*Quantitative Finance*, 2021
Vol. 21, No. 8, 1325–1349, https://doi.org/10.1080/14697688.2020.1869292

# Learning the dynamics of technical trading strategies

N. J. MURPHY * and T. J. GEBBIE

Department of Statistical Sciences, University of Cape Town, Rondebosch, Cape Town, South Africa

(*Received 13 December 2019; accepted 4 December 2020; published online 24 March 2021*)

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 dynamics of a large population 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, 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. In addition, the algorithms generalisation error is analysed by recovering a probability of back-test overfitting estimate using a nonparametric procedure. The work aims to explore and better understand the interplay between different technical trading strategies from a data-informed perspective.

*Keywords*: Online learning; Technical trading; Portfolio selection; Statistical arbitrage; Back-test overfitting; Johannesburg Stock Exchange

*JEL Classifications*: G11, G14, O55

## 1. Introduction

Maximising wealth concurrently over multiple time periods is a difficult task; particularly when combined with capital allocations that are made simultaneously with the selection of plausible candidate trading strategies and signals. An approach to combining strategy selection with wealth maximisation is to use online or sequential machine learning algorithms (Györfi *et al.* 2012). Online portfolio selection algorithms attempt to automate a sequence of trading decisions among a set of stocks with the goal of maximising returns in the long run (Li and Hoi 2016a, Dochow 2016). Here, the long run can correspond to months or even years, and is dependent on the frequency at which trading takes place. 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 prices, 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 (Györfi *et al.* 2012). Cover (1991) introduced a 'follow-the-winner' online investment algorithm† called the Universal Portfolio (UP) algorithm.‡ The basic idea of the UP algorithm is to allocate capital to a set of experts characterised by different portfolios or trading strategies, using an extension of the Kelly criteria (Kelly 1956), and to then let them run while at each iterative step to shift capital from losers to winners to find a final aggregate wealth. The UP algorithm is conceptually similar to the Exp3 algorithm, introduced by Auer *et al.* (2002) to solving the adversarial bandit problem.

Here our 'experts' will be similarly characterised by a portfolio, but a portfolio that proxies different trading strategies,

*Corresponding author. Email: nicjmurphy@live.com

† *Follow-the-winner* algorithms give greater weightings to better performing experts or stocks (Li and Hoi 2014, 2016a, 2016b).

‡ The algorithm was later refined by Cover and Ordentlich (1996, see section 2.2).

where a particular agent makes decisions independently of all other experts. The UP algorithm holds parametrised Constant Rebalanced Portfolio (CRP) strategies as its 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. A key innovation was the provision of a mathematical proof for this claim based on arbitrary sequences of ergodic stationary stock return vectors (Cover 1991).

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 (Algoet and Cover 1988, Cover 1991, Cover and Ordentlich 1996, Györfi *et al.* 2008). 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*. Györfi *et al.* (2008) 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 using pattern matching. The pattern matching is implemented using a nearest-neighbour search algorithm to select parameters for experts. The pattern-matching algorithm was extended by Loonat and Gebbie (2018) in order to implement a zero-cost portfolio i.e. a long/short and self-financing portfolio. The algorithm was also re-cast to replicate near-real-time applications using look-up libraries learnt offline. However, there is a computational cost associated with coupling and creating offline pattern libraries, and the algorithms are not truly online.

A key objective in the implementation of online learning is that the underlying experts remain online, and they can be sequentially computed on a moving finite data-window using parameters from the previous time-step. It is for this reason that we choose to ignore the pattern-matching step used in prior algorithms 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. The technical trading strategies we chose are rule based and mechanistic so that the learning takes place at the aggregate process level and not at the component level associated with each individual expert.

Technical analysis indicators are popular tools used to generate trading strategies (Chan 2009, Clayburg 2002). 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 (Creamer and Freund 2010, Chande and Kroll 1994, Rechenthin 2014). What differentiates technical analysis from traditional time-series analysis is that it tends to place an emphasis on recurring time-series patterns rather than invariant statistical properties of times-series.

Many studies have criticised the lack of a theoretical foundation for many of the proposed technical analysis indicators (Lo *et al.* 2000, Aronson 2007, Lo and Hasanhodzic 2009). There has also been an abundance of academic literature utilising technical analysis for the purpose of trading, where several studies have attempted to develop indicators and test them in a more mathematically, statistically and numerically sound manner (Kestner 2003, Park and Irwin 2004, Aronson 2007). Much of this work is still viewed with suspicion given that it is extremely unlikely that this or that particular favoured strategy or approach was not the result of some sort of back-test over-fitting (Bailey *et al.* 2014, 2016, Hou *et al.* 2017).

An important precedent in the literature was the work by Creamer and Freund (2010) who used an Alternate Decision Tree (ADT) algorithm to select the best combination of a set of technical trading rules, and the optimal parameters for those rules. The authors were able to overlay the outputs of several ADT's with an online learning algorithm which recommended a long, or short position. Their experiments show that combined aggregate prediction improved predictive capacity over using a single indicator's predictions. The basis of the algorithm introduced by Creamer and Freund (2010) is similar to ours in the sense that a diverse set of technical trading rule-based expert's recommendations are combined by a performance based expert weighting online learning layer. A key feature of our implementation is that our expert outputs are not that of trained models; but rather the result of deterministic rules. This removes any form of in-sample over-fitting at the level of the rules and ensures that over-fitting can be controlled for at the level of the creation of competitive mixtures of the experts. Teixeira and de Oliveira (2010) develop a k-Nearest Neighbour (k-NN) classifier which takes a set of technical analysis and price features as inputs to a single k-NN model. They did not directly consider a performance-based expert approach where different strategies or model predictions are aggregated into a single strategy. The author's were able to conclude that for 80% of the stocks considered, they were able to outperform a Buy-and-Hold strategy in terms of profitability. However, there were no meaningful controls put in place to account for back-test over-fitting.

Moving away from technical analysis based methods, Uziel and El-Yaniv (2016) present an algorithm, EREP, that focuses on the risk regularisation of ensembles of online learning portfolio selection algorithms. The algorithm makes use of online convex optimisation in order to allocate capital to the underlying portfolio selection algorithms while encouraging non-sparsity over the given grouping. The authors provide theoretical proof for a logarithmic form of the algorithm and show evidence of impressive risk-adjusted return performance. Ban *et al.* (2018) introduce performance-based regularisation (PBR) which makes use of adapted regularisation and cross-validation techniques to constrain the sample variances of the estimated portfolio risk and return, and they provide a robust solution with reduced estimation error in performance. They provide evidence that PBR outperforms the benchmarks for two of three Fama-French data sets. Lunga and Marwala (2006) make use of a combination of an incremental online Learn++ algorithm and fractal theory to classify the direction of movement in the stock market and determine the extent to which such time series can be predicted. The data which is identified to be persistent is then passed into an incremental online neural network algorithm to predict the directional movement of the Dow Jones Average Index (Lunga and Marwala 2006), and the authors provide evidence of a high level of confidence in the model's predictions. Again, here there can be concern with back-test over-fitting due to the experimental design.

If the decision processes were sufficiently Markov or have a reasonably stable low dimensional state-space representation, it could be natural to try use reinforcement learning to estimate an appropriate algorithm to solve the technical trading problem. For example, learning algorithms for regime-switching, such as Maringer and Ramtohul (2010) and Maringer and Ramtohul (2012), or using adaptive alpha trading algorithms to discover market states online with strategy learning, such as in Hendricks and Wilcox (2014), Hendricks *et al.* (2016) and Hendricks (2017). However, these approaches add computational and model complexity to the implementation framework, while having relatively strong assumptions that can cloud the separation of learning from the underlying collections of mechanistic strategies. They provide a variety of challenges with regards to enumerating hypotheses in order to control for generalisation error. In the spirit of Occam's Razor, we have attempted to retain a sufficiently simple, yet feasible mechanistic learning framework.

Our work uses deterministic technical analysis based rules and aggregates these into a single online and adaptive recommended trading strategy, an aggregate strategy whose underlying mixture of component strategies varies and evolves through time in response to changes in the market data inputs. Hence our work cannot address the question: 'Which, if any, technical analysis methods reveal useful information for trading purposes?'; rather, we aim to bag a collection of technical experts and allow them to compete in an adversarial manner using the online learning algorithm. This allows us to consider whether the resulting aggregate strategy can: 1.) pass reasonable tests for statistical arbitrage, and 2.) has a relatively low probability of being the result of back-test over-fitting.

Concretely, here we are concerned with the idea of understanding whether the collective population of technical experts can through time lead to dynamics that can be reasonably be considered a statistical arbitrage (Jarrow *et al.* 2012), and finally with a reasonably low probability of back-test over-fitting Bailey *et al.* (2016). Can we generate wealth, both before costs, and then after costs, using the online aggregation of technical strategies? Then, will the ensemble of strategies emerge as being successful in the sense of positive trading profits with declining variance in losses?

Incorrectly accounting for costs will always be a plausible explanation for any apparently profitable trading strategy (see Loonat and Gebbie 2018), but even after costs there still exists a high likelihood that there was some data over-fitting because we only have a single price path from history with little or no knowledge about the true probability of the particular path that has been measured since the adaptive nature of markets themselves are continually changing the efficacy of various strategies and approaches.

Rather than considering various debates relating to the technicalities of market efficiency (Fama 1970), where one is concerned with the expeditiousness of market prices to incorporate new information at any time, and where information is explicitly exogenous; we restrict ourselves to market efficiency in the sense used by (Black 1986, Aronson 2007, Bouchaud *et al.* 2017). This is the situation where some of the short-term information is in fact noise, and we recognise that this type of noise is a fundamental property of real markets. Although market efficiency may plausibly hold over the longer term, in the short-term there may be small departures that are amenable to tests for statistical arbitrage (Jarrow *et al.* 2012), departures that create incentives to trade, and more importantly, departures that may not be easily traded out of the market due to various asymmetries in costs, market structure 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.* (2004) and further refined by Jarrow *et al.* (2012). Hogan *et al.* (2004) provide a plausible technical definition of statistical arbitrage based on a vanishing probability of loss and variance in the trading profits, and then use this to propose a test for statistical arbitrage using a Bonferroni test (Hogan *et al.* 2004). This methodology was extended and generalised by Jarrow *et al.* (2012) to account for the asymmetry between desirable positive deviations (profits) and undesirable negative deviations (losses), by including a semi-variance hypothesis instead of the originally constructed variance hypothesis which does not condition on negative incremental deviations. The so-called Min-*t* statistic is computed, and used in conjunction with 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 (Black 1986) 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) (Jarrow *et al.* 2012).

We make the effort here to be very clear that we do not attempt to identify profitable (technical) trading strategies nor to make any claims about the informational value of technical analysis, but rather we will generate a large population of strategies, or experts, constructed from various technical trading rules, and combinations of the associated parameters of these rules, in the attempt to learn something about the aggregate profitability of the population dynamics of the set of experts.

Experts will generate trading signals, i.e. buy, sell or hold decisions for each stock held in their portfolio as based on the underlying parameters and the necessary historic data implied by the parameters. Once trading signals for the current time period $t$ have been generated by a given expert, a methodology to transform the signals into a set of portfolio weights, or controls, is required.

We introduce a transformation method that employs a multi-objective Genetic Algorithm that simultaneously targets the maximisation of the net weighted signal and the minimisation of transaction costs, while obeying the self-financing and leverage constraints. The resulting controls are then utilised to compute the corresponding expert's wealth. The experts who accumulate the greatest wealth during a trading period will receive more wealth in the following trading period and thus contribute more to the final aggregated portfolio. 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 explicitly provide equity curves for the individual expert's portfolios, given as the accumulated trading profit through time, along with performance curves for the overall strategy's wealth and the associated profits and losses.

We perform a back-test of the algorithm on two different data sets over two separate time periods: 1) one using daily data over a six-year period, and 2) 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.† The overall strategy performance is compared to the UP algorithm as described in Kalai and Vempala (2002), which makes use of rapidly mixing non-uniform random walks and improves computational efficiency substantially, 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 *before costs*. A key point here, as in Loonat and Gebbie (2018), is that it does seem that plausible statistical arbitrages are detected on the meso-scale and in the short-term. However, after accounting for reasonable costs, only the short term trading strategies seem to pass statistical arbitrage tests. This does not imply profitability, as these remaining strategies might be structural, and hence may not be easily profitably traded out of the system.

Finally, we analyse the generalisation error of the overall strategy to get a sense of whether or not the strategy conveys back-test overfitting, by estimating the probability of back-test overfitting (PBO) inherent in multiple simulations of the algorithm on subsets of historic data.

The paper proceeds as follows: section 2 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 breakdown of the learning algorithm. In section 3, we introduce the concept of a statistical arbitrage, including the methodology for implementing a statistical arbitrage test, calculating the probability of loss and estimating the PBO for a trading strategy. All experimental results and analyses of implementations of the algorithm are presented in section 5. section 6 states 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 time-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. However, considerable care is still required to ensure that one is accounting for the full complexity of market realities that can often 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, but are rather the signature of noise trading.

## 2. 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 either a buy (1), sell $(-1)$ or hold (0) actions. These actions are independent of one another and based on each of the individual experts strategies. The trade signals $\{-1, 1, 0\}$ are transformed into a set of portfolio weights such that their sum is identical to zero; this ensures that the strategy can be self-funding. We also require that the portfolio is unit leveraged, and hence the absolute sum of controls is equal to one. This is to avoid having to introduce a margin account into the trading mechanics.‡

Based on each individual expert's accumulated wealth up until some time $t$, a final aggregate portfolio for the next period $t+1$ 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.

### *2.1. Expert generating algorithm*

**2.1.1. Technical trading.** Technical trading refers to the practice of using trading rules derived from technical analysis indicators to generate trading signals. Here, indicators refer to mathematical formulas based on Open-High-Low-Close (OHLC) price bars, volume traded or a combination of both

† More detail on the data sets can be found in section 4.

