High Frequency Trading

MSE 448 - Group 1

Thibaud Bruyelle
Computational Engineering
thibaudb@stanford.edu

Brian Lui
Computer Science
brianlui@stanford.edu

Tom Morvan
Computational Engineering
tommrvn@stanford.edu

Julius Stener
Computer Science
stenerj@stanford.edu

Stanford University

June 3, 2020

Table of contents

1. Introduction - High Frequency Trading

Goal

Build a fully automated trading strategy that executes large amount of trades based on sub-second data.

Requires:

- Vast amount of *granular* order book data.
- Algorithms that produce *trading signals* in split-seconds.

Maystreet Simulator provides an ideal framework for HFT strategy development.

1. Introduction - Maystreet Simulator

Maystreet Simulator

Full Market Replay order by order.

What's handled by the simulator:

- Real-time order book data retrieval
- Trading order management and execution with slippage

What's not handled by the simulator:

- Market activity fees (Trading fees, Equity Borrow fees...)
- Reaction of other agents to our trading activity
- Multiday simulation

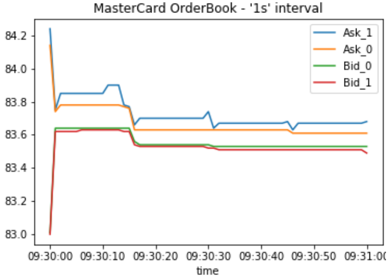
1. Introduction - Maystreet Simulator

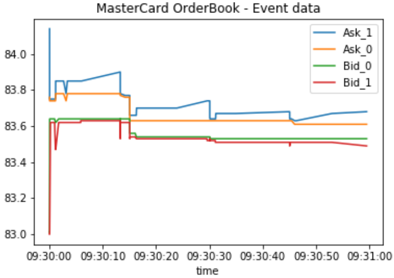
Algorithm Framework

Set parameters and start trading day

- At every time interval:
  - Retrieve Market Data
  - Update statistics
  - Send and/or Cancel a trade
- At every order message from exchange:
  - Check if our orders have been filled
  - Update portfolio

Market Open Data





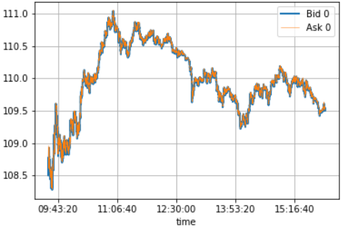
1. Introduction - Statistical studies

Which statistical tools to assess these data ?

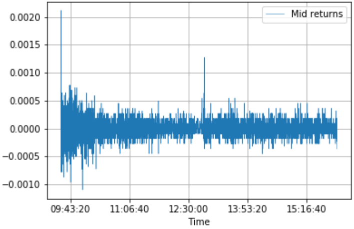
- Statistical tests (*Is the time serie stationary?*, *Are the returns normally distributed?*, *Are these two series cointegrated?*)
- Statiscal/Time Series Analysis (*Any correlation in a given portfolio?* , *Can we create clusters within this portfolio?*)
- Stochastic Modelling (*Can we estimate parameters that provide us relevant information?*)

1. Introduction - Price and Returns Series

- Bid and Ask series over a trading day
- Time interval can be very small (up to $1\mu s$)



AAPL Bid/Ask - 2015/01/21

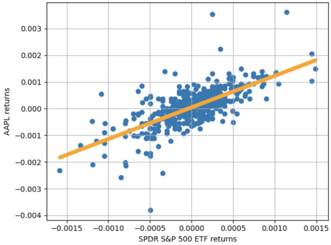


AAPL Mid returns - 2015/01/21

1. Introduction - Returns and market influence

- In order to make an analysis of the returns series R_t, we use SPDR S&P 500 returns M_t to compute the linear regression :

$$R_t = \beta M_t + \alpha + \epsilon_t$$



AAPL returns against Market

- We then used the residuals ϵ_t to do our analysis.

2. Mean Reversal - Statistical Principle

Mean Reversal relies on the assumption that longer term moving averages are more consistent than shorter term moving averages

- Most-Relevant Hyperparameters Available:
  - Short and Long Interval Periods
  - Time Intervals between Strategy Executions
  - Traded Universe
  - Data to be tracked for the "Mean"

Using a statistical modelling approach can be useful to consider some metrics that will bring us relevant information.

