An analysis of the Moving Average Startegy and other ideas

Trend Indicators

These are technical tools which quantify the strength and direction of trends within a chosen time-frame. Here are a couple of examples of trend indicators.

1. Simple Moving Averages

An unweighted mean of the previous k prices.

Given:

$$X = [x_1, x_2, x_3, \ldots, x_n]$$

Our first SMA value given a window of k starting from x_1 to x_k, would be:

$$SMA_1 = \frac{\sum_{i=1}^{k} x_i}{k}$$

The next SMA value is then calculated using the following equation:

$$SMA_2 = SMA_1 + \frac{x_{k+1} - x_1}{k}$$

Notice how we re-use x_1, so we store the first sequence that would be used to compute the SMA_j value, so we would have a general equation such as:

$$j = 0, \ldots, n - k + 1$$

$$SMA_j = SMA_{j-1} + \frac{x_{k+j-1} - x_{j-1}}{k}$$

When are SMA's used?

SMA's are regularly utilised in the technical world for entry and exit signals, however we are better of to look at them as as a transformation of data. It makes data more smooth and thus it helps to reduce the level of noise in the dataset which can be caused from random irregular short term price fluctuations.They can show us the changes in price with respect to time whilst taking into account the previous prices. More so, it a change in the average of prices.

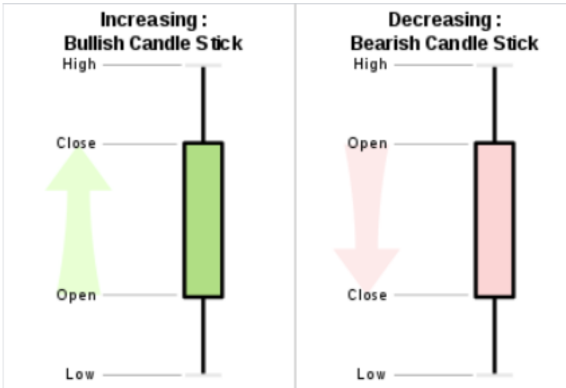
In the technical world however, it can be used as a strategy such as a price cross over with a moving average.

Lets have a look at an image:

(1.0) Real Time use of Simple Moving Averages on ETH (Candlestick)

(1.1) Real Time use of Simple Moving Averages on ETH (Line Price)

Here I will explain a couple of things. The first type of graph is a candlestick graph and the second type of graph is a line graph for prices. Candlestick graphs are different in that they represent 4 prices within that 1 time step. Here is a quick glance at what the candlesticks actually represent.



(1.1) What is a Candlestick?

Close - Close Price (the price at the end of the time step) Open - Open Price (the price at the beginning of the time step) High - High Price (the highest price within that time step) Low - Low Price (the lowest price within that time step)

Each time step, can represent a certain minute, 30 minute period, 1 hour, 3 hours, 1 day, 1 week, 1 month and etc time period.

Now if we have a look at charts 1.0 and 1.1, with MA 7, which means moving average that averages 7 time steps (window size is 7) when looking at daily data where each time step represents 1 day, the crossover periods are much more frequent then MA 25 and MA 99. Let the price chart graph be represents as $g(t_i)$ and t_i, is a certain time step. Let MA 7 be represented as $M_7(t_i)$. Note we are using the line graph for this demonstration.

If t_i is the current time step, when $g(t_{i-1}) > M_7(t_{i-1})$,but on time step t_i, $g(t_i) == M_7 t_i$, then there is an expectation that on the t_{i+1} time step, $g(t_{i+1}) < M_7(t_{i+1})$ is going to occur. The cross over signal occurs on time step t_i, and depending on the direction or gradient at $g(t_{i-1})$, at t_{i+1}, it is assumed to be going on to that same direction. Therefore, the M_7 can tell us when to buy at time step t_i if $g(t_{i-1}) < M_7(t_{i-1})$ or when to sell if $g(t_{i-1}) > M_7(t_{i-1})$.

