Data Snooping in the Stock Market

The Second International School on Actuarial and Financial Mathematics
Kyiv Ukraine
10th of June 1999

Thomas Hellström
University of Umeå
Sweden

http://www.cs.umu.se/~thomash/
email: thomash@cs.umu.se

Thomas Hellström

- "Industrial" background:
- Ionospheric research at EISCAT
- Product development in my own company Seapacer AB
  - Optimisation and Control computers for ferries
  - Real-time data analysis
- Teaching Artificial Intelligence at
  the Department of Computing Science,
  Umeå University, Sweden
- Research interests:
  Stock predictions, Computational intelligence

Contents

What's so Special about Predictions of Stock Time Series?

⊖

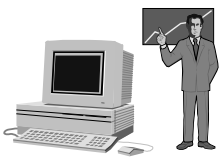
✔ A hard problem! Is it even possible?
✔ Looks very much like a random walk!
✔ The process is "regime shifting". The markets move in and out of periods of "turbulence", "hause" and ''baise''. Hard for traditional algorithms!
✔ The evaluation of predictability is extremely hard!
When have we learned and when have we memorized?

⊕

✔ A successful prediction algorithm does not have to provide predictions for all points in the time series.
Can we predict predictability?

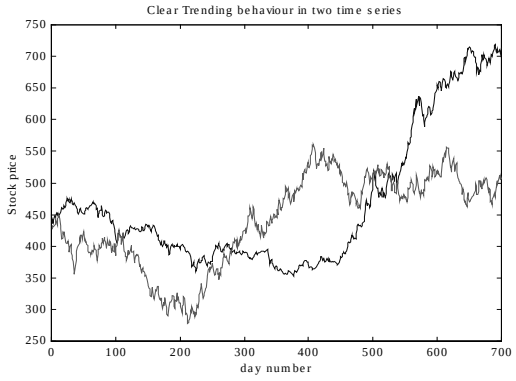
Two viewpoints

The Efficient Market Hypothesis:
"**The price reflects all available information. Prediction is therefore impossible!**"
You might as well look into a crystal ball!"

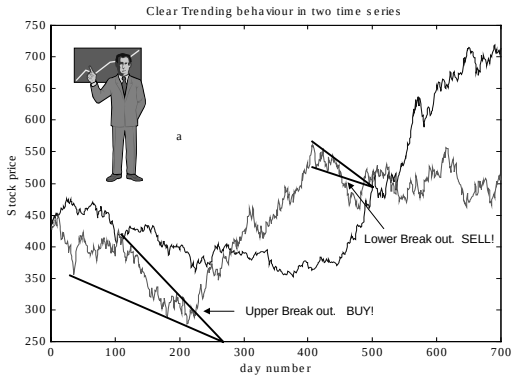


The Market Professional:
"**By utilizing our advanced methods we can predict much better than our competitors**"

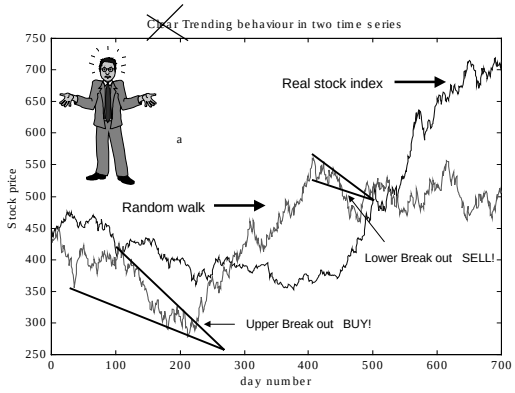
What Does the Data Look Like?



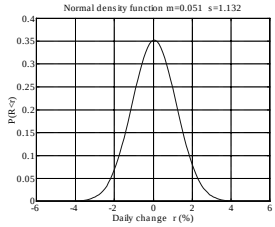
Technical Analysis: Triangles



Ooops!



Does the Dow Jones index Follow a Random Walk?



