**Notes on Chapter 1**

**Overview**

The strategies that will be covered in the book involve different varieties of the following:

1. Mean reversion strategies
   * Drawn from interday/intraday data of stocks, ETFs vs their components, currency pairs, and futures calendar vs. intermarket spreads
   * Why do situations where an ETF will be valued differently than the sum of its components happen?
   * Rollover interest with respect to currencies
   * Backwardation/Contango
2. Momentum strategies
   * Statistical tests for time series momentum
   * 4 main drivers of momentum in stocks and futures
   * Strategies based on news events, news sentiment, leveraged ETFs, order flow, HFT will be covered

Risk in this book will be measured by the Kelly formula.

The statistical techniques used to grade the algorithms includes:

* Dickey-Fuller (ADF test)
* Hurst exponent
* Variance ratio test
* Half-life

**Backtesting & Automated Execution**

One of the most important considerations for backtesting is choosing the platform with which you will be using to backtest.

Backtesting is used in hopes that the historical performance of the strategy will tell us what to expect for its future performance.

The profitability of a strategy is extremely sensitive on the details of implementation including:

* Order type (limit order, market order, market-on-open)
* Time the order will be placed
* Using the bid/ask/last price to trigger a trade

A lot of these are overlooked in papers, so it is important to always backtest a strategy yourself, regardless of where it came from. Note that the author of a paper can tweak their model to yield successful results with their predetermined out-of-sample data. This means that out-of-sample tests really can't occur until after the strategy or research has already been published.

**Common Pitfalls of Backtesting**

**Look-Ahead Bias**

Look-Ahead Bias is when your algorithm is using information in the future to determine today's trading signals. A common mistake made is using the closing price to determine the entry signal of the same day.

An easy way to avoid look-ahead bias is to use the same program for backtesting and live trading, that way you know for sure that you couldn't possibly be using future information (because for the live trading program, it doesn't exist yet)

**Data-Snooping Bias**

This is caused by overfitting your model by having too many parameters, which may look good when testing with historical data, but will not do as well with live data.

The easiest way to mitigate data-snooping is to test the model on OOS data and if it does poorly, then reject that model.

DO NOT TWEAK YOUR MODEL.

If you cannot throw away your whole model, then consider using cross-validation (using different subsets of the data) to train and tweak your data, and make sure it runs for each of these subsets before trying the actual data.

Another easy way to avoid data-snooping bias is to simply make the model as simple as possible, and try to achieve it with as few parameters as possible. Note this also means keeping the number of trading rules low!

Occam's razor dictates that unless there are strong theoretical and empirical reasons to support using a non-Gaussian distribution for your model, use it.

**Stock Splits & Dividend Adjustments**

Failing to adjust for stock splits and dividend payouts can be detrimental to your model. In the case of stock splits, the price of the stock may suddenly be cut by 1/3, and the number of stock you're holding may not be adjusted by your model.

In the case of dividends, the price of the stock will fall by $d, so an adjustment needs to be made for that, too.

**Survivorship Bias**

Failing to include delisted stocks in your models will mean your model suffers from survivorship bias. If your model only includes stocks that are trading today (for example, trading each of the stocks listed on the TSX), your backtest may look great, but if one of the companies fails, it could be very detrimental to your P/L.

**Primary vs. Consolidated Stock Prices**

Many U.S. stocks are traded on multiple exchanges, ECNs, and dark pools. When you look up the closing price of a stock on a specific day, that number represents the last execution price on any of these venues during regular trading hours (and likewise for the opening price).

When you submit a market-on-close (MOC) or market-on-open (MOO) order, it'll only be routed to the primary exchange only (e.g. MOC order on IBM gets routed to NYSE). This means that if you want to include those order types in your model, you'll need the historical prices from the primary exchange.

Note that using consolidated data (i.e. price data from all the venues) is not exactly accurate, since a small number of shares can be traded for a way different price than the price on the primary exchange. This applies to high and low prices, as well, so you're better off just using primary exchange data.

**Venue Dependency of Currency Quotes**

The currency markets are more fragmented when compared to the stock market and no rule dictates that a trade executed at a venue has to be at the best bid/ask across all the venues.

Because of this, we can only use data from the same venues we expect to trade on.

Another feature of currency quotes is that trade prices and sizes are generally not available (at least without a small delay). This is because there is no regulation that say that the dealer must report the trade price to all market participants (unlike stocks).

**Short-Sale Constraints**

Shorting a stock is not as easy as it sounds. A broker first has to locate a quantity of stock from other customers/institutions (usually mutual funds or asset managers), and arrange a stock loan.

A large short interest or a small amount of float of the stock means that the interest you have to pay to the stock lender will be expensive.

This means that if your model includes shorting, you may see amazing results only because other investors were unable to short/it was too expensive for most investors at that time.