However, note that there may be false signals which are signals that cause you to make the wrong decisions and these indicators most likely do not have any predictive power.Despite, there being scenarios where it does work there are also many scenarios where it doesn't work. A good question to ask is, what differences are there among the scenarios it does work in and the scenarios it does not work in? What features of a scenario (we can define a scenario by the window from t to t+k) can we extract and compare with of different scenarios? Are there certain hidden patterns which correlate to a high probability of success when using these indicators?

One of the features of an SMA's is that there is an equal weighting of data points, looking at SMA's from a perspective where technical indicators do not have predictive power, I would not say that this is a limitation but rather just an aspect. This is just the form of data transformation, however how useful is this data transformation anyway?

Say we have:

$$[P_1, P_2, P_3, P_4, \dots, P_n]$$

If we were to average those n points we have A_1.

Now, say we were to randomly generate those prices BUT we do so, in such a way that the average of those points is always A_1.

A simple example to illustrate this point:

Say we have 3 integers. The average of those integers have to be 12, how many possible combinations are there?

$$[12 - P_k, 12 - P_k - P_{k+1}, 12 - P_k - P_{k+1} - P_{k+2}]$$

And

$$12 - P_k - P_{k+1} - P_{k+2} = 0$$

We can first restrict what P_k is by first looking at what P_{k-1} is (the previous price), by looking at the historical changes in price. We can randomly select what P_k is, we can do this for the selection of other P_{k+i}. If we do this n times, and randomly generate fairly accurate sequences of prices which don't violate the above condition. So say we have m different sequences. Each different sequence is a different reflection of conditions, which are unknown and may have some sort of predictive power. We are compressing m different scenarios, each which holds some unique information to 1 value. Thus, there is a great loss of information. We can possibly, implement the algorithm above to create different prices for the SAME moving average curve thus demonstrating that the probability of success using a moving average is lower than what someone would expect it to be when just looking at charts.

2. Exponential Moving Average

It is a type of weighted moving average, which gives more importance to recent price data. It is used to see price trends over time too and it is used in the same way as an SMA is. Since new data carries more weight it responds to changes more quickly and thus would be quicker to help you realize if a certain trend might occur.

We initialize our EMA and say our window size is k, j is the day number we are on. i is the EMA number we are on.

$$j = k + i - 1, j \leq n - k + 1, i = 1, j = k$$

$$EMA_1 = \frac{\sum_{i=1}^{j} x_i}{k}$$

$$EMA_2 = x_{j+1} \times (\frac{s}{1+j}) + EMA_1 \times (1 - (\frac{s}{1+j}))$$

$$EMA_{i+1} = x_{j+1} \times (\frac{s}{1+j}) + EMA_i \times (1 - (\frac{s}{1+j}))$$

Running a Simulation of Moving Average Cross Over strategies.

Here is a general description of my Naive Simulations Algorithm, which only focuses on going long. For this example we are using no transaction fees. The rules of the simulation are, you cannot make simultaneous trades. You can ONLY buy if you have sold, or if you haven't bought for example. You can ONLY sell if you have BOUGHT or not. The trading signals are produced every iteration.

Simulation of Naive Algorithm:

1. Initialize Starting amount, and initialise starting point. (Point should ensure that you can work with the maximal amount of points, or from a common point if you are comparing strategies). Initialise your Sell Signal as 0, Buy Signal as 1.

2. While the index is less than the size of the total number of prices, pass in data up till that index to your strategy function.

3. Strategy function should either return 0,1,-1. 0 means no Signal was generated. 1 means a buy signal was generated, 0 means a sell signal was generated.

