Can ETF Arbitrage be extended to sector trading?

An experimental analysis

Ramnik Arora & Utkarsh Upadhyay

Indian Institute of Technology, Kanpur

Outline

Standard ETF Arbitrage

ETF: Exchange Traded Fund

A security that tracks an index and represents a basket of stock like an Index fund but trades like a stock on an exchange

- Index and component securities can be exchanged
- However, only authorized dealers can effect this conversion

Standard ETF Arbitrage

ETF: Exchange Traded Fund

A security that tracks an index and represents a basket of stock like an Index fund but trades like a stock on an exchange

- Index and component securities can be exchanged
- However, only authorized dealers can effect this conversion

Pricing relationships

- Index price is linearly related to stock prices

$$y = \sum_{i=1}^{i=N} \alpha_i \, x_i$$

where:

y	is the price of the index
N	is the number of stocks involved
x_i	is the price of the i^{th} stock
α_i	is the component of the i^{th} stock in the index

- This relationship remains fixed

Arbitrage Technique

- Arbitrage is a riskless profit strategy
- *Triggering conditions:*

Index price $<$ calc. price	Buy index, exchange with stocks
Index price $>$ calc. price	Buy stock, exchange with index

Pair trading

Pair trading:

Trading strategy of simultaneously buying a particular security and selling a related security against it.

Assumption: Homogenous effect of news

Potentially highly correlated pairs:

Pepsi	Coca Cola
Dell	Hewlett-Packard
Dow Jones	S&P 500

Outline

1. Introduction
2. New Strategy
3. Empirical Analysis
4. Further work

Why not mirror *ETF Arbitrage* for sector trading?

Same assumptions: Homogenous effect of *general* news.

- One security priced against many others:
  - Target price would be linearly related to stock prices

$$y = \sum_{i=1}^{i=N} \alpha_i \, x_i$$

where:

y	is the price of the target security
N	is the number of stocks involved
x_i	is the price of the i^{th} stock
α_i	is the component of the i^{th} stock in the target security

- Can be easily automated
- Does not require special privileges

Why not mirror *ETF Arbitrage* for sector trading?

Same assumptions: Homogenous effect of *general* news.

- One security priced against many others:
  - Target price would be linearly related to stock prices

$$y = \sum_{i=1}^{i=N} \alpha_i \, x_i$$

where:

y	is the price of the target security
N	is the number of stocks involved
x_i	is the price of the i^{th} stock
α_i	is the component of the i^{th} stock in the target security

- Can be easily automated
- Does not require special privileges

Artificial Neural Networks

- **Non-parametric** data specific regression
- Detect multi-dimensional non-linear relationship between stock prices
- However, the relationship will need to be linearly approximated
- Assumption: The linear relationship holds for the period of the trade

Artificial Neural Networks

- **Non-parametric** data specific regression
- Detect multi-dimensional non-linear relationship between stock prices
- However, the relationship will need to be linearly approximated
- Assumption: The linear relationship holds for the period of the trade

Non parametric regression

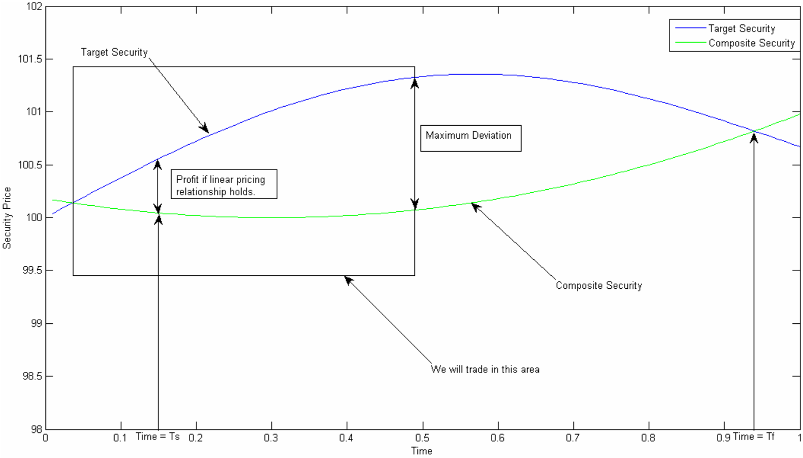
- Market data is complex
- Limitations of pameterized regressors
- Can be seen as a black box which:
  - Uses initial data to train itself
  - Predicts the value of a function at new points

When to buy/sell?

Buying or selling of target security depends on the predicted price. Like ETF Arbitrage.

