

Dynamic Time Warping for Lead-Lag Relationship Detection in Lagged Multi-Factor Models

Yichi Zhang

Department of Statistics
Oxford-Man Institute of Quantitative Finance
University of Oxford
Oxford, United Kingdom
yichi.zhang@stats.ox.ac.uk

Mihai Cucuringu

Department of Statistics
Mathematical Institute
Oxford-Man Institute of Quantitative Finance
University of Oxford
The Alan Turing Institute
Oxford, United Kingdom
mihai.cucuringu@stats.ox.ac.uk

Alexander Y. Shestopaloff

School of Mathematical Sciences
Queen Mary University of London
Department of Mathematics and Statistics
Memorial University of Newfoundland
London, United Kingdom
a.shestopaloff@qmul.ac.uk

Stefan Zohren

Oxford-Man Institute of Quantitative Finance
Department of Engineering
University of Oxford
Oxford, United Kingdom
stefan.zohren@eng.ox.ac.uk

ABSTRACT

In multivariate time series systems, lead-lag relationships reveal dependencies between time series when they are shifted in time relative to each other. Uncovering such relationships is valuable in downstream tasks, such as control, forecasting, and clustering. By understanding the temporal dependencies between different time series, one can better comprehend the complex interactions and patterns within the system. We develop a cluster-driven methodology based on dynamic time warping for robust detection of lead-lag relationships in lagged multi-factor models. Since multivariate time series are ubiquitous in a wide range of domains, we demonstrate that our algorithm is able to robustly detect lead-lag relationships in financial markets, which can be subsequently leveraged in trading strategies with significant economic benefits.

KEYWORDS

Dynamic Time Warping; High-dimensional time series; Lead-lag relationships; Unsupervised learning; Clustering; Financial markets

1 INTRODUCTION

Natural physical systems often produce high-dimensional, nonlinear time series data, which are prevalent across various domains. Numerous contributions have been made to analyze such time series from different perspectives [8–11, 13, 23, 24, 29, 32]. For example, [10] explored financial time series and emphasized various statistical properties such as distributional characteristics, tail properties, and extreme fluctuations.

High-dimensional time series can offer valuable insights through the discovery of latent structures, such as lead-lag relationships. These relationships are commonly observed and play a significant role in the field of finance [2, 5, 7, 19, 21, 25, 31, 35, 36], the environment [12, 34], and biology [26]. For instance, [5] created a directed network to capture pairwise lead-lag relationships among

equity prices in the US market. The analysis revealed clusters with significant directed flow imbalance.

Dynamic time warping (DTW) is an algorithm to quantify similarities between two time series, even when they exhibit variations in speed, enabling the calculation of an optimal alignment between them [4, 6, 20, 28]. The versatility of DTW is evident in its application to diverse domains, such as financial markets [16, 17, 19, 30], bioinformatics [1, 14], and robotics [27]. For example, [19] introduced Multinomial Dynamic Time Warping (MDTW) to explore lead-lag relationships in FX market data. [17] applied DTW and found that global market price discovery oscillates between S&P 500 futures and FTSE 100 futures. Despite the widely recognized importance and potentially high impact of the problem, limited progress has been made in robustly using DTW for the inference of lead-lag relationships in lagged multi-factor models.

1.1 Summary of main contributions

We introduce a computationally scalable framework for lead-lag detection in high-dimensional time series based on DTW, with clustering used as a denoising step. We show it is capable of reliably detecting lead-lag relationships in a variety of factor model-based simulated high-dimensional time series. In financial markets, we leverage the detected lead-lag relationships to construct a profitable trading strategy and demonstrate that our algorithm outperforms the benchmark in most cases. Our algorithm is also faster than the benchmark in terms of computing time by a factor of 10.

2 DYNAMIC TIME WARPING

In this section, we introduce the DTW algorithm. Suppose we have two time series, denoted as A and B with lengths n and m , respectively, shown in Figure 1 Step 1.

$$\begin{aligned} A &= a_1, a_2, \dots, a_i, \dots, a_n \\ B &= b_1, b_2, \dots, b_j, \dots, b_m \end{aligned} \tag{1}$$

