COMP226: Slides 11

Equity curves

Rahul Savani

rahul.savani@liverpool.ac.uk

Overview

- We consider **equity curves**: time series plots that show the performance of a **sequence of trades**
- First we introduce **position vectors**
- The we create equity curves by **combining position vectors with prices/returns**

Position (signal) vectors

Let k be an integer (we can only trade a whole number of lots)

Signal	Position
-k < 0	Short k lots
0	Flat
k > 0	Long k lots

- By signals, we mean **time series of positions**
- **Trades occur when the value of the signal changes**
- Sometimes (see note below) we just consider trading one lots

  then the signals are 1, -1 or 0

Aggregating returns

Given a sequence of **simple returns**, they are aggregated by:

- **adding 1** to each of them, **multiplying them together**, and then **subtracting 1 from the product**

Given a sequence of **log returns**, they are aggregated by:

- **adding them up**

This produces the **cumulative return**. Note that in R we use the functions `cumsum` and `cumprod`, where `cum` stands for cumulative or cumulate.

We will see loads of examples of this later

Note

The methods we have just described assume that each time we place a new trade **all capital is reinvested**.

In reality, this is at best only approximately true due to the fact one can generally only buy whole units (of shares or contracts) and additionally there are trading costs such as commissions. For simplicity, we ignore these issues.

For simple profit and loss, as opposed to returns, it is easier to account for injecting or removing capital, or trading costs, when computing an equity curve. However, returns are more usual for reporting performance.

Equity curves

- **Time series plot** of **profitability of a trading strategy**
- Either in terms of profit/loss or return (log or simple)

```r
# Profit/loss:
# multiply differences by positions and sum:
equity_diff <- cumsum(simple_diff * pos)

# Log returns:
# multiply log returns by positions and sum:
equity_log <- cumsum(log_ret * pos)

# Convert log returns to simple returns:
equity_simple_from_log <- exp(equity_log) - 1
```

Equity curves

Simple returns: to directly compute an equity curve with simple returns, things are a bit more complicated:

- apply simple return formula for short trades
- set the return for flat trades as zero
- add 1 to the returns, take their product, then subtract 1

```
simple_ret[pos==-1] <- -simple_ret[pos==-1]/(simple_ret[pos==-1]+1)
simple_ret[pos==0] <- 0

equity_simple <- cumprod(1 + simple_ret) - 1
```

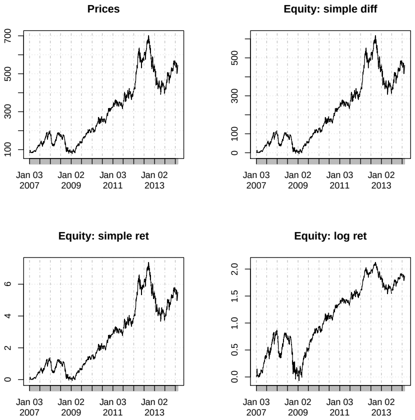
Gives same answer as going via log returns

Comparing results

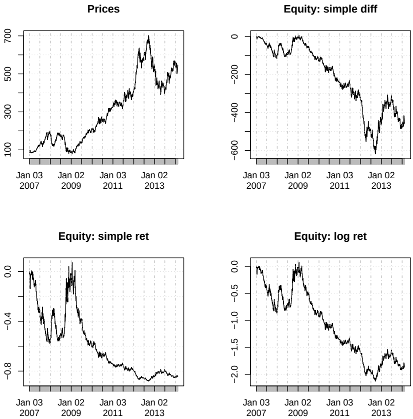
For the three different position vectors, buy and hold (all one), sell and hold (all -1), and randomly chosen (from -1,0,1), we plot the three equity curves on one graph

```
> par(mfrow = c(2,2)) # setup 2 by 2 grid for plots
> plot(prices)
> plot(equity_diff)
> plot(equity_simple)
> plot(equity_log)
```

Example: Buy and Hold



Example: Sell (and Hold)