- *Buy* when the target security is underpriced.
- *Sell* when the target security is overpriced.
- The underlyings are traded in the ratio as specified by the *pricing relationship*

Profit/Loss analysis



At time t_s

Transactions committed at the start of a trade:

- Short the target security (overpriced)
- Long the composite security (underpriced)

Our current account is:

$$A(t_s) = -\lambda_1(t_s)S_1(t_s) - \lambda_2(t_s)S_2(t_s) + S_{target}(t_s)$$

$A(t)$	the investment at time t
$S_{target}(t)$	the value of the target security
$\lambda_i(t)$	the weight of the i^{th} security
$S_i(t)$	the price of the i^{th} security

At time t_s

Transactions committed at the start of a trade:

- Short the target security (overpriced)
- Long the composite security (underpriced)

Our current account is:

$$A(t_s) = -\lambda_1(t_s)S_1(t_s) - \lambda_2(t_s)S_2(t_s) + S_{target}(t_s)$$

$A(t)$	the investment at time t
$S_{target}(t)$	the value of the target security
$\lambda_i(t)$	the weight of the i^{th} security
$S_i(t)$	the price of the i^{th} security

At time t_f

Transactions committed at closing of a trade:

- Cover the target security
- Sell the composite security

Our current account is:

$$A(t_f) = \lambda_1(t_f)S_1(t_f) + \lambda_2(t_f)S_2(t_f) - S_{target}(t_f)$$

$A(t)$	the investment at time t
$S_{target}(t)$	the value of the target security
$\lambda_i(t)$	the weight of the i^{th} security
$S_i(t)$	the price of the i^{th} security

Analysis

$$Total\ cash\ flow = A(t_s) + A(t_f)$$

but prices converge at t_f, therefore:

$$A(t_f) = 0 \Rightarrow Total\ cash\ flow = A(t_s) > 0$$

However, calculating outstanding stock value at t_f:

$$[\lambda_1(t_s) - \lambda_1(t_f)]\, S_1(t_f) + [\lambda_2(t_s) - \lambda_2(t_f)]\, S_2(t_f)$$

Analysis

$$Total\ cash\ flow = A(t_s) + A(t_f)$$

but prices converge at t_f, therefore:

$$A(t_f) = 0 \Rightarrow Total\ cash\ flow = A(t_s) > 0$$

However, calculating outstanding stock value at t_f:

$$[\lambda_1(t_s) - \lambda_1(t_f)]\, S_1(t_f) + [\lambda_2(t_s) - \lambda_2(t_f)]\, S_2(t_f)$$

Robustness of Linear Approximation

If the linear relationship holds for the entire duration of the trade:

$$\lambda_i(t_s) = \lambda_i(t_f),\ i \in \{1, 2\}$$

And,

$$Profit = A(t_s) > 0$$

Since the relationship does not always hold, it is only statistically profitable.

Outline

1. Introduction
2. New Strategy
3. Empirical Analysis
4. Further work

ANN Setup

4 different ANNs were trained.

- Training:
  - Data segments taken one day at a time
  - Trained on first 80% of data
- Testing:
  - Also treats one day as a unit
  - Tested on 20% data each day

That ANN was chosen which best fit the training data.

Training and testing

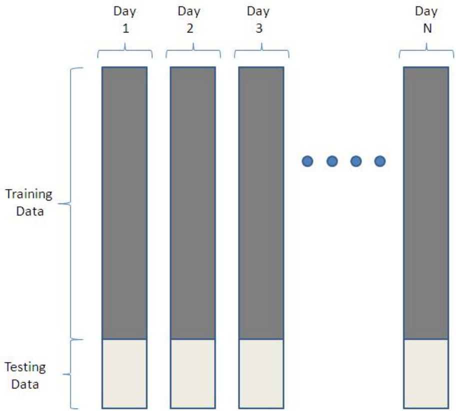


Figure: Segmentation of data for training and testing

Testing the hypothesis

For testing, Yahoo (*YHOO*) for the period *Sept. '05* was priced against:

AAPL	AMZN	CSCO	EBAY	GOOG
IBM	MSFT	NWSA	ORCL	TWX

Tests were done on two different models:

1. Open trade on 1.25σ and close on 0.75σ
2. Open trade on σ and close on 0.75σ

Results

Trades	5.25
Avg Profit/Trade	0.16
Percentage Profitable	82%
Max Profit	0.58
Max Loss	0.14
Average Trade Duration (sec)	17.81
Max Drawdown	0.26

Table: Average results for Model 2

Pros and Cons

- Pros:
  - Statistically profitable
  - Very short trade durations, perfect for automation
- Cons:
  - Commissions
  - Prone to excessive trading in volatile markets
  - Large number of parameter to tweak

Pros and Cons

- Pros:
  - Statistically profitable
  - Very short trade durations, perfect for automation
- Cons:
  - Commissions
  - Prone to excessive trading in volatile markets
  - Large number of parameter to tweak

Outline

Parameters of ANN

- Number of hidden layers
- Number of neurons in each layer
- Transfer functions
- Improving the derivative estimates

Plugging holes

- Stop loss conditions
- Avoiding excessive trading
- Continuous trading

Thank you

Ramnik Arora
Utkarsh Upadhyay
utkarshu@iitk.ac.in & ramnik@iitk.ac.in