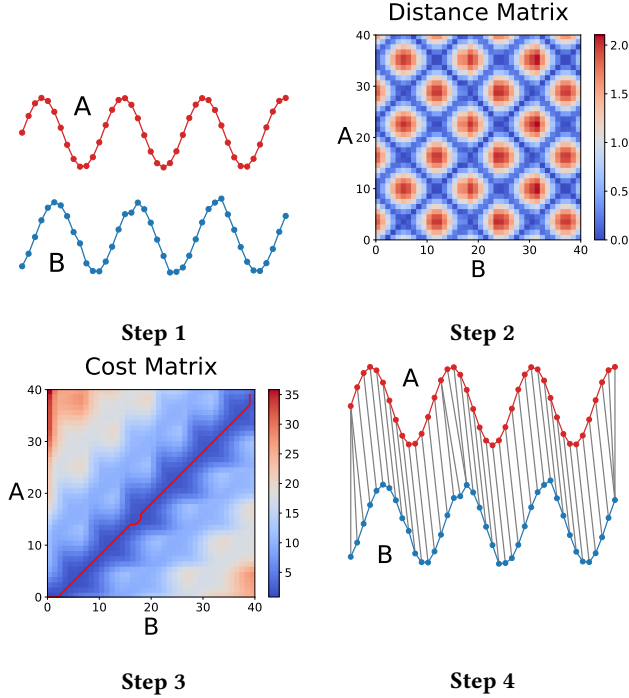


Figure 1: Step 1: Two time series A and B are similar, but they are out of phase with each other. Step 2: The distance matrix, which the $(i^{\text{th}}, j^{\text{th}})$ element of the matrix has the value of the Euclidean distance $d(a_i, b_j)$ from A and B , respectively. Step 3: The cost matrix, which contains the optimal warping path, is visually indicated by the red line. Step 4: The resulting alignment between the two time series, A and B , is shown.

In order to align two time series utilizing DTW, the process involves constructing an $n \times m$ matrix, where the $(i^{\text{th}}, j^{\text{th}})$ element of the matrix contains the Euclidean distance $d(a_i, b_j)$ between the two points a_i and b_j from A and B , respectively, i.e. $d(a_i, b_j) = (a_i - b_j)^2$. Each matrix element (i, j) corresponds to the alignment between the points a_i and b_j . Figure 1 Step 2 illustrates an example of the distance matrix.

A warping path, denoted as W , represents a consecutive set of matrix elements that capture the mapping between time series A and B . The index of the DTW can be expressed as follows

$$W = \{w_1, w_2, \dots, w_k, \dots, w_K\} \quad \max(m, n) \leq K < m + n - 1 \quad (2)$$

where the k^{th} element of W is denoted as $w_k = (i, j)_k$.

The warping path must adhere to several constraints:

- **Boundary:** The warping path starts at $w_1 = (1, 1)$ and ends at $w_K = (m, n)$. This ensures that the warping path initiates from the bottom-left corner cell and terminates at the top-right corner cell of the matrix.
- **Continuity:** For a given $w_k = (x, y)$, the preceding element $w_{k-1} = (x', y')$, where $x - x' \leq 1$ and $y - y' \leq 1$. This restriction allows only adjacent cell transitions within the warping path.
- **Monotonicity:** For a given $w_k = (x, y)$, the preceding element $w_{k-1} = (x', y')$, where $x - x' \geq 0$ and $y - y' \geq 0$. This

constraint ensures that the points in W exhibit monotonically increasing indices over time.

Among the various warping paths that adhere to the aforementioned constraints, our objective is to identify the path that minimizes the warping cost

$$\text{DTW}(A, B) = \min \left\{ \sqrt{\sum_{k=1}^K w_k} \right\} \quad (3)$$

To find this optimal path, dynamic programming techniques are employed to evaluate the following subsequent recurrence relation, which defines the cumulative distance $c(i, j)$ as the distance $d(i, j)$ located in the current cell as well as the minimum of the cumulative distances of the adjacent elements

$$c(i, j) = d(a_i, b_j) + \min\{c(i-1, j-1), c(i-1, j), c(i, j-1)\} \quad (4)$$

In Figure 1 Step 3, the optimal path is depicted by a red line. The resulting alignment can be seen in Figure 1 Step 4.

Note that the Euclidean distance between two time series can be considered as a specific case of DTW, where the warping path W is constrained such that $w_k = (i, j)_k$ with $i = j = k$. In other words, the window size S is set to 0. This constraint is applicable only when the two time series have the same length. A visual comparison of the Euclidean distance and DTW is presented in Figure 2.

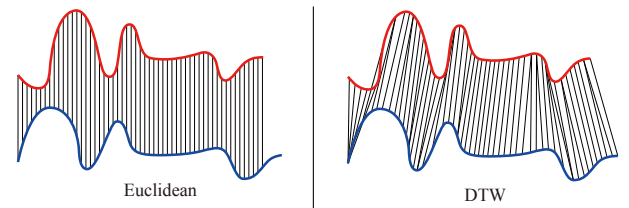


Figure 2: Left: Euclidean distance measures direct the i^{th} point in one time series aligns with the i^{th} point in another time series, assuming the same length, which will yield a pessimistic dissimilarity measure. Right: DTW accounts for variations in speed and calculates optimal alignment between two time series, allowing for different lengths, which is more flexible in calculating the intuitive distance measure.

3 MODEL SETUP

In this section, we will introduce both the standard and lagged versions of the *multi-factor model*, which we will adopt as the underlying model for our time series data. Specifically, we will employ the lagged multi-factor model to validate our algorithm using synthetic data before applying it to real-world scenarios. The fundamental idea behind these models is to represent a time series as a (noisy) combination of factors, with each factor exhibiting different levels of exposure. The model we propose for the detection of lead-lag relationships exhibits significant parallels with the problem of multireference alignment (MRA), which is concerned with the estimation of a single signal from a set of n cyclically and noisily shifted copies of itself, as shown in [3].

