



## École Nationale Supérieure de Techniques Avancées

---

STA202 - Projet d'analyse de série chronologique

### Analyse du cours de l'action d'Apple

---

*Auteurs :* Selim-Antoine Lali - Charlène Krick

*Professeur :* Yannig Goude

29 février 2024

# Sommaire

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>Introduction</b>		<b>2</b>
<b>2</b>	<b>Prétraitement et mise en forme des données</b>	<b>2</b>
2.1	Importation des données . . . . .	2
2.2	Mise en forme des données . . . . .	2
<b>3</b>	<b>Analyse descriptive des données</b>	<b>3</b>
3.1	Visualisation d'ensemble de notre jeu de données . . . . .	3
3.2	Analyse statistique de la série temporelle . . . . .	3
<b>4</b>	<b>Modélisation de la série chronologique des données</b>	<b>6</b>
4.1	Décomposition de la série temporelle . . . . .	6
4.2	Estimation de la tendance . . . . .	7
4.2.1	Estimation de la tendance par moyenne mobile . . . . .	7
4.2.2	Estimation de la tendance par régression linéaire . . . . .	8
4.2.3	Estimation de la tendance par noyau gaussien . . . . .	9
4.2.4	Estimation de la tendance par polynômes locaux . . . . .	10
4.2.5	Estimation semi-paramétrique de la tendance : projection sur des bases de fonctions splines polynomiales par morceau . . . . .	11
4.2.6	Comparaison des méthodes d'estimation de tendance . . . . .	12
4.3	Estimation de la saisonnalité . . . . .	12
4.3.1	Estimation par moyenne mobile . . . . .	12
4.3.2	Estimation par régression de Fourier . . . . .	13
4.3.3	Estimation par noyau gaussien . . . . .	13
4.3.4	Estimation par polynômes locaux . . . . .	14
4.3.5	Estimation par projection sur base de splines . . . . .	14
4.4	Analyse de la composante résiduelle . . . . .	15
4.4.1	Visualisation des résidus et étude de la stationnarité . . . . .	15
4.4.2	Caractère normal des résidus . . . . .	16
<b>5</b>	<b>Simulation de prévision sur un échantillon test</b>	<b>17</b>
5.1	Lissages exponentiels simples . . . . .	17
5.1.1	Application de la méthode . . . . .	17
5.1.2	Prévision sur les données . . . . .	18
5.2	Lissage exponentielle double (Holt) . . . . .	19
5.2.1	Application de la méthode . . . . .	19
5.3	Lissage exponentielle de Holt-Winters . . . . .	20
5.3.1	Application de la méthode . . . . .	20
5.3.2	Prévision sur les données . . . . .	20
<b>6</b>	<b>Méthode d'ARMA</b>	<b>21</b>
6.1	Principe des modèles . . . . .	21
6.1.1	Modèle autorégressif . . . . .	21
6.1.2	Modèle moyenne mobile . . . . .	21
6.1.3	Modèle ARMA . . . . .	21
6.1.4	Choix du modèle . . . . .	22
6.2	Application à notre jeu de données . . . . .	22
6.2.1	Implémentation du modèle . . . . .	22
6.2.2	prévision avec le modèle ARMA . . . . .	22
<b>7</b>	<b>Conclusion</b>	<b>23</b>

# 1 Introduction

Dans le cadre de ce projet, nous allons nous initier à l'analyse de séries chronologiques en effectuant une série de manipulations sur un jeu de données choisi. Nous nous sommes intéressés au jeu de données suivant : évolution du cours de l'action de l'entreprise Apple, une des plus grandes multinationales dans le domaine des produits électroniques. Ce dataset provient de la librairie "quantmod" de Rstudio et la source des données est "Yahoo". Il comporte plusieurs données d'action, dont la valeur quotidienne de l'action d'Apple à la fermeture des marchés financiers depuis 2007 et jusqu'à 2024.

L'analyse de ce dataset revêt un intérêt majeur. Apple est une des entreprises les plus valorisées au monde, avec des activités influençant divers secteurs technologiques et financiers. Comprendre la dynamique de son action permet d'appréhender les réactions du marché aux annonces de produits, aux résultats financiers, et aux tendances économiques globales.

## 2 Prétraitement et mise en forme des données

### 2.1 Importation des données

Les données que nous allons utiliser seront importé sur Rstudio via le package "quantmod" de R qui permet d'avoir les données du cours de l'action de plusieurs entreprises dans le monde. La source est par défaut Yahoo.

L'appel de la fonction `getSymbols("AAPL")` permet d'obtenir les données boursières de l'entreprise Apple. Pour obtenir une vue globale des données, nous avons utilisé la fonction `head` et `str` sur le tableau de données pour visualiser les informations rapides sur le dataset. Nous obtenons les informations suivantes :

```
> getSymbols("AAPL")
[1] "AAPL"
> head(AAPL,n=3)
  AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
2007-01-03  3.081786  3.092143  2.925000   2.992857 1238319600      2.533751
2007-01-04  3.001786  3.069643  2.993571   3.059286  847260400      2.589990
2007-01-05  3.063214  3.078571  3.014286   3.037500  834741600      2.571545
>
> str(AAPL)
An xts object on 2007-01-03 / 2024-02-26 containing:
  Data:    double [4316, 6]
  Columns: AAPL.Open, AAPL.High, AAPL.Low, AAPL.Close, AAPL.Volume ... with 1 more co
lumn
  Index:    Date [4316] (TZ: "UTC")
  xts Attributes:
    $ src    : chr "yahoo"
    $ updated: POSIXct[1:1], format: "2024-02-27 05:47:31"
```

Visual Studio Code

FIGURE 1 – Vue globale des données

Ce sont des données de prix OHLCVA. Nous avons les prix d'ouverture, haut, bas, clôture, volume et les prix de clôture ajustés. Les prix haut et bas sont les prix les plus élevés et les plus bas de la journée de trading. Les prix d'ouverture et de clôture sont les prix auxquels le marché ouvre et ferme pendant la journée de trading. Le volume est le nombre d'actions transigées pendant la journée de trading. Le prix de clôture ajusté est le prix de clôture qui est ajusté pour des événements survenant après la fermeture du marché.

Nous avons donc un jeu de données comportant 4316 observations sur les 6 variables mentionnées ci-dessus.

### 2.2 Mise en forme des données

Nous remarquons que le tableau "AAPL" possède 6 variables qui présentent chacune un type de prix différent. Nous allons nous intéresser dans le cadre de ce projet au prix de l'action à la clôture des marchés. Nous avons donc créé à l'aide de la fonction `data.frame` un nouveau tableau de données composé uniquement des dates "Année-mois-jour" et de la colonne AAPL.Close. La variable des dates est déjà, après importation, sous le format Année-mois-jour.

### 3 Analyse descriptive des données

#### 3.1 Visualisation d'ensemble de notre jeu de données

Utilisons la fonction **plot** de R afin de visualiser la courbe du prix de l'action à la clôture en fonction de la date :

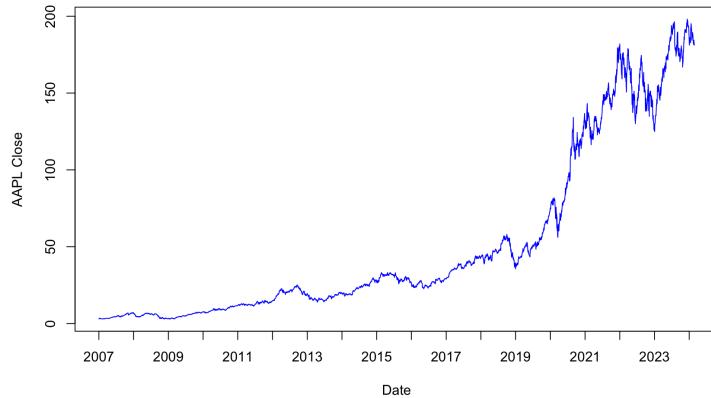


FIGURE 2 – Prix quotidien de l'action d'Apple à la fermeture des marchés de 2007 à 2024

#### 3.2 Analyse statistique de la série temporelle

En utilisant la fonction **xts**, nous avons converti les données de la Data Frame en une série temporelle. Ci-dessous le tracé de cette série :



FIGURE 3 – Graphique de la série temporelle du prix de l'action d'Apple

Il semble y avoir une tendance à long terme (Trend) positive, c'est-à-dire que le prix de l'action Apple a augmenté au fil du temps. On constate toutefois depuis 2019 des périodes de forte volatilité, ce qui pourrait correspondre à des événements importants de l'entreprise ou des crises économiques mondiales comme la crise financière de 2008 ou la pandémie COVID-19. De plus, il paraît difficile de discerner une saisonnalité claire à partir de ce graphique, elle pourrait se manifester par une analyse plus approfondie des motifs répétitifs annuels ou trimestriels. Enfin, au vu de ce tracé, il sera intéressant d'étudier les valeurs aberrantes ou les pics qui ne correspondent pas à la tendance générale afin d'identifier des événements spécifiques ou des erreurs dans les données.

Le logiciel RStudio nous permet également d'obtenir les propriétés statistiques de la série temporelle afin d'effectuer une analyse de base de la série : moyenne, écart-type, boxplot, histogramme, moyenne mensuelle et annuelle de la série...

