

# Tuto Package Quantmod FR

Antoine SERREAU - MSc Data Management

20/12/2020

Le but de ce tutoriel est de vous initier à l'analyse financière sur le cours des actions. Le package Quantmod permet à l'utilisateur de se plonger dans l'environnement du trader. Vous pourrez explorer des modèles d'analyses sur le cours des actions en bourse.

Quantmod a 3 principales fonctions pour vous : - Charger des données de différentes sources - Cartographier la donnée - Mettre en place des indicateurs clés pour comprendre le cours d'une action

Avant de commencer, utilisons le code suivant pour installer et charger quantmod.

```
#install.packages("quantmod")  
library(quantmod)  
library(xts)  
library(TTR)  
library(knitr)
```

## Introduction au Package Quantmod

Afin de charger des données, nous allons utiliser la fonction `getSymbols`. Cette fonction va aller chercher les données d'une ou plusieurs entreprises. Nous allons ici utiliser des données US. Par défaut, le package importe des données boursières US issues du NASDAQ.. Vous pouvez trouver l'ensemble des Indices de marché (Stock Market Index) sur le lien suivant : <http://dev.eoddata.com/stocklist/NASDAQ/A.htm>

Quantmod permet de charger des données à partir d'une grande variété de sources comprenant par exemple : - Yahoo! Finance - Federal Reserve Bank of St. Louis FRED - Google Finance - Bases de données MySQL - R formats binaires - Fichier format csv

Tout d'abord nous allons charger les données d'une entreprise américaine que vous connaissez tous : APPLE. Stock Market Index : AAPL

```
getSymbols("AAPL") #charge les données de APPLE dans votre environnement
```

```
## [1] "AAPL"
```

```
head(AAPL) #permet de voir les données
```

```
##           AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted  
## 2007-01-03  3.081786  3.092143  2.925000  2.992857  1238319600  2.581791  
## 2007-01-04  3.001786  3.069643  2.993572  3.059286   847260400  2.639096  
## 2007-01-05  3.063214  3.078571  3.014286  3.037500   834741600  2.620302  
## 2007-01-08  3.070000  3.090357  3.045714  3.052500   797106800  2.633242  
## 2007-01-09  3.087500  3.320714  3.041071  3.306072  3349298400  2.851985  
## 2007-01-10  3.383929  3.492857  3.337500  3.464286  2952880000  2.988469
```

Si vous souhaitez importer des données d'une certaine période, vous pouvez restreindre l'intervalle de temps en indiquant les dates précises.

```
getSymbols("AAPL",  
           from='2018-01-01', to='2018-12-31') #défini l'intervalle des dates
```

```
## [1] "AAPL"
```

Il est aussi possible d'utiliser la fonction `last()` de `xts` où vous pouvez choisir les données qui date de la dernière année. Voyez l'exemple ci-dessous.

```
getSymbols("GOOG") #charge les données de GOOGLE dans votre environnement
```

```
## [1] "GOOG"
```

```
GOOGLE <- last(AAPL,'1 year') #prend les valeurs de la dernière année de GOOGLE  
head(GOOGLE)
```

```
##           AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted  
## 2018-01-02  42.5400   43.0750  42.3150   43.0650   102223600    41.44208  
## 2018-01-03  43.1325   43.6375  42.9900   43.0575   118071600    41.43486  
## 2018-01-04  43.1350   43.3675  43.0200   43.2575    89738400    41.62732  
## 2018-01-05  43.3600   43.8425  43.2625   43.7500   94640000    42.10126  
## 2018-01-08  43.5875   43.9025  43.4825   43.5875    82271200    41.94489  
## 2018-01-09  43.6375   43.7650  43.3525   43.5825    86336000    41.94008
```

Vous pouvez donc insérer les données financières d'une entreprise pour les dates que vous choisissez dans une variable.

Il est aussi possible d'importer plusieurs indices en une seule fois.

```
getSymbols(c("AAPL", "GOOG"))
```

```
## [1] "AAPL" "GOOG"
```

Si vous souhaitez avoir des informations basiques sur le cours des actions, il est possible d'utiliser les fonctions suivantes : `Op`, `Hi`, `Lo`, `Cl`, `Vo` et `Ad`

```
Open <- Op(AAPL) #Prix d'ouverture  
High <- Hi(AAPL) #Plus haut prix  
Low <- Lo(AAPL) #Plus bas prix  
Close <- Cl(AAPL) #Prix de fermeture  
Volume <- Vo(AAPL) #Volume  
AdjClose <- Ad(AAPL) #Fermeture ajustée
```