‡ We could in fact have considered leveraged trading but avoided this for simpler trading mechanics. This has no impact on the results of the statistical arbitrage test.

(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 and different human operators, 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.[†] 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. These three rules generically conform to a combination of chartists trend followers, fundamentalist contrarians, and short-term correlation traders. An explanation of these three algorithms is provided in section 2 while each of the technical strategies are described in section 1.

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 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 two parameters represent the number of short and long-term look-back periods necessary for the indicators used in the rules. These parameters will determine the amount of historic data considered in the computation of each rule. 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 $K = |\boldsymbol{n}_2|$[‡] 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 $W_2$ and hence $W = W_1 + W_2$.

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 sector clusters of stocks on the JSE, namely, Resources, Industrials and Financials.[§]

The algorithm will loop over all combinations of these four model parameters calling the appropriate strategies, $\boldsymbol{\omega}$, stocks, $\mathbf{c}$, and amount of historic data, $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$, to create a: buy, sell or hold signal at each time period $t$. Each combination of $\boldsymbol{\omega}(i)$ for $i = 1, \ldots, W$, $\mathbf{c}(j)$ for $j = 1, \ldots, C$, $\boldsymbol{n}_1(\ell)$ for $\ell = 1, \ldots, L$ and $\boldsymbol{n}_2(k)$ for $k = 1, \ldots, K$ will represent an expert. It should be clear that some experts may trade all the stocks, *i.e* use the trivial clusters, and others will trade subsets of the stocks, *i.e* resources, industrials and financials. It is also important to note that for rules requiring two parameters, the loop over the long-term parameters will only activate at indices $k$ for which $\boldsymbol{n}_1(\ell) < \boldsymbol{n}_2(k)$ where $\ell$ and $k$ represent the loop index over the short and long-term parameters respectively. The total number of experts, $\Omega$, is then given by

$$
\begin{aligned}
\Omega &= \text{no. of experts with 1 parameter} \\
&\quad + \text{no. of experts with 2 parameters} \\
&= C \cdot L \cdot W_1 + C \cdot W_2 \\
&\quad \cdot \sum \Big[ \sum (\boldsymbol{n}_2 > \max(\boldsymbol{n}_1)) : \sum (\boldsymbol{n}_2 > \min(\boldsymbol{n}_1)) \Big]
\end{aligned}
$$

We will denote each expert's strategy[¶] 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. As mentioned above, from the set of $m$ stocks, 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 expert's 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 in their portfolio. Denote the expert control matrix $\mathbf{H}_t$ 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 number of days, denoted by $\delta_{\text{liq}}$. We make the choice of using average daily volume (ADV) as a proxy for liquidity.[∥] ADV is simply the average volume traded for a given stock over a period of time[**] . The $m$ stocks with the largest ADV over the past period $\delta_{\text{liq}}$ will then be fed into the algorithm for trading at time $t$.

**2.1.2. 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 ($\mathbf{h}_t^n$). 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.

In the classical mathematical finance literature, there are vast variety of models that can be used to manage, not only the decision to buy, sell or hold, but also the optimal direction of trading and the quantities to be traded - this includes both regime-switching methods (Guo 2001, Buffington and Elliot 2002, Zhang and Guo 2004, Yiu *et al.* 2010, Zhou *et al.* 2020), as well as using various impulse response controls (Korn 1998, Guilbaud and Pham 2013). However, in practice many of these are poorly suited for online mechanistic

[†] This is also known as data-mining Aronson 2007.

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

[§] See the supplemental material in section 6 for a breakdown of the three sectors into their constituents for daily and intraday data.

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

[∥] 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.

[**] The ADV of stock $m$ over $\delta_{\text{liq}}$ past periods is $\text{ADV}_m = \frac{1}{\delta_{\text{liq}}} \sum_{t=1}^{\delta_{\text{liq}}} \text{V}_t^m$ where $\text{V}_t^m$ is the volume of the $m$th stock at time $t$.

learning that is often path-dependent, and in the presence of purposeful strategic agents.

In our approach, 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.† All non-hold signals at time $t$ are of course not replaced by the previous periods signals as the expert has taken a completely new position in the stock. This implies that when the position in a given stock was short at period $t-1$ for example and the current periods ($t$) signal is long then the expert takes a long position in the stock rather than neutralising the previous 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 this combined set of signals as *output signals*, denoted by $\boldsymbol{s}$.

We make use of a multi-objective Genetic Algorithm (GA) to perform the signal to weight transform transformation. The two objectives we target are the maximisation of the weighted signal of the portfolio ($f_1$)‡ and the minimisation of transaction costs ($f_2$) as shown in equation (1). For more details on the transaction cost objective, $f_2$, please refer to section 2.5. The constraints that are enforced are: the self-financing constraint where the controls must sum to zero, the leverage constraint that ensures that the absolute value of the controls sum to unity, and that both the long and short sides should have equal capital allocations.

$$\min_{\mathbf{w}} \begin{cases} f_1(\mathbf{w}) = -\mathbf{w}.\boldsymbol{s} \\ f_2(\mathbf{w}) = M \times \text{Spread} + \sigma.\sqrt{n/\text{ADV}} \end{cases}$$

$$\text{s.t.} \quad \sum_i \mathrm{w}_i = 0$$

$$\sum_i |\mathrm{w}_i| = 1$$

$$\sum_i \mathbf{w}(\mathrm{w}_i < 0) = -0.5$$

$$\sum_i \mathbf{w}(\mathrm{w}_i > 0) = 0.5 \qquad (1)$$

In any instance in which all the output signals are zero (hold), we set all controls to zero without optimisation, to avoid the possibility of infeasible solutions. In cases where a Pareto front of multiple solutions is returned from the GA, which is common in multi-objective optimisation, we employ the Weighted Sum Method (WSM) to pick an optimal solution from the front and weight the objectives equally (0.5 for each objective).§ For each Pareto solution we compute the weighted sum over each of the preference based objective weights using the normalised objective function values. The solution with the greatest weighted sum value across the two objectives is then chosen as the final, optimal solution.

If for whatever reason the solution generated by the GA is infeasible, we compute controls on each of the sides of the portfolios proportional to their normalised volatility. This ensures that more volatile stocks receive higher weight allocations. A risk-free asset is then used to balance the portfolio in cases where any constraints were violated.

### 2.2. *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 expert's 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 Cover (1991) and Cover and Ordentlich (1996) and a modified version proposed by Györfi *et al.* (2008). Although, due to the fact that we have several different base experts as defined by the different trading strategies rather than Cover's (see Cover 1991) constant rebalanced UP strategy, our algorithm is better defined as a meta-learning algorithm (Li and Hoi 2016a). 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 $t_{\text{min}}$¶ 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 steps (Loonat and Gebbie 2018):

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

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

‡ We multiply the objective by a negative to convert it into a minimisation problem.

§ The basic idea of the WSM is that each of the objectives are min-max normalised across all the Pareto solutions.

¶ This is the time at which trading commences.

portfolio wealth for the $t$th period:

$$S_t = S_{t-1}\Delta S_t \quad \text{where}$$
$$\Delta S_t = \sum_{m=1}^{M} b_{m,t-1}(x_{m,t+1} - 1) + 1. \tag{2}$$

$S_t$ 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

$$\mathrm{PL}_t = \mathrm{PL}_{t-1} + \Delta S_t - 1. \tag{3}$$

(ii) **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 $\Omega$ experts and $M$ objects about which the experts make expert capital allocation decisions. At the end of the $t$th time period the performance of each expert $n$, $Sh_t^n$, can be computed from the change 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:

$$Sh_t^n = Sh_{t-1}^n \Delta Sh_t^n \quad \text{where}$$
$$\Delta Sh_t^n = \left[\sum_{m=1}^{M} \mathbf{h}_t^n (x_{m,t} - 1)\right] + 1. \tag{4}$$

(iii) **Update expert mixtures**: We consider a UP inspired expert mixture update rule as follows (Cover 1991, Györfi *et al.* 2008, Loonat and Gebbie 2018): 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

$$q_{n,t+1} = Sh_t^n \tag{5}$$

(iv) **Renormalise 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 2018), 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.

$$q_{n,t+1} = \frac{q_{n,t+1} - \frac{1}{N}\sum_{n=1}^{\Omega} q_{n,t+1}}{\sum_{n=1}^{\Omega} |q_{n,t+1} - \frac{1}{N}\sum_{n=1}^{\Omega} q_{n,t+1}|} \tag{6}$$

(v) **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 $\mathbf{h}_t^n$ for each expert $n$ from the expert generating algorithms using information from time period $t$. We then take a weighted average over all $n$ experts by taking the sum with respect to $n$

$$b_{m,t+1} = \sum_n q_{n,t+1}\mathbf{h}_t^n \tag{7}$$

The strategy is to implement the portfolio controls, wait until the end of the current time 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.

### 2.3. *Algorithm implementation for intraday-daily trading*

Intraday trading poses a whole new set of new dynamics that need to be considered. It is not as straight-forward as substituting higher frequency bar-data into an algorithm which has been built and tested on uniformly sampled daily data, due to the inequivalence of uniformly sampled daily closing auction data and uniformly sampled intraday bar-data. The end-of-day price discovery process is entirely different to that found intra-day; the prior is a closing auction, the latter has prices being the result of continuous-time trading in a double auction. In addition to this, the first and last data-points of each day lead to over-night gap effects that will lead to spurious signals if not aggregated correctly over multiple days. Rather than dealing with the full complexity of a money management strategy, the main issue that we are concerned with will be the over-night gap effect, which relates to the deviation in the prices at the end of day $t-1$ and the start of day $t$. To do this, we will implement the learning algorithm for each day completely independent of the previous days dynamics. The algorithm will essentially reset back to the initial setting each day, by setting the wealth processes of the expert's back to unity at the start of the day. We then combine the intraday data implementation with the daily data implementation, such that 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 aggregated. We will refer to trading using a combination of daily and intraday data as *intraday-daily* trading.

The best way to think about it is to consider the experts as trading throughout the day, making decisions based solely on intraday data. They compound their wealth throughout the trading day, and once a trading decision is made at the final time bar,[†] the expert makes one last trading decision on that day based exclusively on daily historic OHLCV data. The last trading decision is based on look-back periods on the passed trading days, and not on any intraday time bars for that day. The daily trading decision can be thought of as representing the last time bar of the day. Here we are just using different set of data (daily time bars as opposed to intraday time bars) to make the decision for that bar. The methodology for each of the intraday and daily trading mechanisms are almost exactly as explained in section 2.2 above, however, there are a few necessary alterations to the algorithm.

To introduce some notation for intraday-daily trading, let $\mathbf{Sh}^F_{t,t_I}$ be the expert wealth vector for all $n$ experts for the $t_I$th 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. Then, expert wealth is computed as before with $Sh^F_{n,t,t_I} = Sh^F_{n,t,t_I-1} \cdot \Delta Sh^F_{n,t,t_I}$ where $\Delta Sh^F_{n,t,t_I}$ is the $n$th experts return at time bar $t_I$ on day $t$, as in equation (7). More specifically, $\mathbf{H}^F_{t,t_I}$ will contain the 88 intraday expert controls followed by the end of day expert controls based on daily closing OHLCV data for each given day over the trading horizon. In other words, $\mathbf{H}^F_{t,t_I}$ is simply the concatenation of the expert's controls for the intraday time bars and the expert's controls for the daily time bar at the end of the day. Denote $T_I$ as the final time bar in a day (4:30pm). The $n$th experts wealth accumulated up until the final time bar $T_I$[‡] on day $t$, $Sh^F_{n,t,T_I+1}$, is calculated from the $n$th column of the expert control matrix $\mathbf{H}^F_{t,t_I}$, denoted $\mathbf{h}^F_{n,t,T_I}$, from the previous period $T_I$ and is computed solely from intraday data for day $t$. Then $\Delta Sh^F_{n,t,t_I} = \sum_m \mathbf{h}^F_{n,t,T_I}(x_{m,t} - 1) + 1$, by equation (6). Overall portfolio controls for the final intraday trade on day $t$ ($\mathbf{b}_{t,T_I+1}$) are computed as before (see equation (10)), along with the overall portfolio wealth $S_{t,T_I+1}$ (equation (5)). This position is held until the close of the day's trading when the closing prices of the $m$ stocks $\mathbf{P}^C_t$ are 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 controls for the daily time bar at the end of the day. 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 $t$ are computed ($\mathbf{r}_t = \frac{\mathbf{P}^C_t}{\mathbf{P}^C_{t-1}}$) and the cumulative wealth 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 $Sh^F_{n,t+1,1}$[§] is set back to unity. 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 4 and hence controls will all be set to zero. As the trading day proceeds, experts will begin producing non-zero controls as soon as there is sufficient 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[¶] (risk-free asset) is only posted daily, we utilise the same STeFI value for trading throughout the day.

### 2.4. *Online portfolio benchmark algorithm*

For comparative purposes, we utilise the Universal Portfolio algorithm (Cover 1991, Cover and Ordentlich 1996) as the performance benchmark and follow the implementation of the algorithm described in Li and Hoi (2016a). The key principle behind the universal algorithms (Cover 1991, Cover and Ordentlich 1996) is that of buying and holding a set of parameterised (CRP) strategies as experts, and to then let them run (Li and Hoi 2016a). The algorithm then pool's together the expert's wealth by investing amounts into each expert strategy proportional to the performance of that expert. For practical reasons, the computational time becomes exponential in the number of stocks. For this reason (Kalai and Vempala 2002) introduced a method that involves rapidly mixing non-uniform random walks to substantially reduce the run-times (Li and Hoi 2016a, 2016b). This is achieved by sampling from a biased distribution of portfolios; rather than sampling completely random portfolios. This biased distribution weights portfolios in proportion to their performance by making a modification to the wealth function, and in doing so, reduces the run-time by orders of magnitude compared to the traditional UP algorithm.

