

Master Data Science and Big Data

Processus et application ARCH et GARCH

Groupe:

*Ghanem Abdelghani, Ouifak Hafsa, Adil BAHAJ,
Guendoul Oumaima, Zainab Elmadhouni*

Le 26-02-2019 _ Ensias Rabat

Abstract

Forecasts are of great importance for companies wishing to anticipate their activities, their needs and the means to implement to satisfy customer demand. The ARCHGARCH approach is introduced to avoid the major gap in ARMA models. This approach is a powerful tool for dealing with non-linear phenomena and unpredictable or volatile movements in the financial market. More specifically, they make it possible to present and model financial series.

Résumé

Les prévisions ont une grande importance pour les entreprises désireuses d'anticiper leurs activités, leurs besoins et les moyens à mettre en oeuvre pour satisfaire la demande des clients. L'approche ARCHGARCH est introduite pour éviter la lacune majeure des modèles ARMA. Cette approche est un outil puissant pour traiter les phénomènes non linéaires et les mouvements imprévisibles ou volatils du marché financier. Plus précisément, elles permettent de présenter et modéliser les séries financières.

Sommaire

Introduction.....	6
Chapitre 1 : Contexte général	7
I. Cadre de travail.....	7
II. Cahier de charges.....	7
III. Planning de travail	7
1. Diagramme de Gantt.....	7
Chapitre 2: Préliminaires et définitions	11
I. Définitions.....	11
1. Processus aléatoire	11
2. Volatilité	11
3. Stationnarité.....	12
4. Bruit blanc	12
5. Modèle linéaire	13
6. Modèle ARMA	13
7. Hétéroscédasticité.....	14
Chapitre 3 : Modèles SARIMA, ARCH et GARCH	15
I. Modèle autorégressif moyenne mobile intégré saisonnier SARIMA	15
1. Définition	15
2. Estimation d'un SARIMA	15
II. Modèles Autorégressifs Conditionnellement Hétéroscédastiques ARCH.....	16
1. Définition	16
III. Modèle ARCH généralisé GARCH	17
1. Modèle GARCH(p, q)	17
2. Modèle GARCH(1, 1)	18
Chapitre 4 : Simulation.....	20
I. Outils de simulation.....	20
1. Logiciel et Langage	20
2. Packages principaux	20
I. Simulation ARCH.....	21
1. Simulation du modèle ARCH(1).....	21
II. Simulation et application GARCH.....	27
1. Simulation de garch(2,1) et garch(1,1).....	27

2. Backtesting le modèle de risque et prévision de la société APPLE	29
Chapitre 5 : Conception et réalisation d'un programme de prédiction des valeurs d'une série temporelles.....	35
I. Diagramme d'activité	35
II. Logique du programme.....	35
III. Exercice d'application	36
1. Problème	36
2. Solution	36
Conclusion	46
Références.....	47

Liste des figures

Figure 1 : Diagramme de Gantt	10
Figure 2 : Figure des rendements.....	22
Figure 3 : Figure des variances	23
Figure 4 : Figure des series et de sigma conditionnelle	23
Figure 5 : Figure des différents rendements	25
Figure 6 : Figure des rendements.....	27
Figure 7 : Observations au cour du temps	28
Figure 8 : Séries simulées et volatilité conditionnelle.....	29
Figure 9: Closing price de APPLE	30
Figure 10 : Rendements mensuels composés de Apple.....	31
Figure 11: Séries avec 1% de risque	33
Figure 12: Diagramme d'activité du programme réalisé	35
Figure13 : Courbe de prédictions SARIMA au cas de stationnarité	42
Figure 14: Courbe de prédictions SARIMA au cas de non stationnarité.....	43
Figure 15:Prédiction de la période 15 avec GARCH	44
Figure 16: Prédiction ARIMA(0,0,0).....	45

Liste des tableaux

Tableau 1 : Différentes tâches réalisées pendant le projet	9
Tableau 2: Statistiques descriptives	37
Tableau 3: Test de normalité	37
Tableau 4: Test de corrélation	38
Tableau 5: Test de l'effet de ARCH et de stationnarité.....	40
Tableau 6: Prédictions de la deuxième feuille de données	41
Tableau 7: Prédictions de la première feuille de données.....	41

Introduction

Toutes les sociétés du monde s'intéressent à améliorer son business et prédire les changements à venir pour se préparer et avancer son travail, d'où le besoin et l'apparition des modèles de prédictions comme ARMA ou autres qui s'intéressent aux données et modèles stationnaires en variance. Mais ça reste une lacune quand on parle des problèmes dont l'incertitude portant sur les prix spéculatifs varie au cours du temps. L'une des solutions proposée est celle de ENGLE qui a introduit le Model de l'hétéroscédasticité conditionnelle autorégressive ARCH pour caractériser ces variances variables

Dans ce projet on va essayer d'introduire le model ARCH ainsi que le ARCH généralisé, et on va par la suite essayer de faire une application sur des données fournies par "Yahoo" sur une société connue, ainsi que simuler des données en format Excel, réaliser le processus de prédictions et générer des fichiers Excel et des courbes représentatives.

Chapitre 1 : Contexte général

I. Cadre de travail

Ce travail fait partie du programme du module de probabilités, pour finaliser le module qui fait partie du premier semestre du master « Data Science and Big Data » de l'école nationale supérieure d'informatique et d'analyse des systèmes de Rabat.

Le projet s'intéresse aux modèles de prédictions de volatilité et intitulé « Processus et application ARCH et GARCH ».

II. Cahier de charges

Le travail demandé est d'abord de faire une synthèse autour des modèles SARIMA, ARCH et GARCH, les simuler et enfin les mettre en pratique.

Travail principal :

- Synthèse sur les trois modèles
- Apprendre à utiliser R Studio
- Apprendre les fonctionnalités des packages principales simplifiant l'usage de ses modèles
- Utiliser les modèles pour des études analytiques sur des données du marché étudié

Objectifs :

- Analyser les données
- Simuler des modèles
- Prédire les volatilités futures

III. Planning de travail

1. Diagramme de Gantt

Le tableau suivant montre les différentes tâches réalisées pendant ce projet, les figures suivantes illustrent les ressources et la répartition des tâches entre le groupe dans la phase de réalisation.

Tableau 1 : Différentes tâches réalisées pendant le projet

Tâches		
Nom	Date de début	Date de fin
Formation sur les notions nécessaires	01/02/19	07/02/19
Etude et simulation ARCH	08/02/19	19/02/19
Etude et simulation GARCH	08/02/19	16/02/19
Application 2.Backtesting le modèle de risque et prévision de APPLE	15/02/19	18/02/19
Analyse des données excel avec R et réponse à l'exercice	19/02/19	25/02/19
Réalisation du rapport	07/02/19	26/02/19

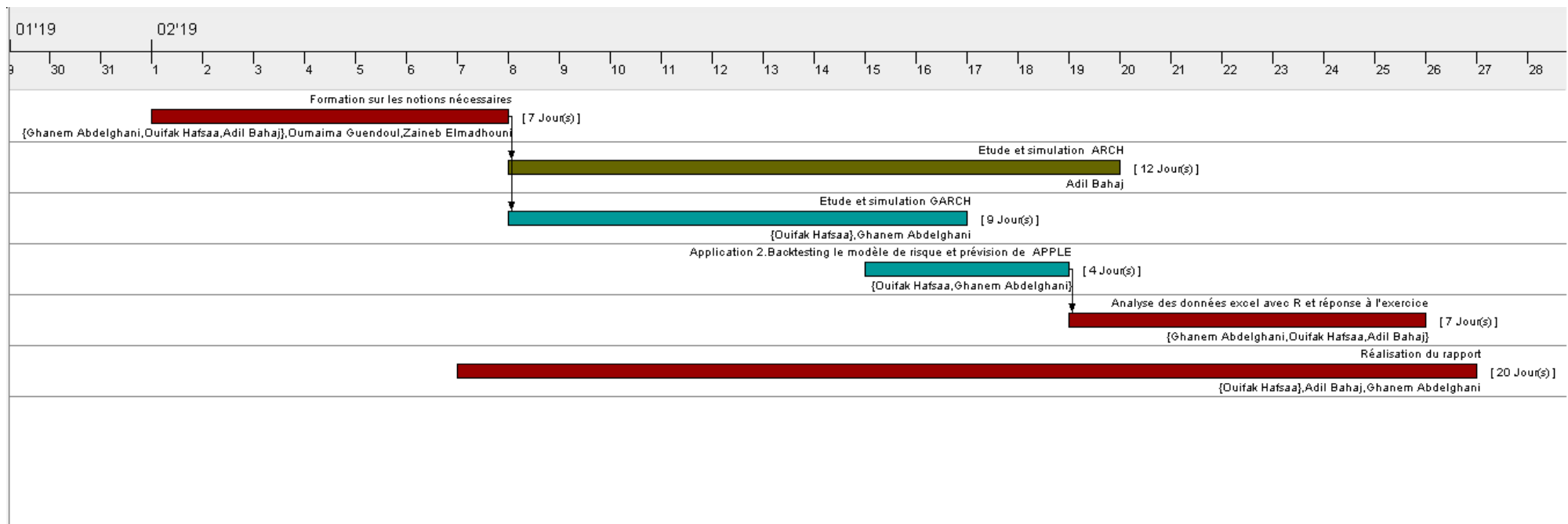


Diagramme des Ressources

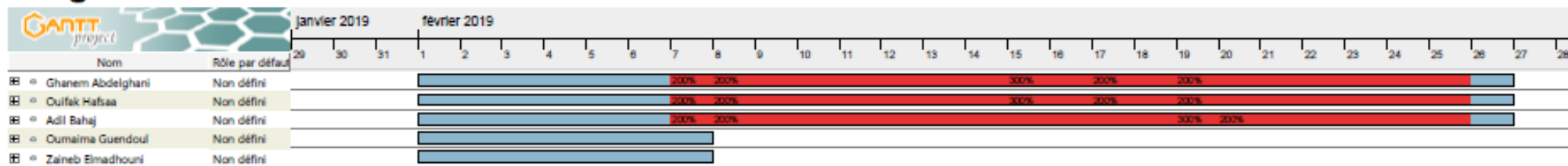


Figure 1 : Diagramme de Gantt

Chapitre 2: Préliminaires et définitions

I. Définitions

1. Processus aléatoire

Un processus aléatoire est une famille de variables ou de vecteurs aléatoires $(X_t)_{t \in T}$, indexée par l'ensemble T des temps dénombrable ou continu, définie sur un espace de probabilité (Ω, \mathcal{A}, P) et a valeurs dans un espace d'états E . La variable ou le vecteur aléatoire X_t décrit l'état du processus au temps t . A toute occurrence ω on fait correspondre la trajectoire de la réalisation $(X_t(\omega))_{t \in T}$ du processus définie par l'application : $t \in T \rightarrow X_t(\omega) \in E$

2. Volatilité

La volatilité est l'ampleur des variations du cours d'un actif financier. Elle sert de paramètre de quantification du risque de rendement et de prix d'un actif financier. Lorsque la volatilité est élevée, la possibilité de gain est plus importante, mais le risque de perte l'est aussi. C'est par exemple le cas de l'action d'une société plus endettée, ou disposant d'un potentiel de croissance plus fort et donc d'un cours plus élevé que la moyenne. Si la croissance des ventes est moins forte qu'espérée ou si l'entreprise peine à rembourser sa dette, la chute du cours sera très forte.

La notion est plus souvent utilisée pour les oscillations à court terme que pour les grandes fluctuations boursières sur plusieurs années, souvent qualifiées (bien qu'irrégulières dans leur fréquence) de cycles boursiers. En réalité, le terme de volatilité concerne aussi bien le court terme que le moyen et long terme. Il ne caractérise pas l'indécision du marché à un instant donné, mais l'ampleur des variations de cours, à la hausse comme à la baisse, qui traduisent notamment des changements d'anticipations.

Sur le marché financier, la volatilité est mesurée comme l'écart type de la rentabilité. L'estimation de l'écart type des rentabilités journalières servent comme une méthode utile pour caractériser l'évolution de la volatilité. Cette statistique mesure la dispersion de la rentabilité:

$$\sigma = \sqrt{\frac{\sum_{t=1}^T (R_t - \bar{R})^2}{T - 1}}$$

Où \bar{R} est la rentabilité moyenne de l'échantillon. L'écart type est une mesure simple mais utile de la volatilité. Quand l'écart type est grand, la chance d'avoir une rentabilité élevée est grande.

Plusieurs études ont utilisé la modification de l'écart type pour mesurer la volatilité.