Exemple d'utilisation : Nous allons calculer la somme des volumes par années.

```
apply.yearly(Vo(AAPL),sum) #yearly -> année, weekly -> semaine, monthly -> mois
```

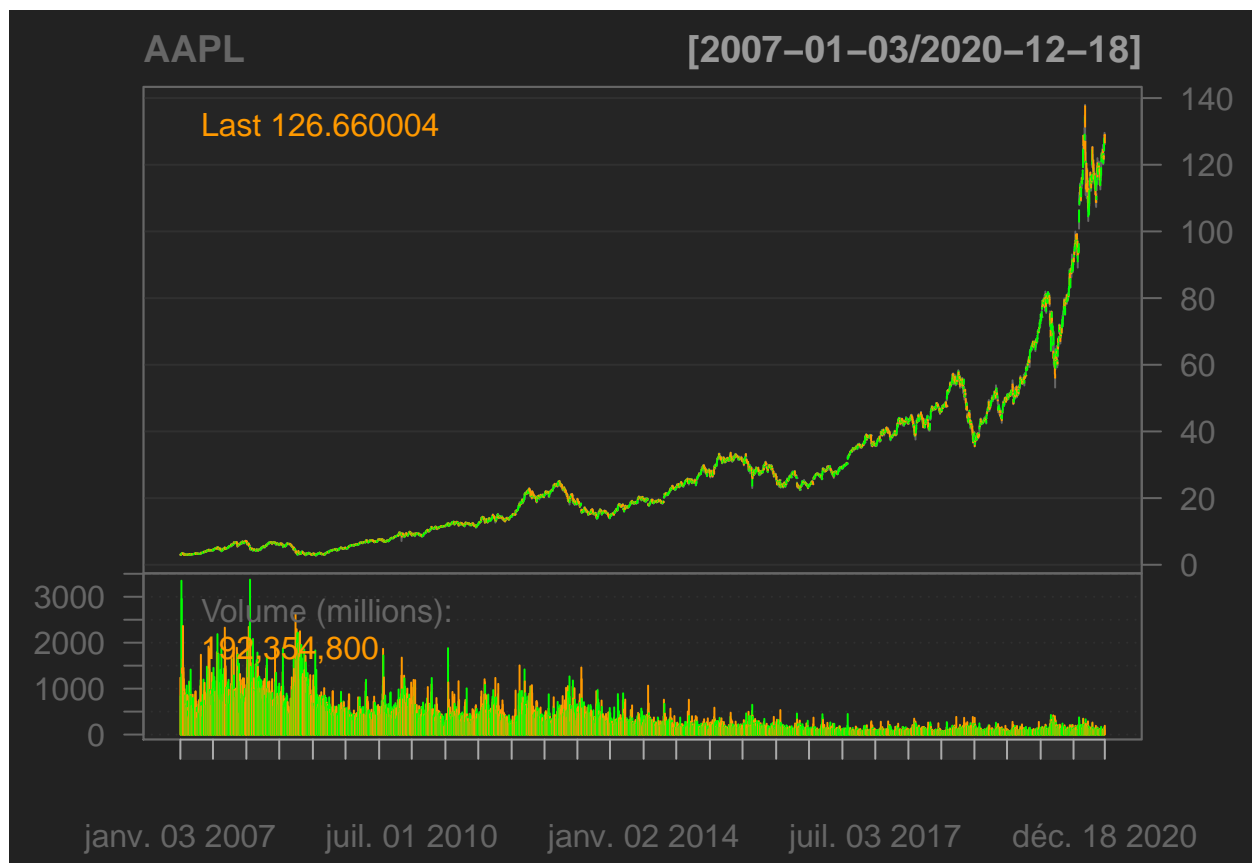
```
##          AAPL.Volume
## 2007-12-31 246995985600
## 2008-12-31 285981206000
## 2009-12-31 143253686800
## 2010-12-31 151024927200
## 2011-12-30 124059339600
## 2012-12-31 131964204400
## 2013-12-31 102421569600
## 2014-12-31 63657952400
## 2015-12-31 52264199600
## 2016-12-30 38729911200
## 2017-12-29 27243106000
## 2018-12-31 34156144800
## 2019-12-31 28254942800
## 2020-12-18 38512571400
```

A présent, vous savez charger des données de différentes sources dans votre espace de travail. Passons à la partie cartographie des données.

## Cartographier les données / Mise en forme

Pour afficher par défaut le cours de l'action (ici de APPLE), il faut utiliser la fonction `chartSeries`.

