T02

# R語言
# 金融數據分析

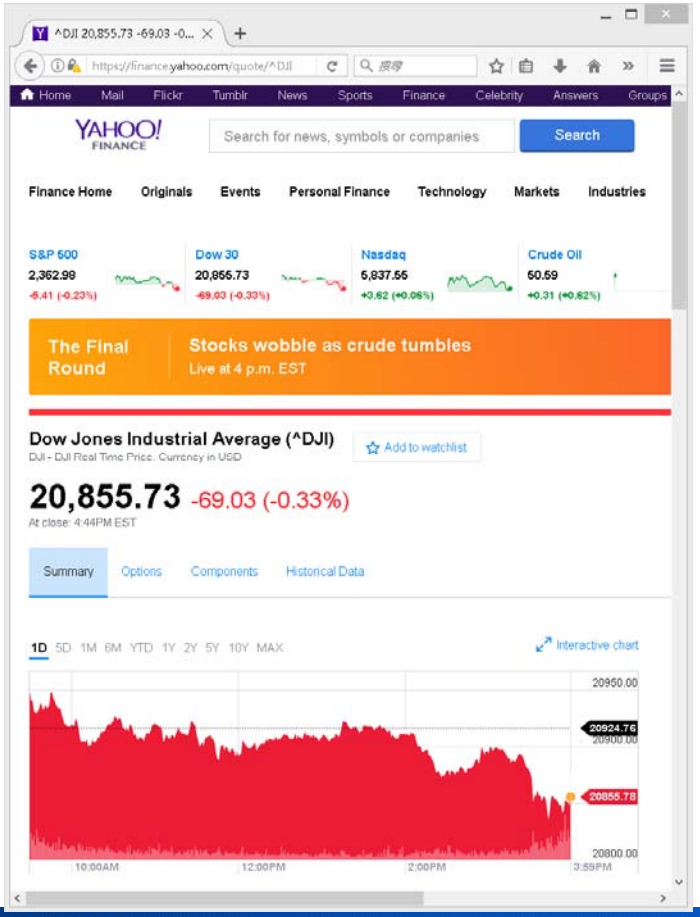
**吳漢銘**

國立政治大學 統計學系


http://www.hmwu.idv.tw

# quantmod
## Quantitative Financial Modelling Framework

- **quantmod**: Quantitative Financial Modelling Framework
- **getSymbols {quantmod}**: Load and Manage Data from Multiple Sources
  - Current **src** methods available are: yahoo, google, MySQL, FRED, csv, RData, and oanda.
- http://www.quantmod.com/
- http://statmath.wu.ac.at/~hornik/QFS1/quantmod-vignette.pdf



# 利用quantmod抓取股市資料

```
> install.packages("quantmod")
> library(quantmod)
> getSymbols("AAPL", src="yahoo")
[1] "AAPL"
> dim(AAPL)
[1] 3525    6
> class(AAPL)
[1] "xts" "zoo"
> head(AAPL, 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      2.581791
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
> # Open: 當日開盤價; High: 當日最高價; Low: 當日最低價; Close: 當日收盤價;
> # Volume: 成交量; Adjust: 調整後股價; # 開高低收: OHLC
> tail(AAPL, 3)
           AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
2020-12-29    138.05    138.79   134.34     134.87   121047300        134.87
2020-12-30    135.58    135.99   133.40     133.72    96452100        133.72
2020-12-31    134.08    134.74   131.72     132.69    98990400        132.69
> head(AAPL$AAPL.Open, 3)
           AAPL.Open
2007-01-03  3.081786
2007-01-04  3.001786
2007-01-05  3.063214
```

```
> getSymbols("AAPL") #default src="yahoo"
```

```
# xts: eXtensible Time Series
# zoo: S3 Infrastructure for Regular and Irregular
Time Series (Z's Ordered Observations)
```

# 利用quantmod抓取股市資料

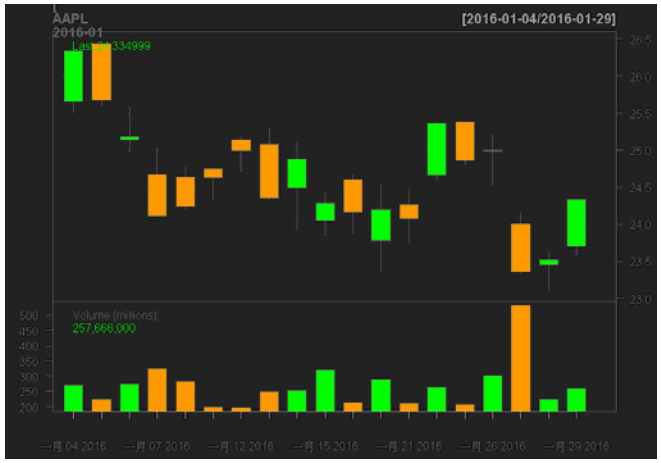
```
> myApple <- get(getSymbols("AAPL", src="google"))
錯誤: ‘getSymbols.google’ is defunct.
Google Finance stopped providing data in March, 2018.
You could try setting src = "yahoo" instead.
See help("Defunct") and help("quantmod-defunct")
> myApple <- get(getSymbols("AAPL"))
> head(myApple, 3)
           AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
2007-01-03  3.081786  3.092143 2.925000   2.992857  1238319600      2.581791
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
> tail(myApple, 3)
           AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
2020-12-29    138.05    138.79   134.34     134.87   121047300        134.87
2020-12-30    135.58    135.99   133.40     133.72    96452100        133.72
2020-12-31    134.08    134.74   131.72     132.69    98990400        132.69
> class(myApple)
[1] "xts" "zoo"
> myApple.df <- as.data.frame(myApple)
> class(myApple.df)
[1] "data.frame"
>
> AAPL.df <- as.data.frame(AAPL)
> class(AAPL.df)
[1] "data.frame"
```

# 畫趨勢圖

```
chartSeries(AAPL)
chartSeries(AAPL, subset="2016")
chartSeries(AAPL, subset="2016-09::2016-12")
chartSeries(AAPL, subset="2016-09::2016-12", type = "line", theme = "white")
```

# 畫趨勢圖

```
chartSeries(AAPL["2016"])
chartSeries(AAPL["2016-01"])
chartSeries(AAPL["2016-01/2016-03"])
chartSeries(AAPL["2016-01-01/2016-03-15"])
```



# 畫趨勢圖

```
candleChart(AAPL["2016-01/2016-03"])
barChart(AAPL["2016-01::2016-03"])
```

# 選取股市某時期的資料

```
> start.date <- as.Date("2012-02-01")
> end.date <- as.Date("2012-02-28")
> APPL.201202  <-  AAPL[index(AAPL) > start.date & index(AAPL) < end.date]
> head(APPL.201202, 3)
           AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
2012-02-02  16.28214  16.32750 16.21357   16.25429   186796400      14.02178
2012-02-03  16.33214  16.42857 16.27000   16.41714   286599600      14.16226
2012-02-06  16.37071  16.60643 16.36429   16.57036   249412800      14.29443>
tail(APPL.201202, 3)
           AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
2012-02-23  18.39571  18.49393 18.19643   18.44250   568027600      15.90944
2012-02-24  18.55964  18.67500 18.52286   18.65750   415072000      16.09491
2012-02-27  18.61821  18.87500 18.43857   18.77714   547582000      16.19812
> AAPL.201604 <- get(getSymbols("AAPL", from = as.Date("2016-04-01"),
+                               to = as.Date("2016-04-30"), src="yahoo"))
> index(AAPL.201604)
 [1] "2016-04-01" "2016-04-04" "2016-04-05" "2016-04-06" "2016-04-07" "2016-04-08"
...
[19] "2016-04-27" "2016-04-28" "2016-04-29"
```

```
> # multiple stocks
> getSymbols(c("VZ", "AAPL", "MMM", "IBM"))
[1] "VZ"   "AAPL" "MMM"  "IBM"
> head(MMM, 3)
           MMM.Open MMM.High MMM.Low MMM.Close MMM.Volume MMM.Adjusted
2007-01-03    77.53    78.85   77.38     78.26    3781500     59.92042
2007-01-04    78.40    78.41   77.45     77.95    2968400     59.68306
2007-01-05    77.89    77.90   77.01     77.42    2765200     59.27726
```

# 儲存股市資料

```
> getwd()
[1] "D:/myR"
> # 儲存資料(Rdata格式)
> # saveSymbols(file.path=".") # save all Symbols in the current directory
> saveSymbols(c("AAPL", "MMM"), file.path=".")
> list.files()
[1] "AAPL.RData"    "Data"    "MMM.RData"    "mysql-connector-odbc-5.3.7-winx64.msi"
> # 刪除此R session的Symbols資料 # removeSymbols() 刪除全部
> removeSymbols("AAPL")
> # 讀取在電腦端的資料(Rdata格式)
> load("AAPL.RData")
>
> # getSymbols取得的股票資料儲存成csv檔
> write.zoo(IBM, "IBM.csv", sep = ",", qmethod = "double") # or
> # 讀取股票資料csv檔進入R，成為xts類別物件  (假設工作目錄下已有IBM.csv檔案)
> removeSymbols("IBM") # > showSymbols()
> getSymbols.csv("IBM", env=globalenv())
[1] "IBM"
> head(IBM, 3)
           IBM.Open IBM.High IBM.Low IBM.Close IBM.Volume IBM.Adjusted
2007-01-03    97.18    98.40   96.26     97.27    9196800     77.73997
2007-01-04    97.25    98.79   96.88     98.31   10524500     78.57116
2007-01-05    97.60    97.95   96.91     97.42    7221300     77.85985
>
> # 以data.frame儲存成csv檔
> write.csv(as.data.frame(IBM), "IBM_2.csv")
> getSymbols.csv('IBM_2', env=globalenv())
[1] "IBM_2"
```

**showSymbols()**: # Objects loaded by getSymbols with auto.assign=TRUE can be viewed with showSymbols

# 下載台股股價歷史資料

- 下載股價資料前需知道台股的代碼，quantmod內建資料庫來源是yahoo finance.
- https://finance.yahoo.com/
- https://tw.stock.yahoo.com/h/kimosel.php
- http://investexcel.net/all-yahoo-finance-stock-tickers/
- http://teachercannon.weebly.com/uploads/6/8/3/8/6838442/smg_tickers.pdf
- https://tw.stock.yahoo.com/h/getclass.php

```
> symbols <- stockSymbols()  # Get investment data from the internet
Fetching NASDAQ symbols...
Fetching non-NASDAQ symbols...
> head(symbols)
  Symbol                                                        Name LastSale
1   AAMC                                Altisource Asset Management Corp  76.6000
2    AAU                                          Almaden Minerals, Ltd.   0.9901
3    ABE Aberdeen Emerging Markets Smaller Company Opportunities Fund I  12.0900
4    ACU                                        Acme United Corporation.  24.5700
5    ACY                                               AeroCentury Corp.  10.2000
6   ADGE                                         American DG Energy Inc.   0.3119
  MarketCap IPOyear           Sector                        Industry Exchange
1  $117.23M      NA          Finance                     Real Estate     AMEX
2   $89.16M    2015 Basic Industries                Precious Metals     AMEX
3   $116.6M      NA             <NA>                            <NA>     AMEX
4   $81.69M    1988    Capital Goods Industrial Machinery/Components     AMEX
5   $15.98M      NA       Technology Diversified Commercial Services     AMEX
6   $15.81M      NA Public Utilities                Power Generation     AMEX
```

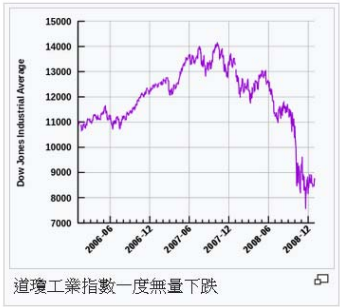
# 抓取台積電(2330)資料並繪出股價圖

```
> TSMC <- getSymbols("2330.TW", auto.assign = FALSE)
> head(TSMC)
           2330.TW.Open 2330.TW.High 2330.TW.Low 2330.TW.Close 2330.TW.Volume 2330.TW.Adjusted
2007-01-02      67.4627      67.6617     66.9653       67.3632       26160600          47.3376
...
2007-01-09      65.5722      66.1692     65.5722       65.6717       26187900          46.1489
> # Must use auto.assign=TRUE for multiple Symbols requests
> getSymbols(c("2330.TW", "2303.TW", "2337.TW"))
[1] "2330.TW" "2303.TW" "2337.TW"
> head(2330.TW) # error
錯誤: unexpected symbol in "head(2330.TW"
> head("2330.TW") # wrong
[1] "2330.TW"
> head('2330.TW') # wrong, Enter鍵旁的「'」
[1] "2330.TW"
> head(`2330.TW`) # correct, Tab 鍵上方的「`」
           2330.TW.Open 2330.TW.High 2330.TW.Low 2330.TW.Close 2330.TW.Volume 2330.TW.Adjusted
2007-01-02      67.4627      67.6617     66.9653       67.3632       26160600          47.3376
...
2007-01-09      65.5722      66.1692     65.5722       65.6717       26187900          46.1489
>
> TSMC <- get("2330.TW") # or  TSMC <- `2330.TW`
> UMC <- get("2303.TW")
> MXIC <- get("2337.TW")
> head(UMC) # View(UMC)
           2303.TW.Open 2303.TW.High 2303.TW.Low 2303.TW.Close 2303.TW.Volume 2303.TW.Adjusted
2007-01-01      20.2501      20.2501     20.2501       20.2501              0          20.6975
...
2007-01-08      19.9002      20.0001     19.7001       19.7001       41413700          20.1353
```

# 抓取台積電(2330)資料並繪出股價圖

chartSeries(TSMC)

chartSeries(UMC)




Dow Jones index

getSymbols("^DJI")
chartSeries(DJI)

後期發生後延續至今的全球金融危機。關於由此引發的的經濟危機，請見「**經濟大衰退**」。

**2007年–2008年全球金融危機**（英語：Financial crisis of 2007–08），又稱**2008年世界金融危機**、**次貸危機**、**信用危機**、**2008年華爾街金融危機**、**2008年金融崩潰**，在2008年又出現了**金融海嘯**及**華爾街海嘯**等名稱，是一場在2007年8月9日開始浮現的金融危機。自次級房屋信貸危機爆發後，投資者開始對抵押證券的價值失去信心，引發流動性危機。即使多國中央銀行多次向金融市場注入巨額資金，也無法阻止這場金融危機的爆發。直到2008年9月[1]，這場金融危機開始失控，並導致多間相當大型的金融機構倒閉或被政府接管，並引發經濟衰退。

getSymbols("^TWII")

# 日、月、季、年資料

**to.period {xts}**

Convert time series data to an OHLC series

```
> class(TSMC)
[1] "xts" "zoo"
> dim(TSMC)
[1] 3450    6
> head(TSMC, 3)
           2330.TW.Open 2330.TW.High 2330.TW.Low 2330.TW.Close 2330.TW.Volume 2330.TW.Adjusted
2007-01-02      66.7904      66.9874     66.2978       66.6919       26292531         38.60290
2007-01-03      66.7904      67.6770     66.4948       66.9874       42463337         38.77394
2007-01-04      66.6919      67.0859     66.5933       66.6919       22444225         38.60290
> TSMC.m <- to.monthly(TSMC)
> head(TSMC.m, 3)
          TSMC.Open TSMC.High TSMC.Low TSMC.Close  TSMC.Volume TSMC.Adjusted
一月 2007   66.7904   70.2806  63.0470    66.3963  97301940531      38.43180
二月 2007   66.7904   69.0561  64.8202    68.2680    602888829      39.51518
三月 2007   65.7067   69.7457  62.0471    66.8889 109758751296      38.71693
> TSMC.q <- to.quarterly(TSMC)
> head(TSMC.q, 3)
        TSMC.Open TSMC.High TSMC.Low TSMC.Close  TSMC.Volume TSMC.Adjusted
2007 Q1   66.7904   70.2806  62.0471    66.8889 207663580656      38.71693
2007 Q2   66.7904   70.2924  63.2631    70.1934   3336169324      42.49558
2007 Q3   70.1934   72.3714  58.0159    62.9661 226144608065      38.12013
> TSMC.y <- to.yearly(TSMC)
> head(TSMC.y, 3)
           TSMC.Open TSMC.High TSMC.Low TSMC.Close  TSMC.Volume TSMC.Adjusted
2007-12-31   66.7904   72.3714  56.8279    58.9070 8.944733e+11      35.66272
2008-12-31   61.3821   69.1043  34.0796    44.1791 1.211131e+12      28.23670
2009-12-31   44.1791   65.2000  37.9602    64.5000 9.349141e+10      43.56685>
> TSMC.2012 <- TSMC["2012"]
> TSMC.2012w <- to.weekly(TSMC.2012)
```

# chartSeries {quantmod}

```
chartSeries(x,
            type = c("auto", "candlesticks", "matchsticks", "bars","line"),
            subset = NULL,
            show.grid = TRUE,
            name = NULL,
            time.scale = NULL,
            log.scale = FALSE,
            TA = 'addVo()',
            TAsep=';',
            line.type = "l",
            bar.type = "ohlc",
            theme = chartTheme("black"),
            layout = NA,
            major.ticks='auto', minor.ticks=TRUE,
            yrange=NULL,
            plot=TRUE,
            up.col,dn.col,color.vol = TRUE, multi.col = FALSE,
            ...)
```

```
> chartSeries(TSMC.2012w)
> chartSeries(TSMC.2012w, theme="white", grid=T,
   up.col="red", dn.col="green")
> chartSeries(TSMC.2012w, TA=NULL)    #no volume
> chartSeries(TSMC.2012w,TA=c(addVo(), addBBands()))
```

```
myTheme <- chartTheme("white", up.col="red", dn.col="green")
chartSeries(TSMC.2012w, theme=myTheme, grid=T)
```

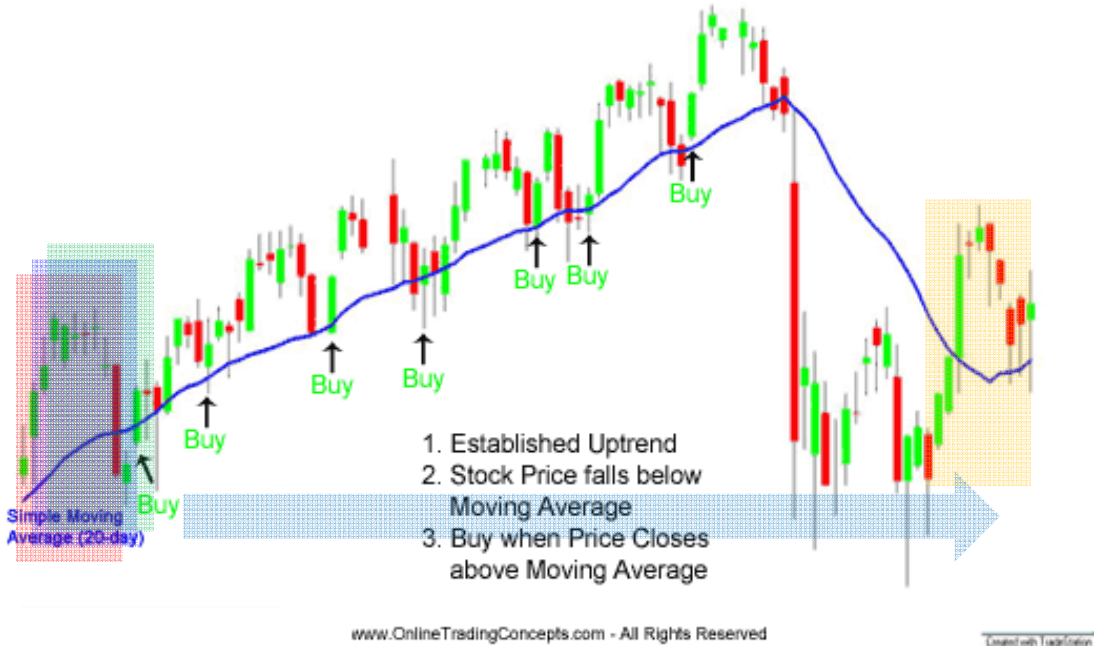
# Simple Moving Average

- A moving average (移動平均) (簡稱均線) is a calculation to analyze data points by creating series of averages of different subsets of the full data set.

$$SMA = \frac{p_1 + p_2 + \cdots + p_n}{n}$$

**When price is in an uptrend** and subsequently, the moving average is in an uptrend, and the moving average has been tested by price and price has bounced off the moving average a few times (i.e. the moving average is serving as a support line), then a trader might buy on the **next pullbacks back** to the Simple Moving Average.

