Learning the population dynamics of technical trading strategies

Nicholas Murphy[1,*] and Tim Gebbie[1,†]

[1]Department of Statistical Sciences, University of Cape Town, South Africa

We use an adversarial expert based online learning algorithm to learn the optimal parameters required to maximise wealth trading zero-cost portfolio strategies. The learning algorithm is used to determine the relative population dynamics of technical trading strategies that can survive historical back-testing as well as form an overall aggregated portfolio trading strategy from the set of underlying trading strategies implemented on daily and intraday Johannesburg Stock Exchange data. The resulting population time-series are investigated using unsupervised learning for dimensionality reduction and visualisation. A key contribution is that the overall aggregated trading strategies are tested for statistical arbitrage using a novel hypothesis test proposed by Jarrow *et al.* [23] on both daily sampled and intraday time-scales. The (low frequency) daily sampled strategies fail the arbitrage tests after costs, while the (high frequency) intraday sampled strategies are not falsified as statistical arbitrages after costs. The estimates of trading strategy success, cost of trading and slippage are considered along with an offline benchmark portfolio algorithm for performance comparison. The work aims to explore and better understand the interplay between different technical trading strategies from a data-informed perspective.

I INTRODUCTION

The problem of selecting plausible trading strategies and allocating wealth among these strategies to maximize wealth over multiple decision periods can be a difficult task. An approach to combining strategy selection with wealth maximisation is to use online or sequential machine learning algorithms [1]. Online portfolio selection algorithms attempt to automate a sequence of trading decisions among a set of stocks with the goal of maximizing return in the long run. Such algorithms typically use historical market data to determine, at the beginning of a trading period, a way to distribute their current wealth among a set of stocks. These types of algorithms can use many more features than merely price, so called "side-information", but the principle remains that same.

The attraction of this approach is that the investor does not need to have any knowledge about the underlying distributions that could be generating the stock prices (or even if they exist). The investor is left to "learn" the optimal portfolio to achieve maximum wealth using past data directly [1].

Cover [2] introduced a "follow-the-winner" online investment algorithm[1] called the Universal Portfolio (UP) algorithm[2]. The basic idea of the UP algorithm is to allocate capital to a set of experts characterised by different portfolios or trading strategies; and to then let them run while at each iterative step to shift capital from losers to winners to find a final aggregate wealth.

Here our "experts" will be similarly characterised by a portfolio (or trading strategy) where a particular agent makes decisions independently of all other experts. The UP algorithm holds parametrized constant rebalanced portfolio (CRP) strategies as it's underlying experts. We will have a more generalised approach to generating experts. The algorithm provides a method to effectively distribute wealth among all the CRP experts such that the average log-performance of the strategy approaches the best constant rebalanced portfolio (BCRP) which is the hindsight strategy chosen which gives the maximum return of all such strategies in the long run. The key innovation that Cover [2] provided was a mathematical proof for this claim based on an arbitrary sequences of ergodic and stationary stock return vectors.

If some log-optimal portfolio exists such that no other investment strategy has a greater asymptotic average growth then to achieve this one must have full knowledge of the underlying distribution and of the generating process to achieve such optimality [2–5]. Such knowledge is unlikely in the context of financial markets. However, strategies which achieve an average growth rate which asymptotically approximates that of the log-optimal strategy are possible when the underlying asset return process is sufficiently close to being stationary and ergodic. Such a strategy is called *universally consistent*. Gyorfi *et al.* [5] proposed a universally consistent portfolio strategy and provided empirical evidence of a strategy based on nearest-neighbour based experts which reflects such asymptotic log-optimality.

The idea is to match current price dynamics with similar historical dynamics (pattern matching) using a nearest-neighbour search algorithm to select parameters for experts. The pattern-matching algorithm was extended by Loonat and Gebbie [6] in order to implement

*Electronic address: mrpnic004@myuct.ac.za
†Electronic address: tim.gebbie@uct.ac.za

[1] *follow-the-winner* algorithms give greater weightings to better performing experts or stocks

[2] The algorithm was later refined by Cover and Ordentlich [4] (see Section II B)

a zero-cost (long/short) portfolio selection algorithm via a quadratic approximation in the context statistical arbitrage trading. The algorithm was also re-cast to replicate near-real-time applications using look-up libraries learnt off-line. However, there is a computational cost associated with coupling creation of off-line pattern libraries; more specifically, the algorithms are not truly online.

A key objective in the implementation of online learning in this work is that the underlying experts are online too; they can be sequentially computed on a moving finite data-window using parameters from the previous time-step. An online approach rather than having the need to search data-histories or make comparisons with a library of patterns learnt off-line.

Here we ignore the pattern matching step in the aforementioned algorithm and rather propose our own expert generating algorithm using tools from technical analysis. Concretely, we replace the pattern-matching expert generating algorithm with a selection of technical trading strategies.

Technical analysis indicators are popular tools from technical analysis used to generate trading strategies [7, 8]. They claim to be able to exploit statistically measurable short-term market opportunities in stock prices and volume by studying recurring patterns in historical market data [9–11]. What differentiates technical analysis from traditional time-series analysis is that it tend to place an emphasis on recurring patterns rather than recurring statistical properties of times-series.

This work does not address the question: Which, if any, technical analysis based methods have useful information for trading? Rather we aim to bag a collection of technical experts and allows them to compete in an adversarial manner using the online learning algorithm. This then allows us to consider whether the resulting aggregate strategy can: first, pass reasonable tests for statistical arbitrage, and second, has a relatively low probability of being the result of back-test overfitting. Can the strategy be considered a statistical arbitrage, and can it generalise well out-of-sample.

Traditionally, technical analysis has been a visual activity whereby traders study the patterns and trends in charts based on price or volume data, and use these diagnostic tools in conjunction with a variety of qualitative market features and news flow to make trading decisions. Perhaps not dissimilar to the relationship between alchemy and chemistry, or astrology and astronomy, but in the setting of financial markets. Although many studies have criticised the lack of a solid mathematical foundation for many of the proposed technical analysis indicators [12, 13, 15]. There has been an abundance of academic literature utilising technical analysis for the purpose of trading and several studies have attempted to develop and test indicators in a more mathematically, statistically and numerically sound manner [13, 14, 16]. However, much of this work needs to be viewed with some suspicion - it is extremely unlikely that this or that particular strategy or approach was not the result of some sort of back-test overfitting [17–19]. However, that is not to say that there may not be some merit in non-linear time-series analysis or pattern-matching in financial markets.

Here we will be concerned with the idea of understanding whether the collective population of technical experts can through time, lead to dynamics that can be considered a statistical arbitrage [20] with a reasonably low probability of back-test overfitting [19].

Can we generate wealth (before costs) using the online aggregation of technical strategies? Then, what broad groups of strategies will emerge as being successful (here in the sense of positive trading profits with declining variance in losses)? Costs are always a plausible explanation for any apparently profitable trading strategy (see [6]), and after costs, the high-likelihood that there was a healthy amount of data-overfitting; particularly given that we only have single price paths from history and have little or no knowledge about the true probability of the particular path that has been measured.

Rather than considering various debates relating the technicalities of market efficiency where one is concerned with the expeditiousness of market prices to incorporate new information at any time where information is explicitly exogenous; we restrict ourselves to market efficiency in the sense used by Fischer Black [13, 21, 22]. This is the situation where some of the short-term information is in fact noise, and that this type of noise is a fundamental property of real markets. Although market efficiency may plausible hold over the longer term, on the short-term there may be departures that are amenable to tests for statistical arbitrage [23], departures that create incentives to trade, and more importantly departures that may not be easily traded out of the market dues to various asymmetries in costs and market access.

In order to analyse whether the overall back-tested strategy depicts a candidate statistical arbitrage, we implement a test first proposed by Hogan *et al.* [20] and then further refined by Jarrow *et al.* [23]. Hogan *et al.* provided a plausible technical definition of statistical arbitrage based on a vanishing probability of loss and semi-variance in the trading profits, and then use this to propose a test for statistical arbitrage using a Bonferroni test [20]. This test methodology was extended and generalized by Jarrow *et al.* [23] by computing a so-called Min-t statistic to then use a Monte Carlo procedure to make inferences regarding a carefully defined no statistical arbitrage null hypothesis.

This is analogous to evaluating market efficiency in the sense of the Noisy efficient market hypothesis [21] whereby a failure to reject the no statistical arbitrage null hypothesis will result in concluding that the market is in fact sufficiently efficient and no persistent anomalies can be consistently exploited by trading strategies over the long-term.

Traders will always be inclined to employ strategies which depict a statistical arbitrage and especially strategies which have a probability of loss that declines to zero quickly as such traders will often have limited capital and short horizons over which they must provide satisfactory returns (profits) [23].

We make the effort here to be very clear that we do not attempt to identify profitable (technical) trading strategies, but rather we will generate a large population of strategies (experts) constructed from various technical trading rules and combinations of the associated parameters of these rules which will in turn represent the experts in the online learning algorithm. The experts who accumulate the greatest wealth during each trading period will receive more wealth in the following trading period and thus contribute more to the final aggregated portfolio.

This requires us to introduce a methodology to transform the signals generated by the technical trading experts into a set of portfolio weights (controls) for each expert in a given time period t to compute the expert wealth's. We then provide the equity curves for the individual experts portfolios (the accumulated trading profit through time). This can be best thought of as some sort of "fund-of-funds" over the underlying collection of trading strategies. This is a meta-expert that aggregates experts that represent all the individual technical trading rules. The overall meta-expert strategy performance is achieved by the online learning algorithm.

We perform a back-test of the algorithm on two different data sets over two separate time periods; one using daily data over a six year period, and the other using a mixture of intraday and daily data over a two month period. A selection of the fifteen most liquid stocks which constitute the Johannesburg Stock Exchange (JSE) Top 40 shares is utilised for the two separate implementations[3].

The overall strategy performance is compared to the BCRP strategy to form a benchmark comparison to evaluate the success of our strategy. The overall strategy is then tested for statistical arbitrage to find that in both a daily, and intraday-daily data implementation, the strategy depicts a statistical arbitrage.

It must be cautioned that this is prior to accounting for costs which can be expected to shift the bias. The key point here (as in [6]) is that is does seem that plausible statistical arbitrages are detected on the meso-scale and short-term, however, for reasonable costing, it may well be that these statistical arbitrages exists because they cannot be profitable traded out of the system. In addition to this, the overall strategy's probability of loss is computed to get an idea of the convergence of such losses to zero; most traders will be concerned about short run losses.

The paper will proceed as follows: Section II explains the construction of the algorithm including details of how the experts are generated, how their corresponding trading signals are transformed into portfolio weights and a step-by-step break-down of the learning algorithm. In Section III, we introduce the concept of a statistical arbitrage and the methodology for implementing a statistical arbitrage test for a trading strategy, and for computing probability of loss of a strategy. Section IV describes the data utilised in the study. All results of implementations of our algorithm on the data and in-depth analysis is presented in Section V. Section VI states all final conclusions from the experiments and possible future work.

In summary, we are able to show that on a daily sampled time-scale there is most likely little value in the aggregate trading of the technical strategies of the type considered here. However, on intraday times-scale things look slightly different. Even with reasonable costs accounted for there still seems to be the possibility that price based technical trading cannot be ruled out as offering an avenue for statistical arbitrage. Considerable care is still required to ensure that one is accounting for the full complexity of market realities that may make it practically difficult to arbitrage these sorts of apparent trading opportunities out of the market as they may be the results of top-down structure and order-flow itself rather than some notion of information inefficiency; the signature of noise trading.

II LEARNING TECHNICAL TRADING

Rather than using a back-test in the attempt to find the single most profitable strategy, we produce a large population of trading strategies (experts) and use an adaptive algorithm to aggregate the performances of the experts to arrive at a final portfolio to be traded. The idea of the online learning algorithm is to consider a population of experts created using a large set of technical trading strategies generated from a variety of parameters and to form an aggregated portfolio of stocks to be traded by considering the wealth performance of the expert population. During each trading period, experts trade and execute buy (1), sell (-1) and hold (0) signals independent of one another based on each of their individual strategies. The signals are then transformed into a set of portfolio weights (controls) such that the overall sum of weights in the portfolio sums to zero (zero-cost) and the strategy becomes self-funding. We also require that the portfolio is unit leveraged. Based on each individual expert's accumulated wealth up until the some time time t, a final aggregate portfolio for the next pe-

[3] More details on the data sets can be found in Section IV

riod $t + 1$ is formed by creating a performance weighted combination of the experts. Experts who perform better in period t will have a larger relative contribution toward the aggregated portfolio to be implemented in period $t+1$ than those who perform poorly. Below, we describe the methodology for generating the expert population.

A Expert Generating Algorithm

A.1. Technical trading

Technical trading refers to the practice of using trading rules derived from technical analysis indicators to generate trading signals. Here, indicator's refer to mathematical formulas based on prices (OHLC), volume traded or a combination of both (OHLCV). An abundance of technical indicators and associated trading rules have been developed over the years with mixed success. Indicators perform differently under different market conditions which is why traders will often use multiple indicators to confirm the signal that one indicator gives on a stock with another indicators signal. Thus, in practice and various studies in the literature, many trading rules generated from indicators are typically back tested on a significant amount (typically thousands of data points) of historical data to find the rules that perform the best[4]. It is for this reason that we consider a diverse set of technical trading rules. In addition to the set of technical trading strategies, we implement three other popular portfolio selection algorithms each of which has been adapted to generate zero-cost portfolio controls. An explanation of these three algorithms is provided in Appendix D while each of the technical strategies are described in Appendix C.

In order to produce the broad population of experts, we consider combinations among a set of four model parameters. The first of these parameters is the underlying strategy of a given expert, $\boldsymbol{\omega}$, which corresponds to the set of technical trading strategies and trend-following strategies where the total number of different trading rules is denoted by W. Each of the rules require at most two parameters each time a buy, sell or hold signal is computed at some time period t. The parameters represent the number of short and long-term look-back periods of indicators used in the rules. We will denote the vector of short-term parameters by $\boldsymbol{n_1}$ and the long-term parameters by $\boldsymbol{n_2}$ which make up two of the four model parameters. Let $L = |\boldsymbol{n_1}|$ and $K = |\boldsymbol{n_2}|$[5] be the number of short-term and long-term look-back parameters respectively. Also, we denote the number of trading rules which utilise one parameter by W_1 and the number of trading rules utilising two parameters by W_2.

The final model parameter, denoted by $\mathbf{c}$, refers to object clusters where $\mathbf{c}(i)$ is the i^{th} object cluster and C is the number of object clusters. We will consider four object clusters ($C = 4$); the trivial cluster which contains all the stocks and the three major sectors of stocks on the JSE, namely, Resources, Industrials and Financials[6]. The algorithm will loop over all combinations of these four model parameters to create a buy, sell or hold signal at each time period t. Each combination of $\boldsymbol{\omega}(i)$, $\mathbf{c}(i)$, $\boldsymbol{n_1}(i)$ and $\boldsymbol{n_2}(i)$ will represent an expert. Thus, some experts may trade all the stocks (trivial clusters) and others subsets of the stocks (Resources, Industrials and Financials). It is important to note that for rules with 2 parameters, the loop over the long-term parameters will only activate at indices for which $\boldsymbol{n_1}(k) < \boldsymbol{n_2}(k)$. Here, k represents the loop index over the long-term parameters. The total number of experts, Ω, is then given by

$$\begin{aligned}\Omega &= \text{no. of experts with 1 parameter} \\ &\quad + \text{no. of experts with 2 parameter} \\ &= C \cdot L \cdot W_1 + C \cdot W_2 \cdot \sum \Big[\sum (\boldsymbol{n_2} > \max(\boldsymbol{n_1})) : \\ &\quad \sum (\boldsymbol{n_2} > \min(\boldsymbol{n_1}))\Big]\end{aligned}$$

We will denote each expert's strategy[7] by $\mathbf{h}_t^n$ which is an $(m + 1) \times 1$ vector representing the portfolio weights of the n^{th} expert for all m stocks and the risk-free asset at time t. Here, m refers to the chosen number of stocks to be passed into the expert generating algorithm. From the set of m stocks, it is important to note that each expert will not necessarily trade all m stocks (unless the expert trades the trivial cluster), since of those m stocks, only a hand full of stocks will fall into a given sector constituency. This implies that even though we specify each experts strategy ($\mathbf{h}_t^n$) to be an $(m + 1) \times 1$, we will just set the controls to zero for the stocks which the expert does not trade. Denote the expert control matrix by $\mathbf{H}_t$ made up of all n experts' strategies at time t for all m stocks i.e. $\mathbf{H}_t = [\mathbf{h}_t^1, \ldots, \mathbf{h}_t^n]$. In order to choose the m stocks to be traded, we take the m most liquid stocks over a specified of days, say δ_{liq} days. We use average daily volume (ADV) as a proxy for liquidity[8]. ADV is simply the average volume traded for a given stock over a period of time. The ADV for stock m over the past δ_{liq}

[4] This is also known as data mining

[5] $|\cdot|$ denotes the dimension of a vector

[6] See Appendix E E.1 for a breakdown of the three sectors into their constituents for daily data and Appendix E E.2 for the breakdown of intraday data

[7] When we refer to strategy, we are talking about the weights of the stocks in the expert's portfolio. We will also refer to these weights as controls.

[8] Other indicators of liquidity do exist such as the width of the bid-ask spread and market depth however ADV provides a simple approximation of liquidity.

periods is

$$\mathrm{ADV}_m = \frac{1}{\delta_{\mathrm{liq}}} \sum_{t=1}^{\delta_{\mathrm{liq}}} \mathrm{V}_t^m \tag{1}$$

where V_t^m is the volume of the m^{th} stock at period t. The m stocks with the largest ADV will then be fed into the algorithm for trading.

A.2. Transforming signals into weights

In this section we describe how each individual expert's set of trading signals at each time period t are transformed into a corresponding set of portfolio weights (controls) which constitute the expert's strategy ($\mathbf{h}_t^n$). As mentioned above, an expert will not necessarily trade all m stocks, however, for the purpose of generality we refer to the stocks traded by a given expert as m even though the weights of many of these m stocks will be zero for multiple periods as the expert will only be considering a subset of these stocks depending on which object cluster the expert trades.

