المندوبية السامية للتخطيط

ⵜⴰⵎⵓⵔⴰⵢⵜ ⵜⴰⵎⴰⵜⵜⴰⵢⵜ ⵏ ⵓⵙⵏⴰⴳⵏ

HAUT-COMMISSARIAT AU PLAN

Royaume Du Maroc
Haut-Commissariat Au Plan
Institut National De Statistique
et d'Economie Appliquée
INSEA

INSEA

RAPPORT : PROJET SERIES CHRONOLOGIQUES

SUJET

Utilisation de modèles ARCH-GARCH pour évaluer la volatilité des rendements boursiers : Etude empirique avec R

Réalisé par :

- Ahmed Bendrioua
- Mohamed Boutaghratine

1. Résumé :

Notre projet cible l'analyse des rendements de Apple en utilisant les modèles ARCH et GARCH en séries chronologiques sous le logiciel R. Après la collecte minutieuse des données, l'application de ces modèles a permis d'estimer la volatilité conditionnelle au fil du temps vu que le modèle ARCH sert à évaluer l'impact des chocs passés sur la volatilité, tandis que le modèle GARCH étend cette analyse en incorporant les volatilités passées. Les résultats ont révélé des tendances clés dans la dynamique de la volatilité de Apple, offrant des insights précieux pour la gestion des risques et la prise de décisions en matière d'investissement. Ces informations sont cruciales pour anticiper les périodes de volatilité accrue ou réduite, aidant ainsi les investisseurs à ajuster leurs stratégies. En conclusion, le projet contribue à notre compréhension des séries chronologiques financières, fournissant des outils avancés pour modéliser et gérer la volatilité dans un contexte financier dynamique. Cette recherche a des implications pratiques significatives pour les professionnels de la finance et enrichit le corpus de connaissances sur la modélisation de la volatilité dans les marchés financiers.

Contents

2. Liste des abréviations

1. GARCH : Generalized Autoregressive Conditional Heteroskedasticity
2. ARCH : Autoregressive Conditional Heteroskedasticity
3. PACF : Partial Autocorrelation Function
4. ACF : Autocorrelation Function
5. KPSS : Kwiatkowski-Phillips-Schmidt-Shin
6. ADF : Augmented Dickey-Fuller

Introduction générale

La volatilité, dans le contexte financier, représente l'amplitude des variations des prix d'un actif financier au fil du temps. Elle est cruciale pour les investisseurs et les analystes, car elle reflète le niveau de risque associé à un instrument financier. Au cours du temps, la volatilité des marchés peut connaître des fluctuations importantes, influencées par divers facteurs tels que les événements économiques, politiques et les dynamiques du marché.

En économétrie, la volatilité a été l'un de sujets de recherche le plus utilisés, et elle l'est toujours, l'un des sujets de recherche les plus actifs dans le domaine de la prévision économique en générale et de l'économétrie financière en particulier. L'une de caractéristique principale du processus de la volatilité est le fait qu'elle n'est pas observable. Les premiers travaux sur la volatilité (modèles ARCH/GARCH) ont été conçus pour fournir des estimations anticipatives ou de prévisions, en d'autre terme de la volatilité conditionnelle. Les modèles autorégressifs conditionnellement hétéroscédastiques (ARCH) ont été introduits par Engle (1982) et leurs extensions GARCH (ARCH généralisés) est due à Bollerslev (1986). Leurs caractérisations reposent essentiellement sur le concept de variance conditionnelle, qui ne dépend que du module des valeurs passées : l'effet sur la volatilité de la date présente des innovations passées positives et négatives est donc identique.

Le modèle ARCH, introduit par Robert Engle en 1982, postule que la variance conditionnelle des rendements dépend linéairement des carrés des erreurs passées. Cette approche permet de capturer l'hétéroscédasticité conditionnelle, c'est-à-dire la variation de la volatilité au fil du temps.

Le modèle GARCH, une extension introduite par Tim Bollerslev en 1986, ajoute une dimension supplémentaire en incluant les carrés des volatilités passées dans l'équation. Cette extension permet de mieux modéliser la persistance de la volatilité, ce qui est crucial pour anticiper les tendances futures.

Cette introduction explore donc la nature dynamique de la volatilité, son importance dans le domaine financier, et met en lumière les modèles ARCH et GARCH comme des outils essentiels pour estimer la volatilité conditionnelle des rendements financiers. Ces modèles offrent une perspective précieuse pour les investisseurs cherchant à mieux comprendre et anticiper les mouvements du marché.

Dans ce projet nous prévoyons explorer deux chapitres distincts. Le premier chapitre sera consacré à la partie théorique des modèles ARCH et GARCH, offrant ainsi une compréhension approfondie des principes sous-jacents à ces modèles, particulièrement dans le contexte des séries temporelles.

Dans le deuxième chapitre, notre attention se portera sur la partie pratique de notre étude. Nous nous attacherons spécifiquement à l'estimation de la volatilité des rendements boursiers d'Apple. Pour cette tâche, nous utiliserons le logiciel R comme principal outil d'analyse, en nous appuyant sur les modèles ARCH et GARCH. Cette analyse reposera sur une base de données couvrant la période de 2009 à 2018.

Chapitre 1 :

Processus ARCH / GARCH de la volatilité des rendements boursiers

1. Les modèle ARCH :

Le modèle ARCH (Autoregressive Conditional Heteroskedasticity) est un modèle économétrique utilisé pour modéliser la volatilité conditionnelle des séries temporelles financières. Il a été introduit par Robert Engle en 1982. Ce modèle est particulièrement utile pour capturer les variations de la volatilité au fil du temps, ce qui est fréquemment observé sur les marchés financiers.

Supposons que nous modélisons la variance d'une série $\boldsymbol{y_t}$. Le modèle ARCH(1) pour la série $\boldsymbol{y_t}$ est la variance au temps t conditionnellement à yt-1 est :

$$\boldsymbol{Var(\text{--}\, y_t / y_{t-1}) = \sigma_t^2 = \alpha_0 + \alpha_1 . {y_{t-1}}^2 \quad ; \; \alpha_0 \, , \alpha_1 \geq 0}$$

Note : La variance au temps t est liée à la valeur de la série au temps t–1. Une valeur relativement élevée de $\boldsymbol{y_{t-1}}$ entraîne une valeur de la variance relativement élevée au temps t. Cela signifie que la valeur de $\boldsymbol{y_t}$ est moins prévisible au temps t-1 que lors de périodes suivant une valeur relativement faible de $\boldsymbol{y_{t-1}}$.