3. Stationnarité

Une des grandes questions dans l'étude de séries temporelles (ou chronologiques) est de savoir si celles-ci suivent un processus stationnaire. On entend par là le fait que la structure du processus sous-jacent supposé évolue ou non avec le temps. Si la structure reste la même, le processus est dit alors stationnaire.

a. Définition 1

Le processus x_t est dit strictement ou fortement stationnaire si \forall le n-uplet du temps $t_1 < t_2 < \dots < t_n$, tel que $t_i \in Z$ et pour tout temps $h \in Z$ avec $t_i + h \in Z, \forall i, i = 1, \dots, n$, la suite $(x_{t_1+h}, \dots, x_{t_n+h})$ à la même loi de probabilité que la suite $(x_{t_1}, \dots, x_{t_n})$.

b. Définition 2

Un processus $(X_t, t \in \mathbb{Z})$ est dit stationnaire au second ordre, ou stationnaire au sens faible, ou stationnaire d'ordre deux si les trois conditions suivantes sont satisfaites :

- i. $E(X_t^2) < \infty \forall t \in \mathbb{Z}$
- ii. $E(X_t) = m$, indépendant de $t, \forall t \in \mathbb{Z}$
- iii. $cov(X_t, X_{t+h}) = E[(X_{t+h} - m)(X_t - m)] = \gamma(h)$, indépendant de $t, \forall (t, h) \in \mathbb{Z}^2$

4. Bruit blanc

a. Définition

Soit $(\varepsilon_t)_{t \in \mathbb{Z}}$ un processus stochastique, on dit que $(\varepsilon_t)_{t \in \mathbb{Z}}$ est un processus stochastique hasard pure ou bruit blanc si les trois propriétés suivantes sont vérifiées :

- i. $E(\varepsilon_t) = 0 \forall t \in \mathbb{Z}$
- ii. $var(\varepsilon_t) = \sigma^2 \forall t \in \mathbb{Z}$

$$\text{iii. } \text{cov}(\varepsilon_t, \varepsilon_s) = E(\varepsilon_t \varepsilon_s) = 0 \quad \forall t \neq s$$

b. Remarque

On parle de bruit blanc fort si les ε_t sont centrées, de variance finie, identiquement distribuées et indépendantes.

5. Modèle linéaire

a. Définition

Un processus $(X_t, t \in \mathbb{Z})$ est un processus linéaire (respectivement linéaire général) de moyenne μ s'il s'écrit sous la forme :

$X_t = \mu + \sum_{k=1}^{\infty} \alpha_k \varepsilon_{t-k}$ où $(\varepsilon_t)_{t \in \mathbb{Z}}$ est un bruit blanc fort (resp faible) avec variance σ^2 et où la suite de coefficients α_k est supposée telle que :

$$\sum_{k=1}^{\infty} \alpha_k^2 < \infty.$$

6. Modèle ARMA

En statistique, les modèles ARMA (modèles autorégressifs et moyenne mobile), ou aussi modèle de Box-Jenkins, sont les principaux modèles de séries temporelles.

Étant donné une série temporelle X_t , le modèle ARMA est un outil pour comprendre et prédire, éventuellement, les valeurs futures de cette série. Le modèle est composé de deux parties : une part autorégressive (AR) et une part moyenne-mobile (MA). Le modèle est généralement noté ARMA(p, q), où p est l'ordre de la partie AR et q l'ordre de la partie MA.

a. Définition

un modèle autorégressif et moyenne-mobile d'ordres (p,q) (abrégé en ARMA(p,q)) est un processus temporel discret $(X_t, t \in \mathbb{N})$ vérifiant :

$$X_t = \varepsilon_t + \sum_{i=1}^p \varphi_i X_{t-i} + \sum_{i=1}^q \theta_i \varepsilon_{t-i}$$

Où les φ_i et θ_i sont les paramètres du modèle et les ε_i les termes d'erreur.

7. Hétéroscédasticité

En statistique, l'on parle d'hétéroscédasticité lorsque les variances des résidus des variables examinées sont différentes. Cette notion provient du grec et est composée du préfixe hétéro- (« autre »), et de skedasê (« dissipation»). Une collection de variables aléatoires est hétéroscédastique, s'il y a des sous-populations qui ont des variabilités différentes des autres.

La notion d'hétéroscédasticité s'oppose à celle d'homoscédasticité, qui correspond au cas où la variance de l'erreur des variables est constante. Tandis que dans le cas d'homoscédasticité, nous avons :

$Var(\varepsilon_i) = \sigma^2 \forall i$, nous avons désormais $Var(\varepsilon_i) = \sigma_i^2$ où σ_i^2 peut être différent de σ_j^2 pour $i \neq j$.

Chapitre 3 : Modèles SARIMA, ARCH et GARCH

I. Modèle autorégressif moyenne mobile intégré saisonnier SARIMA

Les modèles ARMA prédisent la variable de résultat à partir des valeurs du résultat aux instants précédents, mais cela ne convient que pour un processus stationnaire. Si une variation saisonnière claire est observée, le modèle SARIMA pourrait être plus approprié, ce qui permet une tendance à long terme et des effets saisonniers. Le modèle SARIMA est basé sur l'application des modèles ARMA à des séries chronologiques transformées, dans lesquelles le comportement saisonnier et non stationnaire a été éliminé.[1]

1. Définition

De façon générale, soient s_1, \dots, s_n n entiers alors un processus (X_t) est un processus SARIMA (p, d, q) s'il vérifie une équation du type :

$$\phi(L) = (1 - L^{s_1}) \dots (1 - L^{s_n}) X_t = \theta(L)(1 - L)^d \varepsilon_t$$

Où $\phi(L) = 1 - \phi_1 L - \phi_2 L^2 + \dots - \phi_p L^p$ où $\phi_p \neq 0$ et $\theta(L) = 1 + \theta_1 L + \theta_2 L^2 + \dots + \theta_q L^q$ (où $\theta_q \neq 0$) sont des polynômes dont les racines sont de modules supérieurs à 1, et où les conditions initiales

$$Z_{-1} = \{X_{-1}, \dots, X_{-p}, \varepsilon_{-1}, \dots, \varepsilon_{-q}\}$$

Sont non corrélées avec $\varepsilon_0, \dots, \varepsilon_t$ et où le processus ε_t est un bruit blanc de variance σ^2 .

Les deux formes les plus utilisées sont les suivantes :

$$\phi(L) = (1 - L^s) X_t = \theta(L) \varepsilon_t \text{ Pour tout } t \geq 0$$

$$\phi(L) = (1 - L^s)(1 - L)^d X_t = \theta(L) \varepsilon_t \text{ Pour tout } t \geq 0$$

2. Estimation d'un SARIMA

La méthodologie pour estimer un processus SARIMA suivant (BOX & Jenkins) :

- Identification de l'ordre d : Poser $Y_t = (1 - L)^d X_t$
- Identification de l'ordre s : Poser $Z_t = (1 - L)^s Y_t$

- Identification de l'ordre p, q tels que $\phi_p(L)Z_t = \theta_q(L)\varepsilon_t$
- Estimer ϕ_1, \dots, ϕ_p et $\theta_1, \dots, \theta_q$
- Construire la série $(\hat{\varepsilon}_t)$, en déduire un estimateur de σ^2
- Vérifier que $(\hat{\varepsilon}_t)$ est un bruit blanc

II. Modèles Autorégressifs Conditionnellement Hétéroscédastiques ARCH

Il existe deux représentations d'un processus ARCH, une forme sans résidus et une forme avec résidus :

1. Définition

Un processus Y_t satisfait une représentation $ARCH(p)$ si :

$$forme\ 1: \begin{cases} Y_t = z_t \sqrt{h_t} \\ h_t = \alpha_0 + \sum_{i=1}^p \alpha_i Y_{t-i}^2 \\ z_t \sim N(0, 1) \end{cases}$$

Dans cette forme l'étude se fait directement sur la série temporelle, par contre il y'a une autre forme introduite par Engle qui est équivalente à cette forme mais où un résidu est l'élément qui suit $ARCH(p)$ et le processus Y_t est définie en fonction de ce résidu (erreur):

$$forme\ 2: \begin{cases} Y_t = E(Y_t | Y_{t-1}) + \varepsilon_t \\ \varepsilon_t = z_t \sqrt{h_t} \\ h_t = \alpha_0 + \sum_{i=1}^p \alpha_i \varepsilon_{t-i}^2 \\ z_t \sim N(0, 1) \end{cases}$$

On a donc un modèle qui décrit à la fois l'évolution de l'espérance conditionnelle et de la variance conditionnelle du processus Y_t dans le temps.

L'espérance conditionnelle peut être exprimée de plusieurs façons, par des modèles linéaires et non linéaires, si on suppose que Y_t est un processus linéaire alors le modèle devient :

$$\text{forme 3:} \begin{cases} Y_t = X_t \beta + \varepsilon_t \\ \varepsilon_t = z_t \sqrt{h_t} \\ h_t = \alpha_0 + \sum_{i=1}^p \alpha_i \varepsilon_{t-i}^2 \\ z_t \sim N(0, 1) \end{cases}$$

L'un des modèles linéaires les plus utilisés pour exprimer les séries temporelles est le modèle autorégressif AR(q), donc on peut exprimer le processus de cette façon :

$$\text{forme 4:} \begin{cases} Y_t = \mu + \sum_{i=1}^q \theta_i Y_{t-i} + \varepsilon_t \\ \varepsilon_t = z_t \sqrt{h_t} \\ h_t = \alpha_0 + \sum_{i=1}^p \alpha_i \varepsilon_{t-i}^2 \\ z_t \sim N(0, 1) \end{cases}$$

Remarque :

- De façon générale, z_t désigne un ensemble de variables aléatoires indépendantes, identiquement distribuées, centrées, réduites.

-La composante h_t désigne une variable qui, conditionnellement à l'ensemble d'information des valeurs passées de Y_t , i.e. à $\underline{Y_{t-1}} = \{Y_{t-1}, Y_{t-2}, \dots, Y_{t-j}, \dots\}$, est déterministe et positive.

III. Modèle ARCH généralisé GARCH

1. Modèle GARCH(p, q)

a. Définition

On considère un modèle autorégressif exprimé sous la forme :

$$X_t = E \left[\frac{X_t}{I_{t-1}} \right] + \varepsilon_t \text{ avec } \varepsilon_t \text{ est un bruit blanc faible, qui satisfait la propriété } E \left[\frac{\varepsilon_t}{I_{t-1}} \right] = 0.$$

$$\varepsilon_t = \eta_t h_t$$

L'équation de la variance conditionnelle d'un processus GARCH s'écrit :

$$h_t = \sqrt{\alpha_0 + \sum_{i=1}^p \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^q \beta_j h_{t-j}^2}$$

Où $\eta_t \sim N(0, \sigma^2)$, avec les conditions $\alpha_0 > 0, \alpha_i > 0$, pour $i = 1, 2, \dots, p$ et $\beta_j > 0$ pour $j = 1, 2, \dots, q$ satisfaisantes pour garantir la positivité de $\alpha_0 + \sum_{i=1}^p \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^q \beta_j h_{t-j}^2$.

b. Propriétés

Propriété b.1 Le processus ε_t est un bruit blanc, si $E(\varepsilon_t^2) < \infty$.

$$\text{On a } E(\varepsilon_t) = E\left(E\left[\frac{\varepsilon_t}{I_{t-1}}\right]\right) = 0$$

Et

$$\begin{aligned} \text{cov}(\varepsilon_t, \varepsilon_{t-k}) &= E(\varepsilon_t \varepsilon_{t-k}) \\ &= E(\varepsilon_{t-k} E\left[\frac{\varepsilon_t}{I_{t-1}}\right]) = 0, \forall k > 0 \end{aligned}$$

Propriété b.2 Une condition nécessaire de l'existence de la variance d'un processus GARCH(p, q) est :

$$\sum_{i=1}^p \alpha_i + \sum_{j=1}^q \beta_j < 1$$

Propriété b.3 Le processus ε_t^2 d'une représentation GARCH(p, q) peut être représenté sous la forme d'un processus ARMA (max(p, q), q) défini dans une innovation $v_t = \varepsilon_t^2 - h_t^2$ tel que :

$$\varepsilon_t^2 = \alpha_0 + \sum_{i=1}^n (\alpha_i + \beta_i) \varepsilon_{t-i}^2 + \sum_{j=1}^q \beta_j v_{t-j} + v_t, n = \max(p, q)$$

Avec la convention $\alpha_i = 0$ si $i > p$ et $\beta_j = 0$ si $j > q$.

2. Modèle GARCH(1, 1)

Pour modéliser les données empiriques sur le marché financier, un modèle avec erreur GARCH (1, 1) est souvent suffisant. Il est donné par l'équation :

$$Y_t = cX_t + \varepsilon_t$$

Avec $\varepsilon_t = \eta_t h_t$ et $h_t = \sqrt{\alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 h_{t-1}^2}$

Avec $\alpha_0 > 0, \alpha_1 > 0$ et $\beta_1 > 0$.

Dans ce modèle les carrés des résidus suivent un processus ARMA(1, 1),

$$\varepsilon_t^2 = \alpha_0 + (\alpha_1 + \beta_1)\varepsilon_{t-1}^2 + \beta_1 v_{t-1} + v_t$$

Il est stationnaire pour $0 < \alpha_1 + \beta_1 < 1$, où $v_t = \varepsilon_t^2 - h_t^2$ est un processus d'innovation pour ε_t^2 .

Sous la condition de stationnarité de second ordre, la variance inconditionnelle du processus ε_t existe et est constante au cours du temps. Sachant que $V(\varepsilon_t) = E(\varepsilon_t^2)$, il suffit à partir de la forme ARMA (1, 1) sur ε_t^2 de définir la variance du processus :

$$V(\varepsilon_t) = \frac{\alpha_0}{1 - (\alpha_1 + \beta_1)}.$$

Chapitre 4 : Simulation

I. Outils de simulation

1. Logiciel et Langage

a. Langage R



R est un langage de programmation et un logiciel libre destiné aux statistiques et à la science des données soutenu par la R Foundation for Statistical Computing. R fait partie de la liste des paquets GNU3 et est écrit en C (langage), Fortran et R.