Suppose it is currently time period t and the n^{th} expert is trading m stocks. Given that there are m stocks in the portfolio, m trading signals will need to be produced at each trading period. The risk-free assets purpose will be solely to balance the portfolio given the set of trading signals. Given the signals for the current time period t and previous period $t-1$, all hold signals at time t are replaced with the corresponding non-zero signals from time $t-1$ as the expert retains his position in these stocks[9]. All non-hold signals at time t are not replaced as the expert has taken a new position in the stock completely from the previous period. This implies that when the position in a given stock was sort at period $t-1$ for example and the current signal is long then he takes a long position in the stock rather than neutralising his position. Before computing the portfolio controls, we compute a combined signal vector made up of signals from time period $t-1$ and time t using the idea discussed above. We will refer to these combined set of signals as *output signals* for the purpose of brevity. We then consider four possible cases of the output signals at time t for a given expert:

1. All output signals are hold (0)
2. All output signals are non-negative (0 or 1)
3. All output signals are non-positive (0 or -1)
4. There are combinations of buy, sell and hold signals (0, 1 and -1) in the set of output signals

[9] Only the position is retained from the previous period (long/short) not the magnitude of the weight held in the stock

Due to the fact that cases 2 and 3 exist, we need to include a risk free asset in the portfolio so that we can enforce the self-financing constraint; the controls must sum to zero $\sum_i w_i = 0$. We refer to such portfolios as zero-cost portfolios. Additionally, we implement a leverage constraint by ensuring that the absolute value of the controls sum to unity: $\sum_i |w_i| = 1$.

To compute the controls for case 1, we set all stock weights and the risk free asset weight to zero so that the expert does not allocate any capital in this case since the output signals are all zero.

For case 2, we compute the standard deviations of the set of stocks which resulted in buy (positive) signals from the output signals using their closing prices over the last 120 trading periods (working days) in order to give a larger weight to the more volatile stocks. Let the number of buy signals from the set of output signals be denoted by nb and denote the vector of standard deviations of stocks to be bought by $\boldsymbol{\sigma}_+$. Then the weight allocated to stocks with positive signals is given by

$$\mathbf{w} = 0.5 \cdot \frac{1}{\sum_i \boldsymbol{\sigma}_+(i)} \cdot \boldsymbol{\sigma}_+ \tag{2}$$

where the lowest value of $\boldsymbol{\sigma}_+(i)$ corresponds to the least volatile stock and vice versa for large $\boldsymbol{\sigma}_+(i)$. This equation ensures that $\sum_i w_i = 0.5$. We then short the risk free asset with a weight of a half ($w_{rf} = -0.5$). This allows us to borrow using the risk free asset and purchase the corresponding stocks in which we are taking a long position.

Case 3 is similar to Case 2 above, however instead of having positive output signals, all output signals are negative. Again we compute standard deviations of the set of stocks which resulted in sell (negative) signals from the output signals using their closing prices over the last 120 trading periods. Let the number of sell signals from the set of output signals be denoted by ns and denote the vector of standard deviations of stocks to be sold by $\boldsymbol{\sigma}_-$. Then the weight allocated to stocks which have short positions is given by

$$\mathbf{w} = -0.5 \cdot \frac{1}{\sum_i \boldsymbol{\sigma}_-(i)} \cdot \boldsymbol{\sigma}_- \tag{3}$$

We then take a long position in the risk free asset with a weight of one half ($w_{rf} = 0.5$).

For case 4, we use the similar methodology to that discussed above in Case 2 and 3. To compute the weights for the short assets we use the formula

$$\mathbf{w} = -0.5 \cdot \frac{1}{\sum_i \boldsymbol{\sigma}_-(i)} \cdot \boldsymbol{\sigma}_- \tag{4}$$

Similarly, for the long assets we have

$$\mathbf{w} = 0.5 \cdot \frac{1}{\sum_i \boldsymbol{\sigma}_+(i)} \cdot \boldsymbol{\sigma}_+ \tag{5}$$

We then set the risk free rate to be equal to $\sum_i w_i$ in order to enforce the self-financing and fully invested constraints. Finally, for assets which had hold signals, the weights are set to zero.

In Section V, we implement 2 methods for transforming the signals into controls. The method described above is what we will refer to as the *volatility loading* method for transforming signals into controls and the second method we consider is called the *inverse volatility loading* method. The inverse volatility loading method is defined similarly to the method described above, however, instead of multiplying through by the volatility vector in each of the above cases, we multiply through by the inverse of the volatility vector (element-wise inverses). We will not implement the inverse volatility loading method in this study as the results are similar.

Appendix B Algorithm 1 shows the algorithm outline for the Expert Generating Algorithm. The Expert Generating Algorithm calls the *controls* function which transforms trading signals into portfolio controls. The *controls* function is made up of two parts, first to compute the output signals as discussed in Section II A A.2. which is outlined in Appendix B Algorithm 2 and the second part is to transform the output signals into portfolio controls which is outlined in Appendix B Algorithm 3.

B Online Learning Algorithm

Given that we now have a population of experts each with their own controls, $\mathbf{h}_t^n$, we implement the online learning algorithm to aggregate the experts strategies at time t based on their performance and form a final single portfolio to be used in the following period $t+1$ which we denote $\mathbf{b}_t$. The aggregation scheme used is inspired by the Universal Portfolio (UP) strategy taken from the work done by [2, 4] and a modified version proposed by [5]. Although, due to the fact that we have several different base experts as defined by the different trading strategies rather than Cover's [2] constant rebalanced UP strategy, our algorithm is better defined as a meta-learning algorithm [24]. We use the subscript t since the portfolio is created using information only available at time t even though the portfolio is implemented in the following time period. The algorithm will run from the initial time t_{min}[10] which is taken to be 2 until terminal time T. t_{min} is required to ensure there is sufficient data to compute a return for the first active trading day. We must point out here that experts will only actively begin making trading decisions once there is sufficient data to satisfy their look-back parameter(s) and subsequently, since the shortest look-back parameter is 4 periods, the first trading decisions will only be made during day 5. The idea is to take in m stock's OHLCV values at each time period which we will denote by $\mathbf{X}_t$. We then compute the price relatives at each time period t given by $\mathbf{x}_t = (x_{1,t}, \ldots, x_{m,t})$ where $x_{m,t} = \frac{P^c_{m,t}}{P^c_{m,t-1}}$ and where $P^c_{m,t}$ is the closing price of stock m at time period t. Expert controls are generated from the price relatives for the current period t to form the expert control matrix $\mathbf{H}_t$. From the corresponding expert control matrix, the algorithm will then compute the expert performance $\mathbf{Sh}_t$ which is the associated wealth of all n experts at time t. Denote the n^{th} expert's wealth at time t by Sh_t^n. We then form the final aggregated portfolio, denoted by $\mathbf{b}_t$, by aggregating the expert's wealth using the agent mixture update rules.

The relatively simplistic learning algorithm is incrementally implemented online but offline it can be parallelised across experts. Given the expert controls from the Expert Generating Algorithm ($\mathbf{H}_t$), the online learning algorithm is implemented by carrying out the following steps [6]:

1. **Update portfolio wealth**: Given the portfolio control $b_{m,t-1}$ for the m^{th} asset at time $t-1$, we update the portfolio wealth for the t^{th} period

$$\Delta S_t = \sum_{m=1}^{M} b_{m,t-1}(x_{m,t+1} - 1) + 1 \tag{6}$$

$$S_t = S_{t-1}\Delta S_t \tag{7}$$

   S_t represents the compounded cumulative wealth of the overall aggregate portfolio and $\mathbf{S} = S_1, \ldots, S_t$ will denote the corresponding vector of aggregate portfolio wealth's over time. Here the realised price relatives for the t^{th} period and the m^{th} asset, $x_{m,t}$, are combined with the portfolio controls for the previous period to obtain the realised portfolio returns for the current period t. $\Delta S_t - 1$ is in fact the profits and losses for the current trading period t. Thus, we will use it to update the algorithms overall cumulative profits and losses which is given by

$$\mathrm{PL}_t = \mathrm{PL}_{t-1} + \Delta S_t - 1 \tag{8}$$

2. **Update expert wealth**: The expert controls $\mathbf{H}_t$ were determined at the end of time-period $t-1$ for time period t by the expert generating algorithm for Ω experts and M objects about which the experts make expert capital allocation decisions. At then end of the t^{th} time period the performance of each expert n, Sh_t^n, can be computed from the change

[10] this is the time at which trading commences

in the price relatives $x_{m,t}$ for the each of the M objects in the investment universe considered using the closing prices at the start, $P^c_{m,t-1}$, and the end of the t^{th} time increment, $P^c_{m,t}$, using the expert controls.

$$\Delta Sh^n_t = \left[\sum_{m=1}^{M} \mathbf{h}^n_t (x_{m,t} - 1)\right] + 1 \quad (9)$$

$$Sh^n_t = Sh^n_{t-1} \cdot \Delta Sh^n_t \quad (10)$$

3. **Update expert mixtures**: We consider a UP expert mixture update rule and will generically refer to the online update as a function g.

   3.1. *Universal Portfolio* [2, 5, 6] inspired mixture controls: the mixture of the n^{th} expert for the next time increment, $t+1$, is equivalent to the accumulated expert wealth up until time t and will be used as the update feature for the next unrealised increment subsequent appropriate normalisation

$$q_{n,t+1} = Sh^n_t \quad (11)$$

4. **Re-normalise expert mixtures**: As mentioned previously, we will consider experts such that the leverage is set to unity for zero-cost portfolios: 1.) $\sum_n q_n = 0$ and 2.) $\nu = \sum_n |q_n| = 1$. We will not consider the long-only experts (absolute experts as in Loonat and Gebbie [6]), but only consider experts whom satisfy the prior two conditions which we will refer to as *active* experts. This in fact allows for shorting of one expert against another; then due to the nature of the mixture controls the resulting portfolio becomes self-funding.

$$q_{n,t+1} = \frac{q_{n,t+1} - \frac{1}{N}\sum_{n=1}^{\Omega} q_{n,t+1}}{\sum_{n=1}^{\Omega} |q_{n,t+1} - \frac{1}{N}\sum_{n=1}^{\Omega} q_{n,t+1}|} \quad (12)$$

5. **Update portfolio controls**: The portfolio controls $b_{m,t}$ are updated at the end of time period t for time period $t+1$ using the expert mixture controls $q_{n,t+1}$ from the updated learning algorithm and the vector of expert controls $\mathbf{h}^n_t$ for each expert n from the expert generating algorithms using information from time period t and average over all n experts.

$$b_{m,t+1} = \sum_n q_{n,t+1} \mathbf{h}^n_t \quad (13)$$

The strategy is to implement the portfolio controls, wait until the end of the increment, measure the features (OHLCV values), update the experts and then re-apply the learning algorithm to compute the expert mixtures and portfolio controls for the next time increment. More details about the various components of the algorithm are provided in Appendix B.

C Algorithm implementation for intraday-daily trading

Intraday trading poses a whole set of new issues that need to be considered and isn't as easy as simply plugging the data into the algorithm and treating it as if it is similar to daily data just sampled at more regular intervals[11]. Trading on an intraday time scale contains very different dynamics to trading on a daily time scale.

We implement the learning algorithm on a combinations of daily and intraday data whereby decisions made on the daily time scale are made completely independent of those made on the intraday time scale but the dynamics of the associated wealth's generated by the processes are fused together. The best way to think about it is to consider the experts as trading throughout the day, making decisions based solely on intraday data while compounding their wealth, and once a trading decision is made at the final time bar, the expert makes one last trading decision on that day based on daily historic OHLCV data (the look-back periods will be based on passed trading days and not on time bars for that day). The daily trading decision can be thought of as just being the last time bar of the day where we are just using different data to make the decision. The methodology for each of the intraday and daily trading mechanisms are almost exactly as explained in Section II B above however there are a couple of alterations to the algorithm. As in the daily data implementation, a given expert will begin making trading decisions as soon as there is a sufficient amount of data available to them. The idea is to begin the algorithm from day two so that there will be sufficient data to compute a return on the daily time scale. We then loop over the intraday time bars from 9:15am to 4:30pm on each given day.

We will refer to trading using a combination of daily and intraday data as *intraday-daily* trading. To introduce some notation for intraday-daily trading, let $\mathbf{Sh}^F_{t,t_I}$[12] be the expert wealth vector for all n experts for the t^{th}_I time bar on the t^{th} day and denote by $\mathbf{H}^F_{t,t_I}$ the associated expert control matrix. The superscript F refers to the 'fused' daily and intraday matrices. More specifically, $\mathbf{H}^F_{t,t_I}$ will contain the 88 intraday expert

[11] The main issue with this approach is the deviation in the prices at the end of day t and the start of day $t-1$. Often the deviation is significant which causes returns to blow up.

[12] Expert wealth is computed as before with $SH^F_{n,t,t_I} = SH^F_{n,t,t_I-1} \cdot dSh^F_{n,t,t_I}$ where dSh^F_{n,t,t_I} is the n^{th} experts return at time bar t_I on day t.

controls followed by the end of day expert controls based on daily closing OHLCV data for each given day over the trading horizon. Denote T_I as the final time bar in a day (4:30pm). An experts wealth accumulated up until the final time bar T_I[13] on day t, Sh^F_{n,t,T_I+1}, is calculated from the n^{th} column of the expert control matrix, denoted $\mathbf{h}^F_{n,t,T_I}$, from the previous period T_I and is computed solely from intraday data. Overall portfolio controls for the final intraday trade on day t ($\mathbf{b}_{t,T_I+1}$) are computed as before along with the overall portfolio wealth S_{t,T_I+1}. This position is held until the close of the days trading when the closing prices of the m stocks $\mathbf{P}^C_t$ is revealed. Once the closing prices are realised, the final intraday position is closed. That is, an offsetting trade of $-\mathbf{b}_{t,T_I+1}$ is made at the prices $\mathbf{P}^C_t$. This profit/loss is then compounded onto S_{t,T_I+1}. Thus, no intraday positions are held overnight. The experts will then make one final trading decision based on the daily OHLCV data given that the closing price is revealed and will look-back on daily historic OHLCV data to make these decisions. The n^{th} expert's wealth Sh^F_{n,t,T_I+2} is updated using controls $\mathbf{h}^F_{n,t,T_I+1}$. The corresponding portfolio controls for all m stocks are computed for the daily trading decision on day t to be implemented at time $t+1$ ($\mathbf{b}_{t+1}$), the returns (price relatives) for day t is computed ($\mathbf{r}_t = \frac{\mathbf{P}^C_t}{\mathbf{P}^C_{t-1}}$) and the wealth is compounded onto the cumulative wealth achieved is $S_{T_I+2} = S_{t,T_I+1} \cdot (\mathbf{b}_t \cdot (\mathbf{x}_t - 1) + 1)$ where $S_{t,T_I+1} = S_{t,T_I} \cdot (\mathbf{b}_{T_I}(\mathbf{x}_{T_I} - 1))$ with $\mathbf{r}_{T_I} = \frac{\mathbf{P}^C_{T_I}}{\mathbf{P}^C_{T_I-1}}$. The daily position $\mathbf{b}_{t+1}$ is then held until the end of the following day or possibly further into the future (until new daily data portfolio allocations are made). This completes trading for day t.

At the beginning of day $t+1$, the expert wealth $Sh^F_{n,t+1,1}$[14] is set back to 1. Setting experts wealth back to 1 at the beginning of the day rather than compounding on the wealth from the previous day is due to the fact that learning on intraday data between days is not possible due to the fact that conditions in the market have completely changed. Trading will begin by computing expert controls $\mathbf{h}^F_{n,t+1,2}$ for the second time bar, however all experts will not have enough data to begin trading since the shortest look-back parameter is 3 and hence controls will all be set to zero. As the trading day proceeds, experts will begin producing non-zero controls as soon as they have enough data to satisfy the amount of data needed for a given look-back parameter. Something to note here is that due to the fact that the STeFI index[15] (risk-free asset) is only posted daily, we utilise the same STeFI value for trading throughout the day. Finally, in order to differentiate between daily OHLCV data and intraday OHLCV data, we will denote them as $\mathbf{X}_d$ and $\mathbf{X}_I$ respectively. The algorithm outline for the intraday-daily algorithm is illustrated in Appendix B Algorithm 5.

[13] T_I will always be equal to 88 as there are 88 5 minute time bars between 9:15am and 4:30pm

[14] We do not start the trading day at the first time bar $t_I = 1$ since we need to compute a return which requires 2 data points.

D Online Portfolio Benchmark Algorithm

To get an idea of how well our online algorithm performs, we compare it's performance to that of the offline BCRP. As mentioned previously, the hindsight CRP strategy chosen which gives the maximum return of all such strategies in the long run. To find the portfolio controls of such a strategy, we perform a brute force Monte Carlo approach to generate 5000 random CRP strategies on the entire history of price relatives and choose the BCRP strategy to be the one that returns the maximal terminal portfolio wealth. As a note here, the CRP strategies we consider are long-only.

E Transaction Costs and Market Frictions

Apart from the (direct) transaction fees (commissions) charged by exchanges for the trading of stocks, there are various other costs (indirect) that need to be considered when trading such assets. Each time a stock is bought or sold there are unavoidable costs and it is imperative that a trader takes into account these costs. The other three most important component of these transaction costs besides commissions charged by exchanges are the spread[16], price impact and opportunity cost [26].

To estimate indirect transaction costs (TC) for each period t, we will consider is the *square-root formula* [27]

$$\text{TC} = \text{Spread} + \sigma.\sqrt{\frac{n}{\text{ADV}}} \tag{14}$$

where:

1. **Volatility of the returns of a stock (σ)**: See Section II E E.1. below.

2. **Average daily volume of the stock (ADV)**: ADV is computed using the previous 90 days trading volumes for daily trading and the previous 5

[15] See Section IV for more details on the STeFI index

[16] spread = best ask price minus best bid price

days intraday trading volume for intraday-daily trading.

3. **Number of shares traded (n)**: The number of shares traded (n) is taken to be five tenths of a basis point of ADV for each stock per day for daily trading. The number of stocks traded for intraday-daily trading is 1% of ADV for the entire portfolio per day which is then split evenly among all active trading periods during the day and between all 15 stocks to arrive at a final value of 0.00012 (1%/85/15) of ADV per stock per trading period.

4. **Spread**: Spread is assumed to be 1bps per day (1%% /pd) for daily trading. For intraday-daily trading, we assume 12bps per day[17] which we then split evenly over the day to incur a cost of 0.0012/85bps per time bar.

The use of the square-root rule in practice dates back many years and is used as a pre-trade transaction cost estimate [27]. The first term in Eq. (14) can be regarded as the term representing the *slippage*[18] or *temporary price impact* and results due to our demand for liquidity [28]. This cost will only impact the price at which we execute our trade at and not the market price (hence subsequent transactions). The second term in Eq. (14) is the (transient) price impact which will not only affect the price of the first transaction but also the price of subsequent transactions by other traders in the market, however the impact decays over time as a power-law [29]. In the following subsection, we will discuss how the volatility (σ) is estimated for the square-root formula. Technically, σ, n and ADV in Eq. (14) should each be defined by a vector representing the volatilities, number of stocks traded and ADV of each stock in the portfolio respectively however for the sake of generality we will write it as a constant thus representing the volatility for a single portfolio stock.

