Detecting multivariate market regimes via clustering algorithms

2024-03-13

James Mc Greevy
james.mcgreevy@kaiju.ai
KCM Quantitative & AI Researcher
(Corresponding Author)

Aitor Muguruza
aitor.muguruza-gonzalez@kaiju.ai
KCM Director of Data Science

Zacharia Issa
zacharia.issa@kcl.ac.uk
King's College, London

Cris Salvi
c.salvi@imperial.ac.uk
Department of Mathematics
Imperial College, London

Jonathan Chan
jonathan.chan@kaiju.ai
KCM Quantitative & AI Researcher

Zan Zuric
zan.zuric@kaiju.ai
KCM Quantitative & AI Researcher

Detecting multivariate market regimes via clustering algorithms

2024-03-13

Abstract

In this paper we study the joint dynamics of multivariate time series using an unsupervised learning technique and demonstrate its use in pairs trading and portfolio design. We present a novel non-parametric market regime detection method for multidimensional data. The detection procedure is based on a k-means clustering algorithm which makes use of distances between distributions to discriminate between different market regimes, which are categorised by their mean, variance and correlation. In particular, we empirically investigate the performance of the algorithm endowed with either Wasserstein distances or Maximum Mean Discrepancies. We suggest a two-step approach to clustering multivariate data using this new method, which we show to be effective on both synthetic and real world data. We demonstrate how our new approach can be used to obtain approximations to the mean, variance and correlation between two assets at a given point in time. We further show that these values can be used in the context of Modern Portfolio Theory to form profitable trading strategies when using two assets.

Key takeaways

- We develop an adapted k-means algorithm that uses the 2-Wasserstein distance metric or Maximum Mean Discrepancy, and d-dimensional data in order to identify changes in joint market regimes between assets, in particular correlation.
- We create a two-step process for finding the marginal and joint market regimes in synthetic and real data.
- Using the two-step process, we form approximations to the mean, variance and correlation which then subsequently inform profitable portfolios of pairs of stocks.

Keywords: market regimes, k-means clustering, p-Wasserstein, MMD, mean-variance optimisation
JEL Codes: C14, C15, G6, G11

1 Introduction

Clustering is a well-known and studied field in unsupervised learning. The goal of a clustering algorithm is to discover patterns and relationships between the input data that are currently unknown. Datum grouped into a particular cluster should have more similarity with data within the same group than with data in another group according to some metric [TX15]. Naturally, clustering algorithms are particularly useful when we can not immediately categorise data into groups via other means or when we are approaching a new data set with unknown characteristics. Clustering techniques are thus often employed with new time series data such as those of a financial origin. In the field of time series clustering, working with a single time series, referred to as univariate data, is often simpler than working with multiple time series, referred to as multivariate time series, in part due to lower time and memory complexity. Indeed, there is a wide body of research on clustering univariate time series [DM12] and a variety of clustering methods for time series data have been studied and employed, such as those summarised in [Lia05].

Clustering multivariate time series is a particularly interesting and difficult challenge. When applying clustering algorithms to multivariate data, we open up new areas of study. For example, suppose we wanted to form a series of risk diversified investment portfolios from the Russell 1000 Index. Forming a time series from the returns for each stock, we might then employ a clustering algorithm such as the celebrated k-means algorithm [Mac67] in order to form our portfolios [AAH$^+$17]. Often such an approach is accompanied by a dimension reduction technique such as Principal Component Analysis (PCA) [Pea01] and a variety of multivariate clustering methods are built using PCA and a well-known clustering technique [Li19] [BBRW20] [HJKS01].

Of course, instead of clustering each stock into clusters based on the entire multivariate time series, we might cluster *segments* of the multivariate time series. Such clusters might then represent certain periods where the joint dynamics or distribution of the multivariate time series have changed, perhaps due to a *market regime* change from a *bull* to a *bear* market and vice versa. This can be a particularly interesting area of study that can inform both trading strategies and risk management.

The idea of bull and bear markets however is a simplification of the general philosophy of market regimes, where segments of returns, univariate or multivariate, are split into different groups or regimes each characterised by a different underlying distribution [HI23]. Thus one need not be limited to just two market regimes. Indeed, we might instead divide market returns into a three state model: a bull regime of particularly high returns, a bear regime of particularly low returns and a 'normal' regime of somewhat low, positive returns in the middle [PRSX07]. We might otherwise subdivide the bull and bear markets into two states of bull correction and bull, and bear and bear rally respectively as in [MMS12], [GT05].

The method through which these regimes are determined and classified is referred to as the *market regime clustering problem* (MRCP) and has been the subject of rigorous study. This problem may also be viewed as a subset of studies related to *change point detection*. In such studies, one works to identify change points in financial markets, points at which the underlying regime or model driving the market returns is assumed to have changed [HNZ16] and where the number of such change points may be known or unknown [Cho07]. Certain studies work specifically on determining anomalous datum using outlier detection methods [IM17] [ACFL20].

One of the most popular regime detection methods is the Hidden Markov Model [Ham89]. In such a model, we would have an observed process, X, and an unobserved process, S. The observed process X may be the current return of a stock we are interested in while S would be the Markovian latent state or regime that it is currently following. We say that X is observed as we may actively monitor its evolution whereas S is unobserved as we have no method of directly monitoring it. Instead, we use observations of X to approximate the current state S [LR09]. One drawback of this approach is that it is not model free - we must model the latent state variable, often using a Gaussian distribution [CDL$^+$16]. Other approaches for determining regimes include directional change [CT17] and Bayesian techniques [CGOY15].

Classic unsupervised learning techniques have also been deployed more recently on the MRCP. The fuzzy c-means algorithm, a close relative of the k-means clustering algorithm, has been used to identify time series clusters in stock and sector data as in [CPS$^+$22]. These clusters were then used to train single non-linear regime models in order to forecast future stock prices. A more distinctive approach comes from clustering not the time series data itself, but the empirical measures to which each segment of the time series can be associated [ANN14]. Indeed, this will be the approach utilised in this paper which we will endeavour to explain more clearly in the coming chapters.

Many of the techniques discussed may theoretically be applied to both univariate and multivariate time series. They are, however, often beguiled by problems concerning the *curse of dimensionality*, and thus efficiency, in the multivariate case. One must also consider that the underlying distribution characterising a regime is a joint distribution in the case of multivariate time series. In the univariate case, these regimes can be identified by their mean and variance [PRSX07]. When working with multivariate time series, other characteristics, such as the correlation between individual univariate series, also play a role. Clustering based on the cross-correlation between time series has been studied before and we refer the reader to the examples listed for further discussion [EHK+17] [LOV21] [GLMT08].

1.1 Our contributions

This paper expands upon the work in [HIM21], in which an unsupervised learning algorithm is deployed to effectively cluster one-dimensional financial time series into market regimes with a high degree of success and is found to be superior to comparable algorithms, such as the Hidden Markov Model. This algorithm is a variant of the classic k-means algorithm and uses the p-Wasserstein distance as its distance metric. The algorithm is thus dubbed the WK-means algorithm.

Our main contributions are as follows. We create a novel, model-free algorithm that extends the WK-means algorithm such that it may work with data of dimension d for $d > 1$, before showing that this algorithm is effective in detecting, offline, times of market regime change and, in particular, changes in the correlation structure between two assets. We also show that this new algorithm is mutable and can be used with the Maximum Mean Discrepancy instead of the p-Wasserstein distance. We then use our findings in conjunction with [HIM21] to develop a method for determining the correct number of clusters in a univariate time series and further, to develop a profitable two-asset trading strategy based on Modern Portfolio Theory.

Section 2 will review and summarise the k-means algorithm, the Maximum Mean Discrepancy, the p-Wasserstein distance and the WK-means algorithm. Section 3 will introduce the 1-dimensional WK-means algorithm and our extension to the d-dimensional space. Section 4 will showcase our experiments using synthetic data and the ability of our algorithm to detect changes in market regimes characterised by changes in the mean, variance and correlation of the underlying probability distribution used to generate the synthetic data. Section 5 will show the effectiveness of the algorithm when using real market data while section 6 will discuss how knowledge of the newly formed clusters can help us build profitable portfolios. Finally, section 7 will conclude our work and discuss possible future research.

2 Distance metrics and clustering algorithms

We refer to changes in the mean and variance of the underlying marginal distributions of the tested assets as changes in *mean-variance regime* while changes in the correlation between the assets are referred to as changes in *correlation regime*. Collectively, we refer to changes in the underlying joint distribution as regime change, and we refer to the overall clustering problem as the *market regime clustering problem* (MRCP). In order to study the regimes, we must first define our method of choice: an adaption of the well-known k-means algorithm, wherein we replace the commonly used Euclidean distance with one of the p-Wasserstein distance or the Maximum Mean Discrepancy.

2.1 p-Wasserstein distance

The p-Wasserstein distance was first defined by Leonid Kantorovich in the context of optimisation methods [Kan60]. It is oft-associated with the optimal transport problem and, where $p = 1$, the Earth mover's distance (EMD) [Mon81].

Definition 2.1 (One-dimensional p-Wasserstein distance)**.** *Let μ and ν be probability measures on $\mathbb{R}$, such that $\mu, \nu \in \mathcal{P}_p(\mathbb{R})$, where $\mathcal{P}_p(X)$ is the set of probability measures on X with finite p^{th} moment. Suppose μ and ν have cumulative distribution functions $F_\mu(x)$ and $F_\nu(x)$ respectively. Then the p^{th} Wasserstein distance is defined as*

$$\mathcal{W}_p(\mu, \nu) := \left(\int_0^1 |F_\mu^{-1}(q) - F_\nu^{-1}(q)|^p dq \right)^{\frac{1}{p}}$$

The multi-dimensional definition is similar (B.5). It has been shown that the p-Wasserstein distance is indeed a metric, and we therefore can in fact use the p-Wasserstein distance as our distance metric in the k-means clustering algorithm on the relevant space. Furthermore, the concept of an average or central measure amongst a set of measures has a natural form when using the p-Wasserstein distance. This is called the Wasserstein barycentre (B.6).

2.2 Maximum Mean Discrepancy

The Maximum Mean Discrepancy is an integrable probability metric, first proposed in [BGR+12].

Definition 2.2 (Maximum Mean Discrepancy (MMD), [BGR+12] (Definition 2, page 726))**.** *Let $\mathcal{F}$ be a class of functions $f : \mathcal{X} \to \mathbb{R}$ and let X and Y be random variables defined on a topological space $\mathcal{X}$. Let μ and ν be the Borel probability measures of X and Y respectively and suppose that x and y are independent samples drawn from X and Y. Then the Maximum Mean Discrepancy (MMD) between μ and ν is defined as*

$$MMD[\mathcal{F}, \mu, \nu] := \sup_{f \in \mathcal{F}} \Bigg(\mathbb{E}_\mu[f(x)] - \mathbb{E}_\nu[f(y)] \Bigg).$$

If $x = (x_1, \dots, x_n)$ and $y = (y_1, \dots, y_m)$ where $x_i \sim \mu$ and $y_j \sim \nu$, then a biased empirical estimate of the MMD is given by

$$MMD_b[\mathcal{F}, x, y] := \sup_{f \in \mathcal{F}} \left[\frac{1}{n} \sum_{i=1}^{n} f(x_i) - \frac{1}{m} \sum_{j=1}^{m} f(y_j) \right].$$

An important aspect to the MMD is the class of functions $\mathcal{F}$ over which it is defined. Although other candidates exist, kernel methods are often used to define $\mathcal{F}$. In [BGR+12] the unit ball in a reproducing kernel Hilbert space (RKHS), defined as a Hilbert space $\mathcal{H}$ (B.10) and an associated reproducing kernel $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ (B.11), is suggested for $\mathcal{F}$. When defined on a RKHS, if the choice of RKHS is universal (B.12), then the MMD can be shown to be a metric when defined on a compact space $\mathcal{X}$.

In the case of $\mathcal{X}$ being non-compact, then it has been shown that the MMD is still a metric if the kernel κ is a characteristic kernel [FGSS08] (B.13). The Gaussian kernel defined as

$$\kappa_G : \mathbb{R}^d \times \mathbb{R}^d \to [0, \infty], \quad \kappa_G(x, y) = \exp(-\|x - y\|^2_{\mathbb{R}^d} / 2\sigma^2),$$

for (x, y) in $\mathbb{R}^d \times \mathbb{R}^d$, is a characteristic kernel on the set of Borel measures on $\mathcal{X}$ [FGSS08]. We will use either the MMD with $\mathcal{F} = (\mathcal{H}, \kappa_G)$ or the p-Wasserstein distance as the distance metric when clustering.

2.3 The WK-means algorithm

In this section we explain how we can combine the k-means algorithm and p-Wasserstein distance in order to yield the WK-means algorithm as in [HIM21]. Indeed, in [HIM21], the authors argue that the p-Wasserstein distance is the natural choice when clustering probability measures $\mathcal{P}_p(\mathbb{R})$ defined on the metric space (X, D), in particular for the univariate case of dimension $d = 1$ and indeed, they show the WK-means algorithm to be superior to the classic Hidden Markov Model.

We note that the goal of the market regime clustering problem (MRCP) can be stated as the task of classifying segments of return series $(r_i)_{i \geq 0}$ where

$$r_i = (r_i^1, \dots, r_i^n),$$

for $n \in \mathbb{N}$ and $r_i^j \in \mathbb{R}^d$ for $j \in \{1, \dots, n\}$, where $d \in \mathbb{Z}_+$. We note as well that for any vector $r_i \in \mathbb{R}^n$, we may form an empirical measure $\mu_{r_i} = \frac{1}{n} \sum_{j=1}^{n} \delta_{r_i^j}$, using the dirac delta function δ. Therefore, we may rewrite the goal of the MRCP as being equivalent to assigning labels to empirical probability measures $\mu \in \mathcal{P}_p(\mathbb{R}^d)$, where $\mathcal{P}_p(\mathbb{R}^d)$ is the set of probability measures on $\mathbb{R}^d$ with finite p^{th} moment.

In practice we wish to take potentially overlapping segments of returns data for d assets and gather the empirical measures associated to these segments together into distinct clusters, each defined by an underlying distribution, which we then refer to as market regimes.

We take $\mathcal{X} = \mathbb{R}^d$ for some $d \in \mathbb{Z}_+$ and fix $N \in \mathbb{N}$. We work with the log-returns of elements of the form $S = (s_0, \ldots, s_N) \in \mathcal{S}(\mathbb{R}^d)$, which are price paths of d financial assets. For $S \in \mathcal{S}(\mathbb{R}^d)$, we define the vector of log-returns r^S associated to S as

$$r_i^S = \log(s_{i+1}) - \log(s_i),$$

for $0 \leq i \leq N-1$ and thus $r^S \in S(\mathbb{R}^d)$.

Using definition B.1 we define a lift l that will take our original returns data of length N, and divide it into M segments of length h_1 using an overlap parameter of length h_2, where $h_1,\ h_2 \in \mathbb{N}$, $M := \lfloor \frac{N}{h_1 - h_2} \rfloor$ and $\ell := \ell_{h_1,h_2}$ such that

$$\ell^i(r^S) = (r^S_{(h_1-h_2)(i-1)}, \ldots, r^S_{h_1+(h_1-h_2)(i-1)}),$$

for $i = 1, \ldots, M$. We now have M segments of length h_1 for which h_2 returns overlap between each adjacent segment. As discussed, any segment of returns data $r_i \in \ell(r^S)$ with $r_i = (r_i^1, \ldots, r_i^{h_1})$ can be associated to an empirical measure $\mu_i = \frac{1}{n}\sum_{j=1}^n \delta_{r_i^j}$. Therefore we may define a family of measures

$$\mathcal{K} := \{(\mu_1, \ldots, \mu_M) : \mu_i \in \mathcal{P}_p(\mathbb{R}^d), i = 1, \ldots, M\}. \tag{1}$$

We now have a set of measures associated to our raw multivariate time series of returns data and it is this set of measures we wish to cluster into regimes. In order to do so, we make certain adaptions to the traditional k-means algorithm.

Step 1: Choose k initial centroids

The initial step of choosing k initial centroids, $\bar{\mu} := \{\bar{\mu}_j\}_{1 \leq j \leq k}$ is similar to the standard k-means algorithm. If using the naive k-means clustering algorithm, then we choose uniformly at random k measures from our family of measures $\mathcal{K}$ to be our initial centroids. If using the k-means++ algorithm [AV07], then the distance metric used in the weighted probability distribution, D, is the p-Wasserstein distance, $\mathcal{W}_p$, while all other steps remain the same.