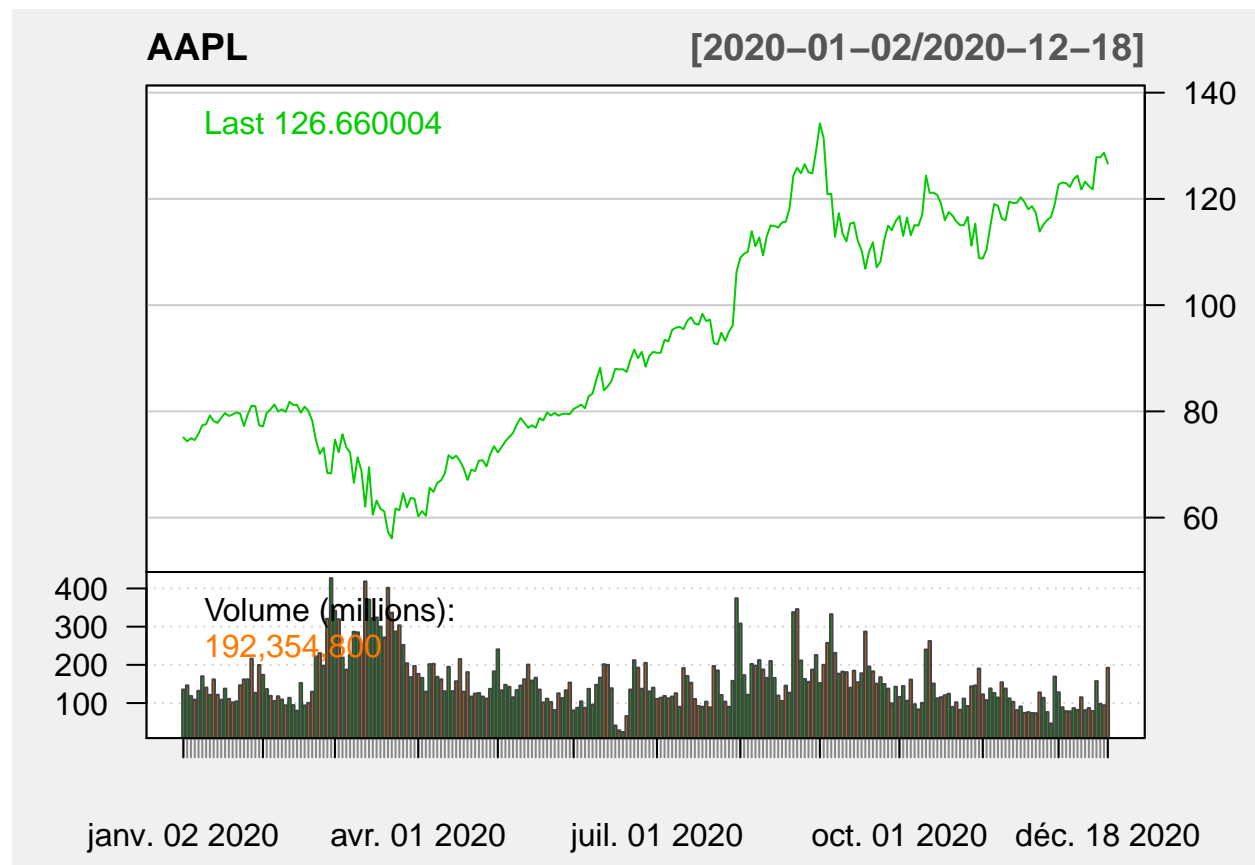
```
chartSeries(AAPL)
```



Si vous souhaitez utiliser différents style, il existe lineChart, barChart et candleChart qui vous sont présentés ci-dessous.

De plus, vous pouvez customiser le graphe à l'aide des paramètres suivants : - “subset” permet de définir la période à afficher - “theme” permet de définir le fond en blanc ou noir par défaut