In addition to the indirect costs associated with slippage and price impact as accounted for by the square-root formula, we include direct costs such as the borrowing of trading capital, the cost of regulatory capital and the various fees associated with trading on the JSE [6]. Such costs will also account for small fees incurred in incidences where short-selling has taken place. For the daily data implementation, we assume a total direct cost of 4bps per day. For the intraday-daily implementation a total direct cost of 70bps per day is assumed (following Loonat and Gebbie [6]) which we then split evenly over each days active trading periods (85 time bars since first expert only starts trading after the 5^{th} time bar) to get a cost of 70bps/85 per period.

For daily trading, we recover an average daily transaction cost of roughly 9.38bps which is approximately the same as the assumed 10bps by Loonat and Gebbie [6]. Loonat and Gebbie argue that for intraday trading, it is difficult to avoid a direct and indirect cost of about 50-80bps per day in each case leaving a conservative estimate of total costs to be approximately 160bps per day. We realise an overall average cost per period of 1.5bps while the average cost per day assuming we trade for 85 periods throughout each day is roughly 130bps (85*1.5) for intraday-daily trading.

E.1. Volatility Estimation for Transaction Costs

In this section we will discuss different methods for calculating the estimates for volatility (σ) for daily and intraday data in the square-root formula (Eq. (14)).

1.a. Daily Data Estimation

The volatility of daily prices at each day t is taken to be the standard deviation of closing prices over the last 90 days. If 90 days have not passed then the standard deviation will be taken over the number of days available so far.

1.b. Intraday Data Estimation

The volatility for each intraday time bar t_I on day t is dependent on the time of day. For the first 15 time bars, the volatility is taken to be a forecast of a GARCH(1,1) model which has been fitted on the last 60 returns of the previous day $t-1$. The reason for this choice is that the market is very volatile during the opening hour as well as the fact that there will be relatively few data points to utilise when computing the volatility. The rest of the days volatility estimates are computed using the Realized Volatility (RV) method [30]. RV is one of the more popular methods for estimating volatility of high-frequency returns[19] computed from tick data. The measure estimates volatility by summing up intraday squared returns at short intervals (eg. 5 minutes). Andersen *et al* [30] propose this estimate for volatility at higher frequencies and derive it by showing that RV is an approximate of quadratic variation under the assumption that log returns are a continuous time stochastic process with zero mean and no jumps. The idea is to show that the RV converges to the continuous

[17] We follow the conservative approach taken by Loonat and Gebbie [6] as opposed to the moderate approach where 9bps per day is assumed

[18] Slippage is often calculated as the difference between the price at which an order for a stock is placed and the price at which the trade is executed

[19] Most commonly refers to returns over intervals shorter than one day. This could be minutes, seconds or even milliseconds.

time volatility (quadratic variation) [31], which we will now demonstrate.

Assume that the instantaneous returns of observed log stock prices (p_t) with unobservant latent volatility (σ_t) scaled continuously through time by a standard Wiener process (dW_t) can be generated by the continuous time martingale [31]

$$dp_t = \sigma_t dW_t \tag{15}$$

The it follows that the conditional variance of the single period returns, $r_{t+1} = p_{t+1} - p_t$ is given by

$$\sigma_t^2 = \int_t^{t+1} \sigma_s^2 ds \tag{16}$$

Eq. (16) is also known as the *integrated volatility* for the period t to $t+1$.

Suppose the sampling frequency of the tick data into regularly spaced time intervals is denoted by f such that between period $t-1$ and t there are f continuously compounded returns. Then

$$r_{t+1/f} = p_{t+1/f} - p_t \tag{17}$$

Hence, we get the *Realised Volatility* (RV) based on f intraday returns between periods $t+1$ and t as

$$RV_{t+1} = \sum_{i=1}^{f} r_{t+i/f}^2 \tag{18}$$

The argument here is that provided we sample at frequent enough time steps (f), the volatility can be observed theoretically from the sample path of the return process and hence [31, 32]

$$\lim_{f\to\infty}\left(\int_t^{t+1} \sigma_s^2 ds - \sum_{i=1}^{f} r_{t+i/f}^2\right) = 0 \tag{19}$$

which says that the RV of a sequence of returns asymptotically approaches the integrated volatility and hence the RV is a reasonable estimate of current volatility levels.

III TESTING FOR STATISTICAL ARBITRAGE

To test the overall trading strategy for statistical arbitrage, we implement a novel statistical test originally proposed by [20] and later modified by [23] by applying it to the overall strategy's profit and losses **PL**. The idea is to axiomatically define the conditions under which a statistical arbitrage exists and assume a parametric model for incremental trading profits in order to form a null hypothesis derived from the union of several sub-hypotheses which are formulated to facilitate empirical tests of statistical arbitrage. The modified test, proposed by [23], called the Min-t test, is derived from a set of restrictions imposed on the parameters defined by the statistical arbitrage null hypothesis and is applied to a given trading strategy to test for statistical arbitrage. The Min-t statistic is argued to provide a much more efficient and powerful statistical test compared to the Bonferroni inequality used in [20]. The lack of statistical power is reduced when the number of sub-hypotheses increases and as a result, the Bonferroni approach is unable to reject an incorrect null hypothesis leading to a large Type II error.

To set the scene and introduce the concept of a statistical arbitrage, suppose that in some economy, a stock (portfolio)[20] s_t and a money market account B_t[21] are traded. Let the stochastic process $(x(t), y(t) : t \geq 0)$ represent a zero initial cost trading strategy that trades $x(t)$ units of some portfolio (s_t) and $y(t)$ units of a money market account at a given time t. Denote the cumulative trading profits at time t by V_t. Let the time series of discounted cumulative trading profits generated by the trading strategy be denoted by $\nu(t_1), \nu(t_2), \ldots, \nu(t_T)$ where $\nu(t_i) = \frac{V_{t_i}}{B_{t_i}}$ for each $i = 1, \ldots, T$. Denote the increments of the discounted cumulative profits at each time i by $\Delta\nu_i = \nu(t_i) - \nu(t_{i-1})$. Then, a statistical arbitrage is defined as:

Definition 1 (Statistical Arbitrage [20, 23])**.** A *statistical arbitrage* is a zero-cost, self-financing trading strategy $(x(t) : t \geq 0)$ with cumulative discounted trading profits $\nu(t)$ such that

1. $\nu(0) = 0$
2. $\lim_{t\to\infty} \mathbb{E}^{\mathrm{P}}[\nu(t)] > 0$
3. $\lim_{t\to\infty} \mathrm{P}[\nu(t) < 0] = 0$
4. $\lim_{t\to\infty} Var[\Delta\nu(t) | \Delta\nu(t) < 0] = 0$

In other words, a statistical arbitrage is a trading strategy that 1) has zero initial cost, 2) in the limit has positive expected discounted cumulative profits, 3) in the limit has a probability of loss that converges to zero and 4) variance of negative incremental trading profits converge to zero in the limit. It is clear that deterministic arbitrage stemming from traditional financial mathematics is in fact a special case of statistical arbitrage [33].

---

[20] In our study, we will be considering a portfolio

[21] The money market account is initialised at one unit of a currency i.e. $B_0 = 1$.

In order to test for statistical arbitrage, assume that the incremental discounted trading profits evolve over time according to the process

$$\Delta\nu_i = \mu i^{\theta} + \sigma i^{\lambda} z_i \tag{20}$$

where $i = 1, \ldots, T$. There are two cases to consider for the innovations: 1) z_i i.i.d N(0,1) normal uncorrelated random variables satisfying $z_0 = 0$ or 2) z_i follows an MA(1) process given by:

$$z_i = \epsilon_i + \phi\epsilon_{i-1} \tag{21}$$

in which case the innovations are non-normal and correlated. Here, ϵ_i are i.i.d. N(0,1) normal uncorrelated random variables. It is also assumed that and $\Delta\nu_0 = 0$ and in case of our algorithm $\nu_{t_{min}} = 0$. We will refer the first model as the unconstrained mean (UM) model and the second model as the and unconstrained mean with correlation (UMC) model. Furthermore, we refer to the model with $\theta = 0$ as the constrained mean (CM) which assumes constant incremental profits over time and hence has an incremental profit process given by:

$$\Delta\nu_i = \mu + \sigma i^{\lambda} z_i \tag{22}$$

The discounted cumulative trading profits for the UM model at terminal time T discounted back to the initial time which are generated by a trading strategy are given by

$$\nu(T) = \sum_{i=1}^{T} \Delta\nu_i \sim N\left(\mu \sum_{i=1}^{T} i^{\theta}, \sigma^2 \sum_{i=1}^{T} i^{2\lambda}\right) \tag{23}$$

From Eq. (23), it is straightforward to show that the log-likelihood function for the discounted incremental trading profits is given by

$$\begin{aligned}\ell(\mu, \sigma^2, \lambda, \theta|\Delta\nu) &= \log L(\mu, \sigma^2, \lambda, \theta|\Delta\nu) \\ &= -\frac{1}{2}\sum_{i=1}^{T} \log(\sigma^2 i^{2\lambda}) \\ &\quad - \frac{1}{2\sigma^2}\sum_{i=1}^{T} \frac{1}{i^{2\lambda}}(\Delta\nu_i - \mu i^{\theta})^2 \end{aligned} \tag{24}$$

The probability of a trading strategy generating a loss after n periods is as follows [23]

$$\mathbf{Pr}\{\text{Loss after } n \text{ periods}\} = \Phi\left(\frac{-\mu\sum_{i=1}^{n} i^{\theta}}{\sigma(1+\phi)\sqrt{\sum_{i=1}^{n} i^{2\lambda}}}\right) \tag{25}$$

where $\Phi(\cdot)$ denotes the cumulative standard normal distribution function. For the CM model, Eq. (25) is easily adjusted by setting ϕ and θ equal to zero. This probability converges to zero at a rate that is faster than exponential.

As mentioned previously, to facilitate empirical tests of statistical arbitrage under Definition 1, a set of sub-hypotheses are formulated to impose a set of restrictions on the parameters of the underlying process driving discounted cumulative incremental trading profits and are as follows:

Proposition 1 (UM Model Hypothesis [23]). *Under the four axioms defined in Definition 1, a trading strategy generates a statistical arbitrage under the UM model if the discounted incremental trading profits satisfy the intersection of the following four sub-hypotheses jointly:*

1. $H_1 : \mu > 0$
2. $H_2 : -\lambda > 0$ *or* $\theta - \lambda > 0$
3. $H_3 : \theta - \lambda + \frac{1}{2} > 0$
4. $H_4 : \theta + 1 > 0$

An intersection of the above sub-hypotheses defines a statistical arbitrage and as by De Morgan's Laws[22], the null hypothesis of no statistical arbitrage is defined by a union of the sub-hypotheses. Hence, the no statistical arbitrage null hypothesis is the set of sub-hypotheses which are taken to be the complement of each of the sub-hypotheses in Proposition 1:

Proposition 2 (UM Model Alternative Hypothesis [20, 23]). *Under the four axioms defined in Definition 1, a trading strategy does not generate a statistical arbitrage if the discounted incremental trading profits satisfy any one of the following four sub-hypotheses:*

1. $H_1 : \mu \leq 0$
2. $H_2 : -\lambda \leq 0$ *or* $\theta - \lambda \leq 0$
3. $H_3 : \theta - \lambda + \frac{1}{2} \leq 0$
4. $H_4 : \theta + 1 \leq 0$

The null hypothesis is not rejected provided that a single sub-hypothesis holds. The Min-t test is then used to test the above null hypothesis of no statistical arbitrage by considering each sub-hypothesis separately using the t-statistics $t(\hat{\mu}), t(-\hat{\lambda}), t(\hat{\theta}-\hat{\lambda}), t(\hat{\theta}-\hat{\lambda}+0.5)$, and $t(\hat{\theta}+1)$ where the hats denote the Maximum Likelihood Estimates (MLE) of the parameters. The Min-t statistic is defined as [23]

$$\begin{aligned}\text{Min-}t = \text{Min}\{&t(\hat{\mu}), t(\hat{\theta}-\hat{\lambda}), t(\hat{\theta}-\hat{\lambda}+0.5), \\ &\text{Max}[t(-\hat{\lambda}), t(\hat{\theta}+1)]\}\end{aligned} \tag{26}$$

[22] This states that the complement of the intersection of sets is the same as the union of their complements.

The intuition is that the Min-t statistic returns the smallest test statistic which is the sub-hypothesis which is closest to being accepted. The no statistical arbitrage null is then rejected if Min-$t > t_c$ where t_c depends on the significance level of the test which we will refer to as α. Since the probability of rejecting cannot exceed the significance level α, we have the following condition for the probability of rejecting the null at the α significance level

$$\mathbf{Pr}\{\text{Min-}t > t_c | \mu, \lambda, \theta, \sigma\} \leq \alpha \tag{27}$$

What remains is for us to compute the critical value t_c. We will implement a Monte Carlo simulation procedure to compute t_c which we describe in more detail in Section III A step 5 below.

A Outline of the Statistical Arbitrage Test Procedure

The steps involved in testing for statistical arbitrage are outlined below:

1. **Trading increments** $\Delta\nu_i$: From the vector of cumulative trading profits and losses, compute the increments $(\Delta\nu_1, \ldots, \Delta\nu_T)$ where $\Delta\nu_i = \nu(t_i) - \nu(t_{i-1})$.

2. **Perform MLE**: Compute the likelihood function as given in Eq. (24) and maximize it to find the estimates of the four parameters, namely, $\hat{\mu}, \hat{\sigma}, \hat{\theta}$ and $\hat{\lambda}$. The log-likelihood function will obviously be adjusted depending on whether the CM ($\theta = 0$) or UM test is implemented. We will only consider the CM test in this study. Since MATLAB's built-in constrained optimization algorithm[23] only performs minimization, we minimize the negative of the log-likelihood function.

3. **Standard errors**: From the estimated parameters in the MLE step above, compute the negative Hessian estimated at the MLE estimates which is indeed the Fisher Information (FI) matrix denoted by $\mathbf{I}(\Theta)$. In order to compute the Hessian, the analytical partial derivatives are derived from Eq. (24). Standard errors are then taken to be the square roots of the diagonal elements of the inverse of $\mathbf{I}(\Theta)$ since the inverse of the Fisher information matrix is an asymptotic estimator of the covariance matrix.

4. **Min-t statistic**: Compute the t-statistics for each of the sub-hypotheses which are given by $t(\hat{\mu}), t(-\hat{\lambda}), t(\hat{\theta} - \hat{\lambda}), t(\hat{\theta} - \hat{\lambda} + 0.5)$, and $t(\hat{\theta} + 1)$ and hence the resulting Min-t statistic given by Eq. (26). Obviously $t(\hat{\theta} - \hat{\lambda})$, $t(\hat{\theta} - \hat{\lambda} + 0.5)$ and $t(\hat{\theta} + 1)$ will not need to be considered for the CM test.

5. **Critical values**: Compute the critical value at the α significance level using the Monte Carlo procedure (uncorrelated normal errors) and Bootstrapping (correlated non-normal errors)

   (a) **CM model**
   First, simulate 5000 different profit process using Eq. (22) with $(\mu, \lambda, \sigma^2) = (0, 0, 0.01)$[24]. For each of the 5000 profit processes, perform MLE to get estimated parameters, the associated t-statistics and finally the Min-t statistics. t_c is the taken to be the 1-α quantile of the resulting distribution of Min-t values.

6. **P-values**: Compute the empirical probability of rejecting the null hypothesis at the α significance level using Eq. (27) by utilising the critical value from the previous step and the simulated Min-t statistics.

7. **n-Period Probability of Loss**: Compute probability of loss after n periods for each $n = 1, \ldots, T$ and observe the number of trading periods it takes for the probability of loss to converge to zero (or below 5% as in the literature). This is done by computing the MLE estimates for the vector $(\Delta\nu_1, \Delta\nu_2, \ldots \Delta\nu_n)$ for each given n and substituting these estimates into Eq. (25).

IV THE DATA

The two data sets utilised for the purpose of back-testing the online algorithm are outlined below.

A Daily Data

The daily data is sourced from Thomson Reuters and contains data corresponding to all stocks listed on the JSE Top 40[25]. The data set consists of data for 42 stocks over the period 01-01-2005 to 29-04-2016 however we will only utilise the stocks which traded more than 60% of the time over this period. Removing such stocks leaves us with a total of 31 stocks. The data comprises of the opening price (P^o), closing price (P^c), lowest price (P^l), the highest price (P^h) and daily traded volume (V)

[23] Here we are referring to MATLAB's *fmincon* function

[24] t_c is maximized when μ and λ are zero. σ^2 is set equal to 0.01 to approximate the empirical MLE estimate [23].

[25] Please refer to Appendix E E.1 for the full names of the Bloomberg ticker symbols.

(OHLCV). In additions to these 31 stocks, we also require a risk-free asset for balancing the portfolio. We make the choice of trading the Short Term Fixed Interest (STeFI) index. The STeFI benchmark is a proprietary index that measures the performance of Short Term Fixed Interest or money market investment instruments in South Africa. It is constructed by Alexander Forbes (and formerly by the South African Futures Exchange (SAFEX)) and has become the industry benchmark for short-term cash equivalent investments (up to 12 months) [34].

B Intraday-Daily Data

Bloomberg is the source of all tick (intraday) data used in this paper. The data set consists of 30 of the Top 40 stocks on the JSE from 02-01-2018 to 29-06-2018. The data is then sampled at 5 minute intervals to create an OHLCV entry for all 5 minute intervals over the 6 month period. We remove the first 10 minutes and last 20 minutes of the continuous trading session (9:00-16:50) as the market is relatively illiquid and volatile during these times which may lead to spurious trade decisions. We are thus left with 88 OHLCV entries for each stock on any given day. In addition to the intraday data, daily OHLCV data for the specified period is required for the last transaction on any given day. Again, we make use of the STeFI index as the risk-free asset. The data was sourced from a Bloomberg terminal using the R Bloomberg API, *Rblpapi*, and all data processing is done in MATLAB to get the data into the required form for the learning algorithm.

V RESULTS AND ANALYSIS

A Daily Data

In the section below, we implement the various algorithms described above in order to plot a series of graphs for daily JSE Top 40 data as discussed in Section IV above. We will plot five different graphs: first is the overall portfolio wealth over time which corresponds to S_t as described above, second, the cumulative profit and losses over time PL_t, third, the overall portfolio controls corresponds to $\mathbf{b}_t$ over time, fourth, the relative population wealth of experts corresponds to the wealth accumulated over time by each of the experts competing for wealth in the algorithm $\mathbf{Sh}_t$ and finally, the relative population wealth of the strategies takes the mean over all experts for each given trading strategy to get an accumulated wealth path for each technical trading rule.

