Université des sciences et de la Technologie Houari Boumediene

Faculté de Mathématiques

Département de Probabilités et Statistique

La méthodologie de Box et jenkins et application sur Microsoft corporation

FERGUOUS Wafa
Master2 SPA
17173203558

0.1 Introduction

La prévision est un terme que nous associons généralement à la météo. Pendant que nous écoutons ou regardons les NEWS, il y a toujours un segment séparé appelé «Weather Report» où le commentateur des NEWS nous fournit les informations sur les prévisions météorologiques. Pourquoi la prévision est-elle si importante ? Eh bien, tout simplement parce que nous pouvons prendre des décisions éclairées. Lorsque nous associons un composant temporel à la prévision, il devient Time Series Forecasting et les données sont appelées Time Series Data.

Chapitre 1

Principe de séries temporelles

1.1 Définition

Les séries chronologiques font référence à une série de données enregistrées en fonction des observations dans l'ordre chronologique. Il agit comme une entrée dans divers modèles d'analyse et de prévision pour en tirer des informations utiles.telle que :

Il peut être catégorisé en différents types l'une est la catégorisation en séries chronologiques non stationnaires et stationnaires. S'il est stationnaire, il possède des propriétés stochastiques telles que la variance qui ne varie pas avec le temps. Alors que pour le non-stationnaire, ses propriétés varient avec le temps, et il peut s'agir d'une tendance, d'occurrences aléatoires, de saisons, de cycles, etc. Il est facile et efficace de modéliser lorsqu'il est stationnaire en appliquant des méthodes de modélisation statistique.

1.2 La décomposition

- Tendance : Augmentation ou diminution à long terme des données. La tendance peut être n'importe quelle fonction, telle que linéaire ou exponentielle, et peut changer de direction au fil du temps.
- Saisonnalité : Cycle répétitif dans la série avec des fréquences fixes (heure de la journée, semaine, mois, année, etc.). Il existe un modèle saisonnier d'une période fixe connue.
- Cyclicité : se produit lorsque les données montent et descendent, mais sans fréquence et durée fixes causées, par exemple, par les conditions économiques.

— Bruit : la variation aléatoire dans la série.

1.3 La Stationnarité

La stationnarité est une caractéristique essentielle des séries chronologiques. Une série temporelle est dite stationnaire si ses propriétés statistiques ne changent pas dans le temps. En d'autres termes, sa moyenne et sa variance sont constantes et la covariance est indépendante du temps
Idéalement, nous voulons avoir une série temporelle stationnaire pour la modélisation. Bien sûr, tous ne sont pas stationnaires, mais nous pouvons faire différentes transformations pour les rendre stationnaires.
Un moyen plus populaire et plus précis de vérifier la stationnarité consiste à effectuer des tests statistiques. Il existe différents tests disponibles à cet effet.ici quelques-uns des plus populaires.

— Test de Dickey-Fuller (DF)
— Test de Dickey-Fuller augmenté (ADF)
— Test de Philips-Perron (PP)

Chapitre 2

Modélisation des séries chronologiques

Il existe de nombreuses façons de modéliser une série chronologique afin de faire des prédictions.

2.1 modèles linéaire

2.2 modèles ARIMA

Les modèles ARIMA offrent une autre approche de la prévision des séries chronologiques. Le lissage exponentiel et les modèles ARIMA sont les deux approches les plus largement utilisées pour la prévision des séries chronologiques et fournissent des approches complémentaires au problème. Alors que les modèles de lissage exponentiel sont basés sur une description de la tendance et de la saisonnalité des données, les modèles ARIMA visent à décrire les autocorrélations dans les données.
Dans cette section, nous considérons les modèles linéaires proposés par Box et Jenkins et qui sont les plus utilisés pour modéliser une série temporelle

2.2.1 modèle AR

Dans un modèle de régression multiple, nous prévoyons la variable d'intérêt en utilisant une combinaison linéaire de prédicteurs. Dans un modèle d'autorégression, nous prévoyons la variable d'intérêt en utilisant une combinaison linéaire des valeurs passées de la variable.