$\boldsymbol{Si \;\; E(y_t) = 0}$:

$$\boldsymbol{y_t = \sigma_t . \varepsilon_t}$$

Avec : $\boldsymbol{\sigma_t = \sqrt{\alpha_0 + \alpha_1 . {y_{t-1}}^2} \quad et \quad \varepsilon_t \; \overrightarrow{iid} \;\; N(0, 1)}$

- **Généralisation mathématique du Modèle ARCH(P) :**

Un processus ARCH(P) est un processus pour lequel la variance au temps t est conditionnelle aux observations de P périodes précédentes, la version simple du modèle ARCH(P) s'exprime mathématiquement comme suit :

$$\boldsymbol{\sigma_t^2 = \alpha_0 + \sum_{i=1}^{P} \alpha_i \, . \, y_{t-i}^2}$$

2. Les modèles GARCH :

Le modèle GARCH(1,1) est une spécification particulière du modèle GARCH qui prend en compte la variance conditionnelle au temps t en fonction des erreurs passées et des volatilités passées. Sa formulation est la suivante :

$$\boldsymbol{\sigma_t^2 = w + \alpha_1 . {y_{t-1}}^2 + \beta_1 \, . \, \sigma_{t-1}^2}$$

- **Généralisation mathématique du Modèle GARCH (p, q) :**

La formulation générale du modèle GARCH(p,q) est donnée par :

$$\sigma_t^2 = w + \sum_{i=1}^{P} \alpha_i \cdot y_{t-i}^2 + \sum_{j=1}^{q} \beta_j \cdot \sigma_{t-j}^2$$

- **Choix du Modèle (ARCH ou GARCH) :**

Si vous ne constatez qu'une hétéroscédasticité conditionnelle sans une forte persistance, un modèle ARCH pourrait suffire. En revanche, si la volatilité semble avoir une composante persistante, un modèle GARCH peut être plus adapté.

3. Tests statistiques de détection de non-stationnarité

tests de détection de la présence de la racine unitaire sont couramment utilisés pour évaluer la non-stationnarité dans les séries temporelles. Ces tests sont conçus pour évaluer si une série temporelle possède une racine unitaire, ce qui impliquerait qu'elle est non stationnaire. Voici quelques-uns des tests statistiques

• Test de Dickey-Fuller (ADF) :
Le test de Dickey-Fuller est l'un des tests les plus classiques pour détecter la présence d'une racine unitaire dans une série temporelle. Il est basé sur une régression augmentée qui inclut les retards de la variable dépendante. La statistique de test est comparée à des valeurs critiques pour déterminer la stationnarité.
• Test de Kwiatkowski-Phillips-Schmidt-Shin (KPSS) : Contrairement aux tests ci-dessus qui testent l'hypothèse nulle de la présence d'une racine unitaire, le test KPSS teste l'hypothèse nulle de la stationnarité autour d'une tendance déterministe. Ainsi, il peut être utilisé pour tester la stationnarité contre la non-stationnarité.

Chapitre 2 :
Missions Réalisée

Objectifs :

- Explorer la performance boursière de Apple.
- Analyser les tendances boursières.
- Évaluer le prix des actions pour l'autocorrélation, la stationnarité, ou d'autres caractéristiques des séries temporelles.
- Examiner les rendements des actions de Apple pour évaluer les résultats potentiels.
- Évaluation de GARCH/ARCH pour la prévision des prix.
- Informer les décisions d'investissement basées sur l'analyse.

Resources:

- Dataset: Apple Stock Dataset on Yahoo’s finance website
- Programming Language: R

Libraries:

- Tidyquant: Tidy Quantitative Financial Analysis
- Timetk: A Tool Kit for Working with Time Series
- ggplot2: The Grammar of Graphics plot
- dplyr: Grammar of Data Manipulation
- rugarch: Univariate GARCH Models

1. Importation des bibliothèques sous R :

```
library(tidyquant)
library(timetk)
library(ggplot2)
library(dplyr)
library(rugarch)
```

Nous commençons par obtenir des données quantitatives au format 'tibble' à partir du site web financier de Yahoo en utilisant la fonction tq_get de la bibliothèque tidyquant :

```
apple <- tq_get("AAPL", from = "2009-01-01",to = "2018-03-01",get = "stock.prices")
```

Sample Output:

```
   symbol date        open  high   low close     volume adjusted
   <chr>  <date>     <dbl> <dbl> <dbl> <dbl>      <dbl>    <dbl>
 1 AAPL   2009-01-02  3.07  3.25  3.04  3.24  746015200     2.75
 2 AAPL   2009-01-05  3.33  3.43  3.31  3.38 1181608400     2.86
 3 AAPL   2009-01-06  3.43  3.47  3.30  3.32 1289310400     2.82
 4 AAPL   2009-01-07  3.28  3.30  3.22  3.25  753048800     2.76
 5 AAPL   2009-01-08  3.23  3.33  3.22  3.31  673500800     2.81
 6 AAPL   2009-01-09  3.33  3.34  3.22  3.23  546845600     2.74
 7 AAPL   2009-01-12  3.23  3.25  3.13  3.17  617716400     2.68
 8 AAPL   2009-01-13  3.15  3.20  3.08  3.13  798397600     2.66
 9 AAPL   2009-01-14  3.08  3.12  3.03  3.05 1021664000     2.58
10 AAPL   2009-01-15  2.88  3.00  2.86  2.98 1831634000     2.52
# ℹ 2,295 more rows
# ℹ Use `print(n = ...)` to see more rows
```

tibble [2,305 × 8]

Explication du tableau :

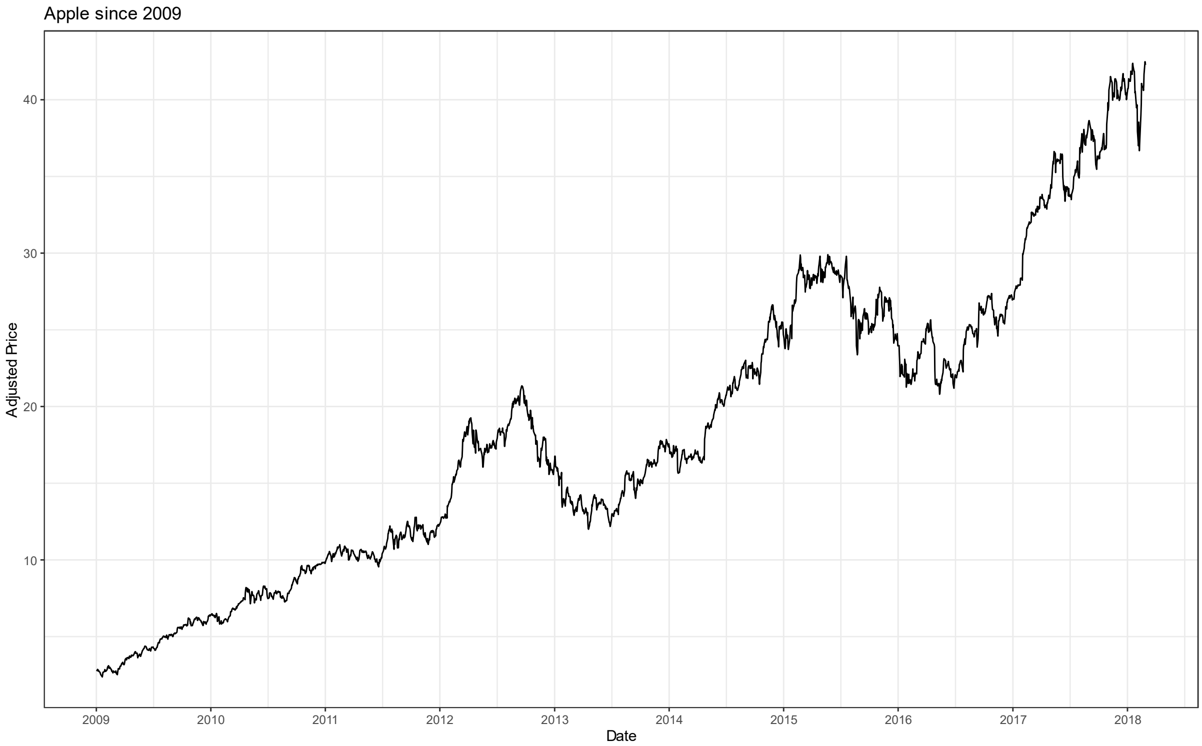
Ces colonnes fournissent des informations détaillées sur les prix des actions Apple et l'activité de trading :

