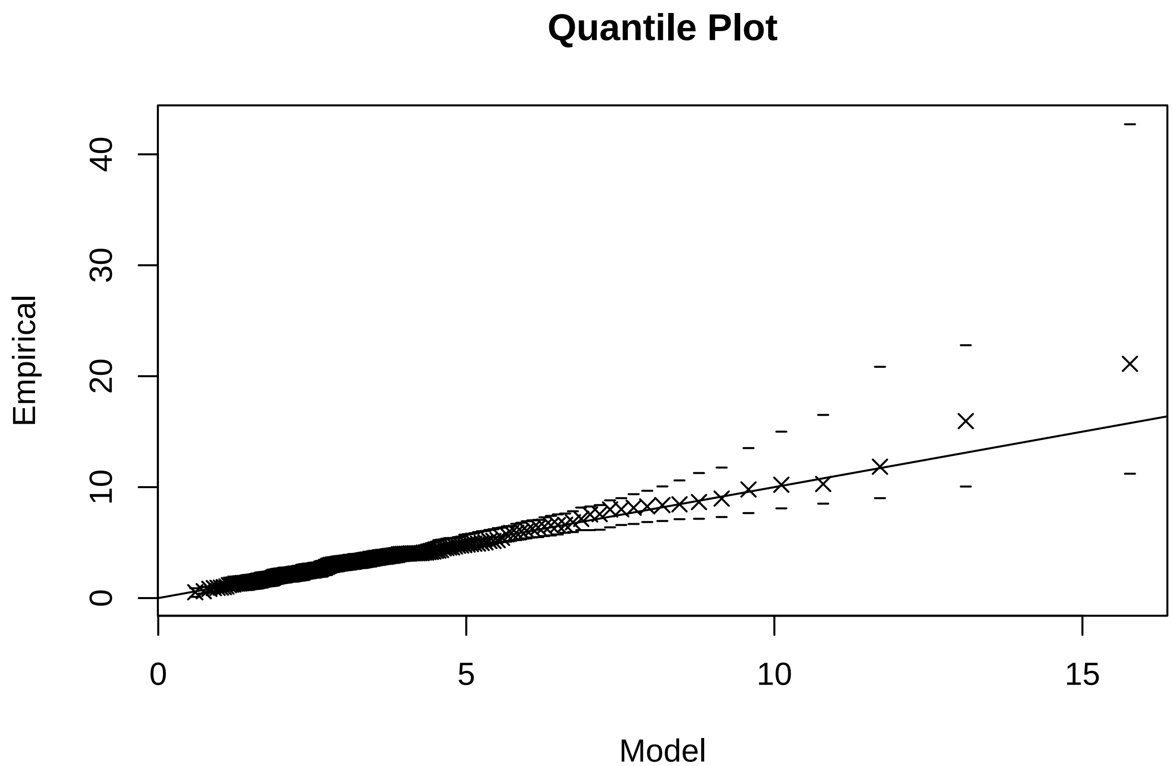
```
##            lower     upper
## loc   2.3231961 2.7515965
## scale 1.2097739 1.5520573
## shape 0.1039602 0.3291368
```