- Normal distribution is a consequence of pure random walk. Statistics for daily changes
  Dow Jones 1984-1996:
  Mean=0.05% Std. dev.=1.1
- Question:
  How often can we expect a crash like November 1987 (-28% in one day) ?

r	P(R<r)	Years between events	No. of real obs.
0	5.00E-01	0	1063
-1	2.00E-01	0	201
-2	4.00E-02	0	56
-3	4.00E-02	1	19
-4	2.00E-04	23	9
-5	4.00E-06	982	3
-6	5.00E-08	88244	3
-7	3.00E-10	20,000,000	1
-8	6.00E-13	7000,000,000	1
-9	7.00E-16	6000,000,000,000	1

Data Available in Technical Analysis

Only historical price information:

For each day:

- **"Close" : Last paid**
- **"High" : Highest paid**
- **"Low" : Lowest paid**
- **"Volume" : Number of traded stocks**

Derived entities:

- **"Return" : (Close(T) - Close(T-1)) / Close(T)**
- **"Volatilitet": standard deviation for "Return" in a window backwards**

Chart of Technical Analysis

Derived Entities

✔ **k-day Returns:**

$$R_k(t) = \frac{Close(t) - Close(t-k)}{Close(t-k)} \approx \log\left(\frac{Close(t)}{Close(t-k)}\right)$$

✔ **Moving average of order k:**

$$mav_{y,k}(t) \equiv \frac{1}{k}\sum_{i=0}^{k-1} y(t-i)$$

The time series *y* can be, for example, Close, High, Low or Volume

Derived Entities

✔ **Volatility (standard deviation of the log returns) :**

$$V = \sqrt{\frac{1}{N-1}\sum_{t=1}^{N} ln\left(\frac{Close(t)}{Close(t-1)}\right) - m}$$

where

$$m = \sqrt{\frac{1}{N}\sum_{t=1}^{N} ln\left(\frac{Close(t)}{Close(t-1)}\right)}$$

Data in Fundamental Analysis

1) The general economy
- **Inflation**
- **Interest rates**
- **Trade balance etc.**

2) The condition of the industry
- **Other stocks' prices, normally presented as indexes**
- **The prices of related commodities such as oil, metal prices, and currencies**
- **The value on competitor stocks**

Data in Fundamental Analysis

3) The condition of the company

- **p/e: Stock price divided by last 12 months earning per share**
- **Book value per share: Net assets (assets minus liabilities) divided by total number of shares**
- **Net profit margin: Net income divided by total sales**
- **Debt ratio: Liabilities divided by total assets**
- **Prognoses of future profits**
- **Prognoses of future sales**

Two Formulations of the Prediction Task

Methods with a fixed prediction horizon

- "The Time Series Approach"

- "The Trading Rule Approach"

Problem:
We don't necessarily intend to sell the stocks *h* days after we bought them

No fixed prediction horizon:

- Simulated trading with buy- and sell-rules

Problem:
Fewer points gives lower statistical significance

The Time Series Approach

Detrend the prices by computing "returns":

$$y(t) = \frac{Close(t) - Close(t-k)}{Close(t-k)}$$

Find a function g:

$$g(y(t), y(t-1),\ldots,y(t-k)) \approx y(t+h)$$

Minimize the RMSE:

$$P_h(t) = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(g(t) - y(t+h))^2}$$

Example of "Bias" for the Choice of the Function g:

Linear AR model:

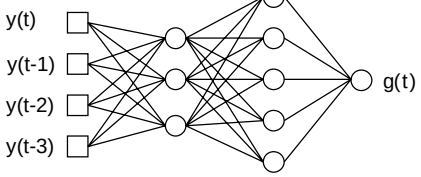
$$\mathbf{y(t+h) \approx \sum_{i=0}^{k} W_i\, y(t-i)}$$

Nonlinear neural network:

$$\mathbf{y(t+h) \approx \varphi\left(\sum_j W_j \varphi\left(\sum_{i=0}^{k} W_{i,j}\, y(t-i)\right)\right)}$$

Feed-Forward Neural Network

- ✔ Input layer with 4 inputs
- ✔ Two Hidden layers with 3 and 5 nodes
- ✔ Output layer with 1 output node

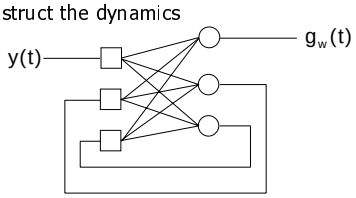


The weights w are computed to minimize

$$|E| = \sqrt{\frac{1}{N-4}\sum_{t=4}^{N-1}(g_w(t) - y(t+1))^2}$$

Recurrent Neural Network

- ✔ Feedback to input layer
- ✔ The hidden layer stores previous values and can reconstruct the dynamics



The weights w are computed to minimize:

$$|E| = \sqrt{\frac{1}{N}\sum_{t=1}^{N-1}(g_w(t) - y(t+1))^2}$$

Drawbacks of the Time Series Approach

The RMSE measure and the profit made by applying the prediction algorithm have different maxima.