Step 2: Define the clusters

We then calculate the distance between each point μ_i and each centroid in $\bar{\mu}$. Point μ_i is then assigned to the j^{th} cluster for which its distance to the j^{th} centroid is smallest. More formally, we define the set of clusters such that for $j = 1, \ldots, k$

$$\mathcal{C}_j := \{\mu_i \in \mathcal{K} : \underset{l=1,\ldots,k}{\arg\min}\, \mathcal{W}_p(\mu_i, \bar{\mu}_l) = j\}.$$

Step 3: Define the new centroids

Finally, we define a new centroid for each cluster using the p-Wasserstein barycentre (B.6). This is prudent as the barycentre of a family of measures $\{\mu_i\}_{i \geq 1}$ is the measure $\bar{\mu}$ which minimises the within-cluster variation (B.2). By minimising the within cluster variation, the total-cluster variation is thus minimised (B.3).

The final adaption is to the stopping rule (B.4) and its loss function l. The adaption is quite natural wherein, instead of using the squared Euclidean distance to measure the distance between the old and new cluster centres, we simply replace this with the p-Wasserstein distance. We then stop the algorithm once the difference in the old and new centroid values after each iteration falls below a preset parameter ϵ.

Definition 2.3 (WK-means algorithm, [HIM21] (Definition 2.7, page 11))**.** *Let $\mathcal{K} \subset \mathcal{P}_p(\mathbb{R}^d)$ be a family of measures with finite p^{th} moment. We refer to the k-means clustering algorithm on $(\mathcal{P}_p(\mathbb{R}^d), \mathcal{W}_p)$, with aggregation method given by the p-Wasserstein barycentre and loss function B.4 as the Wasserstein k-means algorithm or WK-means algorithm.*

3 The WK-means and MMDK-means algorithms

In this section, we show how the WK-means algorithm is implemented in practice for the generic d-dimensional case ($d > 1$). Implementation of the univariate ($d = 1$) case can be found in [HIM21]. We also show how we may use the Maximum Mean Discrepancy in order to form a similar algorithm which we dub the d-dimensional MMDK-means algorithm.

3.1 The d-dimensional WK-means algorithm

3.1.1 Background

Computation of the p-Wasserstein distance in $\mathbb{R}^d$, $d > 1$, is known to be computationally demanding for h_1 or M large [BDPR12], with evaluation between multi-dimensional measures often numerically intractable [BKN+19]. In addition, the Wasserstein barycentre in higher dimensions is often difficult to compute if not completely infeasible. Many algorithms have been proposed to help overcome this problem. These include a mixture of fixed support methods, where the atoms of the barycentre are fixed in advance and their optimal weightings need only be found, and free support methods, where the atoms and weightings are computed. A number of methods rely on some form of entropy regularisation [CGJ20], [CD14] while others use tools such as Bregman projections [LWWY17], stochastic gradient descent methods [BDPR12] and interior point methods [GWXY19]. Sliced Wasserstien distances are also often used as a substitute for the true p-Wasserstein distance.

3.1.2 Approach

In our approach we build on the framework of the WK-means algorithm as presented in section 2 but we use different methods to that of the uni-dimensional implementation. We work on the space $(\mathcal{P}_2(\mathbb{R}^d), \mathcal{W}_2)$, $d > 1$. We recall that each empirical measure μ_i is formed from a segment of returns data $r_i \in \ell(r^S)$ with $r_i = (r_i^1, \dots, r_i^{h_1})$ and hence μ_i is in fact a discrete density distribution in $\mathbb{R}^d$. Suppose we take two measures μ_i and μ_j from $\mathcal{K}$ (1) formed from segments $r_i \in \ell(r^S)$ with $r_i = (r_i^1, \dots, r_i^{h_1})$ and $r_j \in \ell(r^S)$ with $r_j = (r_j^1, \dots, r_j^{h_1})$ respectively, for $i, j \in 1, \dots, M$.

We describe the elements of r_i and r_j, and thus the atoms of μ_i and μ_j respectively, as being point clouds X and Y in $\mathbb{R}^d$ of size h_1. As discussed in [BDPR12] we may consider the 2-Wasserstein distance between X and Y as

$$\mathcal{W}_2(\mu_i, \mu_j)^2 = \mathcal{W}_2(\mathrm{X}, \mathrm{Y})^2 = \min_{\pi \in \Pi_{h_1}} \sum_{i \in I} \|\mathrm{X}_i - \mathrm{Y}_{\pi(i)}\|^2,$$

where Π_{h_1} is the set of all permutations of h_1 elements and $I = \{1, \dots, h_1\}$. This distance is then found by computing the optimal assignment $i \to \pi^*(i)$ that minimises $\mathcal{W}_2(\mathrm{X}, \mathrm{Y})^2$. The problem can be viewed as a linear programming problem such that we aim to find

$$\mathcal{W}_2(\mathrm{X}, \mathrm{Y})^2 = \min_{B \in \mathcal{B}_{h_1}} \sum_{i,j \in I^2} B_{i,j} \|\mathrm{X}_i - \mathrm{Y}_j\|^2,$$

where $\mathcal{B}_{h_1}$ is the set of bistochastic matrices - that is, nonegative matrices with rows and columns summing to 1. This can be solved using the Hungarian algorithm [Kuh55] which is worst-case $\mathcal{O}(h_1^3)$, or $\mathcal{O}(h_1^{2.5} log(h_1))$ using dedicated solvers [BDM12].

The computational complexity of these algorithms typically makes this problem unattractive to solve for large h_1 or M. However, in python, use of the cython *cdist* function from the *scipy.spatial* package when computing the euclidean distance and the cython *linear_sum_assignment* function from the *scipy.optimize* give an acceptable speed of computation for our experiments.

As discussed in section 3.1.1, computation of the Wasserstein barycentre can be difficult if not infeasible. In order to reduce the computational complexity, we restrict the possible atoms of the barycentre to those of the measures it is clustering. If the data is particularly sparse, this may lead to incorrect barycentres but assuming we work with a sufficient size of returns data, we reason that it should act as a sufficient approximation. Having restricted our search space, we then form an $M \times M$ distance matrix W, where the i^{th}, j^{th} entry is the 2-Wasserstein distance between the i^{th} and j^{th} measures.

Such a matrix is naturally symmetric which can be used to further cut down our computation time. We also know that $\mathcal{W}_{ij} = 0$ where $i = j$, and thus $\mathrm{W}_{ii} = 0$ for $i = 1, \dots, M$. Using the *cdist* and *linear_sum_assignment* functions to find the 2-Wasserstein distance, $\mathcal{W}_2(\mu_i, \mu_j)$, as discussed, we form our distance matrix, which is $\mathcal{O}(\frac{M(M-1)}{2})$, and proceed to sum the rows. The row with the minimum sum is judged to be the entry of the measure which minimises the distance to all other measures in the cluster and is hence the 2-Wasserstein barycentre. Exhibit 1 shows a flow diagram of the described steps.

$$\mathrm{W}_{ij} := \mathcal{W}_2(\mu_i, \mu_j) \rightarrow \begin{pmatrix} \mathrm{W}_{11} & \mathrm{W}_{12} & \cdots & \mathrm{W}_{1M} \\ \mathrm{W}_{21} & \mathrm{W}_{22} & \cdots & \mathrm{W}_{2M} \\ \vdots & \vdots & \cdots & \vdots \\ \mathrm{W}_{M1} & \mathrm{W}_{M2} & \cdots & \mathrm{W}_{MM} \end{pmatrix} \rightarrow \begin{pmatrix} \sum_j \mathrm{W}_{1j} \\ \sum_j \mathrm{W}_{2j} \\ \vdots \\ \sum_j \mathrm{W}_{Mj} \end{pmatrix} \rightarrow$$

$$q := \arg\min_i \Big\{ \sum_j \mathrm{W}_{ij} \Big\},$$

$$\bar{\mu} := \mu_q$$

Exhibit 1: Barycentre computation flow diagram

3.2 The d-dimensional MMDK-means algorithm

Our approach when using the MMD as the distance metric is analogous to that of the d-dimensional WK-means algorithm. An important difference between the two is that the MMD is often easier to compute in higher dimensions. When working with discrete distributions, we compute an unbiased estimate of the squared MMD (B.14).

Our choice of kernel function is the Gaussian kernel (2.2), a choice which requires a bandwidth parameter σ. The optimum choice of kernel size is an area of active research and it remains a heuristic. We found that averaging across a number of choices of bandwidth parameter on the scale of the constituent data worked well in our experiments, as opposed to instances of our bandwidth parameter being much larger than the data, which led to poor results. Finally, we follow the same framework as the d-dimensional WK-means algorithm and define the barycentre to be the measure in a given cluster which minimises the distance to all other measures.

Remark 3.1. *Going forward, we will refer to our new algorithms as the 2-d WK-means algorithm and the 2-d MMDK-means algorithm respectively, and to the uni-dimensional WK-means algorithm as either the uni-d 1-WK-means algorithm or the uni-d 2-WK-means algorithm. We do this in order to highlight both the dimensionality of the data we run each algorithm on, and the choice of p used for the p-Wasserstein distance in each uni-dimensional algorithm.*

4 Synthetic data experiments

In this section, we test the 2-d WK-means and 2-d MMDK-means algorithms on synthetic data using two time series generated from either a classic geometric Brownian motion (GBM) or a Merton jump diffusion (MJD) process.

The synthetic data generated from the GBM and MJD processes have a shared schema. Given $T \in \mathbb{N}$ and a time interval $[0, T]$, we define a mesh with increments that roughly represent one market hour. Taking $n := 252 \times 7$ to be the number of market hours in a market year, we set

$$\Delta := \left\{ \left[\frac{i-1}{n}, \frac{i}{n} \right] : i = 1, 2, \ldots, n \times T \right\}.$$

Therefore, T can be thought of as the number of market years and we will take $T = 20$ for our experiments. When generating our data using this mesh, the majority of points will fall into what we designate as a 'normal' regime with a particular underlying distribution.

At certain points this regime will change. Our goal is to study changes in the joint distribution of our assets, and so these changes will be changes of the mean and variance of the synthetic assets, of their correlation structure or a mixture of both.

Using the mesh structure above, we generated two synthetic price series S_1 and S_2, and calculated the log-returns $r^S \in \mathcal{S}(\mathbb{R}^2)$ where $S := (S_1, S_2)$. The log-returns r^S have a joint distribution f_{S_1,S_2} and marginal distributions given by f_{S_1} and f_{S_2}. In line with B.1, we created a lifted stream of data $l(r^S) \in \mathcal{S}(\mathcal{S}(\mathbb{R}^2))$, and subsequently a family of empirical measures $\mathcal{K}$, by segmenting our return series into M segments.

In order to create the lifted stream, we first choose values for h_1 and h_2. In [HIM21], when testing the original univariate WK-means algorithm, the authors choose $h_1 = 35$ and $h_2 = 28$ given $h_1 = 35$ corresponds to approximately one market week in market hours with h_2 being one market day short of that. Moving forward, we will take $h_1 = 35$ and $h_2 = 28$ but we will further discuss the values of h_1 and h_2 for the 2-d WK-means and 2-d MMDK-means algorithms in the context of real data in section 5.

We ran the uni-d p-WK-means and 2-d algorithms over the lifted stream of data $l(r^S)$, returning k centroids $\{\bar{\mu}_i\}_{i=1,\dots,k}$ and clusters $\{\mathcal{C}_j\}_{j=1,\dots,k}$. We display our results primarily using a combination of three different plots. Any given plot will consist of a number of colours, each belonging to a different cluster $\mathcal{C}_j$, for $j = 1, \dots, k$. Should our algorithms be effective in the given experimental environment, we would expect elements placed in the same cluster, and thus designated with the same colour, to appear in well-defined groups.

The first type of plot is referred to as a mean-variance plot. Given each empirical joint distribution $\mu \in \mathcal{K}$ is formed from a multivariate time series and is a discrete distribution, it may be written in the form given in exhibit 2.

$$\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1h_1} \\ r_{21} & r_{22} & \cdots & r_{2h_1} \end{pmatrix}$$

Exhibit 2: Matrix form of a $2 \times h_1$ empirical distribution μ.

Each row is thus a marginal distribution of the joint distribution and each marginal distribution will have an associated mean and variance. Therefore each marginal distribution can be projected onto $\mathbb{R}^2$ via the mapping

$$M_p : \mathcal{P}_p(\mathbb{R}) \to \mathbb{R}^2,$$

$$\mu_{f_{S_i}} \to \left(\sqrt{\mathbb{V}(\mu_{f_{S_i}})}, \mathbb{E}(\mu_{f_{S_i}}) \right),$$

for $i = 1, 2$, which is simply a scatter plot of each measure in mean-variance space. Each point is then colored according to its cluster membership. Using this mean-variance plot allows us to visualise how well our algorithms cluster along each of the marginal distributions f_{S_1} and f_{S_2}.

The second plot we make use of is a plot of the log-returns of each asset which aids us in visualising the correlation between the points in each cluster. We identify each cluster, and its points therein, with a color.

Finally, our third plot will be a colored time series plot. In a similar manner to the second plot, we partition the two price series S_1, S_2 such that each partition is colored according to its centroid membership. The centroid membership is the first centroid to which a particular price can be associated. We will also apply a colored shade to the grid points indicating the correct periods of regime change which should allow us to visually ascertain how well the algorithm is picking up periods of regime change.

In order to numerically evaluate the accuracy of our clustering algorithm, we follow the approach of [HIM21] and consider three scores of accuracy: total accuracy, accuracy during the normal regime (regime-off) and accuracy during periods of regime change (regime-on). Confidence intervals of 95% are calculated by assuming the CLT holds such that $\text{CI} = 1.96 * \frac{\sigma}{\sqrt{n}}$, where σ is the standard deviation of the scores and n is the number of trials.

Each joint regime is a combination of the mean-variance regime and the correlation regime. The mean-variance regime of a particular joint regime is dictated by the mean-variance regime of each underlying asset. This is a combination of a lower variance/higher mean (bull) and higher variance/lower mean (bear) market for each asset. The correlation regime is also one of two. We refer to the correlation regime as being either the ‘normal’ correlation regime or the ‘abnormal’ correlation regime. We refer to the combination of joint Regime (JR) = (Bull, Bull, Normal) as being the standard regime. The number of joint regimes in each of our experiments will dictate the number of clusters k we test for.

4.1 Merton jump diffusion process

In this section we generate two synthetic price series modelled as Merton jump diffusion processes. A similar analysis was conducted using synthetic prices series generated from a Geometric Brownian motion and results will be discussed. Detailed discussion of the price generation process is included in appendix C.

We take $\mathcal{M}(\theta)$ to be a family of models indexed by a parameter set $\theta \subset \mathbb{R}^{10}$. For each regime, the parameter set θ is characterised by the means (μ^i) and standard deviations (σ^i) per unit time of the assets used to generate the geometric Brownian motion, the intensity (λ) of the Poisson process, the means (γ^i) and standard deviations (δ^i) per unit time of the jump process, and the instantaneous correlation (ρ) of both assets for $i = 1, 2$ such that $\theta = (\mu^1, \mu^2, \sigma^1, \sigma^2, \lambda, \gamma^1, \gamma^2, \delta^1, \delta^2, \rho)$.

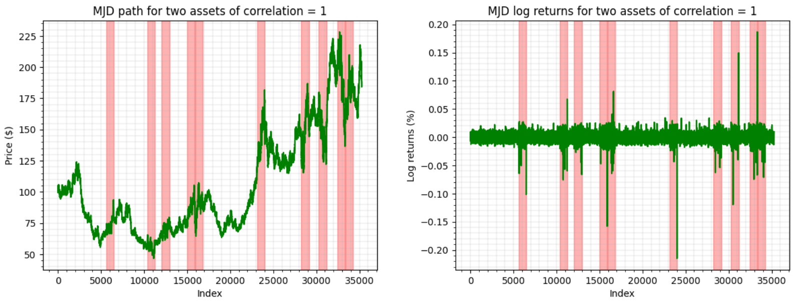
4.1.1 Fixed correlation regime

In this experiment we tested how well the 2-d algorithms picked up a change in mean-variance regime characterised by a change in the mean and variance of the generator MJD. We fixed the correlation ρ for values of $\rho = -1, -0.5, 0, 0.5$, and 1 and compared the results of our novel algorithms to those of the univariate algorithm tested on each asset individually using the 1- and 2-Wasserstein distances. We generated Merton jump diffusion paths for each asset using the parameter sets

$$\theta_0 = (0.05, 0.05, 0.2, 0.2, 5, 0.02, 0.02, 0.005, 0.005, \rho), \quad \text{and}$$
$$\theta_1 = (-0.05, -0.05, 0.35, 0.35, 10, -0.04, -0.04, 0.04, 0.04, \rho)$$

such that θ_0 (JR$_0$) and θ_1 (JR$_1$) correspond to a bull and bear market regime respectively. Once again, we took $S_0^i = 100$ for $i = 1, 2$. In these experiments we always had exactly two different correlation regimes and thus we took $k = 2$ throughout.