3.1 Description

Let us begin by revisiting the standard multi-factor model for a multivariate time series

$$X_i^t = \sum_{j=1}^k B_{ij} f_j^t + \epsilon_i^t \quad i = 1, \dots, n; \quad t = 1, \dots, T, \quad (5)$$

where X_i^t is the time series i (e.g., the excess return of a financial asset) at time t , k is the number of factors, B_{ij} is the exposure of time series i to factor j , f_j^t is the factor j at time t , and ϵ_i^t is the noise at time t , with variance σ^2 . Furthermore, we have n as the total number of time series, and T as the total number of time steps.

In this paper, our emphasis is on the lagged version of the multi-factor model, which can be expressed as follows

$$X_i^t = \sum_{j=1}^k B_{ij} f_j^{t-L_{ij}} + \epsilon_i^t \quad i = 1, \dots, n; \quad t = 1, \dots, T, \quad (6)$$

where the primary distinction in the lagged multi-factor model, in comparison to the standard multi-factor model, is the inclusion of L_{ij} , representing the lag at which time series i is exposed to factor j . Consequently, $f_j^{t-L_{ij}}$ corresponds to the value of factor j at time $t - L_{ij}$.

In the lagged multi-factor model (6), we consider two key settings, which are first introduced in [36]

- **Single Membership:** Each time series has a lagged exposure to a single factor. We consider the following two main categories.
 - **Homogeneous Setting:** The model only has one factor, i.e. $k = 1$.
 - **Heterogeneous Setting:** The model has more than one factor, i.e. $k \geq 2$. However, each time series is exposed only to a single factor.
- **Mixed Membership:** Each time series is permitted to have a lagged exposure to more than one factor, resulting in a mixed configuration. Therefore, the model comprises at least two factors, indicated by $k \geq 2$.

In this paper, our primary objective is to perform inference on the lag values L_{ij} in the lagged multi-factor model, specifically focusing on the single membership setting. We do not place emphasis on the inference of the unknown coefficient matrix B and factors f . As shown later, the estimation of L_{ij} alone holds practical significance in specific applications, such as finance. The investigation of the mixed membership setting is left for future work.

4 METHODOLOGY

In this section, we propose a robust algorithm for detecting lead-lag relationships using the combination of DTW and K-Medoids (DTW_KMed).

We consider a set of time series denoted as $X_{n \times T}$ as our input. Initially, we employ DTW to compute pairwise distances between every pair of time series from $X_{n \times T}$. Subsequently, we apply K-Medoids clustering to group similar time series into clusters based on the DTW distance matrix. The pairs of time series i and j are denoted as $\{X_i, X_j\}$. For each cluster ϕ_d ($d = 1, \dots, K$), we record the path $W\{X_i, X_j\}$ by performing DTW on $\{X_i, X_j\}$. Then, we

calculate the difference for each index pair in the path W , denoted as $\Delta(W\{X_i, X_j\})$. The calculation can be expressed as follows

$$\Delta(W\{X_i, X_j\}) = \{\Delta(w_1), \Delta(w_2), \dots, \Delta(w_k), \dots, \Delta(w_K)\} \quad (7)$$

where $\max(m, n) \leq K < m + n - 1$, and $\Delta(w_k) = \Delta((i, j)_k) = i - j$.

Then, the value of relative lags of $\{X_i, X_j\}$ in ϕ_d (estimated by mode or median) can be expressed as

$$\gamma\{X_i, X_j\} = \begin{cases} \text{Mode}(\Delta(W\{X_i, X_j\})) & \text{Mode estimation} \\ \text{Median}(\Delta(W\{X_i, X_j\})) & \text{Median estimation} \end{cases} \quad (8)$$

For instance, once we have applied K-medoids clustering using the DTW distance matrix computed from $X_{n \times T}$, let's consider two time series, X_1 and X_2 , that contain 100 data points and belong to the same cluster, with a known ground truth lag value of 3. Subsequently, we calculate the relative lags of $\Delta(W\{X_1, X_2\})$. Table 1 illustrates the relative lags obtained from $\{X_1, X_2\}$.

Lag	0	1	2	3
Counts	2	13	24	75

Table 1: The relative lags from two time series.

By employing mode or median estimation of $\Delta(W\{X_1, X_2\})$, denoted as $\gamma\{X_1, X_2\}$, with a value of 3, as evident from Table 1, we can conclude that the result remains consistent and robust with respect to the ground truth value. Despite the presence of outliers, such as $\{0, 1, 2\}$, in $\Delta(W\{X_i, X_j\})$, the estimated lag value aligns well with the ground truth value of 3.

Next step, the construction of the lead-lag matrix $\Gamma_{n \times n}$ is accomplished by

$$\Gamma_{ij} = \begin{cases} \gamma\{X_i, X_j\} & \text{if } X_i \text{ and } X_j \text{ are in the same cluster} \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

We summarize the above procedures in Algorithm 1.

Algorithm 1 DTW_KMed for Lead-lag Relationship Detection

Input: Time series matrix $X_{n \times T}$. **Output:** Lead-lag matrix $\Gamma_{n \times n}$.