For the purpose of testing the learning algorithm, we will identify the 15 most liquid stocks over one year prior to the start of active trading. The stocks, ranked by liquidity, are as follows: FSRJ.J, OMLJ.J, CFRJ.J, MTNJ.J, SLMJ.J, NTCJ.J, BILJ.J, SBKJ.J, WHLJ.J, AGLJ.J, SOLJ.J, GRTJ.J, INPJ.J, MNDJ.J and RMHJ.J.

A.1. No Transaction Costs

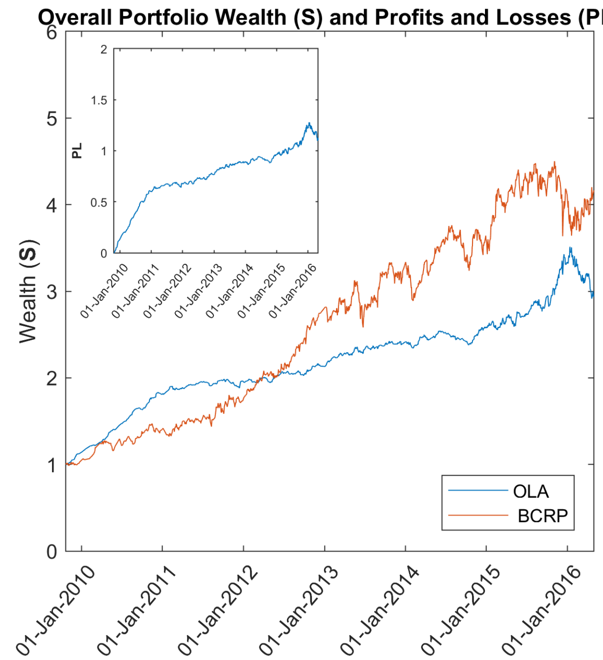


Figure 1: Overall cumulative portfolio wealth (S) for daily data with no transaction costs (blue) and the benchmark BCRP strategy (orange). The figure inset illustrate the associated profits and losses (**PL**) of the strategy.

Barring transaction costs, it's clear that the portfolio makes a favourable cumulative returns on equity over the 6 year period as is evident in Figure 1. The performance of the algorithm is not too far off that of the benchmark BCRP strategy which is promising as the original literature proves that the algorithm should track such a benchmark. The figure inset in Figure 1 illustrates that the provides consistent positive trading profits over the entire trading horizon. Figure 2(a) shows the expert wealth for all Ω experts and Figure 2(b) shows the mean expert wealth for each strategy. These figures illustrate that on average, the underlying experts perform fairly poorly compared to the overall strategy however there is evidence that some experts make satisfactory returns over the period.

Table II and Table III provide the group summary statistics of the terminal wealth's of experts and of the expert's profits and losses over the entire trading horizon respectively where experts are grouped based on their

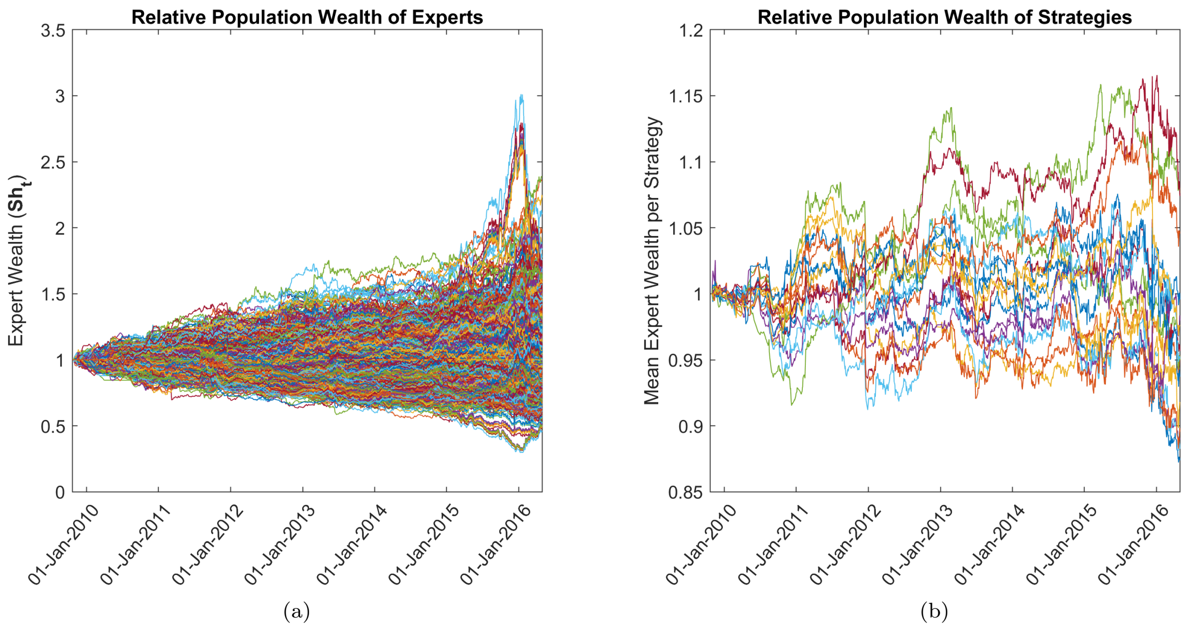


Figure 2: Figure 2(a) illustrates the expert wealth (**Sh**) for all Ω experts for daily data with no transaction costs. Figure 2(b) illustrates the mean expert wealth of all experts for each trading strategy ($\boldsymbol{\omega}(i)$) for daily data with no transaction costs.

underlying strategy ($\boldsymbol{\omega}(i)$). The online Z-Anticor[26] algorithm produces the best expert (maximum terminal wealth) followed closely by the slow stochastic rule while Z-Anticor also produces experts with the greatest mean terminal wealth over all experts (column 2). Additionally, Z-Anticor produces expert's with wealth's that vary the most (highest standard deviation). Williams %R produces the worst expert by quite a long way (minimum terminal wealth). The trading rule with the worst mean terminal wealth and worst mean ranking are SAR and slow stochastic respectively. With regards to profits and losses (Table III), the momentum rule (MOM) produces the expert with the greatest profit in a single period. SAR followed by Anti-Z-BCRP produce the worst and second worst mean profit/loss per trading period respectively whereas Z-Anticor and Z-BCRP achieve the best mean profit/loss per trading period.

Figure 3(a) illustrates the 2-D plot of the latent space of a Variational Autoencoder (VAE) for the time series' of wealth's of all the experts and is coloured by object cluster. It is not surprising that the expert's wealth time series' show quite well defined clusters in terms of the stock which experts trade in their portfolio as the stocks that each expert trades will be directly related to the decisions they make given the incoming data and hence the corresponding returns (wealth) they achieve.

[26] Please refer to Appendix C and Appendix D for a detailed description of the various trading rules mentioned in the tables Table II and Table III.

To provide some sort of comparison, in Figure 3(b) we plot the same results as above but this time we colour the experts in terms of their underlying strategy $\boldsymbol{\omega}(i)$. The VAE seems to be able to pick up much clearer similarities (dissimilarities) between the experts based on the stocks they trade compared to which strategy they utilise providing evidence that the achieved wealth has a much stronger dependence on the stock choice rather than the chosen strategy. This may be an important point to consider and gives an indication that it may be worth considering more sophisticated ways to choose the stocks to trade rather than developing more (profitable) strategies. A discussion on the features that should be considered by a quantitative investment manager in assessing an assets usefulness is provided in Samo and Hendricks [35].

Next, we implement the CM test for statistical arbitrage on the daily cumulative profits and losses (**PL**) for the strategy without transaction costs. In order to have a result that is synonymous with [23], we choose a period of 400 days to test our strategy. We test the realised profits and losses for the 400 day period stretching from the 30^{th} trading day until the 430^{th} trading day. This is to allow for the algorithm to initiate and leave enough time for majority of the experts to have sufficient data to begin making trading decisions. Having simulated the 5000 different Min-t statistics as in Section III A step 5 (a) using simulations of the profit process in Eq. (22), Figure 4 illustrates the histogram of Min-t values. The critical value t_c is then computed as the 0.95-quantile of

Strategy	Mean (mean rank)	St. Dev.	Min	Max
EMA X-over	0.8739 (673.6343)	0.1767	0.5216	1.4493
Ichimoku Kijun Sen	0.9508 (623.3194)	0.2313	0.5424	1.5427
MACD	0.9504 (657.7639)	0.1750	0.5601	1.6065
Moving Ave X-over	0.8895 (632.6944)	0.1930	0.5206	1.4505
ACC	1.0994 (736.5833)	0.3131	0.5283	1.9921
BOLL	1.0499 (569.1944)	0.3536	0.6076	1.7746
Fast Stochastic	0.9995 (778.6111)	0.3699	0.6006	1.8555
MARSI	1.0723 (639.3611)	0.2081	0.6947	1.6917
MOM	1.0403 (681.4444)	0.1353	0.7349	1.3595
Online Anti-Z-BCRP	0.7579 (731.9444)	0.1935	0.4649	1.0924
Online Z-Anticor	1.3155 (694.5278)	0.4388	0.6363	2.3886
Online Z-BCRP	1.2818 (652.8611)	0.2637	0.8561	1.8341
PROC	0.8963 (718.0833)	0.1631	0.6305	1.2161
RSI	1.1339 (757.3889)	0.2544	0.6440	1.7059
SAR	0.7314 (654.1111)	0.0619	0.6683	0.8683
Slow Stochastic	1.1135 (793.2222)	0.3302	0.6955	2.1023
Williams %R	0.9416 (728.6944)	0.3150	0.4662	1.5131

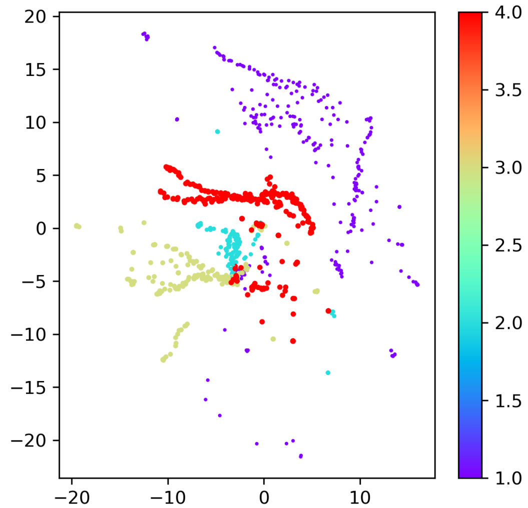
Table II: Group summary statistics of the overall rankings of experts grouped by their underlying strategy ($\boldsymbol{\omega}(i)$ where $i = 1, \ldots, 17$) for the daily trading. In brackets next to mean are the mean overall ranking of experts within the group of their underlying strategy.

Strategy	Mean	St. Dev.	Min	Max
EMA X-over	-0.00010	0.00633	-0.09745	0.08074
Ichimoku Kijun Sen	-0.00004	0.00723	-0.10467	0.06157
MACD	-0.00003	0.00725	-0.15993	0.08074
Moving Ave X-over	-0.00009	0.00644	-0.15993	0.11482
ACC	0.00007	0.00760	-0.15993	0.08028
BOLL	0.00002	0.00711	-0.06457	0.06480
Fast Stochastic	-0.00001	0.00847	-0.06469	0.06279
MARSI	0.00006	0.00612	-0.06788	0.06527
MOM	0.00004	0.00603	-0.06051	0.15820
Online Anti-Z-BCRP	-0.00022	0.00773	-0.09847	0.09336
Online Z-Anticor	0.00021	0.00759	-0.06475	0.09773
Online Z-BCRP	0.00021	0.00771	-0.09336	0.09847
PROC	-0.00007	0.00733	-0.10467	0.09745
RSI	0.00010	0.00666	-0.06460	0.09745
SAR	-0.00023	0.00724	-0.10467	0.08724
Slow Stochastic	0.00009	0.00809	-0.06480	0.06820
Williams %R	-0.00006	0.00815	-0.06820	0.06317

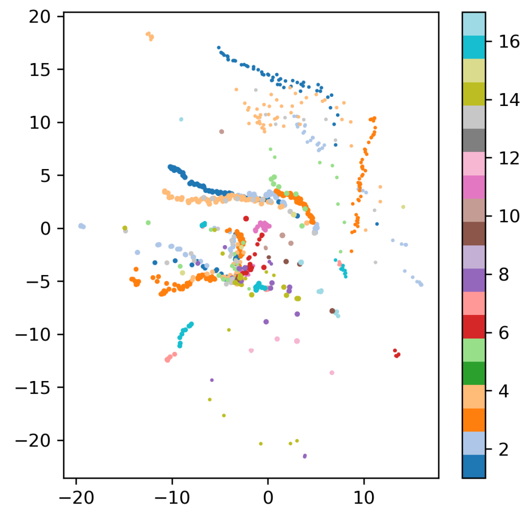
Table III: Group summary statistics of the expert's profits and losses per period grouped by their underlying strategy ($\boldsymbol{\omega}(i)$ where $i = 1, \ldots, 17$).

the simulated distribution which refers to a significance level of $\alpha = 5\%$ and is illustrated by the red vertical line. The resulting critical value is $t_c = 0.7263$. The Min-t resulting from the realised incremental profits and losses of the overall strategy is 3.0183 (vertical green line). By Eq. (27), we recover a p-value of zero. Thus, we can conclude that there is significant evidence to reject the null of no statistical arbitrage at the 5% significance level.

In addition to testing for statistical arbitrage, we also report the number of days it takes for the probability of loss of the strategy to decline below 5% using Eq. (27) adjusted for the case of the CM model. As discussed in 7, for each $n = 1, \ldots, T$, we perform MLE for $\Delta\nu_{1:n}$ to get the parameter estimates. We then substitute these estimates into Eq. (27) to get an estimate of the probability of loss for the n^{th} period. This is all done in terms of the CM model. The figure inset of Figure 4 illustrates the probability of loss for each of the first 25 trading days where we compute the probability of loss of the profit and



(a)



(b)

Figure 3: Figure 3(a) and show the latent space of Variational Autoencoder on the time series' of expert wealth's implemented using Keras in Python. In Figure 3(a) experts are coloured by which of the 4 object clusters they trade whereas in Figure 3(b), experts are coloured by their underlying trading strategy $\boldsymbol{\omega}(i)$.

loss process from the first trading period up to the n^{th} period for each $n = 1, \dots, 25$. As is evident from the figure inset, it takes roughly 10 periods for the probability of loss to converge below 5%.

A.2. Transaction Costs

In this section we reproduce the results from above but this time including transaction costs for daily trading as discussed in Section II E. Once direct and indirect (Eq. (14)) costs have been computed, the idea is to subtract off the transaction cost from S_t and PL_t on each day t and compound the resulting profit/loss onto the cumulative wealth and profits and losses respectively up until day t.

It is clear from the inset of Figure 5, which illustrates the profits and losses (**PL**) of the overall strategy less transaction costs for each period, that consistent losses are incurred when transaction costs are incorporated. Furthermore, there is no evidence to reject the no statistical arbitrage null hypothesis as the Min-t statistic resulting from the overall strategy is well below the critical value at the 95^{th} percentile of the histogram as illustrated in Figure 6. In addition to this, although the probability of loss of the strategy with transaction costs included initially converges to zero, soon after it jumps up to a probability of one and remains there for the rest of the periods. This is illustrated in the inset of Figure 6.

Considering the above evidence contained in Figure 5, Figure 6 and it's associated figure inset, the overall strategy does not survive historical back tests in terms of profitability when transaction costs are considered and may not be well suited for an investor utilising daily data whom has a limited time to make adequate profits.

B Intraday-Daily Data

Below we report the results of the algorithm implementation for a combination of intraday and daily JSE data. We run the algorithm on the the OHLCV data of 15 most liquid stocks from a set of 30 of the JSE Top 40[27]. Liquidity is calculated in terms of average daily trade volume for the first 4 days of the period 02-01-2018 to 09-03-2018. The set of 15 stocks is as follows: FSR:SJ ,GRT:SJ, SLM:SJ, BGA:SJ, SBK:SJ, WHL:SJ, CFR:SJ, MTN:SJ, DSY:SJ, IMP:SJ, APN:SJ, RMH:SJ, AGL:SJ, VOD:SJ and BIL:SJ. The remaining 40 days' data for the aforementioned period is utilised to run the learning algorithm on. As in the daily data implementation, we again analyse the two cases of trading with and without transaction costs which we report in the following two subsections below.

[27] See Appendix E E.2 for the list of the 30 stocks along with their Bloomberg ticker symbols

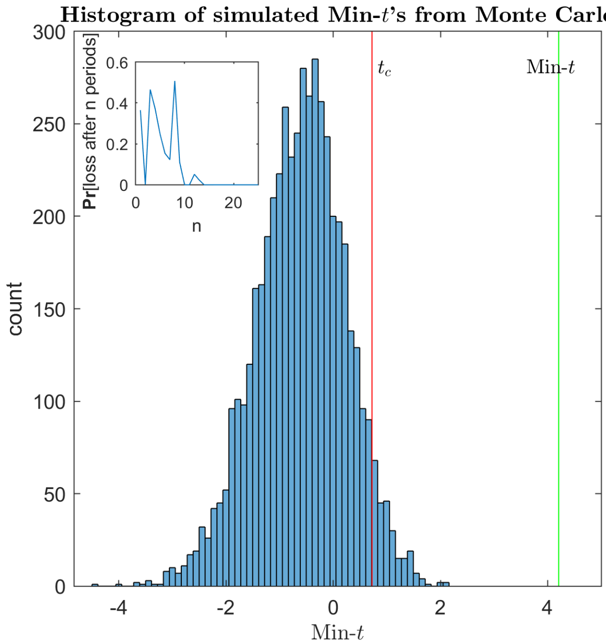


Figure 4: Histogram of the 5000 simulated Min-t statistics resulting from the CM test implemented on the simulated incremental process given in Eq. (22) along with the Min-t statistic (green) for the overall strategy's profit and loss sequence over the 400 day period stretching from the 30^{th} trading day until the 430^{th} trading day without any account for transactions costs. The figure inset displays the probability of loss for each of the first 30 trading days where we compute the probability of loss of the profit and loss process from the first trading period up to the n^{th} period for each $n = 1, \ldots, 25$.