We began by fixing our assets such that both would undergo a regime change at the same time. Thus we had two combinations of joint regime JR$_0$ = (Bull, Bull, Normal) and JR$_1$ = (Bear, Bear, Normal) and took $k = 2$. We then fixed $\rho = 1$. Approximately one quarter of the path, split equally to ten randomly generated points on the time axis, was generated using the JR$_1$ parameter set with the rest generated from the JR$_0$ set. Exhibit 3(a) shows an example of such a MJD path with the regime change periods highlighted on the grid in red. Alongside, exhibit 3(b) shows the log-returns associated to this example path, again with the regime changes highlighted.



(a) MJD price paths with $\rho = 1$, regime changes highlighted.

(b) MJD log-returns with $\rho = 1$, regime changes highlighted.

Exhibit 3: MJD synthetic price paths with $\rho = 1$, and their associated log-returns.

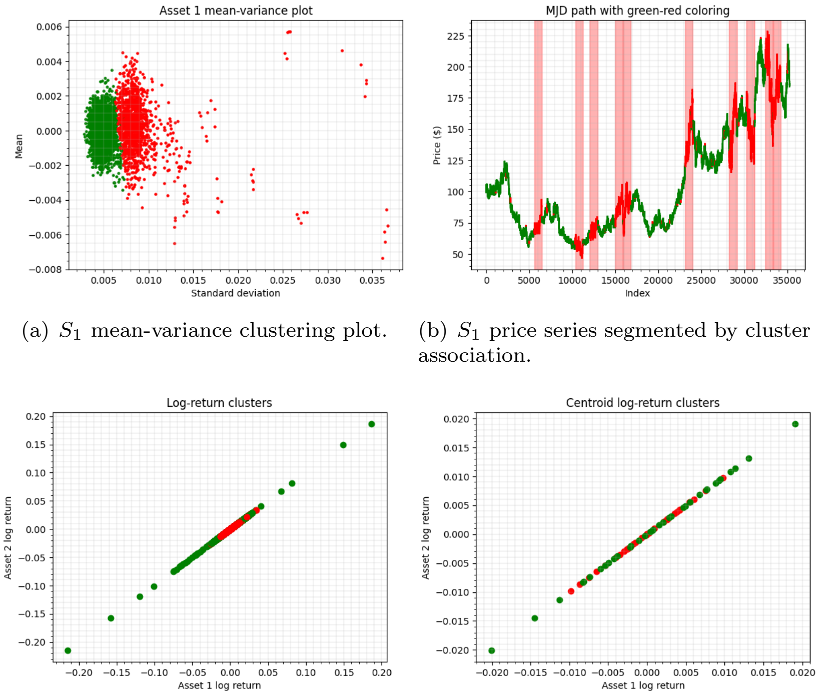
We ran the algorithms on the same simulated path. We report the accuracy scores for each algorithm in exhibit 4 for a total of $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	94.04% ± 1.90%	81.46% ± 8.05%	98.00% ± 0.15%
Uni-d 1-WK-means	93.06% ± 2.45%	85.37% ± 8.38%	95.40% ± 2.33%
Uni-d 2-WK-means	94.63% ± 1.88%	81.80% ± 7.73%	98.68% ± 0.07%
2-d MMDK-means	90.82% ± 2.12%	93.97% ± 1.00%	89.56% ± 3.00%

Exhibit 4: Accuracy scores with 95% CI, MJD synthetic path with simultaneous mean-variance regimes and $\rho = 1$, $n = 50$ runs.

All four algorithms exhibit strong regime-on, regime-off and total accuracy scores. The 2-d MMDK-means algorithm, in particular, returns an impressive regime-on accuracy score indicating that it is very effective at picking up changes in the mean and variance of our underlying assets. We note that the 2-d WK-means, and 2-WK-means algorithms exhibit lower regime-on accuracy scores when compared to the geometric Brownian motion case. This is not unreasonable given both the 2-d WK-means and the uni-d 2-WK-means algorithms use the 2-Wasserstein distance and thus may be more susceptible to extreme outliers that would occur in a Merton jump diffusion process.

We visualise the clustering generated by the 2-d WK-means using our three types of plot in exhibit 5. We see that the algorithm has done a good job of picking up the two distinct regimes and we note that the red cluster centroid has higher variance than that of the green cluster centroid, as we would expect from our parameter set. Exhibit 5(b), which shows the simulated historical price series for S_1 coloured according to each segment's associated cluster, also reinforces the strong detection properties of our algorithms. As expected, the correlation amongst all values in both clusters is approximately one as seen in exhibits 5(c) and 5(d).



(a) S_1 mean-variance clustering plot.

(b) S_1 price series segmented by cluster association.

(c) Clusters 1 and 2 log-returns plot with cluster colours.

(d) Centroids 1 and 2 log-returns plot with cluster colors.

Exhibit 5: MJD, $\rho = 1$, example mean-variance, correlation and price series plots.

Having tested the case of $\rho = 1$, we then subsequently varied the correlation to values of $\rho \in \{-1, -0.5, 0, 0.5\}$ with results for each test included in appendix C.3.1. The accuracy scores for the 2-d WK-means algorithm tend to be lower than the analogous GBM case but still strong. The 2-d MMDK-means algorithm, on the other hand, outperforms the 2-d WK-means algorithm, returning very strong scores in each test.

4.1.2 Fixed mean-variance regime

In this series of experiments we tested how well the 2-d algorithms picked up a change in correlation regime, characterised by a change in the correlation ρ between the two assets. In order to do so, we fixed the mean-variance regime and varied the correlation between two values ρ_0 and ρ_1. We generated Merton jump diffusion paths for each asset using the parameter sets

$$\theta_0 = (0.05, 0.05, 0.2, 0.2, 5, 0.02, 0.02, 0.005, 0.005, \rho_0), \quad \text{and}$$

$$\theta_1 = (0.05, 0.05, 0.2, 0.2, 5, 0.02, 0.02, 0.005, 0.005, \rho_1)$$

for the normal and abnormal regimes respectively, and once again we took $S_0^i = 100$ for $i = 1, 2$ and $k = 2$. Formally, our regimes were $\text{JR}_0 =$ (Bull, Bull, Normal), and $\text{JR}_1 =$ (Bull, Bull, Abnormal) and as in the fixed correlation experiments, approximately one quarter of the path, split equally to ten randomly generated points on the time axis, was generated using the JR_1 parameter set with the rest generated from the JR_0 set.

We report the accuracy scores for $\rho = (\rho_0, \rho_1) = (0, 1)$ in exhibit 6 for a total of $n = 50$ runs. As in the GBM case, both 2-d algorithms perform very well in picking up the regime changes.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	95.72% ± 3.35%	92.02% ± 6.30%	96.73% ± 3.69%
2-d MMDK-means	96.02% ± 2.87%	96.91% ± 2.11%	95.50% ± 3.26%

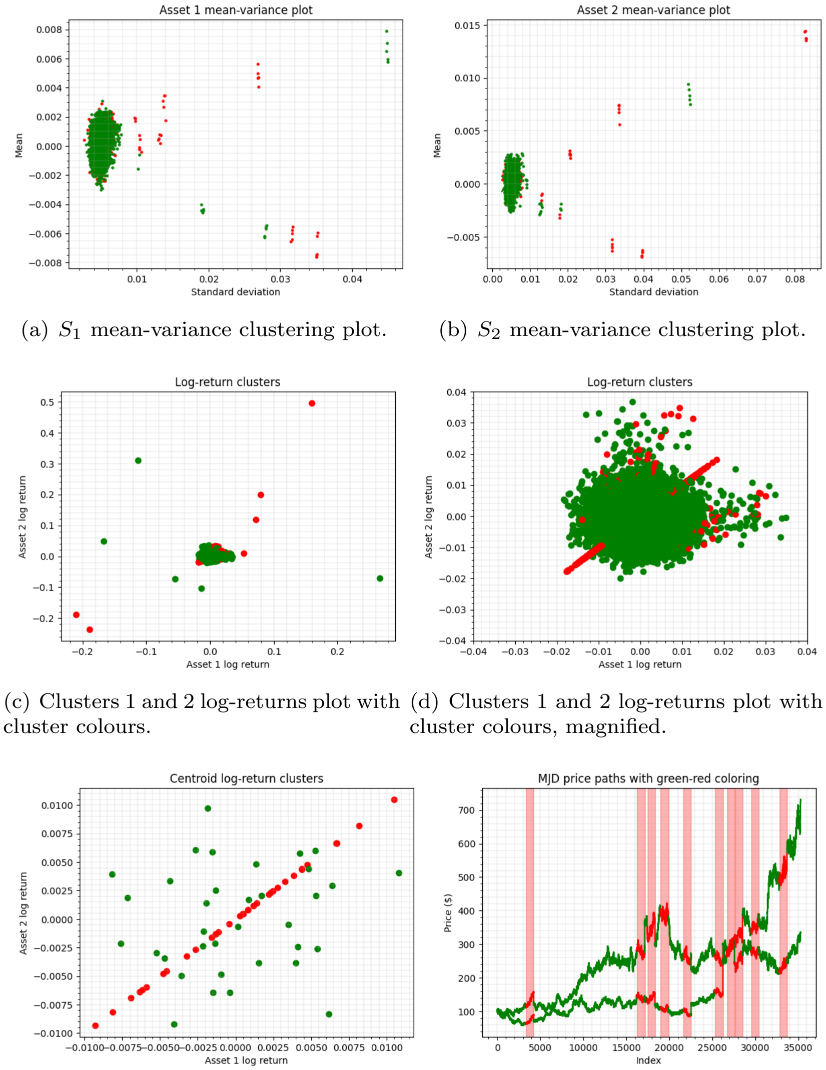
Exhibit 6: Accuracy scores with 95% CI, MJD synthetic path with simultaneous correlation regimes and $\rho_0 = 0$, $\rho_1 = 1$, $n = 50$ runs.

Exhibits 7(a) and 7(b) show our mean-variance plot for each asset, S_1 and S_2. As we would expect, there are no discernible clusters when plotting in mean-variance space due to the fixed mean-variance regime. In contrast, our correlation plots for each cluster's log-returns in exhibit 7(d) and each centroid's log-returns in 7(e) show that our algorithm has found clear relationships in the data. The second cluster's log-returns and the associated centroid's atoms clearly fall on the line $x = y$ while those of the first cluster are spread out evenly around the centre. Finally, exhibit 7(f) gives a visual demonstration of the strength of our algorithms in picking up changes in correlation through time. These results were similar but stronger in the GBM case.

Having tested the case of $\rho = (0, 1)$, we then subsequently varied the values of ρ_0 and ρ_1 between -1 and 1. Again, the results remain strong and are included for each supplementary test in appendix C.3.2. Taking $\rho = (0.5, 1)$ we observed similar results to the case of $\rho = (0, 1)$ for the 2-d WK-means algorithm. The 2-d MMDK-means algorithm, however, showed a marked decrease in regime-on accuracy score. We report the accuracy scores for each algorithm in exhibit 8 for a total of $n = 50$ runs. Exhibit 9 includes the three plots associated with $\rho = (0.5, 1)$.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	80.38% ± 5.94%	85.58% ± 7.41%	78.46% ± 7.50%
2-d MMDK-means	78.04% ± 8.11%	68.17% ± 11.91%	81.14% ± 6.88%

Exhibit 8: Accuracy scores with 95% CI, MJD synthetic path with simultaneous correlation regimes and $\rho_0 = 0.5$, $\rho_1 = 1$, $n = 50$ runs.



(a) S_1 mean-variance clustering plot.

(b) S_2 mean-variance clustering plot.

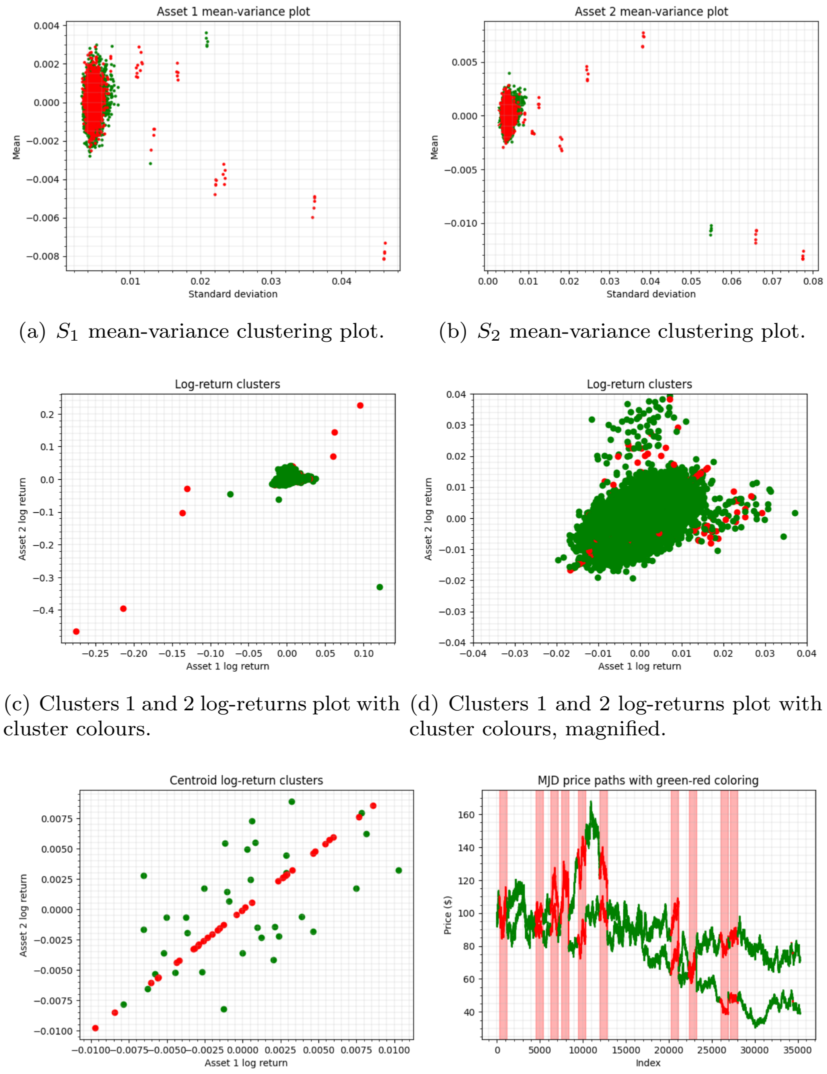
(c) Clusters 1 and 2 log-returns plot with cluster colours.

(d) Clusters 1 and 2 log-returns plot with cluster colours, magnified.

(e) Centroids 1 and 2 log-returns plot with cluster colors.

(f) Price series segmented by cluster association.

Exhibit 7: MJD with correlation regimes, $\rho_0 = 0$ and $\rho_1 = 1$, example mean-variance, correlation and price series plots.



(a) S_1 mean-variance clustering plot.

(b) S_2 mean-variance clustering plot.

(c) Clusters 1 and 2 log-returns plot with cluster colours.

(d) Clusters 1 and 2 log-returns plot with cluster colours, magnified.

(e) Centroids 1 and 2 log-returns plot with cluster colors.

(f) Price series segmented by cluster association.

Exhibit 9: MJD with correlation regimes, $\rho_0 = 0.5$ and $\rho_1 = 1$, example mean-variance, correlation and price series plots.

4.1.3 Free correlation and mean-variance regime

Finally, we tested the algorithms while varying the mean-variance regime and correlation regime. We tested four joint regimes: JR_0 = (Bull, Bull, Normal), JR_1 = (Bull, Bull, Abnormal), JR_2 = (Bear, Bear, Normal) and JR_3 = (Bear, Bear, Abnormal). We generated paths for each asset using the parameter sets

$$\theta_0 = (0.05, 0.05, 0.2, 0.2, 5, 0.02, 0.02, 0.005, 0.005, 0),$$

$$\theta_1 = (0.05, 0.05, 0.2, 0.2, 5, 0.02, 0.02, 0.005, 0.005, 1),$$

$$\theta_2 = (-0.05, -0.05, 0.35, 0.35, 10, -0.04, -0.04, 0.04, 0.04, 0), \quad \text{and}$$

$$\theta_3 = (-0.05, -0.05, 0.35, 0.35, 10, -0.04, -0.04, 0.04, 0.04, 1).$$

We took $S_0^i = 100$ for $i = 1, 2$. One eighth of the path was generated from each of the JR_1, JR_2 and JR_3 parameter sets, each split randomly across 5 independent and non-overlapping periods, while the other five eighths was generated using the JR_0 parameter set. We ran the algorithms for $k = 4$ clusters. We report the accuracy scores for each algorithm in exhibit 10 for a total of $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	68.25% ± 4.34%	53.94% ± 5.37%	76.63% ± 5.68%
2-d MMDK-means	70.76% ± 5.12%	66.78% ± 5.20%	72.95% ± 6.26%

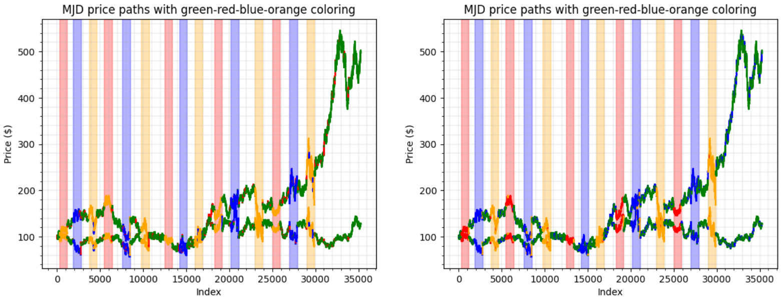
Exhibit 10: Accuracy scores with 95% CI, MJD synthetic path, $k = 4$, with correlation and mean-variance regimes, $n = 50$ runs.