### 2.5. *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 components of these transaction costs, besides commissions charged by exchanges, are the spread[‖], price impact (Harvey *et al.* 2017) and opportunity cost (The Hidden Costs of Trading n.d.).

[†] A time bar is defined as some fixed time interval (one day or 5-minutes as referred to above) over which we observe a set of features, such as prices or volume.

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

[§] 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.

[¶] See section 4 for more details on the STeFI index.

[‖] Spread = best ask price minus best bid price.

To estimate indirect transaction costs (TC) for each period $t$, we will consider is the *square-root formula* (Gatheral 2010)

$$\mathrm{TC} = \text{Spread Cost} + \sigma\sqrt{\frac{n}{\mathrm{ADV}}} = M \times \text{Spread} + \sigma\sqrt{\frac{n}{\mathrm{ADV}}} \tag{8}$$

where:

(i) **Volatility of the returns of a stock ($\sigma$)**: See section 2.5.1 below.
(ii) **Average daily volume of the stock (ADV)**: ADV is computed using the previous 90 days trading volumes for daily trading and the previous 5 days intraday trading volume for intraday-daily trading.
(iii) **Number of shares traded ($n$)**: The number of shares traded ($n$) is taken to be 1bp of ADV for each stock per day for daily trading. The number of stocks traded for intraday-daily trading is assumed to be 70bps of ADV for the entire portfolio per day which is then split evenly among all active trading periods during the day, to get 70bps/85 (assuming 88 5-minute time bars per day) of ADV per stock, per trading period.
(iv) **Spread**: Spread is assumed to be 1bps per day (1%% /pd) for daily trading. For intraday-daily trading, we assume 20bps per day, which we then split evenly over the day to incur a cost of 0.002/85 per time bar.
(v) $M$: Number of times the trading signal changes from a buy (sell) to a sell (buy,) for all stocks in the portfolio, over consecutive trading periods.†

The use of the square-root rule in practice dates back many years and is often used as a pre-trade transaction cost estimate (Gatheral 2010). The first term in equation (8) can be regarded as the term representing the *slippage*‡ or *temporary price impact* and results due to our demand for liquidity (Huberman and Stanzl 2005). This cost will only impact the price at which we execute our trade at and not the market price (and hence the price of subsequent transactions). The second term in equation (8) 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 (Bouchaud *et al.* 2004). In the following subsection, we will discuss how the volatility ($\sigma$) is estimated for the square-root formula. Technically, $\sigma$, $n$ and ADV in equation (8) 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.

Each trade (say an entry) is consider a child order of the trading strategy across the period. This is not really a faithful representation but was chosen in order to optimise the historic simulation. It should be realised that each child-order is not of a fixed size, as this is determined by the algorithm. However, at the end of the day we have $n$ shares traded, and $M$ entry and exit pairs.

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 (Loonat and Gebbie 2018). 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. This assumption is purely made to approximately match the total daily transaction cost assumption made by Loonat and Gebbie (2018). For the intraday-daily implementation a total direct cost of 70bps per day is assumed (following Loonat and Gebbie 2018) which we then split evenly over each day's active trading periods (85 time bars since first expert only starts trading after the 5th time bar) to get a cost of 70bps/85 per period. These costs are indicative and actual price impact requires either real trading experiments or full-scale market simulation - both are intractable in the context of our approach.

For daily trading, we recover an average daily transaction cost of roughly 17.75bps which is almost double the 10bps assumed by Loonat and Gebbie (2018). The authors 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 2.17bps, while the average cost per day assuming we trade for 85 periods throughout each day is roughly 184bps (85*2.17) for intraday-daily trading.

### 2.5.1. Volatility estimation for transaction costs.

In this section, we will discuss different methods for calculating the estimates for volatility ($\sigma$) for daily and intraday data in the square-root formula (equation (8)).

*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.

*Intraday Data Estimation.* The volatility for each intraday time bar $t_l$ 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 day's volatility estimates are computed using the Realised Volatility (RV) method (Andersen *et al.* 2001). RV is one of the more popular methods for estimating volatility of high-frequency returns§

† Every trade with an entry and exit should be assumed to have crossed the spread and as such incurred a direct cost associated with the spread, here estimated as in the above bullet point. The price impact of a trade is assumed to follow the square-root-law, so that the price impact cost of the trade is proportional to the realised volatility across the trade. Here, this is costed as a parent order of size $n$ over each trading period.

‡ 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.

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

computed from tick data. The measure estimates volatility by summing up intraday squared returns at short intervals (eg. 5 minutes). Andersen *et al.* (2001) 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 time volatility (quadratic variation) (Poon 2008).

## 3. Testing for statistical arbitrage

To test the overall trading strategy for statistical arbitrage, we implement a novel statistical test originally proposed by Hogan *et al.* (2004) and later modified by Jarrow *et al.* (2012), 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 Jarrow *et al.* (2012), 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 Hogan *et al.* (2004). 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)† $s_t$ and a money market account $B_t$‡ 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 the 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 Hogan et al. 2004, Jarrow et al. 2012*

*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:*

(i) $\nu(0) = 0$,
(ii) $\lim_{t\to\infty} \mathbb{E}^{\mathrm{P}}[\nu(t)] > 0$,
(iii) $\lim_{t\to\infty} \mathrm{P}[\nu(t) < 0] = 0$, *and*
(iv) $\lim_{t\to\infty} Var[\Delta\nu(t)|\Delta\nu(t) < 0] = 0$.

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 (losses) 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 (De Wit 2013).

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

$$\Delta\nu_i = \mu i^{\theta} + \sigma i^{\lambda} z_i \tag{9}$$

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:

$$z_i = \epsilon_i + \phi\epsilon_{i-1} \tag{10}$$

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

$$\Delta\nu_i = \mu + \sigma i^{\lambda} z_i \tag{11}$$

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

$$\nu(T) = \sum_{i=1}^{T} \Delta\nu_i \sim N\left(\mu \sum_{i=1}^{T} i^{\theta}, \sigma^2 \sum_{i=1}^{T} i^{2\lambda}\right) \tag{12}$$

From equation (12), it is straightforward to show that the log-likelihood function for the discounted incremental trading profits is given by:

$$\ell(\mu, \sigma^2, \lambda, \theta|\Delta\nu) = \ln L(\mu, \sigma^2, \lambda, \theta|\Delta\nu) = -\frac{1}{2}\sum_{i=1}^{T} \ln(\sigma^2 i^{2\lambda}) - \frac{1}{2\sigma^2}\sum_{i=1}^{T} \frac{1}{i^{2\lambda}}(\Delta\nu_i - \mu i^{\theta})^2 \tag{13}$$

The probability of a trading strategy generating a loss after $n$ periods is as follows (Jarrow *et al.* 2012)

$$\mathbf{Pr}\,\{\text{Loss after } n \text{ periods}\} = \Phi\left(\frac{-\mu\sum_{i=1}^{n} i^{\theta}}{\sigma(1+\phi)\sqrt{\sum_{i=1}^{n} i^{2\lambda}}}\right) \tag{14}$$

where $\Phi(\cdot)$ denotes the cumulative standard normal distribution function. For the CM model, equation (14) is easily

† In our study, we will be considering a portfolio.
‡ The money market account is initialised at one unit of a currency *i.e.* $B_0 = 1$.

adjusted by setting $\phi$ and $\theta$ 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 3.1 (UM Model Hypothesis Jarrow *et al.* 2012) *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: i.)* $H_1 : \mu > 0$*, ii.)* $H_2 : -\lambda > 0$ *or* $\theta - \lambda > 0$*, iii.)*$H_3 : \theta - \lambda + \frac{1}{2} > 0$*, and* $H_4 : \theta + 1 > 0$.

An intersection of the above sub-hypotheses defines a statistical arbitrage, and as by De Morgan's Laws,† 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 3.1:

Proposition 3.2 *(UM Model Alternative Hypothesis Hogan et al. 2004, Jarrow et al. 2012) 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: i.)* $H_1 : \mu \leq 0$*, ii.)* $H_2 : -\lambda \leq 0$ *or* $\theta - \lambda \leq 0$*, iii.)* $H_3 : \theta - \lambda + \frac{1}{2} \leq 0$ *, and iv.)* $H_4 : \theta + 1 \leq 0$

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 Jarrow *et al.* (2012)

$$\begin{aligned} \text{Min}-t = \min\{ & t(\hat{\mu}), \quad t(\hat{\theta} - \hat{\lambda}), \quad t(\hat{\theta} - \hat{\lambda} + 0.5), \\ & \text{Max}[t(-\hat{\lambda}), t(\hat{\theta} + 1)]\} \end{aligned} \tag{15}$$

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 $\alpha$. Since the probability of rejecting cannot exceed the significance level $\alpha$, we have the following condition for the probability of rejecting the null at the $\alpha$ significance level

$$\mathbf{Pr}\{\text{Min}-t > t_c | \mu, \lambda, \theta, \sigma\} \leq \alpha \tag{16}$$

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 3.1 step 3.1 below.

### *3.1. Outline of the statistical arbitrage test procedure*

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

(i) **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 $\Delta\nu_i = \nu(t_i) - \nu(t_{i-1})$.

(ii) **Perform MLE**: Compute the likelihood function, as given in equation (13), and maximise 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 optimisation algorithm‡ only performs minimisation, we minimise the negative of the log-likelihood function *i.e.* maximise the log-likelihood.

(iii) **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 equation (13). 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.

(iv) **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 equation (15). 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.

(v) **Critical values**: Compute the critical value at the $\alpha$ 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 equation (11) with $(\mu, \lambda, \sigma^2) = (0, 0, 0.01)$.§ 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-$\alpha$ quantile of the resulting distribution of Min-$t$ values.

(vi) ***P*-values**: Compute the empirical probability of rejecting the null hypothesis at the $\alpha$ significance level using equation (16) by utilising the critical value from the previous step and the simulated Min-$t$ statistics.

(vii) **n-Period Probability of Loss**:Compute the 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 equation (14).

† This states that the complement of the intersection of sets is the same as the union of their complements.

‡ Here, we are referring to MATLAB's *fmincon* function.

§ $t_c$ is maximised when $\mu$ and $\lambda$ are zero. $\sigma^2$ is set equal to 0.01 to approximate the empirical MLE estimate Jarrow *et al.* 2012.

### 3.2. Estimates of the probability of back-test overfitting (PBO)

Bailey *et al.* (2014) heavily criticise recent studies which claim to have designed profitable investment or trading strategies since many of these studies are only based on in-sample (IS) statistics, without evaluating out-of-sample (OOS) performance. We briefly addressed this concern by computing estimate of the probability of back-test overfitting (PBO) using the combinatorially symmetric cross-validation (CSCV) procedure outlined in Bailey *et al.* (2016). Typically, an investor/researcher will run many (N) trial back-tests to select the parameter combinations which optimize the performance of the algorithms used, based on statistical inference using a performance criterion such as the Sharpe ratio (Lo 2002, Bailey and López de Prado 2014). The idea is to perform CSCV on the matrix of performance series over time of length $T_{\mathrm{BL}}$[†] for the $N$ separate trial simulations of the algorithm.

Here, we must be clear that when we refer to IS, we do not mean the 'training set' per say, during which the moving average look-back parameters were calculated for example. Rather, we refer to IS as being the subset of observations utilised in selecting the optimal strategy from the $N$ back-test trials.

In the case of the algorithm proposed in this study, since the large set of trialled parameters form the basis of the learning algorithm in the form of the experts, we cannot observe the effect of different parameters settings on the overall strategy, as these are already built into the underlying algorithm. Rather, we will run $N$ trial back-test simulations on independent subsets of historical data to get an idea of how the algorithm performs on different subsets of unseen data. We can then implement the CSCV procedure on the matrix of profits and losses resulting from the trials to recover a PBO estimate. Essentially, there is no training of parameters taking place in our model, as all parameter combinations are considered, and the weights of the performance weighted average of the expert's strategies associated with the different parameter combinations are 'learnt'.

More specifically, we choose a back-test length $T_{\mathrm{BL}}$ for each subset and split the entire history of OHLCV data into subsets of this length. The learning algorithm is then implemented on each subset to produce $N = \lfloor T/T_{\mathrm{BL}} \rfloor$ profit and loss time series. Note that the subsets will be completely independent from one another as there is no overlapping of the data that each separate simulation is run on. The results from the simulations: First, for the daily strategies with the number of back-test trials as $N = 30$, and with a back-test length of $T_{\mathrm{BL}} = 60$ days, the resulting probability of back-test over-fitting was PBO = 1.4%. Second, for the intra-daily strategies we had $N = 22$ trials, over $T_{\mathrm{BL}} = 3$ days, with a PBO = 11.4%.

## 4. The data

### 4.1. Daily data

The daily data is sourced from Thomson Reuters and contains data corresponding to all stocks listed on the JSE Top 40.[‡] The data set consists of data for 42 stocks over the period 01-01-2005 to 29-04-2016 (Murphy and Gebbie 2019). Here we will only utilise the stocks which have been listed on the exchange before more than 40% of the total number of days over this period have passed. 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 ($V$) (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) (etfSA 2011).

### 4.2. 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 (Murphy and Gebbie 2019). 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. As in the daily data case, we make use of the STeFI index as the risk-free asset, and hence the daily entries for the STeFI index are included in this data set. 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.

## 5. Results and analysis

### 5.1. Daily data

In this section, 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 4 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 $\mathrm{PL}_t$. Third, the cumulative expert wealth achieved by each expert over time $\mathbf{Sh}_t$. Finally, the cumulative wealth paths of each of the strategies calculated by computing the mean over all experts for each given trading strategy or 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

[†] The subscript BL stands for back-test length.

[‡] Please refer to the supplemental material in section 6 for the full names of the Bloomberg ticker symbols.