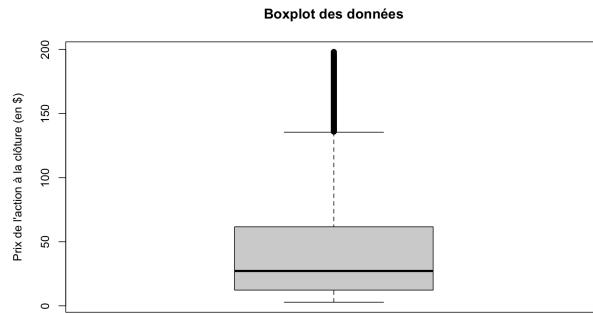
```

> # Calculer la moyenne et l'écart-type
> mean_AAPL <- mean(AAPL_xts)
> sd_AAPL <- sd(AAPL_xts)
> summary(AAPL_Close_df)
   Date           AAPL_Close
Min. :2007-01-03  Min. : 2.793
1st Qu.:2011-04-13 1st Qu.:12.277
Median :2015-07-29 Median :27.194
Mean  :2015-07-29  Mean :51.415
3rd Qu.:2019-11-08 3rd Qu.:61.651
Max. :2024-02-26  Max. :198.110
> # Afficher les valeurs
> print(paste("La moyenne est :", mean_AAPL))
[1] "La moyenne est : 51.4150097643598"
> print(paste("L'écart-type est :", sd_AAPL))
[1] "L'écart-type est : 55.7412101217823"

```

FIGURE 4 – Propriétés statistiques de la série temporelle

Nous avons aussi tracé le **boxplot** de la série temporelle afin de voir la répartition des données de la valeur de l'action :



La moyenne de la valeur quotidienne de l'action d'Apple depuis 2007 est de 51.415 \$, d'écart-type de 55.741 \$. La plus basse valeur enregistrée de l'action est 2.793\$ enregistrée en novembre 2008. La plus haute valeur de l'action est de 198.110 \$, et a été enregistré en octobre 2023.

Nous avons tracé l'**autocorrélogramme** de la série temporelle :

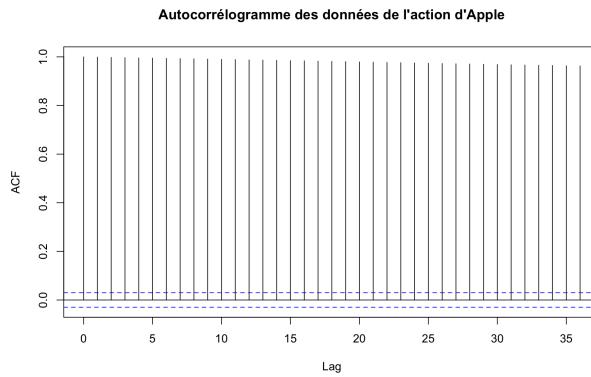


FIGURE 5 – Autocorrélogramme des données

Ce graphique montre l'autocorrélation de la série temporelle des données de l'action d'Apple pour différents décalages temporels (lags).

**Autocorrélation à différents lags** : la valeur de l'autocorrélation (ACF) est très élevée pour les premiers lags et diminue à mesure que le lag augmente. Cela suggère une forte dépendance temporelle des observations rapprochées dans le temps.

**Déclin de l'autocorrélation** : l'autocorrélation décroît avec le lag. Cela pourrait indiquer un processus où les valeurs passées ont un impact décroissant sur les valeurs futures au fur et à mesure que l'intervalle de temps s'allonge.

**Significativité statistique** : les lignes bleues horizontales représentent les seuils de significativité pour l'autocorrélation au niveau de confiance 95%. Nos autocorrélations dépassent toutes ces limites. Elles sont donc statistiquement significatives, indiquant que l'autocorrélation observée à ces lags n'est probablement pas due au hasard.

**Comportement de la série temporelle** : le fait que l'autocorrélation soit initialement positive et significative pour les premiers lags suggère que la série temporelle peut présenter une certaine inertie, ce qui est une caractéristique commune dans les données financières où les prix peuvent suivre des tendances sur de courtes périodes.

**Absence d'autocorrélation à long terme** : l'autocorrélation devient très faible et insignifiante pour les lags plus élevés. Cela suggère que les valeurs de la série temporelle deviennent indépendantes les unes des autres à mesure que l'intervalle de temps s'allonge, ce qui est une caractéristique attendue dans de nombreuses séries temporelles économiques et financières.

Afin de mieux observer la tendance de notre jeu de données, nous avons représenté la moyenne mensuelle et annuelle de la valeur de l'action d'Apple. Ainsi, nous avons considérablement réduit le nombre de points tracés et cela donne un aperçu de la tendance générale :

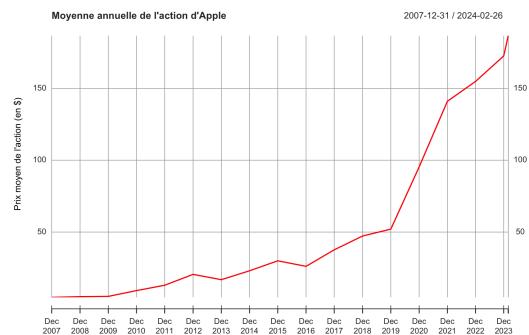
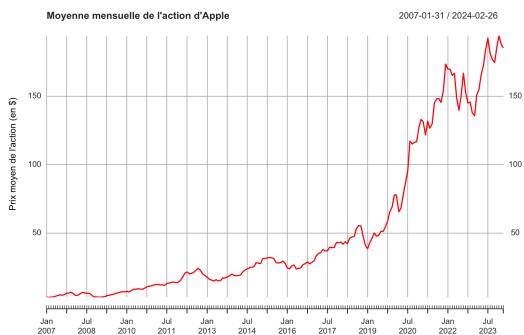


FIGURE 6 – Moyenne mensuelle de l'action d'Apple

FIGURE 7 – Moyenne annuelle de l'action d'Apple

Le cours de l'action d'Apple montre une tendance générale à la hausse sur la période observée. Toutefois, il y a des périodes de volatilité importantes où les prix fluctuent plus rapidement, par exemple, des pics et des creux plus prononcés vers 2013 et entre 2020 et 2022. D'autre part, on peut observer des plateaux où le prix semble se stabiliser avant de soit monter soit descendre brusquement.

D'un point de vue quantitatif, la valeur de l'action a connu une croissance significative, partant d'un niveau inférieur à 50 \$ en 2007 pour atteindre des niveaux supérieurs à 150 \$ en 2024. Enfin, les fluctuations de prix semblent s'amplifier avec le temps, indiquant une augmentation de la volatilité au fil des années.

Quelques éléments du contexte historique peuvent nous aider à comprendre certaines tendances des graphiques ci-dessus. En 2020, la pandémie de COVID-19 a entraîné une chute des marchés mondiaux, suivie d'une reprise rapide pour les entreprises technologiques alors que la demande pour le télétravail et le divertissement numérique explosait. En 2021-2022, Une combinaison de facteurs y compris la reprise après la pandémie, des résultats financiers solides, et l'annonce de nouveaux produits ou services, a pu entraîner une hausse continue du cours de l'action.

Enfin, nous avons représenté les fréquences de la valeur de l'action dans notre jeu de données à l'aide

d'un histogramme :

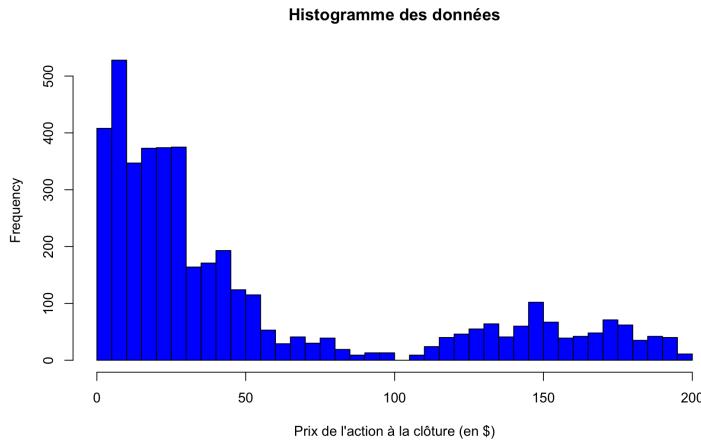


FIGURE 8 – Histogramme de la valeur de l'action de l'Apple

Dans cette distribution des prix de clôture de l'action Apple, la valeur la plus fréquente est dans la deuxième barre, qui est la plus haute, située entre 0 et 10 dollars. La moyenne et la médiane se situent bien dans la plage de prix où l'histogramme est le plus dense (0-50\$). L'étendue et la forme de l'histogramme confirment une variance significative de notre série (55\$) indiquant que les prix varient considérablement sur la période analysée.

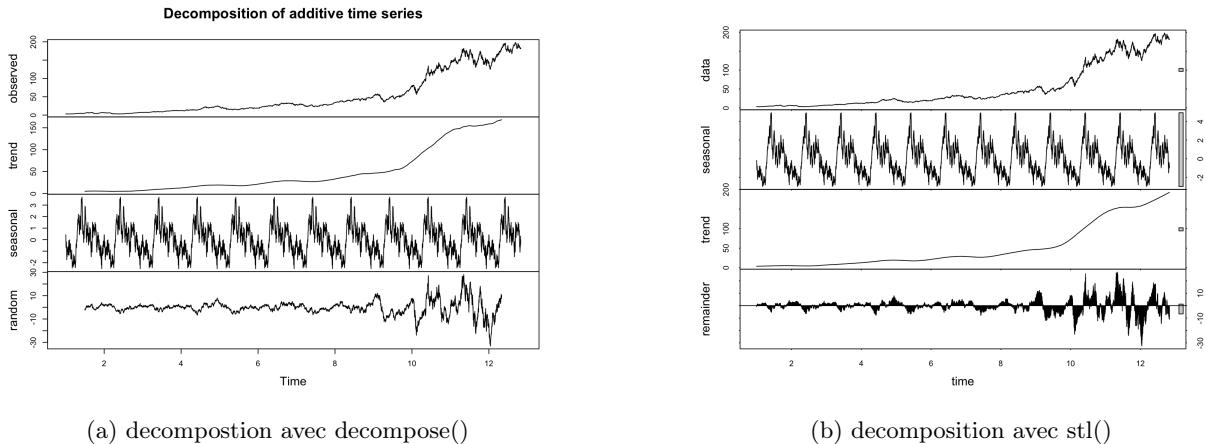
## 4 Modélisation de la série chronologique des données

### 4.1 Décomposition de la série temporelle

Nous allons décomposer notre série temporelle en trois composantes : la tendance (trend) qui correspond à l'évolution à long terme de la série temporelle, le cycle ou saisonnalité qui correspond aux phénomènes périodiques identifiés, et enfin la partie aléatoire (random) de la série temporelle.