Le langage R est largement utilisé par les statisticiens, les data miners, data scientists pour le développement de logiciels statistiques et l'analyse des données.

b. RStudio



RStudio est un environnement de développement gratuit, libre et multiplateforme pour R, un langage de programmation utilisé pour le traitement de données et l'analyse statistique. Il est disponible sous la licence libre AGPLv3, ou bien sous une licence commerciale, soumise à un abonnement annuel.

2. Packages principaux

a. rugarch

Le package rugarch est le premier logiciel open source pour la modélisation univariée de GARCH. Il est écrit en R à l'aide de méthodes et de classes S4 avec une partie importante du

code en C et C ++ pour la rapidité. Il contient un certain nombre de modèles GARCH au-delà de la version vanille, notamment IGARCH, EGARCH, GJR, APARCH, FGARCH...

b. tseries

Package avec des méthodes de simulation du chemin d'un modèle GARCH à partir d'une variété de modèles GARCH univariés. C'est une fonction pratique qui ne nécessite pas d'objet ajusté (voir la remarque ci-dessous).

c. Forecast

Des prévisions automatiques d'un grand nombre de séries chronologiques univariées sont souvent nécessaires dans le monde des affaires et dans d'autres contextes. Nous décrivons deux algorithmes de prévision automatique qui ont été implémentés dans le paquet `\ pkg {forecast}` pour `\ proglang {R}`. La première est basée sur des modèles d'espace à états innovants qui reposent sur des méthodes de lissage exponentielles. Le second est un algorithme de prévision pas à pas avec les modèles ARIMA. Les algorithmes sont applicables aux données saisonnières et non saisonnières. Ils sont comparés et illustrés à l'aide de quatre séries temporelles réelles. Nous décrivons également brièvement certaines des autres fonctionnalités disponibles dans le paquet `\ pkg {forecast}`.

I. Simulation ARCH

1. Simulation du modèle ARCH(1)

a. spécification du modèle :

$$ARCH(1) + ARMA(0,0) \rightarrow \begin{cases} r_t = \mu + \varepsilon_t \\ \varepsilon_t = z_t \sqrt{h_t} (\sim N(0, h_t)) \\ z_t \sim N(0,1) \\ h_t = 0.1 + 0.8 * \varepsilon_{t-1}^2 \end{cases}$$

Avec $\mu = 0$;

```
arch1.spec = ugarchspec(variance.model = list(garchOrder=c(1,0)), mean.model =  
list(armaOrder=c(0,0)), distribution.model = "norm", fixed.pars=list(mu = 0, omega=0.1,  
alpha1=0.8))
```

b. création de série simulée:

```
set.seed(123)
```

```
arch1.sim = ugarchpath(arch1.spec, n.sim=1000)
```

-arch1.sim est un variable compose de deux séries. Une série qui représente les variations de rendement r_t et une autre représente les variations de volatilité h_t .

-la fonction 'seed' fixe la séquence des variables aléatoire que n'importe qu'elle fonction qui génère des variables aléatoires peut générer, pour assurer la reproductibilité des résultats des simulations.

c. simulation de rendement r_t

```
plot(arch1.sim, which=2)
```

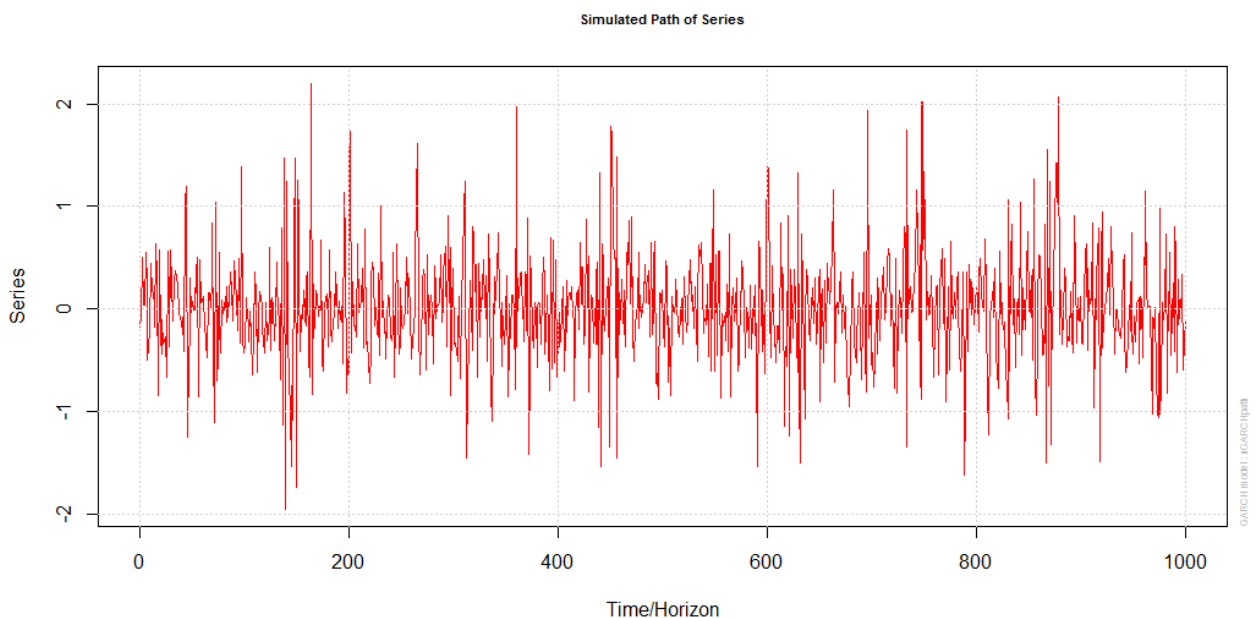


Figure 2 : Figure des rendements

d. simulation de variance (volatilité) h_t :

```
plot(arch1.sim, which=1)
```

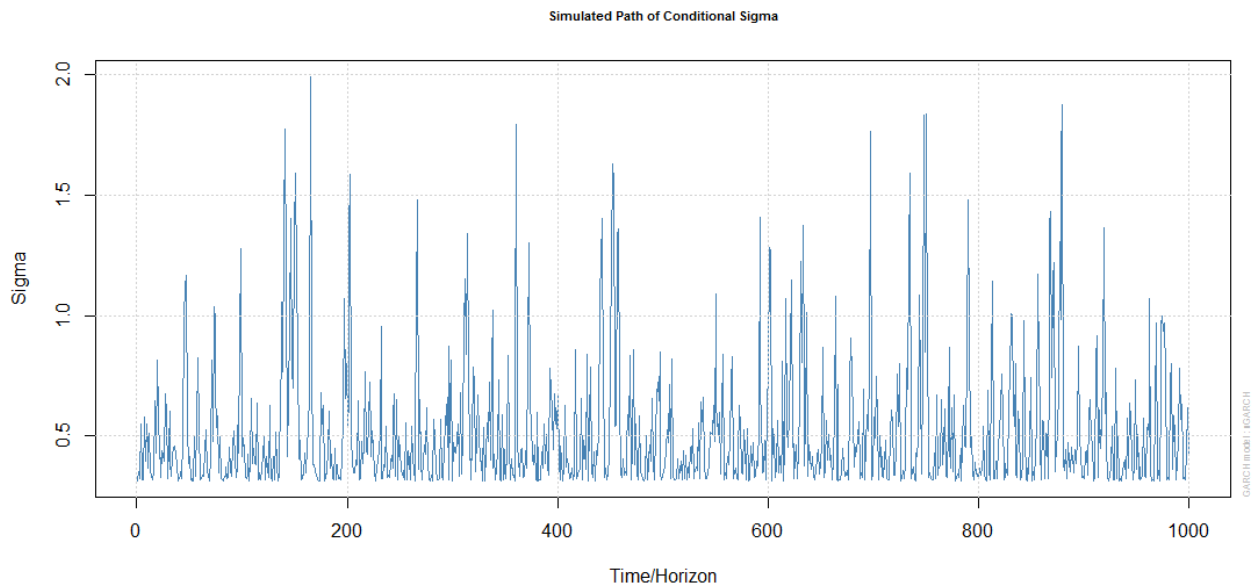


Figure 3 : Figure des variances

e. Parallèles entre la simulation d'ARCH (1) et les propriétés des séries financières :

- Graphes des rendements et de volatilité :

Code R :

```
par(mfrow=c(2,1))
```

```
plot(arch1.sim, which=2)
```

```
plot(arch1.sim, which=1)
```

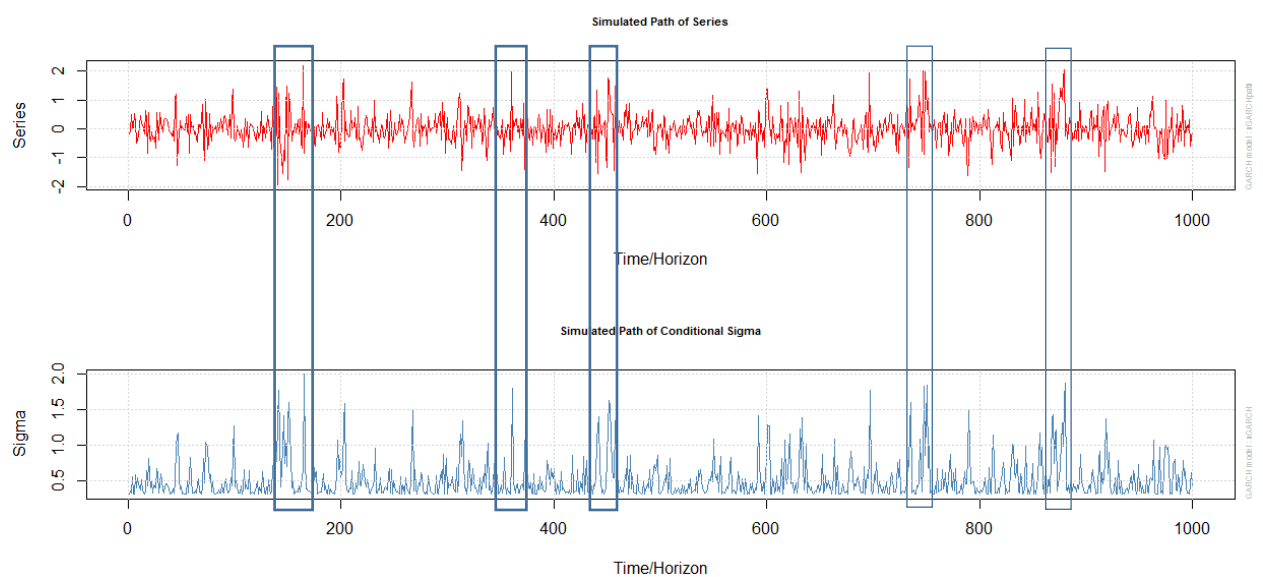


Figure 4 : Figure des series et de sigma conditionnelle

Observations:

→ Stationnarité des rendements r_t .

→ Les forts rendements sont généralement suivis de forts rendements, et les faibles rendements sont généralement suivis de faibles rendements

f. graphes d'autocorrélations de r_t , r_t^2 et $|r_t|$:**acf : Autocorrelation And Cross-Correlation Function Estimation**

- La fonction Acf calcule (et trace par défaut) une estimation de la fonction d'autocorrélation d'une série temporelle (éventuellement multivariée).

- la fonction d'autocorrélation indique la corrélation entre les points séparés par divers décalages temporels dans une série chronologique.

Exemple :

ACF (0) = 1 (toutes les données sont parfaitement corrélées entre elles), ACF (1) = 0.9 (la corrélation entre un point et le point suivant est 0,9), ACF (2) = 0.4 (la corrélation entre un point et un point deux pas en avant est 0.4) ... etc.

Formulation mathématique :

Le concept d'autocorrélation a pour but de calculer le coefficient de corrélation d'une série temporelle avec elle-même, décalée dans le temps. Si les données ont une périodicité, le coefficient de corrélation sera plus élevé lorsque ces deux périodes seront en résonance.