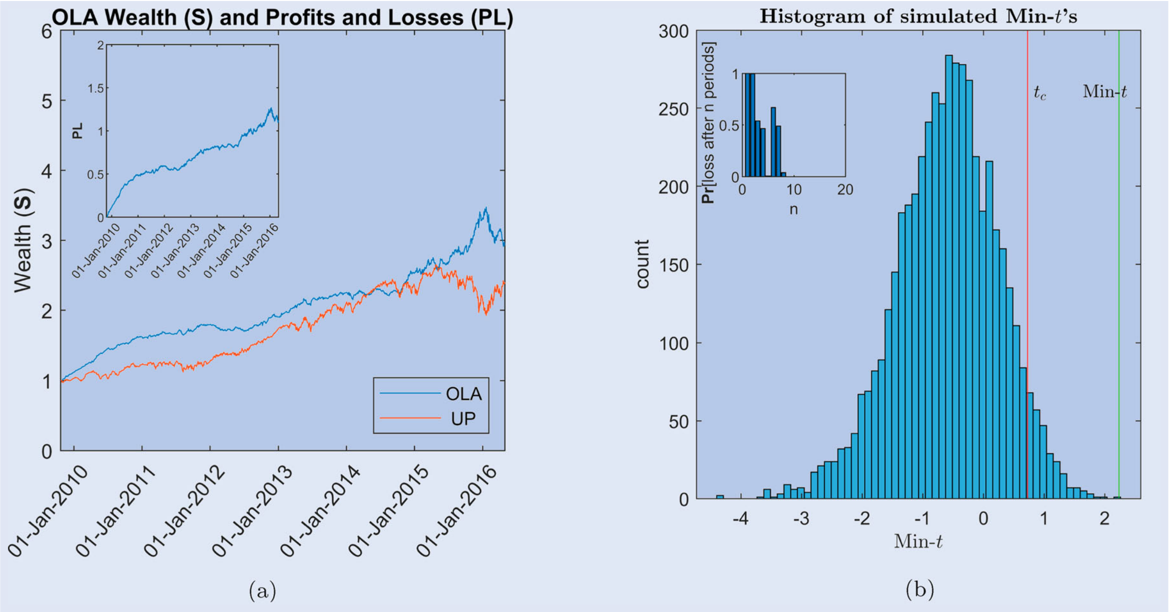


Figure 1. The performance of the algorithm (figure 1(a)) and the results of the statistical arbitrage test (figure 1(b)) on daily data without any transaction costs incorporated. On the left, the overall cumulative portfolio wealth ($S$) for daily data with no transaction costs (blue) and the benchmark UP strategy (orange). The figure inset illustrate the associated profits and losses (**PL**) of the strategy. On the right, the histogram of the 5000 simulated Min-$t$ statistics resulting from the CM test implemented on the simulated incremental process given in equation (11) along with the Min-$t$ statistic (green) for the overall strategy's profit and loss sequence over the 400-day period stretching from the 30th trading day until the 430th trading day without any account for transactions costs. The figure inset displays the probability of loss for each of the first 20 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 $n = 1, \ldots, 20$. (a) Daily algorithm performance without costs and (b) Daily statistical arbitrage test without costs.

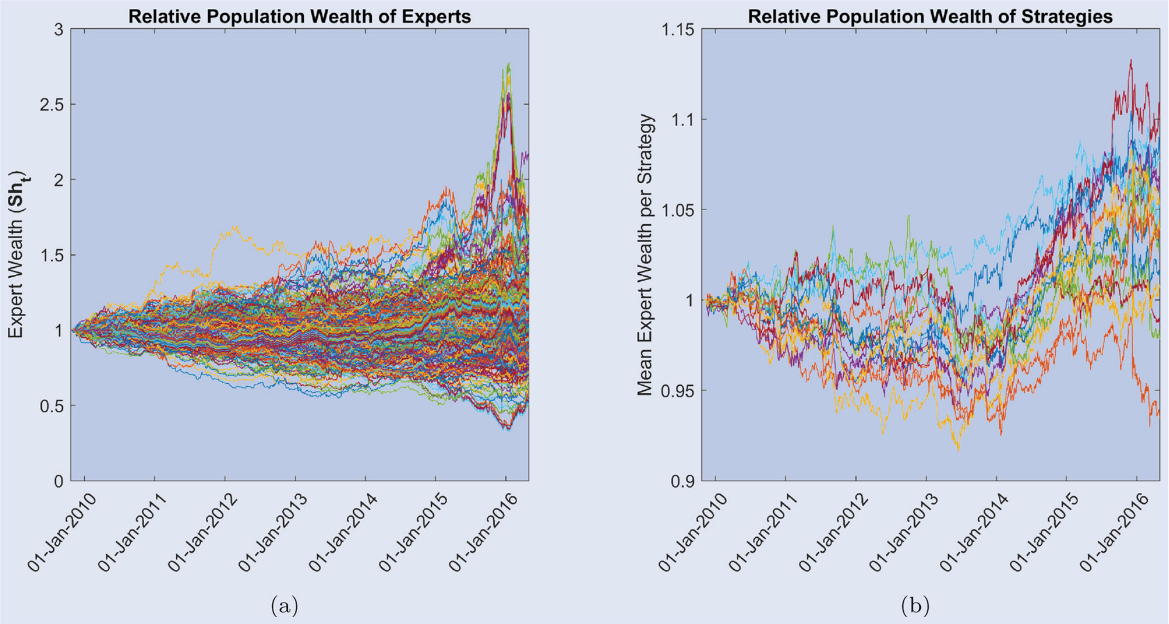


Figure 2. Figure 2(a) illustrates the expert wealth (**Sh**) for all Ω experts for daily data with no transaction costs. figure 2(b) illustrates the mean expert wealth of all experts for each trading strategy ($\boldsymbol{\omega}(i)$) for daily data with no transaction costs.

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.

**5.1.1. No transaction costs.** Barring transaction costs, it's clear that the portfolio makes favourable cumulative returns on equity over the six-year period as is evident in figure 1(a). The performance of the online learning algorithm (blue) is similar to that of the benchmark UP strategy (orange) which is promising as the original literature proves that the algorithm should track such a benchmark in the long-run. The figure inset in figure 1(a) illustrates that the overall strategy 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 show that on average, the underlying

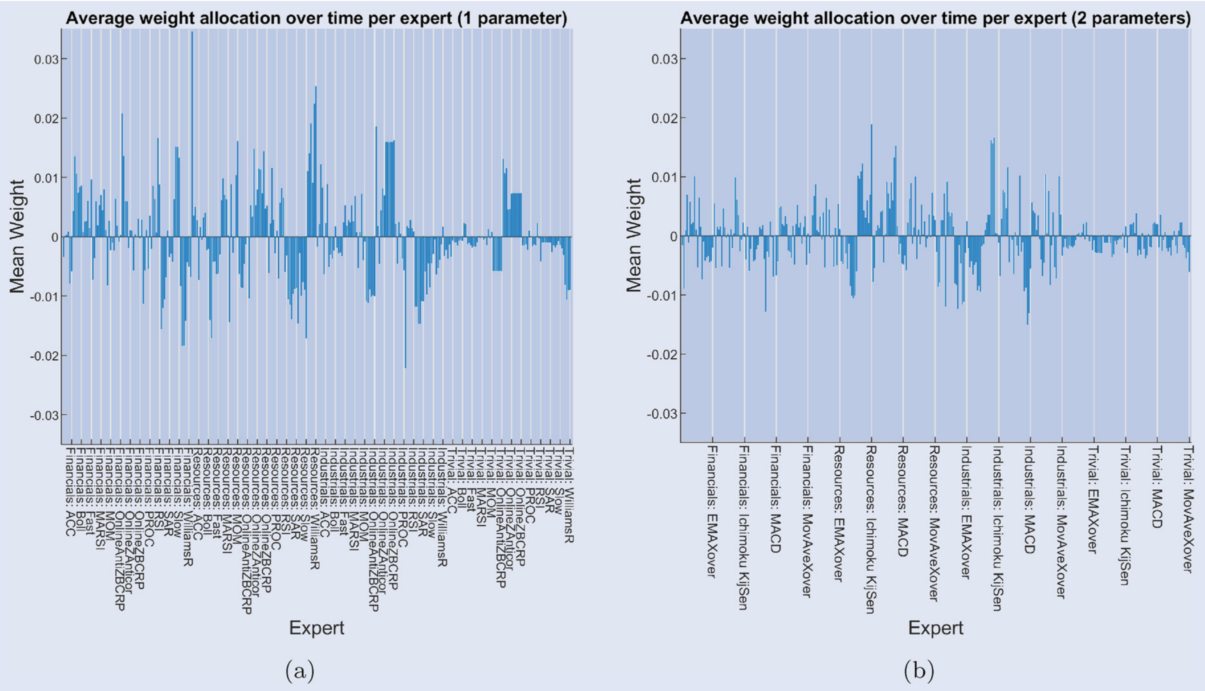


Figure 3. Figure 3(a,b) illustrate the mean expert mixture, per expert over time for the daily data implementation, for experts with a single lookback and two lookback parameters, respectively. The labels on the $x$-axis are of the format ‘object cluster ($\mathbf{c}$)’: ‘strategy ($\boldsymbol{\omega}$)’.

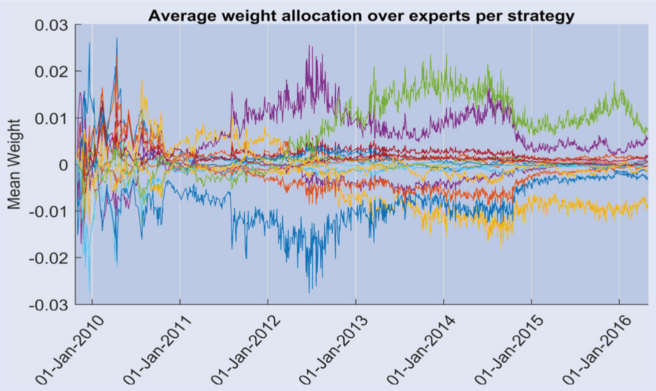


Figure 4. The mean expert mixture value ($q$) across all experts for a given strategy $\boldsymbol{\omega}$, over time for the daily data implementation.

experts perform fairly poorly compared to the overall strategy however there is evidence that some experts make satisfactory returns over the period.

In figure 3(a,b) we present an illustration of the mean expert mixture ($q$), per expert, over time for experts with a single lookback and two lookback parameters, respectively. We label experts on the x-axis in the format ‘object cluster ($\mathbf{c}$)’: ‘strategy ($\boldsymbol{\omega}$)’. It is quite clear from the figures that certain experts seem to be favoured more consistently over time than others, and this is especially clear in figure. 4, which illustrates the mean expert mixture allocation per strategy $\boldsymbol{\omega}$. The trend-following strategies do receive higher allocation on average than the other strategies, in a more consistent manner over time. That’s not to say that there are certain periods where the other strategies outperform the trend-following counterparts, but there is evidence of certain strategies being slightly redundant. Rather than cutting out the poorly performing experts completely, and possibly introducing overfitting, we believe that the learning algorithm is able to adjust to such cases, and effectively pick the winners over the losers. Another interesting observation is the initial allocations for the first year or so are quite erratic before the algorithm develops more confidence in allocating wealth among the experts in a more consistent manner, due to the expert’s wealth series’ gathering momentum.

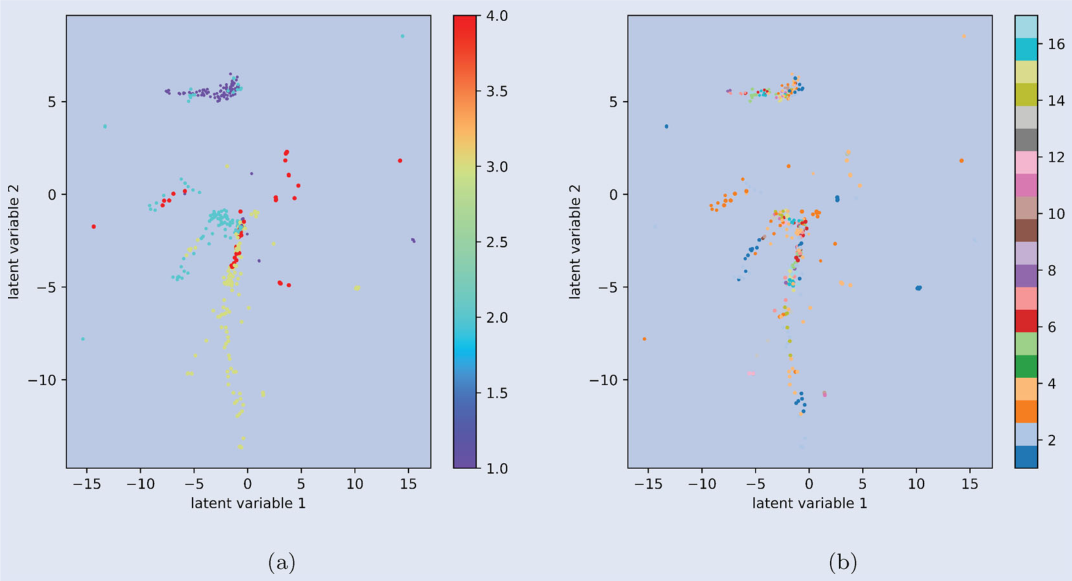
Table 1 provides the group summary statistics of the terminal wealth’s of experts, where experts are grouped based on their underlying strategy $\boldsymbol{\omega}(i)$. The online Z-Anticor† algorithm produces the best expert (maximum terminal wealth) followed by the Online Z-BCRP algorithm, while Online Z-BCRP also produces experts with the greatest mean terminal wealth over all experts (column 2). Additionally, Williams %R and Z-Anticor produce expert’s with wealth’s that vary the most (highest standard deviation). Williams %R produces the worst expert (minimum terminal wealth). The trading rules with the lowest mean terminal wealth and worst mean ranking‡ are Online Anti-Z-BCRP and MOM, respectively.

