Efficient computation of mean reverting portfolios using cyclical coordinate descent

T. Griveau-Billion & B. Calderhead

Published online: 08 Sep 2020.

Submit your article to this journal ↗

View related articles ↗

View Crossmark data ↗

Routledge
Taylor & Francis Group

Check for updates

Efficient computation of mean reverting portfolios using cyclical coordinate descent

T. GRIVEAU-BILLION ⬤* and B. CALDERHEAD

Department of Mathematics, Imperial College London, 180 Queen's Gate, Kensington, London SW7 2AZ, UK

The econometric challenge of finding sparse mean reverting portfolios based on a subset of a large number of assets is well known. Many current state-of-the-art approaches fall into the field of co-integration theory, where the problem is phrased in terms of an eigenvector problem with sparsity constraint. Although a number of approximate solutions have been proposed to solve this NP-hard problem, all are based on relatively simple models and are limited in their scalability. In this paper, we leverage information obtained from a heterogeneous simultaneous graphical dynamic linear model (H-SGDLM) and propose a novel formulation of the mean reversion problem, which is phrased in terms of a quasi-convex minimisation with a normalisation constraint. This new formulation allows us to employ a cyclical coordinate descent algorithm for efficiently computing an exact sparse solution, even in a large universe of assets, while the use of H-SGDLM data allows us to easily control the required level of sparsity. We demonstrate the flexibility, speed and scalability of the proposed approach on S&P500, FX and ETF futures data.

Keywords: Bayesian modelling; Sparse portfolio; Mean reversion; Cyclical coordinate descent

JEL Classifications: C11, C32, G11

1. Introduction

Finding a linear combination of stocks that mean revert is an econometric problem that has been captivating researchers for years, since such portfolios with reasonably predictable short term behaviour can be used for developing systematic trading strategies. When applied to two assets, this is often referred to as looking for trading pairs. Such combinations however may be hard to find due to non-stationarity and combinatorial considerations, especially for baskets of more than two assets. Furthermore, finding a mean reverting portfolio is not sufficient in itself; indeed, to be viable the effect must persist over a long enough time horizon and be strong enough to overcome trading costs.

Currently, the classic approach to this problem relies on cointegration theory (see Johansen 2005 for an overview), which tries to find a linear combination of non-stationary processes that is stationary. These methods usually rely on statistical tests to check that the time series are cointegrated. However, as stated in Johansen (2005) such tests strongly depend on the modelling assumptions of the data. In this paper, we are interested in the case of choosing a sparse subset of more than two assets from a much larger universe of assets. Recent advances in this area have proposed a new formulation of the problem (d'Aspremont 2010, Cuturi and d'Aspremont 2013, 2015), which relies on a characterisation of portfolio predictability (Box and Tiao 1977). With this definition, a mean reverting portfolio (MRP) is characterised by a low predictability, in contrast to a momentum portfolio which has high predictability. In these three papers, the authors reformulated the MRP definition as an optimisation problem with constraints and compared it to other methods, such as minimising the portmanteau test or minimising the crossing statistic. They emphasised the importance of a good MRP to be sparse and argued in favour of an additional minimum variance constraint to select a portfolio with a variance high enough for the effect to be tradable.

The approach of d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) relies on the hypothesis of Box and Tiao (1977), which requires a model for the price process s_t conditional on all previous information. The authors assumed that the assets follow a vector autoregressive model, $VAR(p)$, of order p and that the covariance matrix of the portfolio be stationary. In d'Aspremont (2010), and subsequently Sipos and Levendovszky (2013) and Fogarasi and Levendovszky (2013), they also assumed that the resulting

*Corresponding author.
Email: theophile.griveau-billion14@imperial.ac.uk

portfolio follows an Orstein-Uhlenbeck process and made the parallel between the mean reversion parameter and the predictability variable. In order to solve this optimisation, Box and Tiao (1977) rewrote the problem as a generalised eigenvalue problem, to which d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) then added a sparsity constraint, making this combinatorial problem NP-hard, and proposed to solve this minimisation with a semi-definite relaxation approach. As an alternative, Sipos and Levendovszky (2013) and Fogarasi and Levendovszky (2013) proposed a simulated annealing algorithm to find the solution. Both methods produce approximations of the solution, are expensive to compute and lack scalability. Finally, given the resulting MRP, Cuturi and d'Aspremont (2013, 2015) followed a strategy introduced by Jurek and Yang (2007) to trade a stationary autoregressive processes, whereas Sipos and Levendovszky (2013) and Fogarasi and Levendovszky (2013) assumed the portfolio follows an Ornstein-Uhlenbeck process and thus used decision theory to implement their trading strategy.

In this paper, we extend the approach of d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) by characterising a sparse mean reverting portfolio according to its predictability but without assuming a VAR(p) process. Instead we will leverage the results of a heterogeneous simultaneous graphical dynamic linear model (H-SGDLM) to set up the MRP optimisation problem for assets following time varying autoregressive TVAR(p) processes. The H-SGDLM builds a sparse covariance matrix which is not assumed stationary and our new formulation of the MRP minimisation problem with normalisation constraint allows us to use an efficient optimisation to compute the global minimum.

The H-SGDLM of Griveau-Billion and Calderhead (2019), Gruber and West (2016) and Corsi (2004) models the entire investment environment by decomposing each time series, and thus the whole market, into economically-meaningful variables. This decomposition creates different groups of variables representing the endogenous and exogenous influences on the behaviour of each asset. Many different signals are created by this model; for this paper we are mainly interested in the graph of connections between the different assets and the resulting sparse covariance matrix and autoregressive coefficients. These variables will allow us to propose a new formulation of the sparse mean reverting portfolio optimisation problem.

While the previous approaches of d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) and Sipos and Levendovszky (2013), Fogarasi and Levendovszky (2013) obtained approximations of the solution to a constrained optimisation problem, we propose a new algorithm that finds the global minimum. We employ a cyclical coordinate descent (CCD) algorithm which is fast and efficient in large-scale problems. Furthermore, Tseng (2001) proved the convergence to the global minimum of cyclical coordinate descent algorithms when the the utility function can be written as the combination of a convex differentiable function and a non-differentiable but additively separable one.

Building on this result, Yen and Yen (2014) used the CCD algorithm to solve the mean variance portfolio with an elastic-net constraint. In addition, Griveau-Billion *et al.* (2013) used

this CCD method to solve the non-convex equal risk contribution portfolio optimisation problem, where it was observed to be faster than the Newton algorithm in large environments. We follow the direction of these approaches and derive a CCD algorithm to solve the sparse MRP minimisation problem with a normalisation constraint. The algorithm we propose can be applied to compute a MRP from any predicted covariance matrix. However, we focus in this paper on the predictions obtained from the H-SGDLM model.

Using this new MRP solution we can then consider a number of different trading strategies. Hence we will numerically compare the approaches proposed by Jurek and Yang (2007), Sipos and Levendovszky (2013) and Fogarasi and Levendovszky (2013), as well as a combination of the two. In order to assess the performance and robustness of our solution we computed the sparse MRP and implemented the trading strategy on three different datasets: stocks from the S&P500, futures on FXs and futures on ETFs. All of the backtests we obtained exhibit a linear growth throughout the time under study, which includes the financial crisis of 2008. While in some environments the backtests are smoother than others, the fact that none has a particularly large draw-down, and they are all roughly linear through time, highlights the robustness of our proposed approach.

Section 2 will recall the sparse mean reversion optimisation problem and briefly review the current approaches for solving it. Section 3 will then detail the H-SGDLM model and the novel formulation of the MRP problem it allows. The CCD algorithm and its application to solve the new MRP optimisation are detailed in Section 4. Once the optimal solution is found we consider possible trading strategies, as detailed in Section 5. Finally, in Section 6 we present results for different assets classes.