La première étape consiste à définir un opérateur pour décaler une série temporelle dans le temps, provoquant un retard de t. Ceci est connu comme l'opérateur de décalage:

$$Lag(X_t, t) = X_{t-t}$$

L'autocorrélation d'une série temporelle avec décalage t est définie comme suit:

$$autocor(X, t) = corr(X, Lag(X, t))$$

Et puisqu'on sait que la corrélation entre deux VAR s'écrit :

$$corr(X, Y) = \frac{cov(X, Y)}{std(Y)std(X)}$$

Avec :

$$\text{std}(X) = \sqrt{\frac{1}{N} \sum_{i=1}^N (X_i - \bar{X})^2}, \bar{X} = \frac{1}{N} \sum_{i=1}^N X_i \text{ et } \text{cov}(X, Y) = \frac{1}{N} \sum_{i=1}^N (X_i - \bar{X})(Y_i - \bar{Y})$$

Alors :

$$\text{autocor}(X, t) = \frac{\text{cov}(X, \text{Lag}(X, t))}{\text{std}(\text{Lag}(X, t))\text{std}(X)} = \frac{\text{cov}(X, \text{Lag}(X, t))}{\text{std}(X)^2} = \frac{\sum_{i=1}^N (X_i - \bar{X})(X_{i-t} - \bar{X})}{\sum_{i=1}^N (X_i - \bar{X})^2}$$

Code R :

```
par(mfrow=c(3,1))
```

```
acf(arch1.sim@path$seriesSim, main="Rendements")
```

```
acf(arch1.sim@path$seriesSim^2, main="Rendements^2")
```

```
acf(abs(arch1.sim@path$seriesSim), main="abs(Rendements)")
```

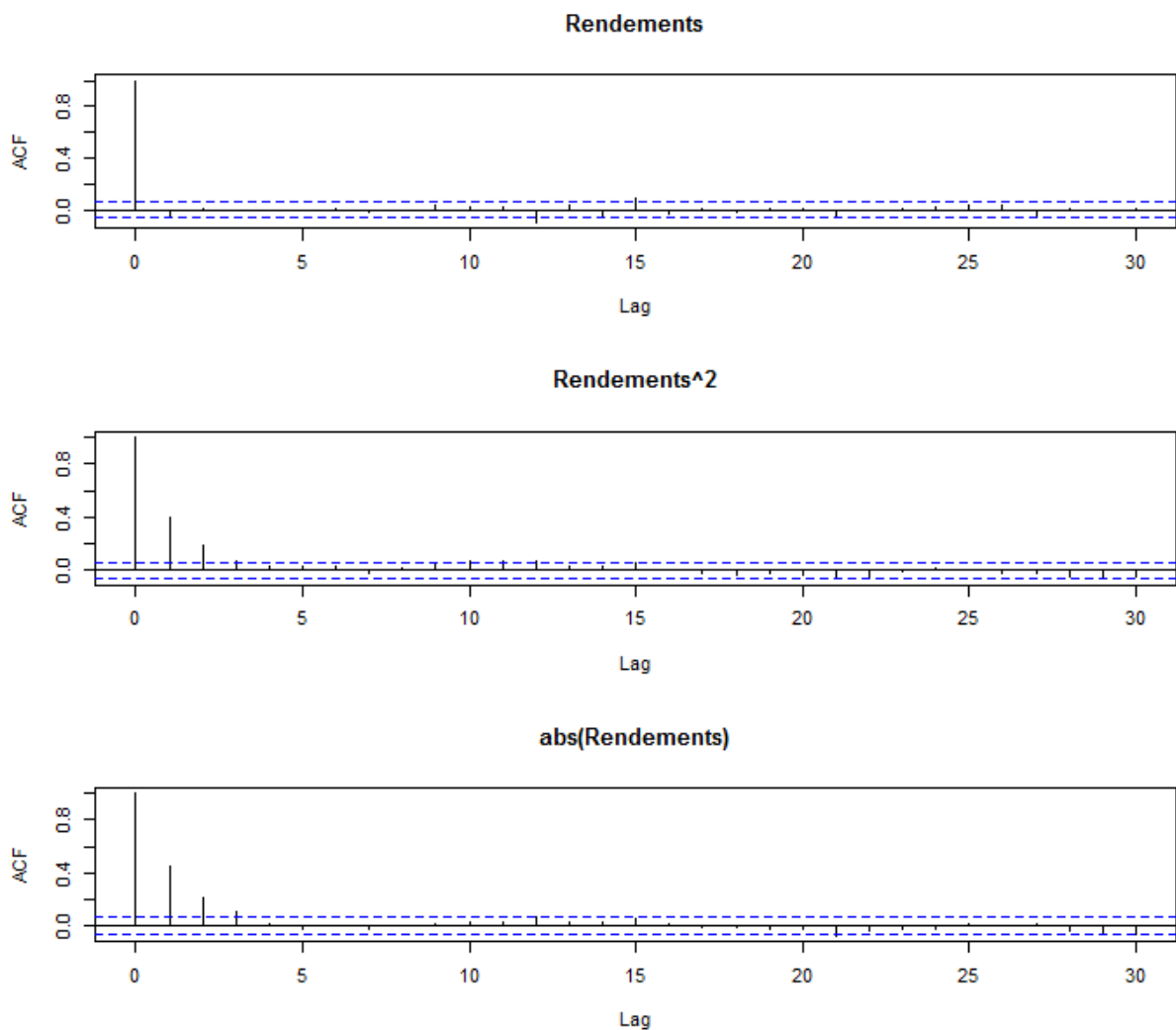


Figure 5 : Figure des différents rendements

Observations :

→ $|r_t|$ et r_t^2 sont plus auto-corrélé que r_t .

g. Diagramme Quantile-Quantile:

En statistiques, le diagramme Quantile-Quantile ou diagramme Q-Q ou Q-Q plot est un outil graphique permettant d'évaluer la pertinence de l'ajustement d'une distribution donnée à un modèle théorique.

Le terme de quantile-quantile provient du fait que l'on compare la position de certains quantiles dans la population observée avec leur position dans la population théorique.

Le diagramme quantile-quantile permet également de comparer deux distributions que l'on estime semblables.

Principe :

À l'issue d'une enquête statistique, on soupçonne celle-ci de suivre une distribution classique (distribution gaussienne, exponentielle ou autre). À partir de la série statistique observée, on calcule alors un certain nombre de quantiles x_i . Si la série statistique suit bien la distribution théorique choisie, on devrait avoir les quantiles x_i observés égaux aux quantiles x_i^* associés au modèle théorique.

On place alors le nuage de points $M_i(x_i^*, x_i)$. En abscisse se trouvent donc les quantiles théoriques et en ordonnée les quantiles observés. Si la distribution théorique choisie est pertinente, les points doivent se positionner suivant la première diagonale.

Si les points sont alignés suivant une droite d'équation $x_i = ax_i^* + b$ c'est que le modèle choisi est valable à une transformation affine près.

Le choix des quantiles peut se faire en divisant la population en $n+1$ tranches égales et en prenant pour x_i le seuil en dessous duquel se trouve une fraction de la population égale à $\frac{i}{n+1}$

Si la distribution théorique a pour fonction de répartition F , on a alors

$$x_i^* = F^{-1}\left(\frac{i}{n+1}\right)$$

Code R :

```
library(car)
```

```
qqPlot(arch1.sim@path$seriesSim, ylab="Rendement de ARCH(1)")
```

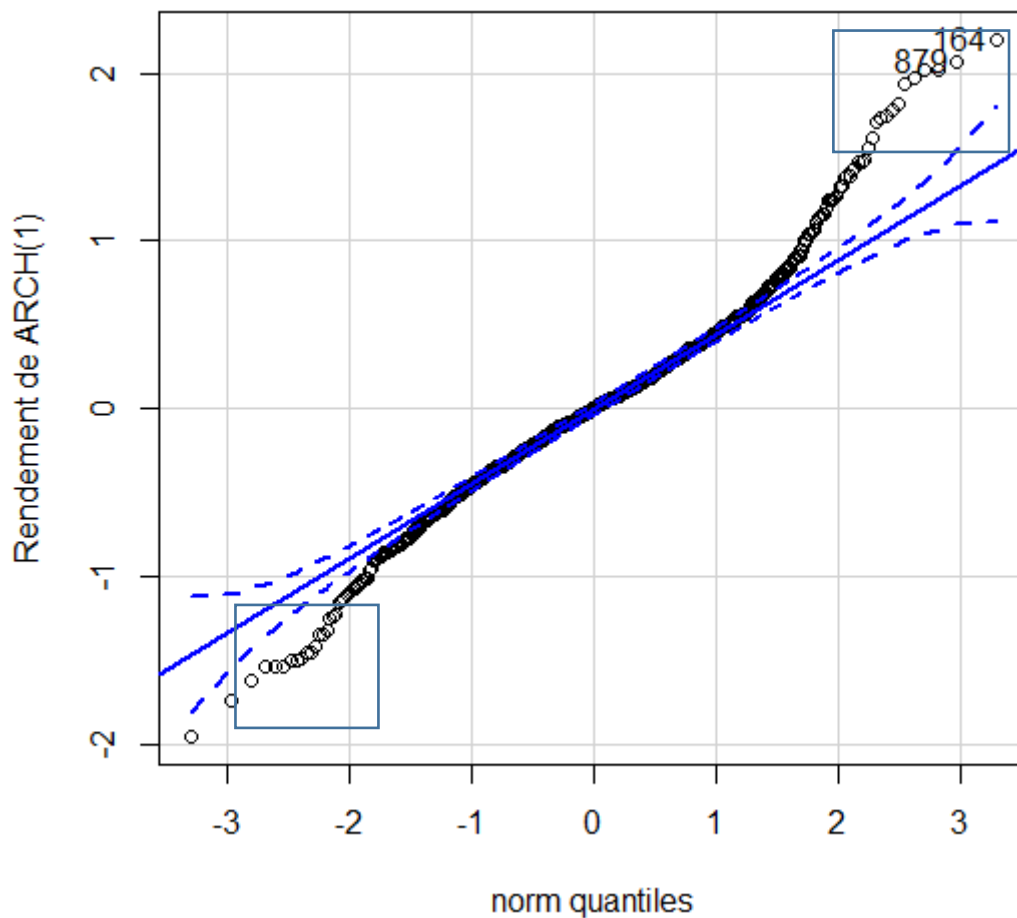


Figure 6 : Figure des rendements

Observations:

→ Queues épaisses conditionnelles ce qui met l'action sur l'aspect leptokurtique de ARCH.

II. Simulation et application GARCH

1. Simulation de garch(2,1) et garch(1,1)

Nous essayons de simuler le modèle :

$$y_t = 2 + \eta_t h_t$$

$$h_t = \sqrt{0.09 + 0.15\varepsilon_{t-1}^2 + 0.3\varepsilon_{t-2}^2 + 0.4h_{t-1}^2}$$

On utilisera la fonction `garchSim()` du package `fGarch`.

On fait d'abord une spécification de notre modèle puis on appelle la fonction de simulation :

```
library(fGarch)

spec=garchSpec(model=list(mu=2,omega=.09,alpha=c(.15,.3),beta=.4),rseed=9647)
y=garchSim(spec,n=400,extended=TRUE)
y1=y[1:400,1]
plot.ts(y1,xlab="temps")
```

La série à étudier est formée des 400 observations contenues dans la première colonne de y.

La courbe des observations au cours du temps est comme suit :

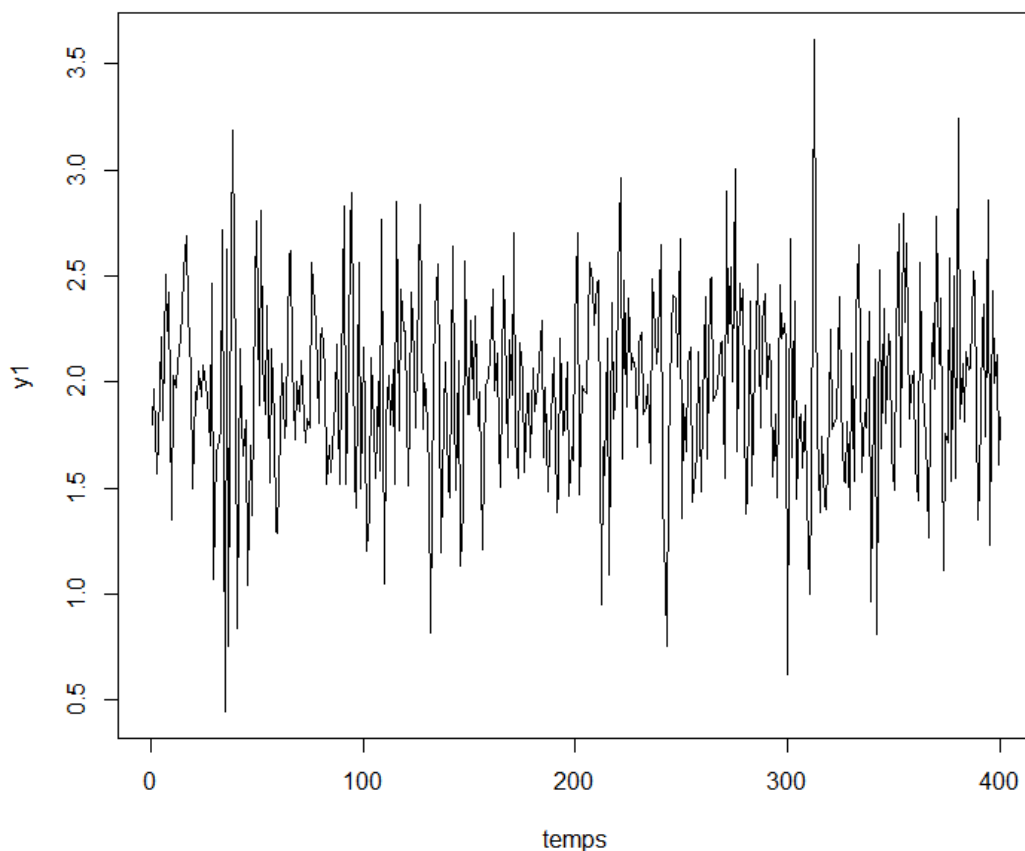


Figure 7 : Observations au cour du temps

On passe à spécifier le modèle GARCH(1,1) en utilisant d'autres paramètres puis le simuler :

```
garch11.spec = ugarchspec(variance.model = list(garchorder=c(1,1)),
                           mean.model = list(armaOrder=c(0,0)),
                           fixed.pars=list(mu = 0, omega=0.1, alpha1=0.1,
                                           beta1 = 0.7))
set.seed(123)
garch11.sim = ugarchpath(garch11.spec, n.sim=1000)
```

On utilise plot pour illustrer les séries simulées et les volatilités conditionnelles :

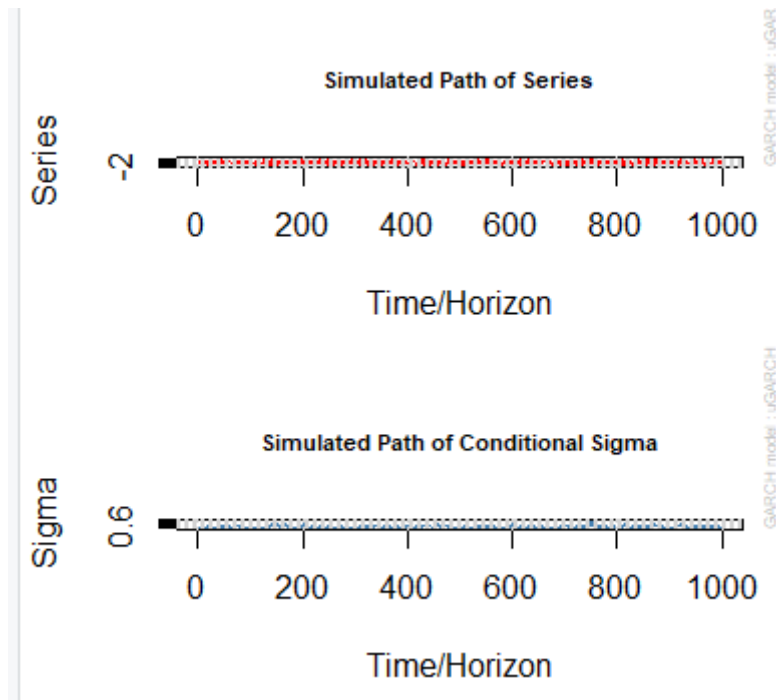


Figure 8 : Séries simulées et volatilité conditionnelle

Remarque : L'illustration des rendements, variances et volatilité se fait de la même manière citée avec le modèle ARCH.