Algorithm	Regime-on JR$_1$	Regime-on JR$_2$	Regime-on JR$_3$
2-d WK-means	48.52% ± 13.09%	47.11% ± 10.89%	66.18% ± 12.25%
2-d MMDK-means	31.75% ± 12.38%	76.23% ± 9.89%	92.36% ± 2.83%

Exhibit 11: Accuracy scores with 95% CI, MJD synthetic path, $k = 4$, with correlation (JR$_1$), mean-variance (JR$_2$) and joint correlation-mean-variance (JR$_3$) regimes, $n = 50$ runs.

The algorithms struggle to cluster for both regimes where one of the mean-variance or correlation regime changes (JR$_1$ and JR$_2$ respectively) with quite a wide confidence interval and low score as seen in exhibit 11. However, both the 2-d WK-means and 2-d MMDK-means algorithms are better at picking up instances of simultaneous change in both the mean-variance and correlation regimes, returning accuracy scores of above 66% and 92% respectively for JR$_3$. The 2-d MMDK-means algorithm again exceeds in detecting changes in the mean-variance regime as seen in exhibit 11 but for both algorithms the overall accuracy scores remain low as seen in exhibit 10. Exhibits 12(a) and 12(b) are examples of our price series coloured according to cluster membership for two independent runs of the 2-d WK-means algorithm. The GBM experiments yielded analogous results.

In exhibit 12(a) the algorithm has successfully identified the presence of certain changes in the market regimes but it struggles to disentangle them and assign them to independent clusters. 12(b) is similar but on this occasion the algorithm has identified each of the independent market regime changes.



(a) Price series segmented by cluster association, incorrect clustering.

(b) Price series segmented by cluster association, correct clustering.

Exhibit 12: MJD price paths with mean-variance and correlation regimes segmented by cluster association, $\rho_0 = 0$ and $\rho_1 = 1$, example plots.

5 Real data experiments

In this section we consider applications of the 2-d algorithms to sets of real data taken from securities in the S&P 500. We learned in our synthetic data experiments that the 2-d WK-means and 2-d MMDK-means algorithms are effective when clustering mean-variance regimes in the marginal distributions or when clustering correlation regimes between the assets. In such experiments, our algorithms return high regime-on and regime-off accuracy scores.

However, the accuracy scores fall significantly when we attempt to cluster both mean-variance and correlation regimes in the same dataset as exhibited in exhibits 10 and 11. Thus should we apply the 2-d WK-means or 2-d MMDK-means algorithm to our real data directly, we likely will find the clustering to be weaker than we might otherwise desire.

Therefore we propose a different approach. Instead of applying one of the 2-d algorithms alone, we make use of both the uni-d 1-WK-means algorithm and one of the 2-d algorithms in a best of both worlds approach. We begin by applying the uni-d 1-WK-means algorithm to the data in order to remove the effects of the marginal distribution on each asset and subsequently apply the 2-d WK-means or 2-d MMDK-means algorithm to this transformed data. This method is based on the theory of copulas.

Definition 5.1 (Copula, [Skl59] (Definition 1, page 229))**.** *Let $X = (X_1, \dots, X_d)$ be a random vector with joint CDF F and continuous marginal CDFs $F_1, \dots, F_d$. Then we say that the copula of F (or X) is the joint CDF C of the random vector $(F_1(X_1), \dots, F_d(X_d))$ on $[0,1]^d$.*

The copula of a random vector X captures the dependency structure between the marginals $X_1, \dots, X_d$ and thanks to Sklar's theorem (D.1) we know that such a copula exists.

In our synthetic experiments we characterised changes in the joint regime between two assets by their respective marginal distribution's changes in mean and variance, and their joint change in correlation. Correlation can tell us how closely related our two assets are in a linear sense but the copula structure is far richer, and it helps us model the inter-correlation dependence between assets. If we apply the marginal distribution to each respective asset then the joint distribution of our transformed data should be the copula according to Sklar's theorem. Moreover, the transformed data for each asset should be uniform (D.2).

We now have a set of tools with which to approach our real data clustering problem. Our approach when treating real data is laid out in the following steps.

Step 1: Univariate clustering and data transformation

We begin by establishing the marginal distributions of each asset using the uni-d WK-means algorithm. In order to estimate the number of clusters present in our uni-dimensional data, we propose an approach based on the Kolmogorov-Smirnov test (D.3).

We first assume that each asset has the minimum number of clusters possible, $k = 2$. If we have chosen the correct number of clusters for the given data then we would expect the uni-dimensional clustering, found using the uni-d 1-WK-means algorithm, to be effective. Furthermore, if we were to then apply the empirical cumulative distribution function of each cluster to its respective returns data, then the transformed returns data should in fact be uniform (D.2). Therefore, application of the Kolmogorov-Smirnov test at a chosen significance level should lead us to not reject the null hypothesis that our data is uniform. If the null hypothesis is in fact rejected, then subsequently we increase the number of clusters k iteratively, until the null hypothesis is not rejected whilst adjusting the significance level and p-value in line with the Bonferroni correction.

Having established approximations to the marginal distributions for each asset, we then proceed to apply the empirical cumulative distribution function of our approximations to the data in their respective clusters. We now have a new dataset of transformed returns where each value falls within [0, 1] and should follow a uniform distribution.

Step 2: Multivariate clustering

The joint distribution of our newly transformed dataset should be the copula of the original data, and so we apply the 2-d WK-means or 2-d MMDK-means algorithm to our transformed data in order to obtain correlation regimes.

5.1 Synthetic data experiments revisited

In order to validate our approach, we will briefly return to our previous synthetic data experiments, specifically to that of the free correlation and mean-variance regime clustering problems introduced in section 4.1.3.

Step 1: Univariate clustering

We first cluster each of our two assets using the uni-d 1-WK-means algorithm where we take $k = 2$. The accuracy scores are summarised in exhibit 13. Exhibit 15 shows the associated color plots.

Algorithm	Asset	Total	Regime-on	Regime-off
Uni-d 1-WK-means	1	97.71%	97.30%	97.62%
Uni-d 1-WK-means	2	97.14%	97.59%	96.77%

Exhibit 13: Accuracy scores of uni-d 1-WK-means applied to MJD synthetic paths with mean-variance and correlation regimes, $\rho_0 = 0$ and $\rho_1 = 1$.

We now apply the empirical cumulative distribution function of each cluster to their respective cluster of log-returns. Conducting the Kolmogorov-Smirnov test on each asset's transformed returns yields p-values of 1.000 and 1.000 respectively. In this instance we have the advantage of foresight and hence we know that $k = 2$ is the correct number of clusters.

Step 2: Multivariate clustering

Finally, we apply the 2-d WK-means and 2-d MMDK-means algorithms to the transformed data and display the accuracy scores in exhibit 14

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	99.46% ± 0.00%	98.69% ± 0.00%	99.49% ± 0.00%
2-d MMDK-means	99.42% ± 0.00%	99.44% ± 0.00%	99.18% ± 0.00%

Exhibit 14: Accuracy scores of 2-d WK-means and 2-d MMDK-means applied to MJD synthetic paths with mean-variance and correlation regimes, $\rho_0 = 0$ and $\rho_1 = 1$, $n = 50$ runs.

Clearly, when compared to our original results in exhibit 11, we have a substantial improvement in our regime accuracy scores and similar results were found for the GBM case (D.1). Exhibit 15 displays the associated price series colored according to cluster association. Exhibits 15(c) and 15(d) show the colored price series for assets 1 and 2 respectively with periods of mean-variance regime change highlighted in red. Similarly, exhibit 15(e) shows both segmented time series with periods of correlation regime change highlighted.

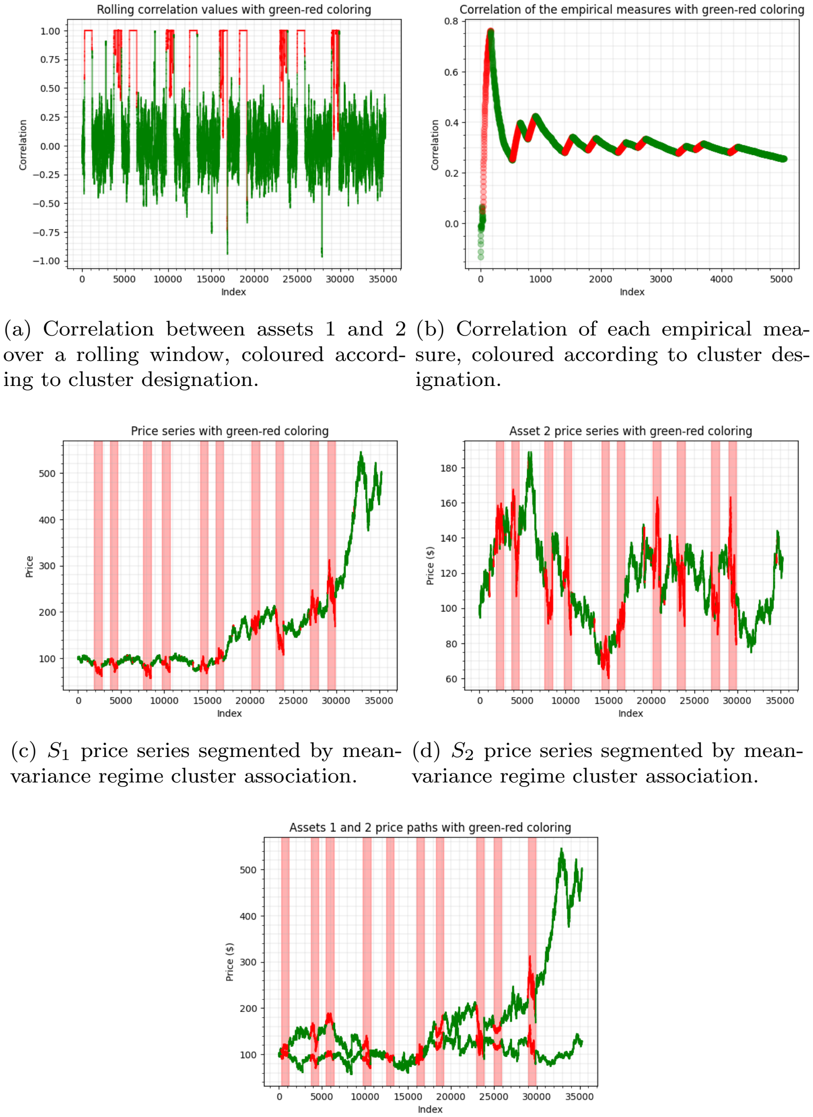
We also introduce two new plots in exhibits 15(a) and 15(b) that we may use to analyse our results. Exhibit 15(a) shows the correlation between the two assets over a rolling window of $h_1 = 35$ market hours. Each point is then assigned a color based on the cluster association of the returns it is formed from. It is noticeable that those periods of high rolling correlation are most commonly associated with cluster 2, which is colored red, as we would expect. All other points are colored green indicating that their associated returns belong to cluster 1.

Exhibit 15(b) shows the correlation of each of the empirical measures formed from our transformed returns data, using the empirical distribution of the associated cluster, and colored according to their cluster association. We note that periods of increasing correlation in the empirical measures are colored red while periods of decreasing or relatively stationary correlation are colored green. This is in line with what we might expect and further supports our claim that the 2-d WK-means algorithm has effectively clustered the empirical measures into their correct correlation regimes. These plots are reassuring when viewed in the context of our synthetic examples but they will become much more useful when we identify the correlation regimes present in real data which we do not know *a priori.*

5.2 Real data

Now that we have validated our approach on our synthetic examples, we proceed to treat real world cases. At this point we diverge slightly from our previous experiments and for simplicity we use only the 2-d WK-means algorithm going forward.

For each asset we take $(h_1, h_2) = (35, 28)$ for the univariate clustering in line with [HIM21]. Choosing values of h_1 and h_2 when using the 2-d clustering algorithm can often be more of an art than a science. The copula structure, and in particular the correlation, between different assets may change with different levels of frequency and over different lengths of time which in turn may justify different values of (h_1, h_2) depending on the pair of assets. Although we do not know the true underlying probability distributions which exist in our real data, we can attempt to test how stable the inferred correlation regimes are for different hyperparameter choices (h_1, h_2). Having tested the stability of a variety of pairs of (h_1, h_2) for each pair of assets, we took the optimal pairing and discuss the results in the following subsections.



(a) Correlation between assets 1 and 2 over a rolling window, coloured according to cluster designation.

(b) Correlation of each empirical measure, coloured according to cluster designation.

(c) S_1 price series segmented by mean-variance regime cluster association.

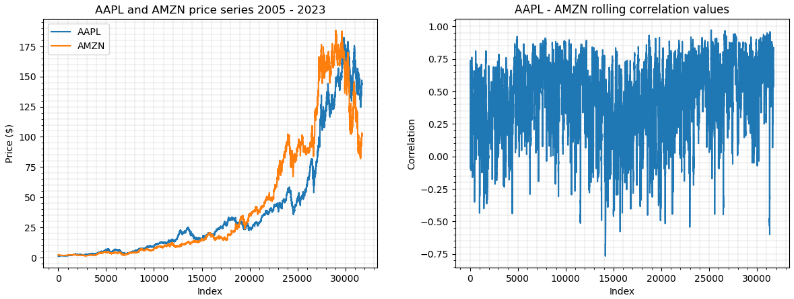
(d) S_2 price series segmented by mean-variance regime cluster association.

(e) S_1 and S_2 price series segmented by correlation regime cluster association.

Exhibit 15: Merton jump diffusion assets 1 and 2 correlation and price series segmented by cluster association.

5.2.1 AAPL - AMZN

The first pair of test assets is Apple Inc (AAPL) and Amazon Inc (AMZN). We use the hourly close prices for each stock from 2005-2023. The correlation of these assets over the entire testing period is 0.433 while the standard deviation of their rolling correlation using a window of $h_1 = 35$ hours is 28.68%. Exhibit 16(a) shows the price series evolution of AAPL and AMZN stock between 2005 and 2023 while exhibit 16(b) shows the rolling correlation over a window of 35 market hours between these two assets during that time.



(a) AAPL and AMZN price series 2005 - 2023.

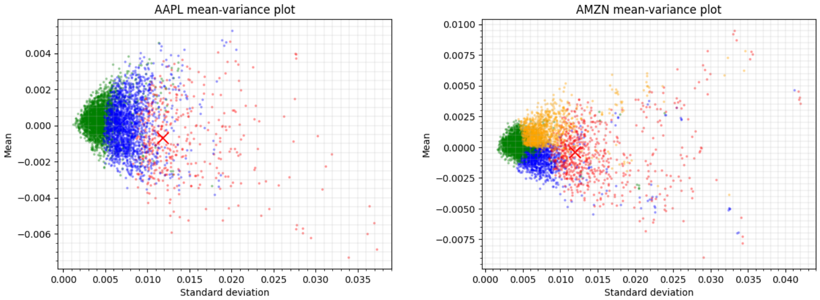
(b) AAPL and AMZN rolling correlation of returns plot using a rolling window of 35 market hours.

Exhibit 16: Price series and rolling correlation plots of AAPL and AMZN.

Step 1: Univariate clustering and data transformation

We initially cluster the empirical measures of each of our two assets using the uni-d 1-WK-means algorithm. We find that AAPL requires 3 clusters while AMZN needs 4.

Exhibits 17(a) and 17(b) show the mean-variance clustering plots of each asset. AAPL seems to exhibit three mean-variance regimes of increasing variance; a low variance regime (green), a medium variance regime (blue) and a high variance regime (red). AMZN also exhibits three regimes of increasing variance, however its medium variance regime is further split into two regimes of high (orange) and low (blue) means.



(a) AAPL mean-variance clustering plot.

(b) AMZN mean-variance clustering plot.