- 1: Compute the DTW distance matrix from $X_{n \times T}$.
 - 2: Apply K-medoids clustering using the DTW distance matrix.
 - 3: For each cluster ϕ_d ($d = 1, \dots, K$):
 - Record path W from every pair of time series $\{X_i, X_j\}$.
 - Calculate the difference $\Delta(W\{X_i, X_j\})$ for every index pair in the path W
 - Record lag $\gamma\{X_i, X_j\}$ by taking *mode* or *median* estimation of the difference $\Delta(W\{X_i, X_j\})$.
 - 4: Calculate the lead-lag matrix $\Gamma_{n \times n}$ by setting $\gamma\{X_i, X_j\}$, where $\{X_i, X_j\}$ is in the same cluster. Otherwise, the entry is set to 0.
-

5 SYNTHETIC DATA EXPERIMENTS

The synthetic data experiments serve the purpose of simulating data generated by a multi-factor model with a known ground truth lead-lag matrix L . The main objective is to evaluate and validate the performance of our proposed algorithms under various scenarios.

5.1 Synthetic data generating process

As noted earlier, our focus is the single membership setting. We generate synthetic data from the lagged multi-factor model (6) with $k = \{1, 2\}$ factors. Furthermore, we specify the maximum number of lags as $M = 5$ and the length of the time series as $T = 100$. The factors f and noise ϵ are drawn from $\mathcal{N}(0, 1)$. We define B and L as follows:

Homogeneous Setting ($k = 1$)		Heterogeneous Setting ($k = 2$)	
B	L	B	L
$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 2 & 0 \\ 4 & 0 \\ 0 & 0 \\ 0 & 2 \\ 0 & 4 \end{bmatrix}$

Table 2: For each setting: Left: Loading matrix B . Right: Lag matrix L .

For validation purposes, we set the number of time series as $n = 120$ to demonstrate the effectiveness of our algorithms. When estimating the lead-lag matrix, we use a sliding window of length $l = 21$ and a shift of $s = 1$. After estimating the lead-lag matrix, we calculate the error matrix E to evaluate the performance, which we denote as

$$E_{n \times n} = \Gamma_{n \times n} - \Psi_{n \times n}, \quad (10)$$

where $\Gamma_{n \times n}$ is the estimated lead-lag matrix, and $\Psi_{n \times n}$ is the ground truth lead-lag matrix, which can be obtained from $L_{n \times k}$.

5.2 Simulation results

In the homogeneous setting ($k = 1$), as shown in Figure 3, all five algorithms (KM as K-Means, Euc_KMed as Euclidean + K-Medoids, Man_KMed as Manhattan + K-Medoids, Cos_KMed as Cosine + K-Medoids, and DTW_KMed as DTW + K-Medoids) demonstrate optimal performance with an Adjusted Rand Index (ARI) of 1. This indicates that they successfully detect the lead-lag relationships with high accuracy when there is only one underlying factor, and this result is expected based on our experimental setup. However, in the heterogeneous setting ($k = 2$), we observe a general decrease in ARI as the noise level σ increases. Despite this trend, our proposed DTW_KMed algorithm achieves a consistently high ARI value within the range of σ from 0 to 1.5. This performance surpasses the other algorithms, which maintain relatively lower ARI values across the noise levels. This observation highlights that DTW excels in capturing intricate lead-lag patterns with a higher level of robustness.

In DTW, the window size S defines the maximum allowed shifts from the two diagonals smaller than this number. Hence, selecting the appropriate window size S is of critical importance in DTW. The choice of S affects how much temporal distortion is allowed between time series, thereby influencing the alignment and capturing the underlying lead-lag relationships effectively. Properly tuning S enables DTW to strike a balance between capturing complex patterns while avoiding excessive warping that might lead to misclassification.

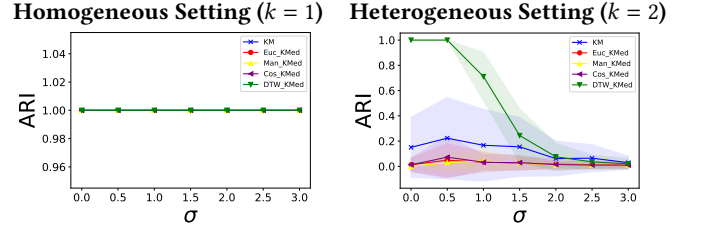


Figure 3: Average and confidence interval for the ARI with different σ levels based on 100 simulations for every iteration.

Across the synthetic experiments, we set the true lag equal to 5. In Figure 4, we observe similar results to the homogeneous setting in Figure 3. However, in the heterogeneous setting, the ARI tends to be relatively lower when the window size ranges from 0 to 5. Conversely, when the window size equals or exceeds 5, the ARI increases significantly.

This behaviour aligns with the rationale that DTW requires a window size that equals or exceeds the true lag of 5. This enables DTW to calculate the optimal alignment with enough flexibility to capture the actual lead-lag relationship effectively. When the window size is too restrictive, it may not allow for sufficient temporal distortion, leading to suboptimal alignment and lower ARI values in the presence of multiple factors. As such, selecting an appropriate window size is crucial to ensure accurate lead-lag relationship detection with DTW.