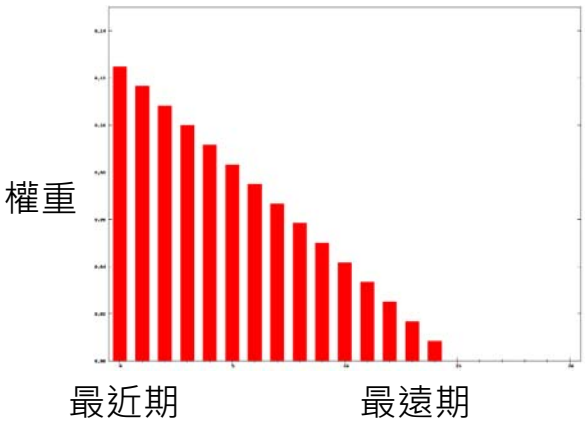
Moving Average Acting as Support
- Potential Buy Signal

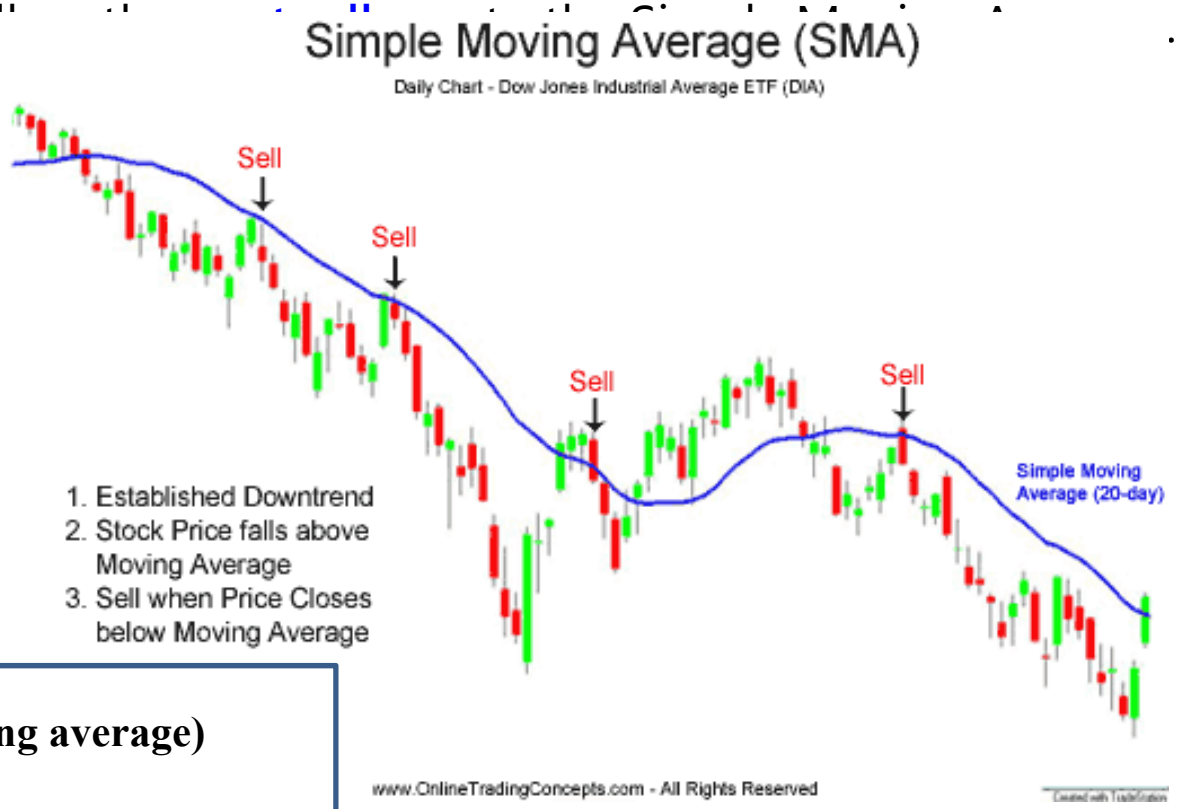



http://www.onlinetradingconcepts.com/TechnicalAnalysis/MASimple.html

# Moving Average Acting as Resistance - Potential Sell Signal

- **At times when price is in a downtrend** and the moving average is in a downtrend as well, and price tests the SMA above and is rejected a few consecutive times (i.e. the moving average is serving as a resistance line), then a trader might se




**An n-day WMA (Weighted moving average)**

$$\mathrm{WMA}_M = \frac{np_M + (n-1)p_{M-1} + \cdots + 2p_{(M-n+2)} + p_{(M-n+1)}}{n + (n-1) + \cdots + 2 + 1}$$

http://www.onlinetradingconcepts.com/TechnicalAnalysis/MASimple.html

# Smoothing in R

**`smooth`: Forecasting Using Smoothing Functions**

https://cran.r-project.org/web/packages/smooth/index.html

- `es()` - Exponential Smoothing;
- `ssarima()` - State-Space ARIMA, also known as Several Seasonalities ARIMA;
- `ces()` - Complex Exponential Smoothing;
- `ges()` - Generalised Exponential Smoothing;
- `ves()` - Vector Exponential Smoothing;
- `sma()` - Simple Moving Average in state-space form;

**`TTR`: Technical Trading Rules**

https://cran.r-project.org/web/packages/TTR/index.html

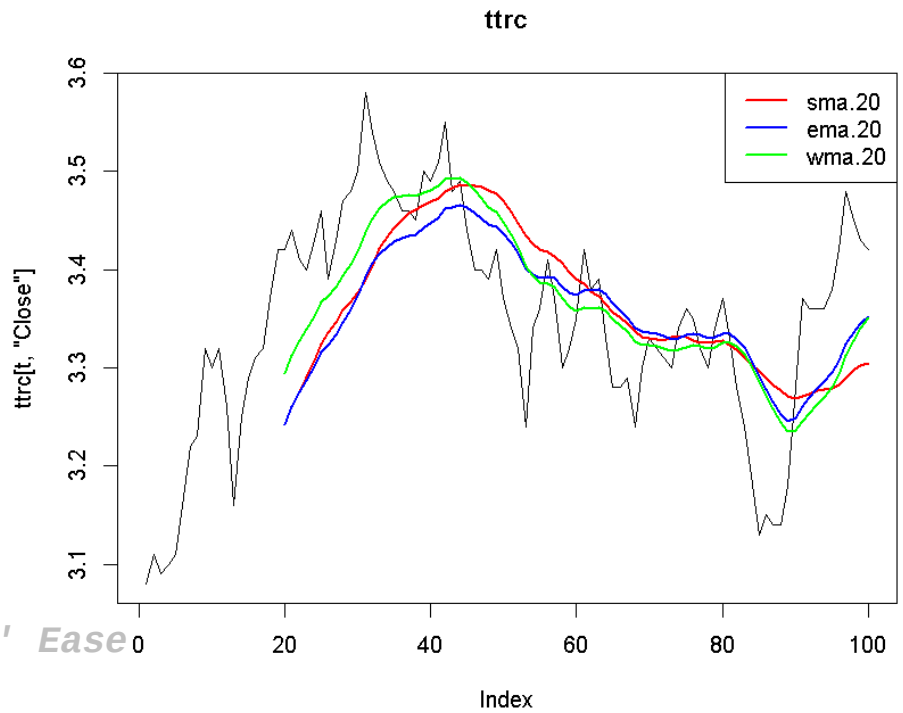
```
SMA(x, n = 10, ...)
EMA(x, n = 10, wilder = FALSE, ratio = NULL, ...)
DEMA(x, n = 10, v = 1, wilder = FALSE, ratio = NULL)
WMA(x, n = 10, wts = 1:n, ...)
EVWMA(price, volume, n = 10, ...)
ZLEMA(x, n = 10, ratio = NULL, ...)
VWAP(price, volume, n = 10, ...)
VMA(x, w, ratio = 1, ...)
HMA(x, n = 20, ...)
ALMA(x, n = 9, offset = 0.85, sigma = 6, ...)
```

# Example

**ttrc {TTR}**: Technical Trading Rule Composite data

Historical Open, High, Low, Close, and Volume data for the periods January 2, 1985 to December 31, 2006. Randomly generated.

```
> # install.packages("TTR")
> library(TTR)
> data(ttrc)
> dim(ttrc)
[1] 5550    6
> head(ttrc)
        Date Open High  Low Close  Volume
1 1985-01-02 3.18 3.18 3.08  3.08 1870906
2 1985-01-03 3.09 3.15 3.09  3.11 3099506
3 1985-01-04 3.11 3.12 3.08  3.09 2274157
4 1985-01-07 3.09 3.12 3.07  3.10 2086758
5 1985-01-08 3.10 3.12 3.08  3.11 2166348
6 1985-01-09 3.12 3.17 3.10  3.16 3441798

> t <- 1:100
> sma.20 <- SMA(ttrc[t, "Close"], 20)
> ema.20 <- EMA(ttrc[t, "Close"], 20) # Arms' Ease
> wma.20 <- WMA(ttrc[t, "Close"], 20)
>
> plot(ttrc[t,"Close"], type="l", main="ttrc")
> lines(sma.20, col="red", lwd=2)
> lines(ema.20, col="blue", lwd=2)
> lines(wma.20, col="green", lwd=2)
> legend("topright", legend=c("sma.20", "ema.20", "wma.20"),
+        col=c("red", "blue", "green"), lty=1, lwd=2)
```

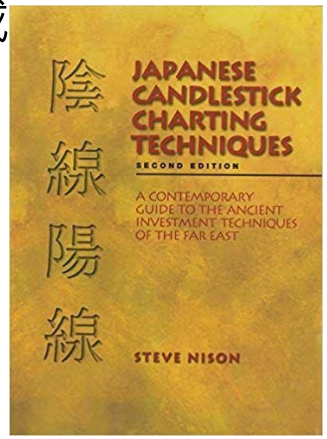


# 趨勢圖+平滑曲線

```
getSymbols("BABA")
head(BABA)
chartSeries(BABA["2014-10-01/2015-03-31"],
            theme=chartTheme("white", up.col="red", dn.col="green"),
            name="Ali BABA K-line", show.grid=TRUE)

addSMA(10) #10日簡單移動平均線
# 策略: 當快線(成交價)向上穿過慢線(MA值)，可買進。
#       當慢線穿過快線，可出場。

# 移動平均線(MA)
addEMA(24, col="blue")

# 指數平滑異同移動平均線(MACD)
# 用於判斷股票的價格變化
# 計算方式=兩條EMA計算差值(DIF)，
# 再對差值進行移動平均計算
addMACD()
```

# K線圖

- K線圖(Candlestick Charts)，又稱為蠟燭線、日本線、陰陽線、紅黑線。
- 源於17世紀，日本江戶時代的大米商**本間宗久**(1724-1803)所發明。
- K線是根據股價一天(或者某一周期)走勢中形成的四個價位：
  開盤價、最高價、最低價、收盤價(開O、高H、低L、收C)繪製而成




**富二代的前半生：大獲成功，再跌落谷底**

- 宗久受海晏寺住持的啟發，開始從其他角度關注市場價格的變化。
- 他到大阪的堂島交易所，記錄每一筆交易: 開盤價，最高價，最低價，收盤價。
- 在昏暗的房間裡，他取光的蠟燭的形象，繪製了世界上第一幅蠟燭圖。據說自那以後，宗久在大米交易中再也沒有失手過。
- 宗久成為了百戰百勝的傳說，人們稱他為「相場之神」。

原文網址：https://kknews.cc/finance/bp9pmxo.html

- 將近190年的時間裡，蠟燭圖這一偉大的發明都不為人知。直到20世紀末，美國人Steve Nison從一名日本經紀商那裡發現了蠟燭圖的存在。
- 1991年，Steve Nison出版了《陰線陽線》一書，把本間宗久發明的蠟燭圖介紹到了西方。
- 當時在美國交易市場上，條狀圖是主流，幾乎沒有人了解來自日本的蠟燭圖。Steve Nison總是說，在西方了解了蠟燭圖之後，再也沒有人想要回到條狀圖的世界。

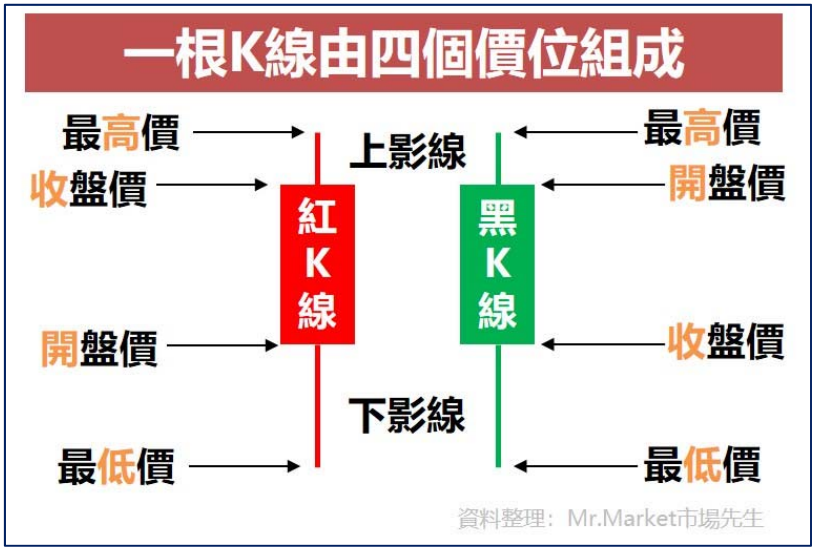
# K線圖

## Mr.Market市場先生

**K線是什麼、有哪些K線種類？K線圖怎麼看？**

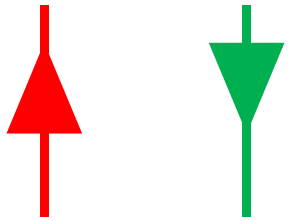
2020.08.31 投資理財入門 投資理財專有名詞 技術分析

原文網址：
https://rich01.com/what-is-k-bar-charts/



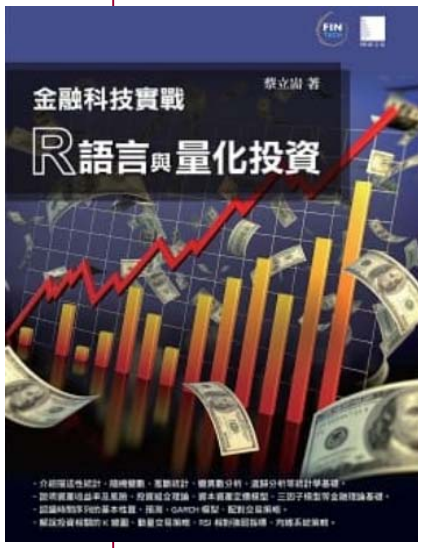

K線的各種形態，如何K線判斷分析？參看
https://rich01.com/what-is-k-bar-charts/

hmwu:
這樣畫不是比較好嗎?
使用黑白顏色也能識別。



# K線圖分析: 上海證券交易所綜合股價指數

```
> # install.packages("quantmod")
> library(quantmod)
> # 定義K線圖佈景主題
> myTheme <- chartTheme("white", up.col="red", dn.col="green")
>
> # 讀入上証綜指2015年日資料
> SSEC2015 <- read.csv("data/SSEC2015.csv", header=TRUE)
> head(SSEC2015, 3)
  X       Date     Open     High      Low    Close Volume
1 1 2015-03-02 3332.721 3336.760 3298.669 3336.285 346400
2 2 2015-03-03 3317.695 3317.695 3260.429 3263.052 382000
3 3 2015-03-04 3264.182 3286.588 3250.484 3279.533 293600
>
> # 將資料框轉成xts時序類別
> SSEC2015 <- xts(SSEC2015[, -c(1,2)], order.by=as.Date(SSEC2015$Date))
> class(SSEC2015)
[1] "xts" "zoo"
> head(SSEC2015, 3)
               Open     High      Low    Close Volume
2015-03-02 3332.721 3336.760 3298.669 3336.285 346400
2015-03-03 3317.695 3317.695 3260.429 3263.052 382000
2015-03-04 3264.182 3286.588 3250.484 3279.533 293600
> tail(SSEC2015, 3)
               Open     High      Low    Close Volume
2015-03-27 3686.134 3710.477 3656.831 3691.096 408900
2015-03-30 3710.612 3795.935 3710.612 3786.568 564700
2015-03-31 3822.987 3835.567 3737.043 3747.899 561700
> # 繪製K線圖(預設值)
> chartSeries(SSEC2015)
> # 繪製K線圖(設定佈景主題，標題名稱)
> chartSeries(SSEC2015, theme=myTheme, name="上證綜指2015年3月份，日K線圖")
```



# 上海證券交易所綜合股價指數(上證指數)

- K線圖是技術分析的基石，有經驗的分析師可從K線形態中抓取出股價變動有用的訊息。
- 紅色蠟燭占主要部份，大多數交易日「收盤>開盤」，處於「牛市」。
- 「紅色蠟燭圖」實體部份較大，收盤開盤價差大，當日市場處於多頭強勢的狀態。
- 十字星、錘子線、吊頂線，有很多變體，但基本形態是一樣的。

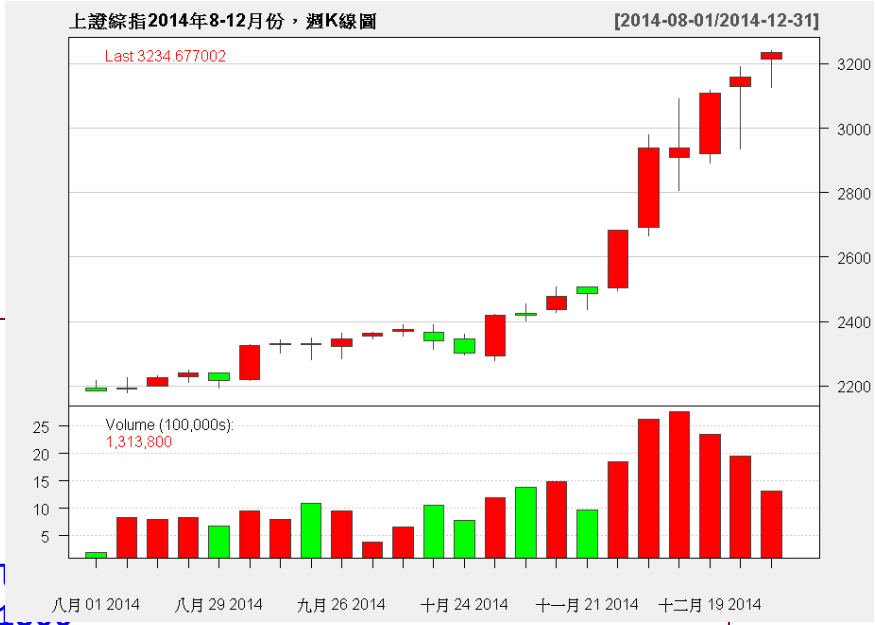
- **十字星**: 實體部份小，上下影線相似。

表示:「收盤，開盤」差別小，當期交易日開市後，買方市場和賣方市場試圖均衡力量。

- **錘子線**: 上影線或下影線比較短或不存在，實體部份也窄。
- **吊頂線**: 一般出現在價格頂部，沒有上影線，或上影線很短，下影線是上影線兩倍以上。

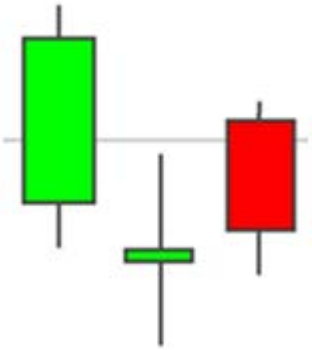
表示:「收盤，最高價」很接近，當期交易日開市後，價格上下波動，快到閉市時才達到最高點。

# 周K線圖

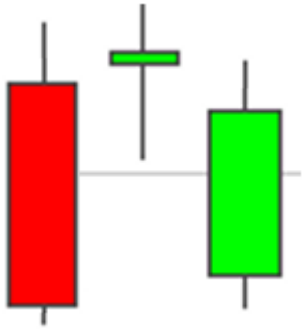



```
> # 讀入上証綜指2014年日資料
> SSEC2014 <- read.csv("data/SSEC2014.csv",
+  header=TRUE)
> SSEC2014 <- xts(SSEC2014[, -c(1,2)],
+  order.by=as.Date(SSEC2014$Date))
> head(SSEC2014, 3)
               Open     High      Low    Close Vol
2014-08-01 2194.170 2218.789 2184.641 2185.303 191
2014-08-04 2190.036 2224.068 2186.917 2223.331 174100
2014-08-05 2224.632 2226.838 2206.850 2219.945 175400
>
> # 將日資料轉成周資料
> SSEC2014W <- to.weekly(SSEC2014)
> head(SSEC2014W)
           SSEC2014.Open SSEC2014.High SSEC2014.Low SSEC2014.Close SSEC2014.Volume
2014-08-01      2194.170      2218.789     2184.641       2185.303          191500
2014-08-08      2190.036      2226.838     2180.597       2194.425          837600
2014-08-15      2199.433      2230.991     2198.314       2226.734          803600
2014-08-22      2229.804      2248.939     2211.630       2240.812          835100
2014-08-29      2241.100      2241.553     2193.263       2217.200          675900
2014-09-05      2220.129      2327.557     2217.685       2326.432          946500
> chartSeries(SSEC2014W, theme=myTheme, name="上證綜指2014年8-12月份，週K線圖")
```

# 「早晨之星」、「黃昏之星」型態



K線圖可分為「持續形態(continuation patterns)」和「反轉形態(reversal patterns)」

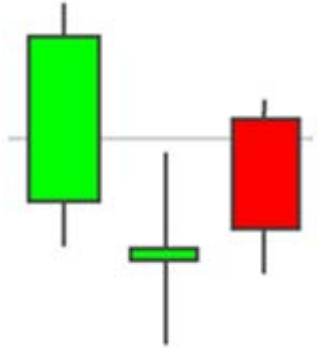


- 「早晨之星」型態: 下跌趨勢的反轉信號。
- 連續3根蠟燭圖組成。
  - 第1根: 較長綠色實體
  - 第2根: 十字星線，實體在第一、二根實體下方。
  - 第3根: 紅色實體(大於等於第1根綠色實體的一半)
- 十字星體要低於前一期的收盤價，和後一期的開盤價。
- 下跌行情中，出現「早晨之星」，可解釋買方市場力量增大，股價有利好的趨勢，股市可能由原來下跌行情轉為上漲行情。

- 「黃昏之星」型態: 上漲趨勢的反轉信號。
- 連續3根蠟燭圖組成。
  - 第1根: 較長紅色實體
  - 第2根: 十字星線，實體在第一、二根實體上方。
  - 第3根: 綠色實體(大於等於第1根紅色實體的一半)
- 十字星體要高於前一期的收盤價，和後一期的開盤價。
- 出現「早晨之星」，預示著股票市場由買方轉變成賣方市場，股價未來有下跌的趨勢，說明股票的上漲行情有可能處於黃昏階段。

注意: 此兩形態只是反轉形態的一個信號，並不能作為反轉形態的唯一依據。

# 捕捉「早晨之星」型態



- 對連續3日的日交易資料進行分析
  - **(a) 描述蠟燭實體:**
    - 第1天收盤價低於開盤價(綠色蠟燭實體)。
    - 第2天收盤價，開盤價大致相等(控制在一範圍內)。
    - 第3天(紅色蠟燭實體)收盤價高於開盤價，差值大於等於第一天(綠色蠟燭實體)開盤價收盤價差值一半。
  - **(b) 定義十字星實體位置:** 第2天的收盤價和開盤價小於第1天的收盤價和第3天的開盤價。
  - **(c) 定義下跌趨勢:** 用收盤價來表示股票的收益率，收益率為負表示下跌。