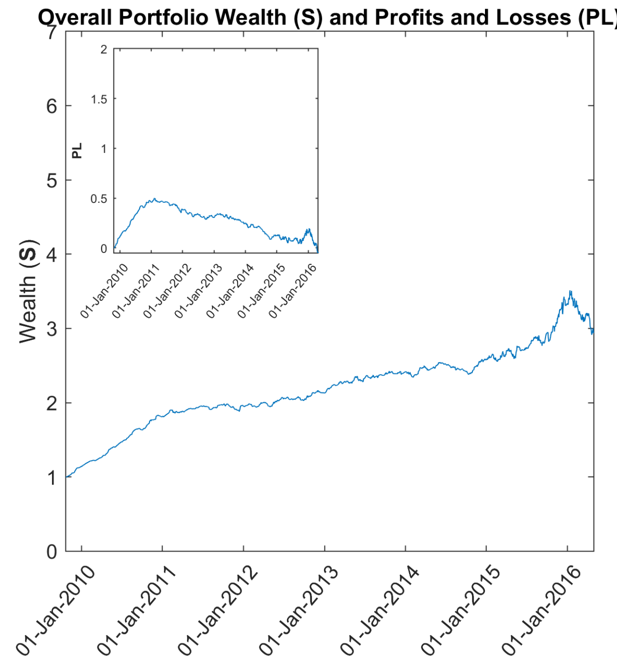


Figure 5: Overall cumulative portfolio wealth (S) for daily data with transaction costs. The figure inset illustrates the profits and losses (**PL**) for overall strategy for daily data with transaction costs.

B.1. No Transaction Costs

Without transaction costs, the cumulative wealth achieved by the overall strategy, illustrated in Figure 7 evolves similar to an exponential function over time. The associated profits and losses are displayed in the figure inset of Figure 7. Incremental profits and losses are obviously a lot smaller compared to the daily data cases resulting in a much smoother function in comparison to the daily data case (Figure 1).

Table IV is the intraday-daily analogue of Table II. In this case, the exponential moving crossover strategy (EMA X-over) produces the expert with the greatest wealth and acceleration (ACC) the expert with the least terminal wealth. Exponential moving crossover also produces experts with the highest variation in terminal wealth's. Price rate of change (PROC) is by far the strategy with the best mean ranking experts among all experts however Z-BCRP produces experts with highest mean terminal wealth.

Again, as for the daily data case, we implement a test for statistical arbitrage for intraday-daily trading without transaction costs for 400 trading periods starting from the 6^{th} time bar of the 2nd trading day[28] using the intraday-daily profit and loss sequence (**PL**). Figure 8 illustrates the histogram of simulated Min-t values with the 0.95-percentile of the simulated distribution representing the critical value t_c (red) and the Min-t (green) resulting from the incremental profits and losses of the overall strategy resulting from the learning algorithm. The resulting critical value is 0.7234 and the Min-t value is 4.2052. Thus, there is strong evidence to reject the null hypothesis of no statistical arbitrage as the resulting p-value is identical to zero.

The figure inset of Figure 8 illustrates the probability of loss for each of the first 25 periods of the 400 periods as discussed in the above paragraph. It takes 13 periods for the probability of loss to converge to zero which is highly desirable.

B.2. Transaction Costs

We now report the results of the algorithm run on the same intraday-daily data as in the subsection above but

[28] This corresponds to the trading period within which the very first trading decisions are made.

Strategy	Mean (mean rank)	St. Dev.	Min	Max
EMA X-over	1.0024 (662.7639)	0.0094	0.9801	1.0375
Ichimoku Kijun Sen	0.9989 (710.3750)	0.0085	0.9663	1.0303
MACD	0.9995 (684.8704)	0.0067	0.9720	1.0202
Moving Ave X-over	1.0012 (708.7824)	0.0058	0.9766	1.0204
ACC	0.9953 (831.3333)	0.0079	0.9646	1.0048
BOLL	0.9974 (712.9722)	0.0069	0.9787	1.0089
Fast Stochastic	0.9991 (711.4167)	0.0040	0.9871	1.0085
MARSI	0.9973 (736.2500)	0.0062	0.9824	1.0094
MOM	0.9982 (723.1389)	0.0087	0.9700	1.0082
Online Anti-Z-BCRP	0.9980 (597.3056)	0.0062	0.9828	1.0103
Online Z-Anticor	1.0015 (655.7778)	0.0058	0.9896	1.0180
Online Z-BCRP	1.0031 (566.5833)	0.0069	0.9898	1.0149
PROC	0.9980 (445.1389)	0.0064	0.9814	1.0140
RSI	0.9997 (535.5833)	0.0065	0.9861	1.0171
SAR	0.9945 (499.7222)	0.0053	0.9790	1.0005
Slow Stochastic	1.0007 (508.5278)	0.0048	0.9927	1.0173
Williams %R	1.0020 (536)	0.0034	0.9957	1.0133

Table IV: Group summary statistics of the overall rankings of experts grouped by their underlying strategy ($\boldsymbol{\omega}(i)$ where $i = 1, \ldots, 17$) for intraday-daily trading. In brackets are the mean overall ranking of experts utilising each strategy.

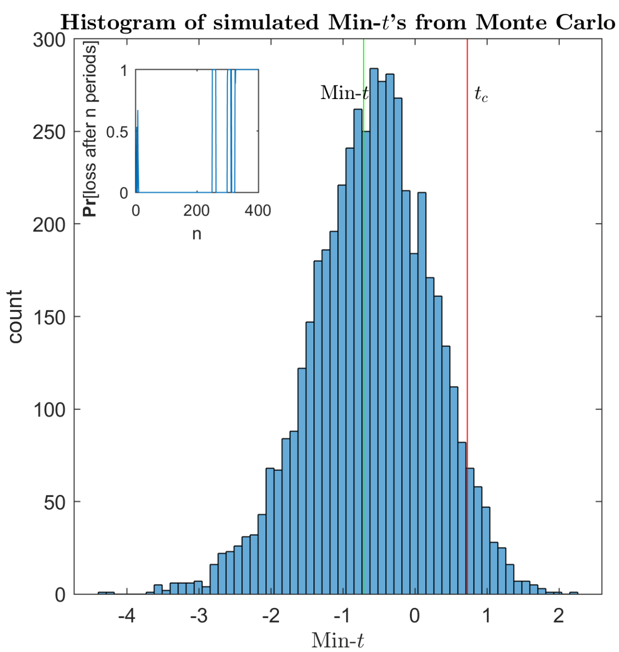


Figure 6: Histogram of the 5000 simulated Min-t statistics resulting from the CM model and the incremental process given in Eq. (22) along with the Min-t statistic (green) for the overall strategy's profit and loss sequence over the 400 day period stretching from the 30^{th} trading day until the 430^{th} trading day with transactions costs incorporated. Also illustrated is the critical value at the 5% significance level (red). The figure inset shows the probability of the overall trading strategy generating a loss for each of the first 400 trading days.

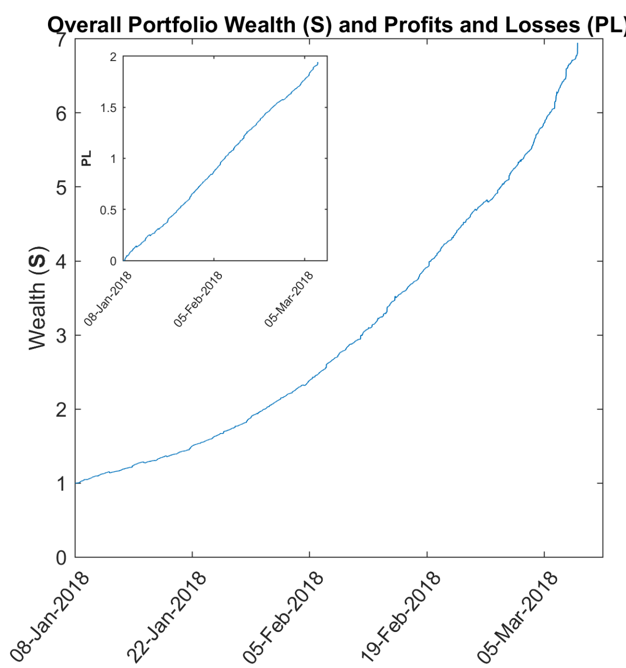


Figure 7: The overall cumulative portfolio wealth (**S**) for intraday-daily data with no transaction costs. The figure inset illustrates the associated profits and losses.

this time with transaction costs incorporated as outlined in Section II E. Figure 9 and the figure inset illustrate the overall cumulative portfolio wealth (**S**) and profits and losses (**PL**) respectively for intraday-daily trading with transaction costs. For comparative reasons, the axes are set to be equivalent to those in the case of no transaction

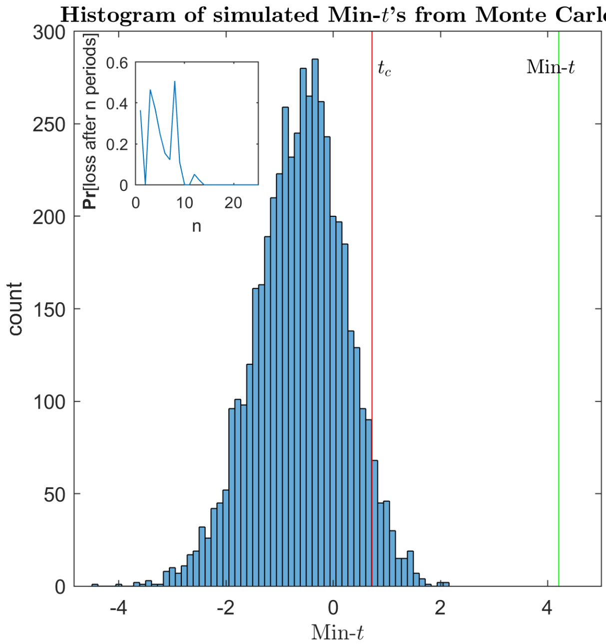


Figure 8: Histogram of the 5000 simulated Min-t statistics resulting from the CM model and the incremental process given in Eq. (22) for the first 400 trading periods for intraday-daily profits and losses without taking into account transaction costs along with the Min-t statistic for the overall strategy (green) and the critical value at the 5% significance level (red). The figure inset shows the probability of the overall trading strategy generating a loss after n periods for each $n = 5, \ldots, 25$ of the intraday-daily profit and loss process (**PL**) taken from the 5^{th} time bar of the second day when active trading commences.

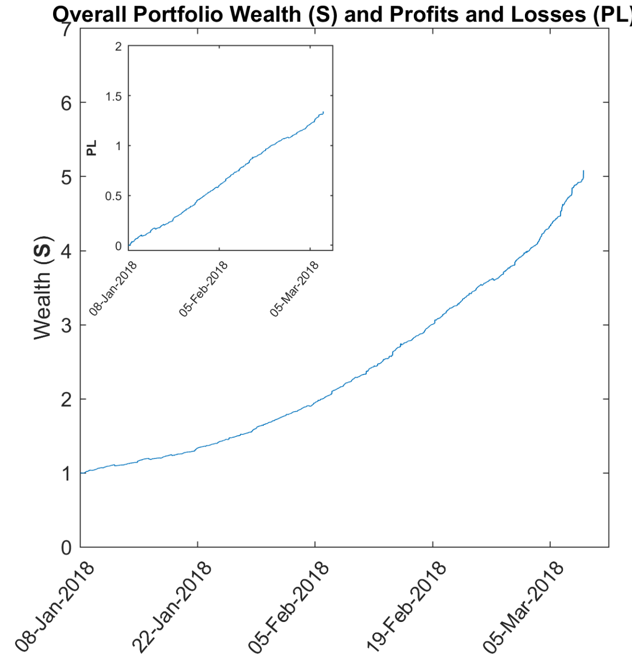


Figure 9: The overall cumulative portfolio wealth (**S**) for intraday-daily data with transaction costs. The figure inset illustrates the associated profits and losses.

costs (Figure 9 and the figure inset). Surprisingly, even with a total daily trading cost (direct and indirect) of roughly 130bps, which is a fairly conservative approach, the algorithm is able to make satisfactory returns which is in contrast to the daily trading case (Figure 5). Furthermore, Figure 10 provides significant evidence that the no statistical arbitrage null hypothesis can be rejected and has an almost identical Min-t statistic (4.32 in the transaction costs case compared to 4.21) to that of the case of no transaction costs (Figure 8). What is even more comforting is the fact that even when transaction costs are considered, the probability of loss per trading period converges to zero albeit slightly slower (26 trading periods) than the case of no transaction costs (13 trading periods as illustrated in the inset of figure Figure 8).

The above results for intraday-daily trading are in complete contrast to the case of daily trading with transaction costs whereby the no statistical arbitrage null could not be rejected, the probability of loss did not converge to zero and remain there, and trading profits steadily declined over the trading horizon. This suggests that the proposed algorithm may be much better suited to trading at higher frequencies. This is not surprising and is in complete agreement with Schulmeister [36] who argues that the profitability of technical trading strategies had declined over from 1960 before becoming unprofitable from the 1990's. A substantial set of technical trading strategies are then implemented on 30-minute data and the evidence suggests that such strategies returned adequate profits between 1983 and 2007 however the profits declined slightly between 2000 and 2007 compared to the 1980's and 1990's. This suggests that markets may have become more efficient and even the possibility that stock price and volume trends have shifted to even higher frequencies than 30 minutes [36]. This supports the choice to trade the algorithm proposed in this paper on at least 5-minute OHLCV data and reinforces our conclusion that ultimately, the most desirable implementation of the algorithm would be in volume-time which is best suited for high frequency trading.

VI CONCLUSION

We have developed a learning algorithm built from a base of technical trading strategies for the purpose of trading equities on the JSE that is able to provide favourable returns when ignoring transaction costs, under both daily and intraday trading conditions. The returns are reduced when transaction costs are considered in the daily setting, however there is sufficient evidence to

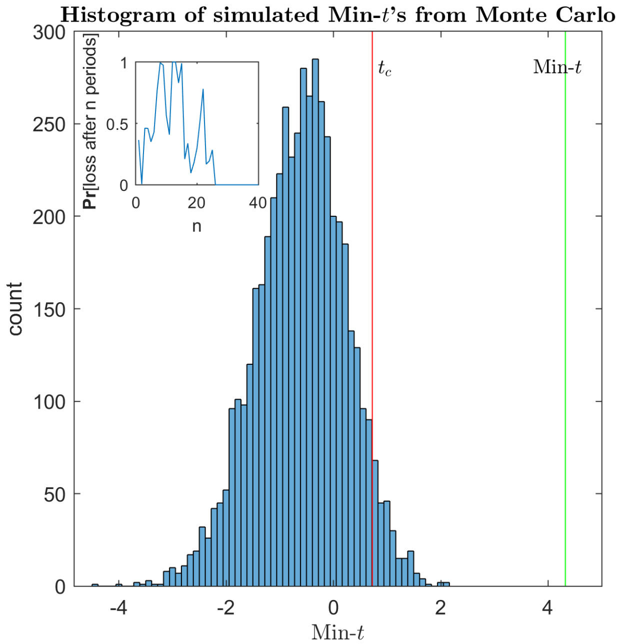


Figure 10: Histogram of the 5000 simulated Min-t statistics resulting from the CM model and the incremental process given in Eq. (22) for the first 400 trading periods for intraday-daily profit and losses less transaction costs along with the Min-t statistic for the overall strategy (green) and the critical value at the 5% significance level (red).

suggest that the proposed algorithm is really well suited to intraday trading.

This is reinforced by the fact that there exists meaningful evidence to reject a carefully defined null hypothesis of no statistical arbitrage in the overall trading strategy even when a reasonably conservative view is taken on intraday trading costs. We are also able to show that it in both the daily and intraday-daily data implementations that the probability of loss declines below 5% relatively quickly which strongly suggests that the algorithm is well suited for a trader whose preference or requirement is to make adequate returns in the short-run.

The superior performance of the algorithm for intraday trading is in agreement with Schulmeister [36] who concluded that daily profitability of a large set of technical trading strategies has steadily declined since 1960 and has been unprofitable since the onset of the 1990's. This is in contrast to trading 30-minute data between 1983 and 2007 which has produced decent average gross returns but this has slowly declined since the early 2000's. In conclusion, the proposed algorithm is much better suited to trading at higher frequencies.

We are however cognisant of the fact that intraday trading will also typically required a large component of accumulated trading profits to finance frictions, concretely to fund direct, indirect and business model costs [6]. For this reason we are careful to remain sceptical with this class of algorithms long-run performance when trading with real money in a live trading environment. The current design of the algorithm is not yet ready to be traded on live market data, however with some effort it is easily transferable to such use cases given the sequential nature of the algorithm and its inherent ability to receive and adapt to new incoming data while making appropriate trading decisions based on the new data. Concretely, the algorithm should be deployed in the context of volume-time trading rather than the calendar time context considered in this work.

Possible future work includes implementing the algorithm in volume-time which will be best suited for dealing with a high frequency implementation of the proposed algorithm given the intermittent nature of order-flow. We also propose replacing the learning algorithm with an online (adaptive) neural network that has the ability to predict optimal holding times of stocks. Another interesting line of work that has been considered is to model the population of trading experts as competing in a predator-prey environment [37, 38]. This was a an initial key motivation for the research project, to find which collections of technical trading strategies can be grouped collectively and how these would interact with each other. This includes utilising cluster analysis to group together or separate trading experts based on their similarities and dissimilarities and hence make appropriate inferences regarding their interactions and behaviours at the level of collective and emergent dynamics.

VII ACKNOWLEDGEMENTS

NM and TG would like to thank the Statistical Finance Research Group in Department of Statistical Sciences for various useful discussion relating to the work. In particular we would like to thank Etienne Pienaar, Lionel Yelibi and Duncan Saffy. We thank Michael Gant for his help with developing some of the strategies and for numerous valuable discussions with regards to the statistical arbitrage test. We would also like thank Guenter Schmidt for various insightful comments and discussion relating to the testing and validation of online learning algorithms and optimal conversion problems. TG would like to thank UCT FRC for funding (UCT fund 459282).