- Date: The date of each trading day.
- Open: Day's starting price
- High: Intraday peak price
- Low: Lowest intraday price
- Close: End-of-day price
- Adjusted: The adjusted closing price, accounting for corporate actions such as dividends and stock splits
- Volume: The total number of shares traded on that day.

Maintenant, nous allons tracer le prix ajusté de Apple :

```
plt_adjusted <- apple %>%
  ggplot(aes(x = date, y = adjusted)) +
  geom_line() +
  ggtitle("Apple since 2009") +
  scale_x_date(date_breaks = "years", date_labels = "%Y") +
  labs(x = "Date", y = "Adjusted Price") +
  theme_bw()
print(plt_adjusted)
```

The plot:



Le prix de clôture ajusté (Adjusted close price) d'une action nous fournit toutes les informations dont nous avons besoin pour suivre l'évolution d'une action.

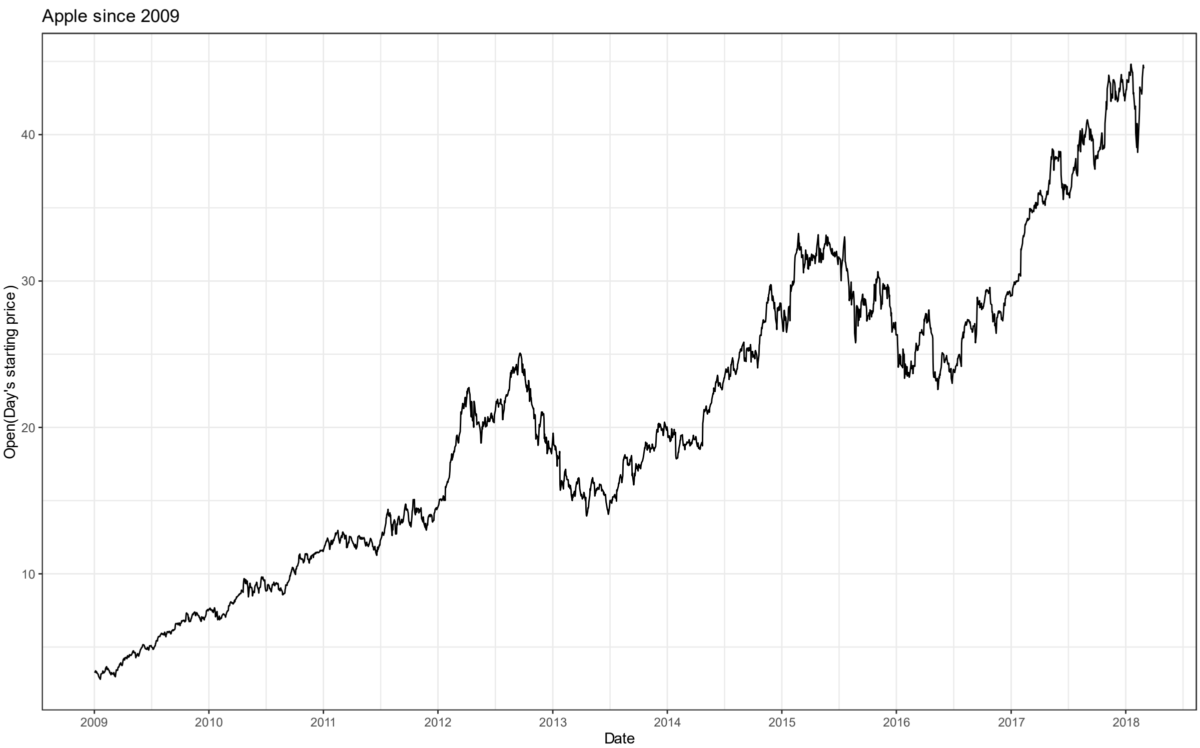
Observations:

- La tendance générale du prix de l'action Apple est généralement à la hausse.
- Des périodes de volatilité avec des augmentations et des baisses marquées du prix de l'action.

Maintenant on va traçer le prix d'ouverture (the open price) de Apple :

```
plt_open <- apple %>%
  ggplot(aes(x = date, y = close)) +
  geom_line() +
  ggtitle("Apple since 2009") +
  scale_x_date(date_breaks = "years", date_labels = "%Y") +
  labs(x = "Date", y = "Open(Day's starting price)") +
  theme_bw()
print(plt_open)
```

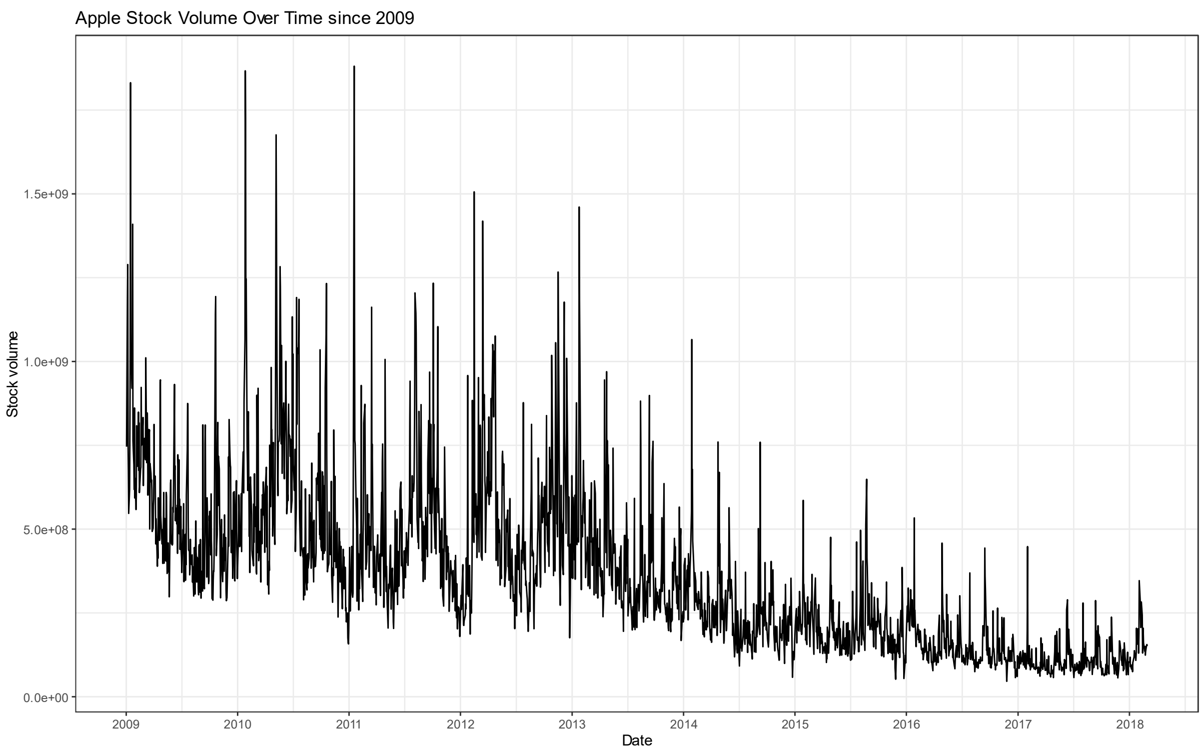
The plot:



Maintenant nous traçons le volume des transactions boursières au fil du temps :

```
plt_volume <- apple %>%
  ggplot(aes(x = date, y = volume)) +
  geom_line() +
  ggtitle("Apple Stock Volume Over Time since 2009") +
  scale_x_date(date_breaks = "years", date_labels = "%Y") +
  labs(x = "Date", y = "Stock volume") +
  theme_bw()
print(plt_volume)
```

The plot:



Observations:

- Trading volume saw occasional spikes, potentially influenced by major news like Apple releases or partnerships, The major spike occurred in late 2011.

- Trading volume has been relatively low in the last months. This suggests that there is less investor interest in Apple stock at that moment.

Analyse des rendements quotidiens:

Pourcentage de rendement quotidien = (Prix de clôture d'aujourd'hui - Prix de clôture d'hier) / Prix de clôture d'hier

$$r_t = \frac{p_t - p_{t-1}}{p_{t-1}} \approx \log(p_t) - \log(p_{t-1}).$$

Implémentation

```
apple_daily_returns <-
  tq_transmute(data=apple,select = adjusted,
               mutate_fun = periodReturn,
               period = "daily",
               col_rename = "aapl_returns")
```

Output:

```
# A tibble: 2,305 × 2
   date       aapl_returns
   <date>            <dbl>
 1 2009-01-02       0
 2 2009-01-05       0.0422
 3 2009-01-06      -0.0165
 4 2009-01-07      -0.0216
 5 2009-01-08       0.0186
 6 2009-01-09      -0.0229
 7 2009-01-12      -0.0212
 8 2009-01-13      -0.0107
 9 2009-01-14      -0.0271
10 2009-01-15      -0.0229
# ℹ 2,295 more rows
# ℹ Use `print(n = ...)` to see more rows
```

Statistiques descriptives pour les rendements quotidiens de Apple :

```
summary_apple_daily_returns <- summary(apple_daily_returns$aapl_returns)
standard_dev <- sd(apple_daily_returns$aapl_returns)

output
     Min.       1st Qu.      Median      Mean       3rd Qu.     Max.        sd.
  -0.1235580 -0.0069102  0.0009703  0.0013243  0.0103285  0.0887410   0.01659819
```

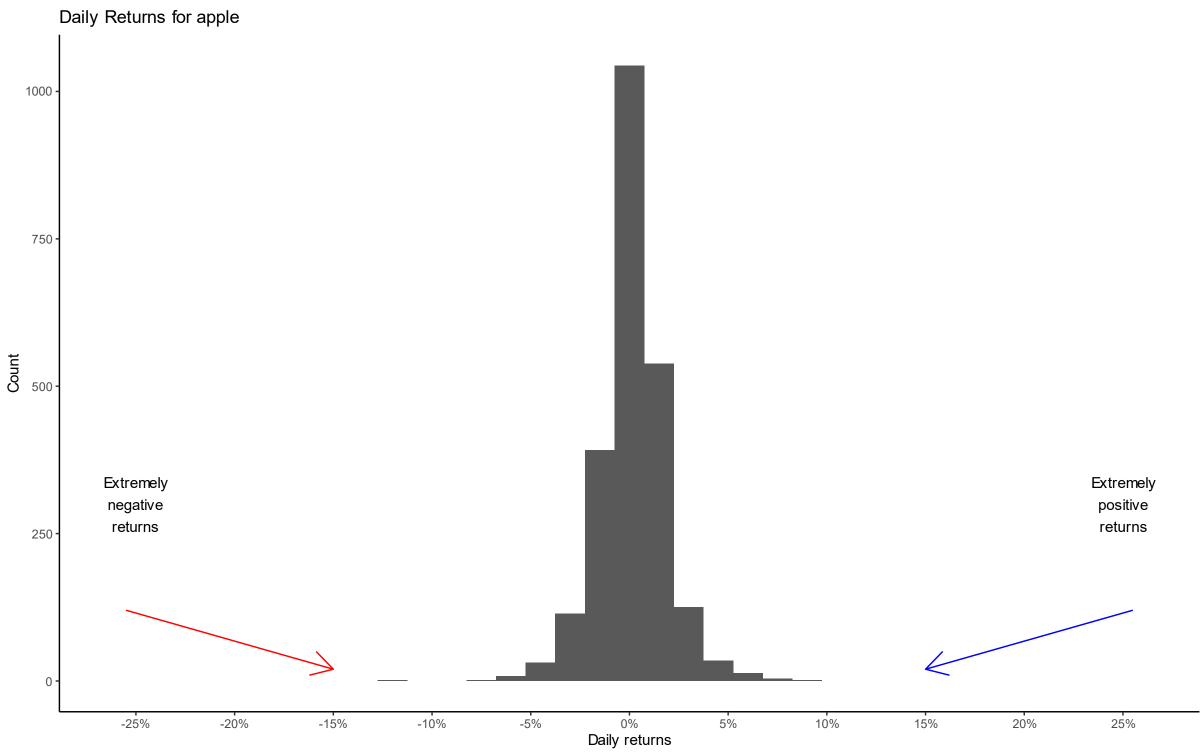
Observations:

- The average daily return is roughly 0.0013243, indicating a small positive daily gain.
- Data shows significant price swings, from -0.1235580 to 0.0887410.
- 25th to 75th percentiles: -0. 0069102 to 0.0103285, signifying typical daily returns.
- The daily return's standard deviation is about 0.01659819, indicating moderate volatility in daily price movements.