Le terme autorégression indique qu'il s'agit d'une régression de la variable sur elle-même.
Ainsi, un modèle autorégressif d'ordre p peut s'écrire
$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \varepsilon_t,$
où ϵ_t est le bruit blanc.C'est comme une régression multiple mais avec des valeurs décalées de yt comme prédicteurs. On parle alors de modèle AR(p), un modèle autorégressif d'ordre p

Pour un modèle AR(1) :
- lorsque ϕ1=0 , yt équivaut à un bruit blanc
- lorsque ϕ_1=0 et c=0, y_t équivaut à une marche aléatoire
- lorsque ϕ_1=0 et c≠0 , y_t équivaut à une marche aléatoire avec dérive

Nous restreignons normalement les modèles autorégressifs aux données stationnaires, auquel cas certaines contraintes sur les valeurs des paramètres sont nécessaires.
- Pour un modèle AR(1) : -1 < ϕ_1 < 1
- Pour un modèle AR(2) : $-1 < \phi_2 < 1, \phi_1 + \phi_2 < 1 \phi_2 - \phi_1 < 1$

Lorsque p≥3, les restrictions sont beaucoup plus compliquées. R prend en compte ces restrictions lors de l'estimation d'un modèle.

2.2.2 modèle MA

Au lieu d'utiliser les valeurs passées de la variable de prévision dans une régression, un modèle de moyenne mobile utilise les erreurs de prévision passées dans un modèle de type régression.

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} \tag{2.1}$$

où ϵ_t est le bruit blanc.Nous appelons cela un modèle MA(q), un modèle à moyenne mobile d'ordre q . Bien sûr, on n'observe pas les valeurs de ϵ_t , donc ce n'est pas vraiment une régression au sens habituel.

2.2.3 modèle ARIMA

Si nous combinons la différenciation avec l'autorégression et un modèle de moyenne mobile, nous obtenons un modèle ARIMA non saisonnier. ARIMA est l'acronyme de AutoRegressive Integrated

Moving Average (dans ce contexte, « l'intégration » est l'inverse de la différenciation). Le modèle complet peut être écrit comme

$$y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \qquad (2.2)$$

où y'_t est la série différenciée (elle peut avoir été différenciée plus d'une fois). Les « prédicteurs » sur le côté droit incluent à la fois les valeurs décalées de $y_t$et les erreurs décalées. Nous appelons cela un ARIMA(p,d,q) modèle , où

- p=ordre de la partie autorégressive ;
- d=degré de différenciation première impliqué ;
- q=ordre de la partie moyenne mobile.

Les mêmes conditions de stationnarité et d'inversibilité qui sont utilisées pour les modèles autorégressifs et à moyenne mobile s'appliquent également à un modèle ARIMA.
Une fois que nous commençons à combiner des composants de cette manière pour former des modèles plus compliqués, il est beaucoup plus facile de travailler avec la notation backshift. Par exemple, l'équation (2.2) peut être écrite en notation rétrograde comme

$$\underset{\substack{\uparrow \\ \text{AR}(p)}}{(1 - \phi_1 B - \cdots - \phi_p B^p)} \quad \underset{\substack{\uparrow \\ d \text{ differences}}}{(1 - B)^d y_t} \quad = \quad c + \underset{\substack{\uparrow \\ \text{MA}(q)}}{(1 + \theta_1 B + \cdots + \theta_q B^q)} \varepsilon_t \qquad (2.3)$$

Comprendre les modèles ARIMA

La constante c a un effet important sur les prévisions à long terme obtenues à partir de ces modèles.