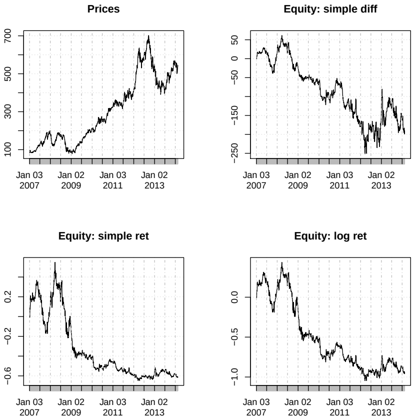
Example: Random Positions

- We generate an artifical position vector
- We sample uniformly at random from

$$\{-1,\ 0,\ 1\}$$

```
> len <- length(prices)
> pos <- sample(c(-1,0,1),size=len,replace=TRUE)
> head(pos)
[1] -1 -1 -1 -1  0 -1
```

Example: Random Positions



Perfect profit/return

- Perfect profit/return is computed via the **perfect position**, i.e., going **long** when the market goes **up** and **short** when it goes **down**

```r
ppos <- ifelse(returns > 0,1,-1)
```

- They naturally can be incorporated into performance measures:

$$\frac{FinalCumulativeProfit}{PerfectProfit}$$

$$\frac{FinalCumulativeReturn}{PerfectReturn}$$

```
> cbind(returns,ppos)
             ..1 ..2
2000-01-01  0.01   1
2000-02-01  0.02   1
2000-03-01 -0.10  -1
2000-04-01  0.05   1
2000-05-01  0.01   1
2000-06-01 -0.02  -1
2000-07-01  0.03   1
2000-08-01 -0.02  -1
2000-09-01  0.09   1
2000-10-01  0.06   1
2000-11-01 -0.10  -1
2000-12-01  0.05   1
2001-01-01 -0.06  -1
2001-02-01  0.20   1
```

Example

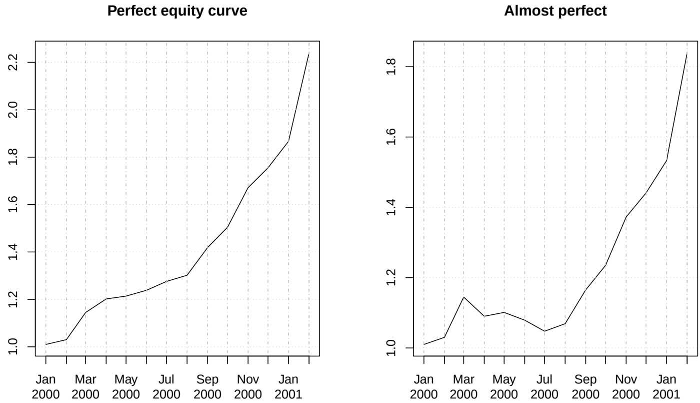
```
preturns <- returns
preturns[ppos==-1] <- -preturns[ppos==-1]/(preturns[ppos==-1]+1)
preturn <- prod(1 + preturns) - 1

pos <- ppos
pos[c(4,6,7)] <- -1 * pos[c(4,6,7)]  # swap some positions

mreturns <- returns
mreturns[pos==-1] <- -mreturns[pos==-1]/(mreturns[pos==-1]+1)
mreturn <- prod(1 + mreturns) - 1

> mreturn
[1] 0.8395281
> preturn
[1] 1.240306
> mreturn/preturn
[1] 0.6768718 # we achieved about 68% of possible returns
```

Example: Perfect return



Simple strategy: Copycat

Now we use a more complicated rule to generate a position vector:

Copycat strategy

- If the close is higher than the open, buy the next day
- Otherwise, sell the next day

- Trys (**stupidly**) to capture **momentum** in price moves
- We call it **copycat** to indicate that it copies what would have worked on the previous day

When would this strategy work?

If there are

consecutive days where price moves in the same direction

and

when the price move direction differs from the previous day's (i.e. we lose), the loss should not be too large, that is:

losing days should not wipe out the gains from (possibly multiple) winning days

Testing the strategy

From now on, we will repeatedly use functions from: 'utilities.R'.