VIII APPENDIX

A Variable Definitions

Variable	Definition

$\boldsymbol{\omega}$	parameter of underlying strategy of a given expert which corresponds to one of the technical trading strategies or trend-following strategies. $\boldsymbol{\omega}(i)$ is then the i^{th} strategy from the set of all strategies
W_1	number of trading rules which utilise one parameter
W_2	number of trading rules utilising two parameters
$\boldsymbol{n_1}$	vector of short-term look-back parameters
$\boldsymbol{n_2}$	vector of long-term look-back parameters
L	number of short-term look-back parameters
K	number of short-term look-back parameters
$\mathbf{c}$	object clusters parameter where $\mathbf{c}(i)$ is the i^{th} object cluster
C	number of object clusters
Ω	total number of experts
ADV	average daily volume
δ_{liq}	number of days to look-back to choose the m most liquid stocks
m	number of stocks to be passed into the expert generating algorithm ranked by their liquidity
$\mathbf{h}_t^n$	n^{th} expert's strategy (m+1 portfolio controls) for period t
$\mathbf{H}_t$	expert control matrix by made up of all n experts' strategies at time t for all m stocks i.e. $\mathbf{H}_t = [\mathbf{h}_t^1, \dots, \mathbf{h}_t^n]$
$\mathbf{Sh}_t$	wealth of all n experts at time t
Sh_t^n	n^{th} expert's wealth at time t
$\mathbf{b}_t$	final overall aggregate portfolio to be used in the following period $t+1$ which we denote
$\mathbf{S}$	vector of the overall aggregate portfolio compounded wealth over time
S_t	overall aggregate portfolio compounded wealth at time t
$\mathbf{PL}$	vector of the overall aggregate portfolio cumulative profits and losses
PL_t	the overall aggregate portfolio cumulative profits and losses at time t
nb (ns)	number of buy (sell) signals from the set of output signals
$\boldsymbol{\sigma}_+$ ($\boldsymbol{\sigma}_-$)	vector of standard deviations of stocks to be bought (sold)
$\mathbf{w}$	$m+1$ vector of weights allocated to m stocks and the risk-free asset
w_{rf}	weight allocated to the risk-free asset
t_{min}	time period at which trading commences (either daily/intraday)
T	terminal time of trading (generally daily but can be either daily/intraday)
$\mathbf{X}_t$	$m\times5$ matrix of stock's OHLCV values at each time period t (either daily/intraday)
$\mathbf{X}_d$	daily OHLCV data
$\mathbf{X}_I$	intraday OHLCV data
$\mathbf{x}_t$	vector of $m+1$ price relatives at time period t
$\mathbf{P}_t^c$	vector of closing prices for all m stocks at time period t
$P_{m,t}^c$	closing price of stock m at time period t
t_I	t_I^{th} time bar of a given day
T_I	final (terminal) time bar of a given day (4:30pm)
$\mathbf{H}_{t,T_I}^F$	fused intraday-daily expert control matrix
$\mathbf{h}_{n,t,T_I}^F$	n^{th} expert's controls for all m assets at time bar t_I on the t^{th} day
$\mathbf{Sh}_{t,t_I}^F$	fused intraday-daily expert wealth vector for all n experts for the t_I^{th} time bar on the t^{th} day
Sh_{n,t,T_I+1}^F	n^{th} expert's wealth at time bar t_I on the t^{th} day
σ	volatility of the returns of a stock
$\nu(t)$	cumulative discounted trading profits/losses at time t
$\Delta\nu_t$	cumulative discounted trading profit/loss increment at time t i.e. $\Delta\nu_t = \nu(t) - \nu(t) - 1$
Min-t	statistic used for accepting/rejecting the no statistical arbitrage null hypothesis

Table V: Variable definitions

B Online Learning Algorithm

Algorithm 1 Expert Generating Algorithm

Require:

1. OHLCV prices up to current time $\boldsymbol{X}_t$
2. short-term parameters $\boldsymbol{n_1}$
3. long-term parameters $\boldsymbol{n_2}$
4. set of strategies to be considered $\boldsymbol{\omega}$
5. set of cluster indices to be considered $\mathbf{c}$
6. current portfolio controls $\boldsymbol{b}_t$
7. past agent-controls $\boldsymbol{H}_{t-1}$
8. current agent-controls $\boldsymbol{H}_t$

`Expert_index` $= 0$

for $t = t_{min}$ to T **do**
  for $c = 1$ to C **do**
    for $w = 1$ to W **do**
      Define w^{th} strategy as string and convert to function

      for $\ell = 1$ to L **do**
        $\ell_1 = \boldsymbol{n_1}(\ell)$
        for $k = 1$ to K **do**
          if w^{th} strategy only has 1 param **then**
            break
          end if
          $k_1 = \boldsymbol{n_1}(k)$
          if $k_1 > \ell_1$ **then**
            `Expert_index` $=$ `Expert_index`$+\ 1$
            Call *controls* function to compute weights for w^{th} strategy $\rightarrow \mathbf{h}_{n,t}^{c,\ell,k} = \mathbf{w}$
          else
            continue
          end if

end for
if Strategy has 1 parameter **then**
Call *controls* function to compute weights for w^{th} strategy $\rightarrow \mathbf{h}^{c,\ell}_{n,t} = \mathbf{w}$
end if
end for
end for
end for
end for

Algorithm 2 Compute *output signals*

Require:
1. past expert controls $\boldsymbol{h}^n_{t-1}$
2. current period signals $\mathbf{s}$

initialise combined signals: `output_s` $=$ zeros(size($\mathbf{s}$))
if all previous controls ($\boldsymbol{h}^n_{t-1}$) were NaN's or zeros **then**
`output_s` $= \mathbf{s}$
else
for $i = 1$ to length($\mathbf{s}$) **do**
if s(i) == 0 **then**
`output_s`(i) = sign($\boldsymbol{h}^n_{t-1}$(i))
else
`output_s`(i) = s(i)
end if
end for
end if

Algorithm 3 Transform signals to controls: volatility loading

Require:
1. OHLCV prices up to current time $\mathbf{X}_t$
2. **output_s**

extract closing prices from $\mathbf{X}_t$: $\mathbf{P}^c$

if all **output_s** equal zero **then**
$\mathbf{w}$ =zeros(length(**output_s**)+1)

else if output_s ≥ 0 **then**
compute standard deviations of closing prices over last 120 days for stocks where elements of **output_s** > 0:

$\mathbf{vol}^+ = \text{std}(\mathbf{P}^c(\textbf{output\_s} > 0, \text{end-119})$
$\mathbf{w} = [0.5 \cdot \textbf{output\_s}; -0.5]$
$\mathbf{w}(\textbf{output\_s} > 0) = (1/\text{sum}(\mathbf{vol}^+)) * \mathbf{w}(\textbf{output\_s} > 0) \cdot \mathbf{vol}^+$
else if output_s ≤ 0 **then**
compute standard deviations of closing prices over last 120 days for stocks where elements of **output_s** < 0:

$\mathbf{vol}^- = \text{std}(\mathbf{P}^c(\textbf{output\_s} < 0, \text{end-119})$
$\mathbf{w} = [0.5 \cdot \textbf{output\_s}; 0.5]$
$\mathbf{w}(\textbf{output\_s} < 0) = (1/\text{sum}(\mathbf{vol}^-)) * \mathbf{w}(\textbf{output\_s} < 0) \cdot \mathbf{vol}^-$
else
compute standard deviations of closing prices over last 120 days for stocks where elements of **output_s** > 0 and where **output_s** < 0:

$\mathbf{vol}^+ = \text{std}(\mathbf{P}^c(\textbf{output\_s} > 0, \text{end-119})$
$\mathbf{vol}^- = \text{std}(\mathbf{P}^c(\textbf{output\_s} < 0, \text{end-119})$
$\mathbf{w} = [\textbf{output\_s}; 0]$
$\mathbf{w}(\textbf{output\_s} > 0) = 0.5 \cdot \left(\frac{1}{\text{sum}(\text{abs}(\mathbf{vol}^+))}\right) \cdot \mathbf{vol}^+ \cdot \mathbf{w}(\textit{output\_s} > 0)$
$\mathbf{w}(\textbf{output\_s} < 0) = 0.5 \cdot \left(\frac{1}{\text{sum}(\text{abs}(\mathbf{vol}^-))}\right) \cdot \mathbf{vol}^- \cdot \mathbf{w}(\textit{output\_s} < 0)$
$\mathbf{w}(\text{end}) = \text{sum}(\mathbf{w})$
end if
return w

$$\begin{tikzcd}
(\mathbf{x}_t, \mathbf{H}_t) \arrow[r, "q_n"] \arrow[d, "r_m = x_m - 1"'] \arrow[dr, "r_m = x_m - 1"] & b_m = \sum_n q_n \mathbf{h}^n_t \arrow[d, "b_m"] \\
S_n = \sum_m \mathbf{h}^n_t r_m + 1 \arrow[r, "q_n"] & \begin{array}{l} S = \sum_n q_n S_n \\ S = \sum_m b_m r_m + 1 \end{array}
\end{tikzcd}$$

Figure 11: Relationship between the components of the Online Learning Algorithm

---

Algorithm 4 Online Learning Algorithm

Require:
1. updated agent-controls $\mathbf{H}_{t+1}$
2. current price relatives $\mathbf{x}_t$
3. current portfolio controls $\mathbf{b}_t$
4. current agent-controls $\mathbf{H}_t$
5. past agent-wealth $\mathbf{Sh}_{t-1}$
6. past portfolio wealth S_{t-1}

for $t = t_{min}$ to T **do**
Update portfolio wealth:
$S_t = S_{t-1}(\mathbf{b}_t(\boldsymbol{x}^T_t - 1) + 1)$

Update expert wealth's:
$Sh^n_t = Sh^n_{t-1}(\mathbf{h}^n_t(\boldsymbol{x}^T_t - 1) + 1)$

Update expert mixtures:
$q_{n,t+1} = Sh^n_t$

Renormalise the expert mixtures:

$$q_{n,t+1} = \begin{cases} \sum_n q_{n,t+1} = 1, & q_{n,t+1} \geq 0 \\ \sum_n |q_{n,t+1}| = 1, & \sum_n q_{n,t+1} = 0 \end{cases}$$

  Update the portfolio:
  $\mathbf{b}_{t+1} = \sum_n q_{n,t+1}\mathbf{h}^n_{t+1}$

  Leverage corrections:
  if $(\nu = \sum_m |b_{m,t}|) \neq 1$ **then**
    Renormalise controls:
    $b_{n,t+1} = \frac{1}{\nu} b_{n,t+1}$
    Renormalise mixtures:
    $q_{n,t+1} = \frac{1}{\nu} q_{n,t+1}$
  end if
end for
return $(\mathbf{b}_{t+1}, Sh^n_t, S_t, q_{n,t+1})$

Algorithm 5 Intraday-Daily Algorithm

Require:
1. OHLCV daily prices $\mathbf{X}_d$
2. OHLCV intraday time bars $\mathbf{X}_I$
3. vector indicating index for start of each day `uniqueday`

initialise daily price relatives: $\mathbf{ret}_d = \text{repeat}(1, m+1)$

for $t = 2$ to T **do**

  initialise intraday price relatives for t-th day:
  $\mathbf{ret}_I = \text{repeat}(1, m+1)$

  initialise expert wealth's:
  $\mathbf{Sh}^F$(`uniqueday`(t)+t-1) = repeat(1)

  repeat daily STEFI for all time bars:
  STEFI_I = repeat(STEFI(t),`uniqueday`(t+1) - `uniqueday`(t))

  for t_I = `uniqueday`(t)+1 to `uniqueday`($t+1$)-1 **do**

    get closing prices for time bars $t_I - 1$ and t_I:
    $\mathbf{P}^c_I$

    compute price relatives for current time bar and append to previous period price relatives:
    $\mathbf{ret}_I = [\mathbf{ret}_I; \mathbf{P}^c_I(t)/\mathbf{P}^c_I(t-1)]$

    run expert generating algorithm:
    `expert_gen`($t_0, t, \mathbf{ret}_I$)

    run online learning algorithm:
    `online_learn`($t_0, t, \mathbf{ret}_I$)

  end for

  get closing prices for day $t-1$ and day t: $\mathbf{P}^c_{t-1}$ and $\mathbf{P}^c_t$

  compute price relatives for day t:
  $\mathbf{ret}_d = [\mathbf{ret}_d; \mathbf{P}^c_d(t)/\mathbf{P}^c_d(t-1)]$

  run expert generating algorithm:
  `expert_gen`(t, `uniqueday`($t+1$))

  run online learning algorithm:
  `online_learn`(t, `uniqueday`($t+1$))

end for

for $c = 1$ to C **do**
  for $w = 1$ to W **do**
    Define w^{th} strategy as string from file name and convert to function

    for $\ell = 1$ to L **do**
      $\ell_1 = \boldsymbol{n_1}(\ell)$
      for $k = 1$ to K **do**
        if w^{th} only has 1 parameter **then**
          break
        end if
        $k_1 = \boldsymbol{n_1}(k)$
        if $k_1 > \ell_1$ **then**
          `Expert_index` = `Expert_index` + 1
          Call *controls* function to compute weights for w^{th} strategy $\rightarrow \mathbf{h}^{n,c,\ell,k}_t = \mathbf{w}$
        else
          continue
        end if
      end for
      if Strategy has 1 parameter **then**
        Call *controls* function to compute weights for w^{th} strategy $\rightarrow \mathbf{h}^{n,c,\ell}_t = \mathbf{w}$
      end if
    end for
  end for
end for

C Technical Indicators and Trading Rules

We follow [9, 14] in introducing and describing some of the more popular technical analysis indicators as well as a few others that are widely available. We also provide some trading rules which use technical indicators to generate buy, sell and hold signals.

C.1 Simple Moving Average

The most common moving average is the *simple moving average* (SMA). The SMA is the mean of a time series (typically of closing prices) over the last n trading days and is usually updated every trading period to take into account more recent data and drop older values. The smaller the value of n, the closer the moving average will fit to the price data

1.a SMA Indicator: $\mathbf{SMA}_t^c(n)$

$$\mathrm{SMA}_t(\mathbf{P}^c, n) = \frac{1}{n} \sum_{i=0}^{n-1} P_{t-i}^c \tag{28}$$

C.2 Exponential Moving Average

The exponential moving average (EMA) makes use of today's close price, yesterdays moving average value and a smoothing factor (α). The smoothing factor will determine how quickly the exponential moving average responds to current market prices [14]. A simple moving average is used for the initial EMA value.

2.a EMA Indicator: $\mathbf{EMA}_t^c(n)$

$$\mathrm{EMA}(\mathbf{P}^c, n) = \alpha P_t^c + (1-\alpha)\mathrm{EMA}(P_{t-1}^c, n) \tag{29}$$

where $\alpha = \frac{2}{n+1}$

C.3 Highest High

Highest high is the greatest high price in the last n periods.

$$\mathrm{HH}(n) = \max(\boldsymbol{P_n^h}) \tag{30}$$

where the vector with high prices of last n periods is given by $\boldsymbol{P_n^h} = (P_{t-n}^h, P_{t-n+1}^h, P_{t-n+2}^h, \ldots, P_t^h)$

C.4 Lowest Low

Lowest low is the smallest low price in the last n periods.

$$\mathrm{LL}(n) = \min(\boldsymbol{P_n^l}) \tag{31}$$

where the vector with low prices of last n periods is given by $\boldsymbol{P_n^l} = (P_{t-n}^l, P_{t-n+1}^l, P_{t-n+2}^l, \ldots, P_t^l)$

C.5 Moving Average Crossover Trading Rule

The moving average crossover rule uses 2 SMA's, a short SMA and a longer SMA. A buy signal occurs when the faster (shorter) moving average crosses above the slower (longer) moving average, and a sell signal occurs when the shorter moving average crosses below the longer moving average.

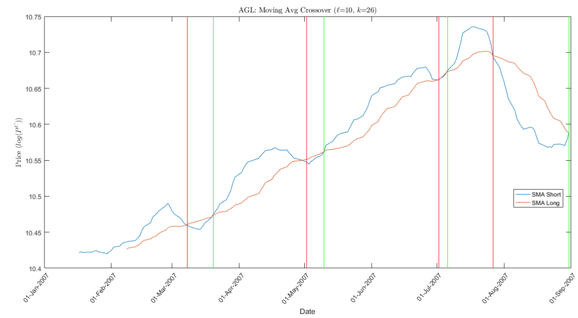


Figure 12: Moving average crossover trading rule implemented on 8 months of Anglo American PLC daily closing prices from 01-01-2007 to 30-08-2007. Green and red vertical lines represent buy and sell signals respectively.

C.6 Exponential Moving Average Crossover Rule

The calculation is identical to the Moving Average Crossover rule above however instead of using a SMA, an EMA is used.

C.7 Ichimoku Kinko Hyo

The Ichimoku Kinko Hyo (at a glance equilibrium chart) system consists of five lines and the Kumo (cloud) [39–41]. The five lines all work in concert to produce the end result. The size of the Kumo is an indication of the current market volatility, where a wider Kumo is a more volatile market.

7.a Ichimoku Kinko Hyo Indicators

1. **Tenkan-sen (Conversion Line):** $(\mathrm{HH}(n_1) + \mathrm{LL}(n_1))/2$
2. **Kijun-sen (Base Line):** $(\mathrm{HH}(n_2) + \mathrm{LL}(n_2))/2$
3. **Chikou Span (Lagging Span):** Close plotted n_2 days in the past
4. **Senkou Span A (Leading Span A):** (Conversion Line + Base Line)/2
5. **Senkou Span B (Leading Span B):** $(\mathrm{HH}(n_3) + \mathrm{LL}(n_3))/2$

6. **Komu (Cloud):** Area between the Leading Span A and the Leading Span B form the Cloud

Ichimoku uses three key time periods for it's input parameters: Typically, $n_1 = 7$, $n_2 = 22$, and $n_3 = 44$. We will keep the n_1 parameter fixed at 7 but vary the other 2 parameters.

7.b Ichimoku Kinko Hyo Strategies

The Kijun Sen Cross strategy is one of the most powerful and reliable trading strategies within the Ichimoku system due to the fact that it can be used on nearly all time frames with exceptional results [41].

Decision	Condition
Buy	Kijun Sen crosses the closing price curve from the bottom up
Sell	Kijun Sen crosses the closing price curve from the top down
Hold	otherwise

Table VI: Ichimoku Kijun Sen Cross strategy

The buy and sell (bullish and bearish) signals are classified into three major classifications: strong, neutral and weak. The strength is determined by where the crossover occurs in relation to the cloud.

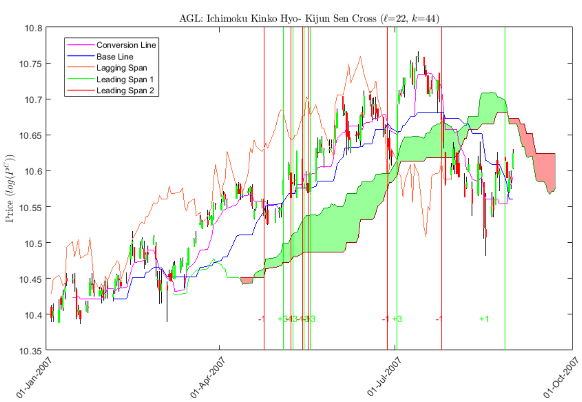


Figure 13: Ichimoku Kijun Sen Cross strategy trading rule implemented on 9 months of Anglo American PLC closing prices from 01-01-2007 to 30-09-2007. The log of the closing price is used for illustrative purposes. Green and red candle sticks refer to bullish and bearish trading days respectively. When the Leading span 1 is greater than the Leading span 2, the outlook is bullish (green cloud) and vice versa for a bearish outlook (red cloud). The number next to each of the vertical green and red lines indicates the strength of the buy and sell signals respectively.