- Si c=0 et d=0, les prévisions à long terme tomberont à zéro.
- Si c=0 et d=1, les prévisions à long terme iront vers une constante non nulle.
- Si c=0 et d=2, les prévisions à long terme suivront une ligne droite.
- Si c≠0 et d=0, les prévisions à long terme iront à la moyenne des données.
- Si c≠0 et d=1, les prévisions à long terme suivront une ligne droite.
- Si c≠0et d=2, les prévisions à long terme suivront une tendance quadratique.

La valeur de d a également un effet sur les intervalles de prédiction — plus la valeur de d, plus la taille des intervalles de prédiction

augmente rapidement. Pour d=0, l'écart type des prévisions à long terme ira à l'écart type des données historiques, de sorte que les intervalles de prédiction seront tous essentiellement les mêmes.

La valeur de p est important si les données montrent des cycles. Pour obtenir des prévisions cycliques, il faut disposer $p \geq 2$, ainsi que des conditions supplémentaires sur les paramètres. Pour un modèle AR(2), un comportement cyclique se produit si$\phi_1^2 + 4\phi_2 < 0$. Dans ce cas, la période moyenne des cycles est de

$$\frac{2\pi}{\mathbf{arc\ cos}(-\phi_1(1-\phi_2)/(4\phi_2))}.$$

Une fois l'ordre du modèle identifié (c'est-à-dire les valeurs de p,d et q), il faut estimer les paramètres c, $\phi_1, \ldots, \phi_p, \theta_1, \ldots, \theta_q$. Lorsque R estime le modèle ARIMA, il utilise l'estimation du maximum de vraisemblance (MLE). Cette technique trouve les valeurs des paramètres qui maximisent la probabilité d'obtenir les données que nous avons observées. Pour les modèles ARIMA, MLE est similaire aux estimations des moindres carrés qui seraient obtenues en minimisant

$$\sum_{t=1}^{T} \varepsilon_t^2.$$

Critères d'information

Le critère d'information d'Akaike (AIC), qui a été utile pour sélectionner des prédicteurs pour la régression, est également utile pour déterminer l'ordre d'un modèle ARIMA. Il peut être écrit comme

$$\mathbf{AIC} = -2\log(L) + 2(p+q+k+1), \tag{2.4}$$

où L est la vraisemblance des données, k=1 si $c \neq 0$ et k=0 si c=0 .Notez que le dernier terme entre parenthèses est le nombre de paramètres dans le modèle (σ^2 , la variance des résidus).
Pour les modèles ARIMA, l'AIC corrigé peut être écrit comme

$$\mathbf{AICc} = \mathbf{AIC} + \frac{2(p+q+k+1)(p+q+k+2)}{T-p-q-k-2}, \tag{2.5}$$

et le critère d'information bayésien peut être écrit comme

$$\textbf{BIC} = \textbf{AIC} + [\log(T) - 2](p + q + k + 1). \tag{2.6}$$

De bons modèles sont obtenus en minimisant l'AIC, l'AICc ou le BIC. Notre préférence est d'utiliser l'AICc.

2.2.4 modèle SARIMA

Jusqu'à présent, nous avons limité notre attention aux données non saisonnières et aux modèles ARIMA non saisonniers. Cependant, les modèles ARIMA sont également capables de modéliser un large éventail de données saisonnières.

Un modèle ARIMA saisonnier est formé en incluant des termes saisonniers supplémentaires dans les modèles ARIMA que nous avons vus jusqu'à présent. Il s'écrit comme suit :

$$\textbf{ARIMA } (p, d, q)(P, D, Q)_m$$

- (p, d, q) :Partie non saisonnière du modèle
- (P, D, Q) :Partie saisonnière du modèle
La partie saisonnière du modèle se compose de termes qui sont similaires aux composantes non saisonnières du modèle, mais impliquent des décalages de la période saisonnière

Chapitre 3

méthodologie de box et jenkins

La méthode Box-Jenkins a été proposée par George Box et Gwilym Jenkins dans leur manuel de 1970.
L'approche part de l'hypothèse que le processus qui a généré la série chronologique peut être approximé à l'aide d'un modèle ARMA s'il est stationnaire ou d'un modèle ARIMA s'il n'est pas stationnaire.
c'est une approche itérative qui se compose des 3 étapes suivantes :

- Identification
- Estimation
- Validation

La première étape du développement d'un modèle de Box-Jenkins consiste à déterminer si la série est stationnaire et s'il existe une saisonnalité significative à modéliser.