Sur RStudio, nous avons décomposé notre série temporelle en trois composantes à l'aide de la fonction **decompose** et la fonction **stl()**. La figure 9 montre les composantes observées de notre série temporelle analysée à l'aide des fonctions **decompose()** et **stl()** :

FIGURE 9 – Décomposition de la série temporelle



Pour décomposer cette série, la fonction **decompose()** de R est conçue pour les séries temporelles avec une saisonnalité connue. Etant donné que nous ne connaissons pas clairement la saisonnalité, nous pouvons utiliser la méthode plus flexible de la décomposition STL (Seasonal and Trend decomposition using Loess), avec la fonction **stl()**. Cette fonction est plus flexible et peut gérer des séries temporelles sans saisonnalité.

claire ou avec des changements de saisonnalité.

**Composante de tendance :** les graphiques montrent une tendance croissante sur la période observée. Cette tendance positive est due à la croissance globale de l'entreprise, à son expansion sur de nouveaux marchés, à des innovations réussies et à une performance financière solide. Nous pouvons également remarquer une ressemblance entre la moyenne annuelle du prix de l'action tracée **figure 7** et la composante tendancielle de la série temporelle. La moyenne annuelle donne donc un bon aperçu de la tendance.

**Composante saisonnière :** la composante saisonnière de la **décomposition STL** semble être plus complexe ce qui peut indiquer des variations saisonnières plus subtiles ou des effets de calendrier. Toutefois, on constate clairement l'apparition d'un motif. Cela pourrait refléter des comportements d'achat périodiques, des cycles de produits, ou des événements réguliers qui affectent le cours de l'action. **La décomposition classique (avec decompose())** montre également ce motif qui pourrait indiquer des cycles saisonniers réguliers. La fréquence de décomposition annuelle semble adapté pour étudier la saisonnalité.

**La composante aléatoire :** elle montre les fluctuations non expliquées par la tendance ou la saisonnalité. Cela inclut le bruit, les erreurs ou les événements aléatoires non prévisibles. On remarque une incertitude importante ces 4 dernières années, comparativement aux années 2000s. En effet, les pics et les creux dans cette composante sont le résultat d'événements spécifiques tels que des annonces d'entreprise, des changements réglementaires, des crises économiques, ou encore la crise du Covid-19 qui influencent le cours de l'action.

Après cette première analyse graphique, le modèle de décomposition le plus convenable est le modèle additif. la série notée  $X_t$  s'écrit :

$$X_t = T_t + S_t + \epsilon_t$$

Dans un premier temps, nous allons nous intéresser à la modélisation des composantes déterministes de la série :  $T_t$  et  $S_t$ .

## 4.2 Estimation de la tendance

Il existe différentes méthodes pour estimer la tendance d'une série temporelle. Nous allons les tester sur notre série afin de sélectionner la meilleure méthode.

### 4.2.1 Estimation de la tendance par moyenne mobile

La moyenne mobile, également connue sous le nom de moyenne glissante, est une technique qui vise à atténuer les variations à court terme de la série en calculant une moyenne sur chaque fenêtre de données autour de la valeur courante. Cette méthode contribue à lisser les données, mettant en évidence les tendances à long terme de la série temporelle. La moyenne mobile se calcule ainsi :

$$\hat{y}_t = \frac{1}{2l+1} \sum_{i=t-l}^{t+l} y_i$$

Sur Rstudio, il faut utiliser la fonction **filter**. Pour ces données quotidiennes, nous définissons la taille de la fenêtre pour la moyenne mobile à 30 jours.

On remarque que cette estimation figure 10 est très similaire à la courbe obtenue dans la décomposition faite par RStudio avec **decompose()**, ce qui est logique vu que la fonction **decompose** utilise le principe de moyenne mobile pour estimer la tendance.



FIGURE 10 – Estimation (en rouge) de la tendance par moyenne mobile

#### 4.2.2 Estimation de la tendance par régression linéaire

Nous avons essayé la méthode de régression linéaire à différents ordres afin d'estimer la tendance de notre série temporelle. Nous avons donc tracé figure 11 la série en bleue et sa régression linéaire en rouge :

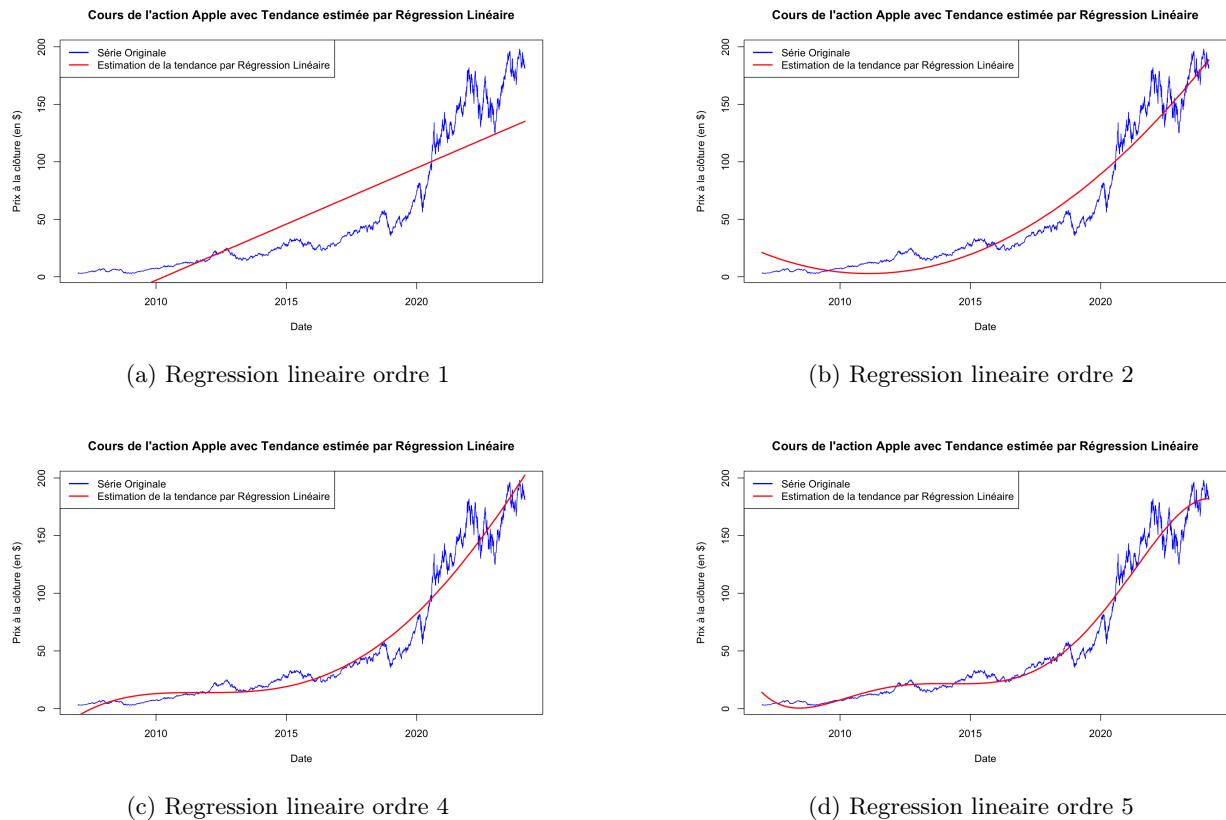


FIGURE 11 – Estimation de la tendance par régression linéaire à différents ordres

En utilisant la fonction **summary()** sur la série temporelle après application de la régression linéaire, nous obtenons les détails statistiques de notre modèle de régression sous la forme du tableau suivant :

```

Call:
lm(formula = AAPL_xts ~ time + I(time^2) + I(time^3) + I(time^4))

Residuals:
    Min      1Q  Median      3Q     Max 
-39.333 -5.505 -1.079  5.842 48.358 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -6.374e+00  8.713e-01 -7.316 3.04e-13 ***  
time        5.346e-02  2.796e-03 19.119 < 2e-16 ***  
I(time^2)   -4.890e-05 2.633e-06 -18.569 < 2e-16 ***  
I(time^3)   1.698e-08 9.165e-10 18.526 < 2e-16 ***  
I(time^4)   -1.372e-12 1.053e-13 -13.029 < 2e-16 ***  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 11.43 on 4311 degrees of freedom
Multiple R-squared:  0.958,    Adjusted R-squared:  0.9579 
F-statistic: 2.457e+04 on 4 and 4311 DF,  p-value: < 2.2e-16

```

FIGURE 12 – Summary de la régression linéaire

Dans notre modèle de régression, nous avons choisi d'utiliser une régression linéaire d'ordre 4 car la tendance de notre série n'est clairement pas simplement linéaire mais a une forme plus complexe.

Du point de vue de l'ajustement de notre modèle, le R-squared élevé (près de 0.958) indique que le modèle explique une grande partie de la variabilité des prix de l'action Apple. Un R-squared proche de 1 signifie que le modèle s'adapte très bien aux données. La p-value  $< 2.2e-16$  pour les coefficients suggère que les termes du modèle sont statistiquement significatifs et donc utiles pour le modèle.

Les signes alternés des coefficients (positif pour time et time\*\*3, négatif pour time\*\*2 et time\*\*4) indiquent une relation non linéaire et complexe entre le temps et le prix de l'action. Cela peut refléter des périodes d'accélération et de décélération dans la croissance du prix de l'action. Cela correspond à différentes phases économiques, introductions de produits et d'autres événements influençant la performance de l'entreprise.

La distribution des résidus semble centrée autour de zéro, ce qui est bon pour la régression linéaire. Cependant, la gamme des résidus (de -39.333 à 48.358) suggère qu'il pourrait y avoir de la volatilité dans les prix qui n'est pas capturée par le modèle.

Enfin, même si ce modèle s'ajuste bien, il est important de vérifier qu'il n'y ait pas de surajustement (overfitting), où le modèle serait trop spécifique aux données historiques et ne prédirait pas bien l'avenir.