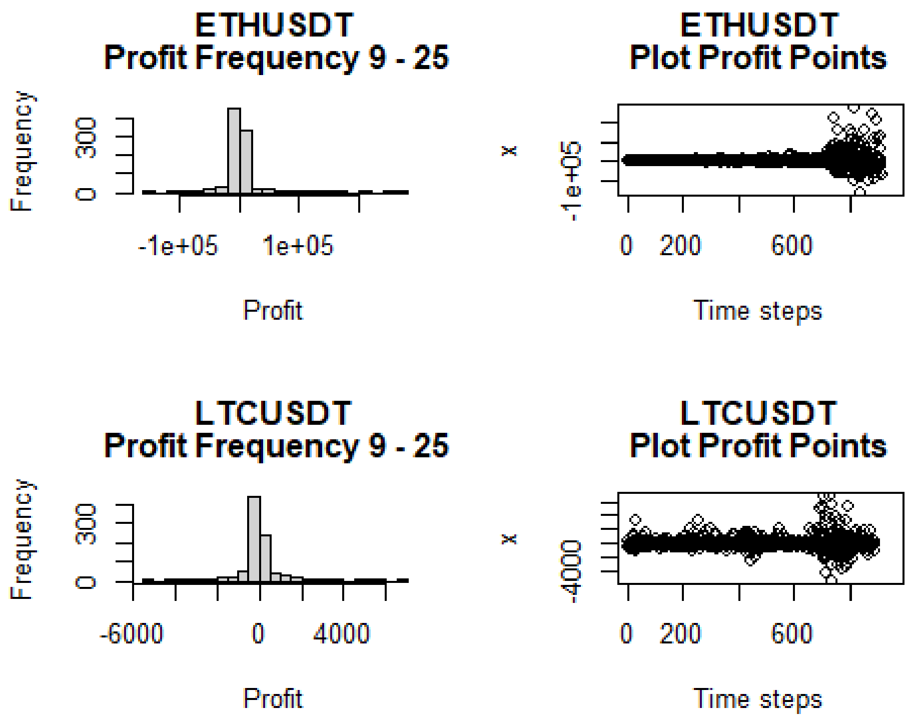
4. Act upon that signal, only sell if your sell Signal values is 1, meaning you have previously bought the stock. Only buy if your buy signal value is 1, meaning you have previously sold the stock and now you can buy. When buying or selling, we buy as much as we can and sell as much as we can.

5. For every transaction charge a %0.005 fee of the total amount involved in the transaction. So when selling, the amount you recieve will be %0.005 less, when buying the amount you buy with will be %0.005 less.

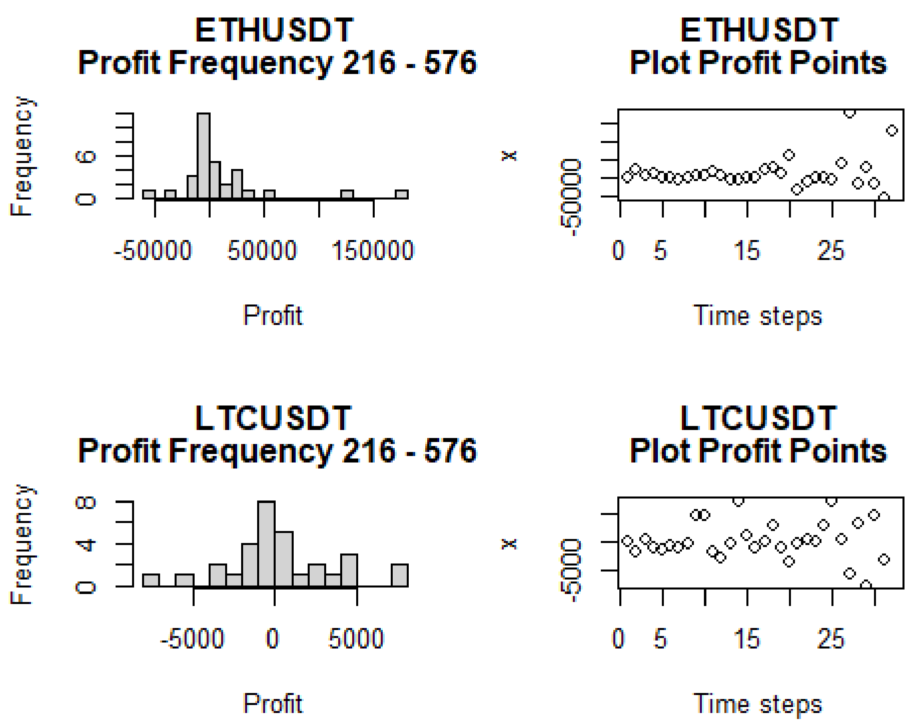
6. Continue, until we run out of points.