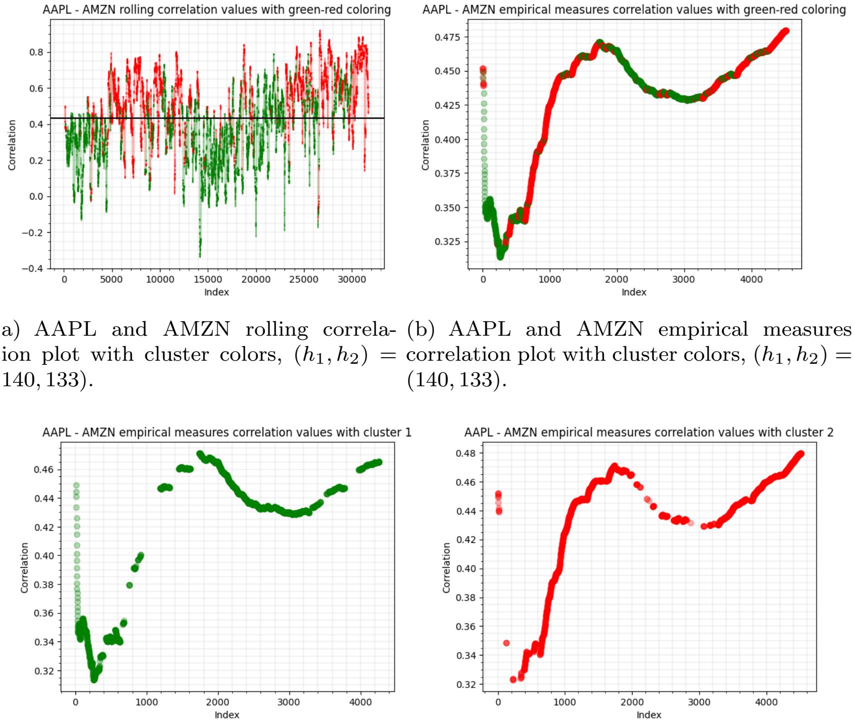
Exhibit 17: Uni-d 1-WK-means applied to AAPL and AMZN mean-variance clustering plots.

Step 2: Multivariate clustering

Now that we have removed the influence of the marginal distributions on the data, we apply the 2-d WK-means algorithm. We take $(h_1, h_2) = (140, 133)$ and as an input to our algorithm, we must decide on the number of clusters k we will run our algorithm for. The simplest possible choice is $k = 2$. These would represent one regime of higher correlation and one of a lower correlation.

Indeed, for $k = 2$, we obtain one correlation regime of high correlation where the correlation of the centroid's atoms is 0.738 and one of a lower correlation, where the correlation of the centroid's atoms is 0.373. Exhibit 18(a) shows the rolling correlation plot with cluster colors. We see from the plot that the second (red) cluster has caught instances of higher correlation above the average line, represented by the black horizontal line, while the first (green) cluster tends to fall below this line indicating that it captures periods of lower correlation.

Exhibit 18(b) shows the correlation of each of the transformed empirical measures that we have clustered with their associated cluster colors. Much like our synthetic example plot in exhibit 15(b), we note that where the correlation is increasing the second (red) cluster tends to dominate while where the correlation is decreasing, the first (green) cluster is more apparent. These trends are further reinforced in exhibits 18(c) and 18(d) which show a portion of figure 18(b) magnified and where only one cluster is presented.



(a) AAPL and AMZN rolling correlation plot with cluster colors, $(h_1, h_2) = (140, 133)$.

(b) AAPL and AMZN empirical measures correlation plot with cluster colors, $(h_1, h_2) = (140, 133)$.

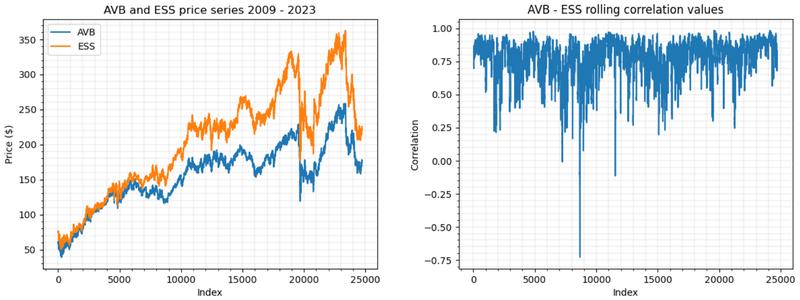
(c) AAPL and AMZN empirical measures correlation plot with cluster colors, $(h_1, h_2) = (140, 133)$ cluster 1 only.

(d) AAPL and AMZN empirical measures correlation plot with cluster colors, $(h_1, h_2) = (140, 133)$ cluster 2 only.

Exhibit 18: Correlation plots of AAPL and AMZN for 2 clusters, $(h_1, h_2) = (140, 133)$.

5.2.2 AVB - ESS

AVB and ESS constitute our second test pairing. These stocks represent AvalonBay Communities Inc (AVB) and Essex Property Trust Inc (ESS), both of which are real estate companies and thus we would expect them to be highly correlated. Indeed, using the hourly close prices for each stock from 2009-2023, the correlation of these assets over the entire testing period is 0.835 while the standard deviation of their rolling correlation using a window of $h_1 = 35$ market hours is 13.96%. This is a noticeably lower variation in the rolling correlation than that of AAPL-AMZN. Exhibit 19(a) shows the price series evolution of AVB and ESS stock between 2009 and 2023 while exhibit 19(b) shows the rolling correlation over a window of 35 market hours between these two assets during that time.



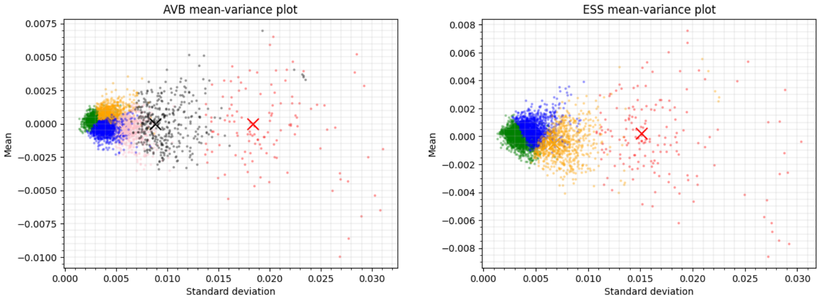
(a) AVB and ESS price series 2009 - 2023.

(b) AVB and ESS rolling correlation of returns plot using a rolling window of 35 market hours.

Exhibit 19: Price series and rolling correlation plots of AVB and ESS.

Step 1: Univariate clustering and data transformation

We initially cluster the empirical measures of each of the two assets using the uni-d 1-WK-means algorithm. After testing, we find that in the case of AVB and ESS, we require 6 and 4 clusters respectively. Exhibits 20(a) and 20(b) show the mean-variance clustering plots of each asset.



(a) AVB mean-variance clustering plot.

(b) ESS mean-variance clustering plot.

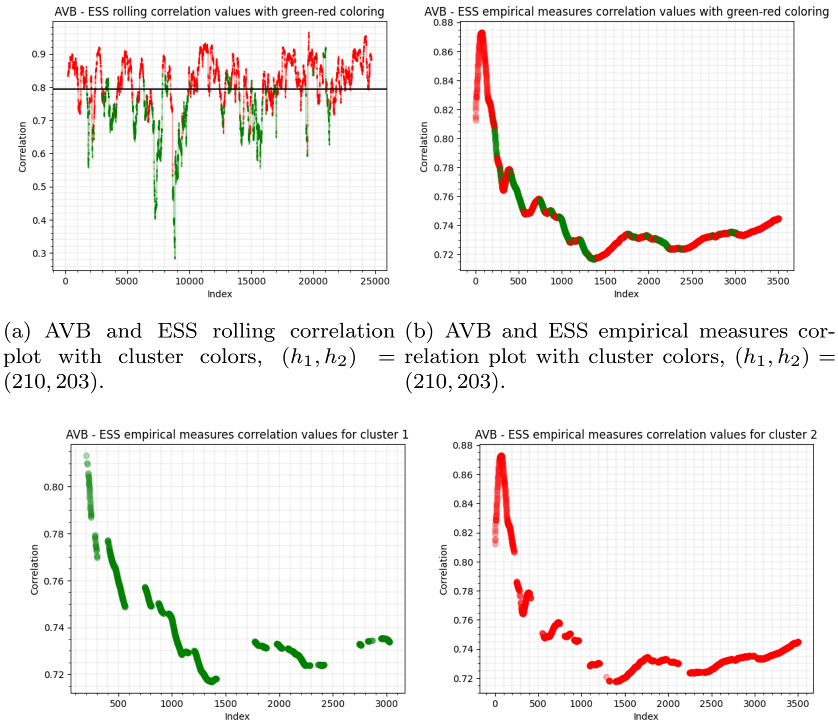
Exhibit 20: Uni-d 1-WK-means applied to AVB and ESS mean-variance clustering plots.

Step 2: Multivariate clustering

For the 2-d clustering, we take $(h_1, h_2) = (210, 203)$. Upon inspection of the rolling correlation graph in 19(b), although there are notable downward spikes in correlation at times, the rolling correlation tends to be quite steady. It therefore seems reasonable to take $k = 2$ clusters for our 2-d WK-means algorithm. These clusters should represent one regime of higher correlation and one of a lower correlation.

Indeed, for $k = 2$, we obtain one correlation regime of high correlation, where the correlation of the centroid's atoms is 0.878, and one of a lower correlation, where the correlation of the centroid's atoms is 0.737. Exhibit 21(a) shows the rolling correlation plot with cluster colors. Overall we see from the plot that the second (red) cluster has caught instances of higher correlation above the average line, represented by the black horizontal line, while the first (green) cluster tends to fall below this line indicating that it captures periods of lower correlation. It is noticeable however that the trends are less distinct than that of the AAPL-AMZN case. This is to be expected due to the much higher correlation between the two assets.

Exhibit 21(b) shows the correlation of each of the empirical measures that we have clustered with their associated cluster colors. We note that where the correlation is increasing the second (red) cluster tends to dominate while where the correlation is decreasing, the first (green) cluster is more apparent. These trends are further reinforced in exhibits 21(c) and 21(d) which show a portion of figure 21(b) magnified and where only one cluster is presented.



(a) AVB and ESS rolling correlation plot with cluster colors, $(h_1, h_2) = (210, 203)$.

(b) AVB and ESS empirical measures correlation plot with cluster colors, $(h_1, h_2) = (210, 203)$.

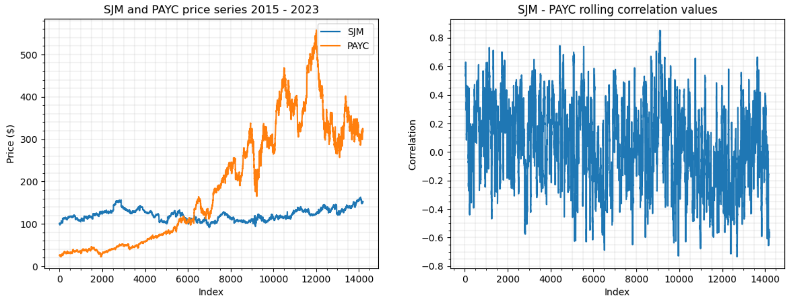
(c) AVB and ESS empirical measures correlation plot with cluster colors, $(h_1, h_2) = (210, 203)$ cluster 1 only.

(d) AVB and ESS empirical measures correlation plot with cluster colors, $(h_1, h_2) = (210, 203)$ cluster 2 only.

Exhibit 21: Correlation plots of AVB and ESS for 2 clusters, $(h_1, h_2) = (210, 203)$.

5.2.3 SJM - PAYC

Finally, SJM and PAYC constitute our last test pairing. J.M. Smucker Co (SJM) is a food and beverage manufacturing company while Paycom Software Inc (PAYC) is an online payroll and human resource technology provider. As one might suspect, the price series of the associated stocks tend to have low correlation on average. Indeed, using the hourly close prices for each stock from 2015-2023, the correlation of these assets over the entire testing period is 0.060 while the standard deviation of their rolling correlation using a window of $h_1 = 35$ market hours is 29.25%. Exhibit 22(a) shows the price series evolution of SJM and PAYC stock between 2015 and 2023 while exhibit 22(b) shows the rolling correlation over a window of 35 market hours between these two assets during that time.



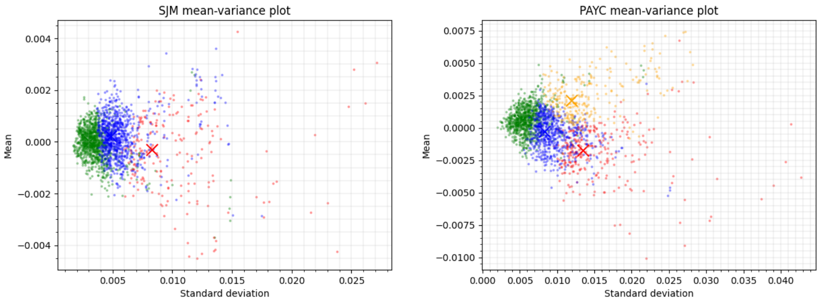
(a) SJM and PAYC price series 2015 - 2023.

(b) SJM and PAYC rolling correlation of returns plot using a rolling window of 35 market hours.

Exhibit 22: Price series and rolling correlation plots of SJM and PAYC.

Step 1: Univariate clustering and data transformation

We initially cluster the empirical measures of each of the two assets using the uni-d 1-WK-means algorithm. After testing, we find that in the case of SJM and PAYC, we require 3 and 4 clusters respectively. Exhibits 23(a) and 23(b) show the mean-variance clustering plots of each asset.



(a) SJM mean-variance clustering plot.

(b) PAYC mean-variance clustering plot.

Exhibit 23: Uni-d 1-WK-means applied to SJM and PAYC mean-variance clustering plots.

Step 2: Multivariate clustering

For the 2-d clustering, we take $(h_1, h_2) = (140, 133)$. We stick to the simplest approach possible and take $k = 2$ in the 2-d WK-means algorithm. The clusters should represent one regime of higher correlation, and one regime of a lower correlation. Indeed, for $k = 2$ we obtain one correlation regime of high correlation, where the correlation of the centroid's atoms is 0.268, and one of a lower correlation, where the correlation of the centroid's atoms is -0.059. Exhibit 24(a) shows the rolling correlation plot with cluster colors. We see from the plot that the second (red) cluster has caught instances of higher correlation above the average line, represented by the black horizontal line, while the first (green) cluster tends to fall below this line indicating that it captures periods of lower correlation.

Exhibit 24(b) shows the correlation of each of the empirical measures that we have clustered with their associated cluster colors. We note that as we might expect, the second (red) and first (green) clusters tend to pick up instances of stronger and weaker correlation respectively. These trends are further reinforced in exhibits 24(c), and 24(d) which show a portion of exhibit 24(b) magnified and where only one cluster is presented.

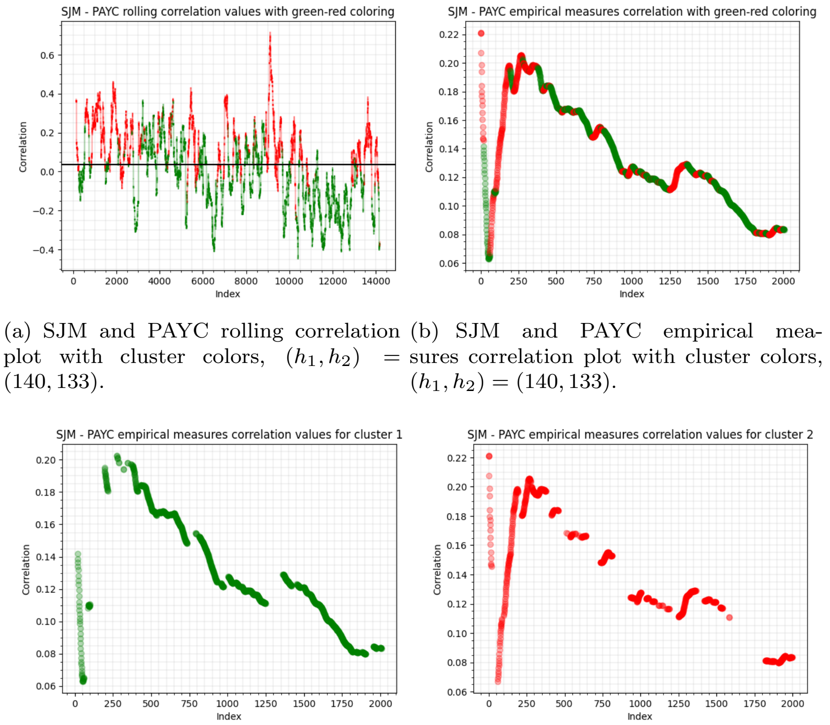
6 Trading strategy

In this section, we use our new two-step clustering-based approach to facilitate a trading strategy. Our candidate assets will be the three pairs of stocks described in section 5 and the overall trading strategy is based on Modern Portfolio Theory (MPT) as described in appendix E.