#### 4.2.3 Estimation de la tendance par noyau gaussien

C'est une estimation non-paramétrique de la tendance. Dans le cas général, on a un modèle du type :

$$y_t = f(t) + \varepsilon_t$$

Si l'on fait une estimation par noyau, la fonction  $f$  est approximée sur une fenêtre  $h$  avec un noyau  $K$ .  $K : \mathbb{R} \rightarrow \mathbb{R}^d$  est un noyau si

$$\int K < \infty \quad \text{et} \quad \int K = 1$$

On appelle estimation à noyau de  $f$  associé à la fenêtre  $h$  et au noyau  $K$  la fonction  $\hat{f}_h$  telle que :

$$\hat{f}_h(x) = \frac{\sum_{t=1}^n y_t K\left(\frac{x-t}{h}\right)}{\sum_{t=1}^n K\left(\frac{x-t}{h}\right)}$$

Pour une estimation par noyau gaussien on a :  $K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$

Pour estimer et tracer la tendance de notre série temporelle en utilisant une régression à noyau gaussien, nous utilisons la fonction ksmooth du package stats en R, en sélectionnant le kernel "standard". La tendance estimée dépend du choix de la fenêtre  $h$ , également appelée bande passante, qui détermine la largeur de la

fenêtre du noyau pour l'estimation. La sélection d'un  $h$  appropriée est cruciale car elle affecte la douceur de l'estimation de la tendance. Une fenêtre trop petite peut mener à un surajustement (overfitting), tandis qu'une fenêtre trop grande peut lisser excessivement la série temporelle et masquer les caractéristiques importantes.

Nous avons donc tracé la série et son estimation par noyau gaussien figure 13, pour  $h = 250$  en rouge, puis la série après suppression de la tendance estimée :

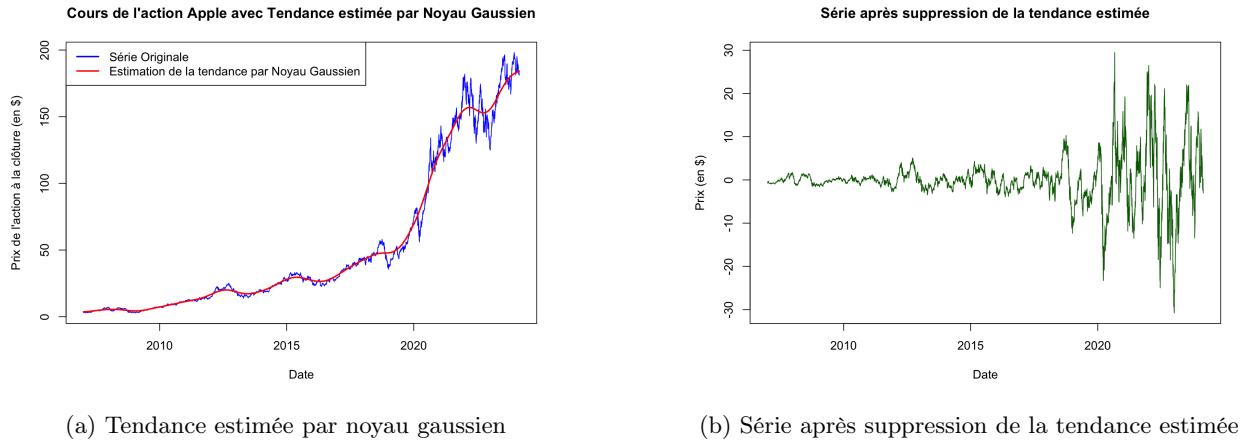


FIGURE 13 – Estimation par noyau gaussien

#### 4.2.4 Estimation de la tendance par polynômes locaux

Soit  $h > 0$  et  $q > 0$ , un estimateur polynomial local de degré  $q$  de  $f$  associé à la fenêtre  $h$  et au noyau  $K$  est la fonction  $\hat{f}_h$  définie par :

$$\hat{f}_h(x) = \arg \min_P \sum_{t=1}^n W_t(x) |y_t - P(x_t - x)|^2$$

où  $P$  est un polynôme de degré  $q$  et

$$W_t(x) = \frac{K\left(\frac{x-t}{h}\right)}{\sum_{t=1}^n K\left(\frac{x-t}{h}\right)}$$

Le principe de cette méthode non paramétrique est que, pour chaque valeur du temps, on estime une fonction polynomiale approximant les données localement. Il faut à nouveau choisir une taille de fenêtre. Autrement formulé, il s'agit d'estimer sur les données un développement limité de la fonction  $f$ .

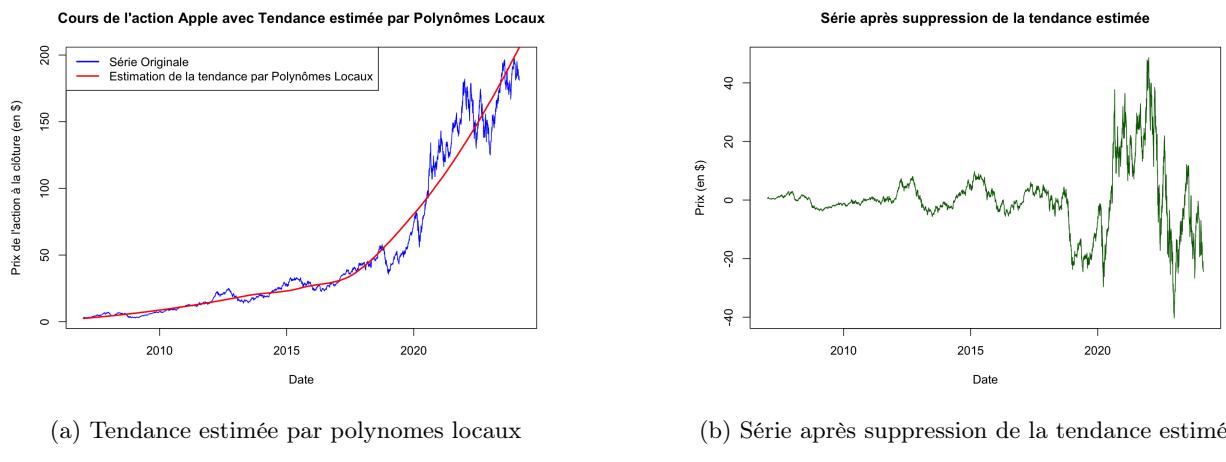


FIGURE 14 – Estimation par polynomes locaux

La fonction R implémentant les polynômes locaux est la fonction **loess**, dont le paramètre span joue le même rôle de fenêtre que le paramètre bandwidth de ksmooth. Nous avons choisi d'estimer la tendance avec une fenêtre span de 0.75, ce qui est un bon point de départ pour une série temporelle quotidienne.

Puis, nous avons tracé figure 14 la série et son estimation par polynomes locaux en rouge, puis la série après suppression de la tendance estimée.

#### 4.2.5 Estimation semi-paramétrique de la tendance : projection sur des bases de fonctions splines polynomiales par morceau

Une autre alternative pour estimer la fonction  $f$  est de procéder par projection sur des bases de fonctions adaptés, par exemple des fonctions splines polynomiales par morceau. En pratique on pourra utiliser la fonction **gam** du package mgcv pour estimer la tendance par cette méthode.

Nous avons donc tracé la série et son estimation de tendance par base de splines, puis la série après suppression de la tendance estimée, figure 15.

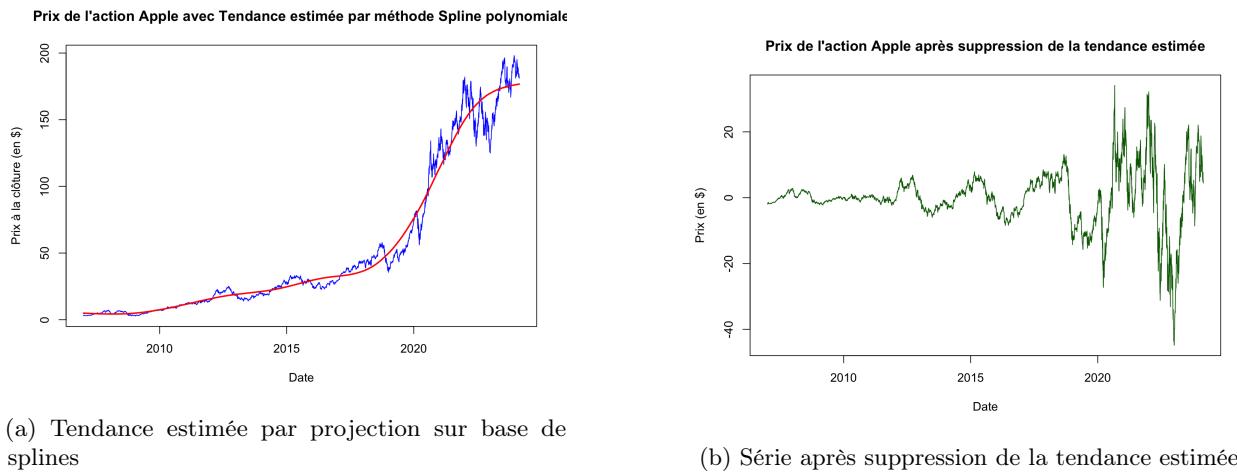


FIGURE 15 – Estimation par projection sur des bases de fonctions splines polynomiales par morceau

Concernant la base de splines choisie, nous avons pris un modèle spline cubique lisse ("cs" pour cubic spline) avec une sélection automatique du nombre de noeuds par la méthode de maximum de vraisemblance restreinte ("REML").

```
> # Pour obtenir un résumé du modèle qui inclura des informations sur les splines
> summary(gam_fit)

Family: gaussian
Link function: identity

Formula:
AAPL_xts ~ s(time, bs = "cs")

Parametric coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 51.4150    0.1266   406.3 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
edf Ref.df   F p-value
s(time) 8.966  9 21072 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq (adj) =  0.978  Deviance explained = 97.8%
-REML = 15300  Scale est. = 69.122 n = 4316
> # Pour obtenir des informations spécifiques sur la base de splines utilisée
> gam_fit$sp
s(time)
1.76611
```

(a) Summary de la base de splines choisie pour notre modèle

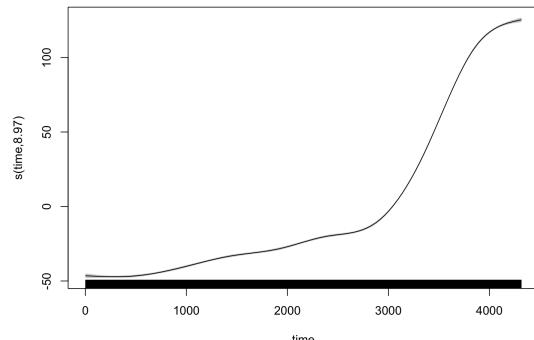


FIGURE 16 – Informations sur la base de splines choisie

Ci-dessus le résumé du modèle choisi, avec les statistiques standard de la régression, y compris l'efficacité

des termes de spline (en termes de p-valeurs et d'estimations des paramètres), ainsi que des informations sur la pénalité de lissage et le choix du paramètre de lissage. Également, des informations sur les paramètres de pénalité de lissage spécifiques que nous avons utilisés dans notre modèle. Nous avons aussi mis une visualisation de la spline pour comprendre comment elle s'adapte à nos données.