# R語言捕捉「早晨之星」型態

```
> # 刪除所有物件
> rm(list=ls())
> myTheme <- chartTheme("white", up.col="red", dn.col="green")
>
> # 讀入上證綜指2012年日交易資料
> SSEC2012 <- read.csv("data/SSEC2012.csv", header=TRUE)
> # 轉為xts時序資料類別
> SSEC2012 <- xts(SSEC2012[, -c(1,2)], order.by=as.Date(SSEC2012$Date))
> head(SSEC2012, 3)
               Open     High      Low    Close Volume
2012-01-04 2211.995 2217.520 2168.644 2169.390  49200
2012-01-05 2160.897 2183.404 2145.556 2148.452  58800
2012-01-06 2148.147 2164.322 2132.626 2163.395  50600
> tail(SSEC2012, 3)
               Open     High      Low    Close Volume
2012-12-27 2223.655 2234.724 2204.790 2205.897 129000
2012-12-28 2207.913 2234.869 2204.003 2233.252 116000
2012-12-31 2236.461 2269.512 2236.461 2269.128 128000
>
> # 取收盤價與開盤價
> Close <- SSEC2012$Close
> Open <- SSEC2012$Open
> # 計算每一個交易日的收盤價與開盤價的差值
> CL_OP <- Close - Open
> names(CL_OP) <- "CL_OP"
```

```
> head(CL_OP)
                 CL_OP
2012-01-04 -42.605224
2012-01-05 -12.445069
2012-01-06  15.248047
2012-01-09  61.148926
2012-01-10  63.911865
2012-01-11  -6.867188
>
> # 摘要差值分佈情況
> summary(CL_OP)
     Index                 CL_OP
 Min.   :2012-01-04   Min.   :-73.685
 1st Qu.:2012-04-09   1st Qu.:-11.442
 Median :2012-07-06   Median :  1.513
 Mean   :2012-07-05   Mean   :  2.244
 3rd Qu.:2012-10-03   3rd Qu.: 14.297
 Max.   :2012-12-31   Max.   : 91.844
```

# 捕捉綠色實體、十字星、紅色實體

```
> # 取前期值(lag)
> head(CL_OP)
                CL_OP
2012-01-04 -42.605224
2012-01-05 -12.445069
2012-01-06  15.248047
2012-01-09  61.148926
2012-01-10  63.911865
2012-01-11  -6.867188
> head(lag(CL_OP, 1))
               CL_OP
2012-01-04        NA
2012-01-05 -42.60522
2012-01-06 -12.44507
2012-01-09  15.24805
2012-01-10  61.14893
2012-01-11  63.91187
> head(lag(CL_OP, 2))
               CL_OP
2012-01-04        NA
2012-01-05        NA
2012-01-06 -42.60522
2012-01-09 -12.44507
2012-01-10  15.24805
2012-01-11  61.14893
```

```
> # 結合三期資料: 本期差值、前一期差值、前二期差值
> dataCL_OP <- merge(CL_OP, lag(CL_OP, 1), lag(CL_OP, 2))
> class(dataCL_OP)
[1] "xts" "zoo"
> head(dataCL_OP, 3)
               CL_OP   CL_OP.1   CL_OP.2
2012-01-04 -42.60522        NA        NA
2012-01-05 -12.44507 -42.60522        NA
2012-01-06  15.24805 -12.44507 -42.60522
```

# 捕捉綠色實體、十字星、紅色實體

```
> # 判斷「早晨之星」型態: 0: 不是。1: 是
> # 補捉綠色實體(up:high-open, lower:close-low)、
> # 十字星(open~close)
> # 紅色實體(up:high-close, lower:open-low)
> candle <- apply(dataCL_OP, 1, function(x) {
+   ifelse(x[3] < (-11) &
+          abs(x[2]) < 2 &
+          x[1] > 6 &
+          abs(x[1]) > 0.5 * abs(x[3]),
+          1, 0)
+ })
>
> # 轉換成xts時間序列類別
> candle <- xts(as.numeric(candle), order.by = index(dataCL_OP))
> names(candle) <- "candle"
> head(candle, 3)
           candle
2012-01-04      0
2012-01-05      0
2012-01-06      0
>
> # 定義十字星實體位置所需要的資料
> dataCOP <- merge(Open, lag(Open, 1), lag(Close, 1), lag(Close, 2))
> tail(dataCOP, 3)
               Open   Open.1    Close  Close.1
2012-12-27 2223.655 2211.594 2219.132 2213.611
2012-12-28 2207.913 2223.655 2205.897 2219.132
2012-12-31 2236.461 2207.913 2233.252 2205.897
```

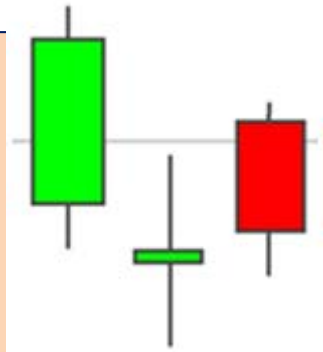
**(a) 描述蠟燭實體:**

第1天收盤價低於開盤價(綠色蠟燭實體)。

第2天收盤價，開盤價大致相等(控制在一範圍內)。

第3天(紅色蠟燭實體)收盤價高於開盤價，差值大於等於第一天(綠色蠟燭實體)開盤價收盤價差值一半。

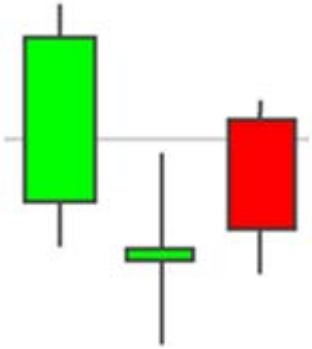
**(b) 定義十字星實體位置:** 第2天的收盤價和開盤價小於第1天的收盤價和第3天的開盤價。




```
> # 摘要差值分佈情況
> summary(CL_OP)
     Index                CL_OP
 Min.   :2012-01-04   Min.   :-73.685
 1st Qu.:2012-04-09   1st Qu.:-11.442
 Median :2012-07-06   Median :  1.513
 Mean   :2012-07-05   Mean   :  2.244
 3rd Qu.:2012-10-03   3rd Qu.: 14.297
 Max.   :2012-12-31   Max.   : 91.844
```

**(b) 定義十字星實體位置:** 第2天的收盤價(`lag(Close, 1)`)和開盤價(`lag(Open, 1)`)小於第1天的收盤價(`lag(Close, 2)`)和第3天的開盤價(`Open`)。

```
> dataCOP <- merge(Open, lag(Open, 1), lag(Close, 1), lag(Close, 2))

> tail(dataCOP, 3)
                Open   Open.1    Close  Close.1
2012-12-27 2223.655 2211.594 2219.132 2213.611
2012-12-28 2207.913 2223.655 2205.897 2219.132
2012-12-31 2236.461 2207.913 2233.252 2205.897
```



```
> # 定義十字星實體位置，要在其前後綠色實體和紅色實體下方
> # 捕捉符合十字星實體位置的K線圖
> Doji <- apply(dataCOP, 1, function(x) {
+   ifelse(x[2] < x[4] &
+          x[2] < x[1] &
+          x[3] < x[4] &
+          x[3] < x[1],
+          1, 0)
+ })
>
> # 轉換成xts時間序列類別
> Doji <- xts(as.numeric(Doji), order.by = index(dataCOP))
> names(Doji) <- "Doji"
> head(Doji, 3)
           Doji
2012-01-04   NA
2012-01-05    0
2012-01-06    0
```

(b) 定義十字星實體位置:

- 第2天的開盤價(`lag(Open, 1)`)小於第1天的收盤價(`lag(Close, 2)`)。
- 第2天的開盤價(`lag(Open, 1)`)小於第3天的開盤價(`Open`)。
- 第2天的收盤價(`lag(Close, 1)`)小於第1天的收盤價(`lag(Close, 2)`)。
- 第2天的收盤價(`lag(Close, 1)`) 小於第3天的開盤價(`Open`)。

# 定義下跌趨勢，計算收益率

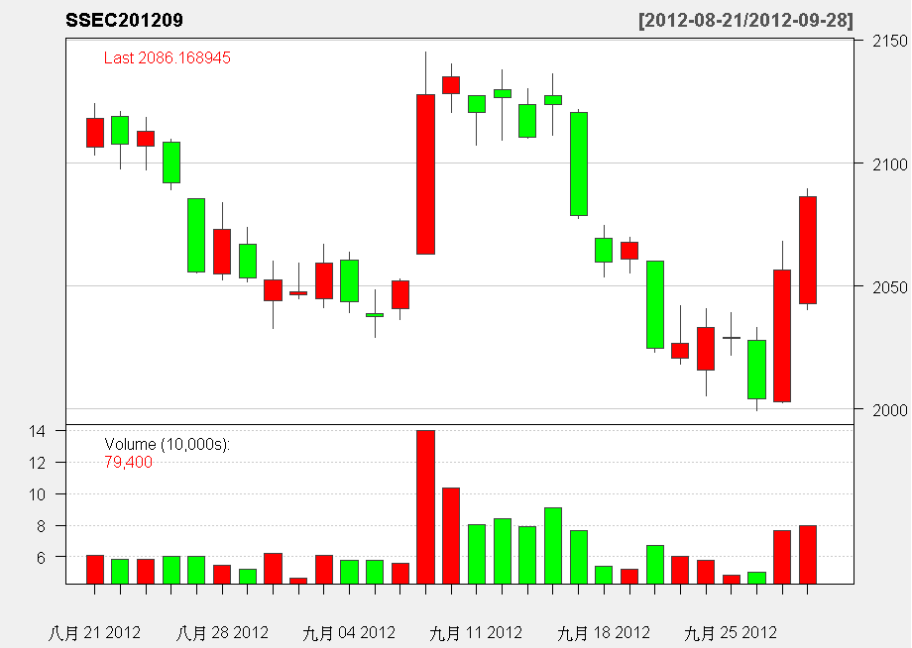
```
> # 定義下跌趨勢
> # 計算收益率
> ret <- ROC(Close, type = "discrete")
> names(ret) <- "SSEC2012.ret"
> head(ret)
           SSEC2012.ret
2012-01-04           NA
2012-01-05 -0.009651556
2012-01-06  0.006955295
2012-01-09  0.028887407
2012-01-10  0.026889921
2012-01-11 -0.004242819
>
>
> # 結合「尋找向下趨勢所需資料」，前一期收益、前二期收益
> dataret <- merge(lag(ret, 2), lag(ret, 1))
>
> # 尋找向下趨勢
> trend <- apply(dataret, 1, function(x) {
+   ifelse(x[1] < 0 &
+          x[2] < 0,
+          1, 0)
+ })
```

$$\text{rate of change} = \frac{P_t - P_{t-1}}{P_{t-1}}$$

```
> trend <- xts(as.numeric(trend), order.by = index(dataret))
> names(trend) <- "trend"
> head(trend)
           trend
2012-01-04    NA
2012-01-05    NA
2012-01-06    NA
2012-01-09     0
2012-01-10     0
2012-01-11     0
```

# 定義「早晨之星」捕捉R函數

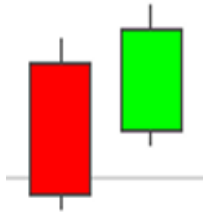
```
> # 定義「早晨之星」捕捉函數
> MorningStar <- function(candle, Doji, trend) {
+   star <- na.omit(merge(candle, Doji, trend))
+
+   signal <- apply(star, 1, function(x) {
+     ifelse(all(x == 1), 1, 0)
+   })
+
+   signal <- xts(as.numeric(signal), order.by = index(star))
+   return(index(signal[signal == 1]))
+ }
>
> # 捕捉上証綜指2012年出現「早晨之星」的日期
> MorningStar(candle, Doji, trend)
[1] "2012-09-06"
>
> # 繪製2012/09/06附近的K線圖
> SSEC201209 <- SSEC2012["2012-08-21/2012-09-30"]
> chartSeries(SSEC201209, theme = myTheme)
```




先處於短期下跌趨勢，又經過了短期的盤整，在9月6日附近出現早晨之星形態，釋出市場反轉的信號，接著幾個交易日，股價處於上漲趨勢。

# R語言捕捉「烏雲蓋頂」型態

```
> rm(list=ls())
> myTheme <- chartTheme("white", up.col="red", dn.col="green")
>
> # 讀入上證綜指2011年日交易資料
> SSEC2011 <- read.csv("data/SSEC2011.csv", header = TRUE)
> SSEC2011 <- xts(SSEC2011[, -c(1, 2)], order.by = as.Date(SSEC2011$Date))
>
> Close <- SSEC2011$Close
> Open <- SSEC2011$Open
> CL_OP <- Close - Open
> names(CL_OP) <- "CL_OP"
> dataCL_OP <- na.omit(merge(CL_OP, lag(CL_OP)))
> head(dataCL_OP, 3)
               CL_OP   CL_OP.1
2011-01-05   6.10498  27.31885
2011-01-06 -14.23804   6.10498
2011-01-07  18.13599 -14.23804
> # 捕捉連續2期的不同顏色的蠟燭實體
> cloudColor <- apply(dataCL_OP, 1, function(x) {
+   ifelse(x[2] > 0 & x[1] < 0, 1, 0)
+ })
>
> cloudColor <- xts(as.numeric(cloudColor), order.by = index(dataCL_OP))
> head(cloudColor, 3)
           [,1]
2011-01-05    0
2011-01-06    1
2011-01-07    0
```

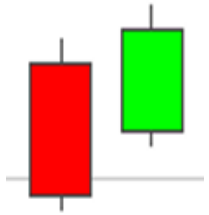


**烏雲蓋頂:**

- 2個不同顏色的蠟燭組成。
- 第一天是上升趨勢的陽線，紅色蠟燭`x[2] > 0`實體較長，表示收盤價>>開盤價。
- 第二天綠色蠟燭實體`x[1] < 0`也較長，開盤價(第二天實體上端)高於期一前的收盤價(第一天實體上端)。
- 第二天收盤價低於第一天收盤價和開盤價之和的一半，且大於第一天的開盤價。
- 第二天開市價格較高，表示多頭市場依舊強勢，閉市時，價格大幅下跌，可能出現利空消息。
- 定義上升趨勢: 連續2期收益為正。

# 描述「烏雲蓋頂」形態收盤價、開盤價的位置條件

```
> # 描述「烏雲蓋頂」形態收盤價、開盤價的位置條件
> dataCOP <- merge(Close, Open, lag(Close), lag(Open))
> cloud <- apply(dataCOP, 1, function(x) {
+   ifelse(x[2] > x[3] &
+          x[1] < 0.5 * (x[3] + x[4]) &
+          x[1] > x[4],
+          1, 0)
+ })
> cloud <- xts(as.numeric(cloud),
+   order.by = index(dataCOP))
> head(cloud, 3)
           [,1]
2011-01-04   NA
2011-01-05    0
2011-01-06    0
> # 定義向上趨勢
> # 計算收益率
> ret <- ROC(Close, type = "discrete")
> names(ret) <- "SSEC2011.ret"
> head(ret, 3)
           SSEC2011.ret
2011-01-04           NA
2011-01-05 -0.004706856
2011-01-06 -0.005291566
```



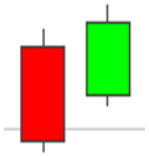
**烏雲蓋頂:**

- 2個不同顏色的蠟燭組成。
- 第一天是上升趨勢的陽線，紅色蠟燭實體較長，表示收盤價>>開盤價。
- 第二天綠色蠟燭實體也較長，第二天開盤價高於第一天的收盤價。`x[2] > x[3]`
- 第二天收盤價低於第一天收盤價和開盤價之和的一半 `x[1] < 0.5 * (x[3] + x[4])`，且大於第一天的開盤價`x[1] > x[4]`。
- 第二天開市價格較高，表示多頭市場依舊強勢，閉市時，價格大幅下跌，可能出現利空消息。
- 定義上升趨勢: 連續2期收益為正。

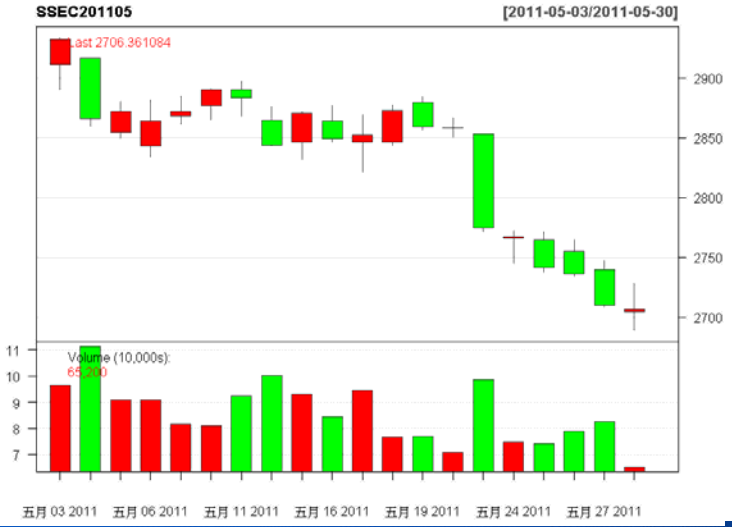
```
> dataret <- merge(lag(ret, 2), lag(ret, 1))
> Uptrend <- apply(dataret, 1, function(x) {
+   ifelse(x[1] > 0 & x[2] > 0, 1, 0)
+ })
> Uptrend <- xts(as.numeric(Uptrend), order.by = index(dataret))
> head(Uptrend, 3)
           [,1]
2011-01-04   NA
2011-01-05   NA
2011-01-06    0
```

# 尋找「烏雲蓋頂」形態

```
> # 尋找「烏雲蓋頂」形態
> darkCloud <- cloudColor + Uptrend + cloud
> names(darkCloud) <- "darkCloud"
> darkCloud[darkCloud == 3]
           darkCloud
2011-05-19         3
2011-08-16         3
>
> # 繪製2011/05/19附近的K線圖
> SSEC201105 <- SSEC2011["2011-05-01/2011-05-30"]
> chartSeries(SSEC201105, theme = myTheme)
>
> # 繪製2011/08/16附近的K線圖
> SSEC201108 <- SSEC2011["2011-08-01/2011-08-30"]
> chartSeries(SSEC201108, theme = myTheme)
```



- 5/8開始，處於上升趨勢，連續幾個交易日價格變化不大。
- 5/19出現「烏雲蓋頂」形態，預示市場中空頭力量的強勢，市場可能要處於下跌的行情。
- 5/20以後的價格走勢，可看出價格一直下跌。
- 「烏雲蓋頂」形態釋放出反轉行情的信號。
- 結論: 市場反轉的具體判斷還需要參照前期的市場走勢以及這些形態所處的位置等綜合分析。



# 動量交易策略

- 動量交易策略 (Momentum Trading Strategy): 「証券的價格」類比為「運動中的物體」。
- 動量=物體的質量 x 速度。質量 和 速度描述了物體運動狀態，同時說明了運動狀態的趨勢，即慣性的大小。
- 証券價格上漲下跌可視為物體運動。
- 証券價格上漲，則具有繼續上漲的動能。証券價格下跌，則可能有繼續下跌的動量。
- 可透由研究証券價格的動量來分析証券價格的變化趨勢，進而制定交易策略，獲取益。

# 價格動量的計算公式

- **作差法**求動量值:

$$Momemtum_t = P_t - P_{t-m}$$

  - $P_t$: 股票$t$時期的價格。
  - $P_{t-m}$: 股票$t-m$時期的價格せ，$m$為時間間隔。

- **作除法**求動量值(價格變化率):

$$Momemtum_t = ROC_t = \frac{P_t - P_{t-m}}{P_{t-m}}$$

- 短期動量: 10天、20天、25天。
- 中期動量: 6個月、9個月。
- 長期動量: 一年或更久。

# 動量交易策略: 萬科(Vanke)集團股票

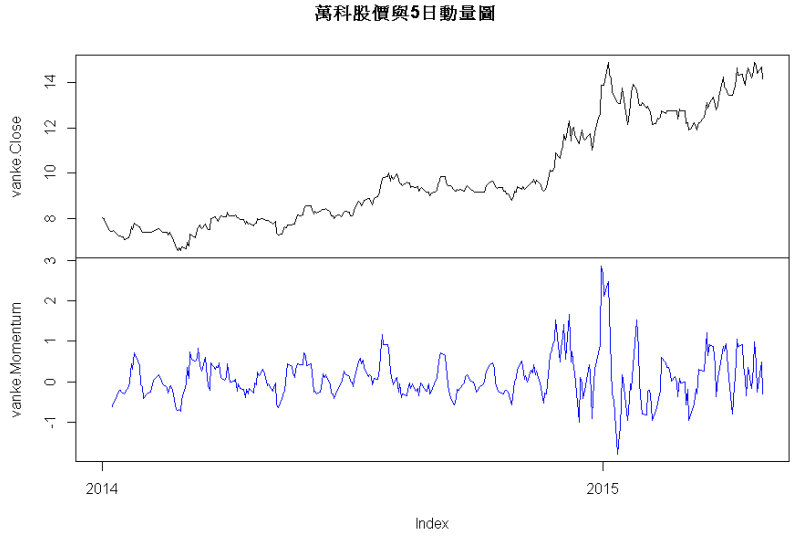
```
> ######################################
> # 價格動量計算公式                    #
> ######################################
>
> rm(list=ls())
> myTheme <- chartTheme("white", up.col="red", dn.col="green")
>
> # 讀取萬科股票2014~20150428，日資料
> Vanke <- read.csv("data/Vanke.csv", header = TRUE)
> Vanke <- xts(Vanke[, -c(1, 2)], order.by = as.Date(Vanke$Date))
> head(Vanke, 3)
           Open High  Low Close   Volume
2014-01-01 8.03 8.03 8.03  8.03        0
2014-01-02 7.99 8.07 7.92  7.99 48529900
2014-01-03 7.97 7.97 7.76  7.84 54046700
> tail(Vanke, 3)
            Open  High   Low Close    Volume
2015-04-24 14.48 14.70 14.15 14.42 362066200
2015-04-27 14.41 14.70 14.30 14.70 422562000
2015-04-28 14.65 14.66 14.11 14.19 386603900
```

```
> Close <- Vanke$Close
> names(Close) <- "vanke.Close"
> tail(Close, 3)
           vanke.Close
2015-04-24       14.42
2015-04-27       14.70
2015-04-28       14.19
> summary(Close)
     Index              vanke.Close
 Min.   :2014-01-01   Min.   : 6.570
 1st Qu.:2014-05-01   1st Qu.: 7.987
 Median :2014-08-30   Median : 9.250
 Mean   :2014-08-30   Mean   : 9.846
 3rd Qu.:2014-12-29   3rd Qu.:11.967
 Max.   :2015-04-28   Max.   :14.910
```