2. Description of the sparse MRP optimisation problem

2.1. *The MRP as a constraint minimisation problem*

In this paper we build on the approach introduced by d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) for characterising a mean reverting portfolio. The authors used the predictability ratio defined for stationary processes by Box and Tiao (1977) and extended it by adding a cardinality constraint and a minimum variance constraint, in d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015), respectively. Let us assume a univariate model such that the asset s_t follows the process $s_t = \hat{s}_t + \epsilon_t$, where $\epsilon_t \sim \mathcal{N}(0, \sigma_\epsilon^2)$ represents i.i.d. noise, and $\hat{s}_t$ is an estimator for x_t, which can use all the information available up to $t-1$. With this formulation, Box and Tiao (1977) used the variance $\sigma^2 = \hat{\sigma}^2 + \sigma_\epsilon^2$ to define the predictability as the ratio of the variance of the estimator over the variance of the asset:

$$\nu = \frac{\hat{\sigma}^2}{\sigma^2}.$$

The interpretation motivating the use of this ratio is as follows: if the variance of the estimator $\hat{\sigma}^2$ is higher than that of the asset σ^2, then the dependence on the past dominates

the noise and thus the predictability is high, while if ν is small then the noise dominates the prediction. It is straightforward to extend this definition to a multivariate model where ν becomes the ratio of the variances of two portfolios. The Box and Tiao (1977) procedure requires us to define a model for the price process; in d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) the authors considered the prices to follow a VAR(1) process:

$$S_t = AS_{t-1} + \epsilon_t.$$

where S_t is a vector of prices, A the autoregressive coefficient and $\epsilon_t \sim \mathcal{N}(0, \Sigma)$ the noise with covariance matrix Σ. For the multivariate case, we use a weight vector x_t to form the portfolio $x_t^T S_t$ and obtain the predictability ratio:

$$\nu = \frac{x_t^T A^T \Sigma A x_t}{x_t^T \Sigma x_t}.$$

Box and Tiao (1977) showed that minimising the predictability is equivalent to solving the generalised eigenvalue problem, $\det(\nu\Sigma - A^T \Sigma A) = 0$. d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) highlighted the fact that we only need to find the extremal generalised eigenvalues to rewrite the problem as a minimisation. They used this new formulation as an optimisation problem to introduce a sparse constraint on the weight vector, however the resulting combinatorial problem has been proven to be a NP-hard. We can write this minimisation problem with normalisation and sparse constraint as:

$$\min_{x_t} \quad \frac{x_t^T A^T \Sigma A x_t}{x_t^T \Sigma x_t} \quad \text{subject to: } \|x\|_2 = 1, \quad (1)$$

$$\|x\|_0 = k. \quad (2)$$

In Cuturi and d'Aspremont (2013, 2015) the authors reformulated this optimisation to include a constraint on the minimum variance of the portfolio and proposed the use of two other utility functions, which have fewer constraining assumptions: the Portmanteau statistic, which tests if a process is white noise, and the crossing statistic, which represents the frequency at which a process crosses its mean. In this paper however we will focus solely on the predictability ratio, since its formulation is suited for the H-SGDLM model and the CCD algorithm.

Sipos and Levendovszky (2013), Fogarasi and Levendovszky (2013) followed the idea of d'Aspremont (2010) but considered the inverse problem. Indeed, they assumed the portfolio follows an Ornstein-Uhlenbeck process and considered the coefficient representing the mean reversion speed of the portfolio to be directly proportional to the predictability variable, as defined by Box and Tiao (1977). Hence they maximised the predictability instead of minimising it. In a second paper Sipos and Levendovszky (2013) proposed replacing the generalised eigenvalue maximisation problem introduced by d'Aspremont (2010) with a maximisation of the average return. This seems to highlight the fact that they compute the inverse of a mean reverting portfolio and, as explained by d'Aspremont (2010), by maximising the predictability they obtained a momentum portfolio. For this reason we follow

the approach of Box and Tiao (1977), d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) in the rest of this paper.

Since d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) modelled the price process as a VAR(1) process, the authors had to estimate A, the autoregressive coefficient, and Σ, the covariance matrix, in order to use the Box and Tiao (1977) formulation of predictability. Box and Tiao (1977) considered the least squares estimate for A and plugged this into their general eigenvalue problem. On the other hand, d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) added an L_1 penalty to obtain more stable estimates and highlight the dependencies between S_t and S_{t-1}. Hence, they used a likelihood with L_1 penalty to compute the covariance matrix and replaced the least squares estimate of A with a LASSO-based estimate. In Cuturi and d'Aspremont (2013, 2015) the $lag - k$ covariance matrices were computed according to their empirical estimates and the minimisation problem was reformulated with those variables.

2.2. Previous techniques used to solve this optimisation

Since the MRP optimisation problem is non-convex and includes a cardinality constraint, the classic minimisation algorithms will not work. In their papers, d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) solve the constraint optimisation problem using semi-definite programming. In particular they relax the problem into a convex semi-definite optimisation programme (SDP), which they then solve with a minimum eigenvalue solver. However, this relaxation only provides a sub-optimal solution. Indeed, while they proved that in special cases the relaxed problem is tight, i.e. gives the exact solution, in general the equivalence is not guaranteed. Nonetheless, when the correspondence is not exact, the solution of this semi definite relaxation can be deflated to produce a good approximation. In order to do so, the authors used a sparse PCA method which takes the SDP solution and converts it into a sparse vector of weights by recovering the leading sparse eigenvectors.

While we do not fully agree on the utility function proposed by Sipos and Levendovszky (2013) and Fogarasi and Levendovszky (2013), the methods the authors use to solve the resulting optimisation problem are nonetheless interesting. Indeed, they proposed two techniques: a simulated annealing (SA) approach and a feed forward neural network (FFNN) algorithm. In their paper the authors also compared them with exhaustive search methods on small enough problems, before using them to tackle bigger ones where those greedy approaches are too expensive to run. For the SA method they converted the problem into a combinatorial one by considering only integer values for the weights x. Thus, they run the SA algorithm on a discrete grid whose dimension corresponds to the number of assets. As a stochastic optimisation procedure this is not guaranteed to find the optimal solution, however, and its performance depends heavily on the choice of the initial vector. For the FFNN they first considered the problem in a smaller dimension and computed some optimal portfolio via exhaustive search to create a training dataset. Once the network has been trained with classic backpropagation on this sub-problem they run it on the large one

to obtain a weight vector, again with no guarantees of convergence. The results of the FFNN algorithm were also used as an initialisation procedure to select the weight vector for the simulated annealing steps.

These different approaches incorporate the cardinality constraint at different steps of the algorithm to obtain the closest approximation to the optimal solution of this non-convex problem. Unfortunately all of these methods provide only an approximation of the optimal weight vector and are expensive to compute. Furthermore, only the SDP technique can prove an optimal convergence in some special cases. In the coming section we will introduce a novel formulation of the MRP optimisation problem and a procedure to compute the weight vector which is guaranteed to converge and works efficiently even in a large environment.

3. New formulation of the MRP with H-SGDLM

3.1. Description of the H-SGDLM model

The Bayesian multivariate modelling of the market, H-SGDLM model of Griveau-Billion and Calderhead (2019), Gruber and West (2016), Corsi (2004) gives interesting insights on the behaviour of each asset and the rest of the market. The letters of H-SGDLM stands for heterogeneous simultaneous graphical dynamic linear model. This model extended the previously introduced SGDLM model of Gruber and West (2016) and HAR-RV model of Corsi (2004) to decompose the moves of the time series into different economically-meaningful variables which can be grouped into endogenous and exogenous ones. In addition, it does not assume stationarity of the time series and sequentially updates its parameters with each newly available data point thus producing new signals on the evolution of the different factors. Furthermore, the flexibility of this model makes it straightforward to extend it to fit our particular problem.

The H-SGDLM models each time series independently and then combines them to form a complete multivariate model of the market hence its matrix formulation:

$$S_t \sim \mathcal{N}\left(H_t \mu_t, \Sigma_t\right),$$
$$\text{With: } H_t = (I - \Gamma_t)^{-1},$$
$$\Omega_t = \Sigma^{-1} = (I - \Gamma_t)^T \Lambda_t (I - \Gamma_t).$$

Where Λ_t is the diagonal precision matrix with $\lambda_{j,t}$ on its diagonal, Γ_t is the matrix of the exogenous state coefficients, and $\mu_{j,t} = e_{j,t}\phi_{j,t}$ the mean vector. Thus, Ω_t corresponds to the sparse precision matrix and Σ_t the corresponding sparse covariance matrix.

The endogenous group of variables represents the influence of the previous information of this time series while the exogenous one represents the influence of the rest of the market. Hence, the endogenous variables include, among others, the leverage effect and the influence of lower frequencies on the current evolution. On the other hand, the exogenous variables combine an evolving set of assets from the market which have been selected for their influence on the time series. While in the paper Griveau-Billion and Calderhead (2019)

they mainly used the H-SGDLM to model and predict the variance of individual time series and the market, in this paper we will apply it directly on the logarithm of stock prices to obtain the variables needed to solve the mean reversion portfolio optimisation.