#### 4.2.6 Comparaison des méthodes d'estimation de tendance

Nous avons affiché figure 17 les estimations de tendances obtenues avec les différentes méthodes.

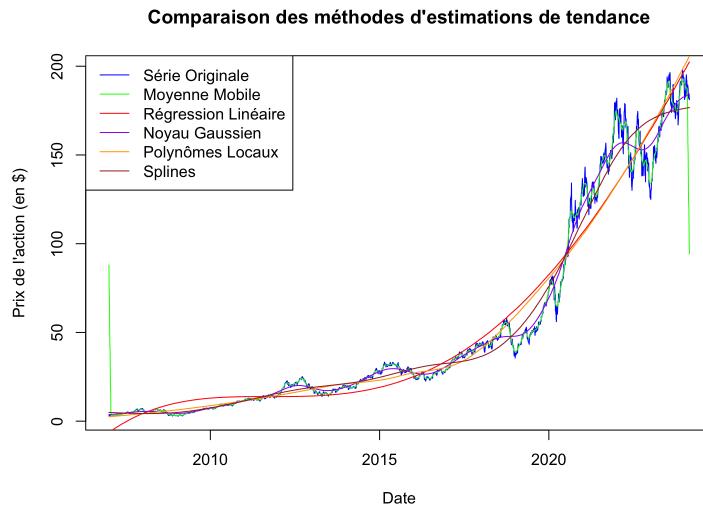


FIGURE 17 – Estimation de la tendance par les différentes méthodes étudiées

Pour déterminer la méthode estimant le mieux la tendance de notre série temporelle, nous avons, pour chaque méthode, créé la série temporelle supprimée de la tendance estimée. Nous avons ensuite calculé les moyennes de ces séries, qui figurent dans le tableau ci-dessous. La série ayant la moyenne la plus proche de 0 nous indique la meilleure méthode d'estimation de la tendance.

Méthode	Moyenne Mobile	Régression Linéaire	Noyau Gaussien	Polynômes Locaux	Splines
Moyenne	-2.1467e-15	-2.1072e-16	3.1543e-02	2.7874e-01	-1.0049e-13

TABLE 1 – Moyennes des séries temporelles après suppression de la tendance estimée pour chaque méthode.

Ainsi, dans le cadre de l'étude de notre série temporelle, la régression linéaire est la méthode donnant la meilleure estimation de la tendance.

### 4.3 Estimation de la saisonnalité

Nous allons effectuer de nouveau les méthodes d'estimation étudiées, afin d'analyser la saisonnalité de notre série temporelle. Nous allons donc dans cette partie analyser et estimer la partie saisonnière, après avoir corrigé la série de la tendance par la méthode de regression linéaire.

#### 4.3.1 Estimation par moyenne mobile

Pour estimer la saisonnalité de notre série temporelle en utilisant une moyenne mobile, nous allons appliquer une moyenne mobile centrée avec une taille de fenêtre correspondant à la période saisonnière. En supposant une saisonnalité annuelle, nous avons pris une taille de fenêtre qui couvre une année de données.

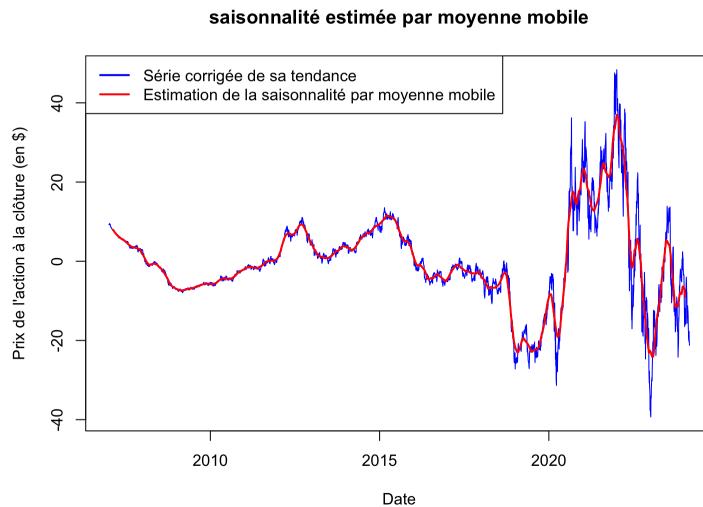


FIGURE 18 – Estimation de la saisonnalité par moyenne mobile sur la série corrigée de sa tendance

La méthode d'estimation de la saisonnalité par moyenne mobile est extrêmement performante pour notre série financière. Nous obtenons une estimation de la saisonnalité qui suit à la trace notre série corrigée de sa tendance.

#### 4.3.2 Estimation par régression de Fourier

La régression de Fourier permet de capturer les motifs saisonniers en décomposant une série temporelle en une somme de fonctions sinusoïdales. On utilise une période de  $t = 360$  qui correspond à un an.

La méthode ne marche malheureusement pas. Nous avons des erreurs lorsque nous compilons notre script R et nous ne sommes pas parvenus à les résoudre.

#### 4.3.3 Estimation par noyau gaussien

Nous avons appliqué l'estimation par noyau gaussien en choisissant une largeur de fenêtre  $h = 100$ .

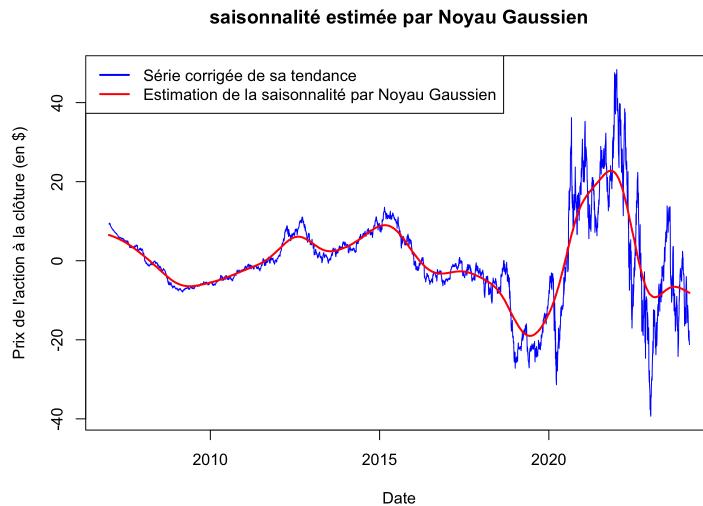


FIGURE 19 – Estimation de la saisonnalité par noyau gaussien sur la série corrigée de sa tendance

#### 4.3.4 Estimation par polynômes locaux

Nous avons appliqué la fonction loess sur la série corrigée de sa tendance avec un paramètre de span de 0,75. Nous avons obtenu le tracé suivant :

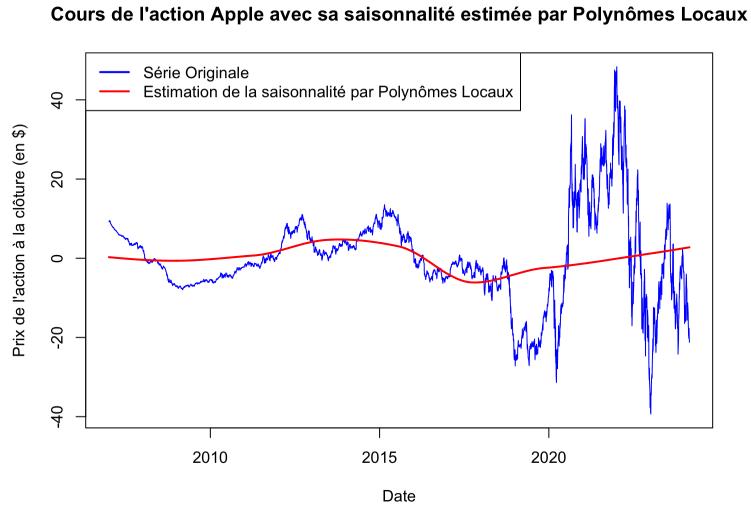


FIGURE 20 – Estimation de la saisonnalité, par polynômes locaux, de la série corrigée de sa tendance

La méthode n'estime pas bien la saisonnalité de notre série car on constate bien que les variations ont du mal à suivre l'amplitude de la série, en particulier à partir de 2020.

#### 4.3.5 Estimation par projection sur base de splines

Nous avons appliqué la fonction gam. Pour l'étude de la saisonnalité, nous avons effectué la projection dans une base de spline cyclique. Nous avons donc indiqué `bs='cc'` dans la fonction gam. Nous avons obtenu le tracé suivant :

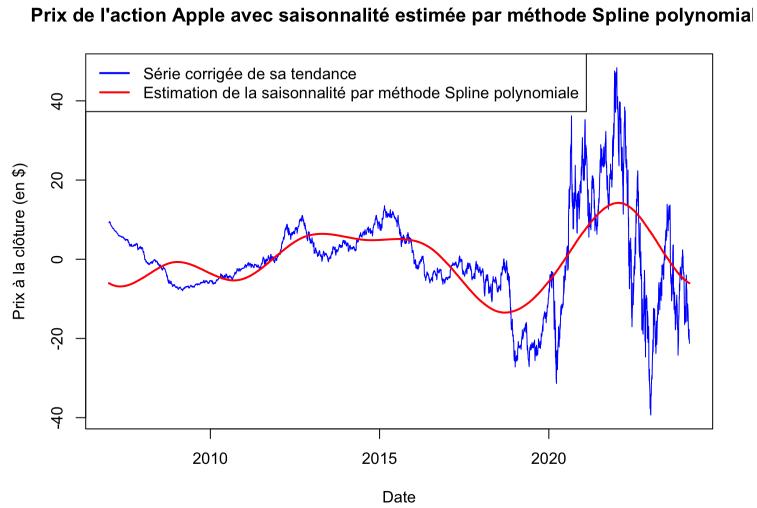


FIGURE 21 – Estimation de la saisonnalité, par projection sur base de splines, de la série corrigée de sa tendance

**Nous avons finalement choisi d'estimer la saisonnalité de notre série temporelle par la méthode de moyenne mobile.** Ci-dessous, un graphique comparatif des différentes méthodes d'estimation.