I ran this naive simulation only JUST using moving average cross overs.

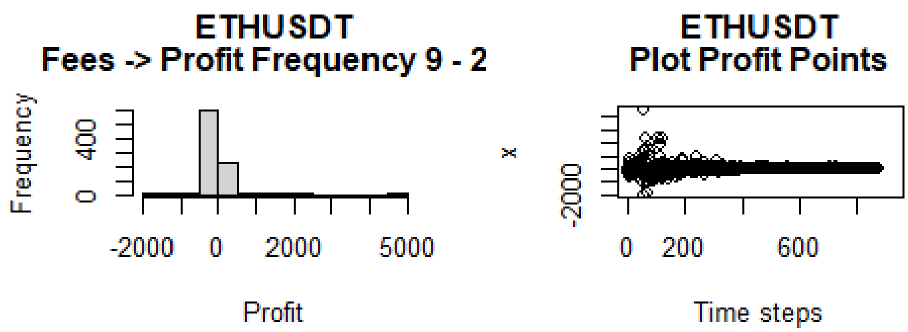
```
## [1] "C:/Users/premp/Desktop/Trading/Source"

##   X BNBUSDT  ADAUSDT  BTCUSDT  ETHUSDT  NEOUSDT QTUMUSDT  XRPUSDT  LTCUSDT
## 1 0 15.3071 5.013334 31.95155 169.8909 15.14783 5.099906 9.034672 1.427001
```

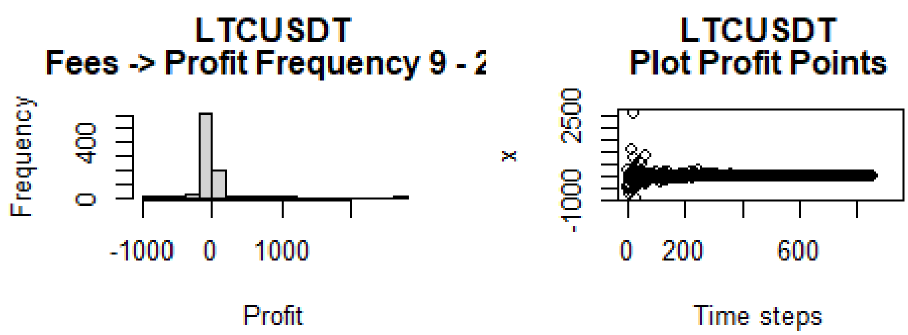



```
##   X  BNBUSDT ADAUSDT BTCUSDT   ETHUSDT   NEOUSDT QTUMUSDT   XRPUSDT LTCUSDT
## 1 0 12.00607  12.901 21.3926 38.47277 3.721223 2.747082 5.857437 1.76919
```