2. Backtesting le modèle de risque et prévision de la société APPLE

- Inclure les librairies nécessaires

```
library("tseries")
library("zoo")
library("forecast")
library("rugarch")
```

- ✓ Importer les données de APPLE

Dans cet exemple, on ne récupère que (Adjusted close prices) de APPLE en utilisant le moteur de « Yahoo »

```
AAPL.data <-
  get.hist.quote(
    instrument = "AAPL",
    start = "2014-01-01",
    end = "2019-02-25",
    quote = "AdjClose",
    provider = "yahoo",
    compression = "d"
  )
```

Pour savoir le contenu de notre variable ou de dessiner la courbe des données :

```
head(AAPL.data)
plot(AAPL.data,
     main = "AAPL closing prices",
     xlab = "Dates",
     ylab = "Prices (USD)")
```

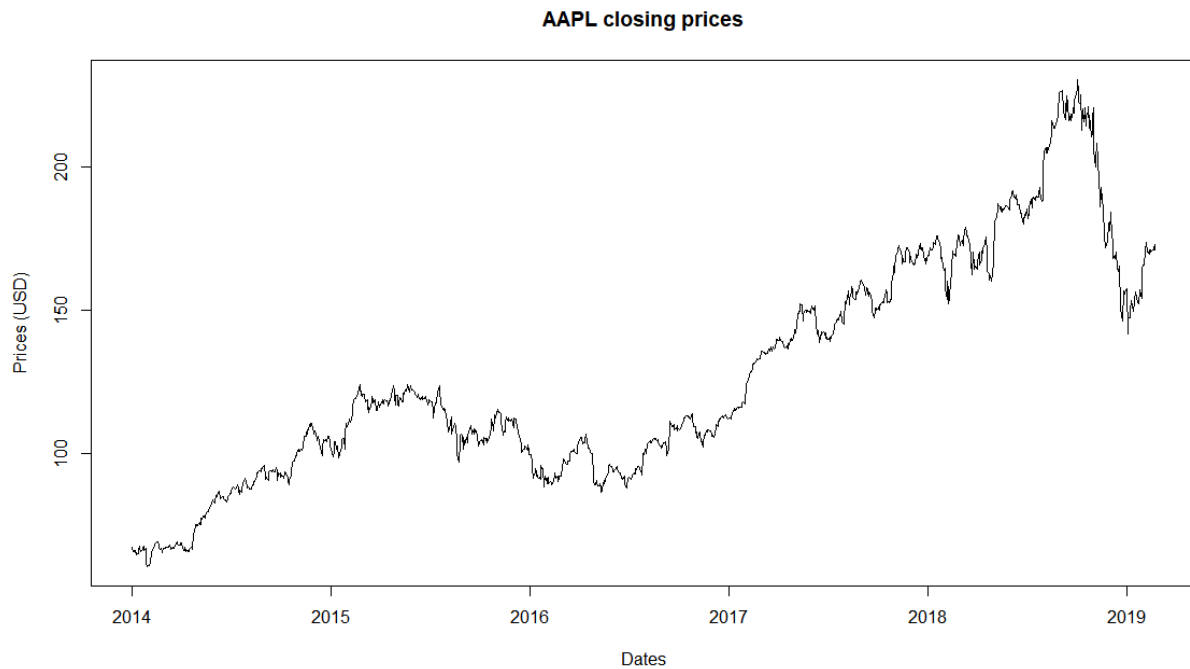


Figure 9: Closing price de APPLE

- **Calcul de « Compound returns »**

Ça va nous permettre de remarquer les différentes variations et les volatilités pendant toutes les périodes passées.

```
AAPL.ret = diff(log(AAPL.data)) * 100
AAPL.ret
plot(AAPL.ret,
     main = "Mounthly compound returns",
     xlab = "Dates",
     ylab = "Reterens in percent")
```

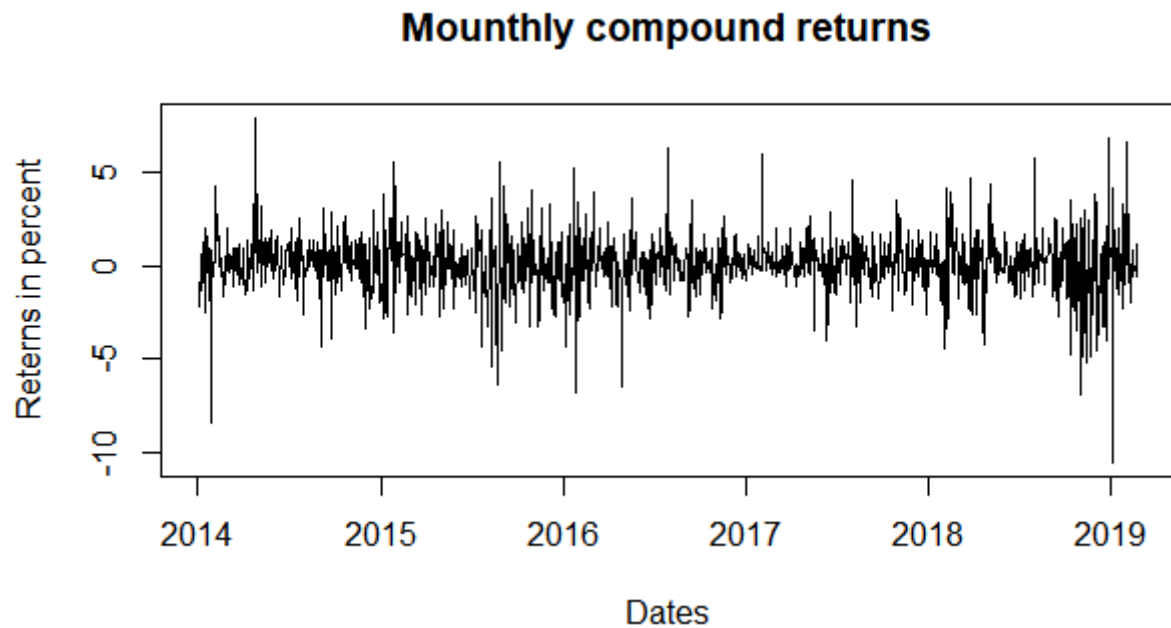


Figure 10 : Rendements mensuels composés de Apple

- **Modèle ARIMA et test de l'effet de ARCH**

```
fit1 = auto.arima(AAPL.ret,
                  trace = TRUE,
                  test = "kpss",
                  ic = "bic")
Box.test(fit1$residuals ^ 2, lag = 12, type = "Ljung-Box")
```

On remarque si dessous qu'on a $p\text{-value} < 5\%$, donc on peut utiliser le model garch.

```
> Box.test(fit1$residuals ^ 2, lag = 12, type = "Ljung-Box")
```

Box-Ljung test

```
data: fit1$residuals^2
X-squared = 94.876, df = 12, p-value = 5.551e-15
```

- **Développer une spécification**

```
res_garch1_spec = ugarchspec(variance.model = list(garchOrder = c(1, 1)),
                             mean.model = list(armaOrder = c(1, 1)))
```

- **Charger les données avec la spécification**

```
res_garch1_fit = ugarchfit(spec = res_garch1_spec, data = AAPL.ret)
res_garch1_fit
```

- Recherche des valeurs à risques

```
ctrl = list(tol = 1e-7, delta = 1e-9)
res_garch1_rol1 = ugarchroll(
  res_garch1_spec,
  AAPL.ret,
  n.start = 120,
  refit.every = 1,
  refit.window = "moving",
  solver = "hybrid",
  calculate.Var = TRUE,
  Var.alpha = 0.01,
  keep.coef = TRUE,
  solver.control = ctrl,
  fit.control = list(scale = 1)
)
```

Cette fonction prend un peu de temps (à peu près 5 min).

Pour voir les valeurs à risque on peut faire comme suit :

```
> plot(res_garch1_fit)
```

Make a plot selection (or 0 to exit):

- 1: Series with 2 Conditional SD Superimposed
- 2: Series with 1% VaR Limits
- 3: Conditional SD (vs |returns|)
- 4: ACF of Observations
- 5: ACF of Squared Observations
- 6: ACF of Absolute Observations
- 7: Cross Correlation
- 8: Empirical Density of Standardized Residuals
- 9: QQ-Plot of Standardized Residuals
- 10: ACF of Standardized Residuals
- 11: ACF of Squared Standardized Residuals
- 12: News-Impact Curve

```
selection: 2
```

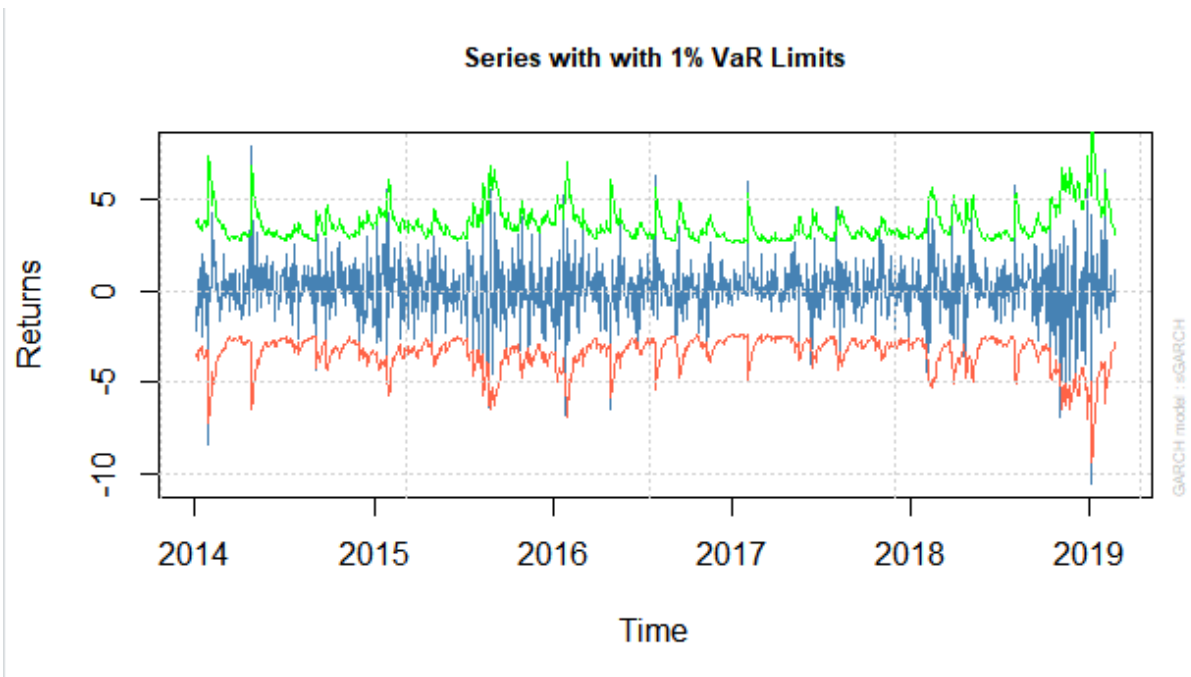



Figure 11: Séries avec 1% de risque

La courbe bleue représente les données rentrées et celle en rouge représente les valeurs de risques.