- **The RMSE treats all predictions, small as large, equally**
- **The RMSE penalizes a large change in the same direction as the predicted change**

The Trading Rule Approach:

$$T(t) = \begin{cases} \text{Buy} & : g(\mathbf{X}(t)) > 0 \\ \text{Sell} & : g(\mathbf{X}(t)) < 0 \\ \text{Do nothing} & : \text{otherwise} \end{cases}$$

$$\mathbf{X}(t) = (R_1(t), \ldots, R_N(t))$$

X can be:
Past values of C,H,L,V
or derived entities: Volatility, Trend, ...

Learning Task:
Find a function g that gives the best performance at a fixed prediction horizon OR
when applying the trading rule T
Drawback:
Statistical significance;
The Buy and Sell signals are << N

Technical Indicators

- ✔ **The tools for Technical trading include principles such as:**
  - The trending nature of prices
  - Volume mirroring changes in price
  - Support/Resistance
- ✔ **Examples:**
  - Moving averages
  - Formations such as triangles
  - RSI - the relation between the average upward price change and the average downward price change within a time window normally 14 days backwards

Example of a Technical Indicator

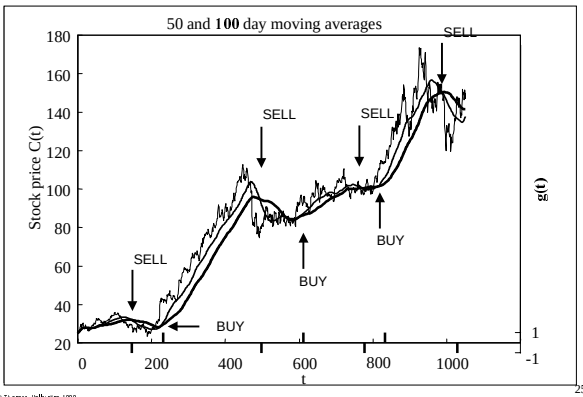
$$T(t) = \begin{cases} \text{Buy} & : g(\mathbf{X}(t)) > 0 \\ \text{Sell} & : g(\mathbf{X}(t)) < 0 \\ \text{Do nothing} & : \text{otherwise} \end{cases}$$

$$g \equiv \Delta(\text{ sign}(mav_{C,50}(t) - mav_{C,100}(t))\ )$$

$$\Delta v(t) \equiv v(t) - v(t-1)$$

$$mav_{C,k}(t) \equiv \frac{1}{k}\sum_{i=0}^{k-1}\text{Close}(t-i)$$

$$g \equiv \Delta(\ \mathrm{sign}(\mathrm{mav}_{C,50}(t) - \mathrm{mav}_{C,100}(t))\ )$$



Benchmarks

✔ **Naive prediction of stock prices:**
Close'(t)=Close(t-1)

✔ **Naive prediction of returns:**

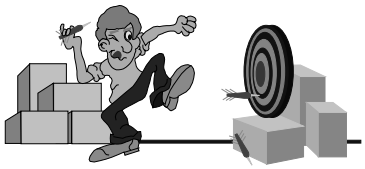
$R'(t) = R(t-1)$

The naive predictors are local minimum in many models e.g AR-models (but also Neural Networks):

$$R'(t) = \sum_{i=1}^{K} a_i R(t-i)$$

✔ **Buy and hold:**
Buy at day 1 and sell at day N.
For multi-stock predictions or portfolio management:
Buy and hold of a index (Dow Jones, FTSE, DAX etc.)

Another Relevant Benchmark



How often is this guy as successful as we are?