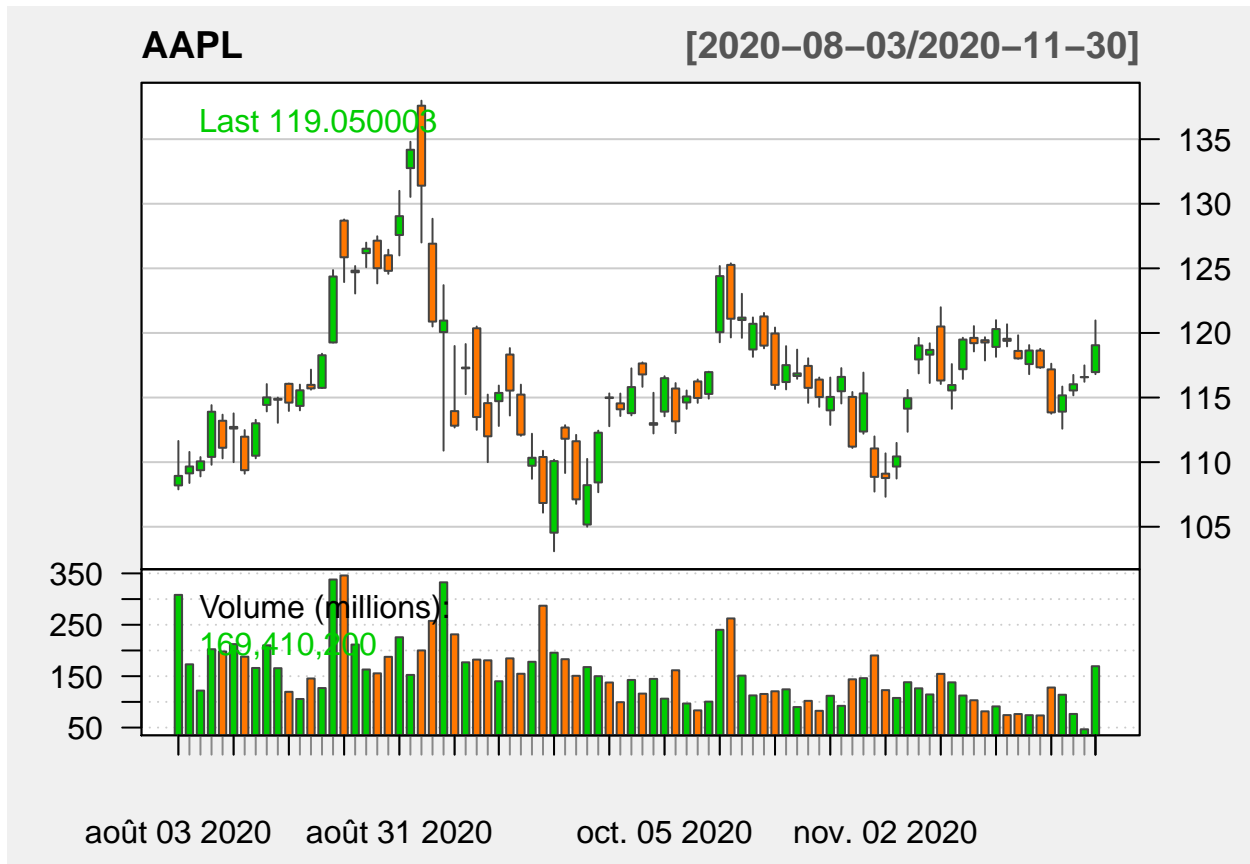
```
lineChart(AAPL,  
  subset="2020", #choix de l'année 2020  
  theme="white") #choix du theme en blanc
```



```
barChart(AAPL,
  subset="2020-11") #choix du mois de novembre 2020
```



```
candleChart(AAPL,  
  subset="2020-08::2020-11", #choix du mois d'aout à novembre 2020  
  theme = "white") #choix du theme en blanc
```