In this case we will use:

- `getLogReturns(prices)` which **computes log returns from adjusted prices**
- `getEquityLog(log_ret,pos)` which **computes a simple returns equity curve** from log returns and a position vector

Utility functions

```
getLogReturns <- function(prices) {
    # returns log returns of Adjusted prices
    # assumes Adjusted price column exists in input
    log_ret <- ROC(Ad(prices),type='continuous')
    log_ret[1] <- 0
    return(log_ret)
}
```

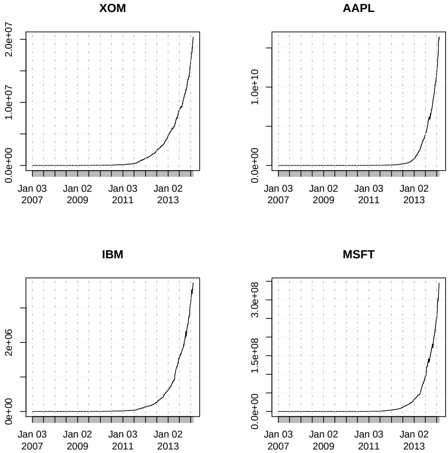
```
getEquityCurve <- function(returns,pos) {
    # takes log returns and pos vector
    # returns equity curve in simple returns
    return(exp(cumsum(returns*pos)) - 1)
}
```

```r
source('utilities.R') # our utilities script
library(quantmod)
tickers <- c("XOM", "AAPL", "IBM", "MSFT")
getSymbols(tickers) # load all tickers in one go

pdf('pdf/pp.pdf') # write figures to pdf file
par(mfrow=c(2,2)) # setup 2x2 grid for equity curves

for (ticker in tickers) {
    prices <- get(ticker)
    # now comes the "strategy logic"
    pos <- ifelse(Cl(prices)-Op(prices)>0,1,-1)
    equity_log <- getEquityLog(getLogReturns(prices),pos)
    print(plot(equity_log,main=ticker))
}
dev.off() # close pdf
```

Equity curves



The problem? Look-ahead bias

These equity curves look **too good to be true** and indeed are:

```
pos <- ifelse(Cl(prices)-Op(prices)>0,1,-1)
equity_log<-cumsum(log_ret*pos))
```

The position **today** uses price data from **today**

This is **look-ahead bias**:

When we trade **we cannot know the close before it happens!**

Warning

When backtesting trading strategies with historical data, **we must avoid look-ahead bias**

The strategy **must only use information that would be available at the time of a trading decision**

In this example the bias **was obvious**, however, if you included "today's" data in a moving average, it might be less obvious, but still could easily improve the strategy performance and would be completely unrealsitic!

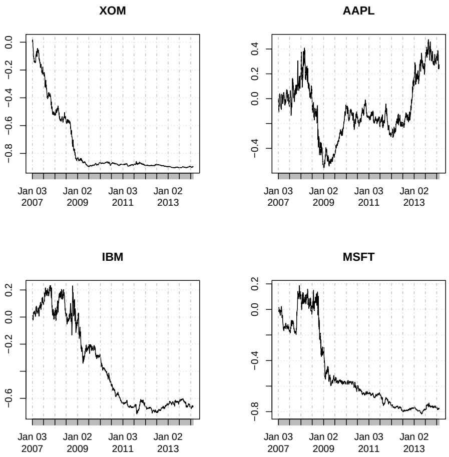
Let's correct it

We use the `lag` function to shift the time series by one position

Thus we **use the previous day's price to make decisions**

```
pos <- ifelse(Cl(prices)-Op(prices)>0,1,-1)
pos <- lag(pos)
pos[1] <- 0
```

Correct equity curves



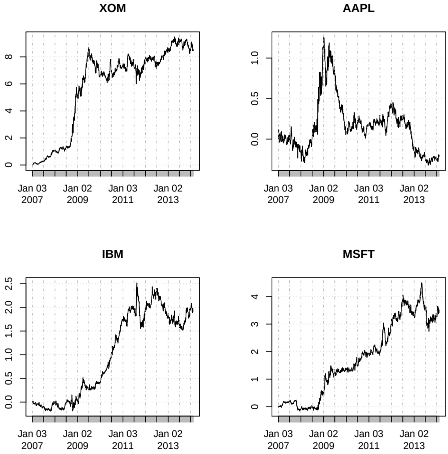
Switch postions

Notice that this strategy does badly on all four tickers, but particularly bad on three of them

That suggests an obvious change to the strategy: **do exactly the opposite**, that is swap long and short trades as follows

```
pos <- ifelse(Cl(prices)-Op(prices)>0,1,-1)
pos <- lag(pos)
pos[1] <- 0
pos <- pos * -1 # here we do the switch
```

Switched equity curves



Warning

We have not included **slippage** in any of these backtests

Slippage is one reason that **bad strategies cannot always be transformed into good ones** by "switching" positions

We will see this when we start to include slippage in our backtests