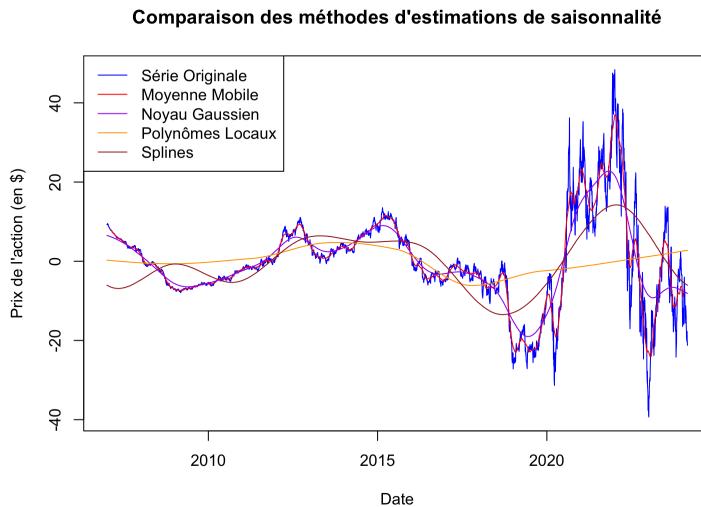


FIGURE 22 – Estimation de la saisonnalité par les différentes méthodes étudiées

#### 4.4 Analyse de la composante résiduelle

Nous allons maintenant étudier la série des résidus :

$$\epsilon_t = Y_t - T_t - S_t$$

La tendance sera estimée par la méthode de régression linéaire tandis que la saisonnalité sera estimée par la méthode de moyenne mobile.

##### 4.4.1 Visualisation des résidus et étude de la stationnarité

Nous avons tracé la courbe des résidus en soustrayant la saisonnalité et la tendance à la série temporelle :

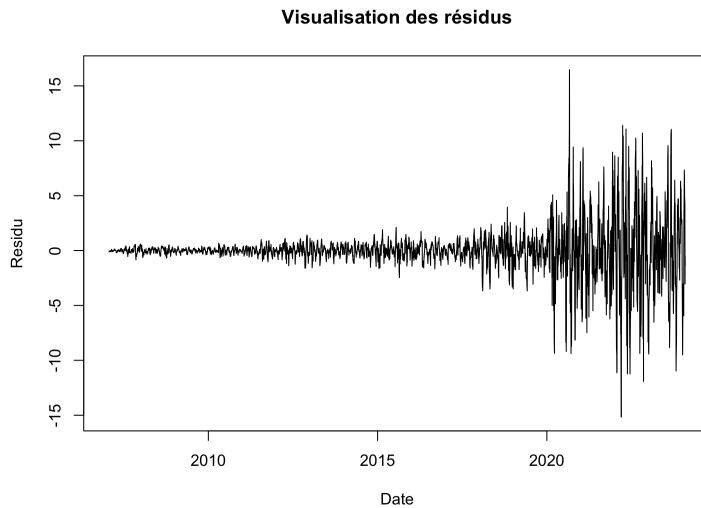


FIGURE 23 – graphique des résidus

Afin d'étudier la stationnarité des résidus, nous avons réalisé le graphe d'autocorrélation des résidus :

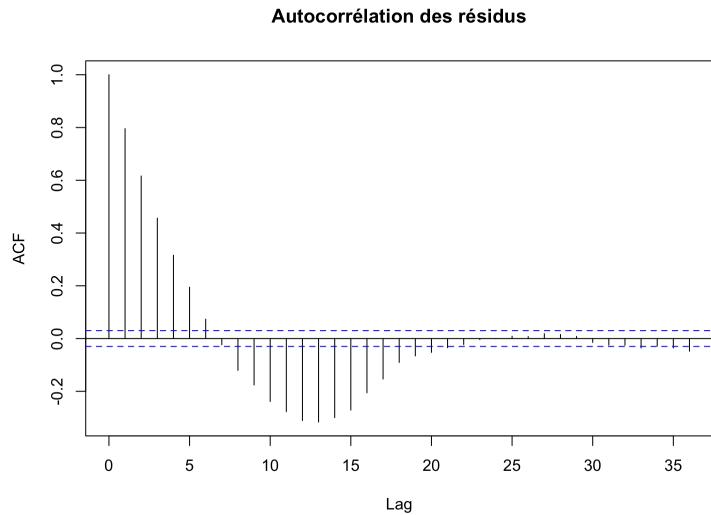


FIGURE 24 – Autocorrélation des résidus

Les résidus semblent être centrés autour de zéro, ce qui indique que la tendance et la saisonnalité ont été suffisamment retirées de la série. Cependant, l'ampleur des fluctuations semble augmenter avec le temps, ce qui pourrait indiquer une volatilité croissante dans les données. Il est également visible que la série présente des pics et des creux, ce qui pourrait suggérer la présence de motifs ou d'anomalies non capturés par la tendance et la saisonnalité.

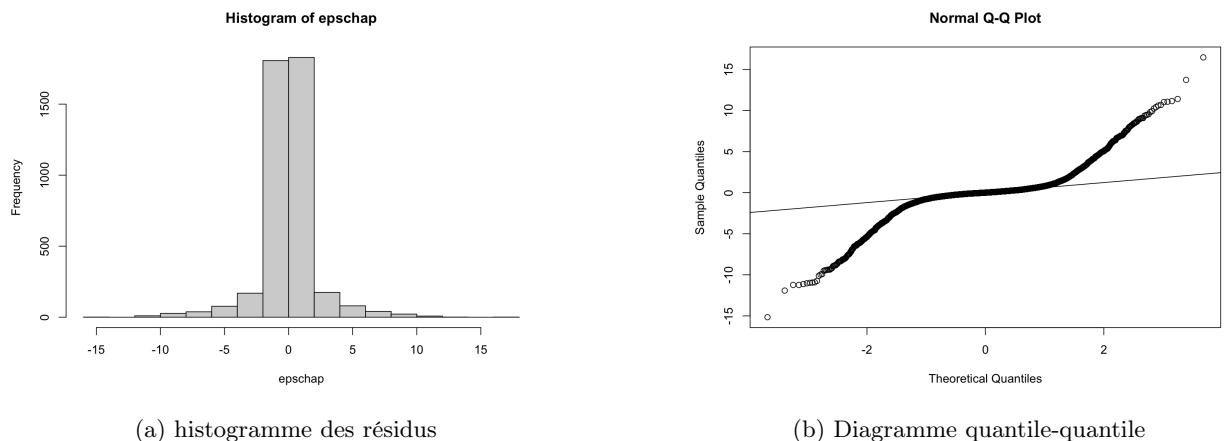
L'autocorrélogramme montre les autocorrelations des résidus à différents lags. Une forte autocorrélation initiale qui décroît avec les lags suggère que la série temporelle a conservé une certaine dépendance temporelle après la suppression de la tendance et de la saisonnalité. Les valeurs d'autocorrélation en dehors des limites de confiance indiquent des corrélations significatives aux lags correspondants.

Nos résidus ressemblent bien à un bruit blanc, à l'exception des 4 dernières années où l'amplitude des fluctuations est plus importante. L'autocorrélation des résidus devient bien non significative à des lags élevés.

Toutefois, il pourrait y avoir des composantes supplémentaires non modélisées ou des irrégularités dans les données qui pourraient nécessiter une investigation plus approfondie. Des modèles supplémentaires, tels que le modèle ARIMA pourraient être envisagés pour modéliser ces dynamiques résiduelles.

#### 4.4.2 Caractère normal des résidus

Afin d'évaluer le caractère normal de la série des résidus, nous avons tracé l'histogramme de cette dernière ainsi que le diagramme quantile-quantile :



Nous pouvons affirmer que nos résidus suivent potentiellement une loi normale centrée.

Toutefois, le graphique Q-Q présente une ligne droite au centre, mais les courbes sont très décalées aux extrémités. Cela signifie certainement que la distribution a une queue plus importante qu'une gaussienne.

## 5 Simulation de prévision sur un échantillon test

Au cours de cette section, nous allons mettre en pratique plusieurs techniques de prévision afin d'anticiper les tendances futures de nos données. Nous allons explorer diverses méthodes de prédiction, les comparer et évaluer leurs performances. Cette approche nous permettra d'obtenir une vue d'ensemble des différentes options de prévision disponibles, ainsi que de choisir la méthode la plus appropriée. Nous allons effectuer les test sur les donnée entre 2007 et 2024 et nous allons essayer de prédire le cours de l'action d'Apple de 2022 à 2024.

Nous allons donc dans cette partie appliquée à notre jeu de données les différentes méthodes de lissage exponentielle vues en cours. Il s'agit de méthodes empiriques de prévision de séries temporelles.

### 5.1 Lissages exponentiels simples

Le lissage exponentiel simple est une méthode de prévision à l'instant suivant, autrement dit l'instant  $t + 1$ .