For asset j the time series of prices $s_{j,t}$ is modelled as a function of the state variables; both assuming a Normal Inverse-Gamma model. Let us denote by $\theta_{j,t}$ the state vector, $F_{j,t}$ the vector of variables and $G_{j,t}$ the state evolution matrix; the model reads:

$$s_{j,t} = F_{jt}\theta_{j,t} + v_{j,t},$$
$$\theta_{j,t} = G_{j,t}\theta_{j,t-1} + \omega_{j,t},$$
$$\text{with:} v_{j,t} \sim \mathcal{N}(0, \lambda_{j,t}^{-1}),$$
$$\omega_{j,t} \sim \mathcal{N}(0, W_{j,t}).$$

Where the variance of the states $W_{j,t}$ and of the observations $\lambda_{j,t}^{-1}$, are both assumed to follow an Inverse-Gamma distribution. The vectors of variables $F_{j,t} = (e_{j,t}, r_{sp_t(j),t})$ and states $\theta_{j,t} = (\phi_{j,t}, \gamma_{j,t})$ are composed of two subsets representing the endogenous and exogenous variables and coefficients, respectively. The endogenous variables represent the influence of the past information of asset j on its future behaviour. In order to model the influence of the past prices on the future evolution of the time series we include the past returns and the average of the past prices over three different frequencies: day, week and month. In addition, because we work at the daily frequency we want to include information from higher frequency behaviours. When available, we use the open, high, low and close prices as summaries of the day's movement. More specifically, we use three variables constructed from these prices. Let us denote the different versions of the daily price of asset j at time t by $s_{j,t}^{test}$ where *test* specifies the metric, for example *open* for the price at the open. Thus, the selected variables representing the intra day moves are:

$$r_t^{low} = log(s_t^{low}) - log(s_{t-1}^{low}),$$
$$CH_t = \frac{s_t^{high} - s_t^{close}}{s_t^{high} - s_t^{low}} - 0.5,$$
$$COHL_t = \frac{s_t^{close} - s_t^{open}}{s_t^{high} - s_t^{low}}.$$

where r represents the log-return. In addition to those variables we will include the leverage effect representing the asymmetric influence of previous positive or negative daily, weekly and monthly return on the future move. Thus the complete vector of endogenous variable is:

$$e_{j,t} = (1, as_{j,t}^d, as_{j,t}^w, as_{j,t}^m,$$
$$r_{j,t}^d, r_{j,t}^w, r_{j,t}^m,$$
$$r_{j,t}^{low}, CH_{j,t}, COHL_{j,t},$$
$$r_{j,t}^{(d)+}, r_{j,t}^{(d)-}, r_{j,t}^{(w)+}, r_{j,t}^{(w)-}, r_{j,t}^{(m)+}, r_{j,t}^{(m)-})^T.$$

Where for each variable the exponent represents the different frequencies: daily, weekly and monthly; as^f represents the

averaged price over the frequency f; r^f is the past log-return at frequency f. Hence the variables of the endogenous vector $e_{j,t}$ represent different economic factors influencing the next price movement.

The exogenous variables are composed of a selected group, $sp_t(j)$, of assets which move simultaneously to stock j. This part is composed of two steps. The first step proposes candidates for inclusion in $sp_t(j)$, which are then included for a certain time period in the vector of variables. After this time period we look at their signal to noise ratio to decide whether to include it in the group of parents $sp_t(j)$ or not. Once the parents with the highest influence are selected, they are included in the vector of exogenous variables $r_{sp_t(j),t}$. The coefficients $\theta_{j,t}$ of the variables $F_{j,t}$ are then learned sequentially with each new data point following the classic state-space update.

3.2. Sparsity with the H-SGDLM

As explained by d'Aspremont (2010), Cuturi and d'Aspremont (2013, 2015) sparsity is an important factor to take into account when building mean reverting portfolios. The previously introduced H-SGDLM model constructs a sparse representation of the whole market. Indeed, instead of the classic covariance matrix the cross series relationships are modelled by the exogenous variables; in each individual DLM the exogenous variables inform the mean of the distribution and are sequentially selected and updated for each asset. When combining these individual DLMs to obtain a multivariate distribution, the exogenous coefficients inform both the mean and variance of the multivariate normal through the matrix Γ. As described previously, the selection of these variables follows a two-step process: first filtering with the Wishart covariance matrix, then testing for its capacity to explain the evolution of the current price given its signal to noise ratio. The number of parents allowed for each DLM is a parameter to be chosen; the smaller this number the greater the sparsity of the resulting multivariate distribution.

Thanks to this two steps process, the cross-series relationships are modelled as asymmetric. While some stocks might not be present in any core-group other might be in many. In other words, some assets influence the behaviour of many others in the market while others are mostly followers. These cross-series relationships are represented in the sparse covariance matrix Γ. Because the number of allowed parents is chosen small, the resulting covariance matrix Σ is sparse. And, thanks to its construction the non-zeros elements represent stocks with simultaneous behaviours. Thus, any mean reverting portfolio should be constructed within one of these subsets build from the stocks with the highest non-zero coefficient in the sparse covariance matrix Γ.

3.3. MRP reformulated with H-SGDLM data

We follow the approach of d'Aspremont (2010) and Cuturi and d'Aspremont (2013, 2015) to minimise the predictability as defined in optimisation problem 2, but instead of using the

ratio we will work with the difference:

$$\min_{x_t} \quad x_t^T A^T \Sigma A x_t - x_t^T \Sigma x_t \quad \text{subject to:} \ \|x\|_1 = 1,$$
$$\|x\|_0 = k.$$

While d'Aspremont (2010) and Cuturi and d'Aspremont (2013, 2015) modelled the price process by a VAR(1) model (i.e. with stationary autoregressive coefficient A and stationary covariance matrix Σ), here we use the outputs from the H-SGDLM thereby relaxing this condition of stationarity. By considering the price process s_t to follow a time varying autoregressive process TVAR(1) we can express the portfolio by: $P_t = x_t^T S_t = x_t^T A_t S_{t-1} + x_t^T \epsilon_t$, where the prediction of S_t assuming all the data up to $t-1$ known is denoted by $\hat{S}_t = A_t S_{t-1}$. We recall that the motivation for the Box and Tiao (1977) predictability ratio was to compare the variance of the prediction to the measured one and see if the prediction dominates the noise, implying that the process is predictable. Thus the ratio of the predicted portfolio variance over the measured one is:

$$\nu_t = \frac{\hat{\sigma}_t^2}{\sigma_t^2} = \frac{\mathbb{E}[x_t^T A_t S_{t-1} S_{t-1}^T A_t^T x_t | D_t]}{\mathbb{E}[x_t^T S_t S_t^T x_t | D_t]} = \frac{x_t^T A_t \tilde{\Sigma}_{t-1} A_t^T x_t}{x_t^T \tilde{\Sigma}_t x_t}.$$

Where $\tilde{\Sigma}_t$ corresponds to the empirical covariance matrix measured with prices up to t. Therefore, this ratio corresponds to the the predicted covariance matrix using the autoregressive coefficient A_t over the measured portfolio variance at t. But since our model can actually infer the next covariance matrix Σ_t we could use instead:

$$\nu_t = \frac{x_t^T \Sigma_t x_t}{x_t^T \tilde{\Sigma}_t x_t}.$$

The H-SGDLM model gives us the autoregressive coefficient A_t, and sparse predicted covariance matrix Σ_t. Therefore the predictability ratio could be computed in different ways using either one of these outputs. Since for the optimisation problem we work with the difference instead of the ratio, possible utility functions include:

$$U_1(x) = x^T \left(A_t \tilde{\Sigma}_{t-1} A_t^T - \tilde{\Sigma}_t \right) x,$$
$$U_2(x) = x^T \left(\Sigma_t - \tilde{\Sigma}_t \right) x,$$
$$U_3(x) = x^T \left(A_t \tilde{\Sigma}_{t-1} A_t^T - \Sigma_t \right) x.$$

Let us detail the different options. U_1 uses the measured covariance matrices at $t-1$ and t and computed autoregressive coefficient A_t. U_2 compares the sparse covariance matrix at t inferred with data up to $t-1$ to the measured covariance matrix using prices up to t. U_3 compares a prediction of the variance using the covariance matrix measured with data up to $t-1$ and the computed autoregressive coefficients to the predicted covariance matrix Σ_t. Now since we assumed all the data up to t is known for computing the predictability ratio at t, we could do the same computation as U_3 but using the data available at t, i.e. compare the $t+1$ predictions. However, this option (as with U_3) does not respect the predictability as defined by Box and Tiao (1977) since it would

compare two predictions instead of a predicted versus a measured value; for this reason we work with U_1 and U_2. With this set-up we can either use the predicted covariance matrix or the predicted autoregressive coefficient. Let us denote the minimisation of the covariance differences as problem *PC* and the other one as problem *PA*. Further, let us denote the utility function by $x^T(D_t - \tilde{\Sigma}_t)x$, where D_t is the predicted covariance matrix. Then, for *PC* we have $D_t = \Sigma_t$, while for *PA* it is $D_t = A_t \tilde{\Sigma}_{t-1} A_t^T$.