C.8 Momentum

8.a Momentum Indicator: $\mathrm{MOM}_t(n)$

Momentum gives the change in the closing price over the past n periods

$$\mathrm{MOM}_t(n) = P_t^c - P_{t-n}^c \tag{32}$$

C.9 Momentum Trading Rule

Decision	Condition
Buy	$\mathrm{MOM}_{t-1}(n) \leq \mathrm{EMA}_t(\mathrm{MOM}_t(n), \lambda)$ & $MOM_t(n) > EMA_t(MOM_t(n), \lambda)$
Sell	$MOM_{t-1}(n) \geq \mathrm{EMA}_t(\mathrm{MOM}_t(n), \lambda)$ & $\mathrm{MOM}_t(n) < \mathrm{EMA}_t(\mathrm{MOM}_t(n), \lambda)$
Hold	otherwise

Table VII: Momentum trading rule

C.10 Acceleration

10.a Acceleration Indicator: $\mathrm{ACC}_t(n)$

Acceleration measures the change in momentum between two consecutive periods t and $t-1$

$$\mathrm{ACC}_t(n) = \mathrm{MOM}_t(n) - \mathrm{MOM}_{t-1}(n) \tag{33}$$

10.b Acceleration Trading Rule

Decision	Condition
Buy	$\mathrm{ACCEL}_{t-1}(n) + 1 \leq 0$ & $\mathrm{ACCEL}_t(n) + 1 > 0$
Sell	$\mathrm{ACCEL}_{t-1}(n) + 1 \geq 0$ & $\mathrm{ACCEL}_t(n) + 1 < 0$
Hold	otherwise

Table VIII: Acceleration trading rule

C.11 Moving Average Convergence/Divergence Oscillator

The Moving Average Convergence/Divergence (MACD) oscillator is a momentum indicator developed by Gerald Appel and attempts to determine whether traders are accumulating stocks or distributing stocks. It is calculated by computing the difference between a short-term and a long-term moving average. The idea is then to compute the signal line, which is accomplished by taking an exponential moving average of the MACD determines instances at which to buy (oversold) and sell (oversold) when used in conjunction with the MACD [11].

11.a MACD Indicators: $\mathrm{MACD}_t(n_1, n_2)$

The MACD indicator is computed using the following steps:

1. $\mathrm{LongEMA}_t = \mathrm{EMA}_t(P^c, n_2)$
2. $\mathrm{ShortEMA}_t = \mathrm{EMA}_t(P^c, n_1)$
3. $\mathrm{MACD}_t(n_1, n_2) = \mathrm{ShortEMA}_t - \mathrm{LongEMA}_t$
4. $\mathrm{SignalLine}_t(n_1, n_2, n_3) = \mathrm{EMA}_t(\mathrm{MACD}_t(n_2, n_1), n_3)$
5. $\mathrm{MACDS}_t(n_1, n_2, n_3) = \mathrm{MACD}_t(n_1, n_2) - \mathrm{SignalLine}_t(n_1, n_2, n_3)$

11.b MACD Trading Rule

Decision	Condition
Buy	$\mathrm{MACD}_{t-1}(n_2, n_1) \leq \mathrm{MACDS}_t(n_2, n_1, n_3)$ & $\mathrm{MACD}_t(n_2, n_1) > \mathrm{MACDS}_t(n_2, n_1 n_3)$
Sell	$\mathrm{MACD}_{t-1}(n_2, n_1) \geq \mathrm{MACDS}_t(n_2, n_1, n_3)$ & $\mathrm{MACD}_t(n_2, n_1) < \mathrm{MACDS}_t(n_2, n_1, n_3)$
Hold	otherwise

Table IX: MACD trading rule

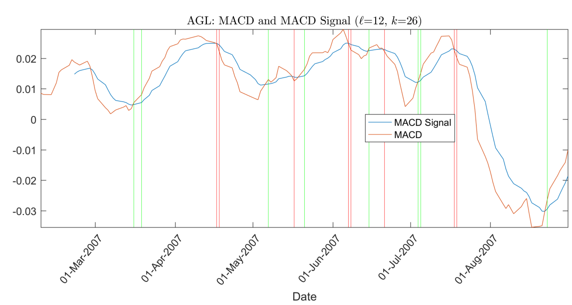


Figure 14: MACD trading rule implemented on 9 months of Anglo American PLC closing prices from 01-01-2007 to 30-08-2007. Green and red vertical lines represent buy and sell signals respectively.

C.12 Fast Stochastics

12.a Fast Stochastic Indicators

Fast%K$_t$(**n**) :

$$\mathrm{Fast\%K}_t(n) = \frac{P_t^c - \mathrm{LL}(n)}{\mathrm{HH}(n) - \mathrm{LL}(n)} \tag{34}$$

Fast%D$_t$(***n***) :

$$\mathrm{Fast\%D}_t(n) = \mathrm{SMA}_t(\mathrm{Fast\%K}_t(n), 3) \tag{35}$$

12.b Fast Stochastic Trading Rule

Decision	Condition
Buy	$\mathrm{Fast\%K}_{t-1}(n) \leq \mathrm{Fast\%D}_t(n)$ & $\mathrm{Fast\%K}_t(n) > \mathrm{Fast\%D}_t(n)$
Sell	$\mathrm{Fast\%K}_{t-1}(n) \geq \mathrm{Fast\%D}_t(n)$ & $\mathrm{Fast\%K}_t(n) < \mathrm{Fast\%D}_t(n)$
Hold	otherwise

Table X: Fast stochastic trading rule

C.13 Slow Stochastics

13.a Slow Stochastic Indicators

Slow%K$_t(n)$:

$$\mathrm{Slow\%K}_t(n) = \mathrm{SMA}_t(\mathrm{Fast\%K}_t(n), 3) \tag{36}$$

Slow%D$_t(n)$:

$$\mathrm{Slow\%D}_t(n) = \mathrm{SMA}_t(\mathrm{Slow\%K}_t(n), 3) \tag{37}$$

13.b Slow Stochastic Trading Rule

Decision	Condition
Buy	$\mathrm{Slow\%K}_{t-1}(n) \leq \mathrm{Slow\%D}_t(3)$ & $\mathrm{Slow\%K}_t(n) > \mathrm{Slow\%D}_t(3)$
Sell	$\mathrm{Slow\%K}_{t-1}(n) \geq \mathrm{Slow\%D}_t(3)$ & $\mathrm{Slow\%K}_t(n) < \mathrm{Slow\%D}_t(3)$
Hold	otherwise

Table XI: Slow stochastic trading rule

C.14 Relative Strength Index

Relative Strength Index (RSI) compares the days that stock prices finish up (closing price higher than the previous day) against those periods that stock prices finish down (closing price lower than the previous day) [9].

14.a RSI Indicator: $\mathrm{RSI}_t(n)$

$$\mathrm{RSI}_t(n) = 100 - \frac{100}{1 + \frac{\mathrm{SMA}_t(\boldsymbol{P}_n^{\mathrm{up}}, n_1)}{\mathrm{SMA}_t(\boldsymbol{P}_n^{\mathrm{dwn}}, n_1)}} \tag{38}$$

where [9]

$$P_t^{\mathrm{up}} = \begin{cases} P_t^c & \text{if } P_{t-1}^c < P_t^c \\ \mathrm{NaN} & \text{otherwise} \end{cases} \tag{39}$$

$$P_t^{\mathrm{dwn}} = \begin{cases} P_t^c & \text{if } P_{t-1}^c > P_t^c \\ \mathrm{NaN} & \text{otherwise} \end{cases} \tag{40}$$

and

$$\boldsymbol{P}_n^{\mathrm{up}} = (P_{t-n}^{\mathrm{up}}, P_{t-n+1}^{\mathrm{up}}, \ldots, P_t^{\mathrm{up}}) \tag{41}$$
$$\boldsymbol{P}_n^{\mathrm{dwn}} = (P_{t-n}^{\mathrm{dwn}}, P_{t-n+1}^{\mathrm{dwn}}, \ldots, P_t^{\mathrm{dwn}}) \tag{42}$$

14.b RSI Trading Rule

Decision	Condition
Buy	$\mathrm{RSI}_{t-1}(n) \leq 30$ & $\mathrm{RSI}_t(n) > 30$
Sell	$\mathrm{RSI}_{t-1}(n) \geq 70$ & $\mathrm{RSI}_t(n) < 70$
Hold	otherwise

Table XII: Relative strength index (RSI) trading rule

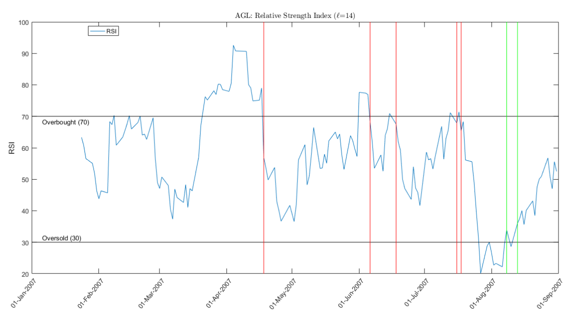


Figure 15: RSI trading rule implemented on 9 months of Anglo American PLC closing prices from 01-01-2007 to 30-08-2007. Green and red vertical lines represent buy and sell signals respectively.

C.15 Moving Average Relative Strength Index

Moving Average Relative Strength Index (MARSI) is an indicator that smooths out the action of RSI indicator [42].

15.a MARSI Indicator: $\mathrm{MARSI}_t(n_1, n_2)$

MARSI is calculated by simply taking an n_2-period SMA of the RSI indicator.

$$\mathrm{MARSI}_t(n_1, n_2) = \mathrm{SMA}(\mathrm{RSI}_t(n_1), n_2) \tag{43}$$

15.b MARSI Trading Rule

Rather buying or selling when RSI crosses upper and lower thresholds (30 and 70) as in the RSI trading rule above, buy and sell signals are generated when the SMA of the MARSI crosses above or below the thresholds [42].

Decision	Condition
Buy	$\mathrm{MARSI}_{t-1}(n) \leq 30$ & $\mathrm{MARSI}_t(n) > 30$
Sell	$\mathrm{MARSI}_{t-1}(n) \geq 70$ & $\mathrm{MARSI}_t(n) < 70$
Hold	otherwise

Table XIII: MARSI trading rule

C.16 Bollinger Band

Bollinger bands uses a SMA ($\mathrm{Boll}_t^m(n)$) as it's reference point (known as the median band) with regards to the upper and lower Bollinger bands ($\mathrm{Boll}_t^u(n)$ and $\mathrm{Boll}_t^d(n)$ respectively) which are calculated as functions of standard deviations (s). When the closing price crosses above (below) the upper (lower) Bollinger band, it is a sign that the market is overbought (oversold).

16.a Bollinger Band Indicator: $\mathrm{Boll}_t^m(n)$

$$\begin{aligned} &\mathrm{Boll}_t^m(n) = \mathrm{SMA}_t^c(n) \\ &\text{Upper Bollinger band: } \mathrm{Boll}_t^m(n) + s\sigma_t^2(n) \\ &\text{Lower Bollinger band: } \mathrm{Boll}_t^m(n) - s\sigma_t^2(n) \end{aligned} \tag{44}$$

where s is chosen to be 2.

16.b Bollinger Trading Rule

Decision	Condition
Buy	$P_{t-1}^c \geq \mathrm{Boll}_t^d(n)$ & $P_t^c \geq \mathrm{Boll}_t^u(n)$
Sell	$P_{t-1}^c \leq \mathrm{Boll}_t^d(n)$ & $P_t^c > \mathrm{Boll}_t^u(n)$
Hold	otherwise

Table XIV: Bollinger trading rule

C.17 Price Rate-Of-Change

17.a PROC Indicator: $\mathbf{PROC}_t(n)$

The rate of change of the time series of closing prices P_t^c over the last n periods expressed as a percentage

$$\mathrm{PROC}_t(n) = 100 \cdot \frac{P_t^c - P_{t-n}^c}{P_{t-n}^c} \tag{45}$$

17.b PROC Trading Rule

Decision	Condition
Buy	$\mathrm{PROC}_{t-1}(n) \leq 0$ & $\mathrm{PROC}_t(n) > 0$
Sell	$\mathrm{PROC}_{t-1}(n) \geq 0$ & $\mathrm{PROC}_t(n) < 0$
Hold	otherwise

Table XV: Price rate of change (PROC) trading rule

C.18 Williams %R

18.a Williams %R: $\mathbf{Will}_t(n)$

Williams Percent Range (Williams %R) is calculated similarly to the fast stochastic oscillator and shows the level of the close relative to the highest high in the last n periods

$$\mathrm{Will}_t(n) = \frac{\mathrm{HH}(n) - P_t^c}{\mathrm{HH}(n) - \mathrm{LL}(n)} \cdot (-100) \tag{46}$$

18.b Williams %R Trading Rule

Decision	Condition
Buy	$\mathrm{Will}_{t-1}(n) \geq -20$ & $\mathrm{Will}_t(n) < -80$
Sell	$\mathrm{Will}_{t-1}(n) \leq -20$ & $\mathrm{Will}_t(n) > -80$
Hold	otherwise

Table XVI: Williams %R trading rule

C.19 Parabolic SAR

Parabolic Stop and Reverse (SAR), developed by J. Wells Wilder, is a trend indicator formed by a parabolic line made up of dots at each time step [43]. The dots are formed using the most recent Extreme Price and an acceleration factor (AF), 0.02, which increases each time a new Extreme Price (EP) is reached. The AF has a maximum value of 0.2 to prevent it from getting too large. Extreme Price represents the highest (lowest) value reached by the price in the current up-trend (down-trend). The acceleration factor determines where in relation to the price the parabolic line will appear by increasing by the value of the AF each time a new EP is observed and thus affects the rate of change of the Parabolic SAR.

19.a SAR Indicator: SAR(n)

The steps involved in calculating the SAR indicator are as follows:

1. initialise variables: trend is initially set to 1 (up-trend), EP to zero, AF AF_0 to 0.02, SAR_0 to the closing price at time zero, lastHigh to high price at time zero (P_0^h) and lastLow to the low price at time zero (P_0^l))
2. update parameters: EP, lastHigh, lastLow and AF based on where the current high is in relation to the lastHigh (up-trend) or where the current low is in relation to the lastLow (down-trend)
3. compute the next period SAR value: update time $t+1$ SAR value, SAR_{t+1}, using Eq. (47)
4. modify the SAR value and the parameters for a change in trend: modify the SAR_{t+1} value, AF, EP lastLow, lastHigh and the trend based on the trend and it's value in relation to the current low P_t^l and current high P_0^h
5. go to next time period and return to step 2

Below is the formula for the Parabolic SAR for time $t+1$ calculated using the previous value at time t:

$$\mathrm{SAR}_{t+1} = \mathrm{SAR}_t + \alpha(\mathrm{EP} - \mathrm{SAR}_t) \tag{47}$$

19.b SAR Trading Rule

Decision	Condition
Buy	$\mathrm{SAR}_{t-1} \geq P_{t-1}^c$ & $\mathrm{SAR}_t < P_t^c$
Sell	$\mathrm{SAR}_{t-1} \leq P_{t-1}^c$ & $\mathrm{SAR}_t > P_t^c$
Hold	otherwise

Table XVII: SAR trading rule

D Trend Following and Contrarian Mean Reversion Strategies

The zero-cost BCRP (trend following), zero-cost anti-BCRP and zero-cost anti-correlation (both contrarian mean reverting) algorithms are explained in more detail in the following subsections.

D.1 Zero-Cost BCRP

Zero-cost BCRP is the zero-cost long/short version of the BCRP strategy and is a trend following algorithm in that long positions are taken in stocks during upward trends while short positions are taken during downward trends. The idea is to first find the portfolio controls that maximize the expected utility of wealth using all in-sample price relatives according to a given constant level of risk aversion. The resulting portfolio equation is what is known to be the Mutual Fund Separation Theorem [44]. The second set of portfolio controls in the mutual fund separation theorem (active controls) is what we will use as the set of controls for the zero-cost BCRP strategy and are given by:

$$\mathbf{b} = \frac{1}{\gamma}\Sigma^{-1}\left[\mathbb{E}[\mathbf{R}] - \mathbf{1}\frac{\mathbf{1}^\intercal\Sigma^{-1}\,\mathbb{E}[\mathbf{R}]}{\mathbf{1}^\intercal\Sigma^{-1}\mathbf{1}}\right] \tag{48}$$

where Σ^{-1} is the inverse of the covariance matrix of returns for all m stocks, $\mathbb{E}[\mathbf{R}]$ is vector of expected returns of the stocks, $\mathbf{1}$ is a vector of ones of length m and γ is the risk aversion parameter. Eq. (48) is the risky Optimal Tactical Portfolio (OTP) that takes optimal risky bets given the risk aversion γ. The risk aversion is selected during each period t such that the controls are unit leverage and hence $\sum_{i=1}^{m}|b_i| = 1$.

The covariance matrix and expected returns for each period t are computed using the set of price relatives from today back ℓ days (short-term look-back parameter).

D.2 Zero-Cost Anti-BCRP

Zero-cost anti-BCRP is exactly the same as zero-cost BCRP except that we reverse the sign of the expected returns vector such that $\mathbb{E}[\mathbf{R}] = -\,\mathbb{E}[\mathbf{R}]$.