Visualisation des rendements quotidiens pour Apple :

```
adr_dist_plot <- apple_daily_returns %>%
   ggplot(aes(x = aapl_returns)) +
   geom_histogram(binwidth=0.015) +
   theme_classic() +
   labs(x = "Daily returns", y = "Count") +
   ggtitle("Daily Returns for apple") +
   scale_x_continuous(breaks = seq(-0.5,0.5,0.05),
                  labels = scales::percent)+
   annotate(geom = 'text', x = -0.25, y= 300, label = "Extremely\nnegative\nreturns")
+
  annotate(geom = 'segment', x = -0.255, xend = -0.15,  y = 120, yend = 20, color =
'red', arrow = arrow()) +
  annotate(geom = 'segment', x = 0.255, xend = 0.15,  y = 120,
        yend = 20, color = 'blue', arrow = arrow(type = "open")) +
  annotate(geom = 'text', x = 0.25, y = 300, label = "Extremely\npositive\nreturns")
```

The plot:



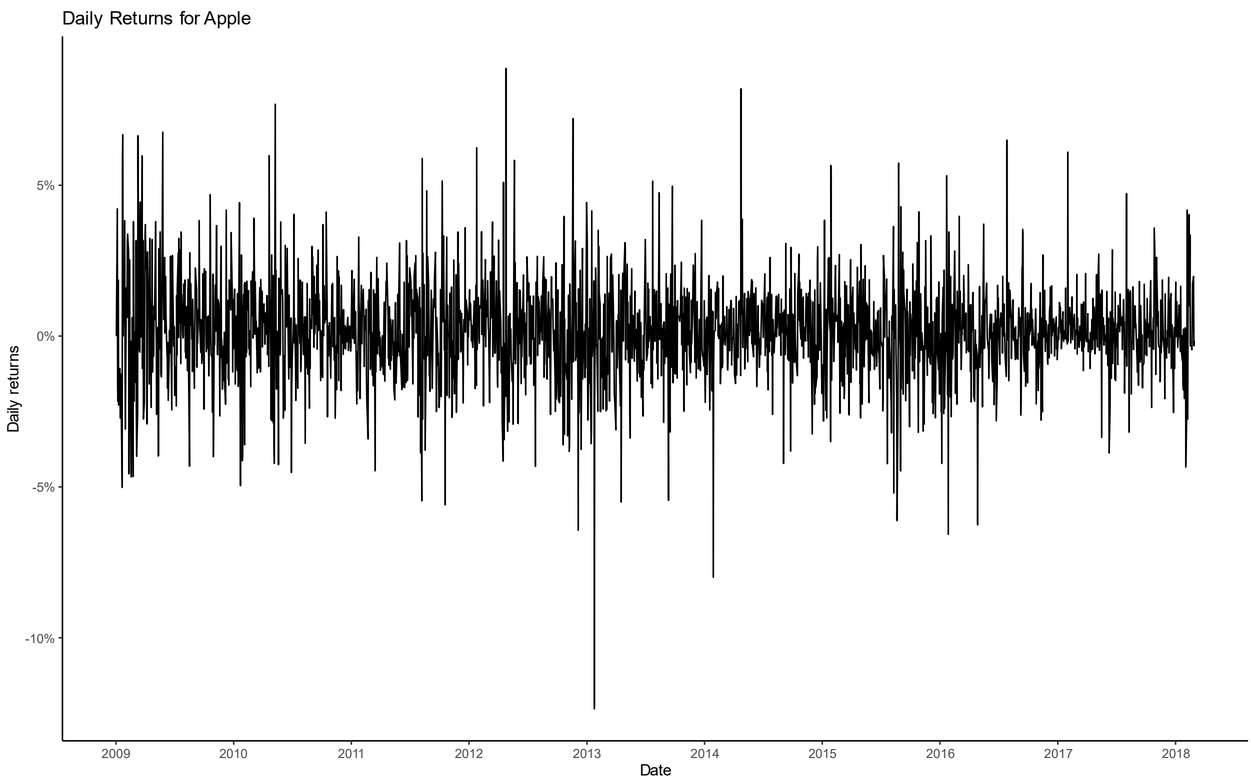
Observations:

- histogram shows that the distribution of daily returns is approximately normal
- In other words, Stock price can move +/- 5% on any given day but occasionally quite volatile.

Pour mieux comprendre, nous allons représenter graphiquement les rendements quotidiens :

```
plt_daily_returns <- apple_daily_returns %>%
  ggplot(aes(x = date, y = aapl_returns)) +
  geom_line() +
  theme_classic() +
  labs(x = "Date", y = "Daily returns") +
  ggtitle("Daily Returns for Apple") +
  scale_x_date(date_breaks = "years", date_labels = "%Y") +
  scale_y_continuous(breaks = seq(-0.5,0.5,0.05),
                     labels = scales::percent)
print(plt_daily_returns)
```

The plot:



Après avoir examiné le graphique des rendements quotidiens pour Apple, nous pouvons conclure que les rendements sont assez volatils et que l'action peut varier de +/- 5 % en une seule journée.

Les tests statistiques pour la stationnarité des séries temporelles et la détection de la tendance

(ADF&KPSS Tests)

```
adf_test <- adf.test(apple_daily_returns$aapl_returns)
kpss_test <- kpss.test(apple_daily_returns$aapl_returns,null = 'Level')
print(adf_test)
print(kpss_test)
```

Output

```
        Augmented Dickey-Fuller Test

data:  apple_daily_returns$aapl_returns
Dickey-Fuller = -13.11, Lag order = 13, p-value = 0.01
alternative hypothesis: stationary


        KPSS Test for Level Stationarity

data:  apple_daily_returns$aapl_returns
KPSS Level = 0.36403, Truncation lag parameter = 8, p-value = 0.09266
```

Output Interpretation:

- Augmented Dickey-Fuller (ADF)
  - Test Statistic (Dickey-Fuller): -13.11
  - p_value : 0.01, qui est inférieure à 0.05, ce qui montre que la stationarité de la série chronologiques des rendements.
- Kwiatkowski-Phillips-Schmidt-Shin (KPSS)
  - Test Statistic (KPSS Level): 0.36403
  - p_value : 0.09266, qui est supérieure à 0.05, ce qui montre que la stationarité de la série chronologiques des rendements.