# 繪製收盤價曲線和5日動量曲線

```
>
> # 計算滯後5期(前5期)收盤價
> lagClose <- lag(Close, 5)
> names(lagClose) <- "lagClose"
>
> vankeClose <- merge(Close, lagClose)
> vankeClose <- na.omit(vankeClose)
> head(vankeClose)
           vanke.Close lagClose
2014-01-08        7.42     8.03
2014-01-09        7.46     7.99
2014-01-10        7.38     7.84
2014-01-13        7.24     7.48
2014-01-14        7.23     7.43
2014-01-15        7.17     7.42
>
> # 計算5日(作差法)動量
> momentum5 <- vankeClose$vanke.Close -
vankeClose$lagClose
> names(momentum5) <- "vanke.Momentum"
> tail(momentum5, 3)
           vanke.Momentum
2015-04-24          -0.25
2015-04-27           0.49
2015-04-28          -0.29
```

```
> # 繪製收盤價曲線和5日動量曲線
> plot.zoo(merge(Close, momentum5),
+          col = c("black", "blue"),
+          main = "萬科股價與5日動量圖")
>
> # 計算5日(作除法)動量
> Momentum <- (Close - lagClose) / lagClose
> Momentum <- Momentum[-5:-1, ]
> names(Momentum) <- "vanke.roc5"
> head(Momentum)
            vanke.roc5
2014-01-08 -0.07596513
...
2014-01-15 -0.03369272
```



# 動量相關函數

```
> library(TTR)
>
> # 以 momentum {TTR} 計算5日動量(作差法)
> # na.pad = TRUE: 資料包含NA的日期
> Momentum1 <- momentum(Close, n = 5, na.pad = TRUE)
> head(Momentum1)
           vanke.Close
2014-01-01          NA
2014-01-02          NA
2014-01-03          NA
2014-01-06          NA
2014-01-07          NA
2014-01-08       -0.61
> tail(Momentum1)
           vanke.Close
2015-04-21        0.36
2015-04-22        0.98
2015-04-23        0.46
2015-04-24       -0.25
2015-04-27        0.49
2015-04-28       -0.29
```

- 連續變化率 (type = "continuous")

$$ROCCon_t = \ln \frac{P_t}{P_{t-1}}$$

- 離散變化率 (type = "discrete")

$$ROCDis_t = \frac{P_t - P_{t-1}}{P_{t-1}}$$

```
> # 計算股價的5期收益率(離散變化率)
> ROCDis <- ROC(Close, n = 5,
+   type = "discrete", na.pad = TRUE)
> ROCDis <- na.omit(ROCDis)
> names(ROCDis) <- "vanke.ROCDis"
> head(ROCDis)
           vanke.ROCDis
2014-01-08  -0.07596513
2014-01-09  -0.06633292
2014-01-10  -0.05867347
2014-01-13  -0.03208556
2014-01-14  -0.02691790
2014-01-15  -0.03369272
```

# 股價年走勢及動量線(中期動量: 35日)

```
> # 2015年萬科股價走勢K線圖
> Vanke2015 <- Vanke["2015"]
> chartSeries(Vanke2015, theme = myTheme, name = "萬科2015年K線圖")
>
> # 在K線圖上，繪製收盤價線(technical analysis)
> addTA(Cl(Vanke2015), on = 1, col = "black", type = "l")
> # 在K線圖下面位置，繪製收盤價線
> addTA(Cl(Vanke2015), col = "black", type = "l")
> # 在K線圖下面位置，繪製35日動量線
> addTA(momentum(Cl(Vanke2015), n = 35, na.pad = TRUE), col = 4, type = "l")
```

- 動量線在0值下方時，萬科股票價格大致處於下跌趨勢。
- 動量線在0值以上時，股價走勢整體處於上升趨勢。
- 動量線的走勢和蠟燭的價格走勢方向大致類似，
- 但動量線變化的趨勢可能會比蠟燭的走勢提前。

# 動量交易策略的一般思路

```
> # 提取萬科股票收盤價資料
> Close <- Vanke$Close
> names(Close) <- "vanke.Close"
>
> # 計算35日動量值
> Momen35 <- momentum(Close, n = 35,
+  na.pad = FALSE)
> names(Momen35) <- "momentum35"
> head(Momen35, 5)
           momentum35
2014-02-20      -0.71
2014-02-21      -0.83
2014-02-24      -1.15
2014-02-25      -0.91
2014-02-26      -0.75
>
>
> # 35日動量值 < 0 : signal = -1, 表示賣出
> # 35日動量值 > 0 : signal = 1, 表示買入
> signal <- ifelse(Momen35 < 0, -1, 1)
> names(signal) <- "signal"
> head(signal, 3)
           signal
2014-02-20     -1
2014-02-21     -1
2014-02-24     -1
```

**動量交易策略的一般思路**

- 獲取股票價格(一般為收盤價)資料。
- 確定時間跨度和動量表達式，計算股票的動量值。
- 根據動量指標制定交易策略。

- 直覺:
  - 動量>0，股價可能還具備上漲的能量，釋出買入的信號。
  - 走勢整體處於上升趨勢。
  - 動量<0，股價可能有下跌的能量，釋出賣出的信號。

**(模擬)萬科股票動量交易策略:**

- 將時間跨度設定為35日。
- 當35日動量為正值時，市場可能還存在上升的能量，推斷第2期為買入點。
- 當35日動量為負值時，預期市場未來價格會下跌，第2期可能是賣出入點。

# 根據買賣點制定買入賣出交易

```
> # 計算萬科股票單期收益率
> ret <- ROC(Close, 1)
> names(ret) <- "vanke.ret"
> ret <- ret[-(1:35)]
> head(ret)
              vanke.ret
2014-02-20 -0.006807378
2014-02-21 -0.022100347
2014-02-24 -0.067896107
2014-02-25 -0.018100042
2014-02-26  0.016604155
2014-02-27 -0.013564639
>
> # 35日動量指標交易收益率計算
> Mom35Ret <- ret[-1] * lag(signal, 1, na.pad = FALSE)
> names(Mom35Ret) <- "Mom35Ret"
> head(Mom35Ret)
              Mom35Ret
2014-02-21  0.02210035
2014-02-24  0.06789611
2014-02-25  0.01810004
2014-02-26 -0.01660416
2014-02-27  0.01356464
2014-02-28 -0.01953481
```

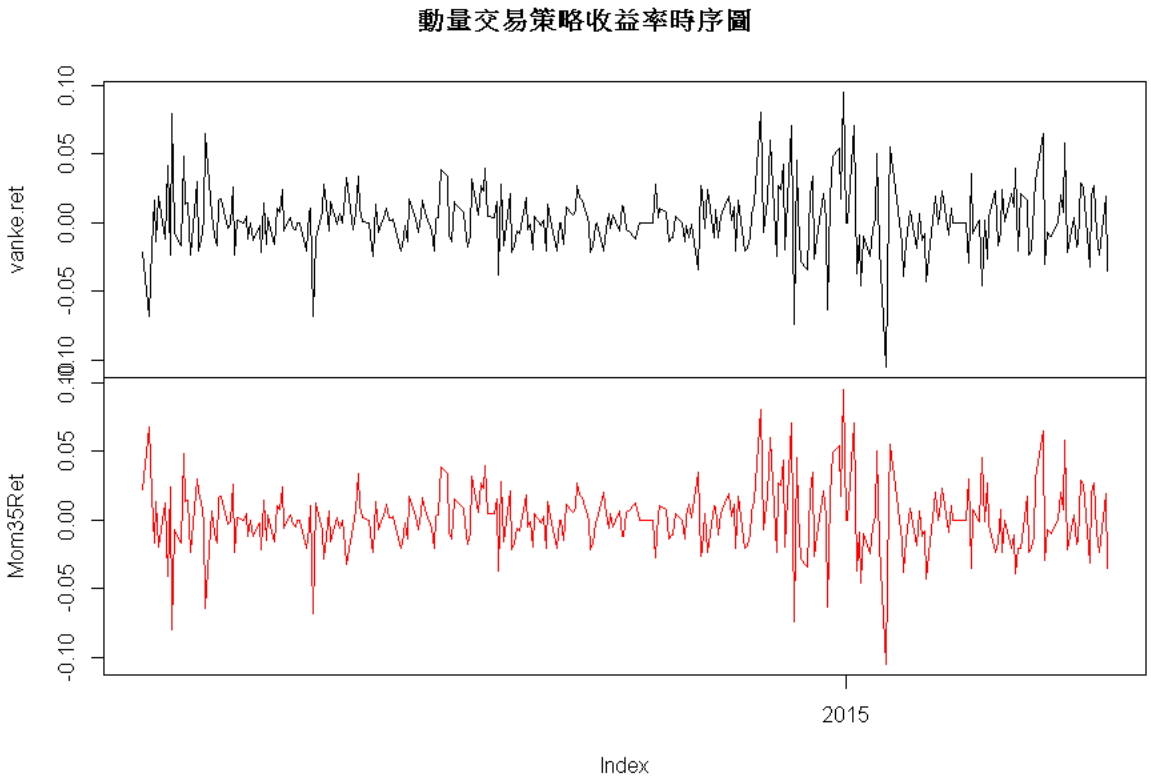
## 計算35日動量指標買賣點的準確率

- 當出現買入信號時，signal取值為1，預計價格上漲，收益率大於0，則signal與ret的乘積大於0。
- 當出現賣出信號時，signal取值為-1，預計價格下跌，收益率小於0，則signal與ret的乘積也大於0。
- 當35日動量值交易的收益大於0時，則說明買賣點預測正確。

```
> # 計算指標交易獲勝率
> win <- Mom35Ret[Mom35Ret >= 0]
> winrate <- length(win) / length(Mom35Ret)
> winrate
[1] 0.5616883
```

# 動量交易策略收益率時序圖

```
> # 繪製動量交易策略收益率時序圖
> ret <- ret[-1]
> plot.zoo(merge(ret, Mom35Ret), col = c("black", "red"),
+          main = "動量交易策略收益率時序圖")
> legend("topright", legend = c("萬科股票", "動量交易"),
+   col = c("black", "red"), lty = 1)
```



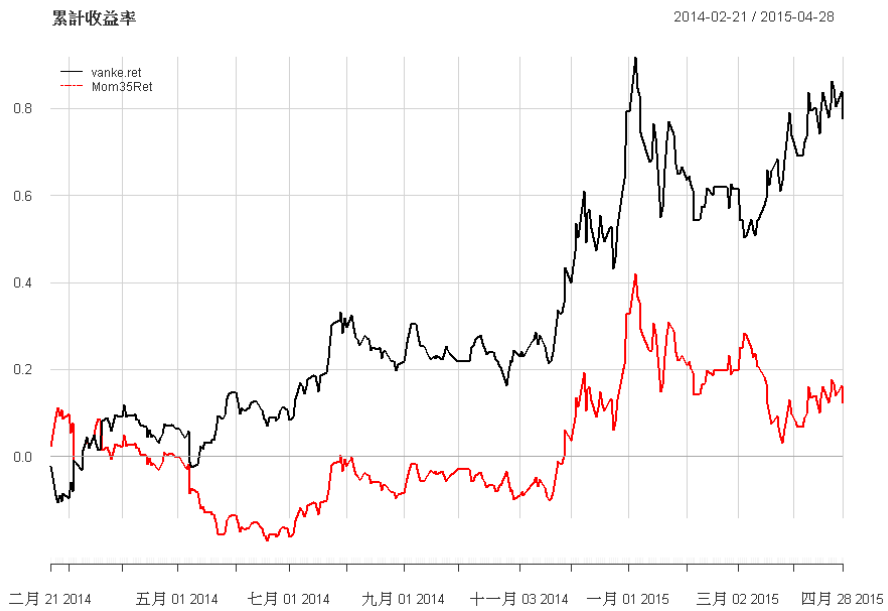
# 累積收益率時序圖

```
> # 計算交易策略的累積收益率
> library(PerformanceAnalytics)
> chart.CumReturns(merge(ret, Mom35Ret), col = c("black", "red"),
+                  lty = c(1, 6), main = "累計收益率", legend.loc = "topleft")
> legend("topleft", legend = c("萬科股票", "動量交易"),
+   col = c("black", "red"), lty = c(1, 6))
```

- $t$期累計收益率(Cumulative Return)

$$cr_t = (1 + r_0)(1 + r_1) \ldots (1 + r_t) - 1$$

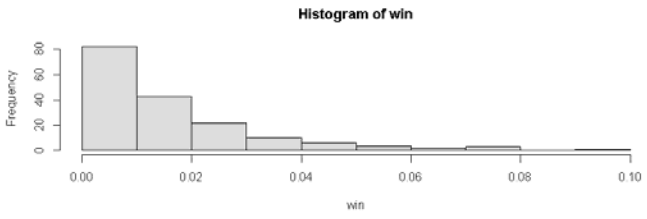
- 此策略獲勝率為0.5616883，而動量交易的累積收益率的後表現不如萬科股票自身累積收益率的表現。
- 可能原因: 預測正確賺取的收益率可能沒有預測失敗損失的收益率大。
- 例如: 預測正確盈利2個百分點，而預測失敗時卻損失4個百分點。
- 綜合起來累計的收益率表現可能會不好。

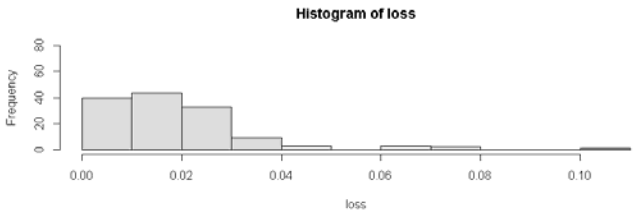


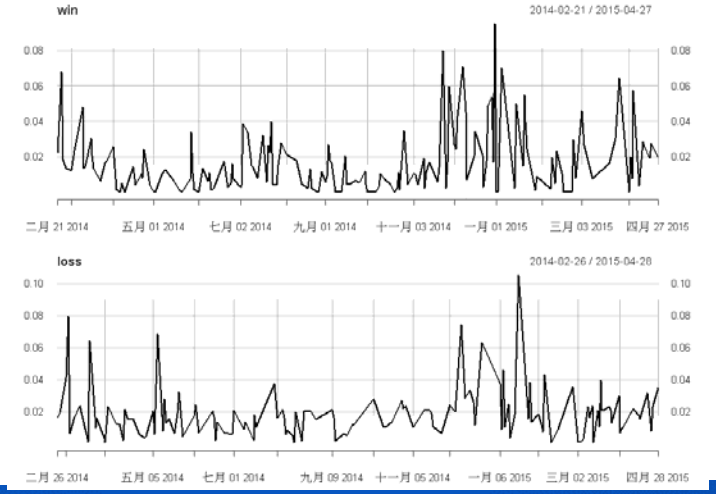
# 策略正確時，盈利的收益率
# 策略失敗時，損失的收益率

```
> # 策略正確時，盈利的收益率
> win <- Mom35Ret[Mom35Ret >= 0]
> # 策略失敗時，損失的收益率
> loss <- Mom35Ret[Mom35Ret < 0]
> loss <- (-loss)
> par(mfrow = c(2, 1))
> hist(win)
> hist(loss, ylim = c(0, 80))
> plot(win)
> plot(loss)
>
> # 平均盈利收益率
> mean(win)
[1] 0.01578579
> # 平均損失盈利收益率
> mean(loss)
[1] 0.01872078
```

```
> summary(win)
     Index               Mom35Ret
 Min.   :2014-02-21   Min.   :0.000000
 1st Qu.:2014-06-13   1st Qu.:0.003486
 Median :2014-09-26   Median :0.010595
 Mean   :2014-09-21   Mean   :0.015786
 3rd Qu.:2014-12-26   3rd Qu.:0.021331
 Max.   :2015-04-27   Max.   :0.095022
> summary(loss)
     Index               Mom35Ret
 Min.   :2014-02-26   Min.   :0.0008055
 1st Qu.:2014-05-30   1st Qu.:0.0083347
 Median :2014-09-23   Median :0.0162936
 Mean   :2014-09-28   Mean   :0.0187208
 3rd Qu.:2015-01-27   3rd Qu.:0.0226401
 Max.   :2015-04-28   Max.   :0.1051962
```




- 動量指標預測錯誤時損失的收益率整體比預測準確時盈利的收益率大。
- 儘管萬科股票35日動量指標交易策略預測準確率高於0.5，但從2014/02/26 ~ 2015/04/28的累計收益率表現則沒有萬科股票原來的累計收益率高。

# RSI 相對強弱指標

- 股票市場: 買方與賣方力量的消長會影響股價。
  - 股票買入力量 > 股票賣出力量 => 價格上漲
  - 股票賣出力量 > 股票買入力量 => 價格下跌
- Wells Wilder (1978) 提出衡量証券自身內在相對強度的指標: Relative Strength Index (RSI)(相對強弱指標)。透過RSI來判斷股票的買入和賣出情況，進而預測未來股票的價格走勢。

$$RSI_t = 100 \times \frac{UP_t}{UP_t + DOWN_t}$$

$t$: 持續天數。
$UP_t$: $t$天內股價上漲幅度的平均值。
$DOWN_t$: $t$天內股價下跌幅度的平均值。

| 日期 | 收盤價 | UP | DOWN |
|---|---|---|---|
| 1日 | 18 | - | - |
| 2日 | 23 | 5 | - |
| 3日 | 21 | - | 2 |
| 4日 | 20 | - | 1 |
| 5日 | 19 | - | 1 |

***EX:*** 利用5日的股票收盤價，計算4日RSI值。

$$UP = \frac{5}{4} = 1.25$$

$$DOWN = \frac{2+1+1}{4} = 1$$

$$RSI = 100 \times \frac{1.25}{1+1.25} = 55.56$$

# RSI 相對強弱指標

$$RSI_t = 100 \times \frac{UP_t}{UP_t + DOWN_t}$$

- 一般使用算術平均數求UP首DOWN。
- 均數可分簡單移動平均(SMA)、加權移動平均(WMA)、指數移動平均(EMA)。
- $RSI \approx 0$，可得$UP \ll DOWN$，上漲幅度遠小於下跌幅度，在跨度時間為t時，股票價格下跌的力道遠大於上漲力道。
- $RSI \approx 100$，可得$UP \gg DOWN$，上漲幅度遠大於下跌幅度，在跨度時間為t時，股票價格上漲的力道遠大於下跌力道。
- $RSI \approx 50$，可得$UP \approx DOWN$，上漲股票價格上漲力道大約等於下跌力道。

# RSI 相對強弱指標

| 日期 | 收盤價 | UP | DOWN |
|---|---|---|---|
| 1日 | 3.82 | NA | NA |
| 2日 | 3.79 | - | 0.03 |
| 3日 | 3.75 | - | 0.04 |
| 4日 | 3.77 | 0.02 | - |
| 5日 | 3.80 | 0.03 | - |
| 6日 | 3.79 | - | 0.01 |

| 日期 | RSI |
|---|---|
| 1日 | NA |
| 2日 | - |
| 3日 | - |
| 4日 | - |
| 5日 | 41.6 |
| 6日 | 50.0 |

```
> # t=4, 簡單移動平均(SMA)
> up <- (0.02+0.03)/4
> down <- (0.03+0.04)/4
> 100*up/(up+down)
[1] 41.66667
>
>
> up <- (0.02+0.03)/4
> down <- (0.04+0.01)/4
> 100*up/(up+down)
[1] 50
```

| 日期 | 收盤價 | UP | DOWN |
|---|---|---|---|
| 1日 | 3.82 | NA | NA |
| 2日 | 3.79 | - | 0.03 |
| 3日 | 3.75 | - | 0.04 |
| 4日 | 3.77 | 0.02 | - |
| 5日 | 3.80 | 0.03 | - |
| 6日 | 3.79 | - | 0.01 |

| 日期 | RSI |
|---|---|
| 1日 | NA |
| 2日 | - |
| 3日 | - |
| 4日 | - |
| 5日 | 41.6 |
| 6日 | 50.0 |

# RSI {TTR}: 相對強弱指標

```
> ########################################
> # 計算 RSI值                            #
> ########################################
> library(quantmod)
> # 讀取交通銀行股票交易資料
> BOCMstock <- read.csv("data/BOCM.csv", header = TRUE)
>
> # 轉成時間序列類別
> BOCMstock <- xts(BOCMstock[, -c(1, 2)],
+                  order.by = as.Date(BOCMstock$Date))
> head(BOCMstock)
           Open High  Low Close   Volume
2014-01-02 3.82 3.84 3.80  3.82 57317900
...
2014-01-09 3.78 3.83 3.77  3.79 39753200
> tail(BOCMstock)
           Open High  Low Close    Volume
2015-04-07 6.46 6.56 6.44  6.55 664765600
...
2015-04-14 6.92 6.92 6.80  6.84 480660300
>
> # 取收盤價資料
> BOCMclp <- BOCMstock[, 4]
> names(BOCMclp) <- "BOCMclp"
> head(BOCMclp, 3)
           BOCMclp
2014-01-02    3.82
2014-01-03    3.79
2014-01-06    3.75
```

- Wells Wilder指出，透過運用月周期28日的一半來計算RSI值進行預測是有效的。
- RSI {TTR} 時間跨度的預設值為14日。

RSI {TTR} R Documentation

Relative Strength Index

**Description**

The Relative Strength Index (RSI) calculates a ratio of the recent upward price movements to the absolute price movement. Developed by J. Welles Wilder.