3.1 Identification

À l'étape d'identification du modèle, notre objectif est de détecter la saisonnalité, si elle existe, et d'identifier l'ordre des termes saisonniers autorégressifs et saisonniers de moyenne mobile. Pour de nombreuses séries, la période est connue et un seul terme de saisonnalité suffit. Par exemple, pour les données mensuelles, nous incluons généralement soit un terme saisonnier AR 12, soit un terme saisonnier MA 12. Pour les modèles Box-Jenkins, nous ne supprimons pas explicitement la saisonnalité avant d'ajuster le modèle. Au lieu de cela, nous incluons l'ordre des termes saisonniers dans la spécification du modèle au logiciel d'estimation ARIMA. Cepen-

dant, il peut être utile d'appliquer une différence saisonnière aux données et de régénérer les tracés d'autocorrélation et d'autocorrélation partielle. Cela peut aider à l'identification du modèle de la composante non saisonnière du modèle. Dans certains cas, la différenciation saisonnière peut supprimer la plupart ou la totalité de l'effet de saisonnalité.

Une fois que la stationnarité et la saisonnalité ont été abordées, l'étape suivante consiste à identifier l'ordre (c'est-à-dire le p et le q) des termes de autorégressive et moyenne mobile.
Les principaux outils pour ce faire sont le diagramme d'autocorrélation et le diagramme d'autocorrélation partielle.

3.2 Estimation

L'estimation des paramètres d'un modèle ARIMA(p,d,q)se réalise par différentes mÈthodes dans le domaine temporel, et parmi ces méthodes on a :

- Maximum de vraissemblance
- Dans le cas q = 0, on utilise les équations de Yule Walker.

Critères de choix des modèles

Souvent, il n'est pas facile de déterminer un modèle unique. Le modèle finalement retenu est celui qui minimise un des critères à partir de T observations.

Critère standard

- L'erreur absolue moyenne
- L'erreur quadratique moyenne
- La racine carrée de l'erreur quadratique moyenne
- Ecart absolu moyen en pourcentage

3.3 Validation

Après avoir estimé les différents modèles ARIMA, il faut maintenant valider ces modèles, en servant d'une part, des tests de signification des paramètres

pour les coefficients et d'autre part, une analyse des résidus estimés.

Chapitre 4

Application de la méthode de Box-Jenkins

Les données utilisées sont les prix quotidiens historiques du MSFT (Microsoft Inc). Le cours de clôture ajusté a été choisi pour être modélisé et prédit. En effet, le cours de clôture ajusté reflète non seulement le cours de clôture comme point de départ, mais il prend en compte des facteurs tels que les dividendes, les fractionnements d'actions et les nouvelles offres d'actions pour déterminer une valeur. Cette étude a utilisé les données de stock Microsoft utilisées qui couvraient la période du 2017-11-03 au 2022-11-01, avec un nombre total de 1257 observations.