Mathématiquement, soit une série temporelle  $y_t$ . On appelle lissage exponentiel simple de paramètre  $\alpha \in [0, 1]$  de cette série le processus  $\hat{y}_t$  défini ainsi :

$$\hat{y}_{t+1/t} = a y_t + (1 - a) \hat{y}_{t-1}$$

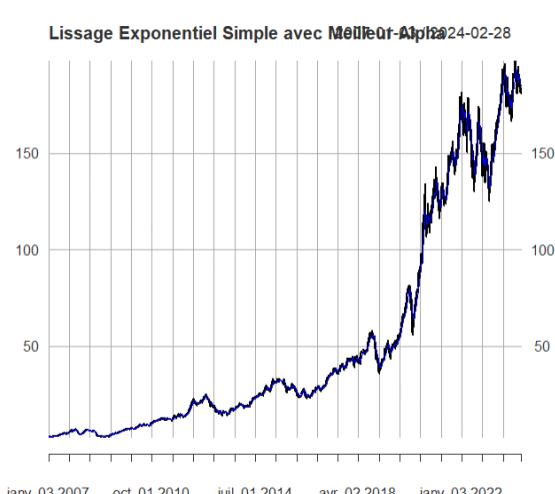
Nous avons donc :

$$\hat{y}_{t+1/t} = \sum_{i=0}^{t-1} \alpha(1 - \alpha)^i y_{t-i}$$

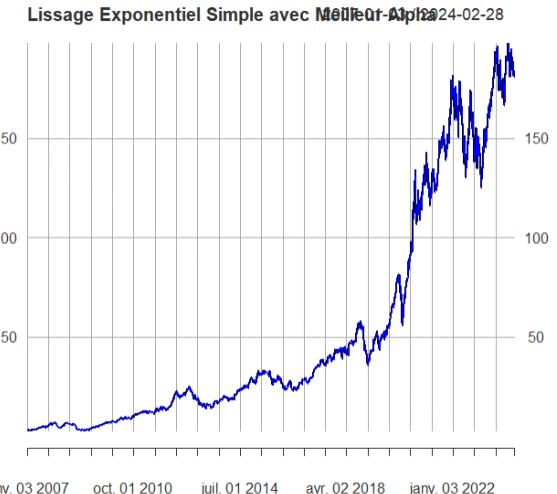
Il s'agit donc d'une prévision basée sur une somme pondérée des valeurs précédentes de la série temporelle. Le coefficient  $\alpha$  joue ici un rôle important : En effet, plus il est proche de 1, plus les observations récentes influent majoritairement sur la prévision. Réciproquement, plus  $\alpha$  est proche de 0, plus la prévision prend en compte les données lointaines.

#### 5.1.1 Application de la méthode

Nous avons tout d'abord appliqué la méthode du lissage exponentiel simple pour un paramètre  $\alpha = 0.9$  et  $\alpha = 0.01$ . Pour 0.01, cela veut dire que notre lissage va prendre en compte les données lointaines dans le passé et conduit à une prévision stable. Nous avons obtenu les tracés suivants :



(a) Lissage avec la méthode exponentielle simple avec  $\alpha = 0.9$



(b) Lissage avec la méthode exponentielle simple avec  $\alpha = 0.01$

Nous observons sur les graphiques ci-dessus que la différence est très faible entre ces lissages simples de paramètre différent. Toutefois, nous remarquons que sur le premier graphique avec  $\alpha = 0.9$ , la courbe est un peu décalée par rapport aux données (visibilité de la couleur noire). En effet, le choix d'un alpha trop élevé peut entraîner des prévisions assez mauvaises. Dans le cas extrême  $\alpha = 1$ , selon la formule :

$$\hat{y}_{t+1/t} = ay_t + (1 - \alpha)\hat{y}_{t/t-1}$$

la prévision est constante égale à la valeur précédemment mesurée, ce qui est très imprécis.

Afin de trouver une meilleure valeur de  $\alpha$ , nous avons cherché la valeur qui minimise l'erreur sur les prévisions à l'horizon 1. Pour cela, nous avons cherché à minimiser la somme des carrés des écarts :

$$\sum_{t=1}^T (y_t - \hat{y}_{t/t-1})^2$$

T désigne la période pendant laquelle on effectue des prévisions. Pour le alpha optimal, nous obtenons :

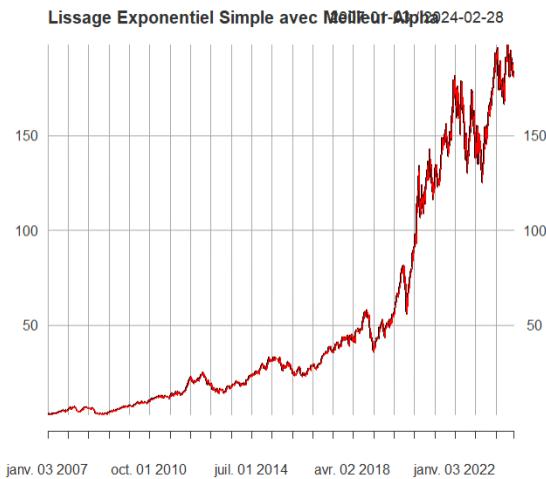


FIGURE 27 – Lissage avec la méthode exponentielle simple avec le alpha optimal

Ce lissage colle davantage à la série. Il prend en compte les irrégularités et semble également bien prendre en compte les données récentes et donc la saisonnalité.

### 5.1.2 Prévision sur les données

Pour réaliser une prévision sur nos données, nous décidons, à partir d'un instant donné, de prédire les valeurs futures en utilisant les valeurs précédentes. En d'autres termes, nous substituons aux valeurs  $y_t$  leurs estimations calculées dans la formule du lissage exponentielle simple.

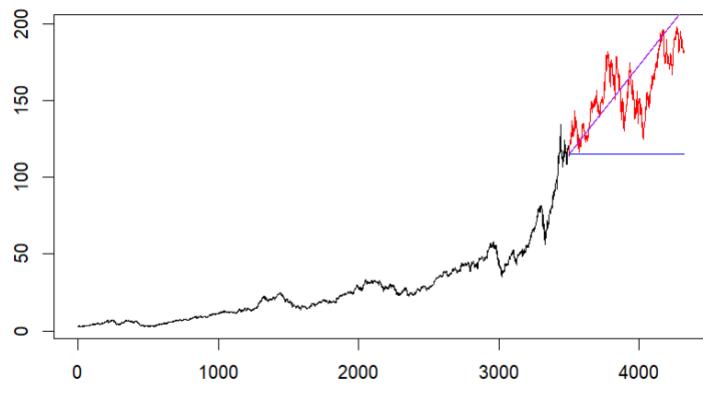


FIGURE 28 – Prévision avec lissage exponentielle simple

Nous obtenons deux résultats de prévision sur la figure :

- Le premier (en bleu : ANN, modèle de lissage additif) est une constante qui correspond à la dernière valeur prévue en se basant sur les données. Nous obtenons donc une prévision constante égale à la dernière valeur mesurée, ce qui n'est pas très intéressant pour faire une bonne prévision.
- Le deuxième (en violet : AAN, modèle adaptatif) est une droite qui semble prédire les valeurs futures par la tendance de la série temporelle. Ce modèle de lissage adaptatif AAN permet d'obtenir une droite au lieu d'une constante.

## 5.2 Lissage exponentielle double (Holt)

L'idée est d'ajuster une droite au lieu d'une constante dans l'approximation locale de la série, comme dans la méthode de lissage exponentiel simple. On appelle ainsi lissage exponentiel double (ou de Holt) de paramètre  $\alpha$  de la série le processus  $\hat{y}_t$  définie ainsi :

$$\hat{y}_{t+1/t} = l_t + b_t h$$

$$\begin{cases} l_t = l_{t-1} + b_{t-1} + (1 - (1 - \alpha)^2)(y_t - \hat{y}_{t/t-1}) \\ b_t = b_{t-1} + \alpha^2(y_t - \hat{y}_{t/t-1}) \end{cases}$$

Cette méthode s'avère efficace quand la série temporelle contient une tendance et permet donc d'avoir de meilleures performances que le lissage simple. Le lissage exponentiel double suit presque la même méthode que le lissage simple, mais avec plus de précision puisque deux listes entrent en jeu lors du calcul du lissage.

### 5.2.1 Application de la méthode

A nouveau, nous avons appliqué le lissage exponentiel double pour une valeur  $\alpha$  optimale. Nous avons obtenu le lissage suivant :



FIGURE 29 – Lissage avec la méthode exponentielle double avec le alpha optimal

Nous remarquons que la courbe de lissage dépasse à certains instants les valeurs des données. Toutefois, le lissage semble suivre assez bien la série de données, notamment en terme d'amplitude.

### 5.3 Lissage exponentielle de Holt-Winters

Nous avons ensuite cherché à utiliser le lissage exponentiel de Holt-Winters. Cette approche est une généralisation du lissage double, qui permet entre autre de proposer les modèles suivants :

- tendance linéaire locale
- tendance linéaire locale + saisonnalité (modèle additif)
- tendance linéaire locale \* saisonnalité (modèle multiplicatif)

Dans ce cas, 2 paramètres de lissage entrent en jeu et on ajuste au voisinage de  $t$  un fonction linéaire  $l_t + hb_t$ ,  $h$  étant l'horizon de prévision.

**définition :** soit une série temporelle  $y_t$ . On appelle lissage exponentiel double de Holt-Winters de paramètres  $\alpha \in [0, 1]$  et  $\beta \in [0, 1]$  de cette série le processus  $\hat{y}_t$  défini ainsi :

$$\hat{y}_{t+h} = l_t + hb_t$$

avec

$$\begin{cases} l_t = \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t = \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1} \end{cases}$$

la encore  $l_t$  est une estimation du niveau de la série,  $b_t$  de sa pente (localement en temps).

#### 5.3.1 Application de la méthode

Nous avons au début écrit une fonction qui fait ce calcul comme vu en cours : cette fonction prend comme argument les trois coefficients  $\alpha$ ,  $\beta$  et  $\delta$ . Trouver les paramètres optimaux pour un bon lissage n'est pas une tâche facile. C'est pour cela que pour tracer le lissage de Holt-Winters, nous avons utilisé la fonction `HoltWinters` qui permet d'ajuster automatiquement les paramètres. Le résultat est le suivant :

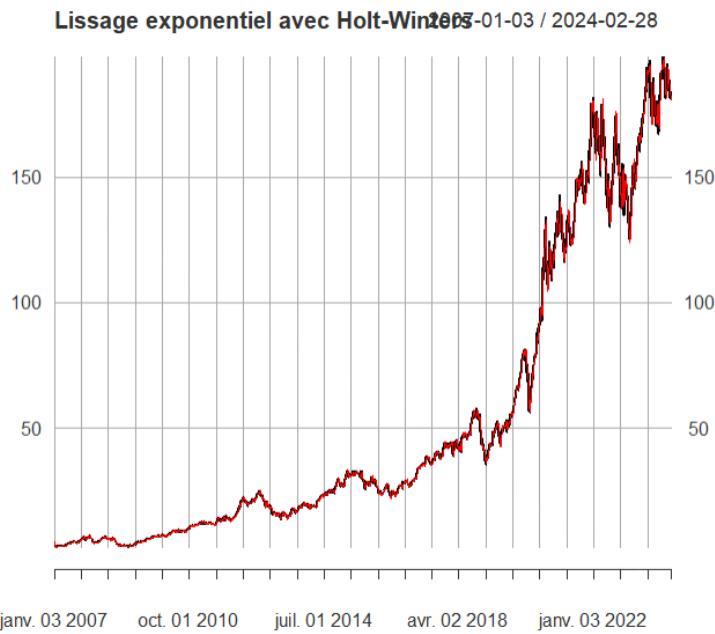


FIGURE 30 – Lissage exponentielle avec la méthode de Holt-Winters

Nous voyons que le lissage est bien adapté à notre jeu de données, même si la courbe est parfois un peu décalée par rapport aux données (visibilité de la couleur noire). Nous pouvons désormais l'utiliser pour effectuer des prédictions.