**Usage**

```
RSI(price, n = 14, maType, ...)
```

**Arguments**

`price`
: Price series that is coercible to xts or matrix.

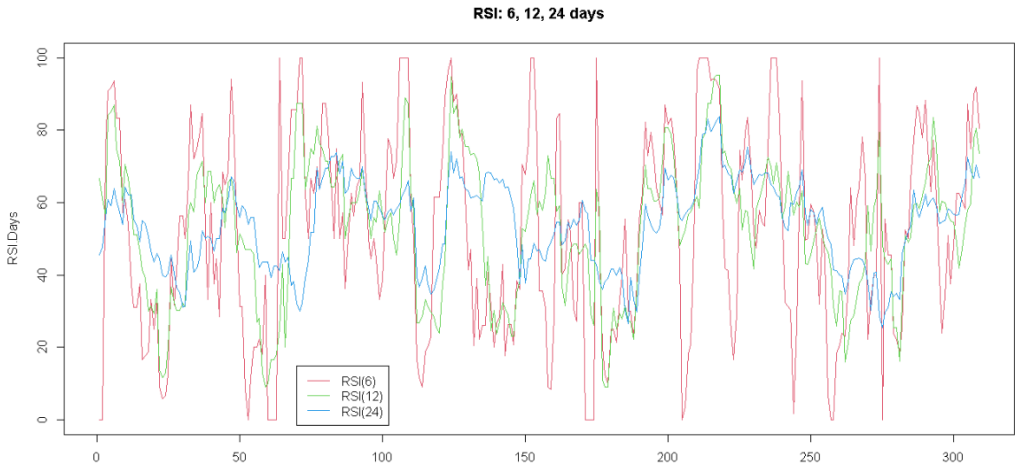
`n`
: Number of periods for moving averages.

`maType`
: Either:

1. A function or a string naming the function to be called.
2. A *list* with the first component like (1) above, and additional parameters specified as *named* components. See Examples.

# 計算 RSI值，RSI天數的差異

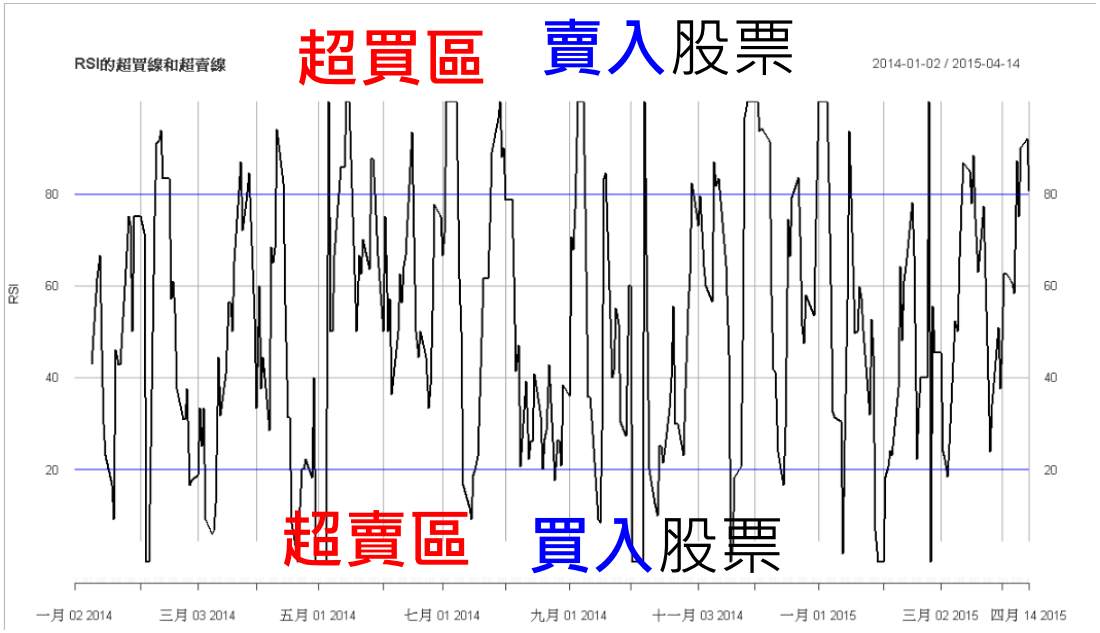
```
> # 計算RSI， maType="SMA", n=6, 12, 24
> RSI6 <- RSI(BOCMclp, n = 6, maType = "SMA")
> names(RSI6) <- "RSI.6"
> RSI12 <- RSI(BOCMclp, n = 12, maType = "SMA")
> names(RSI12) <- "RSI.12"
> RSI24 <- RSI(BOCMclp, n = 24, maType = "SMA")
> names(RSI24) <- "RSI.24"
>
> RSI.Days <- merge(RSI6, RSI12, RSI24)
> RSI.Days <- na.omit(RSI.Days)
> head(RSI.Days)
              RSI.6   RSI.12   RSI.24
2014-02-06  0.00000 66.66667 45.45455
...
2014-02-13 93.75000 86.95652 63.82979
> tail(RSI.Days)
              RSI.6   RSI.12   RSI.24
2015-04-07 58.33333 54.02299 65.48673
...
2015-04-14 80.55556 73.63636 66.97674
>
> matplot(RSI.Days, type="l", col=2:4, lty=1, main="RSI: 6, 12, 24 days")
> legend(70, 15, legend=c("RSI(6)", "RSI(12)", "RSI(24)"), col=2:4, lty=1)
```




- 常用的有: 6日 (RSI1): 一周的時間跨度。(短期)。12日(RSI2): 半個月的時間跨度。(短期)。24日(RSI3): 一個月的時間跨度。(長期)
- 投資分析中，投資者可以參考短期RSI和長期RSI進行投資決策。
  - 短期RSI: 對價格變化情況反應比較靈敏，RSI值波動大。
  - 長期RSI: 對價格變化反應相對遲鈍，RSI值波動小。

# RSI指標判斷股票超買和超賣狀態

```
> ########################################
> # RSI的超買線和超賣線                    #
> ########################################
> plot(RSI6, type = "l",
+      main = "RSI的超買線和超賣線", ylab = "RSI")
>
> RSI6$horizontal.line.20 <- 20
> RSI6$horizontal.line.80 <- 80
> lines(RSI6[, "horizontal.line.20"], col = "blue")
> lines(RSI6[, "horizontal.line.80"], col = "blue")
```




- RSI指標判斷股票
- **常用超買線**: RSI = 80。(RSI = 70，90 皆有)。
- **RSI > 80: 股票出現超買信號。** 股票買入力道過大，買入力道未來會冷卻，所以股票未來價格可能會下跌，此時**賣出**股票，未來下跌後再買入股票，從而賺取價差。
- **常用超賣線**: RSI = 20。(RSI = 30，10 皆有)。
- **RSI < 20: 股票出現超賣信號。** 股票賣出力道過大，賣出入力道未來會回歸正常，所以股票未來價格可能會上漲，此時可**買入**股票，未來上漲後再賣出股票。
- **中心線**: RSI = 50

# RSI的黃金交叉與死亡交叉

**RSI的黃金交叉:**

- 當短期RSI線向上穿過長期的RSI線時，股票近期買入的力道較強，價格上漲的力道很大，釋出一個較強的買入信號，此信號稱為「黃金交叉」。

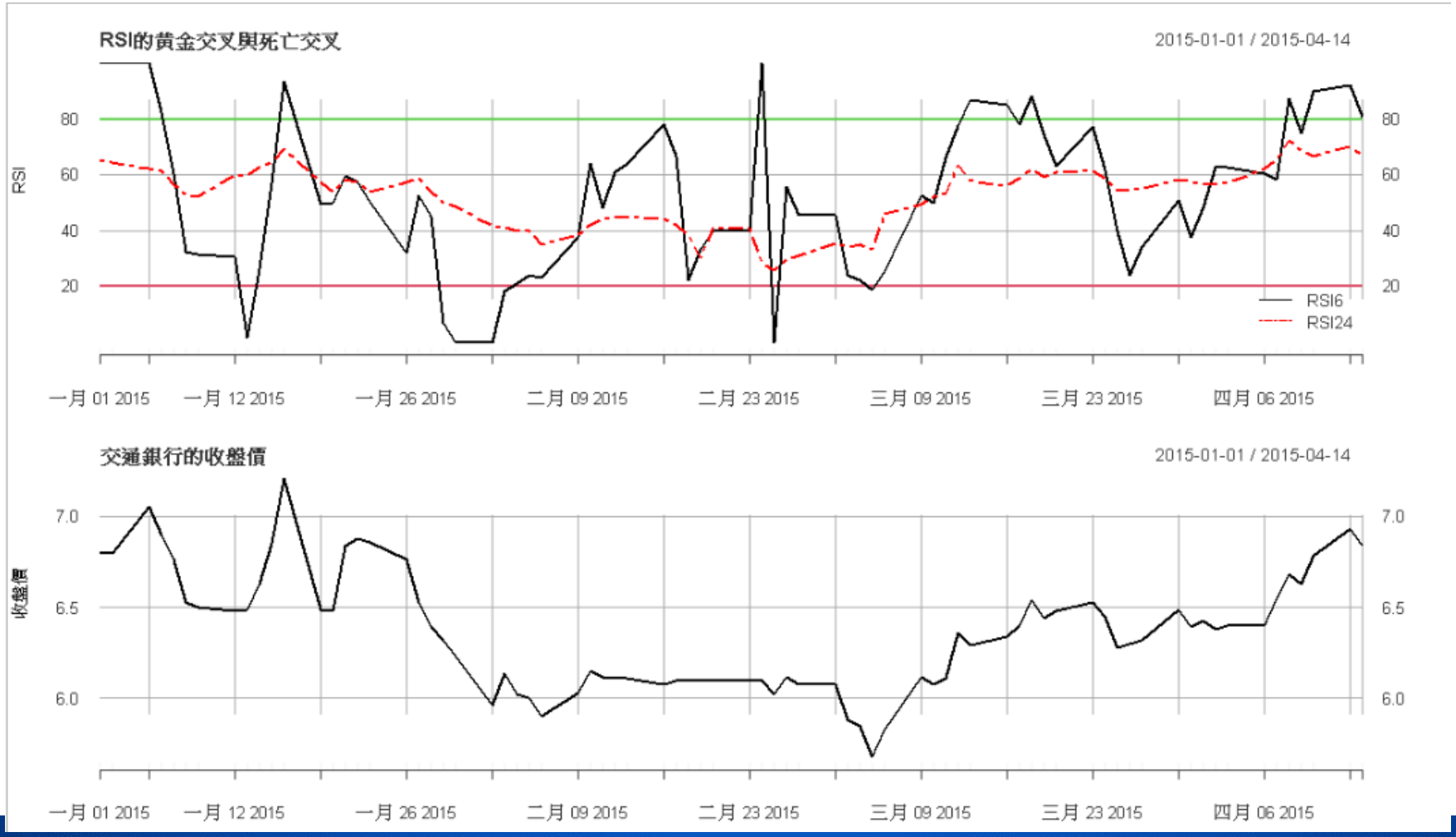
**RSI的死亡交叉:**

- 當短期RSI線向下跌破長期的RSI線時，股票近期賣出的力道較強，價格下跌的力道很大，釋出一個較強的賣出信號，此信號稱為「死亡交叉」。

```
> ########################################
> # RSI的黃金交叉與死亡交叉              #
> ########################################
> par(mfrow=c(2, 1))
> plot(RSI6["2015/"], type = "l",
+      main = "RSI的黃金交叉與死亡交叉", ylab = "RSI")
>
> lines(RSI24["2015/"], col = "red", lty = 6, lwd = 2)
>
> addLegend("bottomright", legend.names = c("RSI6", "RSI24"),
+          col = c("black", "red"), lty = c(1, 6))
> plot(BOCMclp["2015/"], type = "l",
+      main = "交通銀行的收盤價", ylab = "收盤價")
```

# RSI的黃金交叉與死亡交叉

- RSI6曲線在0~100上下波動，波動較大。RSI24曲線在20~80波動，較平滑。
- RSI6取值超過80時，股價處於超買區，預期股票價格將要下跌，而收盤價大致處於下降階段。
- 黃金交叉信號附近，股價大致處於上漲行情。



# 交通銀行股票RSI指標交易實測

```
> ########################################
> # RSI補捉交通銀行股票買賣點                #
> ########################################
> rsi6 <- RSI(BOCMclp, n = 6, maType = "SMA")
> rsi6 <- na.omit(rsi6)
> names(rsi6) <- "rsi6"
> head(rsi6)
               rsi6
2014-01-10 42.85714
2014-01-13 61.53846
2014-01-14 66.66667
2014-01-15 46.15385
2014-01-16 30.00000
2014-01-17 23.07692
>
> rsi24 <- RSI(BOCMclp, n = 24, maType = "SMA")
> rsi24 <- na.omit(rsi24)
> names(rsi24) <- "rsi24"
> head(rsi24)
              rsi24
2014-02-06 45.45455
2014-02-07 47.61905
2014-02-10 56.09756
2014-02-11 60.86957
2014-02-12 59.09091
2014-02-13 63.82979
```

## 運用RSI值制定交易策略

- 獲取銀行股票交易資料，取出收盤價資料。
- 補捉買賣點: 計算RSI6和RSI24的值:
  - 當RSI6 > 80 或RSI6向下穿過RSI24時，為賣出信號。
  - 當RSI6 < 20 或RSI6向上穿過RSI24時，為買入信號。
- 交易策略執行: 按照RSI的買賣點，買入或賣出股票。
- 回測: 對交易策略進行投資評估。

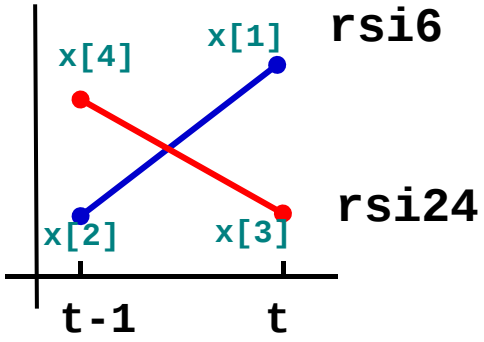
# 補捉交易信號

```
> # 補捉交易信號(1): rsi6補捉買賣點
> # 買入點
> longsig1 <- ifelse(rsi6 < 20, 1, 0)
> names(longsig1) <- "longsig1"
>
> # 第一個買入點信號是在2014年1月20日才出現
> head(longsig1[longsig1 == 1], 4)
           longsig1
2014-01-20        1
2014-01-21        1
2014-02-06        1
2014-02-07        1
>
>
> # 賣出點
> shortsig1 <- ifelse(rsi6 > 80, -1, 0)
> names(shortsig1) <- "shortsig1"
>
> # 第一個賣出點信號是在2014年2月11日才出現
> head(shortsig1[shortsig1 == (-1)], 4)
           shortsig1
2014-02-11        -1
2014-02-12        -1
2014-02-13        -1
2014-02-14        -1
```

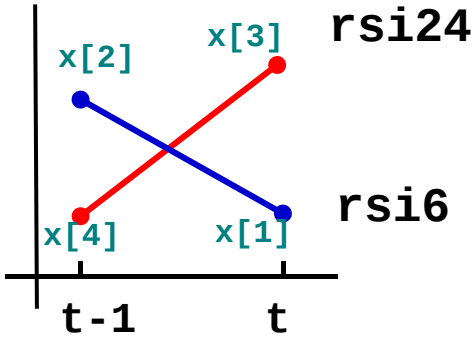
```
> # 交易信號(2): 黃金交叉與死亡交叉
> rsi6lag <- lag(rsi6, 1) #滯後一期的rsi6
> rsi24lag <- lag(rsi24, 1) #滯後一期的rsi24
> head(rsi24lag)
              rsi24
2014-02-06       NA
2014-02-07 45.45455
2014-02-10 47.61905
2014-02-11 56.09756
2014-02-12 60.86957
2014-02-13 59.09091
>
> # 合併資料
> RSIData <- merge(rsi6, rsi6lag, rsi24, rsi24lag)
> RSIData <- na.omit(RSIData)
> head(RSIData)
               rsi6   rsi6.1    rsi24  rsi24.1
2014-02-07  0.00000  0.00000 47.61905 45.45455
2014-02-10 75.00000  0.00000 56.09756 47.61905
2014-02-11 90.90909 75.00000 60.86957 56.09756
2014-02-12 91.66667 90.90909 59.09091 60.86957
2014-02-13 93.75000 91.66667 63.82979 59.09091
2014-02-14 83.33333 93.75000 60.41667 63.82979
```

# 交易信號(2): 黃金交叉與死亡交叉

```
                 t      t-1        t      t-1
> head(RSIData)x[1]   x[2]    x[3]   x[4]
                rsi6   rsi6.1    rsi24  rsi24.1
2014-02-07  0.00000  0.00000 47.61905 45.45455
2014-02-10 75.00000  0.00000 56.09756 47.61905
2014-02-11 90.90909 75.00000 60.86957 56.09756
2014-02-12 91.66667 90.90909 59.09091 60.86957
2014-02-13 93.75000 91.66667 63.82979 59.09091
2014-02-14 83.33333 93.75000 60.41667 63.82979
>
> # 補捉黃金交叉的買入點
> longsig2 <- apply(RSIData, 1, function(x) {
+   ifelse((x[1] > x[3]) & (x[2] < x[4]), 1, 0)
+ })
> head(longsig2, 4)
2014-02-07 2014-02-10 2014-02-11 2014-02-12
         0          1          0          0
> longsig2 <- xts(longsig2, order.by = index(RSIData))
> names(longsig2) <- "longsig2"
>
> # 補捉死亡交叉的賣出點
> shortsig2 <- apply(RSIData, 1, function(x) {
+   ifelse((x[1] < x[3]) & (x[2] > x[4]), -1, 0)
+ })
> head(shortsig2, 4)
2014-02-07 2014-02-10 2014-02-11 2014-02-12
         0          0          0          0
> shortsig2 <- xts(shortsig2, order.by = index(RSIData))
> names(shortsig2) <- "shortsig2"
```




黃金交叉(買入)




死亡交叉(賣出)

# 彙總買賣點信號

```
> # 合併4種交易訊號
> AllSignal <- merge(longsig1, longsig2,
+                    shortsig1, shortsig2, all = TRUE)
> AllSignal <- na.omit(AllSignal)
> head(AllSignal)
           longsig1 longsig2 shortsig1 shortsig2
2014-02-07        1        0         0         0
2014-02-10        0        1         0         0
...
2014-02-14        0        0        -1         0
> tail(AllSignal)
           longsig1 longsig2 shortsig1 shortsig2
2015-04-07        0        0         0         0
2015-04-08        0        1        -1         0
...
2015-04-14        0        0        -1         0
>
> # 彙總買賣點信號
> ComboSignal <- apply(AllSignal, 1, sum)
> ComboSignal <- xts(ComboSignal, order.by = index(AllSignal))
> head(ComboSignal)

> # 提取買入信號
> longsig <- ifelse(ComboSignal >= 1, 1, 0)
> index(ComboSignal)[which(longsig == 1)]

> # 提取賣出信號
> shortsig <- ifelse(ComboSignal <= (-1), -1, 0)
> index(ComboSignal)[which(shortsig == -1)]
```

2015/04/08

- longsig2=1: 短期rsi6向上穿過長期rsi24，價格有上升趨勢。
- shortsig1=-1: 短期rsi6取值大於80，股價進入超買區。
- 結論: 2015/04/08 股票價格走勢不確定，暫不作買賣點的判斷。

# RSI交易策略執行及回測

```
> # 計算交通銀行的收益率
> ret <- ROC(BOCMclp, type = "discrete")
> head(ret)
                BOCMclp
2014-01-02           NA
2014-01-03 -0.007853403
2014-01-06 -0.010554090
2014-01-07  0.005333333
2014-01-08  0.007957560
2014-01-09 -0.002631579
> buy <- lag(longsig, 1)
> sell <- lag(shortsig, 1)
> allsig <- longsig + shortsig
> trade <- lag(allsig, 1)
> head(trade)
           [,1]
2014-02-07   NA
2014-02-10    1
2014-02-11    1
2014-02-12   -1
2014-02-13   -1
2014-02-14   -1
```