Commençons par load les librairies suivantes dans l'environnement R :

```
acf(diftrain)
pacf(diftrain)
```

Ensuite, nous suivrons un processus des étapes comme indiqué ci-dessous :

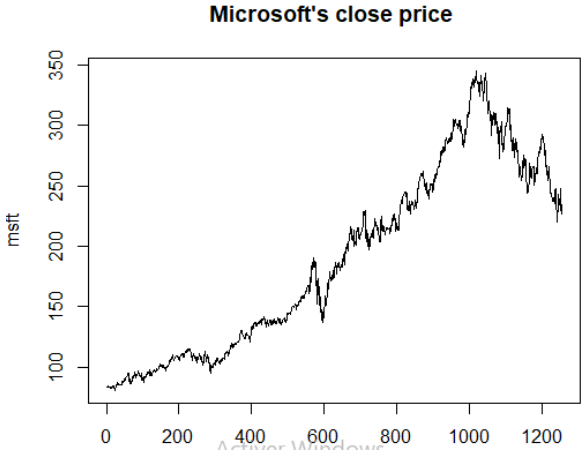
Attribution de la série temporelle

```
data=read.csv("MSFT.csv")
head(data)
```

```
> data=read.csv("MSFT.csv")
> head(data)
        Date  Open  High   Low Close Adj.Close   Volume
1 2017-11-03 84.08 84.54 83.40 84.14  79.06213 17633500
2 2017-11-06 84.20 84.70 84.08 84.47  79.37221 19860900
3 2017-11-07 84.77 84.90 83.93 84.27  79.18429 17939700
4 2017-11-08 84.14 84.61 83.83 84.56  79.45679 18034200
5 2017-11-09 84.11 84.27 82.90 84.09  79.01514 21178400
6 2017-11-10 83.79 84.10 83.23 83.87  78.80843 19397800
```

Représentation de prix de cloture

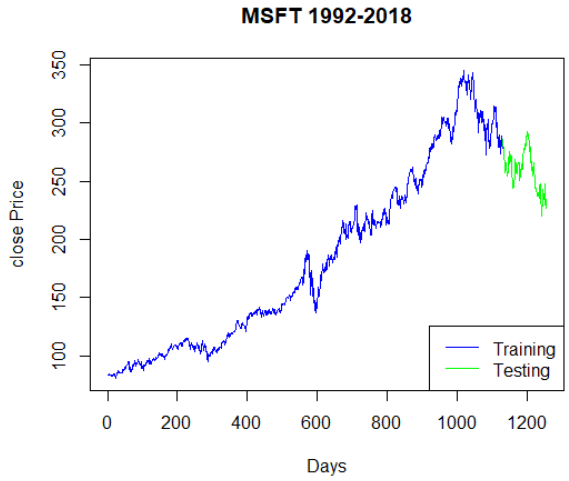
```
msft=ts(data[,2],frequency=1)
plot.ts(msft, main = "Microsoft's close price ")
```



diviser les données en un ensemble d'entraînement et de test

```
msft.train = window(msft,end=1132)
msft.test = window(msft,start=1132)
```

```
plot(msft,main="MSFT 1992-2018",ylab="close Price"
,xlab="Days")
lines(msft.train,col="blue")
lines(msft.test,col="green")
legend("bottomright",col=c("blue","green"),lty=1,
legend=c("Training","Testing"))
```



Nous effectuerons le test ADF pour vérifier la stationnarité dans la série temporelle ci-dessus

```
adf.test(msft.train,alternative="stationary")
```

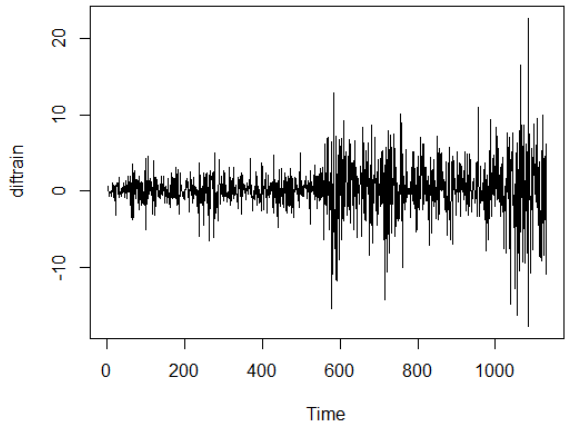
```
> adf.test(msft.train,alternative="stationary")

        Augmented Dickey-Fuller Test

data:  msft.train
Dickey-Fuller = -2.5823, Lag order = 10, p-value = 0.3318
alternative hypothesis: stationary
```