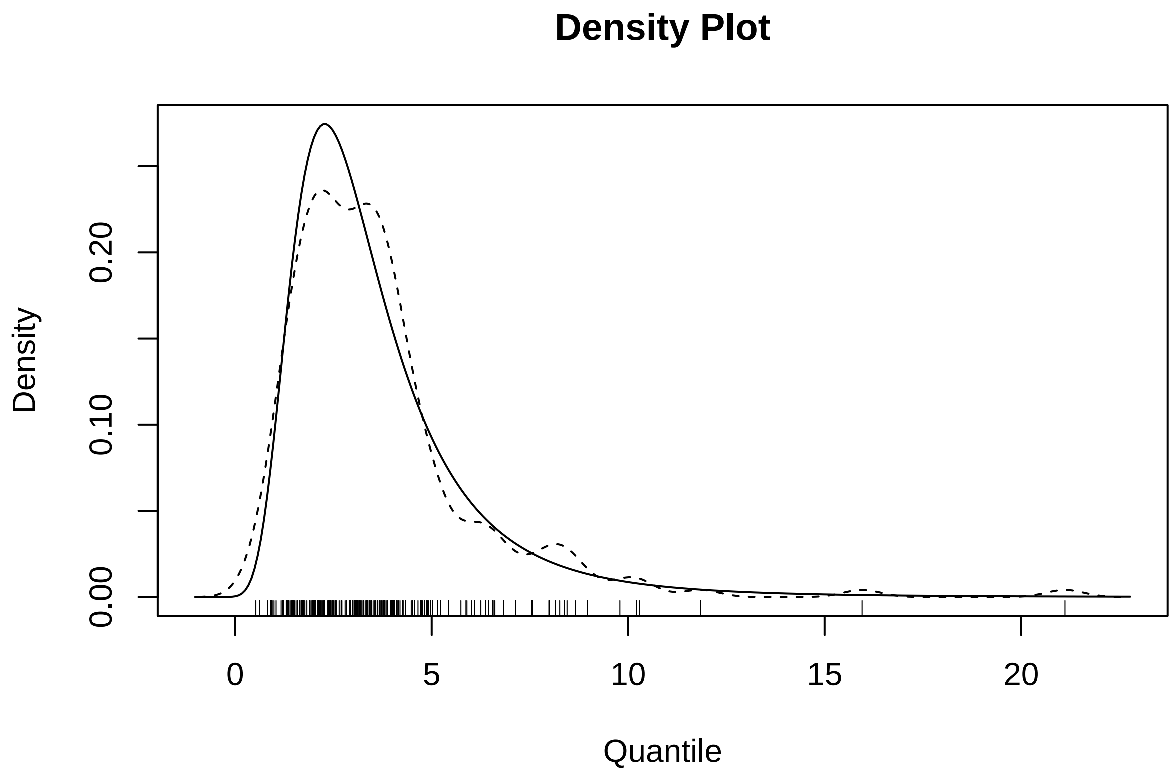
Validation du modèle

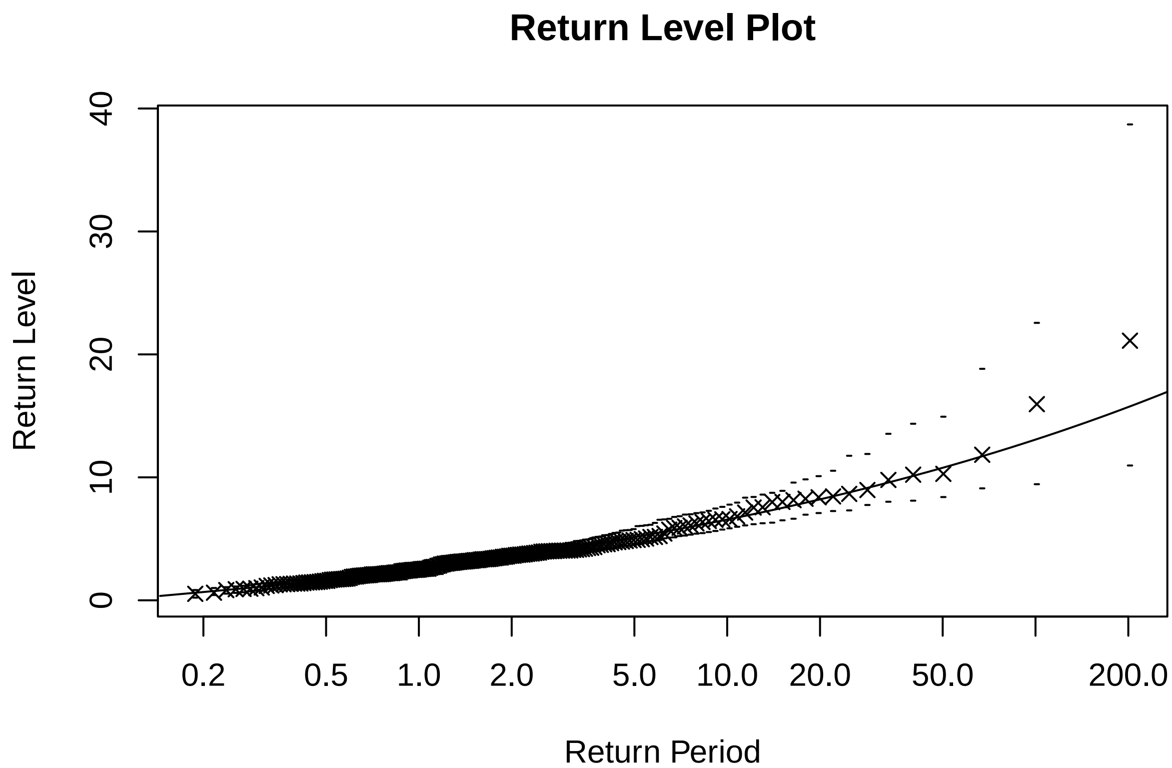
La pertinence de l'ajustement sera discutée graphiquement sur la base du P-P et Q-Q plot.

```
## Model checking
plot(fit)
```


Probability Plot
Model
Empirical
0.0
0.2
0.4
0.6
0.8
1.0


Quantile Plot
Empirical
Model
0
10
20
30
40
0
5
10
15


Density Plot
Density
0.00
0.10
0.20
0
5
10
15
20
Quantile


Return Level Plot
Return Level
0
10
20
30
40
0.2
0.5
1.0
2.0
5.0
10.0
20.0
50.0
200.0
Return Period

Conclusion