4. Solving the minimisation problem with CCD

4.1. The CCD algorithm for non-convex function

Coordinate descent algorithms work by optimising one direction at a time instead of all of them simultaneously as classical descent algorithms do. The main drawback for this type of algorithm is their requirement for the utility function to be strictly convex and differentiable. Tseng (2001) proved that the convergence can be extended to a specific type of non-convex and non-differentiable function. In particular they considered a non-convex objective function that can be decomposed into a quasi-convex and differentiable function f_0 and a non-differentiable but additively separable one f_k:

$$f(w) = f_0(w) + \sum_{k=1}^{m} f_k(w_k).$$

If a function can be written in such a way, then a block coordinate descent algorithm applied on this function will converge to the global minimum of f with respect to w. More specifically, the convergence is guaranteed according to theorem 5.1 of Tseng (2001) if the functions respect the following conditions: f_0 must be continuous, f is quasi-convex and hemivariate in each coordinate block and the functions f_k are left side continuous $\forall k$. In addition, Tseng (2001) made assumptions on the domain of f_0, for which we refer the reader to the original paper.

The coordinate descent algorithm works by iteratively updating the weight vector w^r. At each step it updates the value of the weights w_i^r for all $i \in [1, N]$ by minimising f assuming all the other weights $w_{j \neq i}^r$ fixed: $w_i^r = min_{w_i^r} f(w^r)$. Then, it minimises the next weights using the updated values of the other weights. Once the entire vector w^r has been updated the iteration starts again until convergence: $\|w^r - w^{r-1}\| \leq \epsilon$.

Subsequently, Yen and Yen (2014) used this algorithm to solve the minimum variance portfolio optimisation with elastic net constraints since it naturally fits in the formulation of Tseng (2001). The Lagrangian of this problem is:

$$\begin{aligned} L&(W, \gamma, \Sigma, \beta, \alpha) \\ &= W^T \Sigma W + \beta \left(\alpha \|W\|_{l_1} + (1 - \alpha) \|W\|_{l_2}^2 \right) \\ &\quad - \gamma \left(W^T 1 - 1 \right), \end{aligned}$$

where β is the penalty parameter, α the weight balancing the importance of the L_1 and L_2 terms, γ the Lagrange multiplier

and W the weights vector. Fixing γ, we can solve the KKT conditions with respect to the weight w_i to obtain:

$$w_i = \frac{ST(\gamma - z_i, \beta \alpha)}{2 \left(\sigma_i^2 + \beta (1 - \alpha) \right)},$$

$$z_i = 2 \sum_{j \neq i}^{N} w_j \sigma_{i,j},$$

where $\sigma_{i,j}$ corresponds to the covariance between weights i and j, and $ST(x, y) = sign(x)max(|x| - y, 0)$ is the soft-threshold function. The CCD algorithm recursively updates each weight w_i with this equation until the vector w converges. The following section will detail the application of the CCD algorithm to solve the sparse mean reverting portfolio optimisation problem.

4.2. The CCD algorithm applied to sparse mean reverting portfolios

Instead of the cadinality constraint used by d'Aspremont (2010) and Cuturi and d'Aspremont (2013, 2015) to fix the number of non-zeros weights to a pre-selected value, we use the graphical structure learned by the H-SGDLM model to select a group of assets of size k. While in this paper we use the predicted covariance matrix D_t obtained from the H-SGDLM model, the CCD algorithm we propose in this section could be applied using any other covariance matrix prediction, however we note that without the H-SGDLM sparsity selection, an L_1 regularisation coefficient would be needed to impose sparsity. Since we work with the difference instead of the ratio we add a L_2 regularisation factor to the utility function. The minimisation problem becomes:

$$\min_{x_t} \quad x_t^T \left(D_t - \tilde{\Sigma}_t \right) x_t + \beta \|x_t\|_2^2, \quad \text{subject to: } \|x_t\|_1 = 1, \tag{3}$$

where β is the weight of the L_2 norm. With this formulation only the normalisation constraint remains. We use the L_1 norm of the weight as the normalisation constraint because we want to take into account the full exposure of the portfolio. Hence, the normalisation takes into account the short positions in the total weight. In addition to its smoothness property, the L_2 regularisation term here guarantees the positive semi-definiteness of the problem. Indeed, although D_t and Σ_t are positive semi-definite covariance matrices, their difference is not guaranteed to be. We circumvent this issue with the addition of an L_2 weight, which acts as a matrix regularisation coefficient to keep the difference within the positive semi-definite domain.

This approach is not perfect however, since the L_2 constraint adds a bias to the solution. Therefore we want to select the smallest possible coefficient β that guarantees the positive semi-definiteness of the matrix $M = D_t - \tilde{\Sigma}_t + \beta I$, where I corresponds to the identity matrix. We select a value $\beta = \beta_0$ and at each time t compute the optimal solution, then we check for the positivity of the resulting utility function. In case of negativity, we increase the value of β by 10^{-5} until the function becomes positive, but only for a maximum of 10 steps. If the maximum number of iterations is reached we

increase the step-size to 10^{-1} and start again. This incremental approach is computationally feasible due to the speed of the CCD algorithm, although we have to take care with choice of β_0; if it is chosen too small it will end up in the loop step too often, while if chosen too big it adds an unnecessary bias. In addition, if the value of the MRP part becomes negligible compared to the value of the L_2 constraint then the optimisation algorithm will just minimise the L_2 norm of the weight and thus produce the equal weighted portfolio. We detail in Section 6 a pragmatic selection of values for the problems considered.

With the parameter β selected as the smallest value that guarantees the positive semi-definiteness of the matrix $M = D_t - \tilde{\Sigma}_t + \beta I$. When the matrix M is singular we incrementally increase the value of the parameter β to shift M into the positive semi-definite space. There are different ways to prove that such shrinkage method is guaranteed to bring the matrix M into the positive semi-definite space by adding positive elements to the diagonal. For example, if β tends to infinity, all non-diagonal elements of the matrix M will become negligible compared to the diagonal values, β. Thus, $M \sim \beta I$ becomes a diagonal matrix with strictly positive elements on its diagonal and thus strictly positive determinant. Which proves its positive semi-definiteness.

For the sake of clarity, since we consider all the data up to t known and assume all the variables in the optimisation problem are taken at time t, we drop the subscript t in the following expressions. The Lagrangian of our minimisation problem (3) follows as:

$$\mathcal{L}(x, \gamma) = x^T \left(D - \tilde{\Sigma} + \beta I \right) x + \gamma \left(\|x\|_1 - 1 \right), \quad (4)$$

This function satisfies the convergence requirements defined by Tseng (2001) since we can write the problem as the combination of a differentiable, quasi-convex term and a non-differentiable but additively separable one. Thus, we can follow a coordinate descent to find the global optimum. Without loss of generality we will fix the Lagrange coefficient to zero and renormalise the weight vector at the end of each loop. Let us denote the elements of matrix D by $d_{i,j}$ and of Σ by $\sigma_{i,j}$. Then, the line i of the matrix $(D - \tilde{\Sigma} + \beta I)x$ can be written $(D - \tilde{\Sigma} + \beta I)x|_i = (d_i^2 - \sigma_i^2 + \beta)x_i + \Sigma_{j \neq i}(d_{i,j} - \sigma_{i,j})x_j$. Where $\sigma_i^2 = \sigma_{i,i}$ and $d_i^2 = d_{i,i}$ are the diagonal elements of D and Σ respectively. Thus, the gradient of the Lagrangian in Equation (4) with respect to x_i is:

$$\nabla_{x_i} \mathcal{L}(x, \gamma) = 2 \left(d_i^2 - \sigma_i^2 + \beta \right) x_i + 2 \Sigma_{j \neq i} \left(d_{i,j} - \sigma_{i,j} \right) x_j,$$

Therefore, for the gradient to be zero each weight x_i must follow:

$$x_i = -\frac{2 \Sigma_{j \neq i} \left(d_{i,j} - \sigma_{i,j} \right) x_j}{2 \left(d_i^2 - \sigma_i^2 + \beta \right)}. \quad (5)$$

The coordinate descent algorithm iteratively updates the weights with Equation (5) until convergence. While we did not include an L_1 norm on the weights because we enforce sparsity directly using the results from the H-SGDLM model, it is straightforward to include one. Indeed, with an L_1 regularisation term and coefficient λ_1, the numerator of Equation 5 simply becomes $ST\left(2\Sigma_{j \neq i}(d_{i,j} - \sigma_{i,j})x_j, \lambda_1\right)$, where $ST(.,.)$ is the soft-thresholding function.