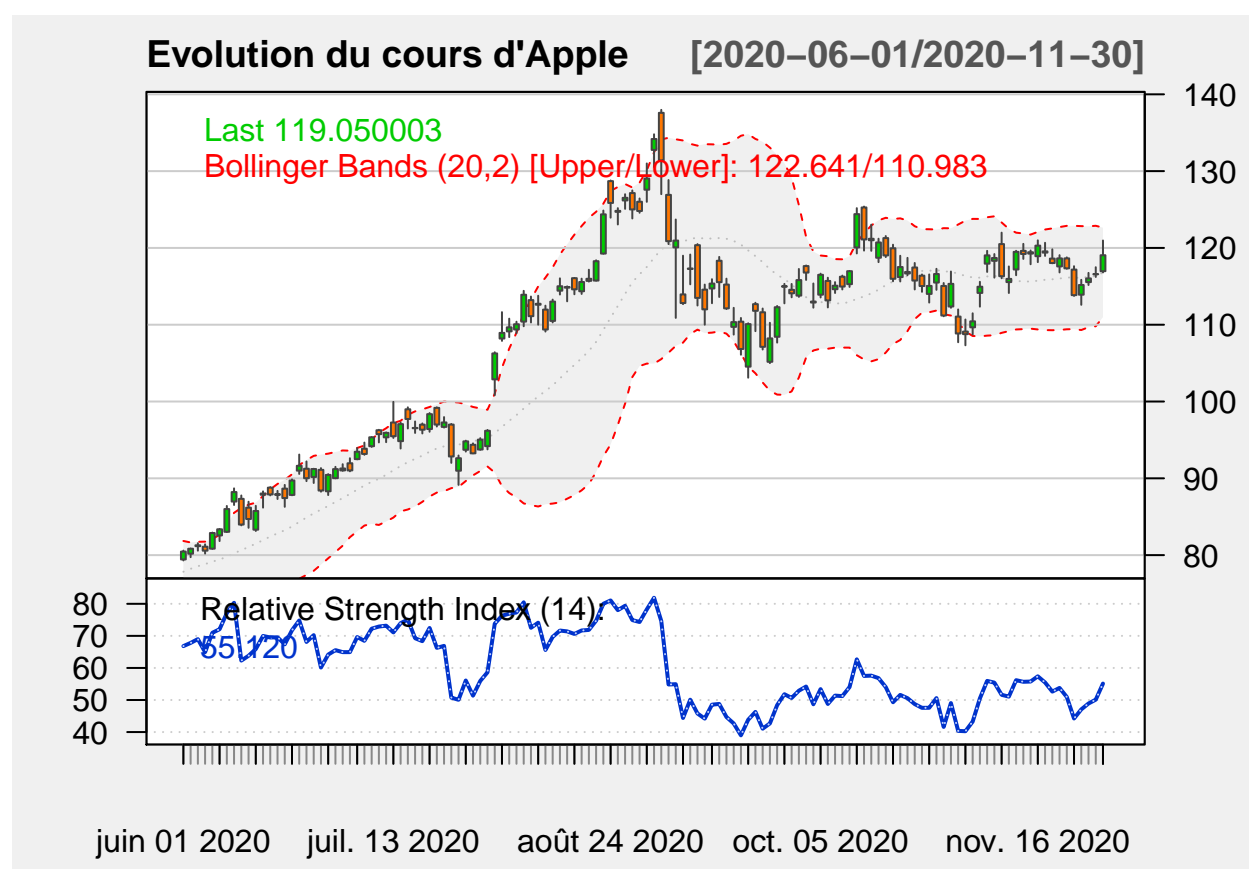


## Ajout et utilisation des indicateurs

Dans cette partie, nous allons utiliser des fonctions issues du package (TTR) : Technical Trading Rule. Ce package est téléchargé automatiquement lors de téléchargement de Quantmod.

Afin d'ajouter des indicateurs, il faut spécifier un paramètre dans chartSeries, c'est "TA", c'est un vecteur d'indicateurs technique.

```
chartSeries(AAPL,  
  name = "Evolution du cours d'Apple", #ajout d'un titre  
  subset="2020-06::2020-11", #choix du mois de mars à novembre 2020  
  type = "candlesticks",  
  theme = "white", #choix du theme en blanc  
  TA = c(addRSI(), addBBands()))#Expl : ajout d'indicateurs
```



Il y a de nombreux indicateurs techniques que vous pouvez utiliser pour faire des analyses très spécifiques. Voici certains de ces indicateurs, avec des explications.

### addWPR()

Le %R de Larry Williams est un indicateur boursier qui participe à l'analyse technique. Il consiste en l'étude du cours de l'action dans le but d'identifier les tendances et d'anticiper l'évolution des marchés.