- 作差法求動量值:

$$Momemtum_t = P_t - P_{t-m}$$

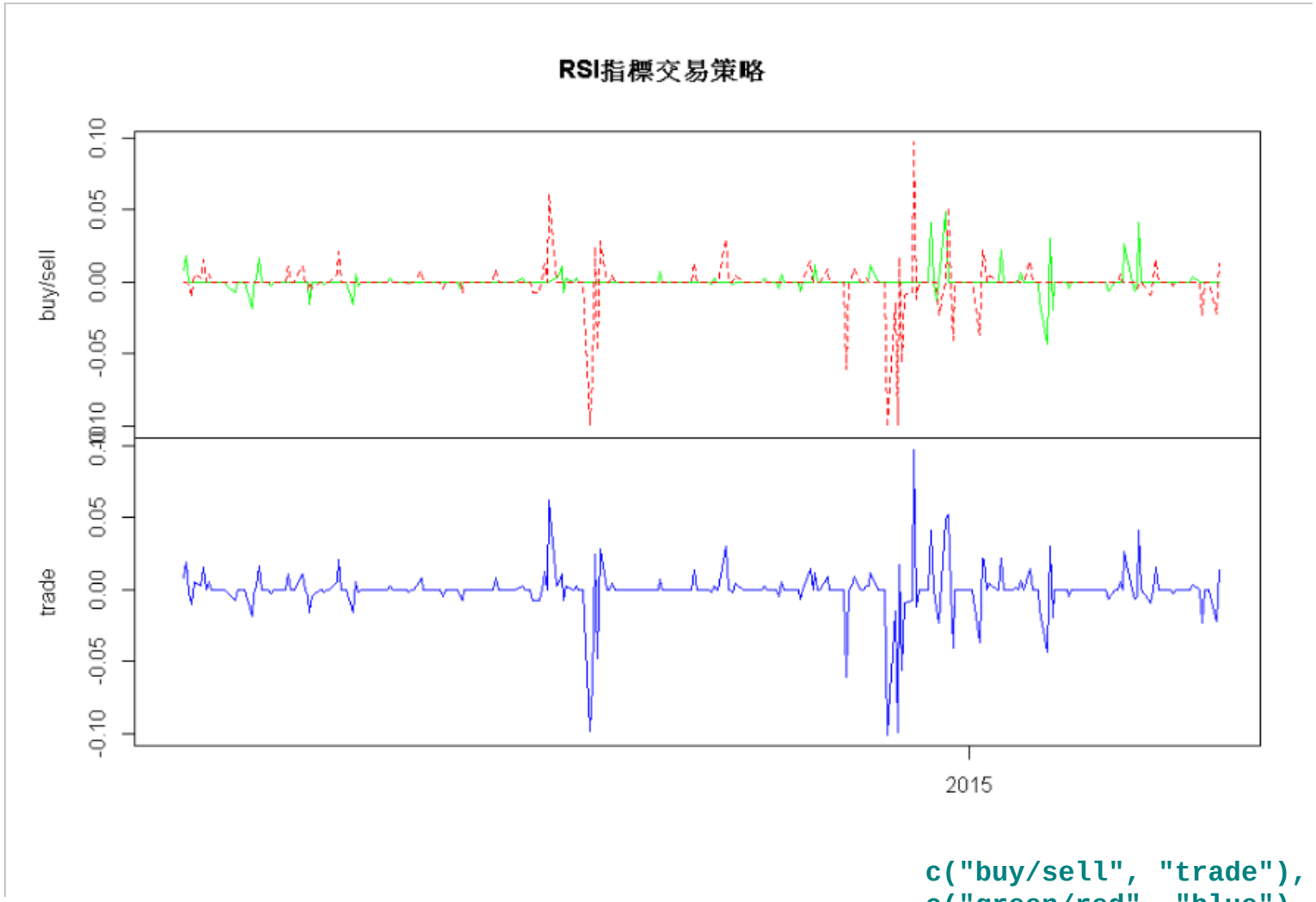
- 作除法求動量值(價格變化率):

$$Momemtum_t = ROC_t = \frac{P_t - P_{t-m}}{P_{t-m}}$$

```
> # 計算買入交易的收益率
> buyRet <- buy * ret
>
> # 計算賣出交易的收益率
> sellRet <- sell * ret
>
> # 計算買入賣出合併交易的收益率
> tradeRet <- trade * ret
>
> plot.zoo(cbind(buyRet, sellRet, tradeRet),
+   screens = c(1, 1, 2),
+   xlab = NULL, ylab = c("buy/sell", "trade"),
+   lty = c(1, 2, 1), col = c("green", "red", "blue"),
+   main = " RSI指標交易策略")
>
> legend("topright", legend = c("buy", "sell"),
+   col = c("green", "red"), lty = c(1, 2))
>
> #addLegend("topright", on=NA, legend.names = c("buy",
"sell"),
> #          col = c("green", "red"), lty = c(1, 2))
```

longsig1: rsi6 < 20, # 買入點
shortsig1: rsi6 > 80, # 賣出點
longsig2: 黃金交叉的買入點
shortsig2: 死亡交叉的賣出點

# RSI交易策略執行及回測




```
c("buy/sell", "trade"),
c("green/red", "blue")
```

# 計算信號預測準確率、平均獲勝收益率、平均損失收益率

```
> # 建立strat函式，計算信號預測準確率、平均獲勝收益率、平均損失收益率
> strat <- function(signal, ret) {
+   RSIRet <- signal * ret
+   WinRate <- length(RSIRet[RSIRet > 0]) / length(RSIRet[RSIRet != 0])
+   meanWin <- mean(RSIRet[RSIRet > 0])
+   meanLoss <- mean(RSIRet[RSIRet < 0])
+   return(c(WinRate, meanWin, meanLoss))
+ }
>
> Buy <- strat(buy, ret) #買入信號
> Sell <- strat(sell, ret) #賣出信號
> Trade <- strat(trade, ret) # 整個交易點
> Test <- rbind(Buy, Sell, Trade)
> colnames(Test) <- c("WinRate", "meanWin", "meanLoss")
> Test
        WinRate    meanWin     meanLoss
Buy   0.5510204 0.01261439 -0.009529424
Sell  0.5396825 0.01670012 -0.027675918
Trade 0.5446429 0.01489168 -0.019848019
```

# 比較RSI的累計收益率

```
> # 比較RSI的累計收益率
> # 本策略: RSI釋放買賣點訊號，隔一天，即進行買賣操作
> library(PerformanceAnalytics)
> names(ret) <- "stockRet"
> names(tradeRet) <- "tradeRet"
> charts.PerformanceSummary(cbind(ret, tradeRet), lty = c(1, 4),
+                           main = "RSI指標交易策略績效表現")
```

charts.PerformanceSummary {PerformanceAnalytics} R Documentation

Create combined wealth index, period performance, and drawdown chart

## Description

For a set of returns, create a wealth index chart, bars for per-period performance, and underwater chart for drawdown.

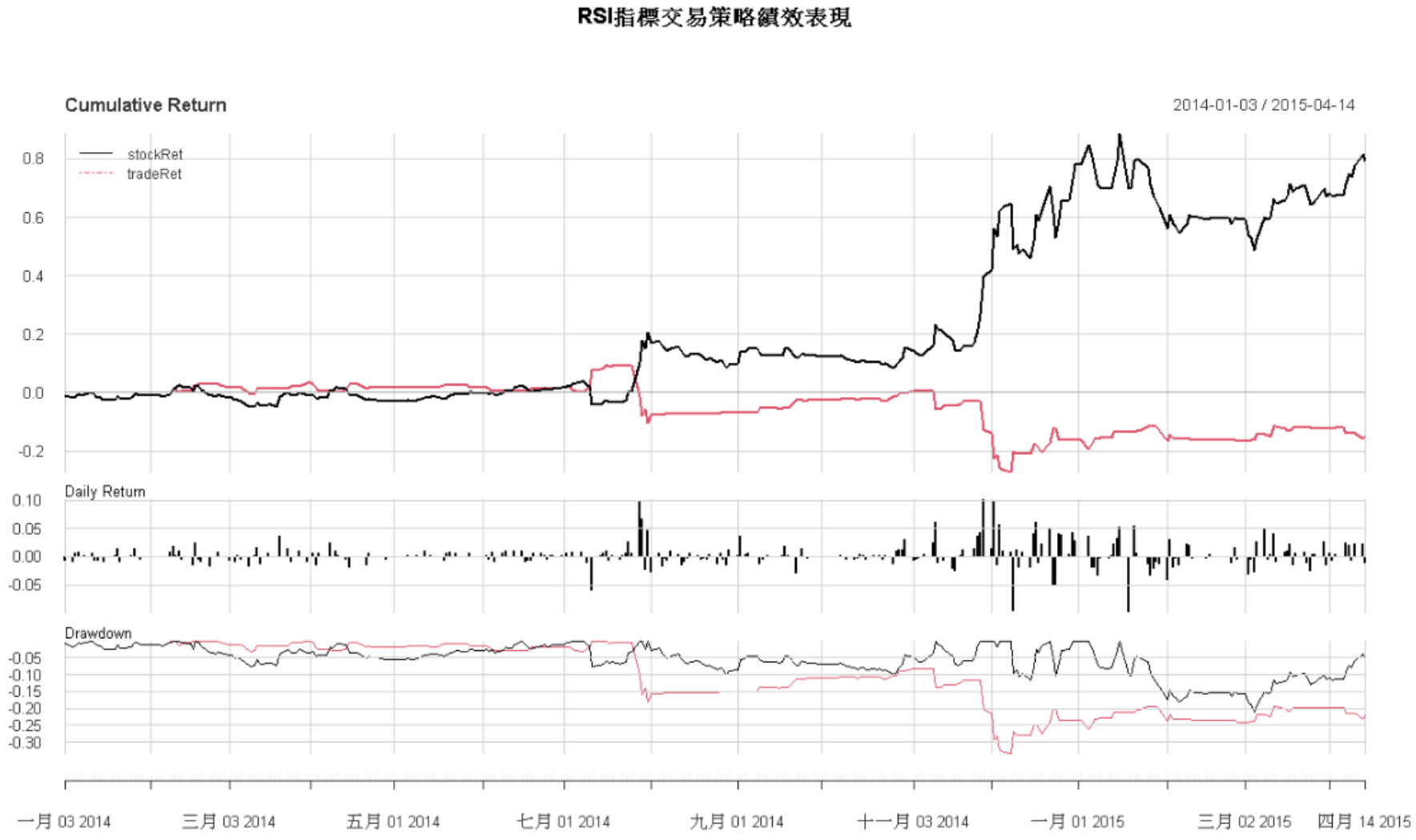
## Usage

```
charts.PerformanceSummary(
  R,
  Rf = 0,
  main = NULL,
  geometric = TRUE,
  methods = "none",
  width = 0,
  event.labels = NULL,
  ylog = FALSE,
  wealth.index = FALSE,
  gap = 12,
  begin = c("first", "axis"),
  legend.loc = "topleft",
  p = 0.95,
  plot.engine = "default",
  ...
)
```

## Arguments

R
: an xts, vector, matrix, data frame, timeSeries or zoo object of asset returns

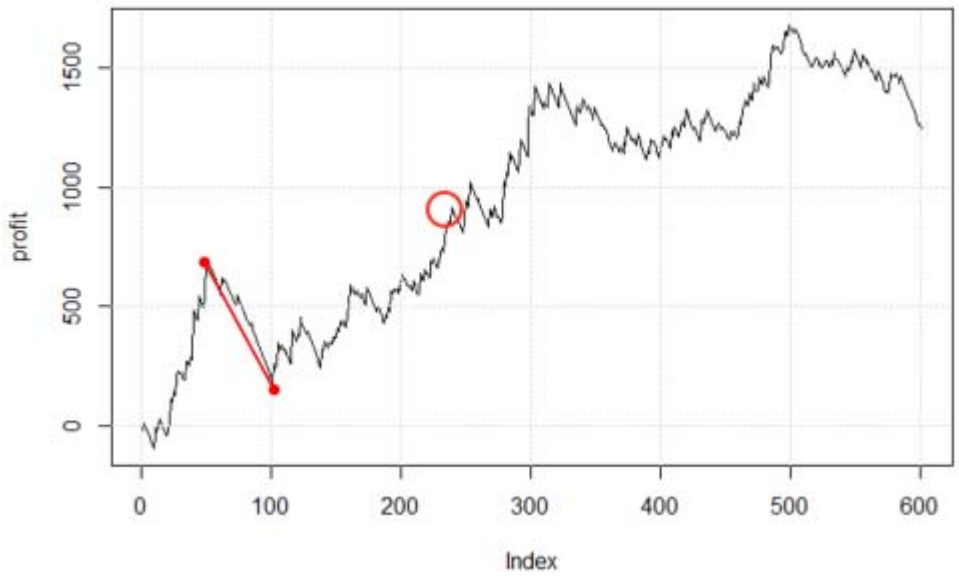
Rf
: risk free rate, in same period as your returns


RSI指標交易策略績效表現
Cumulative Return
2014-01-03 / 2015-04-14
stockRet
tradeRet
0.8
0.6
0.4
0.2
0.0
-0.2
Daily Return
0.10
0.05
0.00
-0.05
Drawdown
-0.05
-0.10
-0.15
-0.20
-0.25
-0.30
一月 03 2014
三月 03 2014
五月 01 2014
七月 01 2014
九月 01 2014
十一月 03 2014
一月 01 2015
三月 02 2015
四月 14 2015

# 回撤(DrawDown)

## Drawdown ( DD )

DD 又稱 “回撤” 或 “回檔” ，看的是策略 “累計獲利” 的損失，也就是損益曲線中一個波峰到波谷的距離。要注意的是波谷的確認方法，是指兩次波段高點形成的這段區間中的最低點，而 DD 就是計算前波高點到這中間低點的距離 ( 如圖上紅線 ) ，當策略損益創新高後，前波高點又會重新計算 ( 如圖上紅圈 ) 。總之 DD 就是在衡量獲利回吐的程度。



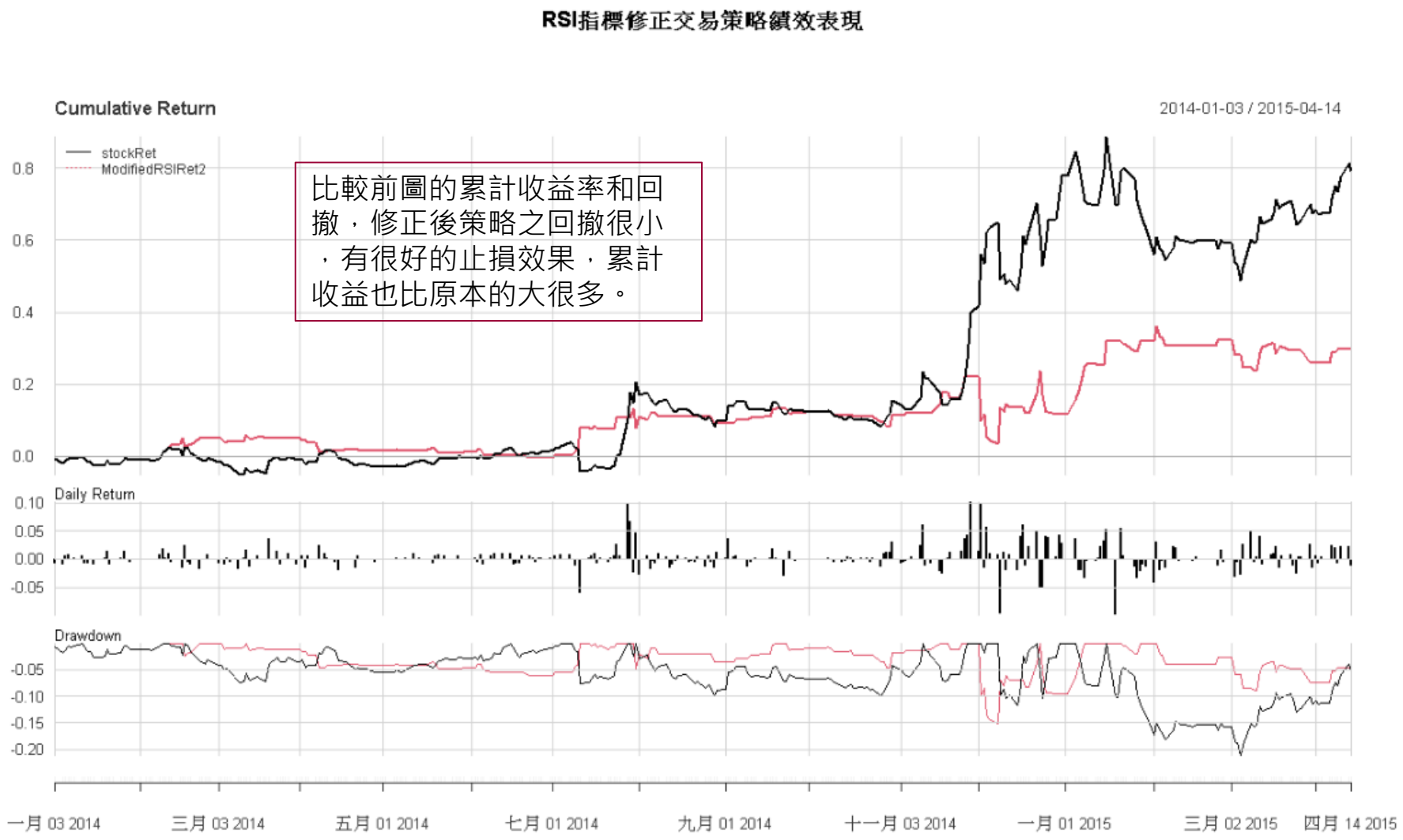

連續虧損或稱資金回撤 (DrawDown)

Source: https://www.stockfeel.com.tw/%E6%9C%80%E5%A4%A7%E5%9B%9E%E6%92%A4-max-drawdown-%E7%AD%96%E7%95%A5%E7%AE%A1%E7%90%86/

# 比較RSI的累計收益率

```
> # 修正策略: RSI釋放買賣點訊號，再隔三天，才進行買賣操作
> buy2 <- lag(longsig, 3)
> sell2 <- lag(shortsig, 3)
> trade2 <- lag(allsig, 3)
> Buy2 <- strat(buy2, ret)
> Sell2 <- strat(sell2, ret)
> Trade2 <- strat(trade2, ret)
> Test2 <- rbind(Buy2, Sell2, Trade2)
> colnames(Test2) <- c("WinRate", "meanWin", "meanLoss")
> Test2
         WinRate    meanWin    meanLoss
Buy2   0.5714286 0.01592665 -0.01254626
Sell2  0.5538462 0.01711650 -0.01785657
Trade2 0.5614035 0.01659594 -0.01562624
> tradeRet2 <- trade2 * ret
> names(tradeRet2) <- "ModifiedRSIRet2"
> charts.PerformanceSummary(cbind(ret, tradeRet2), lty = c(1, 3),
+                           main = "RSI指標修正交易策略績效表現")
```

# RSI指標修正交易策略績效表現圖




比較前圖的累計收益率和回撤，修正後策略之回撤很小，有很好的止損效果，累計收益也比原本的大很多。

# 均線系統策略

- 趨勢交易策略長期以來被認為具有良好的獲利能力。
- 如何判斷趨勢，及如何利用它進行交易?
- 最普遍方法為均線系統策略:
  - 簡單移動平均 (simple moving average, SMA)
  - 加權移動平均 (weighted moving average, WMA)
  - 指數移動平均 (Exponential moving average, EMA)

# 移動平均 (moving average)

- 移動平均法使用$n$期的資料進行平均:
- $SMA_t = \frac{x_t + x_{t-1} + x_{t-2} + \cdots + x_{t-n}}{n+1}$
- 期數選擇影響很大，該如何選擇:
  - 事件發展有無週期性。若有，應以此週期為期數，藉以消除週期影響。
  - 對趨勢平均性的要求。期數愈多，修勻效果愈平均。
  - 對趨勢反映近期變化敏感程度的要求。想得到長期趨勢，期數選擇要大。短期趨勢則選擇小期數。
- $WMA_t = w_0 x_t + w_1 x_{t-1} + \cdots + w_n x_{t-n}$
- $EMA_t = \alpha x_t + \alpha(1-\alpha)x_{t-1} + \alpha(1-\alpha)^2 x_{t-2} + \cdots$

| 日期 | 股價 | 權重 | EMA (5日) |
|---|---|---|---|
| 5 | | 0.2 | **56** |
| 6 | 67 | 0.2 | $0.8 \times$ **56** $+ 0.2 \times 67 =$ **58.2** |
| 7 | 60 | 0.2 | $0.8 \times$ **58.2** $+ 0.2 \times 60 = 58.56$ |

指數移動平均(EMA)是過去所有期價格的加權平均。(使用全部資料)

# TTR Package

SMA {TTR} R Documentation

Moving Averages

**Description**

Calculate various moving averages (MA) of a series.

**Usage**

```
SMA(x, n = 10, ...)

EMA(x, n = 10, wilder = FALSE, ratio = NULL, ...)

DEMA(x, n = 10, v = 1, wilder = FALSE, ratio = NULL)

WMA(x, n = 10, wts = 1:n, ...)

EVWMA(price, volume, n = 10, ...)

ZLEMA(x, n = 10, ratio = NULL, ...)

VWAP(price, volume, n = 10, ...)

VMA(x, w, ratio = 1, ...)

HMA(x, n = 20, ...)

ALMA(x, n = 9, offset = 0.85, sigma = 6, ...)
```

- `wilder=T`:
  $$\alpha = 1/n$$
- `wilder=F`:
  $$\alpha = 2/(n+1)$$

SMA: Simple moving average.
EMA: Exponential moving average.
WMA: Weighted moving average.
DEMA: Double-exponential moving average.
EVWMA: Elastic, volume-weighted moving average.
ZLEMA: Zero lag exponential moving average.
VWMA: Volume-weighed moving average (same as VWAP).
VWAP: Volume-weighed average price (same as VWMA).
VWA: Variable-length moving average.
HMA: Hull moving average.
ALMA: Arnaud Legoux moving average.