Performance Measures

✔ **Theil coefficient:**
Compares the RMSE (root mean square error) for our predictions with the naive price predictions

Predicting {Close(t), t=1,N} with {Close'(t), t=1,N}

$$T = \frac{\sqrt{\sum_{i=1}^{N} (\ \mathrm{Close}(t) - \mathrm{Close'}\ (t)\ )^2}}{\sqrt{\sum_{i=1}^{N} (\ \mathrm{Close}(t)\ - \mathrm{Close}(t\ -1)\ )^2}}$$

T<1 for real predictive power

Performance Measures

✔ **Directional prediction "Hit rate"**

Predicting {R(t), t=1,N} with {R'(t), t=1,N}

$$H = \frac{\left|\left\{ t \,|R(t)R'\ (t) > 0,\ t = 1, N \right\}\right|}{\left|\left\{ t \,|R(t)R'\ (t) \neq 0,\ t = 1, N \right\}\right|}$$

For the naive return predictor:

$$H_N = \frac{\left|\left\{ t \,|R(t)R(t\ -1) > 0,\ t = 1, N \right\}\right|}{\left|\left\{ t \,|R(t)R(t\ -1) \neq 0,\ t = 1, N \right\}\right|}$$

✔ **Normalized hit rate:**

$$H_0 = \frac{H}{H_N}$$

$H_0 > 1$ for real predictive power

Performance Measures

Mean profit per trading day:

✔ Fixed horizon predictions C'(t) of the close price C(t).
A trade is assumed at every time step, in the direction of the predicted change:

$$100\,\frac{1}{h}\,\frac{1}{N-h}\,\sum_{t=h+1}^{N} \mathrm{sign}(C'(t)\text{-}C(t\text{-}h))\,\frac{(C(t)\text{-}C(t\text{-}h))}{C(t-h)}$$

Benchmark: Mean daily return for a Buy and Hold strategy

Mean profit per year:

✔ Trading simulation:
– "Run" the trading and compute the mean profit
Benchmark: Annual returns for Buy and Hold on index

What Is a Reasonable Goal?

- ✔ Efficient market hypothesis implies random walk, which is impossible to predict!
- ✔ Published research (with proper evaluation) often shows about **54%** hit rate. I.e: correct prediction of the sign of the future return y(t+h).
- ✔ Even 54% real hit rate is enough to make a fortune!
- ✔ Compare with a casino: They don't know what number comes up next, they just improve the odds by adding the 0 and 00.

Data Snooping with the Time Series Approach

- ✔ We are predicting a stock with equal numbers of moves up and down during one year of 250 trading days.
- ✔ Apply a totally random prediction algorithm on each day
- ✔ What is the probability that the hit rate>54% ?

The distribution for number of hits is given by:

$$P(H = x) = \binom{250}{x} 0.5^{x}\ 0.5^{250-x}$$

$$P(H > x) = 1 - P(H \le x) = 1 - \text{binocdf}(x,250,0.5)$$

x=0.54*250=135 gives P(H>135)=0.092

I.e.There is a 9% risk that a random algorithm gives 54% hit rate.

Data Snooping with the Trading Rule Approach

- ✔ We want to compare 100 indicators, each producing Sell and Buy signals once a week on average. The test period is 10 years! We demand 55% hit rate!
- ✔ Apply 100 totally random indicators on each week of data.
- ✔ The probability that one specific indicator gets more than x hits is:

$$P(H > x) = 1 - P(H \le x) = 1 - \text{binocdf}(x,500,0.5)$$

P(H>0.55*500)=0.0112

The probability that ANY one of the 100 indicators produces 55% hit rate is 1-minus the probability that all are less than 55%:

$$1 - (1 - 0.0112)^{100} = 0.68$$

I.e: The probability for a type II error (accepting a false hypothesis) is 68%.

Evaluating Performance

Algorithm evaluation is part of the learning process because it involves searching and selecting the best performing algorithm.

Therefore:

- ✔ It must be done "in sample" and not on the test set. Best: A final test on data that didn't exist at the time of the development of the algorithm
- ✔ It is sensitive to "over training".
- ✔ Be aware of the data-snooping problem!

Some Guide Lines for Developing Prediction Algorithms

- ✔ **Ensure high data quality. Watch out for outliers**
- ✔ **Handle missing data correctly**
- ✔ **When predicting one-step ahead, make sure this is what you are really doing. "off-by-one" errors are seldom more fatal than in financial predictions.**

Some Guide Lines for Evaluating Prediction Algorithms

- ✔ **Evaluate on previously non-used data!**
- ✔ **Use a lot of data; many stocks and long time periods**
- ✔ **Compute annual performance results**
- ✔ **Test the algorithm on random-walk data**
- ✔ **For comparison: test a random algorithm on the data**