### addSAR()

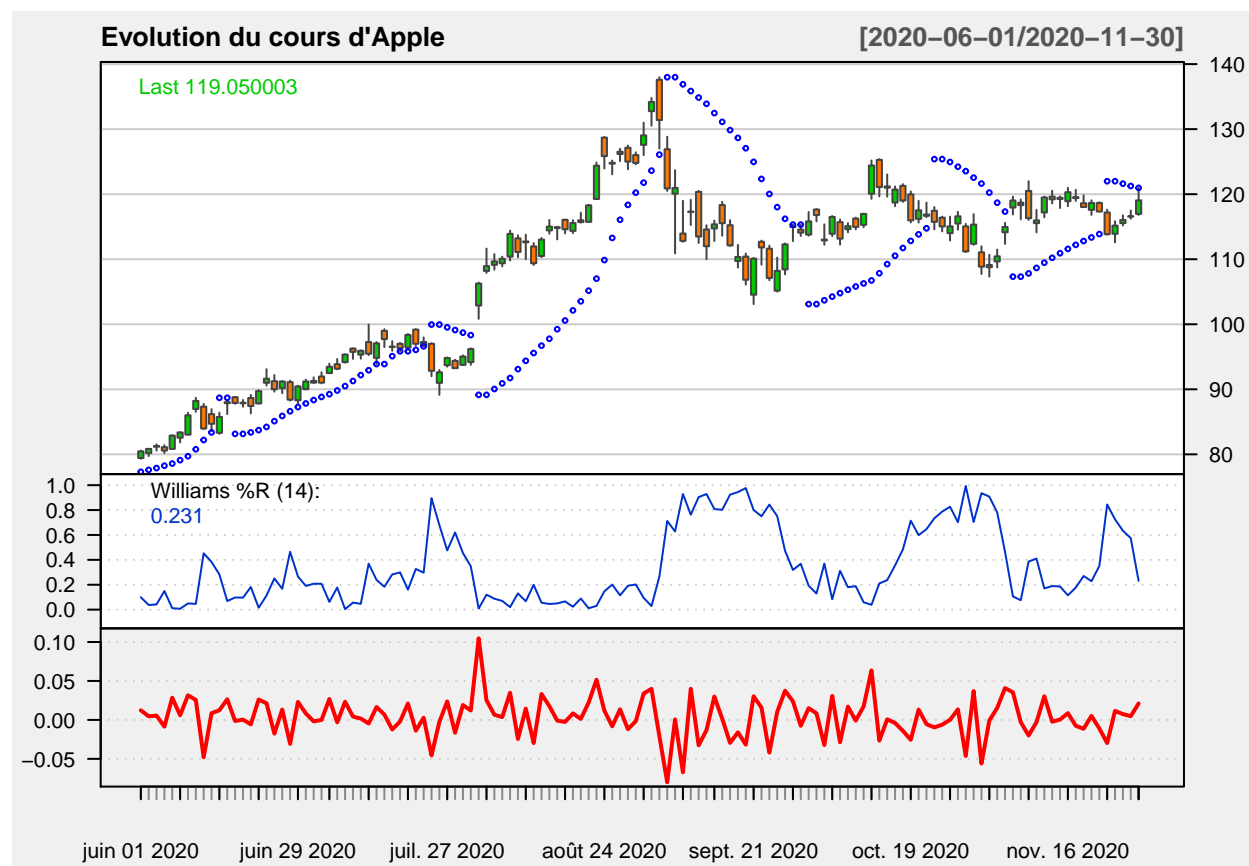
Le SAR parabolique est un indicateur technique utilisé pour déterminer l'orientation des prix d'un actif. Cet indicateur permet d'attirer votre attention sur le moment où l'orientation des prix change. Sur un graphique, l'indicateur apparaît comme une série de points placés au-dessus ou en dessous des barres de prix. Un point

en dessous du prix est considéré comme un signal haussier. Inversement, un point au-dessus du prix est utilisé pour illustrer que les cours sont en contrôle et que l'élan est susceptible de rester à la baisse. Lorsque les points se retournent, cela indique un potentiel changement dans la direction des prix qui est en cours. Par exemple, si les points sont au-dessus du prix, quand ils retournent en dessous du prix, il pourrait signaler une nouvelle hausse des prix.

### addROC()

Le taux de changement (ROC) est la vitesse à laquelle une variable change sur une période de temps spécifique. Graphiquement, le taux de changement est représenté par la pente d'une ligne (en rouge ci-dessous). Le ROC est un concept financier extrêmement important, car il permet aux investisseurs de repérer l'élan de sécurité et d'autres tendances. Par exemple, un titre à forte dynamique, ou qui a un ROC positif, surpasse normalement le marché à court terme. Inversement, un titre qui a un ROC qui tombe en dessous de sa moyenne mobile (moyenne qui s'affranchit des aberrations de cours), ou celui qui a un ROC faible ou négatif est susceptible de baisser en valeur. C'est peut être le moment de vendre dans ce cas...

```
chartSeries(AAPL,
  name = "Evolution du cours d'Apple",
  subset="2020-06::2020-11",
  type = "candlesticks",
  theme = "white",
  TA = c(addWPR(), addSAR(), addROC()))
```



### addRSI()

Le Relative Strength Index (RSI) est un indicateur très populaire mis au point par Welles Wilder en 1978. Il est généralement utilisé pour mesurer la force ou la "pression" des cours. Si le RSI est compris entre 0 et 30, il faut acheter et il faut vendre lorsque le RSI est entre 70 et 100.