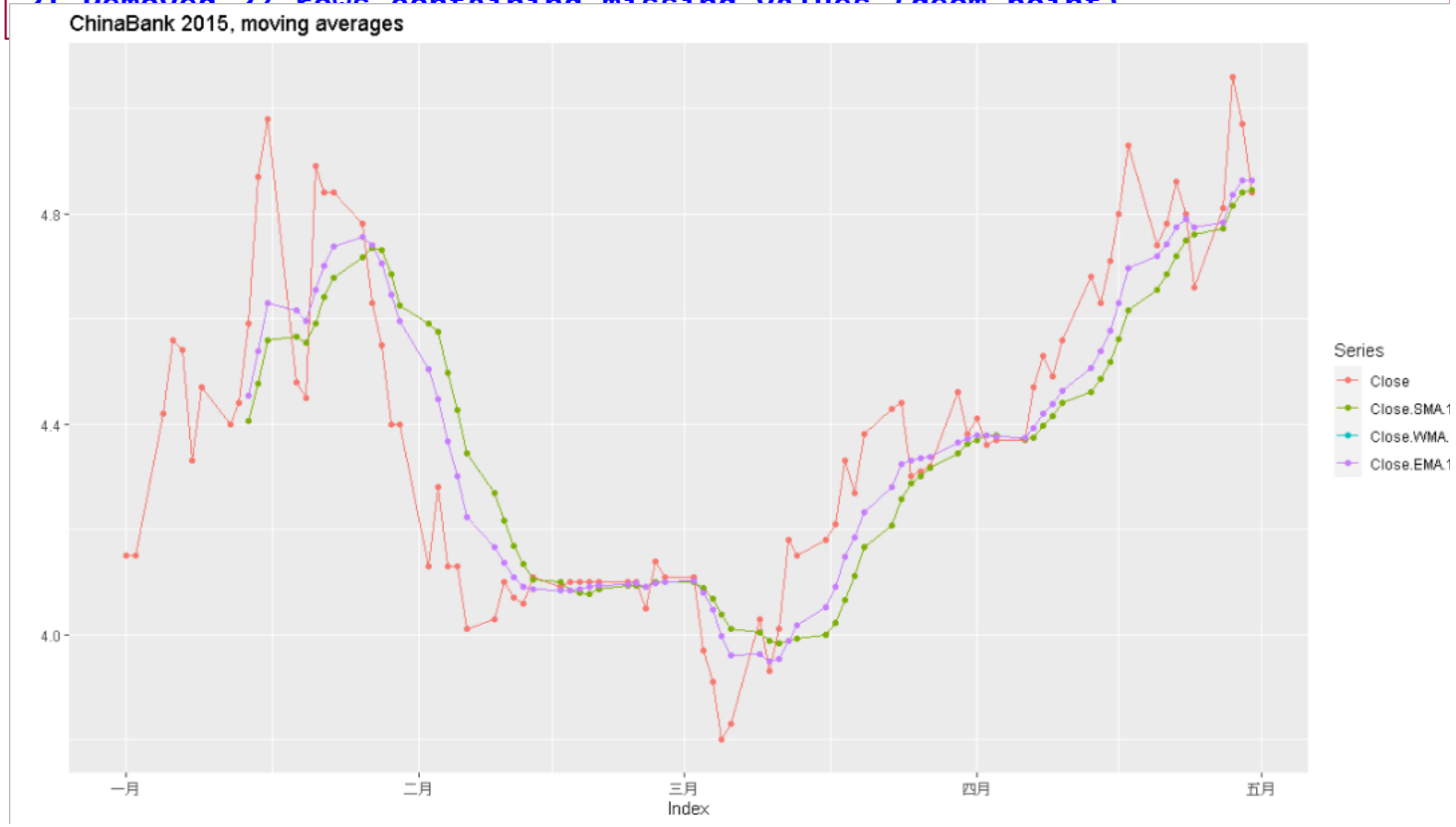
# 中國銀行的股票資料與均線分析

```
> library(quantmod)
> ChinaBank <- read.csv("data/ChinaBank.csv", header = TRUE)
> head(ChinaBank)
  X       Date Open High  Low Close   Volume
1 1 2014-01-02 2.62 2.62 2.59  2.61 41632500
...
6 6 2014-01-09 2.51 2.53 2.49  2.50 45339800
>
> dfclose <- ChinaBank[, c(2, 6)]
> head(dfclose)
        Date Close
1 2014-01-02  2.61
...
6 2014-01-09  2.50
> CBstclose <- xts(dfclose[, -1],
+ order.by = as.Date(dfclose$Date))
> names(CBstclose) <- "Close"
> head(CBstclose)
           Close
2014-01-02  2.61
...
2014-01-09  2.50
> summary(CBstclose)
     Index                Close
 Min.   :2014-01-02   Min.   :2.450
 1st Qu.:2014-05-05   1st Qu.:2.600
 Median :2014-09-02   Median :2.700
 Mean   :2014-09-01   Mean   :3.146
 3rd Qu.:2014-12-31   3rd Qu.:3.890
 Max.   :2015-04-30   Max.   :5.060
```

```
> ########################################
> # 計算2015年資料之三種移動平均            #
> ########################################
> close <- CBstclose["2015"]
> sma10 <- SMA(close, 10) # 10日SMA
> names(sma10) <- "Close.SMA.10"
> tail(sma10)
           Close.SMA.10
2015-04-23        4.749
...
2015-04-30        4.845
> length(sma10)
[1] 86
>
> wma10 <- WMA(close, 10) # 10日WMA
> names(wma10) <- "Close.WMA.10"
> tail(wma10)
           Close.WMA.10
2015-04-23     4.788909
...
2015-04-30     4.862364
>
> ema10 <- WMA(close, 10) # 10日EMA
> names(ema10) <- "Close.EMA.10"
> tail(ema10)
           Close.EMA.10
2015-04-23     4.788909
...
2015-04-30     4.862364
```

# 中國銀行的股票資料與均線分析
# 繪製2015年資料之三種移動平均線

```
> library(ggplot2)
> CB.close.2015.ma <- merge(close, sma10, wma10, ema10)
> # CB.close.2015.ma <- na.omit(CB.close.2015.ma)
> autoplot(CB.close.2015.ma, facet = NULL) +
+   geom_point() +
+   labs(title="ChinaBank 2015, moving averages")
Warning messages:
1: Removed 27 row(s) containing missing values (geom_path).
2: Removed 27 rows containing missing values (geom_point).
```




- 三條均線對原有收盤價都進行了一定程度的平均，走勢類似，幾乎重合。
- 在上升或者下降的趨勢上，這三條均線對收盤價曲線有一定的滯後性。
- 移動平均線是跟蹤行情的曲線，從本性上說就具有滯後性(其它的技術指標也是如此)，也就是說，移動平均線對市場行情的反應總是落後的，
- 移動平均線的時間周期越長，反應越滯後。

# 均線時間跨度

```
> sma5 <- SMA(close, 5)
> names(sma5) <- "Close.SMA.5"
> sma30 <- SMA(close, 30)
> names(sma30) <- "Close.SMA.30"
>
> CB.close.2015.sma5.sma30 <- merge(close, sma5, sma30)
> autoplot(CB.close.2015.sma5.sma30, facet = NULL) +
+   geom_point() +
+   labs(title="ChinaBank 2015, SMA5 vs SMA30")
```

- **時間跨度很重要**
  - 以日為單位: 5日均線、10日均線、20日均線、25日均線
  - 以周為單位: 5周均線、10周均線
  - 以月為單位: 3月均線、12月均線、24月均線
- 不同時間跨度的均線對於價格趨勢的刻畫和敏感度會有差別。
- 短期均線/長期均線: 相對的。例如: 5日均線是短期均線，而20日均線是長期均線。
- 5日均線和收盤價上下走勢一致，30日均線較為平均。30日均線是描述收盤價的月度趨勢。
- **5日均線相對於價格線有壓力與支撐的意義:**
  - 當價格線向下突破5日均線，價格短期處於向下趨勢。
  - 當價格線向上突破5日均線，價格短期處於上升趨勢。
  - 若價格線沒有向下突破5日均線，價格未來有可能有上漲趨勢。

# 交易策略: SMA制定買賣點 (ChinaBank)

```
> ChinaBank <- read.csv("data/ChinaBank.csv", header = TRUE)
> head(ChinaBank)
  X       Date Open High  Low Close   Volume
1 1 2014-01-02 2.62 2.62 2.59  2.61 41632500
...
6 6 2014-01-09 2.51 2.53 2.49  2.50 45339800
>
> dfclose <- ChinaBank[, c(2, 6)]
> class(dfclose)
[1] "data.frame"
> head(dfclose)
        Date Close
1 2014-01-02  2.61
...
6 2014-01-09  2.50
>
> CBstclose <- xts(dfclose[, -1], order.by = as.Date(dfclose$Date))
> names(CBstclose) <- "Close"
> head(CBstclose)
           Close
2014-01-02  2.61
...
2014-01-09  2.50
> tail(CBstclose)
           Close
2015-04-23  4.80
...
2015-04-30  4.84
```

# 交易策略: SMA制定買賣點

```
> sma10 <- SMA(CBstclose, 10)
> names(sma10) <- "Close.SMA.10"
>
> # 繪製close, sma10的時序圖
> par(mfrow = c(2, 1))
> plot(CBstclose)
> lines(sma10, col = "red")
>
> plot(CBstclose["2015"])
> lines(sma10["2015"], col = "red")
>
> # 補捉價格線從下向上穿10日均線，和從上向下穿10日均線的日期
> # 從下向上穿10日均線 => 釋放買入訊號
> # 從上向下穿10日均線 => 釋放賣出訊號
> # 執行買賣交易的時點，為買賣訊號出現後的第2期
> # 然後評估此交易策略好壞
>
> CBsma <- na.omit(merge(sma10, lag(sma10, 1)))
> head(CBsma)
           Close.SMA.10 Close.SMA.10.1
2014-01-16        2.515          2.526
...
2014-01-23        2.494          2.496
>
> CBclose <- na.omit(merge(CBstclose, lag(CBstclose, 1)
> head(CBclose)
           Close Close.1
2014-01-03  2.56    2.61
...
2014-01-10  2.49    2.50
```

- 補捉價格線從下向上穿10日均線，和從上向下穿10日均線的日期。
- 從下向上穿10日均線 => 釋放買入訊號
- 從上向下穿10日均線 => 釋放賣出訊號
- 執行買賣交易的時點，為買賣訊號出現後的第2期
- 評估此交易策略好壞

# 繪製close, sma10的時序圖

# 補捉價格線從下向上穿10日均線，和從上向下穿10日均線的日期

```
> #  刪除前9期CBclose的資料，保持與CBsma的期數一致
> CBclose <- CBclose[-(1:9), ]
> head(CBclose)
           Close Close.1
2014-01-16  2.50    2.51
...
2014-01-23  2.48    2.51
>
> # 作差法: 求收盤價與10日移動平均的差值
> sigdata <- CBclose - CBsma
> colnames(sigdata) <- c("close_sma10", "lag_close_sma10
> head(sigdata)
           close_sma10 lag_close_sma10
2014-01-16      -0.015          -0.016
...
2014-01-23      -0.014           0.014
>
> # 穿越訊號
> cross <- function(x) {
+   ifelse(x[1] > 0 & x[2] < 0,  1,
+          ifelse(x[1] < 0 & x[2] > 0, -1, 0))
+ }
```

```
> # 捕捉價格線突破均線的日期
> SmaSignal <- apply(sigdata, 1, cross)
> SmaSignal <- xts(SmaSignal, order.by = index(sigdata))
> head(SmaSignal)
            [,1]
2014-01-16    0
...
2014-01-23   -1
```

- 補捉價格線從下向上穿10日均線，和從上向下穿10日均線的日期。
- 從下向上穿10日均線 => 釋放買入訊號
- 從上向下穿10日均線 => 釋放賣出訊號
- 執行買賣交易的時點，為買賣訊號出現後的第2期

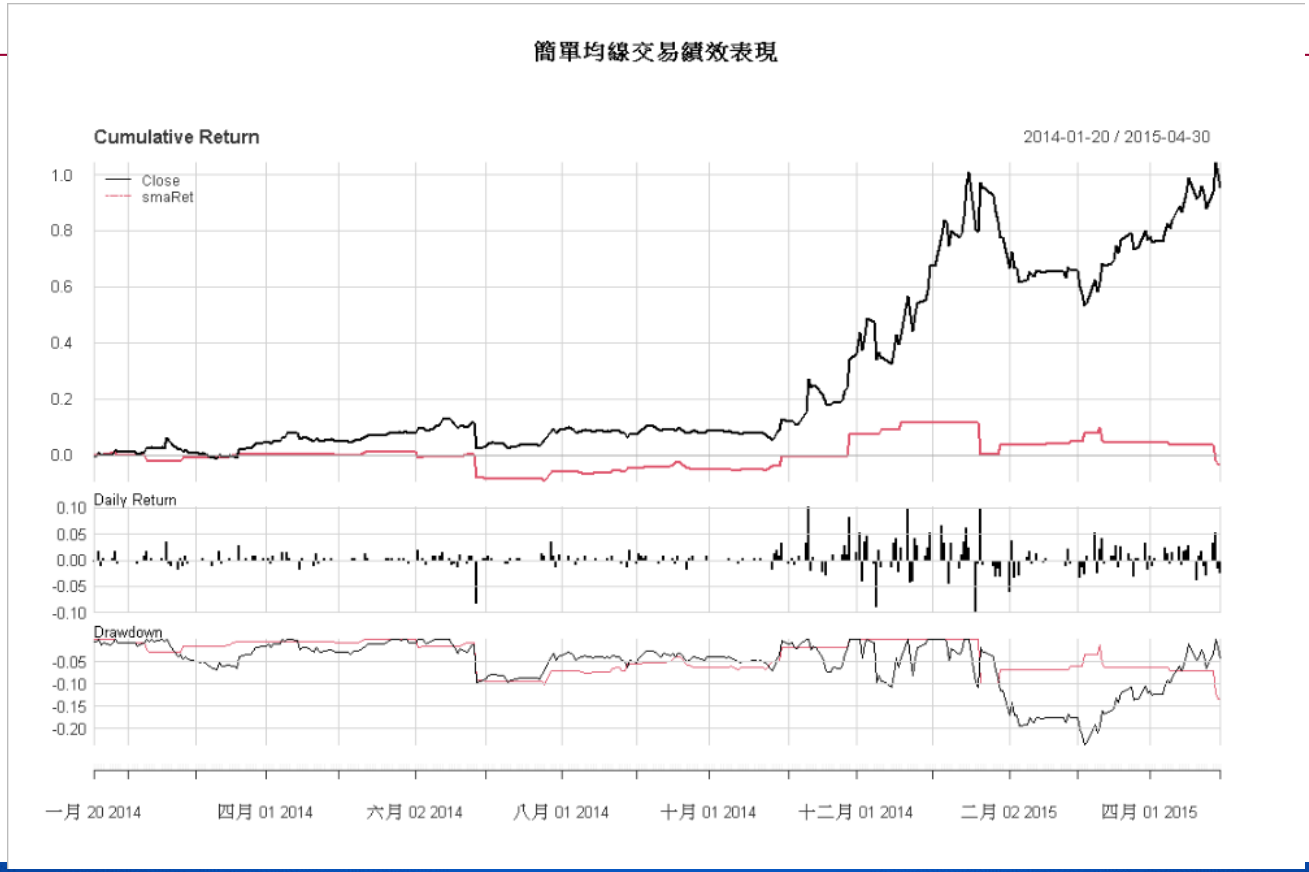
# 制定買賣交易日期

```
> # 制定買賣交易日期: 交易信號滯後2期
> SmaTrade <- lag(SmaSignal, 2)
> SmaTrade <- na.omit(SmaTrade)
> head(SmaTrade)
           [,1]
2014-01-20    0
...
2014-01-27   -1
>
> # 取出買入點
> SmaBuy <- SmaTrade[SmaTrade == 1]
> length(SmaBuy)
[1] 31
> head(SmaBuy)
           [,1]
2014-01-24    1
...
2014-03-25    1
>
> # 取出賣出點
> SmaSell <- SmaTrade[SmaTrade == (-1)]
> length(SmaSell)
[1] 31
> head(SmaSell)
           [,1]
2014-01-27   -1
...
2014-04-17   -1
```

```
> # 計算日收益率
> CBret <- ROC(CBstclose, type = "discrete")
> names(CBret) <- "CBret"
> head(CBret)
                  CBret
2014-01-02           NA
2014-01-03 -0.019157088
...
2014-01-09 -0.003984064
> smaRet <- CBret * SmaTrade
> names(smaRet) <- "smaRet"
> head(smaRet)
                smaRet
2014-01-20 0.000000000
...
2014-01-27 0.004016064
```

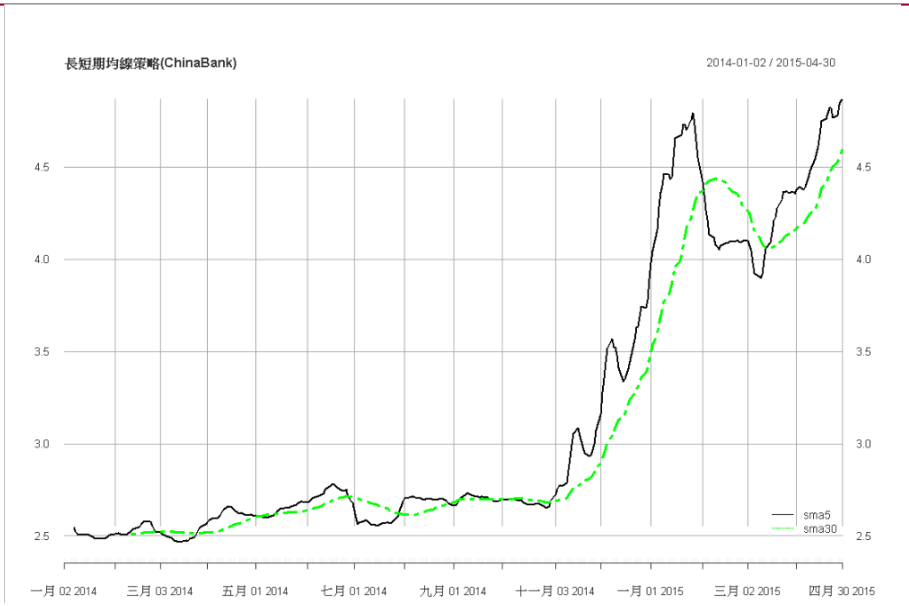
# 繪製績效表現圖，計算買賣點預測準確率

```
> # 繪製績效表現圖
> library(PerformanceAnalytics)
> charts.PerformanceSummary(merge(CBret["2014-01-20/"], smaRet), lty = c(1, 6),
+                           main = "簡單均線交易績效表現")
> # 計算買賣點預測準確率
> win <- smaRet[smaRet > 0]
> smawin <- length(win) / length(smaRet[smaRet != 0])
> smawin
[1] 0.6
```



# 交易策略: 雙均線制定買賣點 (ChinaBank)

```
> sma5 <- SMA(CBstclose, 5)
> length(sma5)
[1] 345
> sma30 <- SMA(CBstclose, 30)
> head(na.omit(sma30))
                SMA
2014-02-13 2.511667
...
2014-02-20 2.516333
> # 繪製長短期均線圖
> plot(sma5, type = "l", main = "長短期均線策略(ChinaBank)")
> lines(sma30, lty = 6, lwd = 3, col = "green")
> addLegend("bottomright", legend.names = c("sma5", "sma30"),
+           col = c("black", "green"), lty = c(1, 6))
```




```
> # 取lag
> # example:
> # x <- 1:10
> # embed (x, 3)
> head(sma5)
             SMA
2014-01-02    NA
2014-01-03    NA
2014-01-06    NA
2014-01-07    NA
2014-01-08 2.546
2014-01-09 2.524
> SMA5 <- embed(sma5, 2)
> head(SMA5)
      [,1]  [,2]
[1,]    NA    NA
[2,]    NA    NA
[3,]    NA    NA
[4,] 2.546    NA
[5,] 2.524 2.546
[6,] 2.510 2.524
> SMA30 <- embed(sma30, 2)
```

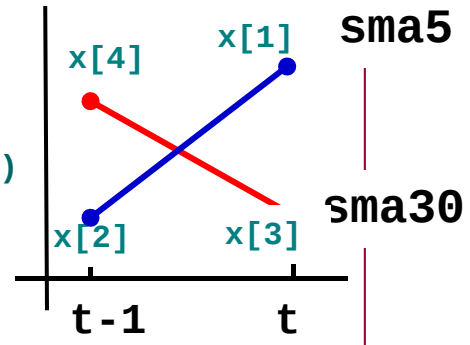
# 交易策略: 雙均線制定買賣點 (ChinaBank)

```
> # 合併長短期sma
> smaLS <- cbind(SMA5, SMA30)
> smaLS <- xts(smaLS, order.by = index(sma5[-1]))
> smaLS <- na.omit(smaLS)
> colnames(smaLS) <- c("sma5", "lagsma5", "sma30", "lagsma30")
> head(smaLS)
            sma5 lagsma5    sma30 lagsma30
2014-02-14 2.538   2.528 2.509333 2.511667
...
2014-02-21 2.582   2.574 2.519000 2.516333
> # 判斷向上突破點
> Upcross <- function(x) {
+   ifelse(x[2] < x[4] & x[1] > x[3], 1, 0)
+ }
>
> # 判斷向下突破點
> Downcross <- function(x) {
+   ifelse(x[2] > x[4] & x[1] < x[3], -1, 0)
+ }
>
> # 捕捉短線向上突破長線日期
> Upsig <- apply(smaLS, 1, Upcross)
> Upsig <- xts(Upsig, order.by = index(smaLS))
> names(Upsig) <- "Upsig"
> head(Upsig)
           Upsig
2014-02-14     0
...
2014-02-20     0
2014-02-21     0
```

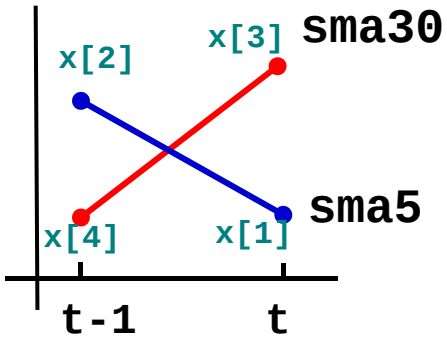
- 雙均線交叉策略: 利用長短期均線的相對關係，來識別價格趨勢的反轉。

**三個步驟:**

- 先求出長短期均線
- 當短期均線從下向上穿過長期均線，釋放買入信號。
- 當短期均線從上向下穿過長期均線，釋放賣出信號。
- 計算買入交易，賣出交易及買賣交易的績效。




(買入)




(賣出)