- Hurst exponent H
- The Hurst exponent, H, measures the long-term memory of a time series, characterising it as either mean-reverting, trending or a random walk. We have :

$$\mathbb{E}(\frac{R(n)}{S(n)}) = Cn^H$$

2. Mean Reversal - Statistical Tools

- The Ornstein–Uhlenbeck process X_t is defined by the following stochastic differential equation:

$$dX_t = \theta(\mu - X_t)dt + \sigma dW_t$$

- Useful model in a mean reversion context because the parameter θ is the speed of reversion.
- Thus, we can model the half-time which is the time it takes for the series to move half the distance towards its long term average :

$$T_{1/2} = \frac{ln(2)}{\theta}$$

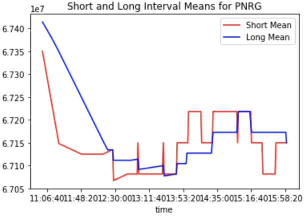
2. Mean Reversal - Strategy

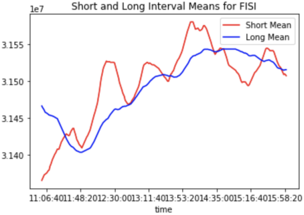
- Traded Universe
  - Utilized the Russell 2000 tickers
  - Chose Small Caps due to their relative stability against market trends due to lower trading volumes
  - Cleaned the Russell tickers for use on the Maystreet Simulator (resulted in < 1500 tickers)
- Type of "Mean"
  - Bid
  - Ask
  - Mid
    - Implemented due to the simplicity of tracking one set of means
    - Naively ignores the bid-ask spread cost when producing a trading signal

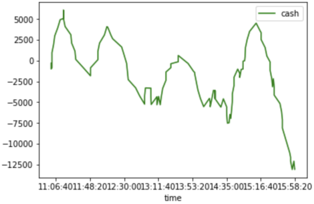
2. Mean Reversal - Results

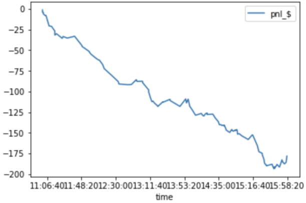
2018-07-16

~$10,000 traded with End of Day PNL at $(178.16)









2. Mean Reversal - Optimizations

2 Primary Methods we have identified as worth exploring

- Large-scale Grid Search
  - Due to the high-dimensionality of hyperparameters, the small-subset version of this will take $\tilde{6}$ days
  - 1420 tickers across $\tilde{2}00$ hyper-parameter configurations with multiple random trials to determine the most profitable/least risk-inclined strategy tickers
- Reinforcement Learning
  - This is a task for RL due the innate ability to learn state and action pairs across high-dimensionality parameters
  - Barrier to implementation: running out of time due to the lost week

- Portfolio : AAPL, ADOBE, BOA, CRM, GS, JPM, MA, MSFT, VISA.
- Residual Returns ϵ_t must be stationary to satisfy the assumptions of th PCA model.
- Augmented Dickey Fuller test assumes that :

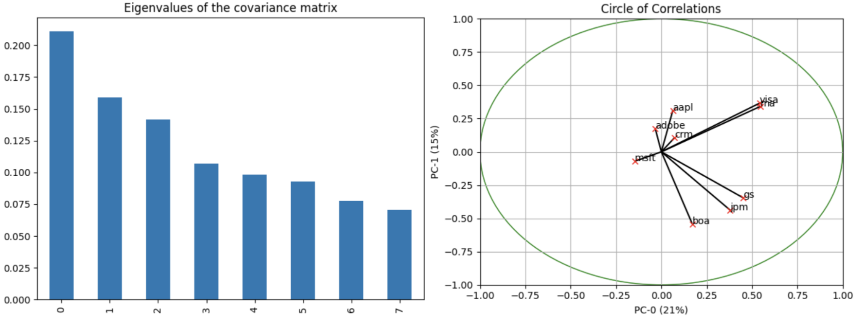
$$\epsilon_t = \rho \cdot \epsilon_{t-1} + u_t$$

  Test for the presence of a unit root, ie, $\mathcal{H}_0 : \rho = 1$

- *Conclusion* : For the returns series, we can always reject $\mathcal{H}_0$ at all confidence levels so the models assuming that our observations must be *iid* can be used.

Pairs Trading - Principal Component Analysis

- PCA to observe correlation patterns within the portfolio.