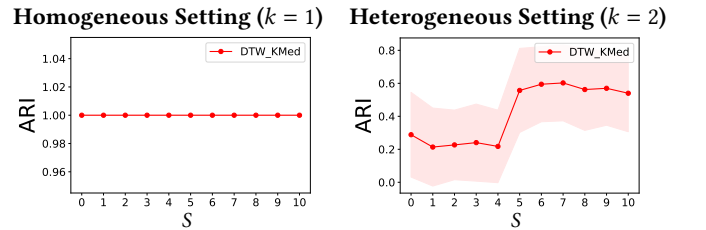


Figure 4: Average and confidence interval for the ARI with different window sizes S based on 100 simulations for every iteration (the true lag is 5).

As depicted in Figure 5, for both the homogeneous and heterogeneous settings, the Mean Squared Error (MSE) remains close to 0 when σ ranges from 0.0 to 0.5. However, as σ increases beyond 0.5, the MSE rises significantly for both algorithms. Overall, both algorithms demonstrate acceptable performance for low noise levels, but the DTW_KMed mode estimation (DTW_KMed_Mod) exhibits better performance across varying levels of noise than DTW_KMed median estimation (DTW_KMed_Med).

With the true lag set as 5 and σ as 1, the DTW_KMed_Mod algorithm outperforms the DTW_KMed_Med algorithm in the homogeneous setting ($k = 1$). In the heterogeneous setting ($k = 2$), when the window size S ranges from 0 to 5, the MSE is higher for both algorithms. However, as the window size S exceeds 5, both algorithms achieve a lower MSE. This behaviour aligns with the understanding that DTW requires a S that equals or exceeds the true lag of 5 to effectively calculate the optimal alignment. Thus, when the S is less than 5, the alignment may not fully capture the

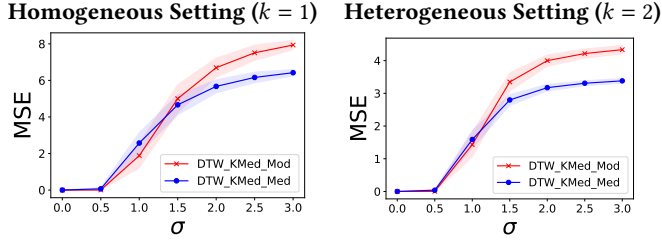


Figure 5: Average and confidence interval for the MSE with different σ levels based on 100 simulations for every iteration.

true lead-lag relationship, resulting in higher MSE values. However, when the S is larger than or equal to 5, both algorithms achieve better alignment, leading to reduced MSE values.

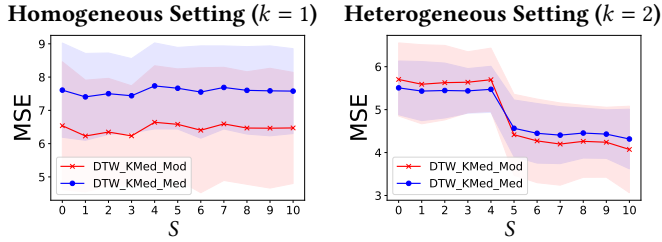


Figure 6: Average and confidence interval for the MSE with different window sizes S based on 100 simulations for every iteration (the true lag is 5, and σ is 1).

6 FINANCIAL DATA EXPERIMENTS

6.1 Data description

In this section, we conduct a large-scale experiment using financial data to apply our algorithms. As mentioned earlier, this is a context where lead-lag relationships naturally occur. For our financial data experiments, we consider three different data sets, each varying in terms of the number and type of assets, as well as the number of days included in the data set. All the data sets are considered at a daily frequency. The summarized details of the data sets are presented in Table 3, and for more detailed information, see [36].

Data source	Type	Freq	# of assets	Start date	End date	# of days
Wharton's CRSP	Equity	Daily	679	2000/01/03	2019/12/31	5211
Wharton's CRSP	ETF	Daily	14	2006/04/12	2019/07/01	3324
Pinnacle Data Corp	Futures	Daily	52	2000/01/05	2020/10/16	5166

Table 3: Summary of the three financial data sets considered in the numerical experiments.

6.2 Data pre-processing

With regard to the US equity and ETF data sets, we use the close-to-close adjusted daily returns from Wharton's CRSP. Due to the large number of NaNs in the equity data set, we drop the days for which more than 10% of the equities have zero returns as well as the equities for which more than 50% of days have zero returns. Instead of working with raw returns, we consider the *market excess returns*, a standard measure of how well each equity performed relative to the broader market. For both of these data sets, the return of

the S&P Composite Index is selected to compute the market excess returns by subtracting it from the return of each asset (i.e., for simplicity, we assume each asset has $\beta = 1$ exposure to the market). Also, we winsorize the extreme value of excess returns for which any value is larger than 0.15 or smaller than -0.15.

For the futures data set, we use the close-to-close price series from the Pinnacle Data Corp CLC data set, and discard the days for which more than 10% of the futures have zero prices in the respective dates, and drop the futures for which more than 160 days have zero prices. Afterwards, we first use forward-fill, then backward-fill to fill out the zero prices. Lastly, we compute the log-return from the close-to-close price. The remainder of the data pre-processing is the same as above.

6.3 Benchmark

In order to evaluate our proposed methodology, we also introduce a benchmark to detect lead-lag relationships without the use of clustering. It is very common to compute a sample cross-correlation function (CCF) between two time series. Also, we use four algorithms (KM_Mod, KM_Med, SP_Mod, and SP_Med) as our benchmark from [36].