#### 5.3.2 Prévision sur les données

Lorsque nous effectuons des prévisions avec le lissage exponentiel de Holt-Winters, nous obtenons :

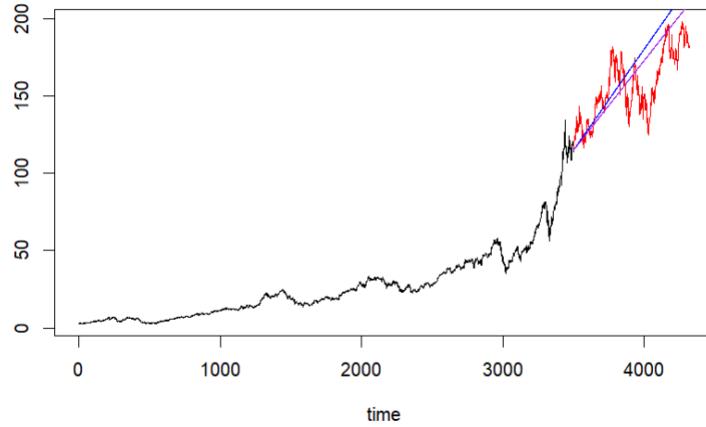


FIGURE 31 – Prévision avec lissage exponentiel Holt-Winters

La méthode en question propose un ajustement linéaire plutôt qu'une constante, ce qui est visible sur la courbe observée (les données prédites par le modèle sont celles correspondant aux droites bleues et violettes). La méthode est alors plus adaptée car elle suit la tendance croissante de notre échantillon sans prendre en compte la saisonnalité qui est très peu présente dans notre cas.

## 6 Méthode d'ARMA

Dans cette partie, nous allons effectuer la prévision de données à l'aide du modèle ARMA (modèle auto-régressif et moyenne mobile). Si on se donne une série temporelle  $Y_t$  stationnaire, le modèle ARMA est un outil puissant pour déterminer et prédire 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 noté ARMA(p,q), où  $p$  est l'ordre de la partie AR et  $q$  celui de la partie MA.

### 6.1 Principe des modèles

#### 6.1.1 Modèle autorégressif

Un modèle autorégressif d'ordre  $p$ , en abrégé AR( $p$ ), s'écrit :

$$X_t = c + \sum_{i=1}^p \phi_i X_{t-i} + \varepsilon_t$$

où  $\phi_1, \dots, \phi_p$  sont les paramètres à estimer du modèle,  $c$  une constante et  $\varepsilon_t$  un bruit blanc. Lorsque la constante est omise, on considère le processus centré.

#### 6.1.2 Modèle moyenne mobile

Le nom MA( $q$ ) (pour Moving Average) fait appel au modèle moyenne-mobile d'ordre  $q$  :

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

où les  $\theta_1, \dots, \theta_q$  sont les paramètres à estimer du modèle et  $\varepsilon_t, \varepsilon_{t-1}, \dots$  sont des bruits blancs (termes d'erreur).

#### 6.1.3 Modèle ARMA

Les processus ARMA( $p,q$ ) généralise les modèles autorégressifs et moyennes mobiles. Ces modèles sont très utiles en pratique pour modéliser des séries réelles en nécessitant moins de paramètres que les modèles AR ou MA simples.

**définition** : un processus stationnaire  $X_t$  admet une représentation ARMA(p,q) minimale s'il satisfait

$$X_t + \sum_{i=1}^p \phi_i X_{t-i} = \varepsilon_t + \sum_{j=1}^q \theta_j \varepsilon_{t-j}, \quad \forall t \in \mathbb{Z}$$

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

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

où les  $\phi_i$  et  $\theta_i$  sont les paramètres du modèle et les  $\varepsilon_i$  les bruits (termes d'erreur).

On peut noter qu'un modèle autorégressif AR(p) est un ARMA(p,0) tandis qu'un modèle moyenne mobile MA(q) est un ARMA(0,q)

#### 6.1.4 Choix du modèle

En pratique, lorsque l'on doit ajuster un modèle AR, MA ou ARMA à des données réelles la première question qui se pose est celle du choix des ordres p et q du modèle ARMA (on considère que les AR et MA sont un cas particulier d'ARMA avec respectivement  $q = 0$  et  $p = 0$ ). Pour choisir ces ordres, nous pouvons exploiter les résultats suivants :

Type de processus	Définition	autocorrélations	autocorrélations partielles
AR(p)	$\Phi(L)X_t = \varepsilon_t$	$\rho(h) \searrow 0$	$r(h) = 0$ pour $h \geq p+1$
MA(q)	$X_t = \Theta(L)\varepsilon_t$	$\rho(h) = 0$ pour $h \geq q+1$	$r(h)$ rien de particulier
ARMA(p,q)	$\Phi(L)X_t = \Theta(L)\varepsilon_t$	$\rho(h) \searrow 0, h \geq q+1$	$r(h) \searrow 0, h \geq \max(q+1, p+1)$

## 6.2 Application à notre jeu de données

### 6.2.1 Implémentation du modèle

Nous allons donc utiliser la méthode de prévision ARMA décrite précédemment. Pour faire cela, il faut d'abord déterminer les ordres (p,q) pour appliquer la méthode. La détermination de ces paramètres se fait à l'aide de l'analyse des autocorrélogrammes de la série des résidus.

Nos modèles ARMA avec nos estimations de paramètres potentiels n'ont pas donné des prévisions pertinentes pour notre jeu de données. Nous avons donc utilisé la fonction **auto.arima** de la bibliothèque forecast qui prend en argument un objet xts et qui permet d'estimer en plus l'ordre d (ordre de différenciation où la série devient stationnaire). Nous appliquons d'abord cette méthode sur la série temporelle du cours de l'action d'Apple. Voici les paramètres trouvés pour l'ARMA automatique :

```
> AAPL.model = auto.arima(AAPL_xts)
> AAPL.model
Series: AAPL_xts
ARIMA(2,1,2) with drift

Coefficients:
ar1      ar2      ma1      ma2      drift
-0.3578  0.4830  0.3420 -0.5337  0.0417
s.e.   0.2354  0.2241  0.2242  0.2151  0.0189

sigma^2 = 1.811: log likelihood = -7406.95
AIC=14825.89  AICc=14825.91  BIC=14864.12
```

FIGURE 32 – Détermination des paramètres choisis par la fonction auto.arima pour le modèle ARMA

### 6.2.2 prévision avec le modèle ARMA

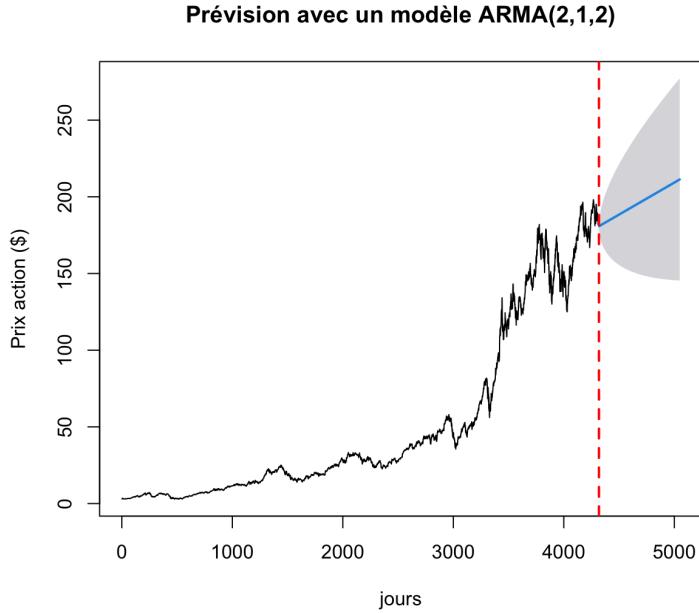


FIGURE 33 – Prévision sur 2 ans du cours de l'action d'Apple à l'aide du modèle ARMA

Cette méthode est très pertinente, comparativement à la prévision avec lissage de Holt-Winters. Elle fournit une prévision qui correspond bien à la tendance de nos données. Par conséquent, nous retenons cette méthode pour prédire l'évolution du cours de l'action d'Apple dans un avenir proche.

## 7 Conclusion

Pour conclure, ce projet s'est révélé extrêmement bénéfique pour approfondir notre compréhension des différentes techniques abordées pendant les cours et les travaux dirigés. Nous avons découvert lors de cette étude la méthodologie mise en oeuvre par un statisticien pour traiter un jeu de données inconnu. De plus, nous avons procédé à une analyse descriptive préliminaire avant de nous lancer dans la modélisation en vue de créer un modèle prévisionnel capable de simuler des prédictions futures.

L'analyse de la série chronologique du cours de l'action Apple offre des informations précieuses pour les investisseurs, les analystes financiers et les décideurs. Elle permet de mieux comprendre les tendances passées, d'évaluer les risques et de formuler des prévisions pour guider les décisions financières futures. Cependant, il est important de prendre en compte les limitations et les incertitudes inhérentes à toute analyse de données et à toute modélisation en vu d'une prévision.

Pour terminer, ce projet a renforcé notre maîtrise du langage R, tout en nous offrant l'occasion d'acquérir de nouvelles compétences pratiques pour rendre la programmation dans ce langage plus aisée.