Figure 5(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 with experts coloured by object cluster. A VAE is a type of autoencoder that maps (encodes) input data, typically onto a lower dimensional representation of the data called the latent space, by learning the parameters of a probabilistic inference model. The inference model approximates the posterior distribution of the latent variables of the generative model, which in turn tries to reproduce the

† See section 1 and section 2 for a detailed description of the various trading rules mentioned in the table 1.
‡ We rank all of the experts, in ascending order, by their terminal wealth and then take the mean rank across all the experts for each given strategy group.

Table 1. Group summary statistics of the terminal wealth of the experts grouped by their underlying strategy ($\boldsymbol{\omega}(i)$ where $i = 1, \ldots, 17$) for daily trading. In brackets next to the mean terminal wealth, we rank all of the experts in terms of their terminal wealth and then take the mean rank of the experts for each given strategy group.

| Strategy | Mean (mean rank) | St. Dev. | Min | Max |
|---|---|---|---|---|
| EMA X-over | 0.99757 (325.96) | 0.21585 | 0.65397 | 1.33044 |
| Ichimoku Kijun Sen | 1.07354 (323.04) | 0.28298 | 0.70637 | 1.58385 |
| MACD | 1.03040 (322.58) | 0.15048 | 0.64901 | 1.27397 |
| Moving Ave X-over | 1.10634 (344.46) | 0.14231 | 0.70668 | 1.38691 |
| ACC | 1.03841 (270.86) | 0.20041 | 0.64234 | 1.59484 |
| BOLL | 1.03847 (275.02) | 0.16928 | 0.60375 | 1.43919 |
| Fast Stochastic | 1.11073 (304.08) | 0.07783 | 0.93109 | 1.20393 |
| MARSI | 1.04047 (353.25) | 0.20797 | 0.55914 | 1.5516 |
| MOM | 1.06311 (260.57) | 0.14015 | 0.71269 | 1.49720 |
| Online Anti-Z-BCRP | 0.76010 (357.04) | 0.16935 | 0.51519 | 1.06755 |
| Online Z-Anticor | 1.22578 (370.17) | 0.34405 | 0.60704 | 2.14513 |
| Online Z-BCRP | 1.29050 (370.67) | 0.23255 | 0.92391 | 1.68929 |
| PROC | 1.06277 (376.83) | 0.15172 | 0.73805 | 1.43295 |
| RSI | 1.07011 (388.67) | 0.16987 | 0.53364 | 1.28639 |
| SAR | 0.94362 (415.63) | 0.16925 | 0.69101 | 1.17789 |
| Slow Stochastic | 1.04431 (480.67) | 0.24306 | 0.63703 | 1.45091 |
| Williams %R | 0.99692 (399.42) | 0.35919 | 0.49766 | 1.76001 |



(a)

(b)

Figure 5. Figures 5(a,b) show the latent space of Variational Autoencoder on the time series' of expert wealth's implemented using Keras in Python. In figure 5(a) experts are coloured by which of the 4 object clusters they trade whereas in figure 5(b), experts are coloured by their underlying trading strategy, $\boldsymbol{\omega}(i)$.

original data, given the latent variables, as closely as possible (Kingma and Welling 2019). We utilise the VAE for non-linear dimensionality reduction purposes and the illustration shows the latent representation learned by the model. 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.

To provide some sort of comparison, in figure 5(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 sophisticated/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 (2018).

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 Jarrow *et al.* (2012), we choose a period of 400 days to test our strategy. We test the realised profits and

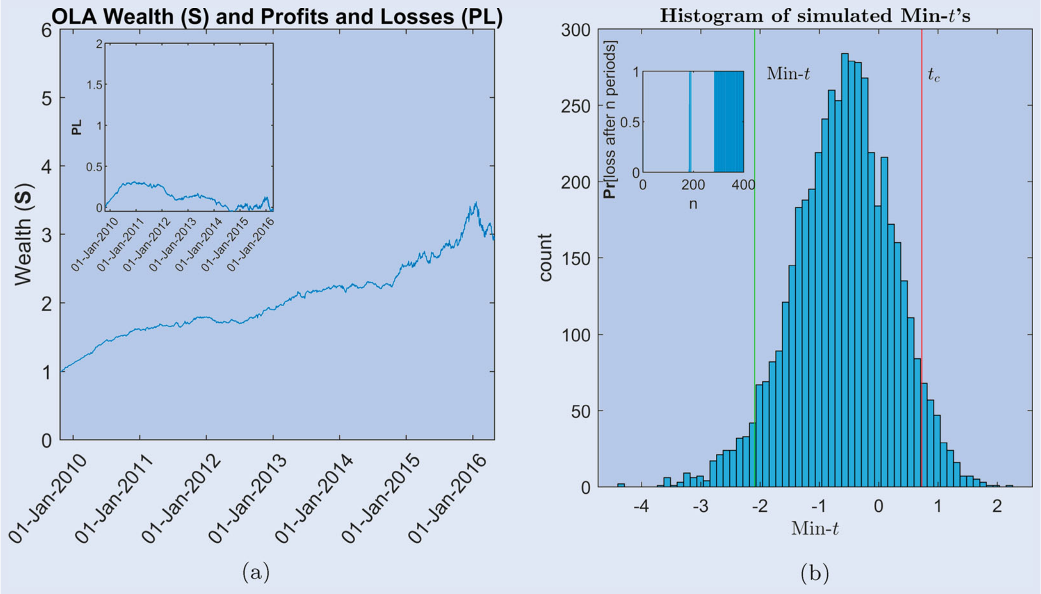


Figure 6. On the left, figure 6(a), the overall cumulative portfolio wealth ($S$) for daily data with transaction costs. Figure 6(a) inset illustrates the profits and losses (**PL**) for overall strategy for daily data with transaction costs. On the right, figure 6(b) the histogram of the 5000 simulated Min-$t$ statistics resulting from the CM model and the incremental process given in equation (11) along with the Min-$t$ statistic (green) for the overall strategy's profit and loss sequence over the 400 day period stretching from the 30th trading day until the 430th trading day with transactions costs incorporated. Also illustrated is the critical value at the 5% significance level (red). Figure 6(b) inset shows the probability of the overall trading strategy generating a loss for each of the first 400 trading days. (a) Daily algorithm performance with costs and (b) Daily statistical arbitrage test with costs.

losses for the 400-day period stretching from the 30th trading day until the 430th 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 3.1 step 3.1 using simulations of the profit process in equation (11), figure 1(b) illustrates the histogram of Min-$t$ values. The critical value $t_c$ is then computed as the 0.95-quantile of the simulated distribution which refers to a significance level of $\alpha = 5\%$ and is illustrated by the red vertical line. The estimated CM model parameters for $\mu$, $\lambda$ and $\sigma^2$ are 2.234, 6.142 and 0.00024, respectively. 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 2.234 (vertical green line). By equation (16), 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 equation (16) adjusted for the case of the CM model. As discussed in section 3.1 step 3.1, 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 equation (16) 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 1(b) illustrates the probability of loss for each of the first 25 trading days, where we compute the probability of loss of the profit and loss process from the first trading period up until the $n$th period for each $n = 1, \ldots, 20$. As is evident from the figure inset, it takes roughly 9 periods for the probability of loss to converge below 5%.

**5.1.2. Transaction costs.** In this section we reproduce the results from above but this time including transaction costs for daily trading as discussed in section 2.5. Once direct and indirect (equation (8)) costs have been computed, the idea is to subtract off the transaction cost from the profit and losses of each day and compound the resulting value onto $S_{t-1}$ to get the wealth for period $t$. These daily profit and losses are added to get the cumulative profit and loss **PL**.

It is clear from the inset of figure 6(a), which illustrates the profits and losses (**PL**) of the overall strategy less the 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 95th percentile of the histogram as illustrated in figure 6(b). In addition to this, although the probability of loss of the strategy with transaction costs included initially converges to zero, it eventually settles on a value of one. This is illustrated in the inset of figure 6(b). The estimated parameters for the CM model for $\mu$, $\lambda$ and $\sigma^2$ are -2.09, 7.04 and 0.00035, respectively.

Considering the above evidence contained in figures 6(a,b) and its 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. This is in agreement with Schulmeister (2009) in that there is a strong possibility that stock price and volume trends have shifted to higher frequencies than the daily time scale, as a resultant the trading strategies' profits have, over time, diminished on such time scales.

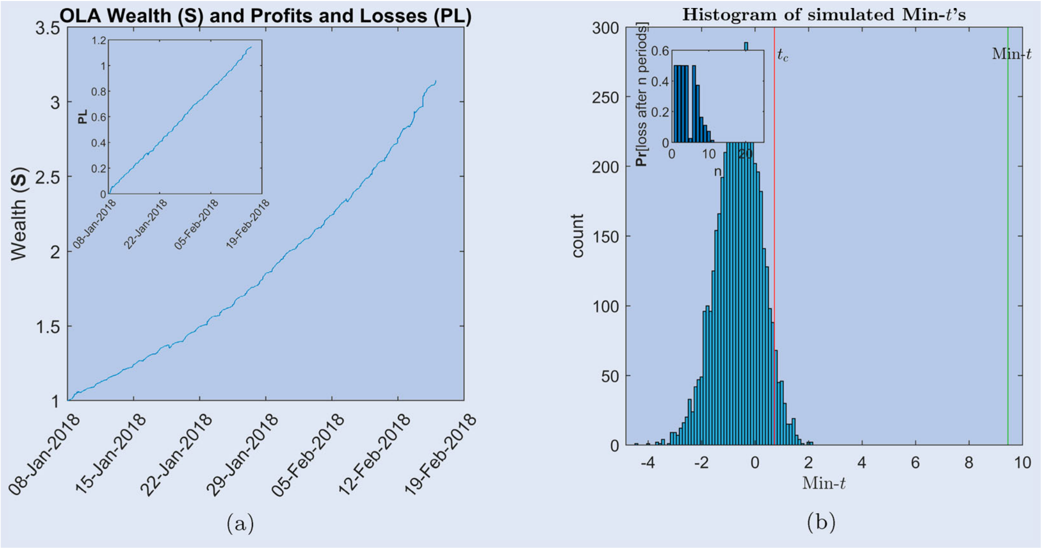


Figure 7. On the left, figure 7(a), the overall cumulative portfolio wealth (**S**) for intraday-daily data with no transaction costs. Figure 7(a) inset illustrates the associated profits and losses. On the right, figure 7(b), the histogram of the 5000 simulated Min-$t$ statistics resulting from the CM model and the incremental process given in equation (11) 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). Figure 7(b) inset shows the probability of the overall trading strategy generating a loss after $n$ periods for each $n = 5, \ldots, 30$ of the intraday-daily profit and loss process (**PL**) taken from the 5th time bar of the second day when active trading commences. (a) Intraday-daily performance without costs and (b) Intraday-daily stat. arb. test without costs.

Table 2. Group summary statistics of the terminal wealth of the experts grouped by their underlying strategy ($\boldsymbol{\omega}(i)$ where $i = 1, \ldots, 17$) for intraday-daily trading. In brackets next to the mean terminal wealth, we rank all of the experts in terms of their terminal wealth and then take the mean rank of the experts for each given strategy group.

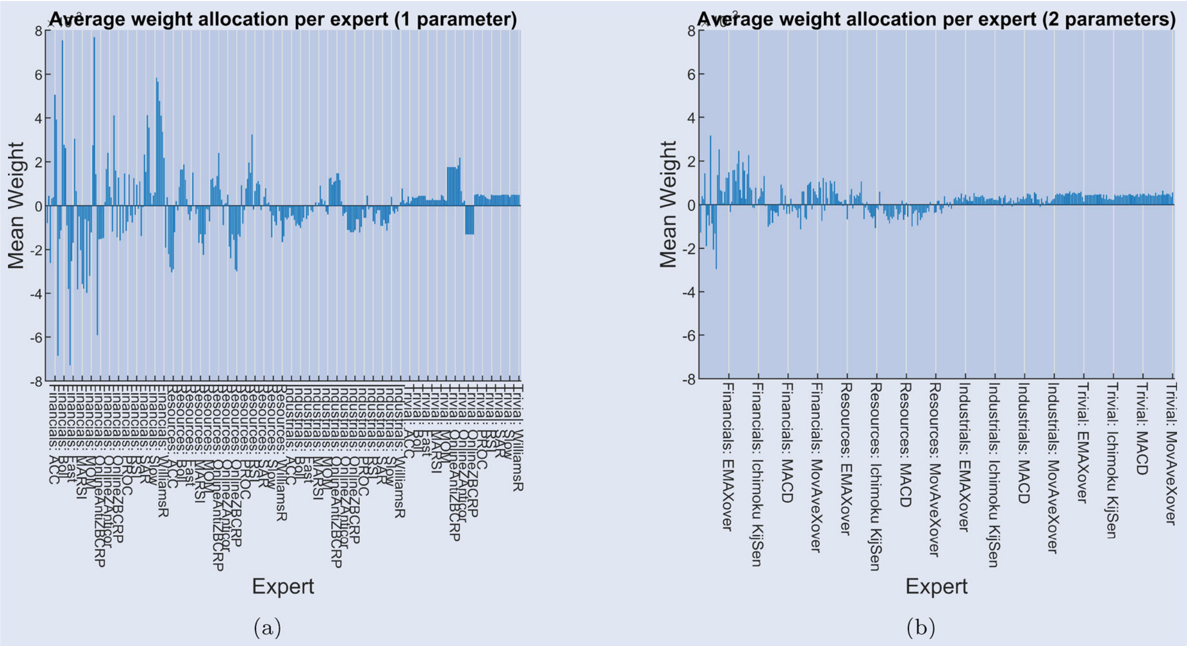
| Strategy | Mean (mean rank) | St. Dev. | Min | Max |
|---|---|---|---|---|
| EMA X-over | 0.99996 (287.08) | 0.00468 | 0.9936 | 1.01059 |
| Ichimoku Kijun Sen | 0.99889 (236.75) | 0.00454 | 0.99093 | 1.01057 |
| MACD | 1.00032 (434.18) | 0.00345 | 0.99404 | 1.00776 |
| Moving Ave X-over | 1.00159 (234.67) | 0.00754 | 0.99199 | 1.01996 |
| ACC | 0.00145 (329.39) | 0.00516 | 0.99641 | 1.01953 |
| BOLL | 0.99958 (382.26) | 0.00408 | 0.99242 | 1.00910 |
| Fast Stochastic | 0.99971 (284.42) | 0.00340 | 0.99558 | 1.00988 |
| MARSI | 0.99899 (341.29) | 0.00449 | 0.99027 | 1.01139 |
| MOM | 0.99980 (385.43) | 0.00350 | 0.99404 | 1.00738 |
| Online Anti-Z-BCRP | 0.99915 (271.13) | 0.00516 | 0.98510 | 1.00554 |
| Online Z-Anticor | 0.99993 (233.46) | 0.00411 | 0.98912 | 1.00702 |
| Online Z-BCRP | 0.99963 (199.38) | 0.00322 | 0.99510 | 1.00750 |
| PROC | 1.00000 (252.83) | 0.00466 | 0.99379 | 1.00653 |
| RSI | 1.00193 (302.00) | 0.00598 | 0.99264 | 1.01403 |
| SAR | 0.99826 (277.83) | 0.00264 | 0.99518 | 1.00517 |
| Slow Stochastic | 1.00013 (239.04) | 0.00426 | 0.99362 | 1.00792 |
| Williams %R | 1.00155 (242.21) | 0.00665 | 0.99237 | 1.01254 |

### *5.2. Intraday-Daily data*

Below we report the results of the algorithm implementation for a combination of intraday and daily JSE data as discussed in section 2.3. We run the algorithm on the OHLCV data of 15 most liquid stocks from a set of 30 of the JSE Top 40. Liquidity is calculated in terms of average daily trade volume for the first 4 days of the period 08-01-2018 to 16-02-2018. 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.