Les résultats du rapport montrent que la société a touché les valeurs de risques 32 fois pendant la période spécifiée, mais en se basant sur un pourcentage de risque de 1%, la société a touché les valeurs de risques seulement 12 fois.

```
report(res_garch1_roll,type="VaR",VaR.alpha=0.01, conf.level=0.99)
```

```
=====
alpha:                                1%
Expected Exceed:                      11.7
Actual VaR Exceed:                    32
Actual %:                             2.7%
```

Prédire le risque des 10 jours prochains :

```
res_garch1_fcst=ugarchforecast(res_garch1_fit,n.ahead = 10)
```

```
> res_garch1_fcst
```

```

*-----*
*          GARCH Model Forecast          *
*-----*
Model: sGARCH
Horizon: 10
Roll Steps: 0
Out of Sample: 0

0-roll forecast [T0=2019-02-22]:
      Series Sigma
T+1  0.1308 1.271
T+2  0.1268 1.298
T+3  0.1302 1.323
T+4  0.1273 1.346
T+5  0.1298 1.366
T+6  0.1276 1.385
T+7  0.1295 1.402
T+8  0.1279 1.417
T+9  0.1293 1.431
T+10 0.1281 1.444
```

Les valeurs prédites sont celles de la case « forecast » et sous une valeur de risque de 1% les valeurs prédites sont celles de sigma.

Chapitre 5 : Conception et réalisation d'un programme de prédiction des valeurs d'une série temporelles

I. Diagramme d'activité

Voici l'approche utilisée afin de prédire à l'aide des modèles ARIMA, SARIMA, ARCH et GARCH :

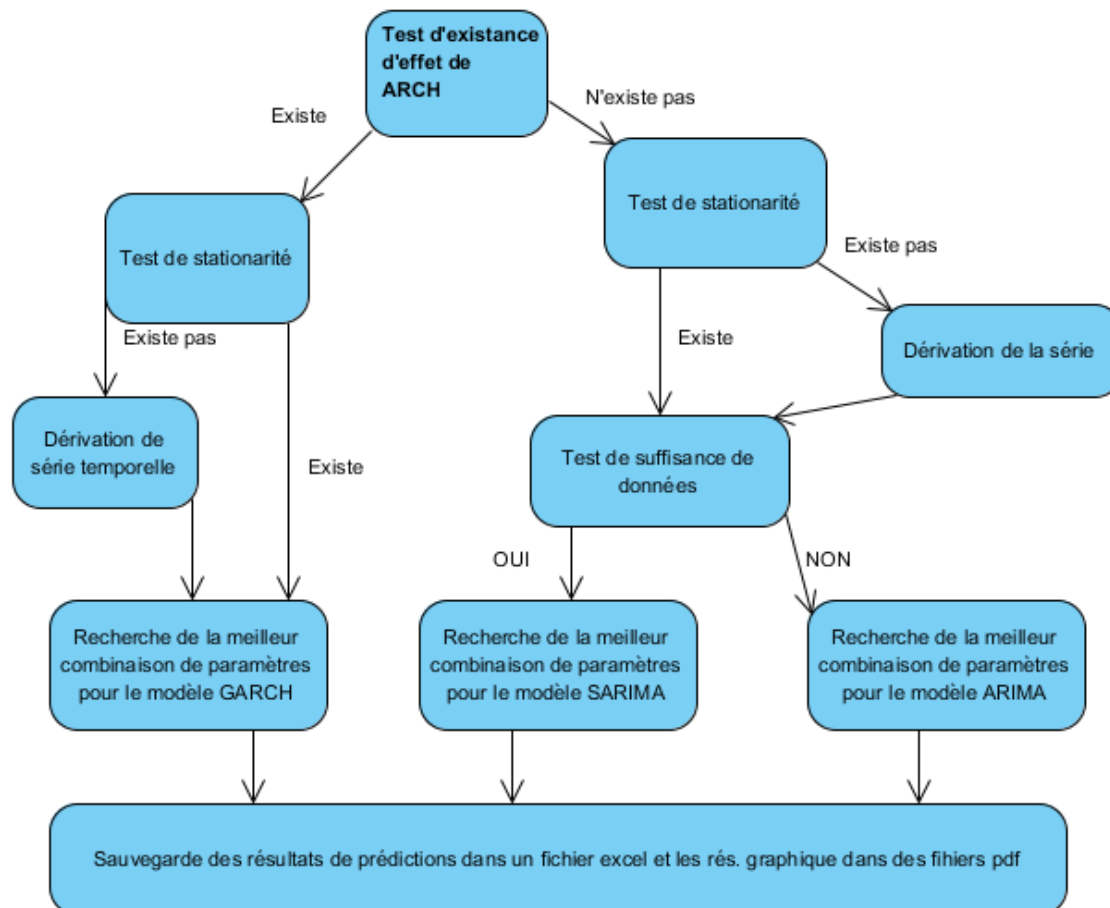


Figure 12: Diagramme d'activité du programme réalisé

II. Logique du programme

Notre programme commence par tester si il existe des autocorrélations ou pas, afin de déterminer si il existe l'effet de ARCH pour savoir le modèle qu'on doit utiliser par la suite. Pour cela on utilise un test qui s'appelle « test de LJUNG BOX », puis notre programme teste si les séries sont stationnaires en utilisant le test de « Dickey Fuller ».

Si la série n'est pas stationnaire on travaille sur sa dérivée. Enfin si le modèle choisis est ARIMA/SARIMA, on teste si on a assez de données sinon on utilise ARIMA pour la prédiction, parce que SARIMA demande encore plus de données d'entrainements.

Pour effectuer une statistique descriptive, il suffit d'appeler la fonction « descStat » du fichier « main.r » pour le « fichier.xlsx » contenant toutes les données :

Tableau 2: Statistiques descriptives

	A	B	C	D	E
1		median	mean	var	std.dev
2	Période.11	1,0805	5,813458	61,73087	7,856899
3	Période.12	2,2545	9,37125	217,4307	14,74553
4	Période.13	3,9895	19,31425	2147,214	46,33804
5	Période.14	6,3715	23,38092	1326,389	36,41962
6	Période.15	2,2605	12,20613	509,1066	22,56339
7	Période.16	2,3875	31,9025	5944,096	77,09797
8	Période.17	13,1405	29,84396	1777,274	42,15772
9	Période.18	7,3165	13,13063	310,9851	17,63477
10	Période.19	1,4235	9,016375	611,2904	24,72429
11	Période.110	0,458	4,565333	87,60231	9,359611
12	Période.111	0,284	2,177542	21,55326	4,642549
13	Période.112	1,968	6,3495	104,4612	10,22063

Le résultat sera automatiquement sauvegardé dans un fichier appelé « statistique_descriptive.xlsx ».

2/ Test de normalité :

On effectue le test de normalité pour chaque période en utilisant la fonction « normalité_test » du fichier « main.r »:

Tableau 3: Test de normalité

	A	B	C	D	E
1	Periode	Statistique.de.te	P.valeur	test.normalite	
2	Période.11	0.7686583197681	9.44600190360695e-05	non gaussien	
3	Période.12	0.6866112256828	6.73535465879107e-06	non gaussien	
4	Période.13	0.4528019176228	2.02373716999713e-08	non gaussien	
5	Période.14	0.6830106540171	6.05471856689964e-06	non gaussien	
6	Période.15	0.5887273457733	4.63505861261455e-07	non gaussien	
7	Période.16	0.4616641270258	2.44210433096941e-08	non gaussien	
8	Période.17	0.7196845282653	1.85438294695735e-05	non gaussien	
9	Période.18	0.7393118981984	3.49095617631819e-05	non gaussien	
10	Période.19	0.3777255209112	4.4379905366607e-09	non gaussien	
11	Période.110	0.5615887698639	2.36660588276607e-07	non gaussien	
12	Période.111	0.5295329735522	1.10521274650433e-07	non gaussien	
13	Période.112	0.6821337532473	5.90026659418224e-06	non gaussien	

3/ Test d'indépendance :

Pour tester l'indépendance, ici dans notre cas les distributions ne sont pas tous normal, mais on peut calculer la matrice de corrélations pour savoir si les éléments sont corrélés entre eux ou pas mais on ne peut pas parler d'indépendance.

On utilise la fonction « correlation_test » du fichier « main.r »

Tableau 4: Test de corrélation

	A	B	C	D	E	F	G	H	I	J	K	L	M
1		variable_51	variable_52	variable_53	variable_54	variable_55	variable_56	variable_57	variable_58	variable_59	variable_511	variable_51	variable_513
2	variable_5	1	0,817333086	0,47046804	0,14065356	0,337627001	0,01071743	-0,01680966	0,19195149	0,435960247	0,701126995	0,62410985	0,676236944
3	variable_5	0,817333086	1	0,226401319	0,13405661	0,280588278	0,0119748	-0,125929136	0,084433216	0,340746433	0,691022048	0,57270042	0,649533241
4	variable_5	0,47046804	0,226401319	1	0,30354836	0,06974093	-0,0422109	-0,132532295	-0,00406338	0,194422596	0,541913619	0,46192563	0,41961062
5	variable_5	0,140653562	0,134056608	0,303548357	1	0,559043043	0,40636702	0,284083506	0,279950377	0,161210081	0,065336147	0,08850917	0,080082135
6	variable_5	0,337627001	0,280588278	0,06974093	0,55904304	1	0,05344095	0,287622752	0,558661386	0,441185139	0,265193995	0,32894503	0,379041511
7	variable_5	0,010717429	0,011974801	-0,04221085	0,40636702	0,053440955	1	0,468040446	0,247347756	0,295298415	0,039441494	0,15271183	0,13853202
8	variable_5	-0,01680966	-0,125929136	-0,132532295	0,28408351	0,287622752	0,46804045	1	0,892912249	0,734204553	-0,1666921	0,25583944	0,279249499
9	variable_5	0,19195149	0,084433216	-0,004063379	0,27995038	0,558661386	0,24734776	0,892912249	1	0,869624973	0,116017716	0,49168402	0,524543846
10	variable_5	0,435960247	0,340746433	0,194422596	0,16121008	0,441185139	0,29529842	0,734204553	0,869624973	1	0,51933759	0,80589383	0,832843672
11	variable_5	0,701126995	0,691022048	0,541913619	0,06533615	0,265193995	0,03944149	-0,166692097	0,116017716	0,51933759	1	0,86633928	0,857149685
12	variable_5	0,624109846	0,572700417	0,461925634	0,08850917	0,328945026	0,15271183	0,255839444	0,491684016	0,805893831	0,866339282	1	0,970437966
13	variable_5	0,676236944	0,649533241	0,41961062	0,08008213	0,379041511	0,13853202	0,279249499	0,524543846	0,832843672	0,857149685	0,97043797	1

Puisque la matrice n'est pas diagonale alors les données sont corrélées.

4/Prédictions pour chaque période en justifiant le choix de modèle :

- Modèles utilisés pour la prédiction

Les modèles que nous avons utilisés pour les prévisions:

1-modèles ARCH/GARCH+ARMA:

$$ARCH(1) + ARMA(0,0) \rightarrow \begin{cases} r_t = \mu + \varepsilon_t \\ \varepsilon_t = z_t \sqrt{h_t} (\sim N(0, h_t)) \\ z_t \sim N(0,1) \\ h_t = \omega + \alpha_1 * \varepsilon_{t-1}^2 \end{cases}$$

$$GARCH(1,1) + ARMA(0,0) \rightarrow \begin{cases} r_t = \mu + \varepsilon_t \\ \varepsilon_t = z_t \sqrt{h_t} (\sim N(0, h_t)) \\ z_t \sim N(0,1) \\ h_t = \omega + \alpha_1 * \varepsilon_{t-1}^2 + \beta_1 h_{t-1}^2 \end{cases}$$

$$ARCH(1) + ARMA(1,0) \rightarrow \begin{cases} r_t = \mu + \varepsilon_t + \theta_1 r_{t-1} \\ \varepsilon_t = z_t \sqrt{h_t} (\sim N(0, h_t)) \\ z_t \sim N(0,1) \\ h_t = \omega + \alpha_1 * \varepsilon_{t-1}^2 \end{cases}$$

$$GARCH(1,1) + ARMA(1,1) \rightarrow \begin{cases} r_t = \mu + \varepsilon_t + \theta_1 r_{t-1} + \alpha_i \varepsilon_{t-i} \\ \varepsilon_t = z_t \sqrt{h_t} (\sim N(0, h_t)) \\ z_t \sim N(0,1) \\ h_t = \omega + \alpha_1 * \varepsilon_{t-1}^2 + \beta_1 h_{t-1}^2 \end{cases}$$

2-modèles SARIMA : (Seasonal ARIMA models)

$$ARIMA(p, d, q)(P, D, Q)_m \rightarrow \Phi(B^S) \varphi(B) (x_t - \mu) = \theta(B^S) \theta(B) w_t$$

$$\rightarrow \text{avec} \begin{cases} \varphi(B) = 1 - \varphi_1 B - \dots - \varphi_p B^p \\ \theta(B) = 1 + \theta_1 B + \dots + \theta_q B^q \\ \Phi(B^S) = 1 - \phi_1 B^S - \dots - \phi_P B^{PS} \\ \Theta(B^S) = 1 + \theta_1 B^S + \dots + \theta_Q B^{QS} \end{cases}$$

Avec :

p = ordre de AR non saisonnier,

d = différentiation non saisonnière

q = ordre de MA non saisonnier

P = ordre de AR saisonnier

D = différentiation saisonnière

Q = ordre de MA saisonnier

S = durée de répétition de pattern saisonnier

$$ARIMA(1, 1, 1)(0,1,1)_{12}$$

$$ARIMA(0,1,1)(0,1,1)_{12}$$

$$ARIMA(1,1,1)(0,1,0)_{12}$$

$$ARIMA(0,1,1)(0,1,1)_{12}$$

Réponse :