4.3. Complexity analysis

Let us consider an environment with k different time series, i.e. $dim(x_t) = k$, and limit the number of iterations of the CCD algorithm to $m_{\max}$. Thus, the computational complexity of solving Equation (3) for a fix β value is $0(m_{\max}k(k-1))$. Then, the algorithm perform a line search of growing β values to find the first β for which that the matrix M becomes positive semi-definite. Let us assume the algorithm needs B steps to find β, the complexity of solving Equation (3) including the β search is $O(Bm_{\max}k(k-1))$. Let us compare the computational complexity of two increments: i_β versus $i_{\beta,s} = i_\beta/s$ with $s > 0$. Using the smaller increment, $i_{\beta,s}$, increases the computational complexity of solving Equation (3) over the bigger increment, i_β, by $O(sBm_{\max}k(k-1))$.

5. Trading a mean reverting portfolio

5.1. Jurek and Yang trading strategy

In the papers Cuturi and d'Aspremont (2013, 2015) used the Jurek and Yang (2007) strategy to trade the mean reversion of the obtained MRP portfolios. This strategy considers the portfolio to follow a stationary VAR(1) process $P_t = \rho P_{t-1} + \sigma \epsilon_t$, with $|\rho| < 1$ and mean μ, where the parameters were obtained using a least squares estimate. Let us denote by w_t the investment weight of portfolio P_t, at t, which the authors define by $w_t = (\mu - p_t)\rho/\sigma^2$. In their paper from 2013 the authors considered trading swaps and concluded that the gains are too small to create a profitable strategy in practice, since the transaction costs cancelled all arbitrage opportunities. On the other hand, in the 2015 paper they built baskets of options to trade implied volatility and with added constraints they obtained performances robust to costs.

5.2. Ornstein-Ulhenbeck based trading strategy

In their papers, Fogarasi and Levendovszky (2013), Sipos and Levendovszky (2013) assumed the portfolio to follow an Ornstein-Ulhenbeck process $P_t \sim \mathcal{N}(\mu, \sqrt{\sigma^2/2\rho})$ and considered three scenarios. Either the portfolio is in a stationary state and no investment is made, or it is outside a threshold and a position is entered. The authors define a threshold ϵ such that:

- H1: $P(p_t < \mu - \alpha) = \epsilon/2$, the portfolio's value is bellow its long term mean and they buy it.
- H2: $P(p_t \in [\mu - \alpha, \mu + \alpha]) = \epsilon$, the portfolio is in its stationary state and thus they stay neutral.
- H3: $P(p_t > \mu + \alpha) = \epsilon/2$, the portfolio's value is above its long term mean and they sell it.

Once a position is entered, they keep it until the portfolio comes back to its stationary state. In this strategy based on O-U thresholds the maximum amount is always invested, in contrast to the original strategy of Jurek and Yang (2007) where the portfolio's weight can take any real number.
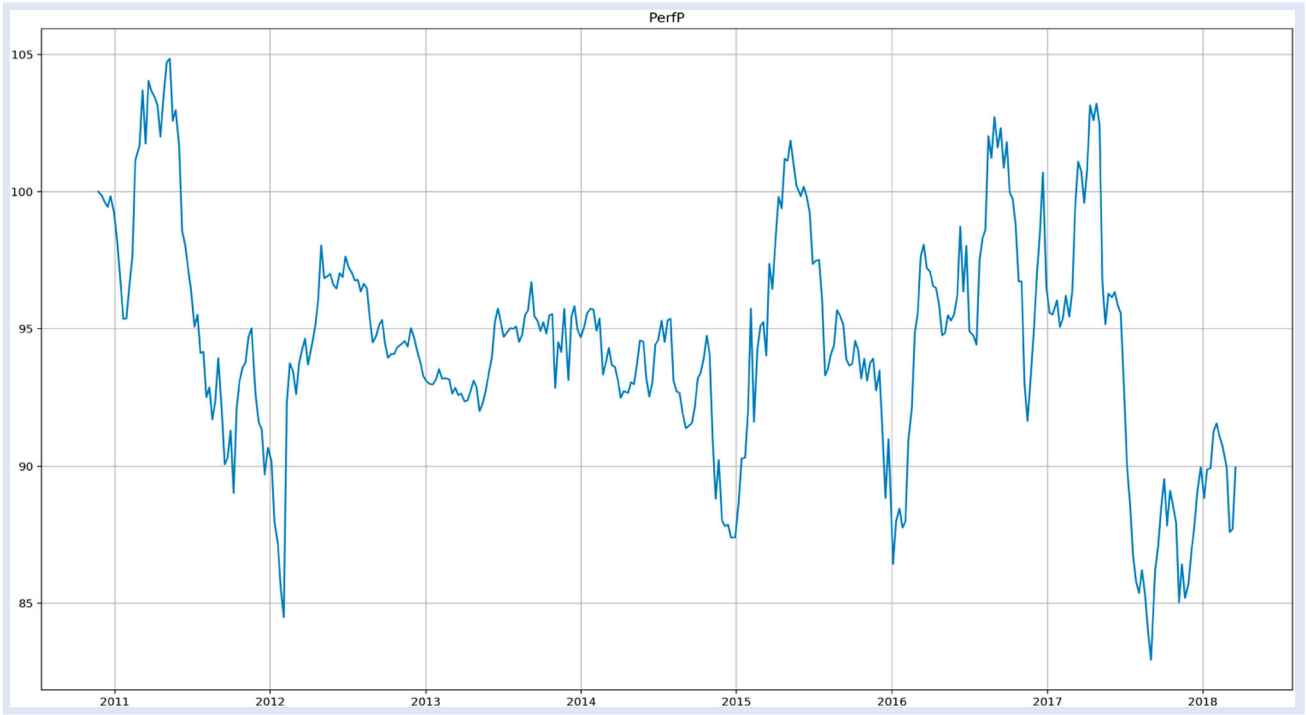
Figure 1. This figure shows the performance of the MRP portfolio computed for 371 stocks from the S&P500. On this figure we can observe the mean reversion of the obtained portfolio.

5.3. *Mixed mean reversion trading strategy*

We could also use a combination of those two ideas. Instead of always investing an amount corresponding to the distance from the mean (as in Jurek and Yang 2007), we could instead adopt this approach only when the move is larger than a threshold, resulting in the strategy:

- H1: $W_t = (\rho/\sigma^2)((\mu - \alpha) - P_t)$,
- H2: $W_t = 0$,
- H3: $W_t = (\rho/\sigma 2)((\mu + \alpha) - P_t)$,

For the rest of this paper we will refer to this strategy as the mixed $JY - OU$ strategy.

6. Results

6.1. *US stocks from the S&P500*

In order to test our algorithm on stocks, we selected the 371 stocks from the S&P500 with data available since 2001. This dataset consists of the OHLCV data for each stock at the end of each trading day. However, instead of using daily time points we worked on weekly time points, i.e. we selected one data every 5 time points. Hence, the one-step ahead inference on which the H-SGDLM learns corresponds to a one week ahead forecast. Since the H-SGDLM takes some time to converge we study the results between 2003 and 2018. The vector of variables includes the OHLC data as described in Section 3.1. The size of the core parent group was fixed to 10 with an update every 10 time steps and due to the memory limit of the GPU card the number of Monte-Carlo samples was fixed to 500.