La p-value du test ADF ci-dessus montre que la série chronologique de MSFT est non stationnaire

Étant donné que la série chronologique ci-dessus n'est pas stationnaire, nous procédons à la différenciation du premier ordre .

```
 diftrain=diff(msft.train)
plot(diftrain)
adf.test(diftrain,alternative="stationary")
```

l'application du test ADF sur la série temporelle différenciée montre ci-dessous que la série temporelle est stationnaire .



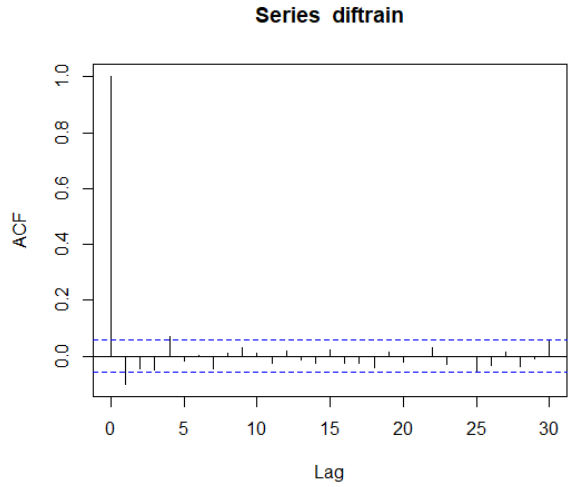

```
> adf.test(diftrain,alternative="stationary")

        Augmented Dickey-Fuller Test

data:  diftrain
Dickey-Fuller = -10.271, Lag order = 10, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test(diftrain, alternative = "stationary") :
  p-value smaller than printed p-value
```

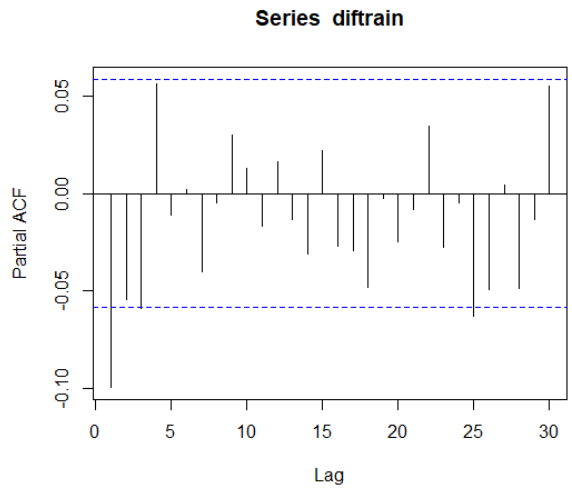
Trouver les paramètres optimaux

Les paramètres p et q peuvent être trouvés à l'aide des tracés ACF et PACF.

```
acf(diftrain)
pacf(diftrain)
```

Les graphiques ci-dessous montrent que nous avons les modèles ARIMA(0,1,4)et ARIMA(0,1,1).
Avec les paramètres en main, nous pouvons maintenant essayer de construire le modèle ARIMA. La valeur trouvée dans la section précédente peut être une estimation approximative