- *Conclusion* : We easily observe several trends and can notice a strong industry influence (technology, finance)

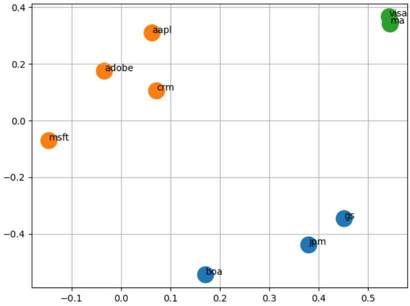
- For several reasons, we would like to study the association between two (or more) stocks (*e.g.* Pairs Trading)
- Correlation : when two securities move together in the same direction or opposite direction.
- Cointegration : when the distance between the pair doesn't change drastically over time.

Why studying cointegration ?

- Granger – Engel test to check if two series x_t, y_t are cointegrated. We must have $x_t, y_t \sim \mathcal{I}(1)$ and $y_t = \alpha + \beta x_t + u_t$ with $u_t \sim \mathcal{I}(0)$.

3. Pairs Trading - Clustering

- Another approaches to make associations within the portfolio using unsupervised algorithms :*clustering* algorithm.
- *K-means* with Euclidean distance



K-means with PCA reduction

3. Pairs Trading - Strategy

Financial Framework

We define a new asset:

$$S = n_A S_A - n_B S_B$$

With $n_A, n_B > 0$ such that $S \approx 0$

- No exposure on market movements
- Cash balance ≈ 0

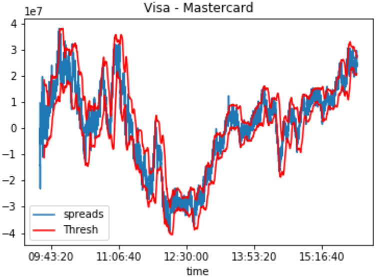
Statistics

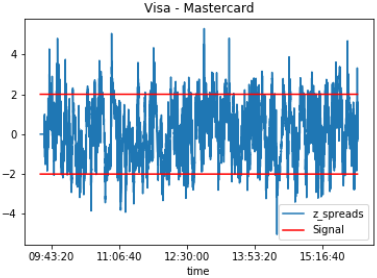
We define:

$$Z = \frac{S - \mu_S}{\sigma_S}$$

Over a rolling time interval.

Day - *20181112*



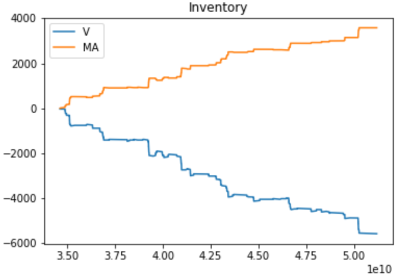


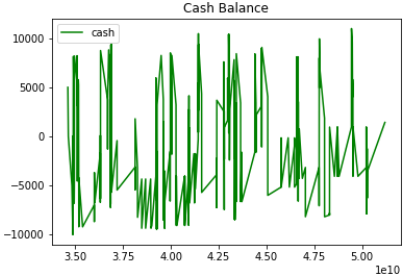
Algorithm Framework

Set parameters and start trading day

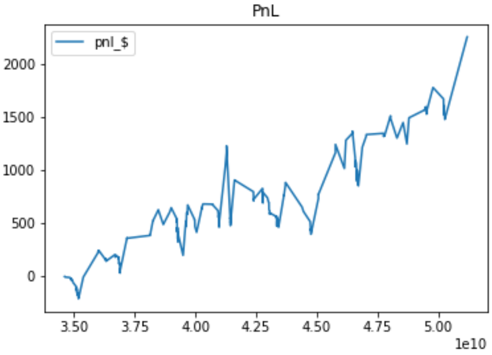
- At every time interval:
  - Retrieve Market Data
  - Update statistics
  - Cancel all existing orders
  - Decide to send a trade
- At every order message from exchange:
  - Check if our orders have been filled
  - Update portfolio

Day - *20181112*





Day - *20181112*



Conclusion

Mean Reversion:

- Few hyper-parameters with high-dimensionality.
- Simulator is powerful, but makes multi-day backtesting slow and inefficient.
- Next Steps: Optimizing the backtesting Class to increase the descent speed to optimum.

Pair Trading:

- A lot hyper-paramters.
- Simulator framework hard to backtest.
- Incorporating mean reversal idea to spot trends in pairs.