For the parametrisation step we selected half the dataset, from 2011 to 2018. We used the second half of the dataset to avoid fitting the parameters over the 2008 crisis, as we wanted our out-of-sample to test the robustness of the parameters over this crisis period. The MRP algorithm has two parameters, β the L_2 weight and λ the coefficient of the exponential weighting we use to compute the empirical covariance matrix $\tilde{\Sigma}$. Since we use two different optimisation problems, *PA* and *PC*, we allow the parameters to be different for each one. To find the acceptable range of values for those parameters we compared the computed MRP portfolios to the equally weighted one. Once this range was found, we selected the value that created the most mean-reverting portfolio over this time window. As an example figure 1 shows the resulting portfolio obtained over the whole dataset with $\beta = 10^{-5}$ and $\lambda = 0.9$. Since we want a sparse portfolio, we fixed the size of the portfolio to 50 different assets following the methodology described in (3.2) that uses the matrix of parents' coefficients Γ. We then selected $\beta = \{10^{-3}, 10^{-4}, 10^{-5}, 10^6\}$, $\lambda_{PA} = 0.98$ and $\lambda_{PC} = 0.85$. For the trading part we have to select the length, T_{tr} of the time window on which the different trading parameters will be computed at each time t. We looked at the performances of the backtests and chose $T_{tr} = 50$. We then compared the different trading strategies described in (5). For each strategy we averaged the results obtained for the different β values and the different *PA* and *PC* problems. The figures 2 and 3 show the results for the Jurek and Yang (2007) and O-U based strategies respectively. Interestingly on this dataset the mixed $JY - OU$ one had worse performance then the other two. Even though the backtests do not include any costs it is interesting to observe the steady performance even during the 2008 crisis.
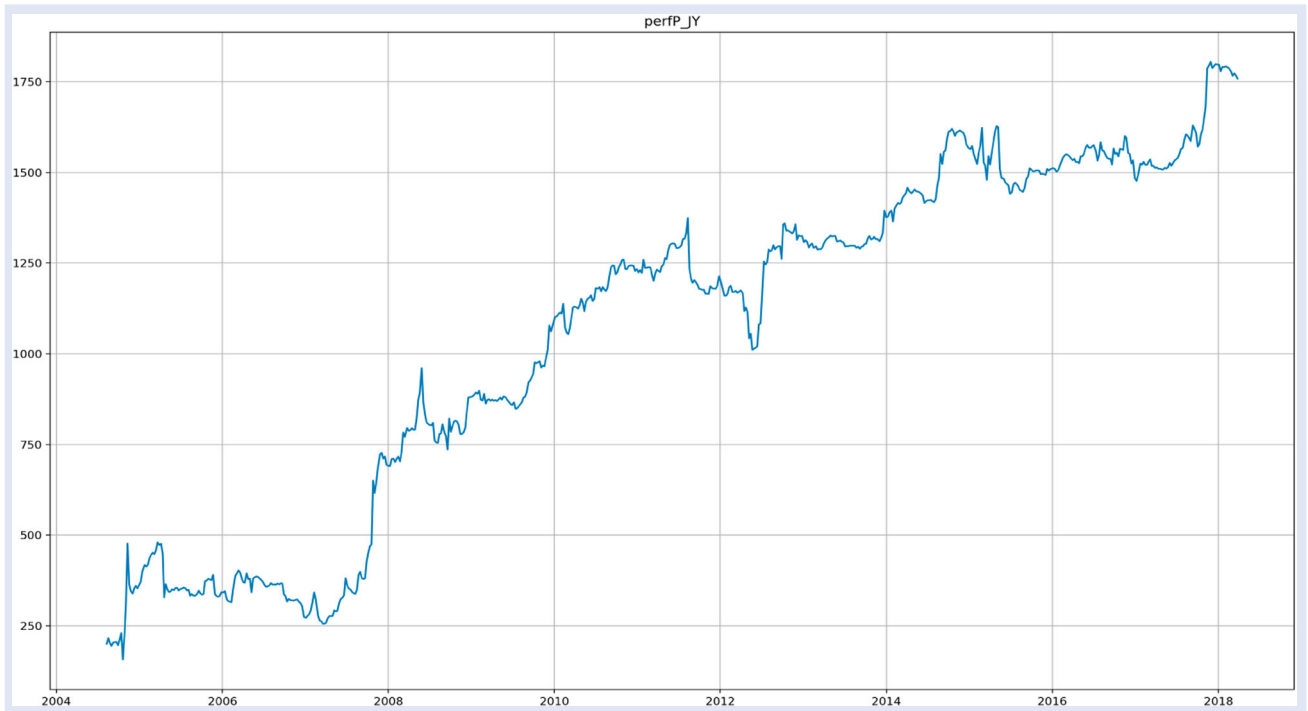
Figure 2. This plot shows the performance of the Jurek and Yang (2007) trading strategy applied on the MRP portfolio computed on an evolving set of 50 stocks from the S&P500.
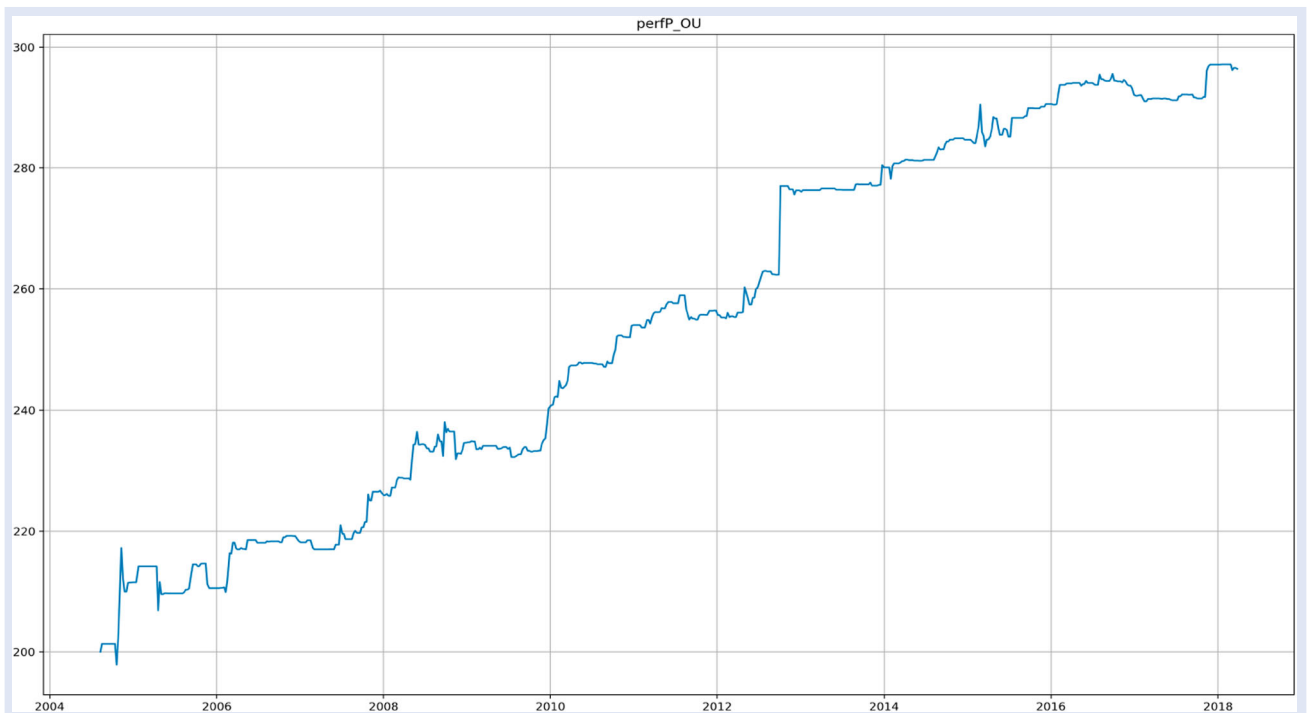

Figure 3. This plot shows the performance of the O-U based trading strategy applied on the MRP portfolio computed on an evolving set of 50 stocks from the S&P500.