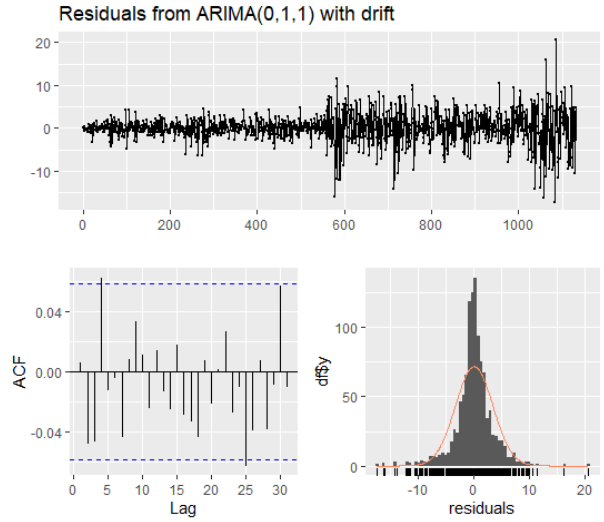


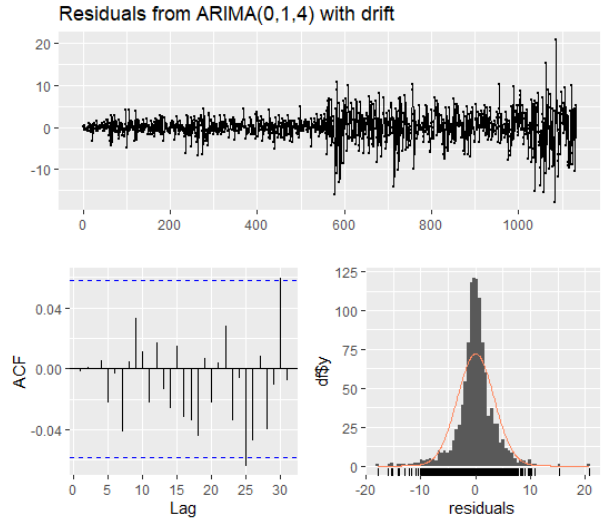



```
 #1st model
model1 =Arima(msft.train,order=c(0,1,1),
include.constant=T)
summary(model1)
#2nd model
model2 = Arima(msft.train,order=c(0,1,4)
,include.constant=T)
summary(model1)
checkresiduals(model1)
checkresiduals(model2)
```

```
> model1 <- Arima(msft.train,order=c(0,1,1),include.constant=T)
> summary(model1)
Series: msft.train
ARIMA(0,1,1) with drift

Coefficients:
          ma1   drift
      -0.1114  0.1755
s.e.   0.0313  0.0887

sigma^2 = 11.3:  log likelihood = -2975.02
AIC=5956.03   AICc=5956.06   BIC=5971.13

Training set error measures:
                     ME     RMSE      MAE       MPE     MAPE      MASE         ACF1
Training set 0.0001509546 3.356991 2.239557 -0.02583217 1.226102 0.9896653 0.005775532
```

```
> model2 = Arima(msft.train,order=c(0,1,4),include.constant=T)
> summary(model2)
Series: msft.train
ARIMA(0,1,4) with drift

Coefficients:
          ma1      ma2      ma3     ma4   drift
      -0.1019  -0.0471  -0.0410  0.0613  0.1756
s.e.   0.0297   0.0298   0.0299  0.0294  0.0867

sigma^2 = 11.24:  log likelihood = -2970.67
AIC=5953.33   AICc=5953.41   BIC=5983.52

Training set error measures:
                     ME     RMSE      MAE        MPE   MAPE      MASE         ACF1
Training set 0.00014390 3.344071 2.23514 -0.02618674 1.2276 0.9877134 -0.001729062
```