Pour effectuer des prédictions, on commence d'abord par le test de l'effet de ARCH en utilisant la fonction « arch_effect_test » du fichier « main.R » :

Tableau 5: Test de l'effet de ARCH et de stationnarité

	A	B	C	D	E	F	G
1	Période	P.valeur	Test.auto.correlation	P.valeur.1	Test.stationarite	Modele	
2	Période.11	0.0913841362044988	non auto-correlee	0.692146602670615	non stationnaire	ARIMA/SARIMA	
3	Période.12	0.567526091290911	non auto-correlee	0.01	stationnaire	ARIMA/SARIMA	
4	Période.13	0.879958125623067	non auto-correlee	0.0201193044722296	stationnaire	ARIMA/SARIMA	
5	Période.14	0.0352820418111738	auto-correlee	0.01	stationnaire	ARCH/GARCH	
6	Période.15	0.0418755247782052	auto-correlee	0.0146925589209756	stationnaire	ARCH/GARCH	
7	Période.16	0.773797448700151	non auto-correlee	0.01	stationnaire	ARIMA/SARIMA	
8	Période.17	0.918196737412545	non auto-correlee	0.01	stationnaire	ARIMA/SARIMA	
9	Période.18	0.578934978172452	non auto-correlee	0.01	stationnaire	ARIMA/SARIMA	
10	Période.19	0.786445427889348	non auto-correlee	0.01	stationnaire	ARIMA/SARIMA	
11	Période.110	0.62040796445395	non auto-correlee	0.01	stationnaire	ARIMA/SARIMA	
12	Période.111	0.839025783538376	non auto-correlee	0.01	stationnaire	ARIMA/SARIMA	
13	Période.112	0.543778585652966	non auto-correlee	0.01	stationnaire	ARIMA/SARIMA	

Les valeurs « P.valeur » nous aident à indiquer si on a le ARCH « effect ». Dans notre cas s'elles sont inférieures à 5% on déduit qu'on peut passer à utiliser les modèles ARCH/GARCH.

On teste la stationnarité afin de savoir si on peut utiliser la série originale ou bien on doit effectuer une transformation pour la rendre stationnaire (*Ex : Dérivée*).

Le résultat de cette opération sera automatiquement sauvegardé dans le fichier « test_effet_arch_stationarite.xlsx » du répertoire courant.

Puis on procède aux prédictions :

Les résultats de prédictions seront automatiquement sauvegardés dans le fichier «predictions.xlsx» en lançant la fonction « prediction » du programme « main.r », qui va calculer un intervalle de prédiction en supposons que l'estimation de l'erreur est normalement distribué, et on calcule l'intervalle comme suit :

$$y_t \mp \alpha \sigma$$

Avec α est le coefficient correspondant au pourcentage de prédictions utilisé. Dans cet exercice on travaille avec un pourcentage de 95% qui correspond au coefficient $\alpha = 1.96$, et σ est l'écart type des résidus.

Les résultats de prédiction seront comme suit :

Dans le cas d'un exemple où la série est stationnaire (*Ex : Feuille2, période 2*) :

Tableau 6: Prédictions de la deuxième feuille de données

27	2008	4.098	1.715	16.778	0.979	4.293	20.373	13.246
28	2009	1.233	0.285	0.447	75.578	12.481	135.884	32.194
29	2010	0	0.21	40.627	30.098	11.252	4.143	27.427
30	2011	0	0	0	0	0	0	0
31	2012	3.922	17.676	8.933	0.962	0.035	0.238	29.14
32	2013	0	0	0	0	0.76	0.631	0.439
33	2014	0.101	0	76.688	19.101	3.867	1.491	2.782
34	2015	[-1.16, 4.175]	[-8.128, 10.792]	[-24.038, 101.579]	[-49.996, 74.253]	[-39.522, 40.584]	[-34.556, 62.976]	[-36.79, 42.546]
35	2016	[-1.548, 3.788]	[-9.186, 9.735]	[-33.255, 92.362]	[-62.389, 61.861]	[-41.14, 38.966]	[-36.75, 60.782]	[-33.202, 46.134]
36	2017	[-1.876, 3.46]	[-2.781, 16.139]	[-36.023, 89.593]	[-61.763, 62.486]	[-37.771, 42.335]	[-42.337, 55.195]	[-33.158, 46.178]
37	2018	[-2.156, 3.179]	[-8.225, 10.695]	[-32.714, 92.903]	[-62.901, 61.348]	[-41.515, 38.591]	[-42.891, 54.641]	[-38.494, 40.842]
38	2019	[-2.23, 3.105]	[-8.731, 10.189]	[-29.331, 96.286]	[-48.311, 75.938]	[-3.468, 76.638]	[-53.423, 44.109]	[-11.679, 67.657]
39	2020	[1.799, 7.134]	[-8.483, 10.437]	[-16.096, 109.521]	[-53.984, 70.265]	[-6.807, 73.298]	[-22.018, 75.514]	[-32.552, 46.784]
40	2021	[-1.066, 4.269]	[-8.204, 10.716]	[-34.987, 90.63]	[-11.874, 112.376]	[-28.21, 51.896]	[108.418, 205.95]	[-22.098, 57.238]
41	2022	[-2.299, 3.036]	[-9.072, 9.849]	[16.361, 141.978]	[-42.842, 81.408]	[-30.563, 49.542]	[-39.486, 58.046]	[-30.294, 49.042]
42	2023	[-2.299, 3.036]	[-0.602, 18.319]	[-36.046, 89.571]	[-53.339, 70.911]	[-29.889, 50.216]	[-42.863, 54.669]	[-26.427, 52.909]
43	2024	[1.623, 6.958]	[-1.786, 17.135]	[-16.324, 109.293]	[-59.712, 64.538]	[-32.485, 47.62]	[-44.526, 53.006]	[-29.461, 49.875]
44	2025	[-2.299, 3.036]	[-7.742, 11.179]	[-34.206, 91.411]	[-59.366, 64.884]	[-31.744, 48.361]	[-44.262, 53.27]	[-32.146, 47.19]

Et dans le cas non stationnaire, on peut voir l'exemple de la première feuille, période 1 :

Tableau 7: Prédictions de la première feuille de données

5	2014	11.871	1.624	200.028	51.23	0.597	2.869	13.4	3.189
6	2015	[0.807, 30.549]	[0.773, 70.592]	[-97.704, 141.822]	[-52.747, 90.018]	[-33.705, 54.743]	[-97.793, 187.054]	[-105.874, 152.31]	[-55.4, 60.397]
7	2016	[-10.614, 19.127]	[-28.066, 41.753]	[-13.309, 226.216]	[-52.747, 90.018]	[-33.705, 54.743]	[-126.712, 158.13]	[-100.496, 157.69]	[-63.838, 51.959]
8	2017	[-19.649, 10.092]	[-16.702, 53.117]	[-62.748, 176.777]	[-52.747, 90.018]	[-33.705, 54.743]	[-113.978, 170.86]	[-86.072, 172.121]	[-57.116, 58.68]
9	2018	[-9.963, 19.779]	[-9.995, 59.825]	[-46.671, 192.854]	[-52.747, 90.018]	[-33.705, 54.743]	[-128.42, 156.427]	[-118.589, 139.60]	[-56.227, 59.57]
0	2019	[-9.954, 19.787]	[-24.744, 45.076]	[-50.669, 188.856]	[-52.747, 90.018]	[-33.705, 54.743]	[-128.167, 156.68]	[-118.585, 139.60]	[-59.467, 56.33]
1	2020	[4.946, 34.688]	[-22.168, 47.652]	[-39.375, 200.15]	[-52.747, 90.018]	[-33.705, 54.743]	[-101.008, 183.83]	[-88.656, 169.537]	[-62.899, 52.898]
2	2021	[-12.662, 17.079]	[-2.31, 67.509]	[-47.579, 191.946]	[-52.747, 90.018]	[-33.705, 54.743]	[232.48, 517.327]	[2.05, 260.244]	[-53.002, 62.795]
3	2022	[-7.413, 22.328]	[-27.268, 42.552]	[74.136, 313.661]	[-52.747, 90.018]	[-33.705, 54.743]	[-101.372, 183.47]	[-16.694, 241.5]	[-15.003, 100.794]
4	2023	[-16.05, 13.691]	[-12.364, 57.455]	[-48.849, 190.676]	[-52.747, 90.018]	[-33.705, 54.743]	[-127.505, 157.34]	[-118.587, 139.60]	[-52.363, 63.434]
5	2024	[-11.825, 17.916]	[30.207, 100.026]	[-17.384, 222.142]	[-52.747, 90.018]	[-33.705, 54.743]	[-128.139, 156.70]	[-20.958, 237.236]	[-46.326, 69.471]
6	2025	[1.687, 31.428]	[-27.263, 42.556]	[-52.394, 187.131]	[-52.747, 90.018]	[-33.705, 54.743]	[-128.438, 156.40]	[-118.587, 139.60]	[-63.99, 51.807]
7	2026	[1.916, 31.657]	[-25.639, 44.18]	[147.673, 387.198]	[-52.747, 90.018]	[-33.705, 54.743]	[-125.569, 159.27]	[-105.187, 153.00]	[-60.801, 54.996]
8	2027	[5.723, 35.465]	[8.419, 78.239]	[-30.314, 209.211]	[-52.747, 90.018]	[-33.705, 54.743]	[-83.808, 201.039]	[-95.364, 162.829]	[-61.491, 54.305]
9	2028	[-5.699, 24.043]	[-20.42, 49.4]	[54.089, 293.614]	[-52.747, 90.018]	[-33.705, 54.743]	[-112.727, 172.12]	[-89.986, 168.208]	[-69.93, 45.867]
0	2029	[-14.734, 15.008]	[-9.056, 60.763]	[4.646, 244.171]	[-52.747, 90.018]	[-33.705, 54.743]	[-99.993, 184.854]	[-75.562, 182.631]	[-63.208, 52.589]
1	2030	[-5.047, 24.695]	[-2.348, 67.471]	[20.725, 260.25]	[-52.747, 90.018]	[-33.705, 54.743]	[-114.435, 170.41]	[-108.079, 150.11]	[-62.318, 53.478]

Remarque générale sur les résultats :

- ✓ Dans le fichier du résultat de prédictions « predictions.xlsx », on trouve que les données d'entraînements utilisées à chaque fois dans une feuille sont soit les données initiales au cas de stationnarité ou les données dérivées au cas d'absence de stationnarité.
- ✓ Ainsi durant tout le processus, les courbes et les figures seront exportés dans des fichiers « pdf » dans le répertoire courant.
- ✓ Un exemple de courbe de prédictions en utilisant les données initiales sera comme suit :

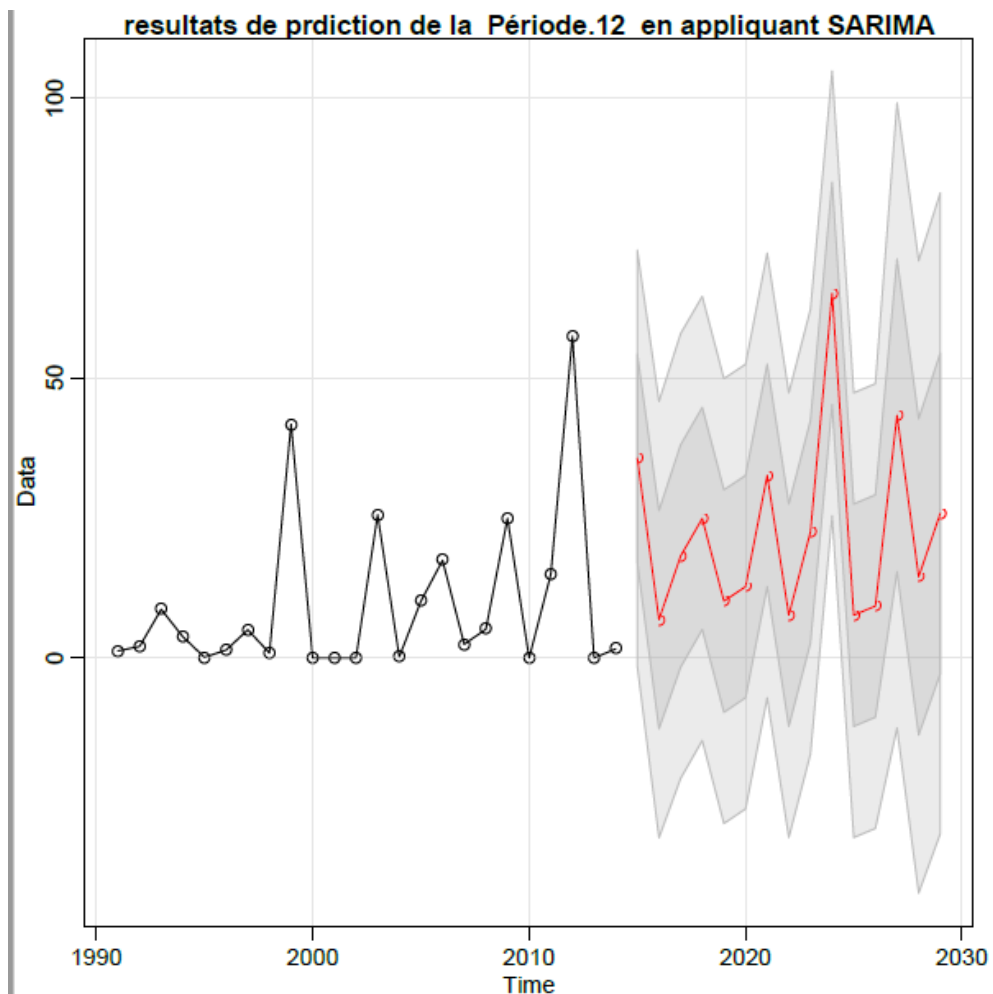


Figure13 : Courbe de prédictions SARIMA au cas de stationnarité

Un autre en utilisant les données dérivées est comme suit :