**5.2.1. No transaction costs.** Without transaction costs, the cumulative wealth achieved by the overall strategy, illustrated in figure 7(a) evolves similarly to an exponential function over time. The associated profits and losses are displayed in the figure inset of figure 7(a). 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(a)).

Table 2 is the intraday-daily analogue of table 1. In this case, the moving average crossover strategy (Moving



(a) (b)

Figure 8. Figure 8(a,b) illustrate the mean expert mixture, per expert over time, for the intraday-daily implementation, for experts with a single lookback and two lookback parameters, respectively. The labels on the x-axis are of the format ‘object cluster (**c**)’: ‘strategy ($\boldsymbol{\omega}$’).

Ave X-over) produces the expert with the greatest wealth and Obline Anti-Z-BCRP the expert with the least terminal wealth. The moving average crossover strategy also produces experts with the highest variation in terminal wealth’s. Momentum (MOM) provides the best mean ranking of experts among all other strategies, however, RSI 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 6th time bar of the 2nd trading day† using the intraday-daily profit and loss sequence (**PL**). Figure 7(b) 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 0.7235 and the Min-$t$ value is 9.4528. Thus, there is strong evidence to reject the null hypothesis of no statistical arbitrage as the resulting p-value is identical to zero. The estimated parameters for the CM model, for intraday-daily trading without costs, for $\mu$, $\lambda$ and $\sigma^2$ are 9.45, 9.63 and 0.000095, respectively.

The figure inset of figure 7(b) illustrates the probability of loss for each of the first 30 periods of the 400 periods as discussed in the above paragraph. It takes roughly an hour (13 periods) for the probability of loss to converge to zero.

Figure 8(a,b) (intraday-daily analogues of figure 3(a,b)), illustrate the mean expert mixture ($q$), per expert over time, for experts with a single lookback and two lookback parameters, respectively. An interesting observation is that it is apparent that the ’Financials’ object cluster is quite heavily favoured in the intraday-daily case, as can be seen on the left hand side of figure 8(a,b). The daily implementation’s allocations appear to be more uniformly distributed among the experts compared to the intraday-daily case, especially for the single parameter experts. Figure 9 illustrates the mean expert mixture value ($q$) across all experts for a given strategy, $\boldsymbol{\omega}$, over time. It is difficult to determine which, if any single strategy is consistently favoured; the inset figure 9 zooms into two days from 1 Feb to end of 2 Feb 2018, to better illustrate that the ensemble algorithm is not just randomly switching between strategies, but is able to learn some underlying dynamics on the intraday scale. In a similar manner to figure 4 (daily data case), the algorithm takes a few periods at the beginning of the day to learn the underlying dynamics before picking the winners and losers, in a more consistent manner.

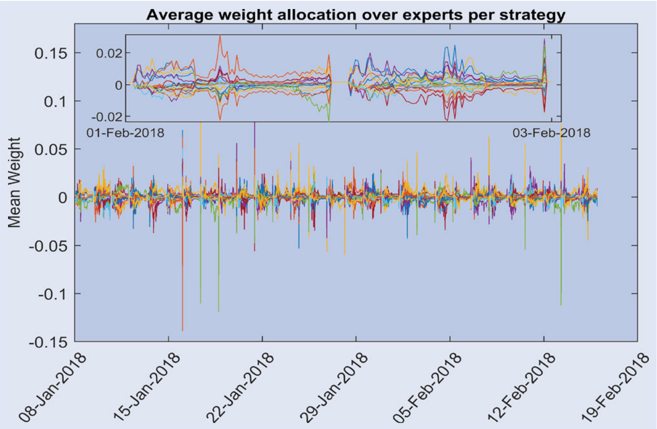


Figure 9. The mean expert mixture value ($q$) across all experts for a given strategy, $\boldsymbol{\omega}$, over time for the intraday-daily implementation. The figure inset zooms into two days from 1 Feb to end of 2 Feb 2018, to provide a better illustration of the intraday dynamics of the learning algorithm.

† This corresponds to the trading period within which the very first trading decisions are made.

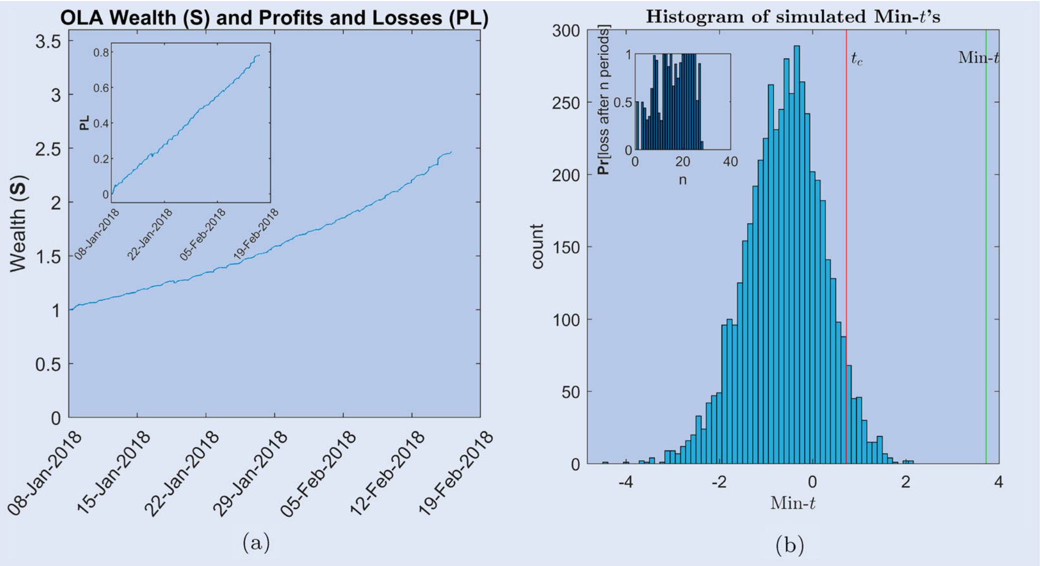


Figure 10. On the left, figure 10(a), the overall cumulative portfolio wealth (**S**) for intraday-daily data with transaction costs and the inset illustrates the associated profits and losses. On the right, figure 10(b), the histogram of the 5000 simulated Min-$t$ statistics resulting from the CM model and the incremental process given in equation (11) 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). Figure 10(b) inset shows the probability of the overall trading strategy generating a loss after $n$ periods for each $n = 5, \ldots, 40$ of the intraday-daily profit and loss process (**PL**) taken from the 5th time bar of the second day when active trading commences. (a) Intraday-daily performance with costs and (b) Intraday-daily statistical arbitrage test with costs.

**5.2.2. Transaction costs.** We now report the results of the algorithm run on the same intraday-daily data as in the subsection above but this time with transaction costs incorporated (see section 2.5). Figure 10(a) 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 illustrated in the case of no transaction costs (figure 10(a) 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 6(a)). Furthermore, figure 10(b) provides significant evidence to reject the no statistical arbitrage null hypothesis† and returns a Min-$t$ statistic of 3.72 in the transaction costs case, compared to 9.45 in the case of no transaction costs (figure 7(b)). 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 (roughly 2 hours or 31 trading periods) than the case of no transaction costs (roughly 1 hour or 13 trading periods, as illustrated in the inset of figure 7(b)). 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 (2009) 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 (Schulmeister 2009)). 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.

## 6. 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 suggest that the proposed algorithm is 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 aggressive view is taken on intraday trading costs. We are also able to show, for both the daily and intraday-daily data implementations, that the probability of loss declines below 5% relatively quickly; this strongly suggests that the

† The estimated parameters for the CM model, for intraday-daily trading with costs, for $\mu$, $\lambda$ and $\sigma^2$ are 3.72, 4.08 and 0.000012, respectively.

algorithm is well suited for a trader whose preference or requirement is to make adequate returns in the short-run. It may well be that the statistical arbitrages we have identified intraday are artefacts' from 'price distorters' (Moffitt 2017) rather than legitimate mispricing in the sense of majority views relative to a trading minority, and hence cannot be easily traded out for profit. This suggests that it can be important to try unpack the difference between the structural mispricing's relative to statistical arbitrages—this is outside of the scope of the current work and cannot be determined using the tests implemented in this work.

The superior performance of the algorithm for intraday trading is in agreement with Schulmeister (2009), who concluded that while the 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, trading the same strategies on 30-minute (intraday) data between 1983 and 2007 has produced decent average gross returns. However, such returns have slowly declined since the early 2000's. In conclusion, the proposed algorithm is much better suited to trading at higher frequencies; but we are also aware that over time, trading strategies that are not structural in nature, are slowly arbitraged away through over-crowding.

We are also cognisant of the fact that intraday trading will require a large component of accumulated trading profits to finance frictions, concretely to fund direct, indirect and business model costs (Loonat and Gebbie 2018). 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 for profit. 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, 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.

Strategy pre-selection can restrict the algorithm to learn sub-optimal classes of strategies relative to the larger theoretical pool of all possible strategies. It could be useful to understand the extent to which the *a-priori* reduced choice of technical indicators could be learnt by the algorithm; this may provide additional insight into how representative the current (albeit domain informed, but arbitrary) pool of rules and indicators are. This does suggest an alternative approach could be to shift the focus onto the constituent experts themselves e.g. to model the population of trading experts as competing in a predator-prey environment (Farmer 2000, Johnson *et al.* 2013) because this may better capture the interactions between various experts using various trading rules, or even more generic adaptive or non-linear learning methods, such as reinforcement learning.

However, the central premise here is that simple mechanistic rules combined with meta-learning is sufficient to generate candidate statistical arbitrages in aggregate. The nature of potential expert learning is an important point to consider because the question of whether one can quantify the extent to which adding learning algorithms at the expert level, rather than merely using deterministic rules, would add any value to the overall aggregate result remains unanswered. Expert level learning combined with meta-learning at the aggregate level may better capture our preconceptions about the strategic nature of real market participants - however, preliminary results have suggested a limited performance gain. This is an important future direction of research, and may in turn lead to insights into why zero-intelligent agent representations can be effective for modelling in financial markets under aggregation and meta-learning in the presence of strategic agents.

## Acknowledgments

NM and TG would like to thank the reviewers for extensive feed-back and suggestions. We would like to thank the Statistical Finance Research Group in the Department of Statistical Sciences for various useful discussions relating to the work. In particular, we would like to thank Etienne Pienaar, Lionel Yelibi and Duncan Saffy. We also extend our thanks to Michael Gant for his help with developing some of the strategies and valuable discussions with regards to the statistical arbitrage test.

## Disclosure statement

No potential conflict of interest was reported by the authors.

## Supplemental data

Supplemental data for this article can be accessed at http://dx.doi.org/10.1080/14697688.2020.1869292.

## ORCID

*N. J. Murphy* http://orcid.org/0000-0001-7966-7446
*T. J. Gebbie* http://orcid.org/0000-0002-4061-2621

## References

Algoet, P.H. and Cover, T.M., Asymptotic optimality and asymptotic equipartition properties of logoptimum investment. *Ann. Probab.*, 1988, **16**, 876–898.

Andersen, T., Bollerslev, T., Diebold, F.X. and Labys, P., The distribution of realized exchange rate volatility. *J. Am. Stat. Assoc.*, 2001, **96**, 42–55.

Aronson, D.R., *Evidence-Based Technical Analysis: Applying the Scientific Method and Statistical Inference to Trading Signals*, 2007 (John Wiley & Sons Inc: Hoboken, NJ).

Auer, P., Cesa-Bianchi, N., Freund, Y. and Schapire, R.E., The nonstochastic multi-armed bandit problem. *SIAM J. Comput.*, 2002, **32**, 48–77.

Bailey, D.H. and López de Prado, M.M., The deflated sharpe ratio: Correcting for selection bias, backtest overfitting and non-normality. *J. Portfolio Manage.*, 2014, **40**, 94–107.

Bailey, D.H., Borwein, J.M., de Prado, M.L. and Zhu, Q.J., Pseudo-mathematics and financial charlatanism: The effects of backtest overfitting on out-of-Sample performance. *Notices AMS*, 2014, **61**, 458–471.