6.2. FX futures

In this section we test our algorithm on exchange rates data with a dataset consisting of 22 different FX-futures between 2006 and 2018. For this environment, the H-SGDLM used the average of the past 40 daily-, weekly-, monthly-log-returns. This average used an exponential weighting with a coefficient of 0.98. The size of the parent group was fixed at 5 and the updates happened every 10 time steps. Just like in the previous example, we used a weekly time point dataset, such that the parent selection corresponded to an update every 50 days. Furthermore, we used 2000 Monte-Carlo samples.
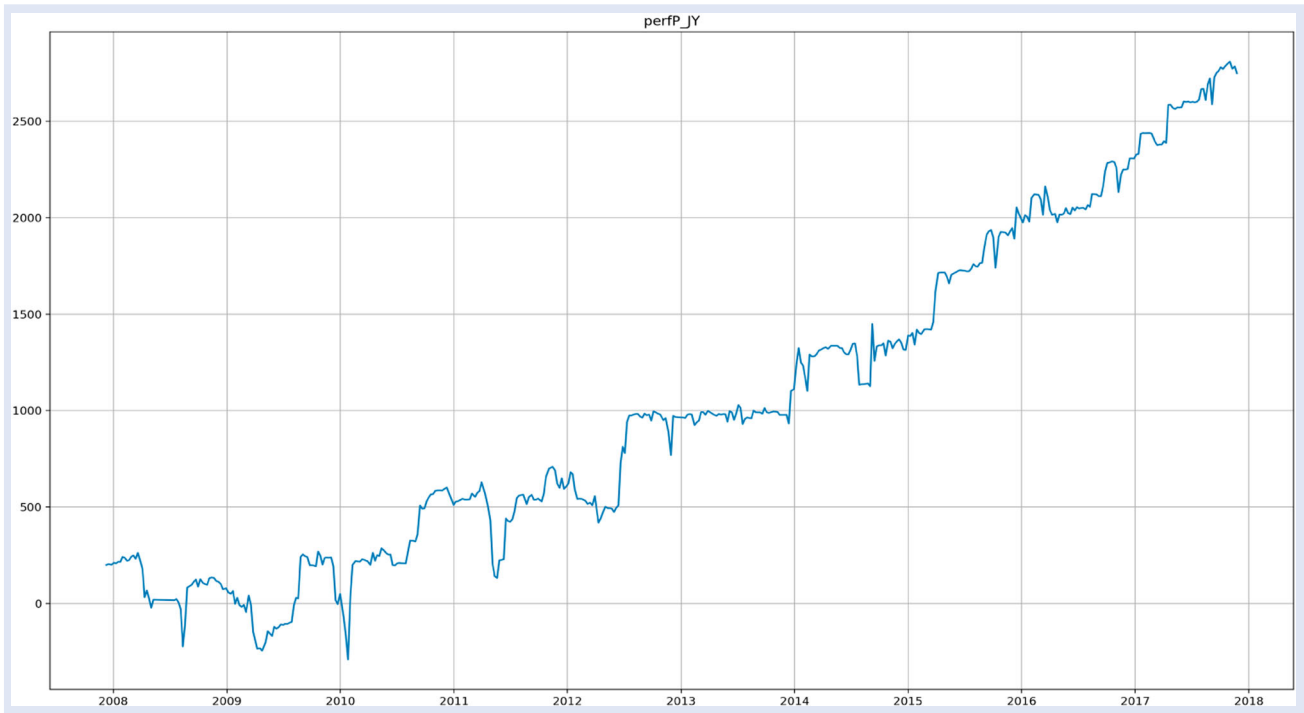
Figure 4. This plot shows the performance of the Jurek and Yang (2007) trading strategy applied on the MRP portfolio computed on an evolving set of 10 FX-futures.
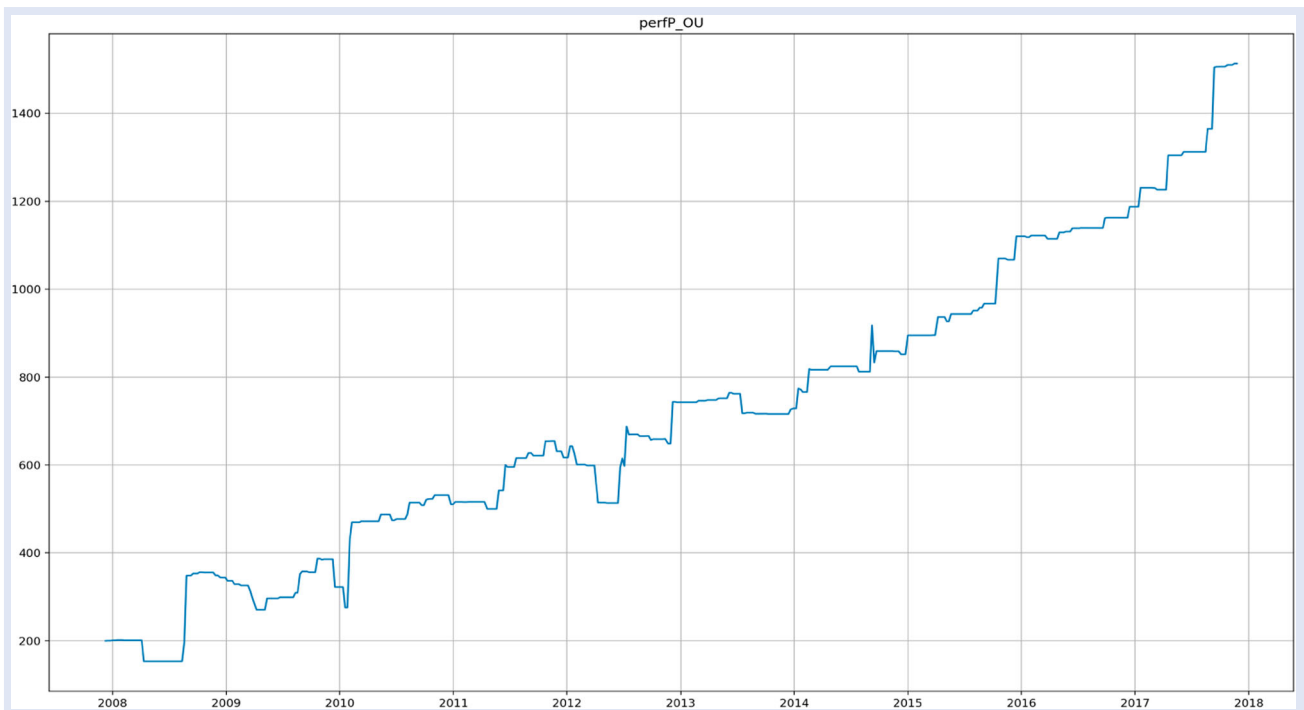

Figure 5. This plot shows the performance of the mixed $JY - OU$ trading strategy applied on the MRP portfolio computed on an evolving set of 10 FX-futures.

We followed the same methodology to find the parameters as the one described for S&P500 data. Hence, the parametrisation step used half the dataset, from 2012 to 2018 and the number of asset of the portfolio, or sparsity constraint, is fixed to 10. The exponential weighting coefficient is chosen to be the same for both problems PA and PC, i.e. $\lambda_{PA} = \lambda_{PC} = 0.8$. But, the L_2 coefficients are different: $\beta_{PA} = \{10^{-1}, 10^{-2}, 10^{-3}\}$ and $\beta_{PC} = \{10^{-2}, 10^{-4}, 10^{-5}\}$. For