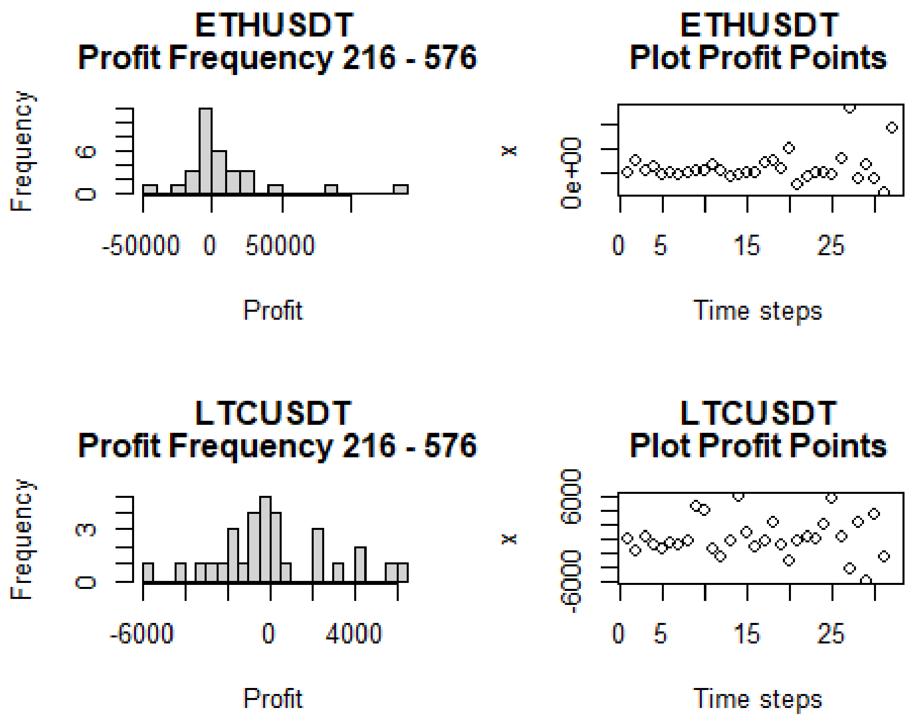


```
##    X   BNBUSDT   ADAUSDT     BTCUSDT     ETHUSDT      NEOUSDT   QTUMUSDT
XRPUSDT
## 1 0 1.234593 0.438866 0.00301764 0.02065873 0.001866393 0.3248682
0.8796684
##         LTCUSDT
## 1 0.0002172145
```






```
##    X   BNBUSDT   ADAUSDT   BTCUSDT   ETHUSDT   NEOUSDT  QTUMUSDT   XRPUSDT
LTCUSDT
## 1 0 10.97025 11.78797 15.99567 27.91455 2.867384  2.43571 5.406011
1.296599
```



Above is an analysis of profit frequencies for different moving averages. Now, despite moving averages not having much predictive power it did generate some results, however how good those results are depends on how it compares to just holding. We also can use other measurements such as the sharpe ratio and etc too.

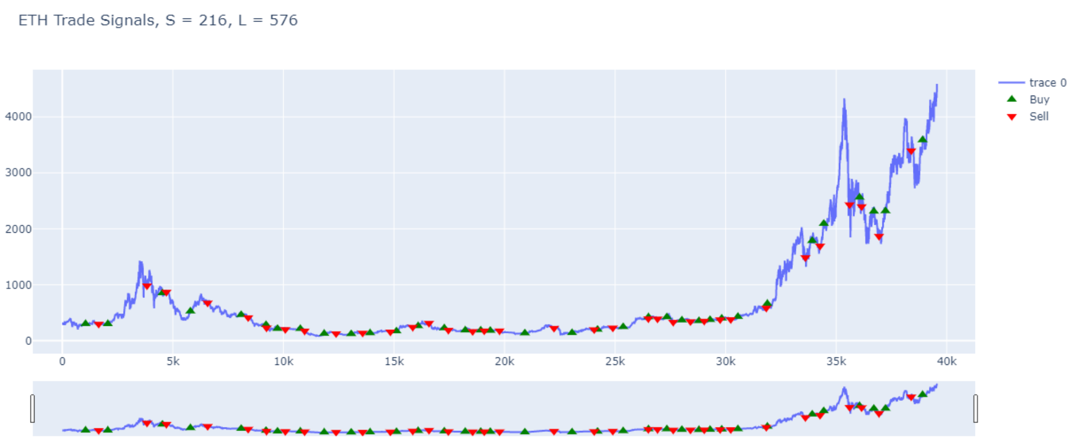
However, all these coins exist until today, what about the ones which did not succeed, how profitable would this strategy even be?

Another aspect of using technical analysis which I think is not intuitive and useful is the actual optimization of windows. There needs to be a discussion as to why some windows are more useful then others, there should also be some ranking to see whether there are big fluctuations in what window sizes are good and bad. For example if (10, 20) is the best but the second best is (30, 100) and the third best (20, 24), this almost seems too random and it speaks to the actual usefulness, reliability and robustness of these strategies.