GARCH(1,1) modélisation

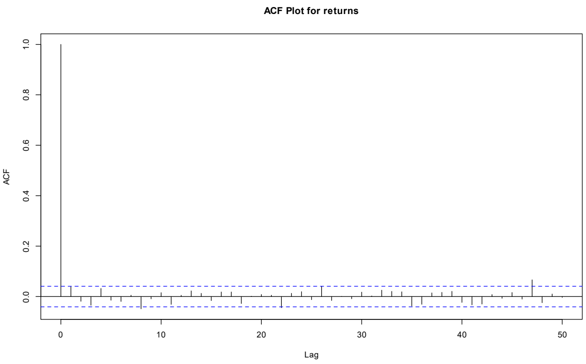
Model Selection:

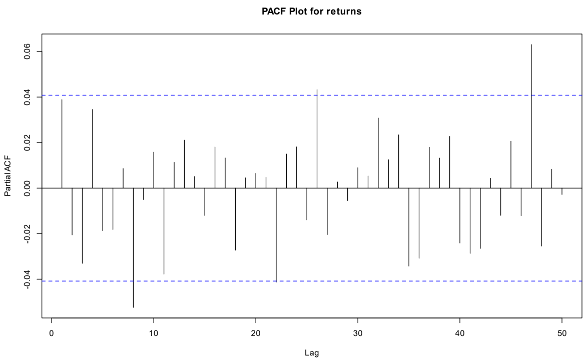
Déterminer les meilleures valeurs pour l'ordre autorégressif (p) et l'ordre de la moyenne mobile (q) dans un modèle GARCH implique d'analyser les graphiques de la fonction d'autocorrélation (ACF) et de la fonction d'autocorrélation partielle (PACF)

ACF and PACF of return

```
acf_result <- acf(apple_daily_returns$aapl_returns,
          lag.max = 50, plot = TRUE, main = "ACF Plot for returns")

pacf_result <- pacf(apple_daily_returns$aapl_returns,
          lag.max = 50, plot = TRUE, main = " PACF Plot for returns ")
```



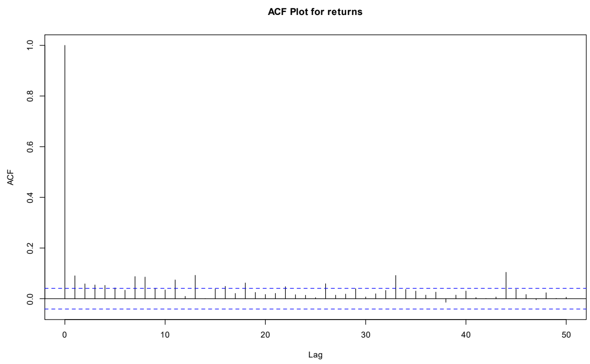


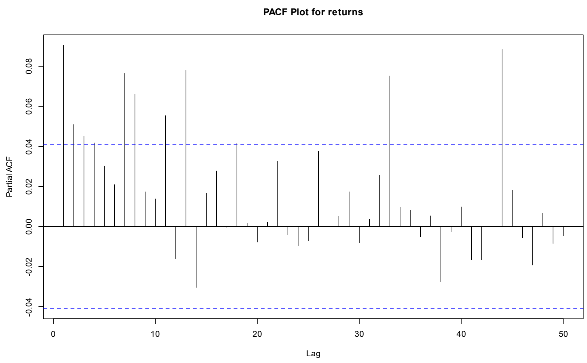
➲ Aucune corrélation n'est significative, donc la série de rendements semble être du bruit blanc.

ACF and PACF of squared return

```
acf_result <- acf(apple_daily_returns$aapl_returns**2,
          lag.max = 50, plot = TRUE, main = "ACF Plot for squared returns")

pacf_result <- pacf(apple_daily_returns$aapl_returns**2,
          lag.max = 50, plot = TRUE, main = " PACF Plot for squared  returns ")
```





- La fonction d'autocorrélation (ACF) de la série au carré suit un schéma ARMA en raison de l'atténuation à la fois dans l'ACF et la PACF. Cela suggère que les valeurs possibles des paramètres p et q sont (1,1), (1,2), (2,1) et (2,2).

Tester garch(1,1)

Nous utilisons la fonction ugarchspec() pour la spécification du modèle et ugarchfit() pour l'ajustement du modèle :

```
garch_spec <- ugarchspec(
          variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
          mean.model = list(armaOrder = c(0, 0), include.mean = TRUE),
          distribution.model = "norm"
)
```

```
garch_fit <- ugarchfit(spec = garch_spec, data = apple_daily_returns)
```

```
print(garch_fit)
```

Output:

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : sGARCH(1,1)
Mean Model      : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001776    0.000316   5.6177    0e+00
omega   0.000018    0.000004   4.4640    8e-06
alpha1  0.093747    0.018145   5.1665    0e+00
beta1   0.841994    0.028152  29.9084    0e+00

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001776    0.000342   5.1906 0.000000
omega   0.000018    0.000007   2.7412 0.006121
alpha1  0.093747    0.026007   3.6047 0.000312
beta1   0.841994    0.042418  19.8499 0.000000

LogLikelihood : 6269.18

Information Criteria
------------------------------------

Akaike       -5.4362
Bayes        -5.4262
Shibata      -5.4362
Hannan-Quinn -5.4325
```

Output Interpretation:

- mu (mean)
  - Estimate: 0.001776
  - Std. Error: 0.000316
  - t value: 5.6177
  - Pr(>|t|): 0e+00

  Le paramètre "mu" représente la moyenne de la série temporelle (rendement). Dans ce cas, la moyenne estimée est de 0, 001776 avec un écart-type de 0,000316. La valeur-t (t-value) et la valeur-p (p-value) aident à évaluer la significativité de l'estimation. Une faible valeur-p (généralement inférieure à 0,05) suggère que l'estimation de la moyenne est statistiquement significative.

- omega (constant):
  - Estimate: 0.000018
  - Std. Error: 0.000004
  - t value: 4.4640
  - Pr(>|t|): 8e-06

  Le paramètre "omega" est la constante dans le modèle GARCH. Il représente la moyenne à long terme des chocs au carré (volatilité).

- alpha1 (ARCH Coefficient):
  - Estimate: 0.093747
  - Std. Error: 0.018145
  - t value: 5.1665
  - Pr(>|t|): 0e+00

  Le paramètre "alpha1" est le coefficient des résidus quadratiques retardés dans la partie ARCH (Hétéroscédasticité Conditionnelle Autorégressive) du modèle. Il mesure l'impact des chocs quadratiques passés sur la volatilité actuelle. Une valeur élevée de la statistique t et une faible valeur p indiquent une signification statistique.

- ## beta1 (GARCH Coefficient):
  - Estimate: 0.841994
  - Std. Error: 0.028152
  - t value: 29.9084
  - Pr(>|t|): 0e+00

  Le paramètre "beta1" est le coefficient de la variance conditionnelle retardée dans la partie GARCH du modèle. Il mesure l'impact de la variance conditionnelle passée sur la volatilité actuelle. La valeur-t extrêmement élevée et la faible valeur-p suggèrent une forte significativité.

Both GARCH Coefficient and ARCH Coefficient are significant and needed (il s'agit bien d'un model garch).

Les coefficients GARCH et ARCH sont tous deux significatifs et nécessaires (il s'agit bien d'un modèle GARCH).

Tester garch(1,2)

Nous allons modifier la spécification du modèle :

```
garch_spec <- ugarchspec(
        variance.model = list(model = "sGARCH", garchOrder = c(1, 2)),
        mean.model = list(armaOrder = c(0, 0), include.mean = TRUE),
        distribution.model = "norm"
)
```

Output:

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : sGARCH(1,2)
Mean Model      : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001752    0.000316   5.5359 0.000000
omega   0.000025    0.000006   4.4829 0.000007
alpha1  0.133181    0.025384   5.2467 0.000000
beta1   0.308606    0.155162   1.9889 0.046709
beta2   0.470226    0.142146   3.3081 0.000939

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001752    0.000350   5.0031 0.000001
omega   0.000025    0.000009   2.8847 0.003918
alpha1  0.133181    0.036956   3.6038 0.000314
beta1   0.308606    0.236775   1.3034 0.192449
beta2   0.470226    0.221061   2.1271 0.033409

LogLikelihood : 6271.901

Information Criteria
------------------------------------

Akaike       -5.4377
Bayes        -5.4252
Shibata      -5.4377
Hannan-Quinn -5.4331
```

Output Interpretation:

- Beta1 et beta2 sont significatives ce que signifie que q=2 .