# 交易策略: 雙均線制定買賣點 (ChinaBank)

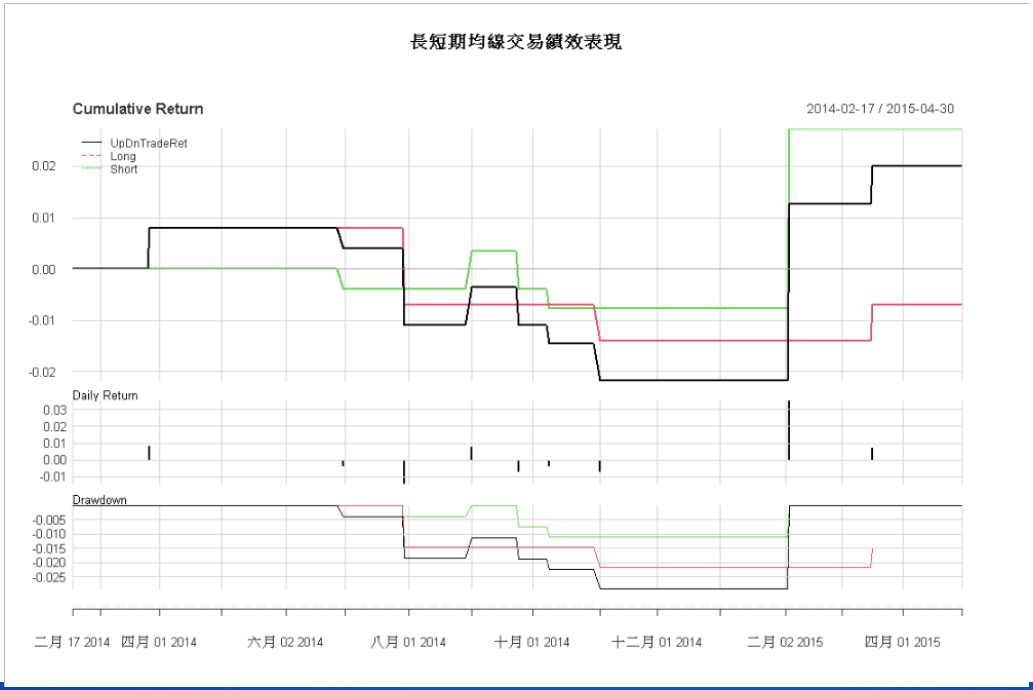
```
> # 短線向上突破長線，釋放買入信號
> UpBuy <- lag(Upsig)
> head(UpBuy)
           Upsig
2014-02-14    NA
2014-02-17     0
...
2014-02-21     0
> UpBuy[UpBuy == 1]
           Upsig
2014-03-27     1
2014-05-14     1
...
2014-11-03     1
2015-03-17     1
>
> # 捕捉短線向下突破長線日期
> Downsig <- apply(smaLS, 1, Downcross)
> Downsig <- xts(Downsig, order.by = index(smaLS))
> names(Downsig) <- "Downsig"
>
> # 短線向下突破長線，釋放賣出信號
> DownSell <- lag(Downsig)
> DownSell[DownSell == -1]
           Downsig
2014-03-04      -1
...
2015-02-04      -1
```

```
> # 計算日收益率
> CBret <- ROC(CBstclose, type = "discrete")
> head(CBret)
                   Close
2014-01-02            NA
...
2014-01-09 -0.003984064
> # 計算買入點的預測正確率
> Long <- UpBuy * CBret
> names(Long) <- "Long"
> winL <- Long[Long > 0]
> winLrate <- length(winL) / length(Long[Long !=
0])
> winLrate
[1] 0.5
```

```
>
> # 計算賣出點的預測正確率
> Short <- DownSell * CBret
> names(Short) <- "Short"
> winS <- Short[Short > 0]
> winSrate <- length(winS) / length(Short[Short != 0])
> winSrate
[1] 0.4
```

# 交易策略: 雙均線制定買賣點 (ChinaBank)

```
> # 計算所有買賣點的預測正確率
> UpDnTrade <- UpBuy + DownSell
> UpDnTradeRet <- UpDnTrade * CBret
> names(UpDnTradeRet) <- "UpDnTradeRet"
> a <- length(UpDnTradeRet[UpDnTradeRet > 0])
> b <- length(UpDnTradeRet[UpDnTradeRet != 0])
> winrate <- a/b
> winrate
[1] 0.4444444
> # 長短期均線交易績效表現圖
> charts.PerformanceSummary(merge(UpDnTradeRet, Long, Short),
+                           lty = c(1, 2, 6), main = "長短期均線交易績效表現")
```



# 指數平滑異同移動平均線
# 異同移動平均線 (MACD)

- 異同移動平均線 (Moving Average Convergence/Divergence, MACD)由Gerald Appel在1979年提出，是股票分析中常見的指標。
- MACD由兩線一柱組成:
  - **DIF: 一般由12日EMA(快線，短期)減去26日EMA(慢線，長期)而得。**
  - **信號線(DEA): 是DIF的9日EMA。**
  - **柱狀為MACD: 由DIF與DEA作差而得。**
- DIF: 描述短期和長期移動平均線的聚合與差離。
- DEA: 對雙均線的差離再次取移動平均，得到的信號線。
- MACD可反映出股票近期價格走勢的能量和變化的強度，透過分析快慢速線和柱狀圖來把握股票的買入和賣出點。
- 一般預設MACD的求值參數為12, 26, 和9。
- MACD是價格指標的重要工具。其利用利用快慢二條〈快線：DIF，慢線：MACD〉移動平均線的變化作為盤勢的研判指標，具有確認中長期波段走勢並找尋短線買賣點的功能。
- MACD的原理在於以長天期(慢的)移動平均線來作為大趨勢基準，而以短天期(快的)移動平均線作為趨勢變化的判定。

https://wiki.mbalib.com/zh-tw/平滑异同移动平均线

# TTR Package

MACD {TTR} R Documentation

MACD Oscillator

## Description

The MACD was developed by Gerald Appel and is probably the most popular price oscillator. The MACD function documented in this page compares a fast moving average (MA) of a series with a slow MA of the same series. It can be used as a generic oscillator for any univariate series, not only price.

## Usage

```
MACD(x, nFast = 12, nSlow = 26, nSig = 9, maType, percent = TRUE, ...)
```

## Arguments

`x`
: Object that is coercible to xts or matrix; usually price, but can be volume, etc.

`nFast`
: Number of periods for fast moving average.

`nSlow`
: Number of periods for slow moving average.

`nSig`
: Number of periods for signal moving average.

maType=ema
percent=T: MACD的值為DIF與DEA差值的百分比。
percent=F: MACD的值為DIF與DEA的差值。

# 異同移動平均線 (MACD) 捕捉買賣點

```
> # 計算中國銀行股票的MACD
> CB.macd <- MACD(CBstclose, nFast = 12, nSlow = 26, nSig = 9,
+   maType = "EMA", percent = FALSE)
>
> CB.macd <- na.omit(CB.macd)
> head(CB.macd)
                   macd      signal
2014-02-19 0.015784353 0.004527369
...
2014-02-26 0.009648815 0.011467374
>
> # MACD捕捉買賣點
> ChinaBank <- xts(ChinaBank[, -c(1, 2)], order.by = as.Date(ChinaBank$Date))
> head(ChinaBank)
           Open High  Low Close   Volume
2014-01-02 2.62 2.62 2.59  2.61 41632500
...
2014-01-09 2.51 2.53 2.49  2.50 45339800
> CB15 <- ChinaBank["2015/"]
> head(CB15)
           Open High  Low Close     Volume
2015-01-01 4.15 4.15 4.15  4.15          0
...
2015-01-08 4.55 4.57 4.31  4.33 1489272600
> CB.macd.lag <- lag(CB.macd, 1)
> macd <- na.omit(merge(CB.macd, CB.macd.lag))
> head(macd)
                   macd      signal      macd.1    signal.1
2014-02-20 0.019772737 0.007576443 0.015784353 0.004527369
...
2014-02-27 0.007669651 0.010707829 0.009648815 0.011467374
```

# 異同移動平均線 (MACD) 捕捉買賣點

```
> # K線圖
> chartSeries(CB15, theme = 'white', name = '中國銀行2015年K線圖',
+   up.col = 'red',  dn.col = 'green')
> addMACD()
```

# MACD交易思想

- (ChinaBank2015) DIF幾乎處於DEA上方，兩線交叉次數少，釋放的買賣訊號也少。
- 2015/03/04附近: DIF從上向下穿越DEA，兩者值都小於-2，觀察K線圖，推測市場可能處於下跌行情。
- 2015/03/09附近: DIF從下向上穿越DEA，兩者呈現上升，觀察K線圖，市場慢慢走強，3/9之後，市場處於上升行情。

柱形圖在零刻度附近，釋出買賣信號。
柱形圖在零刻度上方(DIF > DEA)=>市場走勢較強
柱形圖在零刻度下方(DIF < DEA)=>市場走勢較弱

灰色實線: DIF，紅色虛線: DEA，
柱形圖: DIF與DEA的差值
DIF > 0 & DEA > 0 => 市場處於多頭行情
DIF < 0 & DEA < 0 => 市場處於空頭行情

DIF上穿DEA => 釋出買入信號
DIF下穿DEA => 釋出賣出信號

# 異同移動平均線 (MACD) 捕捉買賣點

```
> # MACD 交易策略函數
> macdcross <- function(x) {
+   ifelse(x[1] > x[2] & x[3] < x[4] & all(x > 0), 1,
+          ifelse(x[1] < x[2] & x[3] > x[4] & all(x < 0),-1, 0))
+ }
>
> # 求MACD指標買賣點信號
> macdSignal <- apply(macd, 1, macdcross)
> macdSignal <- xts(as.numeric(macdSignal), order.by = index(macd))
> head(macdSignal)
           [,1]
2014-02-20    0
...
2014-02-27    0
>
> # 設定買賣交易日期
> macdTrade <- lag(macdSignal, 1)
> head(macdTrade)
           [,1]
2014-02-20   NA
...
2014-02-27    0
> macdTrade <- macdTrade[-1]
```

策略:

- DIF > 0 & DEA > 0 ，DIF向上突破DEA => 買入信號
- DIF < 0 & DEA < 0 ，DIF向下跌破DEA => 賣出信號

```
> # 計算收益率
> CBret <- ROC(CBstclose, type = "discrete")
> head(CBret)
                  Close
2014-01-02           NA
2014-01-03 -0.019157088
...
2014-01-09 -0.003984064
>
> # 計算MACD指標的預測正確率
> macdRet <- CBret * macdTrade
> win <- macdRet[macdRet > 0]
> macdwin <- length(win) / length(macdRet[macdRet != 0])
> macdwin
[1] 0.5
```

# (1) 下載&安裝 元大Smart API

元大期貨的官方網站: https://www.yuantafutures.com.tw/
元大Smart API: https://www.yuantafutures.com.tw/file-repository/content/smartAPI/page1.html

交易API: API的全名是應用編程介面(Application Programming Interface)，是為了程式間互相溝通所設計出來的介面，因為要完成交易的整體流程，必須取得行情與下單指令，所以交易API本質上是由行情API與下單API所組成，

使用手冊: https://www.yuantafutures.com.tw/file-repository/content/smartAPI/SmartAPI.pdf

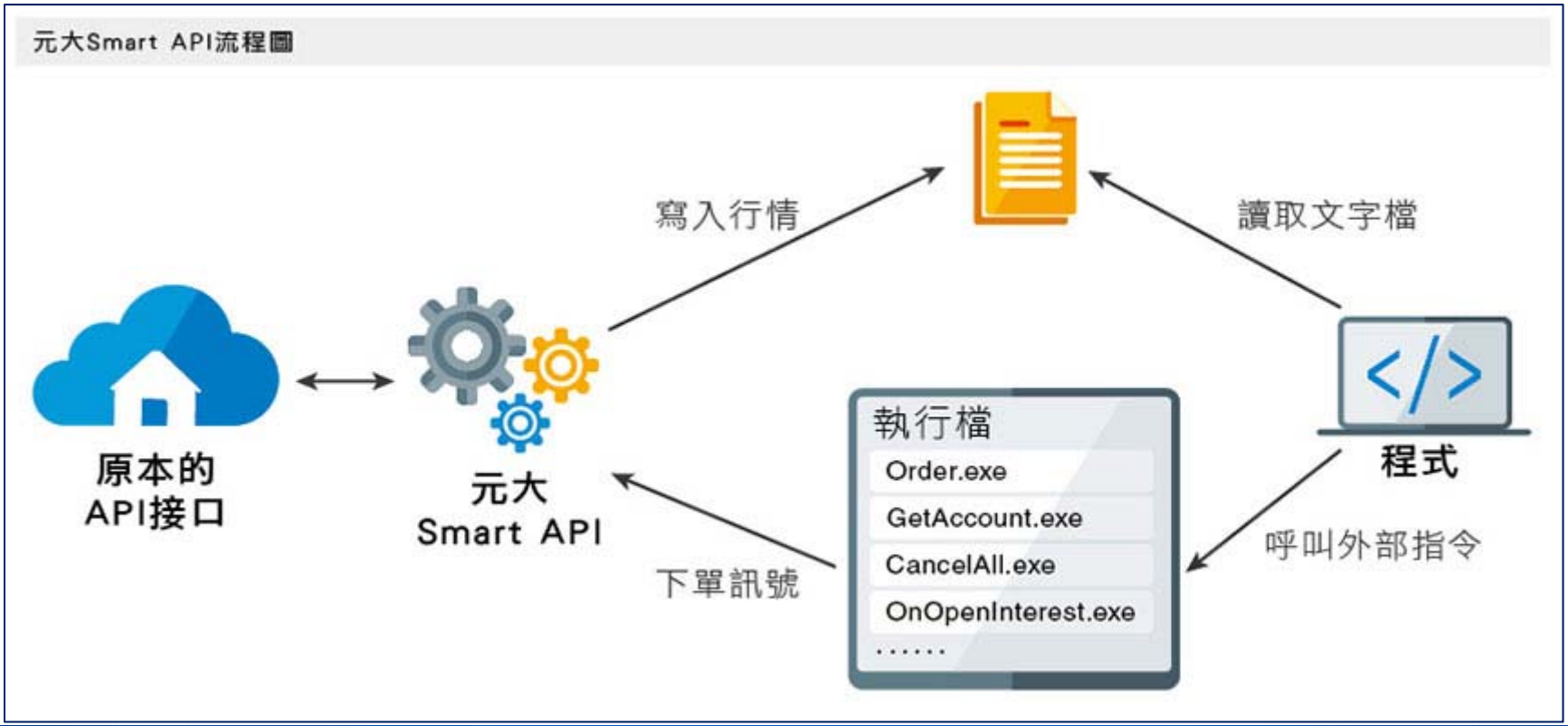
# (2)開通API權限

若要透過元大SmartAPI來實做程式交易，需要開通行情API與下單API，開通方式如下:

- 行情API：請找所屬營業員填寫行情API開通申請表
- 下單API：由以下網址完成線上開通
https://www.yuantafutures.com.tw/Appointment/pages/member-manage/ESign.aspx
或由首頁右方[線上服務申請]

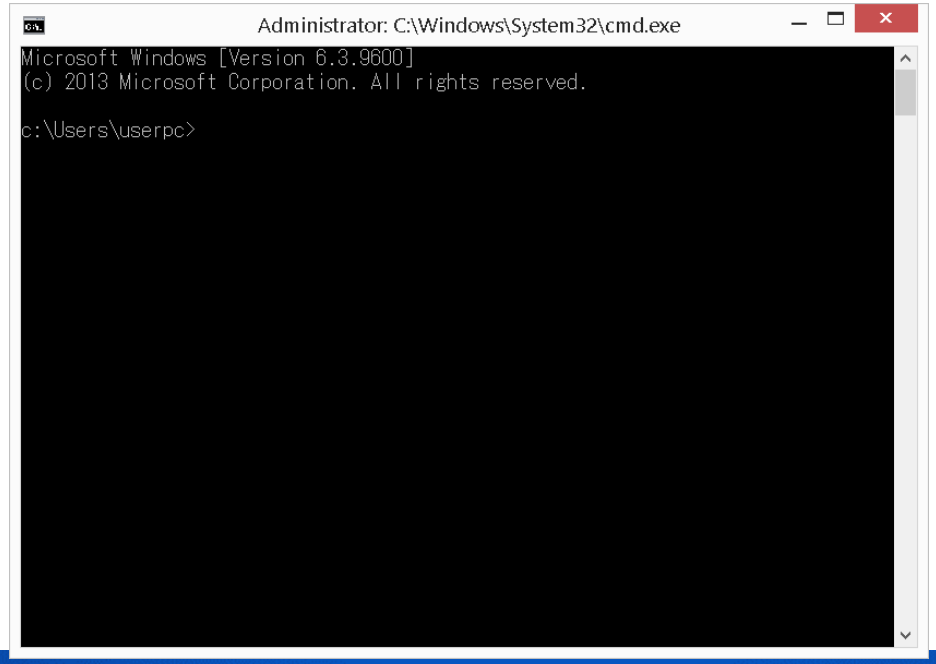
# 元大Smart API架構說明

- **行情API**: 行情API是透過下單機取得使用者設定的**商品代號**後，向行情主機取得行情資料，寫出一個記載著行情的文字檔至指定路徑，當我們的交易程式需要取得行情時，就用讀取該文字檔的方式取得即時的成交、委託行情，完成行情API的取用。
- **下單API**: 下單API則是透過事先封裝好的執行檔(.exe)是執行下單的動作，舉例，若執行Order.exe TXFD8 B 10800 1 LMT IOC 1指令，透過呼叫Order.exe執行檔的同時傳入相對應的參數，就可以執行:商品代號TXFD8的商品下出10800的1口限價買單，另外還有市下單、刪單、委託查詢、未平倉查詢、帳務查詢...等等指令，也都有相對應的執行檔。



# 先了解Windows Command line簡易操作

- 開始功能表搜尋程式欄位輸入cmd，即可看到上方的cmd.exe執行檔，點選並執行。
- 以「元大Smart API資料夾」放在桌面為範例，依序輸入:
- 切換到元大Smart API資料夾下:「**> cd Desktop\元大Smart API**」
- 「**> dir**」查看路徑下檔案，確認**Order.exe**、**CancelALL.exe…**等等執行檔確實存在該路徑下便可接續以下操作。



# 以「行情取用」為範例

## 行情取用 (tail.exe)

觀看教學影片

行情的取用需透過tail.exe執行檔，此執行檔是仿照Linux shell的tail指令製作而成，搭配傳入檔名的參數即可取用該檔案的最後的第n行，若傳入1則可以取得該檔案最後一行的數值(也就是最新的成交資料)，以下以檔案路徑放置在D:\data為例，輸入:

```
tail.exe -n1 D:\data\20180409\TXFD8\20180409_Match.txt
```

即可取得2018/3/5當天的最新一筆成交資料，執行畫面如下:

```
C:\Users\90813\Desktop\元大SmartAPI>tail.exe -n1 D:\data\20180409\TXFD8\20180409
_Match.txt
TXFD8,11:28:51.212000,10879,1,108247,10898,10813
```

日期_Match.txt，表示成交價量揭示訊息。
資料欄位：

商品代號,時間,成交價,成交量,總量,最高價,最低價

若想要知道最新一筆的上下5檔報價，也可以透過:

```
tail.exe -n1 D:\data\20180409\TXFD8\20180409_UpDn5.txt
```

取得上下五檔的報價資訊如下:

```
C:\Users\90813\Desktop\元大SmartAPI>tail.exe -n1 D:\data\20180409\TXFD8\20180409
_UpDn5.txt
TXFD8,11:30:11.395000,10881,4,10880,38,10879,38,10878,52,10877,35,10882,29,10883
,65,10884,64,10885,120,10886,39
```

日期_UpDn5.txt，表示最佳上下五檔價量資訊。
資料欄位：

商品代號,時間,下一檔價格,下一檔數量,下兩檔價格,下兩檔數量,下三檔價格,下三檔數量,下四檔價格,下四檔數量,下五檔價格,下五檔數量,上一檔價格,上一檔數量,上兩檔價格,上兩檔數量,上三檔價格,上三檔數量,上四檔價格,上四檔數量,上五檔價格,上五檔數量

# R呼叫外部指令

- 在R中可透過內建函數**system()**或是**system2()**去呼叫外部指令。
- 語法:

```
> system2("路徑+檔名", args="傳入參數", stdout=T)
```

  - 執行檔 (路徑+檔名) [必要]
  - **args**: 呼叫執行檔時傳入的參數[非必要]
  - **stdout**: 是否要回傳執行後的結果[非必要]

- 例子: 在command line介面呼叫開啟記事本:

```
> system2("notepad.exe", wait = F, invisible = F)
```

  - **wait=F**: 讓進程直接跳離呼叫出來的程式。
  - **invisible=F**: 則是讓視窗顯示出來，執行後可以看到記事本自動打開。

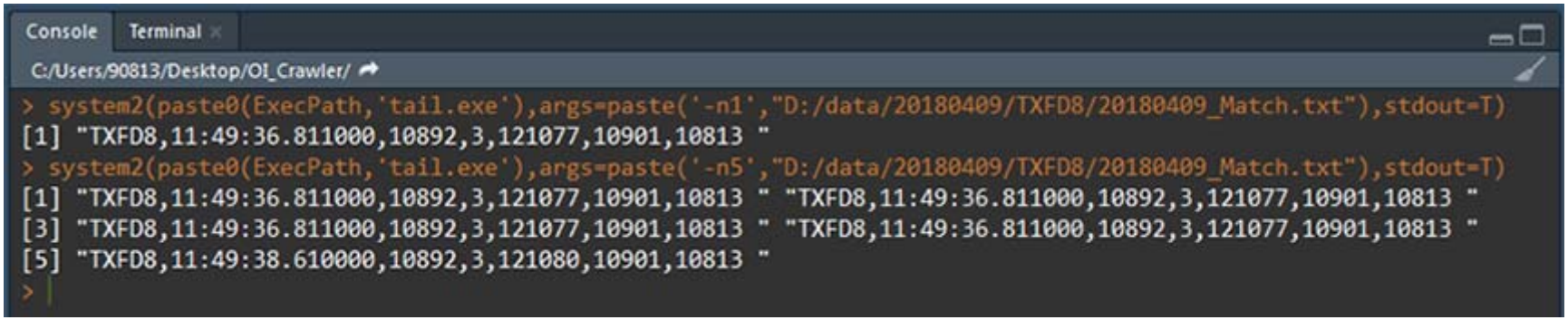
# R + 元大Smart API 基礎操作

- 先把元大Smart API資料夾下所包含的執行檔路徑先儲存在一個字串變數中，以供後續指令調用，以資料夾放置在桌面為範例，指令為:

```
ExecPath = "C:/Users/90813/Desktop/元大Smart API/"
```

- **行情取用**

```
> #取得最後1行
> system2(paste0(ExecPath,'tail.exe'),
     args=paste('-n1',"D:/data/20180409/TXFD8/20180409_Match.txt"), stdout=T)
> #取得最後5行
> system2(paste0(ExecPath,'tail.exe'),
    args=paste('-n5',"D:/data/20180409/TXFD8/20180409_Match.txt"), stdout=T)
```




**其它下單，切換商品，刪單使用範例:**

https://www.yuantafutures.com.tw/file-repository/content/smartAPI/page7-1.html#01