Bailey, D.H., Borwein, J.M., de Prado, M.L. and Zhu, Q.J., The probability of backtest overfitting. *J. Comput. Finance*, 2016, **20**, 39–69.

Ban, G-Y., Karoui, N.E. and Lim, A.E., Machine learning and portfolio optimization. *Manage. Sci.*, 2018, **64**, 1136–1154.

Black, F., Noise. *J. Finance*, 1986, **41**, 529–543.

Bouchaud, J.-P., Gefen, Y., Potters, M. and Wyart, M., Fluctuations and response in financial markets: The subtle nature of 'Random' price changes, 2004. Available at SSRN: https://ssrn.com/abstract = 507322.

Bouchaud, J.-P., Ciliberti, S., Lemperiere, Y, Majewski, A., Seager, P. and Ronia, K.S., Black was right: Price is within a factor 2 of value, 2017. Available at SSRN: https://ssrn.com/abstract = 3070850,

Borodin, A., El-Yaniv, R. and Gogan, V., Can we learn to beat the best stock. *J. Art. Intelligence Res.*, 2004, **21**, 579–594.

Buffington, J. and Elliot, R.J., American options with regime switching. *Int. J. Theor. Appl. Finance*, 2002, **5**, 497–514.

Cover, T.M., Universal portfolios. *Math. Finance*, 1991, **1**, 1–29.

Cover, T.M. and Ordentlich, E., Universal portfolios with side information. *IEEE Trans. Inform. Theory*, 1996, **42**, 348–363.

Clayburg, J.F., *Four Steps to Trading Success: Using Everyday Indicators to Achieve Extraordinary Profits*, 2002 (John Wiley & Sons: New York).

Chan, E., *Quantitative Trading: How to Build Your Own Algorithmic Trading Business*, 2009 (Wiley: Hoboken, NJ).

Chande, T.S. and Kroll, S., *The New Technical Trader: Boost Your Profit by Plugging Into the Latest Indicators*, 1994 (Wiley: New York).

Creamer, G. and Freund, Y., Automated trading with boosting and expert weighting. *Quant. Finance*, 2010, **10**, 401–420.

De Wit, J.-J.D., Statistical arbitrage in south africa. Minor Dissertation Submitted to the Faculty of Commerce in Partial Fulfilment of the Requirements for the Master's degree in Mathematical Finance at the University of Cape Town, 2013.

Dochow, R., *Online Algorithms for the Portfolio Selection Problem*, 2016 (Springer: Wiesbaden).

etfSA, STeFI-linked Money Market Deposit, Nedbank Beta Solutions, 2011. Available online at: https://www.fundsdata.co.za/etf_news/11_04_04_etfsa_stefi_explan_doc.pdf (accessed 15 March 2019).

Guilbaud, F. and Pham, H., Optimal high-frequency trading with limit and market orders. *Quant. Finance*, 2013, **13**(1), 79–94.

Fama, E.F., Efficient capital markets: A review of theory and empirical work. *J. Finance*, 1970, **25**, 383–417.

Farmer, J.D., A simple model for the non-equilibrium dynamics and evolution of a financial market. *Int. J. Theor. Appl. Finance*, 2000, **3**, 425–441.

Fayek, M.B., El-Boghdadi, H.M. and Omran, S.M., Multi-objective optimization of technical stock market indicators using GAs. *Int. J. Comput. Appl.*, 2013, **68**, 41–48.

Guo, X., An explicit solution to an optimal stopping problem with regime switching. *J. Appl. Probab.*, 2001, **38**(2), 464–481.

Györfi, L., Udina, F. and Walk, H., Nonparametric nearest neighbor based empirical portfolio selection strategies. *Statistics and Decisions*, 2008, **26**, 145–157.

Györfi, L., Ottucsák, G. and Walk, H., *Machine Learning for Financial Engineering*, 2012 (Imperial College Press: London).

Gatheral, J., No-dynamic arbitrage and market impact. *Quant. Finance*, 2010, **10**, 749–759.

Gebbie, T., MATLAB Technical Analysis toolbox ("technical"), 2010. https://github.com/timgebbie/MATLAB-quant-finance/technical, http://dx.doi.org/10.25375/uct.11302403.

Hogan, S., Jarrow, R., Teo, M. and Warachka, M., Testing market efficiency using statistical arbitrage with applications to momentum and value strategies. *J. Financ. Econ.*, 2004, **73**, 525–565.

Harvey, M., Hendricks, D., Gebbie, T. and Wilcox, D., Deviations in expected price impact for small transaction volumes under fee restructuring. *Phys. A: Stat. Mech. Appl.*, 2017, **471**, 416–426.

Hendricks, D., Using real-time cluster configurations of streaming asynchronous features as online state descriptors in financial markets. *Pattern. Recognit. Lett.*, 2017, **97**, 21–28.

Hendricks, D. and Wilcox, D., A reinforcement learning extension to the Almgren-Chriss framework for optimal trade execution. 2014 IEEE Conference on Computational Intelligence for Financial Engineering and Economics (CIFEr), IEEE, pp. 457–464, 2014. doi:10.1109/CIFEr.2014.6924109.

Hendricks, D., Gebbie, T. and Wilcox, D., Detecting intraday financial market states using temporal clustering. *Quant. Finance*, 2016, **16**(11), 1657–1678.

Hou, K., Xue, C. and Zhang, L., Replicating Anomalies. NBER Working Paper Series, 2017, No. 23394, 2017.

Huberman, G. and Stanzl, W., Optimal liquidity trading. *Springer: Rev. Finance*, 2005, **9**, 165–200.

Ichimoku 101, IchimokuTrade.com, https://www.ichimokutrade.com/articles/Ichimoku_Ebook.pdf (accessed 28 July 2018).

Jarrow, R., Teob, M., Tse, Y.-K. and Warachka, M., An improved test for statistical arbitrage. *J. Financ. Markets*, 2012, **15**, 47–80.

Johnson, N., Zhao, G., Hunsader, E., Qi, H., Meng, J. and Tivnan, B., Abrupt rise of new machine ecology beyond human response time. *Sci. Rep.*, 2013, **3**.

Kalai, A. and Vempala, A., Effcient algorithms for universal portfolios. *J. Mach. Learn. Res.*, 2002, **3**, 423–440.

Kelly, J.L., A new interpretation of information rate. *Bell. Syst. Tech.*, 1956, **35**, 917–926.

Kestner, L.N., *Quantitative Trading Strategies: Harnessing the Power of Quantitative Techniques to Create a Winning Trading Program*, 2003 (McGraw-Hill: New York).

Kingma, D.P. and Welling, M., An introduction to variational autoencoders, 2019. Available online at: https://arxiv.org/abs/1906.02691 (accessed 12 June 2020).

Korn, R., Portfolio optimisation with strictly positive transaction costs and impulse control. *Finance Stoch.*, 1998, **2**(2), 85–114.

Lee, W., *Theory and Methodology of Tactical Asset Allocation*, 2000 (Wiley & Sons: New Jersey).

Li, B. and Hoi, S.C., Online portfolio selection: A survey. *ACM Comput. Surveys (CSUR)*, 2014, **46**(35).

Li, B. and Hoi, S.C., *Online Portfolio Selection: Principles and Algorithms*, 2016a (CRC Press: Boca Raton, FL).

Li, B. and Hoi, S.C., OLPS: A toolbox for on-line portfolio selection. *J. Mach. Learn. Res.*, 2016b, **55**, 1–5.

Linton, D., *Cloud Charts: Trading Success with the Ichimoku Technique*, 2010 (Updata: London).

Lo, A., The statistics of Sharpe ratio's. *Financ. Anal. J.*, 2002, **58**, 36–52.

Lo, A.W. and Hasanhodzic, J., *The Heretics of Finance: Conversations with Leading Practitioners of Technical Analysis*, 2009 (Bloomberg Press: New York).

Lo, A., Mamaysky, V. and Wang, J., Foundations of technical analysis: Computational algorithms, statistical inference, and empirical implementation. *J. Finance.*, 2000, **55**, 1705–1765.

Loonat, F. and Gebbie, T., Learning zero-cost portfolio selection with pattern matching. *PLoS. ONE.*, 2018, **13**, e0202788.

Lunga, D. and Marwala, T., Time series analysis using fractal theory and online ensemble classifiers. In *AI 2006: Advances in Artificial Intelligence*, edited by A. Sattar, B. Kang, Lecture Notes in Computer Science, 4304, 2006.

Maringer, D. and Ramtohul, T., Threshold recurrent reinforcement learning model for automated trading. European Conference on the Applications of Evolutionary Computation, Springer, 212–221, 2010.

Maringer, D. and Ramtohul, T., Regime-switching recurrent reinforcement learning for investment decision making. *Comput. Manage. Sci.*, 2012, **9**(1), 89–107.

Moffitt, S., Why markets are inefficient: A gambling theory of financial markets for practitioners and theorists, February 22, 2017.

Murphy, N., Learning technical trading, 2019. Available online at: https://github.com/NJ-Murphy/Learning-Technical-Trading.git.

Murphy, N. and Gebbie, T., JSE equity market transaction data: TRTH and BLP daily and intraday data, 2019, Mendeley Data, v4, http://dx.doi.org/10.17632/v5466fbpf9.4.

Park, C.-H. and Irwin, S., The profitability of technical analysis: A review. *AgMAS Project Research Reports*, 2004, No. 04.

Poon, S.-H., Volatility estimation, 2008. Available online at: https://www.cmegroup.com/trading/fx/files/volEst.pdf (accessed 10 September 2017).

Rechenthin, M.D., Machine-learning classification techniques for the analysis and prediction of high-frequency stock direction. PhD Thesis, University of Iowa, 2014.

Samo, Y.-L.K. and Hendricks, D., What makes an asset useful?, 2018. Available online at: https://arxiv.org/abs/1806.08444 (accessed 01 January 2019).

Schulmeister, S., Profitability of technical stock trading: Has it moved from daily to intraday data?. *Rev. Financ. Econom.*, 2009, **18**, 190–201.

Teixeira, L.A. and de Oliveira, A.L., A method for automatic stock trading combining technical analysis and nearest neighbor classification. *Expert. Syst. Appl.*, 2010, **37**, 6885–6890.

The Definitive Reference to the Ichimoku Kinko Hyo. Available online at: http://www.kumotrader.com/ichimoku_wiki/index.php?title = Main_Page (accessed 10 October 2018).

The Hidden Costs of Trading. Available online at: http://pages.stern.nyu.edu/ ∼ adamodar/New_Home_Page/invemgmt/trading.htm (accessed 20 March 2018).

Uziel, G. and El-Yaniv, R., Online learning of portfolio ensembles with sector exposure regularization, 2016. Available online at: https://arxiv.org/abs/1604.03266 (accessed 14 June 2020).

Wilder, J.W., *New Concepts in Technical Trading Systems*, 1978 (Trend Research).

Yiu, K.-F., Liu, J., Siu, T. and Ching, W.-K., Optimal portfolios with regime switching and value-at-risk constraint. *Automatica*, 2010, **46**(6), 979–989.

Zhang, Q. and Guo, X., Closed-form solutions for perpetual american put options with regime switching. *SIAM. J. Appl. Math.*, 2004, **64**(6), 2034–2049.

Zhou, X., Chen, N., Gao, X. and Xiong, Y., Regime switching bandits. arXiv preprint, arXiv:2001.09390, 2020.

## Appendices

## Appendix 1. Technical Indicators and Trading Rules

We follow Creamer and Freund (2010) and Kestner (2003) 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 (Murphy 2019, Gebbie 2010).

## A. Appendix 2. Trend Following and Contrarian Mean Reversion Strategies

### *A.1. Zero-Cost BCRP*

Zero-cost BCRP is the zero-cost long/short version of the BCRP strategy and is a trend following strategy. The portfolio controls for the zero-cost BCRP strategy are the active bets (Lee 2000):

$$\mathbf{b} = \frac{1}{\gamma}\Sigma^{-1}\left[\mathbb{E}[\mathbf{R}] - \mathbf{1}\frac{\mathbf{1}^\intercal\Sigma^{-1}\mathbb{E}[\mathbf{R}]}{\mathbf{1}^\intercal\Sigma^{-1}\mathbf{1}}\right]. \tag{A1}$$

Here $\Sigma^{-1}$ 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 $\gamma$ is the risk aversion parameter. Equation (A1) is the risky Optimal Tactical Portfolio (OTP) that takes optimal risky bets given the risk aversion $\gamma$. 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 $\ell$ days (short-term look-back parameter).