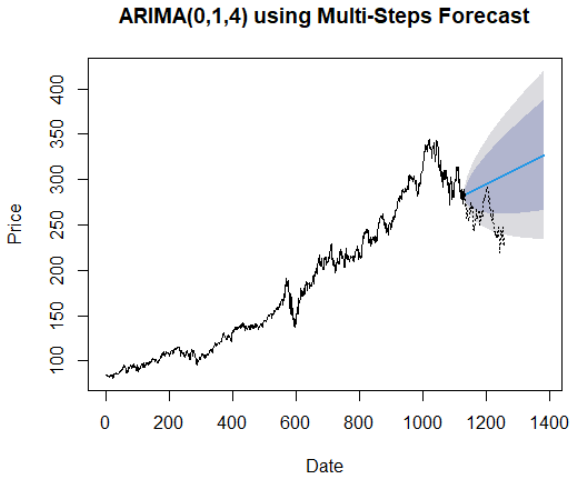

```
> checkresiduals(model1)

        Ljung-Box test

data:  Residuals from ARIMA(0,1,1) with drift
Q* = 13.456, df = 9, p-value = 0.143

Model df: 1.   Total lags used: 10
```

```
> checkresiduals(model2)

        Ljung-Box test

data:  Residuals from ARIMA(0,1,4) with drift
Q* = 4.0192, df = 6, p-value = 0.6741

Model df: 4.   Total lags used: 10
```

Notons qu'il n'y a pas de pics significatives dans les deux autocorrélogrammes, c'est-à-dire : il n'y a pas d'autocorrélation des erreurs, ce qui montre que les deux modèles sont significatifs, donc l'hypothèse du bruit blanc résiduel est validée celle aussi confirmée par le test de Ljung-Box. ARIMA (0,1,4) est celui avec le BIC et l'AIC les plus bas, ce serait notre choix.

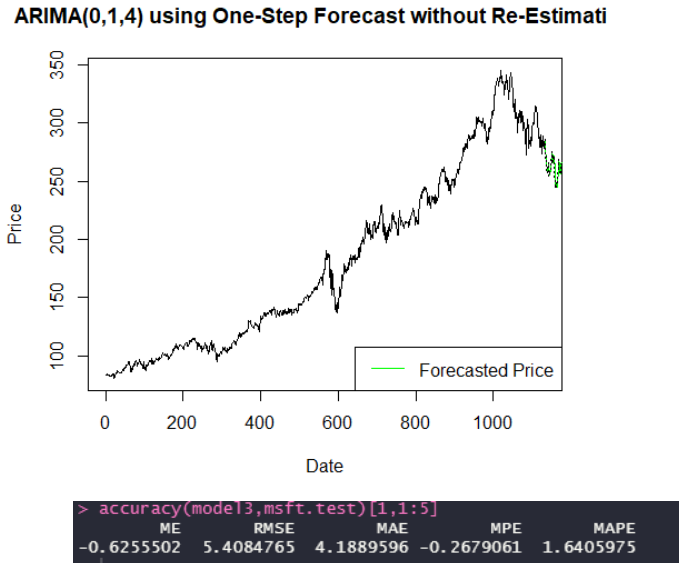
La prévision

```
plot(forecast(model2,h=126),main="ARIMA(0,1,4)
using Multi-Steps Forecast",ylab="Price",xlab="Date")
lines(msft.test,lty=3)
accuracy(forecast(model2,h=252),msft.test)[2,1:6]
```




```
> accuracy(forecast(model2,h=252),msft.test)[2,1:6]
        ME      RMSE       MAE       MPE      MAPE      MASE
-35.37610  40.85196  35.41065 -14.24491  14.25701  15.64804
```

```
model3 = Arima(msft.test,model=model1)$fitted
plot(msft.train,main="ARIMA(0,1,4) using One-Step
Forecast without Re-Estimation",ylab="Price"
,xlab="Date",ylim=c(min(msft),max(msft)))
lines(model3,col="green")
lines(msft.test,lty=3)
legend("bottomright",col="green",lty=1,
legend="Forecasted Price")
accuracy(model3,msft.test)[1,1:5]
```




```
> accuracy(model3,msft.test)[1,1:5]
        ME       RMSE        MAE        MPE       MAPE
-0.6255502  5.4084765  4.1889596 -0.2679061  1.6405975
```

au moins la prévision en une étape sans réestimation fait un travail en donnant un RMSE de 5,4084765. mieux que la prévision multi-étapes qui donne un RMSE de 40,85196

Table des matières