The Modern Portfolio Theory approach requires four variables: approximations of the expected returns μ, the variance σ^2, and the correlation ρ of the assets in our portfolio, and a target return μ_t. In using our two-step method with the uni-d 1-WK-means and 2-d WK-means algorithms we cluster both the marginal and joint distributions into distinct clusters. These clusters each have an associated centroid which should act as an average of that cluster's constituent data.

When we cluster with the uni-d 1-WK-means algorithm, our synthetic and real data experiments would lead us to expect the mean and variance of each cluster's centroid's atoms to be a representation of the returns data in its cluster. Similarly, when clustering with the 2-d WK-means algorithm we would expect the correlation of the centroid's atoms to be representative of the returns data in its cluster. Therefore our two-step approach lends itself nicely to the mean-variance optimisation problem where we may use each cluster's centroid's mean, variance and correlation values as approximations to the true values in our calculations.

The following experiment demonstrates the benefits of knowing the correct clustering for each pair of assets and each new measure formed from the returns path. For the purposes of this experiment, we assume that we have perfect foresight of the returns path of each asset and thus know which cluster each newly formed measure should be associated with.



(a) SJM and PAYC rolling correlation plot with cluster colors, $(h_1, h_2) = (140, 133)$.

(b) SJM and PAYC empirical measures correlation plot with cluster colors, $(h_1, h_2) = (140, 133)$.

(c) SJM and PAYC empirical measures correlation plot with cluster colors, $(h_1, h_2) = (140, 133)$ cluster 1 only.

(d) SJM and PAYC empirical measures correlation plot with cluster colors, $(h_1, h_2) = (140, 133)$ cluster 2 only.

Exhibit 24: Correlation plots of SJM and PAYC for 2 clusters, $(h_1, h_2) = (140, 133)$.

Formally, we apply the two-step approach to the three pairs of assets AAPL-AMZN, AVB-ESS, and SJM-PAYC for the periods 2005-2022, 2009-2022 and 2015-2022 respectively. We assume that we want to trade over a specific window and take $h_1 = 35$, equivalent to one market week in market hours, and $h_2 = 28$ for both the uni-d 1-WK-means and 2-d WK-means algorithms. Each return is then associated with the first empirical measure it is included within and thus we approximate its mean, variance and correlation using the atoms of the centroid of the cluster associated to that measure. We form our mean-variance portfolios using these approximations and refer to this as the Cluster Based (CB) portfolio. In order to benchmark our strategy, we provide an alternative portfolio where we have used the average over a rolling window of 35 returns as an approximation to the mean, variance and correlation of each asset. We refer to this portfolio as the Rolling Average (RA) portfolio.

Each portfolio starts with a value of one and each trading period is one market hour. The portfolio cumulative returns are shown in exhibit 25 for a few chosen values of μ_t. We see that the choice of μ_t does have an impact on the returns generated by the CB strategy but in all cases we generate a positive return. Exhibit 29 shows the performance of each pair when using its optimal target return tested. That is to say, our target return μ_t for the pairings in each respective period is 0.1%, 0.05% and 0.15% for AAPL-AMZN, AVB-ESS, and SJM-PAYC respectively.

Exhibits 26, 27 and 28 show statistics for the CB and RA strategies when trading each asset pair under its optimal target return. Statistics provided include the cumulative return of each portfolio, calculated as the cumulative product of returns over the testing period when starting from a portfolio value of 1 and the classic annualised Sharpe ratio. We also provide the maximum drawdown of each strategy during this period, defined as being the largest drop from a peak in our cumulative returns to a subsequent trough, as well as the hourly standard deviation in returns σ_h. The results generated use the hourly closing prices of each asset and do not account for market intricacies such as trading fees, bid-ask spread, shorting constraints etc.

μ_t Pair	0.05%	0.10%	0.15%
AAPL-AMZN	6401.37	133,781.43	8225.44
AVB-ESS	41.03	5.53	3.16
SJM-PAYC	30.55	268.39	979.85

Exhibit 25: Cluster Based strategy cumulative returns for $\mu_t = 0.05\%, 0.10\%, 0.15\%$.

Strategy	Cumulative returns	Sharpe ratio	Max drawdown	σ_h
CB	133,781.43	3.03	25.55%	0.55%
RA	5.40	0.46	51.45%	0.13%

Exhibit 26: Cluster Based (CB) vs Rolling Average (RA) strategy statistics for AAPL-AMZN.

Strategy	Cumulative returns	Sharpe ratio	Max drawdown	σ_h
CB	41.03	1.80	23.28%	0.37%
RA	0.92	-0.13	45.39%	0.33%

Exhibit 27: Cluster Based (CB) vs Rolling Average (RA) strategy statistics for AVB-ESS.

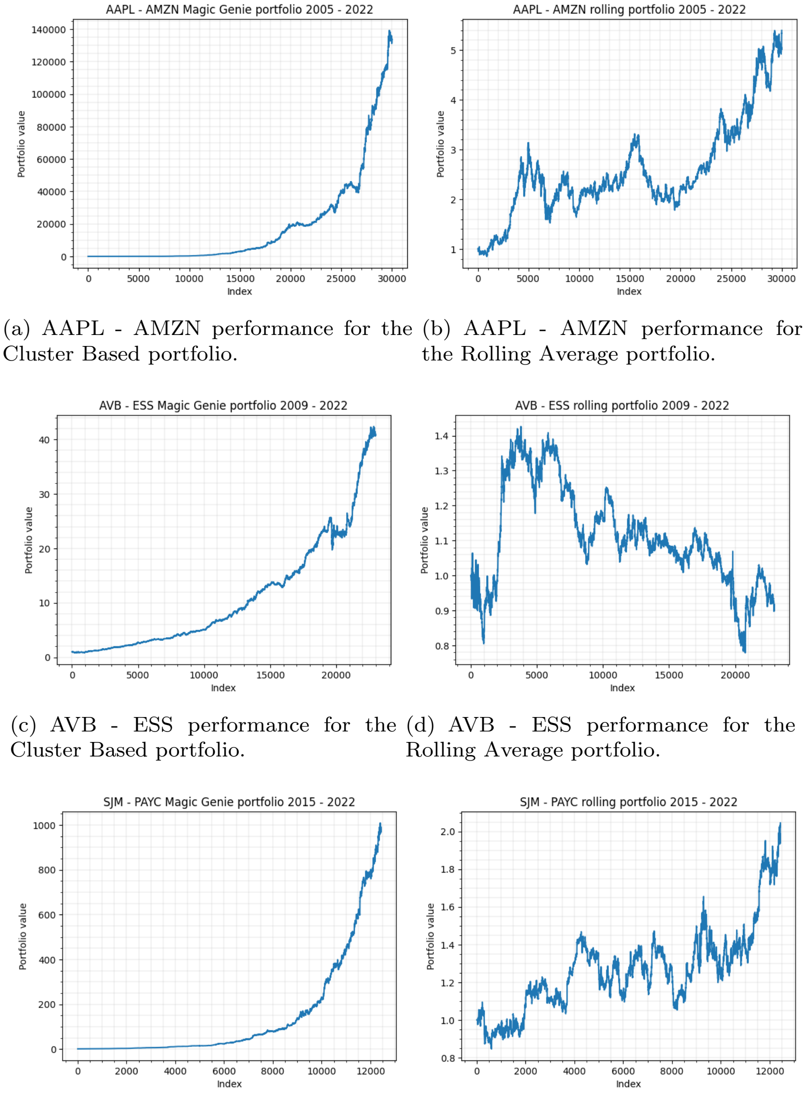
Strategy	Cumulative returns	Sharpe ratio	Max drawdown	σ_h
CB	979.85	3.62	16.97%	0.65%
RA	2.05	0.45	28.49%	0.60%

Exhibit 28: Cluster Based (CB) vs Rolling Average (RA) strategy statistics for SJM-PAYC.

7 Conclusion

In this paper, we have shown that the p-Wasserstein distance and Maximum Mean Discrepancy can be combined with the k-means clustering algorithm in order to effectively partition multidimensional market returns into regimes. These regimes include periods characterised by a change in the mean and variance of the underlying distribution and periods characterised by a change in the correlation between two assets. We compared the use of each distance metric in synthetic settings and found that while the p-Wasserstein distance led to stronger clustering for correlation regimes, the MMD was more effective when clustering regimes characterised by their mean and variance changes. We then demonstrated how combining the WK-means algorithm, as described in [HIM21], with our novel multidimensional algorithms can be effective in clustering real world market returns. Finally, we showed that knowledge of the cluster membership of each market return can be effective in building a profitable portfolio.

Areas ripe for further investigation are plentiful. In particular, a method for predicting the next cluster, possibly based on the overlapping returns or some form of traditional time series prediction, is key as naively assuming the next measure belongs to the same cluster as the current measure yields positive, but weaker results. It would also be interesting to investigate the application of mixture distributions in this context, wherein we would use a mixture of the centroid empirical distributions as opposed to only one when trading a given period of returns.



(a) AAPL - AMZN performance for the Cluster Based portfolio.

(b) AAPL - AMZN performance for the Rolling Average portfolio.

(c) AVB - ESS performance for the Cluster Based portfolio.

(d) AVB - ESS performance for the Rolling Average portfolio.

(e) SJM - PAYC performance for the Cluster Based portfolio.

(f) SJM - PAYC performance for the Rolling Average portfolio.

Exhibit 29: Cluster Based and Rolling Average portfolios.

A Technical Proofs and further results

B Appendix 1

In this appendix we provide statements and proofs related to results in section 2 of the paper.

B.1 The k-means clustering algorithm

Definition B.1 (Stream lift, [BKL+19] (Section 3.3, page 5)). *Let $\mathcal{S}(\mathbb{R}^d)$ be a space of streams over the $\mathbb{R}^d$ and let $m \geq 1$. We call a function*

$$\ell = (\ell^1, \dots, \ell^m) : \mathcal{S}(\mathbb{R}^d) \to \mathcal{S}(\mathcal{S}(\mathbb{R}^d))$$

a lift from the space of streams to the space of streams of segments over $\mathbb{R}^d$.

Definition B.2 (Within-cluster variation, [HIM21] (Definition A.2, page 29)). *Let $k \in \mathbb{N}$ and let $X \in \mathcal{S}(V)$ be a stream of data over a normed vector space V. Suppose $\mathcal{C} \subset \mathcal{C}(X)$ are disjoint clusters over X. Associate to each $\mathcal{C}_j$ its centroid $\bar{x}_j$ for $j = 1, \dots, k$. Then, for a given $\mathcal{C}_j$, the within-cluster variation is defined as*

$$WC(\mathcal{C}_j) := \sum_{x \in \mathcal{C}_j} \|x - \bar{x}_j\|_V^2,$$

for $j = 1, \dots, k$.

Definition B.3 (Total-cluster variation, [HIM21] (Definition A.3, page 29)). *With the notation of definition B.2, define*

$$TC(\mathcal{C}) := \sum_{j=1}^{k} WC(\mathcal{C}_j)$$

to be the total-cluster variation corresponding to a clustering $\mathcal{C} \in \mathcal{C}(X)$ on the normed vector space $(V, \|\cdot\|_V)$.

Definition B.4 (k-means stopping rule, [HIM21] (Definition A.4, page 30)). *Suppose $(V, \|\cdot\|_V)$ is a normed vector space. For fixed $k \in \mathbb{N}$, consider a loss function $l : V^k \times V^k \to \mathbb{R}_+$ given by*

$$l(x, y) := \sum_{i=1}^{k} \|x_i - y_i\|_V.$$

For a tolerance level $\epsilon > 0$, the stopping rule corresponding to the k-means algorithm is given by

$$l(\bar{x}^{n-1}, \bar{x}^n) < \epsilon,$$

where $n \in \mathbb{N}$ denotes the step of the algorithm, $V = \mathbb{R}^d$ for $d \in \mathbb{Z}_+$ and $\bar{x}^n$ is the collection of centroids at the n^{th} step of the algorithm.

B.2 p-Wasserstein distance

Definition B.5 (Multidimensional p-Wasserstein distance, [AGS05] (Equation 7.1.1, page 151)). *Suppose (X, D) is a separable Radon space and that $\mathcal{P}_p(X)$ is the set of probability measures on X with finite p^{th} moment. The p^{th} Wasserstein distance between measures μ, $\nu \in \mathcal{P}_p(X)$ is defined by*

$$\mathcal{W}_p(\mu, \nu) := \left(\min_{\mathbb{P} \in \Pi(\mu,\nu)} \left\{ \int_{X \times X} D(x, y)^p \mathbb{P}(dx, dy) \right\} \right)^{\frac{1}{p}},$$

where

$$\Pi(\mu, \nu) := \{\mathbb{P} \in P(X \times X) : P(A \times X) = \mu(A), \mathbb{P}(X \times B) = \nu(B)\}$$

is the set of transport plans between μ and ν.

Definition B.6 (Wasserstein barycentre, [HIM21] (Definition 2.3, page 9))**.** *Suppose (X, D) is a separable Radon space and let $\mathcal{K} = \{\mu_i\}_{i\geq 1} \subset \mathcal{P}_p(X)$ be a family of Radon measures. The p-Wasserstein barycentre between measures $\bar{\mu}$ of $\mathcal{K}$ is defined to be*

$$\bar{\mu} := \operatorname*{arg\,min}_{\nu\in\mathcal{P}_p(X)} \sum_{\mu_i\in\mathcal{K}} \mathcal{W}_p(\mu_i, \nu).$$

Proposition B.7 (The p-Wasserstein distance is a metric, [Tho19] (Chapter 5.1, proposition 5.4))**.** *The distance $\mathcal{W}_p : \mathcal{P}_p(X) \times \mathcal{P}_p(X) \to [0, \infty)$ is a metric on $\mathcal{P}_p(X)$.*

Proof. As in [Tho19], we show that the first three criteria of being a metric are met. The final criterion, the triangle inequality, can be shown using the 'gluing lemma' as described in [Tho19] chapter 5.1, proposition 5.4. We note that by definition $\mathcal{W}_p(\mu, \nu) \geq 0$ for any $\mu, \nu \in \mathcal{P}_p(X)$, giving us the first requirement. Furthermore, since $D(x, y)$ is itself a metric, we must have that it is symmetrical and since $\mathbb{P} \in \Pi(\mu, \nu)$ if and only if $S\#\mathbb{P} \in \Pi(\mu, \nu)$ where $S(x, y) = (y, x)$ we have that $\mathcal{W}_p(\mu, \nu)$ is symmetrical.

Finally, if $\mu = \nu$ then we may take $\mathbb{P}(x, y) = \delta_x(y)\mu(x)$ such that

$$\mathcal{W}_p(\mu, \nu) \leq \int_{X\times X} D(x, y)^p \mathbb{P}(dx, dy) = 0,$$

as $x = y$ $\mathbb{P}$-almost everywhere. Suppose we have that $\mathcal{W}_p(\mu, \nu) = 0$. Then there must exist some $\mathbb{P} \in \Pi(\mu, \nu)$ such that $x = y$ $\mathbb{P}$-almost everywhere. Hence for any function $f : X \to \mathbb{R}$,

$$\int_X f(x)\mu(dx) = \int_{X\times X} f(x)\mathbb{P}(dx, dy) = \int_{X\times X} f(y)\mathbb{P}(dx, dy) = \int_X f(y)\nu(dy).$$

Since this holds for all f we have that $\mu = \nu$. □

Remark B.8. *The p-Wasserstein distance, defined in definition B.5, is in fact a special case of an integral probability metric via its dual formulation. As shown in [GWX21], where $p = 1$, the dual formulation is given by*

$$\mathcal{W}_1(\mu, \nu) = \sup_{f\in Lip_1(X)} \left(\int_X f(x)d\mu(x) - \int_X f(y)d\nu(y) \right), \tag{2}$$

where $Lip_1(X)$ denotes a collection of Lipschitz functions with Lipschitz constant 1

$$Lip_1 := \{f : \|f\|_{Lip} \leq 1\},$$

and where

$$\|f\|_{Lip} := \sup_{x\in\mathrm{supp}(\mu), y\in\mathrm{supp}(\nu), x\neq y} \frac{|f(x) - f(y)|}{D(x, y)}.$$

Since μ and ν are probability measures we may write equation (2) as

$$\mathcal{W}_1(\mu, \nu) = \sup_{Lip_1}(\mathbb{E}_\mu[f(x)] - \mathbb{E}_\nu[f(y)]).$$

Therefore the 1-Wasserstein distance is an integrable probability metric over $\mathcal{F}$ given by the unit ball in the space of functions

$$Lip(X) = \{f : X \to \mathbb{R} : f \text{ continuous}, \quad \|f\|_{Lip} < \infty\},$$

and it shares a relationship with the Maximum Mean Discrepancy.