6.4 Trading strategies

In this section, we present the trading strategies employed in this paper. Our approach involves a series of steps applied to a dataset consisting of n time series, each having a length of T . Firstly, we extract the data by implementing a sliding window approach with a fixed length of $l = 21$. Subsequently, we employ the DTW_KMed algorithm to detect the lead-lag relationship, which is further validated through a synthetic data experiment. Once the lead-lag matrix is obtained, we then utilize the lead-lag matrix to rank the time series from the most leading to the most lagging using the *Row-Sum* ranking [15, 18], in order to then group the time series into leaders and laggards, where the leaders are employed to forecast the behaviour of the laggards.

In our trading strategy, we identify the top $\alpha = 0.75$ fraction of the time series as Leaders D_α , while the remaining bottom fraction $\beta = 1 - \alpha$ is classified as Laggards G_β . To predict the future performance, we employ the exponentially weighted moving average (EWMA) signal, considering the past $p = \{1, 3, 5, 7\}$ days of average winsorized time series excess returns from D_α . This prediction aims to estimate the average excess returns of G_β and D_α in the subsequent $\delta = \{1, 3, 5, 7\}$ days. We assume that G_β can catch up with D_α , while D_α provides the necessary momentum to sustain the trend over the δ days. To ensure continuous trading, we shift the sliding window by $h = 1$ and repeat the lead-lag matrix calculation and ranking steps until the end of the time series. Figure 7 provides a depiction of our trading pipeline at time t .

6.5 Performance evaluation

When assessing the effectiveness of various trading strategies, we rely on the following metrics to evaluate their performance. We compute the Profit and Loss (PnL) of G_β on a given day $t + \delta$ as

$$\text{PnL}_{G_\beta}^{t+\delta} = \text{sign}(\text{EWMA}(\text{ret}_{D_\alpha}^{t-p} : \text{ret}_{D_\alpha}^t)) \cdot \overline{\text{ret}_{G_\beta}^{t+\delta}}, t = l, \dots, T - \delta, \quad (11)$$

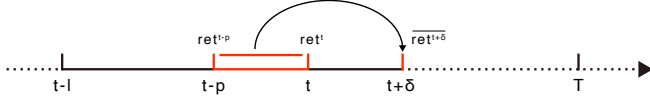


Figure 7: Illustration of the trading pipeline at time t , given the EWMA of past p days of average winsorized time series excess returns to predict future δ days average excess returns.

since the strategy makes profits whenever the sign of the forecast agrees with the sign of the future return. Correspondingly, the PnL of D_α on a given day $t + \delta$ is given by

$$\text{PnL}_{D_\alpha}^{t+\delta} = \text{sign}(\text{EWMA}(ret_{D_\alpha}^{t-p} : ret_{D_\alpha}^t)) \cdot \overline{ret_{D_\alpha}^{t+\delta}}, t = l, \dots, T - \delta, \quad (12)$$

where $ret_{D_\alpha}^{t-p}$ and $ret_{D_\alpha}^t$ are the excess return of D_α at $t - p$ and t , respectively, while $\text{EWMA}(ret_{D_\alpha}^{t-p} : ret_{D_\alpha}^t)$ denotes the exponentially weighted moving average from the excess return of D_α from $t - p$ to t . Furthermore, $\overline{ret_{G_\beta}^{t+\delta}}$ depicts the mean of the excess return of G_β at $t + \delta$, and $\overline{ret_{D_\alpha}^{t+\delta}}$ is the mean of the excess return of D_α at $t + \delta$.

We rescale the PnL by their volatility to target equal risk assignment, and set our annualized volatility target σ_{tgt} to be 0.15.

$$\text{PnL}_{\text{rescaled}} = \frac{\sigma_{\text{target}}}{\text{STD}(\text{PnL}) \cdot \sqrt{252}} \cdot \text{PnL}. \quad (13)$$

Based on $\text{PnL}_{\text{rescaled}}$, we proceed to calculate the following annualized metrics, in line with the works of [22, 33, 36].

- **Profitability:** cumulative PnL, annualized expected excess return ($E[\text{Returns}]$), hit rate.
- **Risk:** volatility, downside deviation, maximum drawdown.
- **Performance:** Sortino ratio, Calmar ratio, average profit / average loss, PnL per trade, Sharpe ratio, P-value.

6.6 Results

In the case of the equity data set, our findings indicate that utilizing the EWMA on the past seven days of the average winsorized time series excess returns of the D_α , with $\alpha = 0.75$, for predicting the average future seven days of the excess return of the G_β and D_α consistently yields favourable performance across all algorithms. Figure 8 presents a comparison of cumulative PnL for the G_β strategy (left) and D_α strategy (right). In the G_β strategy, before 2008, the Sharpe ratio (SR) of DTW_KMed_Mod and DTW_KMed_Med outperformed the other algorithms. However, after 2008, except for CCF, all algorithms displayed a substantial growth trend and eventually achieved similar performance levels. On the other hand, in the D_α strategy, before 2008, all algorithms performed at roughly the same level. However, after 2008, DTW_KMed_Med emerged as the most profitable strategy, clearly outperforming others with an SR of 0.93. Additionally, Tables [4, 5] present the performance of DTW_KMed_Mod, DTW_KMed_Med, and other algorithms for the G_β strategy and D_α strategy based on various metrics (rescaled to target volatility).