D.3 Zero-Cost Anti-Correlation

Zero-cost anti-correlation (Z-Anticor) is an adapted version of the Anticor algorithm developed in [45]. The first step is to extract the price relatives for the two most recent sequential windows each of length ℓ. Let $\boldsymbol{\mu}_2^\ell$ and $\boldsymbol{\mu}_1^\ell$ denote the average log-returns of the ℓ price relatives in the most recent window ($\boldsymbol{x}_t^{t-\ell+1}$) and the price relatives in the window prior to that ($\boldsymbol{x}_{t-\ell}^{t-2\ell+1}$) respectively. Also, let the lagged covariance matrix and lagged correlation matrix be defined as follows:

$$\Sigma^\ell = \frac{1}{\ell-1}[(\boldsymbol{x}_{t-\ell}^{t-2\ell+1} - 1) - \mathbf{1}^\intercal\boldsymbol{\mu}_1^\ell]^\intercal[(\boldsymbol{x}_t^{t-\ell+1} - 1) - \mathbf{1}^\intercal\boldsymbol{\mu}_2^\ell] \tag{49}$$

$$\mathrm{P}_{ij}^\ell = \frac{\Sigma_{ij}^\ell}{\sqrt{\Sigma_{ij}^\ell\Sigma_{ij}^\ell}} \tag{50}$$

Z-Anticor then computes the claim that each pair of stocks have on one another, denoted $\mathrm{claim}_{i\to j}^\ell$, which is the claim of stock j on stock i. This is the extent to which we want to shift our allocation from stock i to stock j [45]. $\mathrm{claim}_{i\to j}^\ell$ exists and is thus non-zero if and only if $\boldsymbol{\mu}_2 > \boldsymbol{\mu}_1$ and $\mathrm{P}_{ij} > 0$. The claim is then calculated as

$$\mathrm{claim}_{i\to j}^\ell = \mathrm{P}_{ij}^\ell + \max(-\mathrm{P}_{ii}^\ell, 0) + \max(-\mathrm{P}_{jj}^\ell, 0) \tag{51}$$

The adaptation we propose for long/short portfolios for the amount of transfer that take places from stock i to stock j is given by:

$$\mathrm{transfer}_{i\to j}^\ell = \frac{1}{3}\mathrm{claim}_{i\to j}^\ell \tag{52}$$

Finally, we calculate the expert control for the i^{th} stock in period $t+1$ as follows:

$$\mathbf{h}_{t+1}^n(i) = \mathbf{h}_t^n(i) + \sum_i[\mathrm{transfer}_{j\to i}^\ell - \mathrm{transfer}_{i\to j}^\ell] \tag{53}$$

Each control is then normalised in order to ensure unit leverage on the set of controls.

E Data Related Appendices

E.1 JSE TOP 40 Sector Constituents: Daily Data

The daily data is sourced from Thomson Reuters. The companies and their associated Reuters Instrument Code (RIC) of the three major sectors in the JSE Top 40 are [52]:

Resources: JSE-RESI (J210)
Anglo American Platinum Ltd (AMSJ.J), Anglo American PLC (AGLJ.J), AngloGold Ashanti Ltd (ANGJ.J), BHP Billiton PLC (BILJ.J), Mondi Ltd (MNDJ.J), Mondi PLC (MNPJ.J), Sasol Ltd (SOLJ.J).

Industrials: JSE-INDI (J211)
Aspen Pharmacare Holdings Ltd (APNJ.J), Bidvest Group Ltd (BVTJ.J), British American Tobacco PLC (BTIJ.J), Compagnie Financiere Richemont SA (CFRJ.J), Capital & Counties Properties PLC (CCOJ.J), Growthpoint Properties Ltd (GRTJ.J), Intu Properties PLC (ITUJ.J), Mediclinic International Ltd (MEIJ.J), MTN Group Ltd (MTNJ.J), Naspers Ltd (NPNJ.J), Remgro Ltd (REMJ.J), Redifine Properties Ltd (RDFJ.J), SABMiller PLC (SABJ.J), Shoprite Holdings Ltd (SHPJ.J), Steinhoff International Holdings (SNHJ.J), Tiger Brands Ltd (TBSJ.J), Vodacom Group

Ltd (VODJ.J), Woolworths Holdings Ltd (WHLJ.J), Mr Price Group Ltd (MRPJ.J), Netcare Ltd (NTCJ.J).

Financials: JSE-FINI (J212)
Discovery Holdings Ltd (DSYJ.J), Firstrand Ltd (FSRJ.J), Investec Ltd (INLJ.J), Investec PLC (INPJ.J), Nedbank Group Ltd (NEDJ.J), Old Mutual PLC (OMLJ.J), RMB Holdings Ltd (RMHJ.J), Rand Merchant Investment Holdings Ltd (RMIJ.J), Sanlam Ltd (SLMJ.J), Standard Bank Group Ltd (SBKJ.J), Brait SE (BATJ.J), Barclays Africa Group Ltd (BGAJ.J), Capitec Bank Holdings Ltd (CPIJ.J), Fortress REIT Ltd (B) (FFBJ.J), Fortress REIT Ltd (A) (FFAJ.J), Reinet Investments SCA (REIJ.J).

E.2 JSE TOP 40 Sector Constituents: Intraday Data

Below are 30 of the JSE Top 40 stocks as of the 30 June 2018. All the tick and daily data from 01-01-2018 to 30-06-2018 is sourced from Bloomberg.

Resources: JSE-RESI (J210)
Anglo American PLC (AGL:SJ), AngloGold Ashanti Ltd (ANG:SJ), African Rainbow Minerals Ltd (ARI:SJ), BHP Billiton PLC (BIL:SJ), Exxaro Resources Ltd (EXX:SJ), Impala Platinum Holdings Ltd (IMP:SJ), Mondi Ltd (MND:SJ), Sasol Ltd (SOL:SJ).

Industrials: JSE-INDI (J211)
Aspen Pharmacare Holdings Ltd (APN:SJ), Bidvest Group Ltd (BVT:SJ), British American Tobacco PLC (BTI:SJ), Compagnie Financiere Richemont SA (CFR:SJ), Capital & Counties Properties PLC (CCO:SJ), Growthpoint Properties Ltd (GRT:SJ), Intu Properties PLC (ITU:SJ), MTN Group Ltd (MTN:SJ), Naspers Ltd (NPN:SJ), Tiger Brands Ltd (TBSJ.J), Shoprite Holdings Ltd (SHP:SJ), Vodacom Group Ltd (VOD:SJ), Woolworths Holdings Ltd (WHL:SJ).

Financials: JSE-FINI (J212)
Discovery Holdings Ltd (DSY:SJ), Firstrand Ltd (FSR:SJ), Investec Ltd (INL:SJ), Nedbank Group Ltd (NED:SJ), RMB Holdings Ltd (RMH:SJ), Sanlam Ltd (SLM:SJ), Standard Bank Group Ltd (SBK:SJ), Barclays Africa Group Ltd (BGA:SJ).

F Back-testing Work Flow

Figure 16 illustrates the flow diagram for the MATLAB learning class designed for the online learning algorithm.

[1] L. Györfi, G. Ottucsák & H. Walk (2012). Machine Learning for Financial Engineering. Imperial College Press.
[2] T. M. Cover (1991). Universal Portfolios, Math. Finance., 1,1,1-29.
[3] P.H. Algoet & T.M. Cover (1988). Asymptotic optimality and asymptotic equipartition properties of logoptimum investment. The Annals of Probability, 16(2), pp.876-898. https://doi.org/10.1214/aop/1176991793.
[4] T. M. Cover & E. Ordentlich (1996). Universal Portfolios with Side Information, IEEE Transactions on Information Theory, 42(2), 348-363.
[5] L. Györfi, F. Udina & H. Walk (2008). Nonparametric nearest neighbor based empirical portfolio selection strategies, Statistics & Risk Modeling, De Gruyter, 26(2), 145-15.
[6] F. Loonat & T. Gebbie (2018). Learning zero-cost portfolio selection with pattern matching. PLoS ONE 13(9): e0202788. https://doi.org/10.1371/journal.pone.0202788.
[7] J. F. Clayburg (2002). Four Steps to Trading Success: Using Everyday Indicators to Achieve Extraordinary Profits. John Wiley & Sons. ISBN: 978-0-471-21799-2
[8] E. Chan (2009). Quantitative Trading: How to Build Your Own Algorithmic Trading Business.John Wiley & Sons Inc, Wiley Trading. ISBN: 9780470466261.
[9] G. Creamer & Y. Freund (2010). Automated trading with boosting and expert weighting. Quantitative Finance, 10(4), 401-420. http://dx.doi.org/10.1080/14697680903104113
[10] T.S. Chande & S. Kroll (1994). The new technical trader: boost your profit by plugging into the latest indicators. John Wiley & Sons, Wiley Finance. ISBN: 9780471597803.
[11] M. D. Rechenthin (2014). Machine-learning classification techniques for the analysis and prediction of high-frequency stock direction. PhD thesis, University of Iowa. http://ir.uiowa.edu/etd/4732.
[12] A. Lo, H. Mamaysky, & J. Wang (2000). Foundations of Technical Analysis: Computational Algorithms, Statistical Inference, and Empirical Implementation. The Journal of Finance, 4.
[13] D. R. Aronson (2007). Evidence-Based Technical Analysis: Applying the Scientific Method and Statistical Inference to Trading Signals. John Wiley & Sons, Inc.
[14] L. N. Kestner (2003). Quantitative Trading Strategies: Harnessing the Power of Quantitative Techniques to Create a Winning Trading Program. McGraw-Hill.
[15] A. W. Lo & J. Hasanhodzic (2009). The Heretics of Finance: Conversations with Leading Practitioners of Technical Analysis. Bloomberg Press, (800) 869-1231.
[16] C-H. Park & S. Irwin (2004). The Profitability of Technical Analysis: A Review. SSRN Electronic Journal.
[17] K. Hou, C. Xue & L . Zhang (2017). Replicating Anomalies. NBER Working Paper Series, No. 23394. http://www.nber.org/papers/w23394

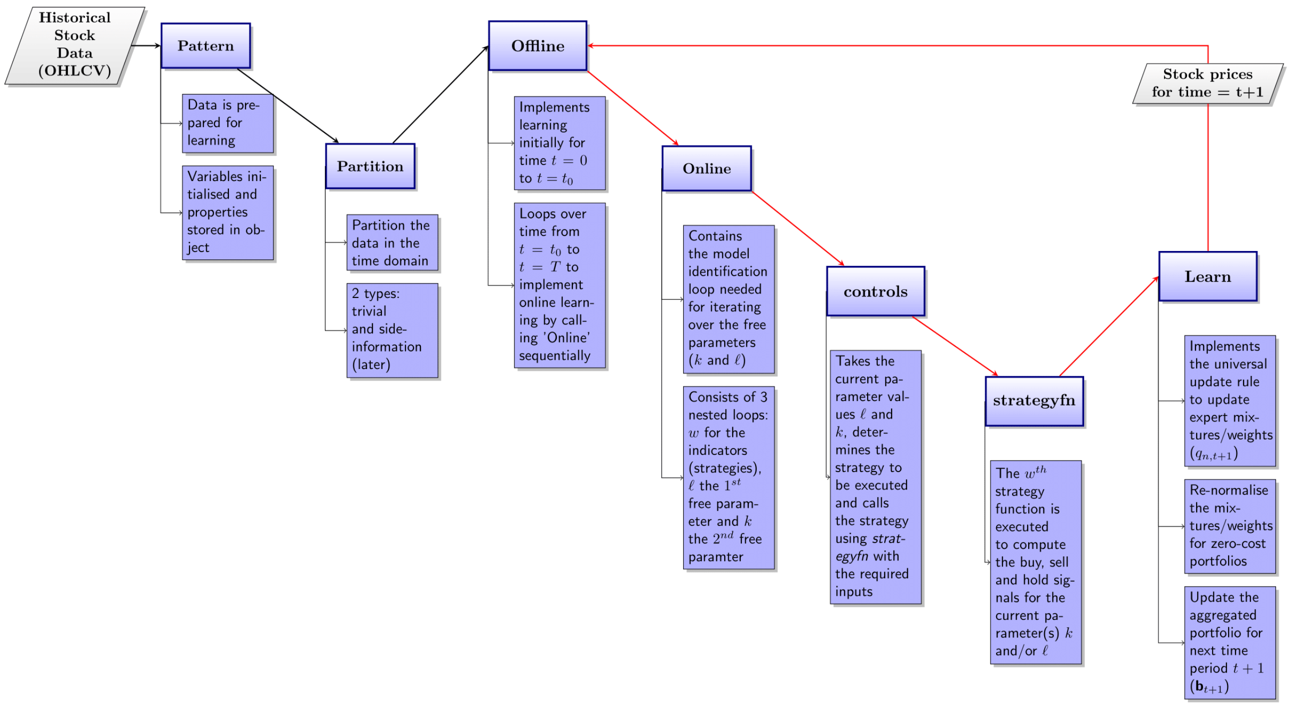


Figure 16: State Flow diagram for the MATLAB learning class.

[18] D. H. Bailey, J. M. Borwein, M. M. López de Prado, & Q. J. Zhu (2014). Pseudo-Mathematics and Financial Charlatanism: The Effects of Backtest Overfitting on Out-of-Sample Performance. Notices of the AMS, 61(5), 458-471.

[19] D. H. Bailey, J. M. Borwein, M. L. de Prado & Q. J. Zhu (2016). The Probability of Backtest Overfitting. Journal of Computational Finance 20, N0. 4, 39-69. DOI:10.21314/JCF.2016.322.

[20] S. Hogan, R. Jarrow, M. Teo & M. Warachka (2004). Testing market efficiency using statistical arbitrage with applications to momentum and value strategies. Journal of Financial Economics 73, 525 - 565.

[21] F. Black (1986). Noise. Journal of Finance, 41,3, 529-543.

[22] J-P. Bouchaud, S. Ciliberti, Y. Lemperiere, A. Majewski, P. Seager, K. S. Ronia (2017). Black Was Right: Price Is Within a Factor 2 of Value. Available at SSRN: https://ssrn.com/abstract=3070850.

[23] R. Jarrow, M. Teob, Y-K Tse & M. Warachka (2012). An improved test for statistical arbitrage. Journal of Financial Markets 15, 47-80.

[24] B. Li & S. C. Hoi (2016). Online Portfolio Selection:Principles and Algorithms. Taylor & Francis Group.

[25] K. Akcoglu, P. Drineas & M-Y. Kao (2004). Fast Universalization of Investment Strategies. SIAM Journal on Computing 34(1), 1-22.

[26] http://pages.stern.nyu.edu/\simadamodar/New_Home_Page/invemgmt/trading.htm

[27] J. Gatheral (2010). No-dynamic arbitrage and market Impact. Quantitative Finance, 10(7), 759.

[28] G. Huberman & W. Stanzl (2005). Optimal Liquidity Trading. Springer: Review of Finance, 9, 165-200. DOI:10.1007/s10679-005-7591-5.

[29] J-P. Bouchaud, Y. Gefen, M. Potters & M. Wyart (2004). Fluctuations and Response in Financial Markets: The Subtle Nature of 'Random' Price Changes. Available at SSRN: https://ssrn.com/abstract=507322

[30] T. Andersen, T. Bollerslev, F. X. Diebold, & P. Labys (2001). The Distribution of Realized Exchange Rate Volatility. Journal of the American Statistical Association, 96, 42-55.

[31] S-H. Poon (2008). Volatility Estimation. https://www.cmegroup.com/trading/fx/files/volEst.pdf

[32] Karatzas, I. & Shreve S.E. (1991). Brownian Motion and Stochastic Calculus. Graduate Texts in Mathematics, Springer New York, ISBN: 9780387976556.

[33] J-J. D. de Wit (2013). Statistical Arbitrage in South Africa. Minor dissertation submitted to the faculty of Commerce in partial fulfilment of the requirements for the Masters degree in Mathematical Finance at the University of Cape Town.

[34] etfSA STeFI-linked Money Market Deposit (5 April 2011). Managed by Nedbank Beta Solutions.

[35] Y-L, K, Samo & D. Hendricks (2018). What Makes An Asset Useful?. arXiv:1806.08444, https://arxiv.org/abs/1806.08444

[36] S. Schulmeister (2009). Profitability of technical stock trading: Has it moved from daily to intraday data?. Review of Financial Economics, vol. 18, no. 4, pp. 190-201.

[37] J. D. Farmer (2000). A Simple Model for the Nonequilibrium Dynamics and Evolution of a Finacnial Market. International Journal of Theoretical and Applied Finance, Vol. 03, No. 03, pp. 425-441.

[38] N. Johnson, G. Zhao, E. Hunsader, H. Qi, J. Meng & Brian Tivnan (2013). Abrupt rise of new machine ecology beyond human response time. Sceintific Reports 3(2627). DOI: 10.1038/srep02627.

[39] http://www.kumotrader.com/ichimoku_wiki/index.php?title=Main_Page

[40] D. Linton (2010). Cloud Charts: Trading Success with the Ichimoku Technique. ISBN: 9780956517104.

[41] Ichimoku 101. IchimokuTrade.com. https://www.ichimokutrade.com/articles/Ichimoku_Ebook.pdf

[42] M. B. Fayek, H. M. El-Boghdadi, S. M. Omran (2013). Multi-objective Optimization of Technical Stock Market Indicators using GAs. International Journal of Computer Applications (0975 - 8887) Volume 68 - No.20.

[43] J.W. Wilder (1978). New Concepts in Technical Trading Systems. Trend Research. ISBN: 9780894590276.

[44] W. Lee (2000). Theory and methodology of tactical asset allocation. John Wiley & Sons, Volume 65.

[45] A. Bordin, R. El-Yaniv & V. Gogan (2004). Can we learn to beat the best stock. Journal of Artificial Intelligence Research, 21 , 579-594.

[46] S. D. Campbell (2005). Review of Backtesting and Backtesting Procedures. Finance and Economics Discussion Series, Board of Governors of the Federal Reserve System, No 2005-21.

[47] F. Loonat & T. Gebbie (2016). MATLAB Pattern Matching and Learning Class ("pattern").

[48] T. Gebbie (2010). MATLAB Technical Analysis toolbox ("technical").

[49] R. Dochow (2016). Online Algorithms for the Portfolio Selection Problem. Springer Gabler.

[50] E. F. Fama & M. E. Blume (1996). Filter rules and stock market trading. Journal of Business, 39, 226-241.

[51] J. L. Kelly (1956). A new interpretation of information rate. Bell. Syst. Tech.,35, 917-926.

[52] M. Harvey, D. Hendricks, T. Gebbie, D. Wilcox, (2017), Deviations in expected price impact for small transaction volumes under fee restructuring, Physica A: Statistical Mechanics and it's Applications, Volume 471, 416-426, https://doi.org/10.1016/j.physa.2016.11.042.

[53] E. F. Fama (1970). Efficient capital markets: A review of theory and empirical work. Proceedings of the Twenty-Eighth Annual Meeting of the American Finance Association. Journal of Finance, 25(2), 383-417.

[54] A. Lo (2002). The statistics of Sharpe Ratio's. Financial Analyst Journal, 58(4), 36-52.

[55] D. H. Bailey & M. M. López de Prado (2014). The Deflated Sharpe Ratio: correcting for selection bias, backtest overfitting and non-normality. Journal of Portfolio Management, 40 (5) 94-107.