the trading strategies the time window was fixed to $T_{tr} = 10$ time points equal to 50 trading days. figure 4 shows the backtest performance using the Jurek and Yang (2007) trading strategy while figure 5 corresponds to the backtest of the mixed $JY - OU$ strategy. As with the previous backtest on stocks it is interesting to observe the algorithm's steady performance. In this case however, contrary to the backtest on stocks, the mixed $JY - OU$ strategy performed slightly better
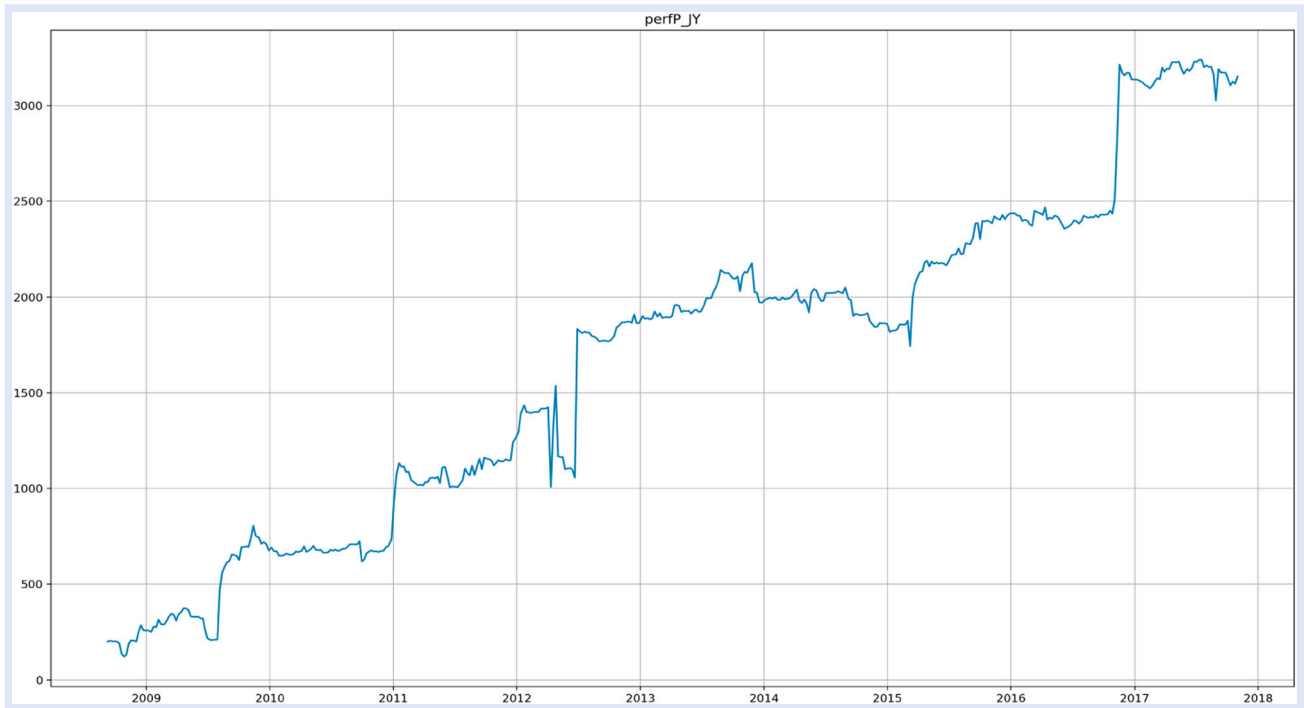
Figure 6. This plot shows the performance of the Jurek and Yang (2007) trading strategy applied on the MRP portfolio computed on an evolving set of 20 ETF-futures.
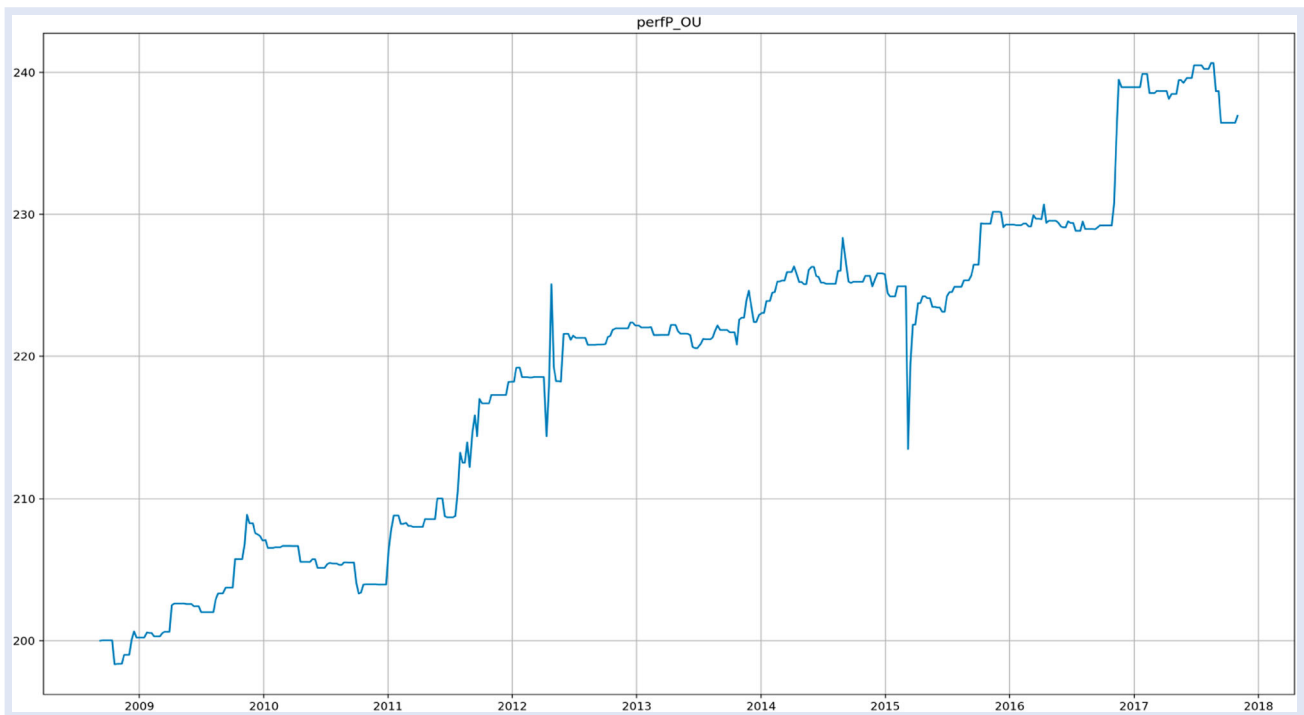

Figure 7. This plot shows the performance of the O-U based trading strategy applied on the MRP portfolio computed on an evolving set of 20 ETF-futures.

than the O-U based one, hence we only show the performance of the mixed one.

6.3. ETF futures

Finally we considered the dataset of futures on ETFs, in particular 75 different futures with OHLCV data from 2008 to 2018. Hence, the H-SGLDM learns on log-prices with the variables described in section 3.1. As for the previous datasets we used weekly time points instead of daily ones. The size of the core parent group is fixed to 10 with an update every 10 time steps and we computed 2000 Monte-Carlo samples.

We followed the same methodology as before to find the MRP parameters. The sparsity variable was fixed to 20

futures, while the time window used for parametrisation was 2013 to 2018. We selected exponential weighting coefficients of $\lambda_{PA} = 0.85$ and $\lambda_{PC} = 0.98$, and L_2 coefficients: $\beta_{PA} = \{10^{-5}, 10^{-6}\}$ and $\beta_{PC} = \{10^{-3}, 10^{-4}\}$. As with FX-futures the time window for the trading strategies was fixed to $T_{tr} = 10$ time points. As for the dataset on stocks, the mixed $JY - OU$ strategy performed worse than the other two. Figure 6 shows the backtest of the Jurek and Yang (2007) trading strategy and figure 7 corresponds to the performance of the O-U based approach. While these backtests are more volatile than the those on FX-futures and US-stocks they still have a relatively steady growth highlighting the robustness of this portfolio construction.

7. Conclusion

In this paper, we presented a new approach for computing a sparse mean reverting portfolio within a large environment. The goal was to find an efficient procedure that is able to find a sparse subset of series from a large environment of time series, for which we could compute a linear combination exhibiting the mean-reverting property. Furthermore, the solution had to work on the more complicated problem of subsets larger than two assets. Given such a mean-reverting portfolio the performance of our solution was numerically evaluated by performing backtests in different investment environments.

The solution we proposed in this paper leveraged the data obtained by the newly introduced H-SGDLM model. Indeed, we were able to reformulate the MRP problem as a quasi-convex optimisation problem with a normalisation constraint. In addition, the sparsity constraint was moved away from the minimisation formulation and instead we used the matrix of core parents from the H-SGDLM to select the subset of time series on which to compute the MRP. Using different variables from the H-SGDLM model we obtained two different formulations of the problem, *PA* and *PC*, using the matrix of the regression coefficients and the predicted sparse covariance matrix, respectively. This new formulation is perfectly suited for a cyclical coordinate descent (CCD) algorithm. Hence, once the data from the H-SGDLM is available the CCD optimisation step finds the global solution of the MRP minimisation problem.

As a result of this formulation it is possible to compute the optimal MRP at each time t even on environments involving hundreds of time series. We then evaluated different trading strategies to take advantage of this mean-reversion. We performed backtests on three different datasets: stocks

from the S&P500, FX-futures and ETF-futures. All the resulting backtests exhibited steady performances through time even on the out-of-sample part of the backtest that included the financial crisis, suggesting a certain robustness of our proposed solution.

Disclosure statement

ORCID

T. Griveau-Billion ⓘ http://orcid.org/0000-0001-6371-827X

References

Box, G.E.P. and Tiao, G.C., A canonical analysis of multiple time series. *Biometrika*, 1977, **64**, 355–365.

Corsi, F., A simple long memory model of realized volatility. *SSRN Electronic Journal*, 2004.

Cuturi, M. et d'Aspremont, A., Mean reversion with a variance threshold. In *International Conference on Machine Learning*, 2013. p. 271–279.

Cuturi, M. and d'Aspremont, A., Mean-reverting portfolios: Tradeoffs between sparsity and volatility. *arXiv preprint arXiv:1509.05954*, 2015.

d'Aspremont, A., Identifying small mean-reverting portfolios. *Quantitative Finance*, 2010, **11**, 351–364.

Fogarasi, N. and Levendovszky, J., Sparse, mean reverting portfolio selection using simulated annealing. content.iospress.com, 2013.

Griveau-Billion, T. and Calderhead, B., A Dynamic Bayesian Model for Interpretable Decompositions of Market Behaviour. arXiv.org, 2019.

Griveau-Billion, T., Richard, J.C. and Roncalli, T., A fast algorithm for computing high-dimensional risk parity portfolios. SSRN, 2013.

Gruber, L.F. and West, M., Bayesian forecasting and scalable multivariate volatility analysis using simultaneous graphical dynamic models. arXiv.org, 2016.

Johansen, S., Cointegration: An overview. ku.dk, 2005, pp. 1–40.

Jurek, J.W. and Yang, H., Dynamic portfolio selection in arbitrage. *SSRN Electronic Journal*, 2007.

Sipos, I.R. and Levendovszky, J., Optimizing sparse mean reverting portfolios. *Algorithmic Finance*, 2013, **2**, 127–139.

Tseng, P., Convergence of a block coordinate descent method for nondifferentiable minimization. *J. Optim. Theory. Appl.*, 2001, **109**, 475–494.

Yen, Y.M. and Yen, T.J., Solving norm constrained portfolio optimization via coordinate-wise descent algorithms. *Computational Statistics and Data Analysis*, 2014, **76**, 737–759.