Tables [6, 7] along with Figure 9 present the results for the ETF data set using the same settings as for the equity data. In the G_β strategy, we do not find evidence of consistently detecting lead-lag relationships that lead to a profitable outcome. However, in the D_α strategy, the SR of SP_Med and DTW_KMed_Mod are leading with SR values of 0.8 and 0.78, respectively.

Results for the futures data, using the same settings as for the equity data, are presented in Tables [8, 9], along with Figure 10. For this data set, we do not observe the ability to consistently detect profitable lead-lag relationships for any of the strategies.

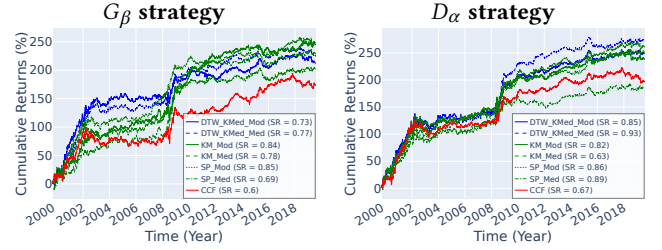


Figure 8: Equity data set: cumulative PnL for G_β strategy (left) and D_α strategy (right) - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

G_β strategy	Benchmark					Proposed	
	CCF	KM_Mod	KM_Med	SP_Mod	SP_Med	DTW_KMed_Mod	DTW_KMed_Med
E[Returns]	0.089	0.126	0.118	0.127*	0.104	0.109	0.115
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.105	0.103	0.101*	0.103	0.105	0.106	0.107
Maximum drawdown	-0.313	-0.26	-0.215	-0.214	-0.287	-0.205	-0.188*
Sortino ratio	0.85	1.222	1.166	1.237*	0.988	1.026	1.084
Calmar ratio	0.285	0.484	0.548	0.594	0.362	0.531	0.614*
Hit rate	0.499	0.521*	0.51	0.516	0.519	0.514	0.512
Avg. profit / avg. loss	1.117*	1.068	1.107	1.091	1.051	1.08	1.1
PnL per trade	3.542	4.996	4.672	5.041*	4.119	4.317	4.581
Sharpe ratio	0.595	0.839	0.785	0.847*	0.692	0.725	0.77
P-value	0.009	0*	0*	0*	0.002	0.001	0.001

Table 4: Equity data set: performance metrics for G_β strategy - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

G_β strategy	Benchmark					Proposed	
	CCF	KM_Mod	KM_Med	SP_Mod	SP_Med	DTW_KMed_Mod	DTW_KMed_Med
E[Returns]	0.101	0.122	0.095	0.129	0.134	0.128	0.14*
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.105	0.107	0.108	0.106	0.105	0.103	0.102*
Maximum drawdown	-0.288	-0.251	-0.21*	-0.283	-0.227	-0.276	-0.297
Sortino ratio	0.964	1.148	0.876	1.222	1.269	1.238	1.376*
Calmar ratio	0.352	0.488	0.452	0.456	0.59*	0.463	0.472
Hit rate	0.518	0.521	0.513	0.522	0.525*	0.52	0.523
Avg. profit / avg. loss	1.05	1.069	1.066	1.072	1.065	1.078*	1.078*
PnL per trade	4.018	4.856	3.765	5.121	5.312	5.073	5.557*
Sharpe ratio	0.675	0.816	0.632	0.86	0.892	0.852	0.934*
P-value	0.003	0*	0.005	0*	0*	0*	0*

Table 5: Equity data set: performance metrics for D_α strategy - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

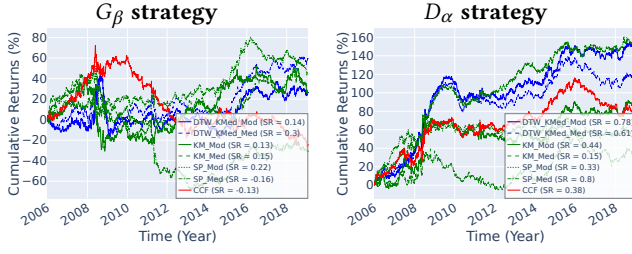


Figure 9: ETF data set: cumulative PnL for G_β strategy (left) and D_α strategy (right) - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

G_β strategy	Benchmark					Proposed	
	CCF	KM_Mod	KM_Med	SP_Mod	SP_Med	DTW_KMod_Mod	DTW_KMod_Med
E[Returns]	-0.019	0.02	0.022	0.033	-0.024	0.021	0.045
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.116	0.115	0.116	0.113	0.123	0.111	0.115
Maximum drawdown	-0.668	-0.525	-0.465	-0.38	-0.653	-0.507	-0.428
Sortino ratio	-0.165	0.176	0.188	0.294	-0.192	0.186	0.394
Calmar ratio	-0.029	0.038	0.047	0.088	-0.036	0.041	0.106
Hit rate	0.492	0.512	0.499	0.509	0.509	0.512	0.513
Avg. profit / avg. loss	1.006	0.978	1.034	1.009	0.935	0.98	1.005
PnL per trade	-0.76	0.8	0.87	1.325	-0.939	0.818	1.799
Sharpe ratio	-0.128	0.134	0.146	0.223	-0.158	0.137	0.302
P-value	0.644	0.628	0.598	0.421	0.568	0.619	0.275