The above results show 4 different simulations.

1. Simulations without ANY transaction fees and window length of 9 and 24.
2. Simulations without ANY transaction fees and window length of 216 and 576
3. Simulations WITH transaction fees and window length of 9 and 24.
4. Simulations WITH transaction fees and window length of 216 and 576.

Trading Signals LTC



Trading Signals ETH

Previous LTC Trading Signals with MA examples

Previous ETH Trading Signals with MA examples

We can clearly see that a lot of bad decisions were made when signals were being triggered despite there being an obvious downward trend, this is the perfect example as to why these indicators can not have any predictive power.

The above, are graphs with the moving average and buy/sell signals. Another moving average of 3000 is included in the graph. As you can see, this forecasting moving average can essentially filter out bad trades from happening when we are going long. For example, for the LTC chart you can clearly see how the first few trades would result in a lot of losses, whilst the trades towards the end would of resulted in gains. The moving average of 3000, essentially helps determine what the long term trends are and in doing so we can filter out a LOT of trades. This might be a slight indication as to why we should be extracting as many features as possible when making predictions, despite these indicators having no predictive power there was a little success.

Here below are a couple more examples comparing forecasts with non-forecasts.

ETH Forecast + MA Cross over strategy

LTC Forecast + MA Cross over strategy

ETH Non Forecast + MA Cross over strategy

LTC Non Forecast + MA Cross over strategy

The returns when we applied the forecasting strategy are listed above. The varying changes in returns can be explained.

1. ETH returns increased with the use of a strong long term trend indicator. ETH in general is a less volatile stock then LTC, there were not many price changes WHICH went strictly against AND for the long term indicator. Executions of cross overs were done normally, as long as the long term indicator allowed these executions occur.

2. LTC returns DECREASED, despite it preventing SOME BAD trades at the beginning, it unfortunately missed out on opportunities to capitalize on the volatility. This is not a fault due to the specific use of forecasting, but the general nature of using moving averages. If you decrease the size of the windows with forecasting the returns actually increases, but if you don't use the long term indicator for trends then the returns increases even more. Hence, using an alternative strategy for assets which

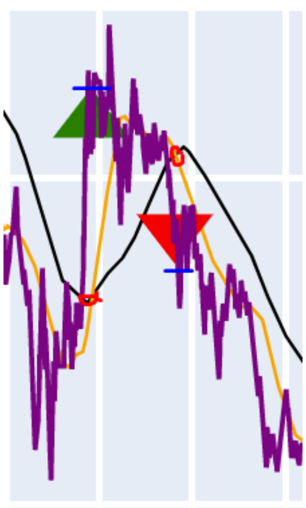
are innately volatile, should be preferred. Also, note that without the forecasting some favorable trades were made, but this is due to the general volatility of asset and luck.

So the forecasting strategy with the moving average cross overs would have:

1. Reduced the number of false signals.
2. Increased profits as less profitable and costly trades are not made. (For example if we lose 5000 from our first trade and make double in the next trade we would be at 10000 but there would be no profit. If we skipped the initial trade, we would be sitting at 20000, so hence reducing the number of false signals can help increase profits).

However it does not: 1. Increase the profitability of each trade taken, how much profits we can make from some trades are limited since we are using moving average cross overs for buy and sell entries. This is an inherent flaw, and it does not consider how price volatility can sometimes make moving average cross over strategies redundant.