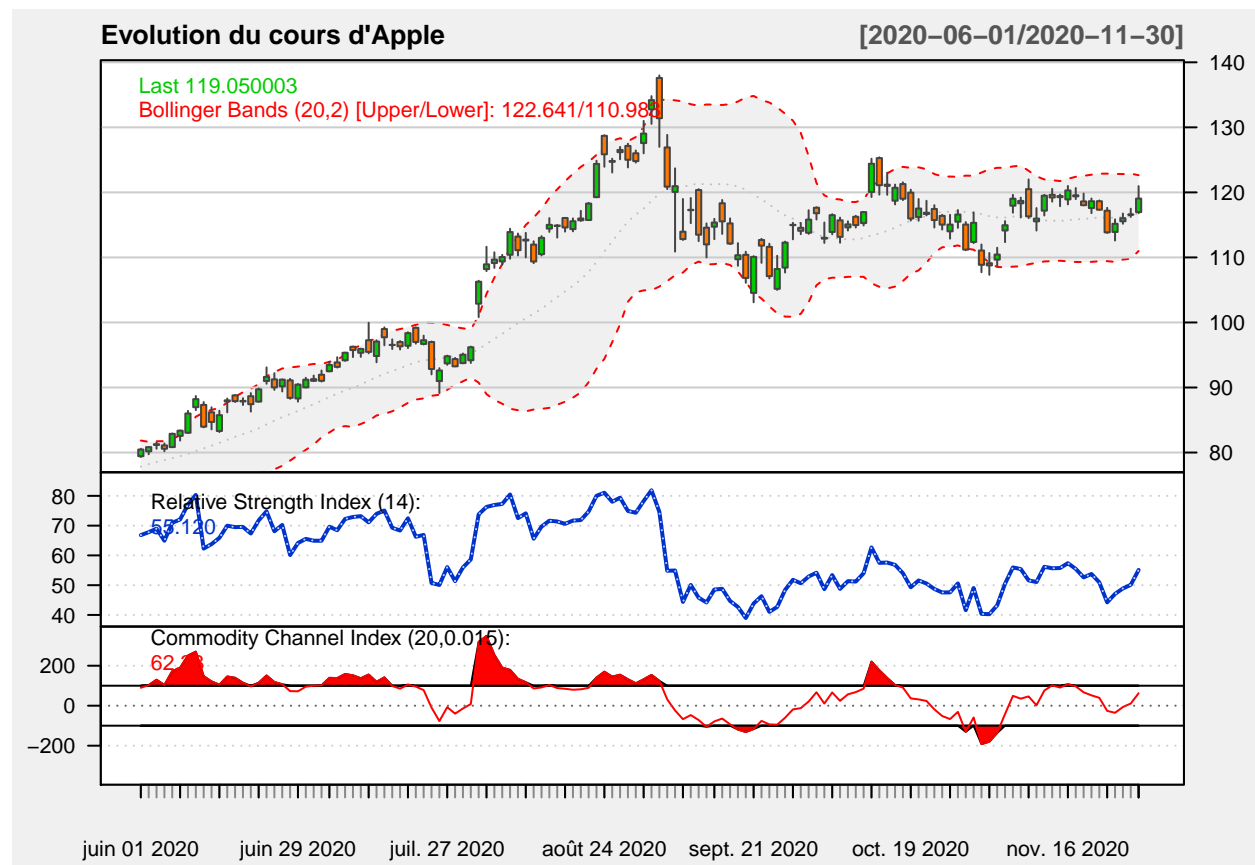
### addCCI()

Le CCI Commodity Channel Index, est un indicateur mis au point par Donald Lambert durant les années 1980. Le CCI nous montre les situations de surachat (avec un CCI élevé) et celles de surventes (avec un CCI faible) pour une unité de temps. L'indicateur CCI est présenté dans un graphe avec une échelle qui est généralement comprise entre -500 et +500 mais d'autres échelles sont possibles. Vous pourrez généralement repérer l'indicateur entre -100 et +100. Ce qu'il faut savoir sur le CCI: - + 100, il indique une situation de surachat - -100, il indique une situation de survente

### addBBands()

Cet indicateur a été développé par John Bollinger. Une bande de Bollinger se compose de 3 lignes. Une moyenne mobile simple (SMA) et deux lignes supplémentaires représentent 2 écarts types au-dessus et au-dessous de la SMA. L'écart-type mesure la volatilité des actions et donc, lorsque les marqueurs sont plus volatils, les bandes de Bollinger s'élargissent. Lorsque le marché est plat, les bandes de Bollinger se contractent.

```
chartSeries(AAPL,  
  name = "Evolution du cours d'Apple",  
  subset="2020-06:2020-11",  
  type = "candlesticks",  
  theme = "white",  
  TA = c(addRSI(), addCCI(), addBBands()))
```



Ci-dessous, un tableau avec les différents indicateurs et le nom de la fonction Quantmod pour afficher l'indicateur sur le graphe.