Tester garch(2,1)

Nous allons modifier la spécification du modèle, puis ajuster le modèle et l'afficher.

```
garch_spec <- ugarchspec(
        variance.model = list(model = "sGARCH", garchOrder = c(2, 1)),
        mean.model = list(armaOrder = c(0, 0), include.mean = TRUE),
        distribution.model = "norm"
)
```

Output:

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : sGARCH(2,1)
Mean Model      : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001776    0.000316   5.6182 0.000000
omega   0.000018    0.000005   3.6853 0.000228
alpha1  0.094169    0.025567   3.6833 0.000230
alpha2  0.000000    0.031419   0.0000 1.000000
beta1   0.841344    0.036054  23.3355 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001776    0.000343   5.1847 0.000000
omega   0.000018    0.000008   2.3224 0.020209
alpha1  0.094169    0.030953   3.0423 0.002348
alpha2  0.000000    0.039240   0.0000 1.000000
beta1   0.841344    0.055276  15.2208 0.000000

LogLikelihood : 6269.506

Information Criteria
------------------------------------

Akaike       -5.4356
Bayes        -5.4231
Shibata      -5.4356
Hannan-Quinn -5.4310
```

Output Interpretation:

Nous remarquons que alpha2 est égal à 0, ce qui signifie que p=1 et non pas 2.

Conclusion

Nous pouvons comparer les critères AIC, BIC et la vraisemblance des différents modèles. Le meilleur modèle est GARCH(1,2) car il a la plus grande valeur de vraisemblance et les valeurs les plus basses d'AIC et de BIC

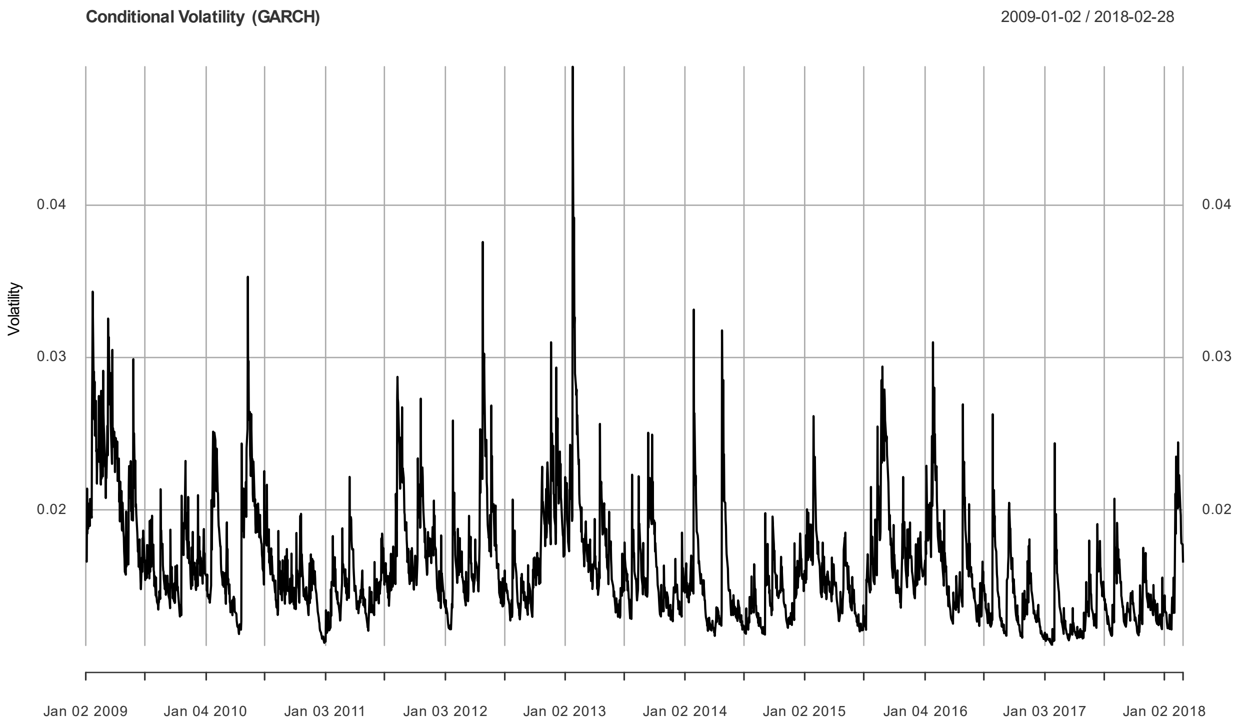
Extraire les valeurs de volatilité ajustée

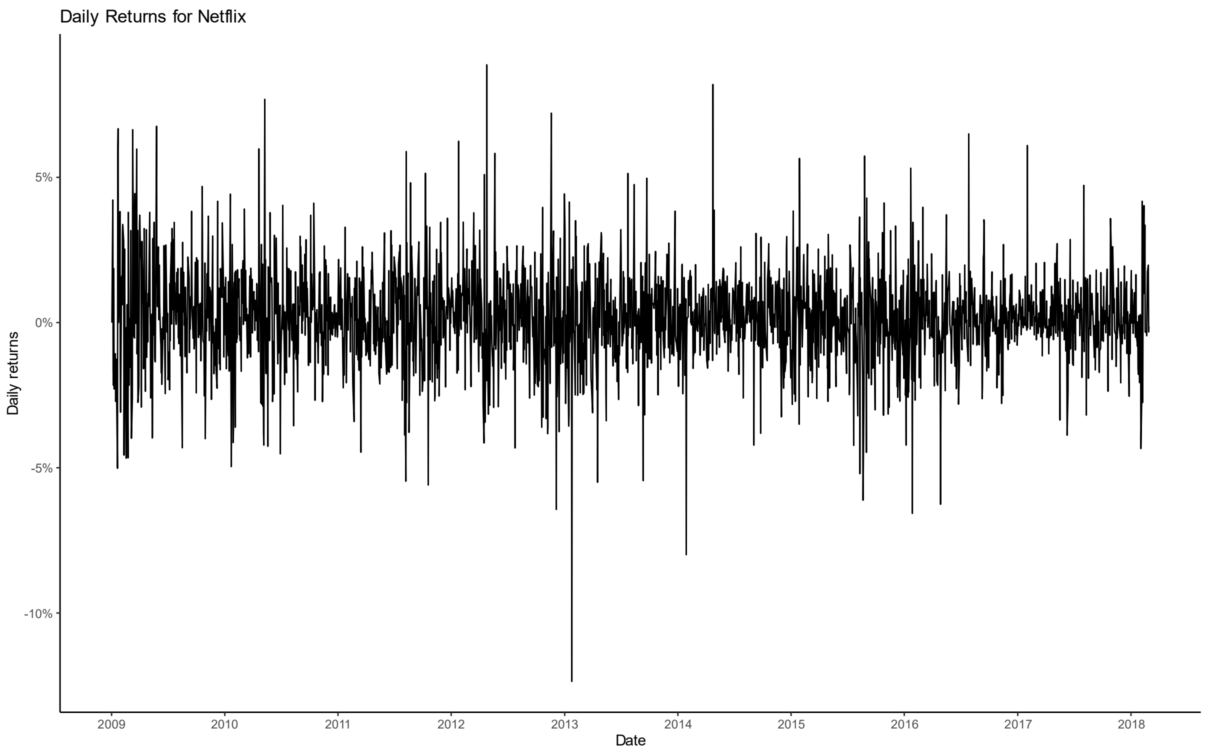
```
fitted_volatility<-sigma(garch_fit)
print(fitted_volatility)
```