Proposition B.9 (Weak convergence of p-Wasserstein distance, [ABH⁺12] (Chapters 2 & 3, pages 1 - 155))**.** *A sequence $(\mu_i)_{\geq 1} \subset \mathcal{P}_p(X)$ converges weakly to $\mu \in \mathcal{P}_p(X)$ if and only if $\mathcal{W}_p(\mu_i, \mu) \to 0$ as $n \to \infty$.*

Proof. See Chapters 2 & 3, pages 1 - 155 in [ABH⁺12]. □

B.3 Maximum Mean Discrepancy

Problem 1 (Two-sample test, [BGR+12] (Problem 1))**.** *Let $(\mathcal{X}, D)$ be a metric space. Suppose X and Y are independent random variables on $\mathcal{X}$. Suppose that the Borel probability measure of X is μ and that of Y is ν, where $\mu, \nu \in \mathcal{P}(\mathcal{X})$, where $\mathcal{P}(\mathcal{X})$ is the set of probability measures on $\mathcal{X}$. If we draw i.i.d. samples $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_m)$ where $x_i \sim \mu$ for $i = 1, \ldots, n$ and $y_j \sim \nu$ for $j = 1, \ldots, m$, when can we determine if $\mu \neq \nu$? In other words, we wish to implement a test for the two-sample problem*

$$H_0 : \mu = \nu \quad H_1 : \mu \neq \nu.$$

Definition B.10 (Hilbert space, [Fol99] (Pages 171 - 172))**.** *A Hilbert space $\mathcal{H}$ is a complex vector space equipped with an inner product, that is complete with respect to the norm induced by its inner product.*

Definition B.11 (Kernel function, [SS05] (Equation 3, page 3))**.** *Let $\mathcal{X}$ be a topological space. We define a symmetric, similarity measure of the form*

$$\begin{aligned} \kappa :& \mathcal{X} \times \mathcal{X} \to \mathbb{R} \\ & (x, x') \to \kappa(x, x'), \end{aligned}$$

such that κ returns a real number characterising the similarity between x and x'. We refer to κ as a kernel function or method.

Definition B.12 (Universal reproducing kernel Hilbert space, [BGR+12] (Section 2.2, page 727))**.** *Let $\mathcal{X}$ be a topological space, $\mathcal{H}$ be a Hilbert space and $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a reproducing kernel. We say that the reproducing kernel Hilbert space $(\mathcal{H}, \kappa)$ is universal if $\kappa(\cdot, \cdot)$ is continuous and $\mathcal{H}$ is dense in $C(\mathcal{X})$, the space of bounded, continuous functions on $\mathcal{X}$, with respect to the L_∞ norm.*

Definition B.13 (Characteristic kernel, [FGSS08] (Section 2))**.** *Let $\mathcal{X}$ be a non-empty set. A kernel κ on $\mathcal{X}$ is called characteristic if the mean mapping*

$$\mu \to \mathbb{E}_{X \sim \mu}[\kappa(\cdot, X)]$$

is injective.

Lemma B.14 (Unbiased empirical estimate of the squared MMD, [BGR+12] (Lemma 6, page 728))**.** *Using the notation of 2.2, let x and x' be samples of X with distribution μ, and y and y' be samples of Y with distribution ν. The squared MMD is given by*

$$MMD^2[\mathcal{F}, \mu, \nu] = \mathbb{E}_{x,x'}[\kappa(x, x')] - 2\mathbb{E}[\kappa(x, y)] + \mathbb{E}_{y,y'}[\kappa(y, y')],$$

where x' and y' are independent copies of x and y respectively. An unbiased empirical estimate is given by

$$MMD_u^2[\mathcal{F}, X, Y] = \frac{1}{m(m-1)} \sum_{i=1}^{m} \sum_{j \neq i}^{m} \kappa(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j \neq i}^{n} \kappa(y_i, y_j) - \frac{2}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \kappa(x_i, y_j).$$

C Appendix 3

In this appendix we provide further results related to the synthetic experiments in section 4 of the paper. We also provide a derivation of the correlated Merton jump diffusion processes used in section 4.1.

C.1 Geometric Brownian motion

C.1.1 Fixed correlation regime results

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	85.61% ± 3.54%	87.11% ± 2.78%	85.12% ± 4.42%
Uni-d 1-WK-means	86.46% ± 4.04%	89.45% ± 2.64%	85.26% ± 4.82%
Uni-d 2-WK-means	94.28% ± 1.82%	86.78% ± 2.34%	96.55% ± 1.87%
2-d MMDK-means	77.57% ± 4.10%	81.59% ± 3.86%	76.05% ± 5.62%

Exhibit 30: Accuracy scores with 95% CI, GBM synthetic path with simultaneous mean-variance regimes and fixed $\rho = 1$, $n = 50$ runs.

C.1.2 Further fixed correlation regime results

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	84.36% ± 4.19%	86.52% ± 4.35%	83.45% ± 4.65%
Uni-d 1-WK-means	86.37% ± 3.90%	89.50% ± 3.10%	85.13% ± 4.68%
Uni-d 2-WK-means	94.94% ± 1.55%	89.82% ± 1.21%	96.43% ± 1.66%
2-d MMDK-means	84.63% ± 4.21%	86.57% ± 5.22%	83.78% ± 4.57%

Exhibit 31: Accuracy scores with 95% CI, GBM synthetic path with simultaneous mean-variance regimes and fixed $\rho = 0$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	88.12% ± 4.48%	83.53% ± 3.88%	89.44% ± 4.96%
2-d MMDK-means	95.02% ± 1.70%	95.31% ± 0.35%	94.70% ± 2.32%

Exhibit 32: Accuracy scores with 95% CI, GBM synthetic path with simultaneous mean-variance regimes and fixed $\rho = 0.5$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	82.52% ± 4.21%	87.99% ± 3.11%	80.51% ± 4.99%
2-d MMDK-means	77.17% ± 5.43%	85.42% ± 6.27%	74.24% ± 5.84%

Exhibit 33: Accuracy scores with 95% CI, GBM synthetic path with simultaneous mean-variance regimes and fixed $\rho = -0.5$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	85.89% ± 4.25%	86.25% ± 2.68%	85.57% ± 5.09%
2-d MMDK-means	74.85% ± 3.69%	87.23% ± 2.91%	70.55% ± 4.32%

Exhibit 34: Accuracy scores with 95% CI, GBM synthetic path with simultaneous mean-variance regimes and fixed $\rho = -1$, $n = 50$ runs.

C.1.3 Fixed mean-variance regime results

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	92.71% ± 4.72%	98.98% ± 0.14%	90.40% ± 6.29%
2-d MMDK-means	99.46% ± 3.76%	99.20% ± 0.00%	99.32% ± 0.00%

Exhibit 35: Accuracy scores with 95% CI, GBM synthetic path with simultaneous correlation regimes and $\rho_0 = 0$, $\rho_1 = 1$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	87.78% ± 5.21%	91.85% ± 4.76%	86.22% ± 6.09%
2-d MMDK-means	97.41% ± 2.78%	96.56% ± 3.22%	97.33% ± 2.81%

Exhibit 36: Accuracy scores with 95% CI, GBM synthetic path with simultaneous correlation regimes and $\rho_0 = 0.5$, $\rho_1 = 1$, $n = 50$ runs.

C.1.4 Further fixed mean-variance regime results

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	82.28% ± 6.78%	84.07% ± 8.43%	81.49% ± 7.01%
2-d MMDK-means	88.32% ± 5.55%	84.42% ± 7.84%	89.41% ± 5.23%

Exhibit 37: Accuracy scores with 95% CI, GBM synthetic path with simultaneous correlation regimes and $\rho_0 = 1$, $\rho_1 = 0$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	98.60% ± 1.69%	97.83% ± 1.93%	98.63% ± 1.60%
2-d MMDK-means	95.03% ± 4.20%	95.18% ± 3.80%	94.76% ± 4.32%

Exhibit 38: Accuracy scores with 95% CI, GBM synthetic path with simultaneous correlation regimes and $\rho_0 = 0$, $\rho_1 = -1$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	85.21% ± 6.00%	91.17% ± 5.31%	83.02% ± 6.82%
2-d MMDK-means	80.01% ± 6.67%	83.91% ± 5.63%	78.52% ± 7.26%

Exhibit 39: Accuracy scores with 95% CI, GBM synthetic path with simultaneous correlation regimes and $\rho_0 = -1$, $\rho_1 = 0$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	94.70% ± 3.84%	95.92% ± 3.23%	94.08% ± 4.47%
2-d MMDK-means	89.49% ± 5.59%	91.55% ± 4.94%	88.59% ± 6.03%

Exhibit 40: Accuracy scores with 95% CI, GBM synthetic path with simultaneous correlation regimes and $\rho_0 = -0.5$, $\rho_1 = -1$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	65.92% ± 4.31%	80.12% ± 5.00%	61.04% ± 5.93%
2-d MMDK-means	52.75% ± 2.70%	59.65% ± 2.14%	50.33% ± 3.73%

Exhibit 41: Accuracy scores with 95% CI, GBM synthetic path with simultaneous correlation regimes and $\rho_0 = 0$, $\rho_1 = 0.5$, $n = 50$ runs.

C.1.5 Free mean-variance regime and correlation regime results

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	63.52% ± 5.04%	61.46% ± 5.10%	64.57% ± 6.41%
2-d MMDK-means	60.49% ± 3.08%	61.85% ± 3.59%	59.51% ± 4.59%

Exhibit 42: Accuracy scores with 95% CI, GBM synthetic path, $k = 4$, with correlation and mean-variance regimes, $n = 50$ runs.

Algorithm	Regime-on JR_1	Regime-on JR_2	Regime-on JR_3
2-d WK-means	58.77% ± 9.08%	43.57% ± 13.09%	82.03% ± 8.88%
2-d MMDK-means	21.87% ± 10.14%	70.01% ± 9.77%	93.67% ± 3.85%

Exhibit 43: Accuracy scores with 95% CI, GBM synthetic path, $k = 4$, with correlation (JR_1), mean-variance (JR_2) and joint correlation-mean-variance (JR_3) regimes, $n = 50$ runs.

C.2 Correlation of Merton jump diffusion processes

The following work has been adapted from [CC11].

Let $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}, \mathbb{P})$ be a probability space and let $W_t = (W_t^1, W_t^2)$ be a bivariate correlated Brownian motion process adapted to the filtration. We define a covariance matrix

$$\Sigma = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix},$$

where $dW_t^1, W_t^2 = \rho dt$ with ρ the instantaneous correlation between the two Brownian motion components. We define a homogeneous Poisson counting measure $p(dy, dt) \equiv p(dy^1, dy^2, dt)$ defined over $\mathbb{R}^2 \times [0, T]$ which is associated with a marked point process $((\mathbf{Y}_n), N_t)$. The intensity of the Poisson measure is $\lambda m_{\mathbb{P}}(dy)dt$, where $\lambda > 0$ is the constant arrival rate of the jumps of the Poisson process N_t under $\mathbb{P}$ and $m_{\mathbb{P}}(dy)$ is the probability distribution on the independently and identically distributed marks $\mathbf{Y}_n$, which is also independent of N_t and W_t.

We denote the compensated measure as

$$\hat{p}(dy, dt) = p(dy, dt) - \lambda m_{\mathbb{P}}(dy)dt.$$

We assume that we have two assets S^1 and S^2 whose return dynamics under the market measure $\mathbb{P}$ are given by

$$\frac{dS_t^i}{S_{t-}^i} = \mu^i dt + \sigma^i dW_t^i + \int_{\mathbb{R}^2} [e^{y^i} - 1]\hat{p}(dy, dt),$$

for $i = 1, 2$ where μ^i is the expected return per unit time, and σ^i is the instantaneous volatility per unit time. Since we have two assets in our model, the jump-size $\mathbf{Y} = (Y^1, Y^2)^T$ is a bivariate process taking values $y = (y^1, y^2)^T$ in $\mathbb{R}^2$. When restricting the Poisson measure to one of the jump-size components, we write

$$p(dy^i, dt) \equiv \int_{\mathbb{R}} p(dy^i, dy^j, dt)dy^j,$$

for $i = 1, 2$ and $j = 2, 1$ with the associated compensated measures

$$\hat{p}(dy^i, dt) = p(dy^i, dt) - \lambda m_{\mathbb{P}}(dy^i)dt,$$

for $i = 1, 2$ where $m_{\mathbb{P}}(dy^i)$ denotes the marginal distribution of jump-sizes Y_n^i under $\mathbb{P}$.

The Y^i for $i = 1, 2$ are random jump-sizes assumed to be correlated pairwise with correlation $\mathrm{Corr}[Y^1, Y^2] = \rho_Y$. For our purposes we assume $\rho_Y = \rho$. We define the expected proportional jump-size as

$$\kappa^i \equiv \mathbb{E}_{\mathbb{P}}[e^{Y^i} - 1] \equiv \int_{\mathbb{R}} [e^{y^i} - 1]m_{\mathbb{P}}(dy^i).$$

We note that in terms of the compensated Poisson measure associated with each particular asset, we may write

$$\begin{aligned}
\frac{dS_t^i}{S_{t-}^i} &= \mu^i dt + \sigma^i dW_t^i + \int_{\mathbb{R}} [e^{y^i} - 1]\hat{p}(dy^i, dt) \\
&= \mu^i dt + \sigma^i dW_t^i + \int_{\mathbb{R}} [e^{y^i} - 1](p(dy^i, dt) - \lambda m_{\mathbb{P}}(dy^i)dt) \\
&= \mu^i dt + \sigma^i dW_t^i + \int_{\mathbb{R}} [e^{y^i} - 1]p(dy^i, dt) - \lambda \int_{\mathbb{R}} [e^{y^i} - 1]m_{\mathbb{P}}(dy^i)dt \\
&= (\mu^i - \lambda\kappa^i)dt + \sigma^i dW_t^i + \int_{\mathbb{R}^2} [e^{y^i} - 1]p(dy^i, dt),
\end{aligned}$$

which yields a solution of the form

$$S_t^i = S_0^i \mathrm{exp}\left[\left(\mu^i - \lambda\kappa^i - \frac{\sigma^i}{2}\right)t + \sigma^i W_t^i + \sum_{n=1}^{N_t} Y_n^i\right],$$

for $i = 1, 2$.

C.3 Merton jump diffusion process

C.3.1 Further fixed correlation regime results

Algorithm	Asset	Total	Regime-on	Regime-off
2-d WK-means	-	85.47% ± 4.83%	79.16% ± 9.73%	87.37% ± 5.30%
Uni-d 1-WK-means	1	90.07% ± 3.79%	83.75% ± 8.38%	91.96% ± 4.67%
Uni-d 2-WK-means	1	94.71% ± 1.65%	81.45% ± 6.77%	98.91% ± 0.05%
Uni-d 1-WK-means	2	89.74% ± 3.23%	80.33% ± 9.47%	92.67% ± 3.63%
Uni-d 2-WK-means	2	89.50% ± 2.35%	60.12% ± 9.65%	99.07% ± 0.08%
2-d MMDK-means	-	83.07% ± 7.44%	75.06% ± 11.07%	85.55% ± 6.30%

Exhibit 44: Accuracy scores with 95% CI, MJD synthetic path with simultaneous mean-variance regimes and $\rho = 0$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	87.54% ± 3.96%	77.76% ± 9.27%	90.59% ± 4.81%
2-d MMDK-means	97.27% ± 1.97%	96.49% ± 0.81%	97.30% ± 2.40%

Exhibit 45: Accuracy scores with 95% CI, MJD synthetic path with simultaneous mean-variance regimes and fixed $\rho = 0.5$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	90.84% ± 3.05%	78.15% ± 9.37%	94.85% ± 2.64%
2-d MMDK-means	90.71% ± 3.67%	98.46% ± 0.15%	87.92% ± 4.92%

Exhibit 46: Accuracy scores with 95% CI, MJD synthetic path with simultaneous mean-variance regimes and fixed $\rho = -0.5$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	89.05% ± 3.81%	76.12% ± 10.65%	94.97% ± 3.50%
2-d MMDK-means	88.04% ± 4.35%	95.13% ± 1.85%	85.47% ± 5.40%

Exhibit 47: Accuracy scores with 95% CI, MJD synthetic path with simultaneous mean-variance regimes and fixed $\rho = -1$, $n = 50$ runs.