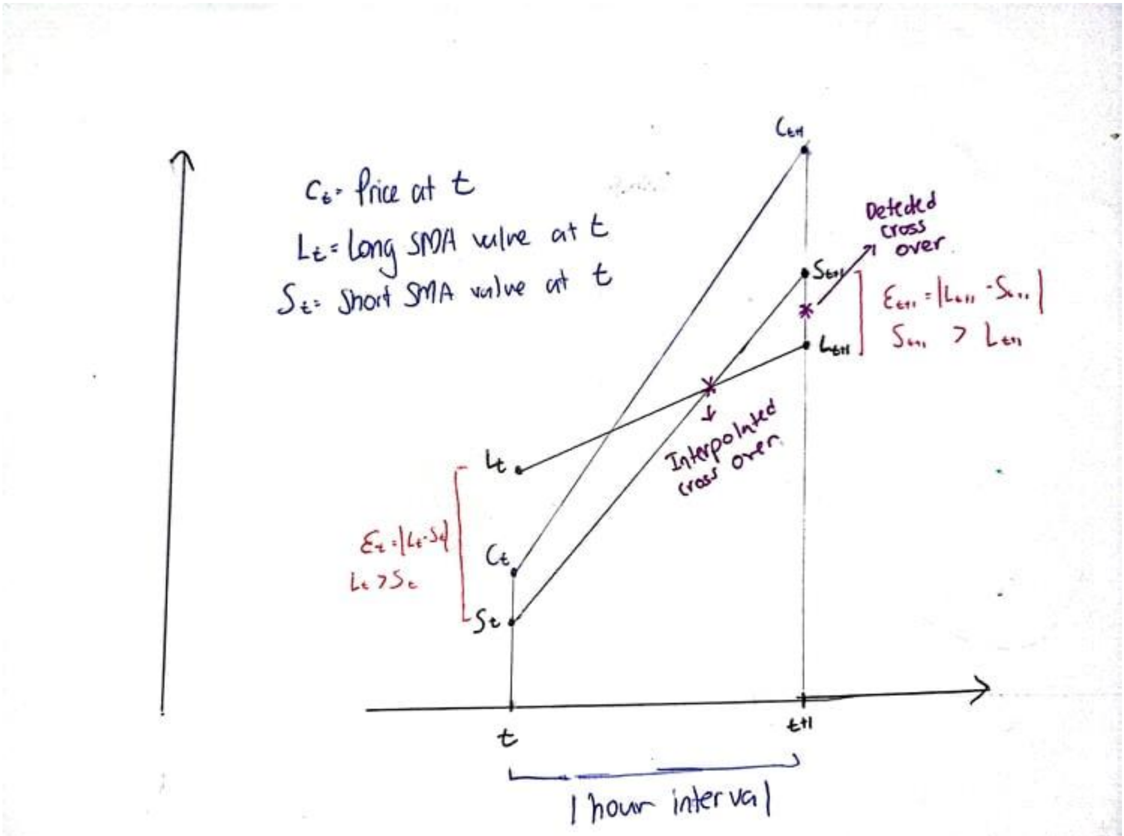
Moving Average cross over strategies have a strong weakness which cannot be resolved by using forecasting techniques. Here is an example why:



(A) Fatal Flaw of Moving Averages?

The red dots classify the period of times where the moving averages cross over. You can see how it crosses over BUT the actual price at that time step where the algorithm registers that there has been a cross over is so vastly different that the trade becomes a loss. In these circumstances we can either use a much more sensitive moving average, but it will be too hard to constantly tune moving average window parameters. Thus, moving averages are only used to confirm trends, but when used during highly volatile periods situations like these might be more frequent leading to more losses, obviously the granularity of data is also a factor too.

Here is a clear illustration:



The problem is demonstrated more clearly here. The buy signal would have been generated at C_{t+1}, but the price increases can be so much so that it can produce an almost false signal, but the actual cross over happens in between the link from C_t to C_{t+1}, so now the question becomes, to what degree of similarity can we predict C_{t+1}, such that when computing S_{t+1} and L_{t+1} it still generates a cross over? This would mean we would have to predict where the price might go and what the volatility of that price might be, which influences the actual price prediction. Hence, the determination of real buy and sell signals is a lot more complicated. This is an obvious pitfalls of indicators which only confirm trends. I also would like to bring up the point that during the data transformation much information is lost, this loss information is the reason as to why such bad signals are produced.

Thus, despite the above graphs and data showing SOME efficacy of the moving average strategy there are obvious limitations to it which do not make it a robust and reliable strategy.