```
           m.c.seq.row..seq.n...seq.col..drop...FALSE.
2009-01-02                                  0.01660012
2009-01-05                                  0.01660012
2009-01-06                                  0.02139256
2009-01-07                                  0.01844397
2009-01-08                                  0.02044407
2009-01-09                                  0.01875267
2009-01-12                                  0.02027050
2009-01-13                                  0.01968177
2009-01-14                                  0.01893498
2009-01-15                                  0.02071288
       ...
2018-02-14                                  0.02020339
2018-02-15                                  0.02165108
2018-02-16                                  0.02228377
2018-02-20                                  0.02005097
2018-02-21                                  0.01965045
2018-02-22                                  0.01839992
2018-02-23                                  0.01780295
2018-02-26                                  0.01773861
2018-02-27                                  0.01773306
2018-02-28                                  0.01653467
```

Créer un graphique en ligne de la volatilité conditionnelle.

```
fitted_volatility_plot<-plot(fitted_volatility, type = "l", col = "#000000", lwd = 2,
    main = "Conditional Volatility (GARCH)",
    xlab = "Time", ylab = "Volatility")
print(fitted_volatility_plot)
```





Prevision de la volatilité pour les 20j suivants:

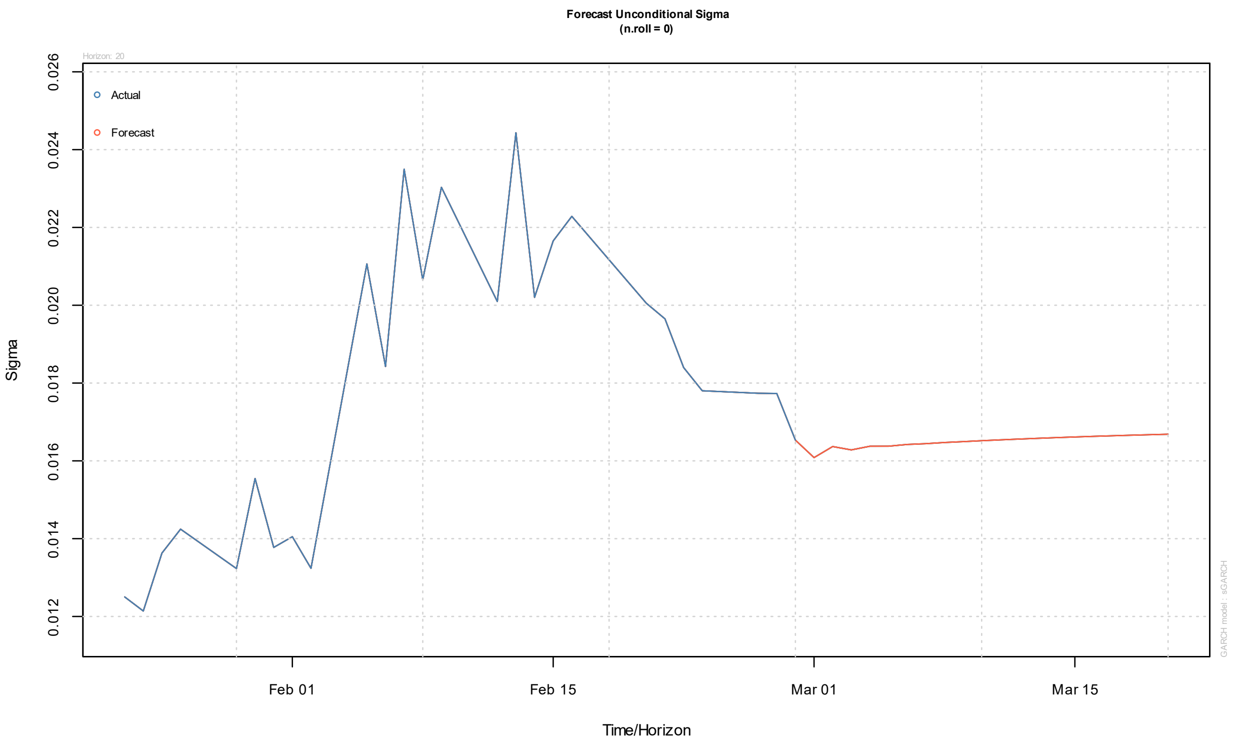
```
n_forecast <- 20
garch_forecast <- ugarchforecast(garch_fit, n.ahead = n_forecast)
print(garch_forecast)
```

```
*------------------------------------*
*       GARCH Model Forecast         *
*------------------------------------*
Model: sGARCH
Horizon: 20
Roll Steps: 0
Out of Sample: 0

0-roll forecast [T0=2018-02-28]:
       Series   Sigma
T+1  0.001752 0.01608
T+2  0.001752 0.01637
T+3  0.001752 0.01628
T+4  0.001752 0.01638
T+5  0.001752 0.01638
T+6  0.001752 0.01642
T+7  0.001752 0.01644
T+8  0.001752 0.01647
T+9  0.001752 0.01649
T+10 0.001752 0.01652
T+11 0.001752 0.01654
T+12 0.001752 0.01656
T+13 0.001752 0.01658
T+14 0.001752 0.01660
T+15 0.001752 0.01661
T+16 0.001752 0.01663
T+17 0.001752 0.01664
T+18 0.001752 0.01666
T+19 0.001752 0.01667
T+20 0.001752 0.01668
```

Créer un graphique en ligne pour la volatilité prévue :

```
Plot(garch_forecast)
```



Les différents graphiques de garch_fit suivants indiquent à quel point notre modèle est performant.

```
plot(garch_fit,which="all")
```