It's not easy to find.n accurate list of hard-to-borrow stocks for your backtest so there are some quick rules of thumb:

* Small cap stocks are affected more by short-sale constraints than large cap
* ETFs are as hard to borrow as stocks

One more constraint is the uptick-rule (1938-2007) which dictated that a short sale had to be executed at a price higher than the last traded price. This was replaced by the Alternative Uptick Rule in 2010 which also requires a short sale to have a trade price higher than that the national best bid, but only after a circuit breaker has been triggered (i.e. a stock is traded at 10% lower than its previous close and remains in effect for the following day). This prevents short sales from being filled at that time, so this is another consideration for your model.

**Futures Continuous Contracts**

Trading futures is different than trading stocks since futures contracts have expiry dates. This means that a strategy which trades crude oil is really a strategy on many different futures contracts.

The way to choose which contract you will trade depends on when you decide to sell the current contract and "roll-over" to the next month.

This can be a fixed amount (e.g. 10 days before expiration), or a variable amount (e.g. when the open interest of the next contract exceeds the open interest of the current contract).

**Futures Close vs Settlement Prices**

Settlement price is used in most cases since it is determined at the end of the day and is closest to the price of the transactions that happen at market close. If the last recorded transaction was hours ago, then it is probably not as useful.

An important note is that contracts can be traded on different exchanges and therefore have different closing prices, so it would be ineffective to create a spread trade using their closing prices.

**Hypothesis Testing**

The general framework of hypothesis testing in backtesting scenarios is as follows:

1. Calculate the test statistic of your backtest on a finite sample of data, call the test statistic your average daily return of a trading strategy in that period
2. Suppose that the true average daily return based on an infinite data set is 0 (i.e. the null hypothesis)
3. Suppose that the distribution of returns is known and has a mean of 0
4. Compute the probability, p, that the returns will be at least as large as the value observed in the backtest (i.e. the p-value), and if the p-value is small, we can reject the null hypothesis and conclude that the backtested strategy is statistically significant

There are ways to choose your distribution for step 3 including:

* Assume returns follow a standard distribution such as the Gaussian distribution with mean zero and SD given by the sample SD
* Use Monte Carlo methods to generate simulated historical price data and feed these into our strategy to determine the empirical probability distribution of profits
* Andrew Lo method:
  1. Generate a set of simulated trades, with the same number of long and short entry trades and same average holding period
  2. Distribute the trades randomly over the actual historical price series
  3. Measure the fraction of how many sets averaged a higher return than the backtest average return

**When Not to Backtest**

**Example 1:**

A backtest with annualized return of 30%, Sharpe ratio of 0.3, maximum drawdown duration of TWO YEARS

* Two years is way too long for traders to be under water
* Low Sharpe ratio
* Both of the above indicate an inconsistent strategy and likely a fluke which will not repeat

Lesson learned: Don't bother backtesting high return but low Sharpe ratio strategies. Do not bother backtesting strategies with a maximum drawdown duration longer than what you or your investors can endure

**Example 2:**

A long only crude oil futures strategy returned 20% in 2007, with a Sharpe Ratio of 1.5

* Simply holding the front month crude oil futures returned 47% with a Sharpe ratio of 1.7
* This strategy isn't better than just buying and holding

Lesson Learned: Always choose an appropriate benchmark to measure a trading strategy

**Example 3:**

A simple “buy-low-sell-high” strategy picks the 10 lowest priced stocks at the beginning of the year and holds them for a year. The backtest return in 2001 is 388 percent.

* Was the strategy backtested using a survivorship-bias free stock database?

Lesson learned: The same strategy applied to a database that included delisted stocks returned -100%

**Example 4:**

A neural net trading model that has about 100 nodes generates a backtest Sharpe ratio of 6.

* Red flag: neural network trading model (100 nodes is also a lot)
* The number of parameters to be fitted with in-sample training data is proportional to the number of nodes which implies at least 100 parameters

Lesson learned: Neural networks can be a red flag for overfitting

**Example 5:**

A high-frequency E-mini S&P 500 futures trading strategy has a backtest annual average return of 200 percent and a Sharpe ratio of 6. Its average holding period is 50 seconds.

* Backtesting a high-frequency strategy doesn't really make sense
* Placing or executing an order can affect the market, especially in HFT settings, so it's not entirely realistic in this case

**Will a Backtest Be Predictive of Future Returns?**

If we avoid all the pitfalls and address all of our assumptions, is that enough for future returns? Not really, here are a few changes that would affect every model:

* Decimalization of U.S. stock quotes on April 9, 2001
* 2008 financial crisis that caused a 50% collapse of average daily trading volumes
* 2008 crisis that created a multiyear bear market in momentum strategies
* SEC's Regulation NMS
* Removal of old uptick rule