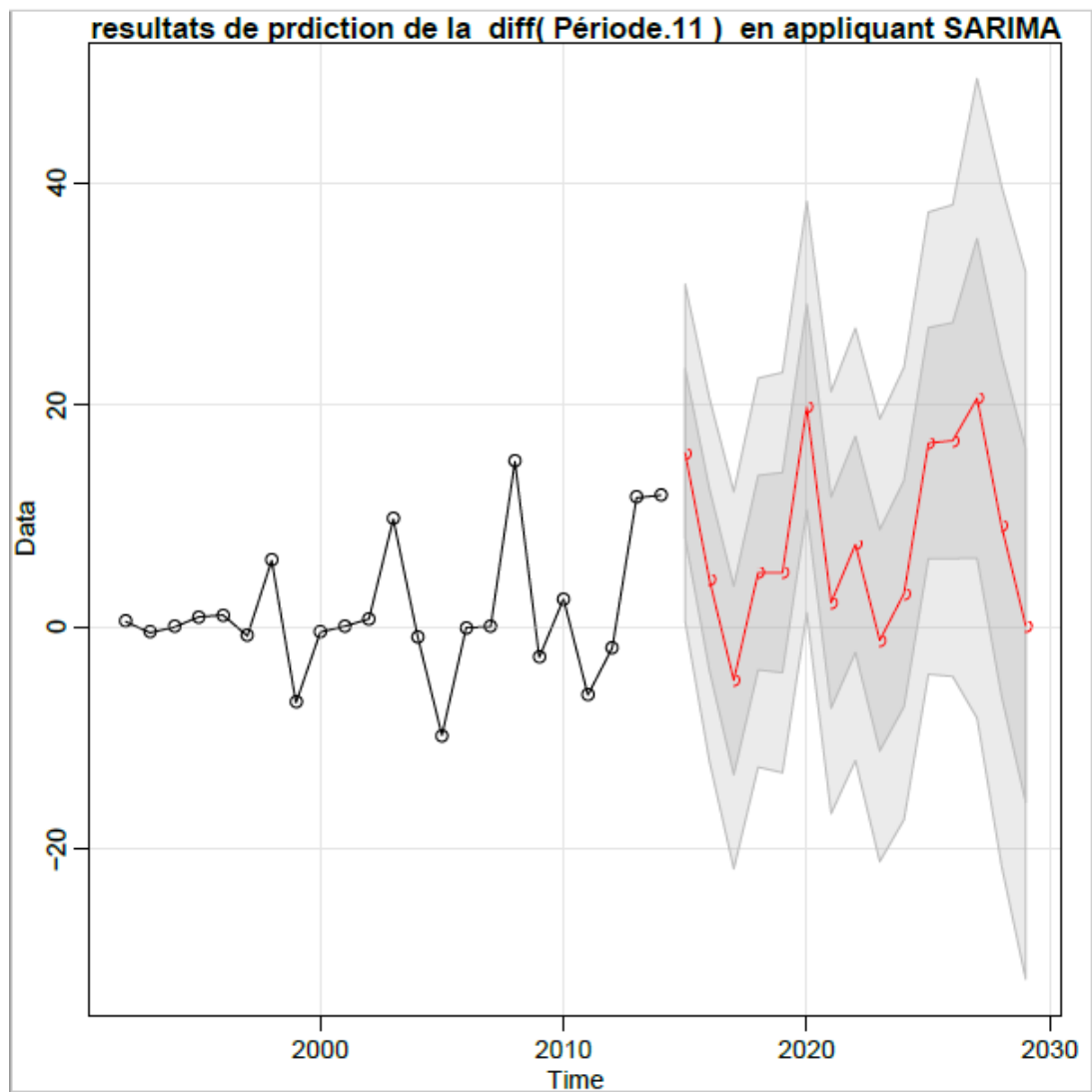


Figure 14: Courbe de prédictions SARIMA au cas de non stationnarité

- ✓ On remarque aussi que la prédiction utilisant un modèle GARCH donne des valeurs à peu près constantes comme suit :

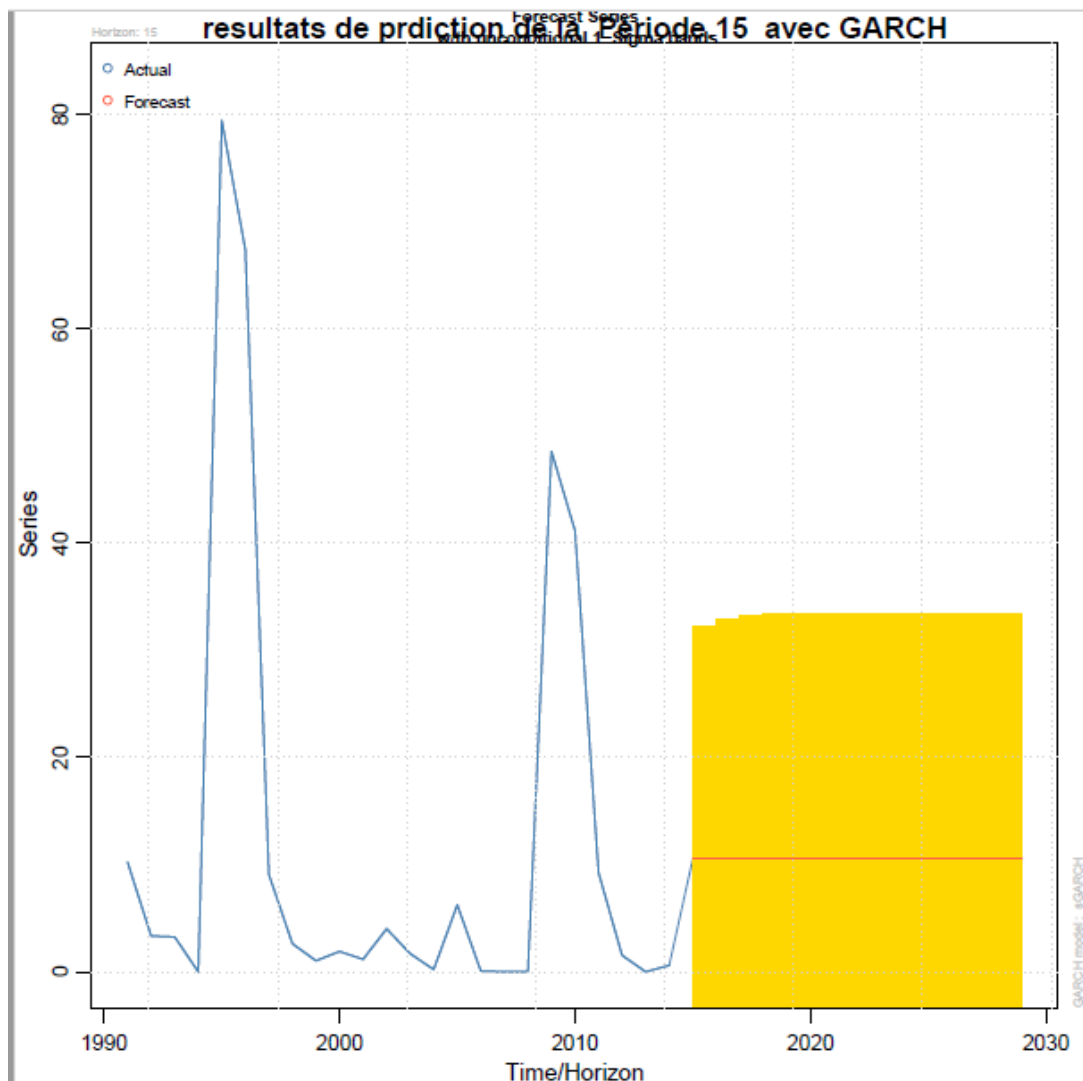


Figure 15:Prédiction de la période 15 avec GARCH

Ce qui est due au nombre non suffisant des données initiales d'entraînement. Avec le modèles GARCH on est toujours besoins d'au moins 100 données avant de commencer les prédictions, sinon le modèles sera incapable de prédire des bons résultats et la marge d'erreur devient plus grande.

- ✓ Pour la quatrième feuille du fichier Excel, on a pas assez de données ce qui nous empêche d'utiliser le modèle SARIMA. Pour ARIMA la prédiction aussi sera mauvaise et donne seulement des données constantes comme suit :

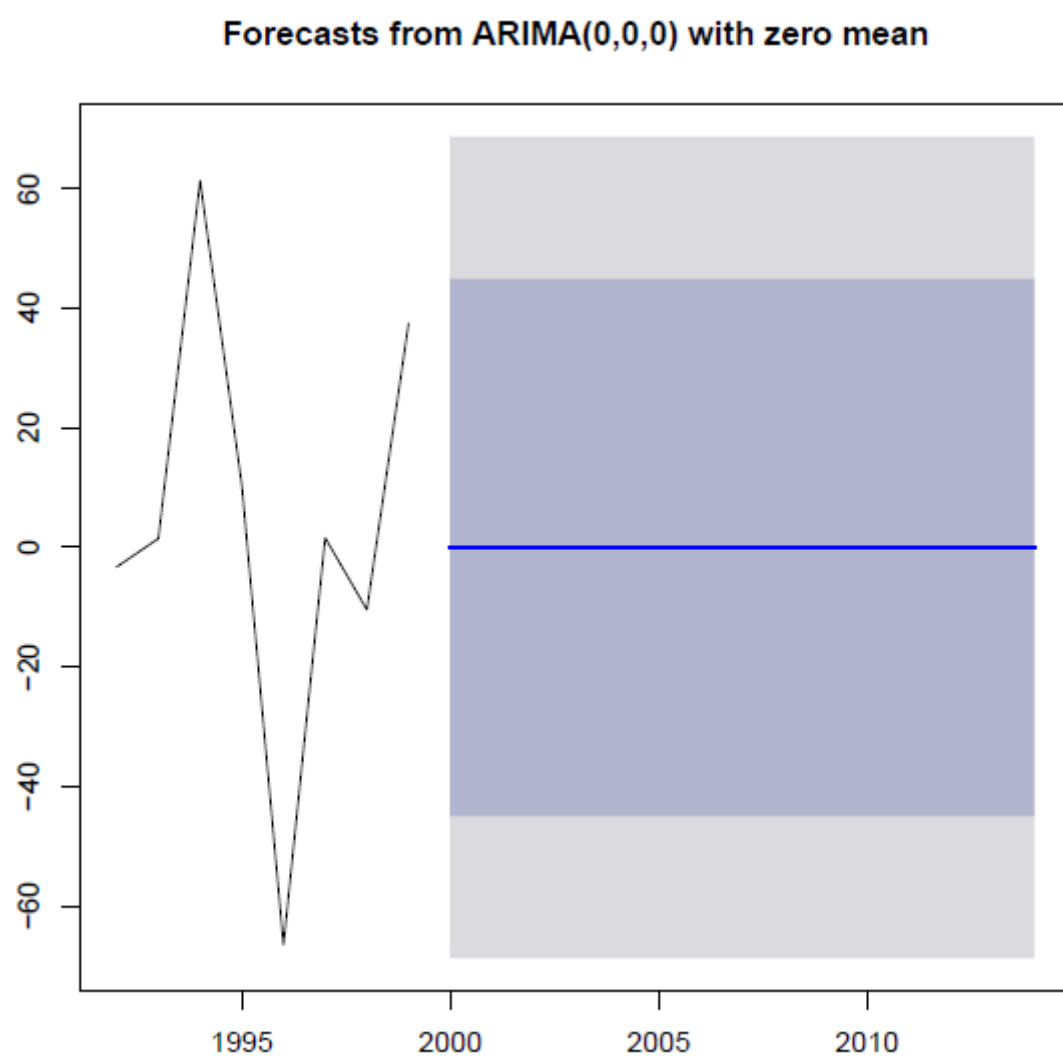


Figure 16: Prédiction ARIMA(0,0,0)

Conclusion

Dans ce travail on a essayé d'introduire les modèles ARCH et GARCH et les utiliser pour simuler et prédire des estimations futures sous le logiciel RStudio.

La volatilité des marchés existait depuis toujours à cause de plusieurs facteurs comme les comportements des investisseurs et les bulles spéculatives. Les modèles ARCH et GARCH malgré leurs limites quand il s'agit de grandes masses de données ou de trop petites bases de connaissances, ils nous permettent de modéliser des séries temporelles complexes avec moins de paramètres et ils sont très utiliser pour prédire la volatilité des marchés.

Références

[1] ShengweiWang, JuanFeng ET GangLiu, Application of seasonal time series model in the precipitation forecast, 2011

[2] YVES ARAGON: séries temporelles avec R, méthodes et cas. Édition : Springer. 2011

[3] ENGLE R.F, The Use of ARCH/GARCH Models in Applied Econometrics, Journal of Economic Perspectives, 2001