INDICATEURS	NOM DE LA FONCTION QUANTMOD
Welles Wilder's Directional Movement Indicator	addADX
Average True Range	addATR
Bollinger Bands	addBBands
Commodity Channel Index	addCCI
Chaiken Money Flow	addCMF
Chande Momentum Oscillator	addCMO
Double Exponential Moving Average	addDEMA
Detrended Price Oscillator	addDPO
Exponential Moving Average	addEMA
Price Envelope	addEnvelope
Exponential Volume Weigthed Moving Average	addEVWMA
Options and Futures Expiration	addExpiry
Moving Average Convergence Divergence	addMACD
Momentum	addMomentum
Rate of Change	addROC
Relative Strength Indicator	addRSI
Parabolic Stop and Reverse	addSAR
Simple Moving Average	addSMA
Stochastic Momentum Index	addSMI
Triple Smoothed Exponential Oscillator	addTRIX
Volume	addVo
Weighted Moving Average	addWMA
Williams %R	addWPR
ZLEMA	addZLEMA

Si vous souhaitez obtenir des informations sur les indicateurs sans passer par un affichage graphique, il est possible d'obtenir les données sous un tableau. Ci-dessous, des exemples de lecture de données sur l'année 2020 chez Apple.

```
getSymbols("AAPL",
           from='2020-01-01', to='2020-12-31')
```

```
## [1] "AAPL"
```

```
sma <-SMA(Cl(AAPL)) #intégration dans une variable de SMA, syntaxe par défaut
tail(sma,n=5) #lecture des 5 dernières données Simple Moving Average (SMA)
```

```
##          SMA
## 2020-12-14 122.833
## 2020-12-15 123.349
## 2020-12-16 123.822
## 2020-12-17 124.398
## 2020-12-18 124.839
```

```
bb <-BBands(Cl(AAPL),s.d=2) #syntaxe par défaut
tail(bb,n=10) #lecture des 10 dernières données Boolinger Band (BBands)
```

```
##           dn      mavg      up      pctB
## 2020-12-07 113.3597 118.9690 124.5783 0.9261683
## 2020-12-08 113.4334 119.3720 125.3106 0.9216490
## 2020-12-09 113.8509 119.6625 125.4741 0.6821800
## 2020-12-10 113.8344 119.8500 125.8656 0.7817677
## 2020-12-11 113.9015 120.0100 126.1185 0.6964480
## 2020-12-14 113.9907 120.1360 126.2813 0.6337614
## 2020-12-15 113.5023 120.5150 127.5277 1.0251169
## 2020-12-16 113.2640 120.9360 128.6080 0.9479926
## 2020-12-17 113.2180 121.4695 129.7210 0.9381297
## 2020-12-18 113.4306 121.8705 130.3104 0.7837418
```

```
# "dn"= La bande inférieure de Bollinger, "mavg" = la moyenne mobile,
# "up"= La bande supérieur de Bollinger, "pctB" = le calcul %B
```

```
ema <-EMA(Cl(AAPL))
tail(ema,n=5)
```

```
##           EMA
## 2020-12-14 121.9214
## 2020-12-15 123.0048
## 2020-12-16 123.8785
## 2020-12-17 124.7551
## 2020-12-18 125.1015
```

## Conclusion

Vous avez pu constater les différentes particularités du package Quantmod. Cependant, il existe encore de nombreuses fonctionnalités. Il est possible de faire fonctionner Quantmod avec les packages suivant afin de pousser l'analyse des cours et de créer soit même ces signaux (pour l'achat ou la vente par exemple).

Si vous souhaitez développer vos propres analyses, télécharger **Performance Analytics** : `install.packages("PerformanceAnalytics") library(PerformanceAnalytics)`

Afin de réaliser des études visuelles et dynamique, vous pouvez utiliser la donnée issue des indicateurs pour créer des dashboard avec le package Shiny par exemple.

Les possibilités d'études, d'analyses sont très variées :

- créer un signal d'achat basé sur une règle de filtrage simple
- créer un signal d'achat basé sur le RSI
- prévoir les échanges en dehors de la journée, en suivant à la fois les liquidités et les actions détenues

A présent, à vous de jouer !

## Sources

### **Documentation fonction Quantmod**

[<https://www.rdocumentation.org/packages/quantmod/versions/0.4.18>]

### **Comprendre et utiliser Quantmod**

[<https://www.quantmod.com/>]