### *A.2. Zero-cost anti-BCRP*

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

### *A.3. Zero-cost anti-Correlation*

Zero-cost anti-correlation (Z-Anticor) is a contrarian mean reverting algorithm and is an adapted version of the Anticor algorithm developed in Borodin *et al.* (2004). The first step is to extract the price relatives for the two most recent sequential windows each of length $\ell$. Let $\boldsymbol{\mu}_2^\ell$ and $\boldsymbol{\mu}_1^\ell$ denote the average log-returns of the $\ell$ 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:

$$\Sigma^\ell = \frac{1}{\ell-1}[(\boldsymbol{x}_{t-\ell}^{t-2\ell+1} - 1) - \mathbf{1}^\intercal\boldsymbol{\mu}_1^\ell]^\intercal[(\boldsymbol{x}_t^{t-\ell+1} - 1) - \mathbf{1}^\intercal\boldsymbol{\mu}_2^\ell] \tag{A2}$$

Z-Anticor then computes the claim that each pair of stocks have on one another, denoted $\text{claim}_{i\to j}^\ell$, which is the claim of stock $j$ on stock $i$ and is the extent to which we want to shift our allocation from stock $i$ to stock $j$ (Borodin *et al.* 2004). $\text{claim}_{i\to j}^\ell$ exists and is thus non-zero if and only if $\boldsymbol{\mu}_2 > \boldsymbol{\mu}_1$ and $\text{P}_{ij} > 0$. The claim is then calculated as

$$\begin{aligned}\text{claim}_{i\to j}^\ell &= \text{P}_{ij}^\ell + \max(-\text{P}_{ii}^\ell, 0) + \max(-\text{P}_{jj}^\ell, 0) \quad \text{where}\\ \text{P}_{ij}^\ell &= \Sigma_{ij}^\ell/\sqrt{\Sigma_{ij}^\ell\Sigma_{ij}^\ell}\end{aligned} \tag{A3}$$

Finally, we calculate the expert control for the $i$th stock in period $t+1$ from the difference in transfers that take place between stock $i$ to stock $j$, and from stock $j$ to stock $i$, but here modified to ensure the resulting portfolio is long/short :

$$\begin{aligned}\mathbf{h}_{t+1}^n(i) &= \mathbf{h}_t^n(i) + \sum_i[\text{transfer}_{j\to i}^\ell - \text{transfer}_{i\to j}^\ell] \quad \text{where}\\ \text{transfer}_{i\to j}^\ell &= \frac{1}{3}\text{claim}_{i\to j}^\ell\end{aligned} \tag{A4}$$

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

Table A1. Trading indicators used by the trading in table A2.

| Indicator | Description | Calculation |
|---|---|---|
| $\mathrm{SMA}^c_t(n)$ | The *Simple Moving Average* (SMA) is the mean of the closing prices over the last $n$ trading days. The smaller the value of $n$, the closer the moving average will fit to the price data. | $\mathrm{SMA}_t(\mathbf{P}^c, n) = \frac{1}{n}\sum_{i=0}^{n-1} P^c_{t-i}$ |
| $\mathrm{EMA}^c_t(n)$ | The *Exponential Moving Average* (EMA) uses today's close price, yesterday's moving average value and a smoothing factor ($\alpha$). The smoothing factor determines how quickly the exponential moving average responds to current market prices (Kestner 2003). | $\mathrm{EMA}_t(\mathbf{P}^c, n) = \alpha P^c_t + (1 - \alpha)\mathrm{EMA}_{t-1}(P^c_{t-1}, n)$ where $\alpha = \frac{2}{n+1}$ and $\mathrm{EMA}_0(P^c_0, n) = \mathrm{SMA}(\mathbf{P}^c, n)$ |
| HH(n) | The *Highest High* (HH) is the greatest high price in the last $n$ periods and is determined from the vector $\boldsymbol{P^h_n}$ of the high-prices of the last $n$ periods. | Given the high prices of last $n$ periods: $\boldsymbol{P^h_n} = (P^h_{t-n}, P^h_{t-n+1}, P^h_{t-n+2}, \ldots, P^h_t)$ to find: $\mathrm{HH}(n) = \max(\boldsymbol{P^h_n})$. |
| LL(n) | *Lowest Low* (LL) is the smallest low price in the last $n$ periods and is found from the vector $\boldsymbol{P^l_n}$ of low prices in the last $n$ periods. | Givne the low prices of the last $n$ periods: $\boldsymbol{P^l_n} = (P^l_{t-n}, P^l_{t-n+1}, P^l_{t-n+2}, \ldots, P^l_t)$ to find: $\mathrm{LL}(n) = \min(\boldsymbol{P^l_n})$. |
| $\mathrm{IKH}(n_1, n_2, n_3)$ | The *Ichimoku Kinko Hyo* (IKH) (at a glance equilibrium chart) system consists of five lines and the Kumo (cloud).[1] 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. Typical input parameters: $n_1 = 7$, $n_2 = 22$, and $n_3 = 44$. Here we keep $n_1$ fixed at 7 but vary the other two parameters. | Tenkan-sen (Conversion Line): $\mathrm{HH}(n_1) + \mathrm{LL}(n_1))/2$ Kijun-sen (Base Line): $\mathrm{HH}(n_2) + \mathrm{LL}(n_2))/2$ Chikou Span (Lagging Span): Closeplotted $n_2$ daysinthepast Senkou Span A (Leading Span A): $(\mathrm{ConversionLine} + \mathrm{BaseLine})/2$ Senkou Span B (Leading Span B): $\mathrm{HH}(n_3) + \mathrm{LL}(n_3))/2$ Kumo (Cloud): Area between the Leading Span A and the Leading Span B from the Cloud |
| $\mathrm{MOM}_t(n)$ | *Momentum* (MOM) gives the change in the closing price over the past $n$ periods. | $\mathrm{MOM}_t(n) = P^c_t - P^c_{t-n}$ |
| $\mathrm{ACC}_t(n)$ | *Acceleration*(ACC) measures the change in momentum between two consecutive periods $t$ and $t - 1$ | $\mathrm{ACC}_t(n) = \mathrm{MOM}_t(n) - \mathrm{MOM}_{t-1}(n)$ |
| $\mathrm{MACD}_t(n_1, n_2)$ | The *Moving Average Convergence/Divergence* (MACD) oscillator 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. A signal line is computed by taking an EMA of the MACD and determines the instances to buy (over-sold) and sell (over-bought) when used in conjunction with the MACD.[2] | Long-term EWM: $\mathrm{EMA}^L_t = \mathrm{EMA}_t(P^c, n_2)$ Short-term EWM: $\mathrm{EMA}^S_t = \mathrm{EMA}_t(P^c, n_1)$ Moving Average Convergence/Divergence: $\mathrm{MACD}_t(n_1, n_2) = \mathrm{EMA}^S_t - \mathrm{EMA}^L_t$ The "Signal Line" (SL): $\mathrm{SL}_t(n_1, n_2, n_3) = \mathrm{EMA}_t(\mathrm{MACD}_t(n_2, n_1), n_3)$ The "MACD Signal" (MACDS): $\mathrm{MACDS}_t(n_1, n_2, n_3) = \mathrm{MACD}_t(n_1, n_2) - \mathrm{SL}_t(n_1, n_2, n_3)$ |
| $\mathrm{Fast\%K}_t(\mathbf{n})$ and $\mathrm{Fast\%D}_t(n)$ | *Fast Stochastic Oscillator* shows the location of the closing price relative to the high-low range, expressed as a percentage, over a given number of periods as specified by a look-back parameter. | $\mathrm{Fast\%K}_t(n) = \frac{P^c_t - \mathrm{LL}(n)}{\mathrm{HH}(n) - \mathrm{LL}(n)}$ $\mathrm{Fast\%D}_t(n) = \mathrm{SMA}_t(\mathrm{Fast\%K}_t(n), 3)$ |

(*Continued*).

Table A1. Continued.

| Indicator | Description | Calculation |
|---|---|---|
| $\text{Slow\%K}_t(n)$ and $\text{Slow\%D}_t(n)$ | The *Slow Stochastic Oscillator* is very similar to the fast stochastic indicator and is in fact just a moving average of the fast stochastic indicator. | $\text{Slow\%K}_t(n) = \text{SMA}_t(\text{Fast\%K}_t(n), 3)\text{Slow\%D}_t(n) = \text{SMA}_t(\text{Slow\%K}_t(n), 3)$ |
| $\text{RSI}_t(n)$ | *Relative Strength Index* (RSI) compares the periods that stock prices finish up (closing price higher than the previous period) against those periods that stock prices finish down (closing price lower than the previous period).[3] | $\text{RSI}_t(n) = 100 - \frac{100}{1 + \frac{\text{SMA}_t(\boldsymbol{P}_n^{\text{up}}, n_1)}{\text{SMA}_t(\boldsymbol{P}_n^{\text{dwn}}, n_1)}}$<br>where finishing up (down) are:<br>$P_t^{\text{up}} = \begin{cases} P_t^c & \text{if } P_{t-1}^c < P_t^c \\ \text{NaN} & \text{otherwise} \end{cases}$<br>$P_t^{\text{down}} = \begin{cases} P_t^c & \text{if } P_{t-1}^c > P_t^c \\ \text{NaN} & \text{otherwise} \end{cases}$<br>to find the vector of up and down finishing cases:<br>$\boldsymbol{P}_n^{\text{up}} = (P_{t-n}^{\text{up}}, P_{t-n+1}^{\text{up}}, \ldots, P_t^{\text{up}})\boldsymbol{P}_n^{\text{down}} = (P_{t-n}^{\text{down}}, P_{t-n+1}^{\text{down}}, \ldots, P_t^{\text{down}})$ |
| $\text{MARSI}_t(n_1, n_2)$ | *Moving Average Relative Strength Index* (MARSI) is an indicator that smooths out the action of RSI indicator.[4] MARSI is calculated by simply taking an n2-period SMA of the RSI indicator. | $\text{MARSI}_t(n_1, n_2) = \text{SMA}(\text{RSI}_t(n_1), n_2)$ |
| $\text{Boll}_t^m(n)$ | *Bollinger* (Boll) bands uses a SMA ($\text{Boll}_t^m(n)$) as it's reference point (known as the median band) with regards to the upper and lower Bollinger bands denoted by $\text{Boll}_t^u(n)\text{Boll}_t^d(n)$ respectively and are calculated as functions of standard deviations ($s$). | Median band:<br>$\text{Boll}_t^m(n) = \text{SMA}_t^c(n)$<br>Upper band:<br>$\text{Boll}_t^u(n) = \text{Boll}_t^m(n) + \text{S}\sigma_t^2(n)$<br>Lower band:<br>$\text{Boll}_t^d(n) = \text{Boll}_t^m(n) + \text{S}\sigma_t^2(n)$<br>Here $s$ is chosen to be 2. |
| $\text{PROC}_t(n)$ | The rate of change of the time series of closing prices $P_t^c$ over the last $n$ periods expressed as a percentage. | $\text{PROC}_t(n) = 100 \cdot \frac{P_t^c - P_{t-n}^c}{P_{t-n}^c}$ |
| $\text{Will}_t(n)$ | *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. | $\text{Will}_t(n) = 100 \cdot \frac{\text{HH}(n) - P_t^c}{\text{HH}(n) - \text{LL}(n)} \cdot (-100)$ |
| $\text{SAR}(n)$ | *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 (Wilder 1978). 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 (downtrend). 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. | Calculating the SAR indicator:<br>(i) Initialise: Set initial trend to 1 (up-trend), EP to zero, $AF_0$ to 0.02, $\text{SAR}_0$ to the closing price at time zero $P_0^c$, Last High (LH) to high price at time zero ($P_0^h$) and Last Low (LL) to the low price at time zero ($P_0^l$))<br>(ii) Level Update: update EP, LH, LL and AF based on the current high in relation to the LH (up-trend), or where the current low is in relation to the LL (down-trend)<br>(iii) Time Update: update time $t + 1$ SAR value, $\text{SAR}_{t+1}$, using the equation for the Parabolic SAR for time $t + 1$ as calculated using the previous value at time $t$:<br>$\text{SAR}_{t+1} = \text{SAR}_t + \alpha(\text{EP} - \text{SAR}_t)$<br>(iv) Trend Update: modify the $\text{SAR}_{t+1}$ value, AF, EP, LL, LH and the trend based on the trend and it's value in relation to the current low $P_t^l$ and current high $P_0^h$<br>(v) Iterate: go to next time period and return to step 2 |

[1] See Linton 2010, The Definitive Reference to the Ichimoku Kinko Hyo n.d., Ichimoku 101 n.d.
[2] See Rechenthin 2014
[3] See Creamer and Freund 2010
[3] See Fayek *et al*. 2013

Table A2. The trading rules implemented in the study.

| Trading Rule | Decision | Condition |
|---|---|---|
| Moving Average Crossover | *Buy* | $SMA_{t-1}(n_1) < SMA_{t-1}(n_2)$ & $SMA_t(n_1) >= SMA_t(n_2)$<br>where $n_1 < n_2$ |
| | *Sell* | $SMA_{t-1}(n_1) > SMA_{t-1}(n_2)$ & $SMA_t(n_1) <= SMA_t(n_2)$<br>where $n_1 < n_2$ |
| | *Hold* | otherwise |
| Exponential Moving Average Crossover | *Buy* | $EMA_{t-1}(n_1) < EMA_{t-1}(n_2)$ & $EMA_t(n_1) >= EMA_t(n_2)$<br>where $n_1 < n_2$ |
| | *Sell* | $EMA_{t-1}(n_1) > EMA_{t-1}(n_2)$ & $EMA_t(n_1) <= EMA_t(n_2)$ |
| | *Hold* | otherwise |
| Ichimoku Kijun Sen Cross | *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 |
| Momentum | *Buy* | $\mathrm{MOM}_{t-1}(n) \leq \mathrm{EMA}_t(\mathrm{MOM}_t(n), \lambda)$ & $\mathrm{MOM}_t(n) > \mathrm{EMA}_t(\mathrm{MOM}_t(n), \lambda)$ |
| | *Sell* | $\mathrm{MOM}_{t-1}(n) \geq \mathrm{EMA}_t(\mathrm{MOM}_t(n), \lambda)$ & $\mathrm{MOM}_t(n) < \mathrm{EMA}_t(\mathrm{MOM}_t(n), \lambda)$ |
| | *Hold* | otherwise |
| Acceleration | *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 |
| MACD | *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 |
| Fast Stochastics | *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 |
| Slow Stochastics | *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 |
| RSI | *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 |
| MARSI | *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 |
| Bollinger | *Buy* | $P^c_{t-1} \geq \mathrm{Boll}^d_t(n)$ & $P^c_t \geq \mathrm{Boll}^u_t(n)$ |
| | *Sell* | $P^c_{t-1} \leq \mathrm{Boll}^d_t(n)$ & $P^c_t > \mathrm{Boll}^u_t(n)$ |
| | *Hold* | otherwise |
| PROC | *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 |
| Williams %R | *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 |
| SAR | *Buy* | $\mathrm{SAR}_{t-1} \geq P^c_{t-1}$ & $\mathrm{SAR}_t < P^c_t$ |
| | *Sell* | $\mathrm{SAR}_{t-1} \leq P^c_{t-1}$ & $\mathrm{SAR}_t > P^c_t$ |
| | *Hold* | otherwise |