C.3.2 Further fixed mean-variance regime results

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	83.41% ± 6.11%	78.35% ± 9.70%	84.90% ± 6.45%
2-d MMDK-means	77.41% ± 6.79%	79.48% ± 6.46%	76.54% ± 7.20%

Exhibit 48: Accuracy scores with 95% CI, MJD synthetic path with simultaneous correlation regimes and $\rho_0 = 1$, $\rho_1 = 0$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	90.05% ± 4.87%	84.63% ± 9.41%	91.64% ± 6.11%
2-d MMDK-means	90.98% ± 5.12%	84.26% ± 9.39%	93.00% ± 4.10%

Exhibit 49: Accuracy scores with 95% CI, MJD synthetic path with simultaneous correlation regimes and $\rho_0 = 0$, $\rho_1 = -1$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	82.95% ± 6.56%	74.09% ± 10.39%	85.71% ± 6.77%
2-d MMDK-means	75.77% ± 6.75%	79.08% ± 7.06%	74.94% ± 7.33%

Exhibit 50: Accuracy scores with 95% CI, MJD synthetic path with simultaneous correlation regimes and $\rho_0 = -1$, $\rho_1 = 0$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	82.09% ± 5.58%	84.48% ± 7.59%	81.10% ± 6.91%
2-d MMDK-means	89.34% ± 5.54%	88.48% ± 6.61%	89.43% ± 5.39%

Exhibit 51: Accuracy scores with 95% CI, MJD synthetic path with simultaneous correlation regimes and $\rho_0 = -0.5$, $\rho_1 = -1$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	64.86% ± 4.63%	77.81% ± 7.73%	60.40% ± 6.94%
2-d MMDK-means	55.86% ± 2.54%	66.40% ± 4.73%	52.23% ± 2.33%

Exhibit 52: Accuracy scores with 95% CI, MJD synthetic path with simultaneous correlation regimes and $\rho_0 = 0$, $\rho_1 = 0.5$, $n = 50$ runs.

D Appendix 4

In this appendix we provide statements and proofs related to results in section 5 of the paper.

Theorem D.1 (Sklar's theorem, [Skl59] (Theorem 2, page 230))**.** *Let $X = (X_1, \dots, X_d)$ be a random vector with joint CDF F and with marginal CDFs $F_1, \dots, F_d$. Then there is a copula $C : [0,1]^d \to [0,1]$ such that*

$$F(x_1, \dots, x_d) = C\Big(F_1(x_1), \dots, F_d(x_d)\Big),$$

where $x_1, \dots, x_d \in \mathbb{R} \cup \{-\infty, \infty\}$. Moreover, the copula is uniquely defined when $F_1, \dots, F_d$ are continuous.

Proposition D.2 (Probability transformation)**.** *Suppose X is a random variable with continuous CDF F. Then $F(X) \sim$ Uniform(0, 1).*

Proof. Let X be a continuous random variable with CDF $F(x)$. We define $Y := F(X)$ such that $G(y)$ is the CDF of Y. Then for $y \in [0,1]$, if $F^{-1}(y)$ exists we have that

$$\begin{aligned} G(y) &= \mathbb{P}(Y \leq y) \\ &= \mathbb{P}(F(X) \leq y) \\ &= \mathbb{P}(X \leq F^{-1}(y)) \\ &= F(F^{-1}(y)) \\ &= y \end{aligned}$$

If $F^{-1}(y)$ does not exist then we replace it with the generalised inverse $F^{\leftarrow}(y) = \inf\{x : F(x) \geq y\}$ for $y \in (0,1)$, $F^{\leftarrow}(-\infty) = 0$ and $F^{\leftarrow}(\infty) = 1$, and the result still holds. □

Theorem D.3 (Kolmogorov-Smirnov two-tail goodness-of-fit test, [Dod08] (pages 283-287))**.** *Let $X = (X_1, \dots, X_n)$ be independent and identically distributed random variables with an empirical cumulative distribution function denoted $F_n(x)$ and let $F(x)$ be a continuous distribution. We set up a goodness-of-fit test using the Kolmogorov-Smirnov test statistic defined as*

$$D_n = \sup_x |F_n(x) - F(x)|,$$

and where our hypotheses to test are

$$\begin{aligned} H_0 &: F_n(x) = F(x) \quad \text{for each } x, \\ H_1 &: F_n(x) \neq F(x) \quad \text{for at least one value of } x. \end{aligned}$$

D.1 Geometric Brownian motion synthetic data experiments revisited

Algorithm	Asset	Total	Regime-on	Regime-off
Uni-d 1-WK-means	1	94.49%	94.43%	94.29%
Uni-d 1-WK-means	2	95.82%	96.06%	95.52%

Exhibit 53: Accuracy scores of uni-d 1-WK-means applied to GBM synthetic paths with mean-variance and correlation regimes, $\rho_0 = 0$ and $\rho_1 = 1$, $n = 50$ runs.

Algorithm	Total	Regime-on	Regime-off
2-d WK-means	99.48% ± 0.00%	99.01% ± 0.00%	99.41% ± 0.00%
2-d MMDK-means	99.44% ± 0.00%	99.41% ± 0.00%	99.22% ± 0.00%

Exhibit 54: Accuracy scores of 2-d WK-means and 2-d MMDK-means applied to GBM synthetic paths with mean-variance and correlation regimes, $\rho_0 = 0$ and $\rho_1 = 1$.

E Appendix 5

In this appendix we provide statements and proofs related to results in section 6 of the paper.

E.1 Modern Portfolio Theory

First proposed by H. Markowitz in 1952 [Mar52], Modern Portfolio Theory suggests we assemble our portfolio such that we maximise our expected returns μ_p for a given level of risk σ_p. This is commonly referred to as mean-variance optimisation. For two assets, this problem may be expressed in symbolic form as

$$\begin{aligned} &\max_w \mu_p = w^T \mu \\ &\text{s.t. } \sigma_t^2 = w^T \Sigma w \text{ and } w^T \mathbf{1} = 1, \end{aligned}$$

where w is our 2×1 weights vector, μ is the 2×1 vector of expected returns, σ_t^2 is our target portfolio variance, $\mathbf{1}$ is a 2×1 vector of ones (highlighted in bold so as to distinguish it from the integer value 1), and Σ is the 2×2 covariance matrix of our constituent assets. For a portfolio of two assets one may write the covariance matrix Σ as

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \rho \times \sigma_1 \times \sigma_2 \\ \rho \times \sigma_1 \times \sigma_2 & \sigma_2^2 \end{pmatrix},$$

where σ_1^2 is the variance of asset 1, σ_2^2 is the variance of asset 2, and ρ is the correlation between the two assets. The constraint of $w^T \mathbf{1} = 1$ ensures the total weight of our portfolio sums to 1. This optimisation problem has a dual form where we instead minimise the variance of our portfolio for a given target return, μ_t,

$$\min_w \sigma_p^2 = w^T \Sigma w \tag{3}$$

$$\text{s.t. } \mu_t = w^T \mu \text{ and } w^T \mathbf{1} = 1. \tag{4}$$

This is a constrained minimisation problem and we may solve it using a Lagrangian function and matrix algebra. Solving this minimisation problem leaves us with an equation of the form $Ax = b$. Should A be invertible, then it has solution $x = A^{-1}b$ where

$$A = \begin{pmatrix} 2\Sigma & \mu & \mathbf{1} \\ \mu^T & 0 & 0 \\ \mathbf{1}^T & 0 & 0 \end{pmatrix}, \quad x = \begin{pmatrix} w \\ \lambda_1 \\ \lambda_2 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ \mu_t \\ \mathbf{1} \end{pmatrix}. \tag{5}$$

In order to solve equation (5) in python, we make use of the *scipy.linalg* package and its *solve* method. Passing the matrix A and vector b as inputs, the method returns the array x. We then proceed to normalise the array such that the weights are less than 1 in absolute value.

References

[AAH$^+$17] Arbey Aragón, Andrés Arévalo, Germán Hernández, Diego León, Javier Sandoval, and Jaime Niño. Clustering algorithms for risk-adjusted portfolio construction. *Procedia Computer Science*, 108:1334–1343, 2017.

[ABH$^+$12] Luigi Ambrosio, Alberto Bressan, Dirk Helbing, Axel Klar, and Enrique Zuazua. *Modelling and Optimisation of Flows on Networks: Cetraro, Italy 2009, Editors: Benedetto Piccoli, Michel Rascle*, volume 2062. Springer, 2012.

[ACFL20] Imanol Perez Arribas, Thomas Cochrane, Peter Foster, and Terry Lyons. Anomaly detection on streamed data. *arXiv preprint arXiv:2006.03487*, 2020.

[AGS05] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2005.

[ANN14] Shun-ichi Amari, Frank Nielsen, and Richard Nock. On clustering histograms with k-means by using mixed α-divergences. *Entropy*, 16(6):3273–3301, 2014.

[AV07] David Arthur and Sergei Vassilvitskii. K-means++ the advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035, 2007.

[BBRW20] Ammar Belatreche, Ahmed Bouridane, Baqar Rizvi, and Ian Watson. Detection of stock price manipulation using kernel based principal component analysis and multivariate density estimation. *IEEE Access*, 8:135989–136003, 2020.

[BDM12] Rainer Burkard, Mauro Dell'Amico, and Silvano Martello. *Assignment problems: revised reprint*. SIAM, 2012.

[BDPR12] Marc Bernot, Julie Delon, Gabriel Peyré, and Julien Rabin. Wasserstein barycenter and its application to texture mixing. In *Scale Space and Variational Methods in Computer Vision: Third International Conference, SSVM 2011, Ein-Gedi, Israel, May 29–June 2, 2011, Revised Selected Papers 3*, pages 435–446. Springer, 2012.

[BGR$^+$12] Karsten M Borgwardt, Arthur Gretton, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

[BKL$^+$19] Patric Bonnier, Patrick Kidger, Terry Lyons, Imanol Perez Arribas, and Cristopher Salvi. Deep signature transforms. *Advances in Neural Information Processing Systems*, 32, 2019.

[BKN+19] Roland Badeau, Soheil Kolouri, Kimia Nadjahi, Gustavo Rohde, and Umut Simsekli. Generalized sliced wasserstein distances. *Advances in neural information processing systems*, 32, 2019.

[CC11] Gerald HL Cheang and Carl Chiarella. Exchange options under jump-diffusion dynamics. *Applied Mathematical Finance*, 18(3):245–276, 2011.

[CD14] Marco Cuturi and Arnaud Doucet. Fast computation of wasserstein barycenters. In *International conference on machine learning*, pages 685–693. PMLR, 2014.

[CDL+16] Nicolas Champagnat, Madalina Deaconu, Antoine Lejay, Nicolas Navet, and Khaled Salhi. Regime switching model for financial data: Empirical risk analysis. *Physica A: Statistical Mechanics and its Applications*, 461:148–157, 2016.

[CGJ20] Marco Cuturi, Alexandre Gramfort, and Hicham Janati. Debiased sinkhorn barycenters. In *International Conference on Machine Learning*, pages 4692–4701. PMLR, 2020.

[CGOY15] A Taylan Cemgil, Fikret S Gürgen, Nesrin Okay, and M Serdar Yümlü. Bayesian changepoint and time-varying parameter learning in regime switching volatility models. *Digital Signal Processing*, 40:198–212, 2015.

[Cho07] Nicolas Chopin. Dynamic detection of change points in long time series. *Annals of the Institute of Statistical Mathematics*, 59(2):349, 2007.

[CPS+22] Rongbo Chen, Jean-Marc Patenaude, Mingxuan Sun, Shengrui Wang, and Kunpeng Xu. Clustering-based cross-sectional regime identification for financial market forecasting. In *Database and Expert Systems Applications: 33rd International Conference, DEXA 2022, Vienna, Austria, August 22–24, 2022, Proceedings, Part II*, pages 3–16. Springer, 2022.

[CT17] Jun Chen and Edward PK Tsang. Constructing a bellwether theory: Regime change detection using directional change. In *2017 9th Computer Science and Electronic Engineering (CEEC)*, pages 112–115. IEEE, 2017.

[DM12] Pierpaolo D'Urso and Elizabeth Ann Maharaj. Wavelets-based clustering of multivariate time series. *Fuzzy Sets and Systems*, 193:33–61, 2012.

[Dod08] Yadolah Dodge. *The concise encyclopedia of statistics*. Springer Science & Business Media, 2008.

[EHK+17] Attila Egri, Illés Horváth, Ferenc Kovács, Roland Molontay, and Krisztián Varga. Cross-correlation based clustering and dimension reduction of multivariate time series. In *2017 IEEE 21st International Conference on Intelligent Engineering Systems (INES)*, pages 000241–000246. IEEE, 2017.

[FGSS08] Kenji Fukumizu, Arthur Gretton, Bernhard Schölkopf, and Bharath K Sriperumbudur. Characteristic kernels on groups and semigroups. *Advances in neural information processing systems*, 21, 2008.

[Fol99] Gerald B Folland. *Real analysis: modern techniques and their applications*, volume 40. John Wiley & Sons, 1999.

[GLMT08] Mauro Gallegati, Fabrizio Lillo, Rosario N Mantegna, and Vincenzo Tola. Cluster analysis for portfolio optimization. *Journal of Economic Dynamics and Control*, 32(1):235–258, 2008.

[GT05] Massimo Guidolin and Allan Timmermann. Economic implications of bull and bear regimes in uk stock and bond returns. *The Economic Journal*, 115(500):111–143, 2005.

[GWX21] Rui Gao, Jie Wang, and Yao Xie. Two-sample test using projected wasserstein distance. In *2021 IEEE International Symposium on Information Theory (ISIT)*, pages 3320–3325. IEEE, 2021.

[GWXY19] Dongdong Ge, Haoyue Wang, Zikai Xiong, and Yinyu Ye. Interior-point methods strike back: Solving the wasserstein barycenter problem. *Advances in Neural Information Processing Systems*, 32, 2019.

[Ham89] James D Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica: Journal of the econometric society*, pages 357–384, 1989.

[HI23] Blanka Horvath and Zacharia Issa. Non-parametric online market regime detection and regime clustering for multidimensional and path-dependent data structures. *arXiv preprint arXiv:2306.15835*, 2023.

[HIM21] Blanka Horvath, Zacharia Issa, and Aitor Muguruza. Clustering market regimes using the wasserstein distance. *arXiv preprint arXiv:2110.11848*, 2021.

[HJKS01] Weiyun Huang, Erik Johnson, Hillol Kargupta, and Krishnamoorthy Sivakumar. Distributed clustering using collective principal component analysis. *Knowledge and Information Systems*, 3:422–448, 2001.

[HNZ16] Ning Hao, Yue S Niu, and Heping Zhang. Multiple change-point detection: a selective overview. *Statistical Science*, pages 611–623, 2016.

[IM17] SR Idate and Harshada C Mandhare. A comparative study of cluster based outlier detection, distance based outlier detection and density based outlier detection techniques. In *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 931–935. IEEE, 2017.

[Kan60] Leonid V Kantorovich. Mathematical methods of organizing and planning production. *Management science*, 6(4):366–422, 1960.

[Kuh55] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[Li19] Hailin Li. Multivariate time series clustering based on common principal component analysis. *Neurocomputing*, 349:239–247, 2019.

[Lia05] T Warren Liao. Clustering of time series data—a survey. *Pattern recognition*, 38(11):1857–1874, 2005.

[LOV21] Ángel López-Oriona and José A Vilar. Quantile cross-spectral density: A novel and effective tool for clustering multivariate time series. *Expert Systems with Applications*, 185:115677, 2021.

[LR09] Theis Lange and Anders Rahbek. An introduction to regime switching time series models. In *Handbook of Financial Time Series*, pages 871–887. Springer, 2009.

[LWWY17] Jia Li, James Z Wang, Panruo Wu, and Jianbo Ye. Fast discrete distribution clustering using wasserstein barycenter with sparse support. *IEEE Transactions on Signal Processing*, 65(9):2317–2332, 2017.

[Mac67] J MacQueen. Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, pages 281–297. University of California Los Angeles LA USA, 1967.

[Mar52] Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952.

[MMS12] John M Maheu, Thomas H McCurdy, and Yong Song. Components of bull and bear markets: bull corrections and bear rallies. *Journal of Business & Economic Statistics*, 30(3):391–403, 2012.

[Mon81] Gaspard Monge. Mémoire sur la théorie des déblais et des remblais. *Mem. Math. Phys. Acad. Royale Sci.*, pages 666–704, 1781.

[Pea01] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901.

[PRSX07] John Powell, Rubén Roa, Jing Shi, and Viliphonh Xayavong. A test for long-term cyclical clustering of stock market regimes. *Australian Journal of Management*, 32(2):205–221, 2007.

[Skl59] M Sklar. Fonctions de répartition à n dimensions et leurs marges. In *Annales de l'ISUP*, volume 8, pages 229–231, 1959.

[SS05] Bernhard Schölkopf and Alex Smola. Support vector machines and kernel algorithms. In *Encyclopedia of Biostatistics*, pages 5328–5335. Wiley, 2005.

[Tho19] Matthew Thorpe. Introduction to optimal transport. *Lecture Notes*, 3, 2019.

[TX15] Yingjie Tian and Dongkuan Xu. A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2:165–193, 2015.