Table 6: ETF data set: performance metrics for G_β strategy - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

G_β strategy	Benchmark					Proposed	
	CCF	KM_Mod	KM_Med	SP_Mod	SP_Med	DTW_KMod_Mod	DTW_KMod_Med
E[Returns]	0.056	0.065	0.022	0.05	0.0121	0.117	0.091
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.097	0.097	0.105	0.104	0.098	0.093	0.095
Maximum drawdown	-0.362	-0.267	-0.385	-0.298	-0.22	-0.256	-0.31
Sortino ratio	0.581	0.676	0.213	0.474	1.236	1.257	0.955
Calmar ratio	0.156	0.245	0.058	0.166	0.548	0.456	0.293
Hit rate	0.504	0.502	0.5	0.508	0.513	0.521	0.517
Avg. profit / avg. loss	1.056	1.077	1.027	1.03	1.097	1.056	1.039
PnL per trade	2.234	2.591	0.889	1.965	4.784	4.635	3.608
Sharpe ratio	0.375	0.435	0.149	0.33	0.804	0.779	0.606
P-value	0.169	0.105	0.588	0.229	0.003	0.004	0.026

Table 7: ETF data set: performance metrics for D_α strategy - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

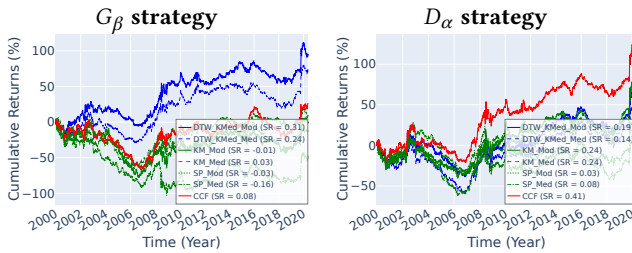


Figure 10: Futures data set: cumulative PnL for G_β strategy (left) and D_α strategy (right) - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

G_β strategy	Benchmark					Proposed	
	CCF	KM_Mod	KM_Med	SP_Mod	SP_Med	DTW_KMod_Mod	DTW_KMod_Med
E[Returns]	0.013	-0.001	0.004	-0.005	-0.024	0.047	0.036
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.104	0.102	0.103	0.107	0.109	0.103	0.104
Maximum drawdown	-0.535	-0.587	-0.538	-0.544	-0.698	-0.316	-0.376
Sortino ratio	0.121	-0.01	0.042	-0.049	-0.217	0.453	0.347
Calmar ratio	0.024	-0.002	0.008	-0.01	-0.034	0.147	0.096
Hit rate	0.502	0.498	0.494	0.497	0.497	0.506	0.505
Avg. profit / avg. loss	1.007	1.008	1.028	1.006	0.984	1.033	1.023
PnL per trade	0.499	-0.039	0.173	-0.206	-0.94	1.848	1.427
Sharpe ratio	0.084	-0.007	0.029	-0.035	-0.158	0.31	0.24
P-value	0.705	0.976	0.896	0.876	0.476	0.159	0.278

Table 8: Futures data set: performance metrics for G_β strategy - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

G_β strategy	Benchmark					Proposed	
	CCF	KM_Mod	KM_Med	SP_Mod	SP_Med	DTW_KMod_Mod	DTW_KMod_Med
E[Returns]	0.061	0.036	0.036	0.005	0.012	0.029	0.021
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.107	0.108	0.108	0.11	0.105	0.107	0.107
Maximum drawdown	-0.393	-0.412	-0.404	-0.581	-0.524	-0.424	-0.502
Sortino ratio	0.574	0.337	0.331	0.042	0.112	0.269	0.193
Calmar ratio	0.156	0.088	0.089	0.008	0.022	0.068	0.041
Hit rate	0.511	0.502	0.499	0.491	0.496	0.498	0.501
Avg. profit / avg. loss	1.032	1.038	1.05	1.042	1.032	1.042	1.022
PnL per trade	2.439	1.445	1.424	0.183	0.468	1.139	0.822
Sharpe ratio	0.41	0.243	0.239	0.031	0.079	0.191	0.138
P-value	0.064	0.272	0.28	0.889	0.723	0.387	0.533

Table 9: Futures data set: performance metrics for D_α strategy - rescaled to target volatility. The experiment has been set with the values $p = 7$, $\delta = 7$, $\alpha = 0.25$, and $K = 5$.

7 ROBUSTNESS ANALYSIS

We test the robustness of the DTW_KMod_Mod and DTW_KMod_Med by conducting experiments with different numbers of clusters K . Specifically, for the equity data set, we consider $K = \{5, 10, 15, 20\}$. In Table 10, we observe that the performance of both algorithms does not change significantly while maintaining a high SR. It is worth noting that the P-values are almost all lower than 0.05, indicating that all the results are statistically significant in our experiments.

Table 11 presents the performance of the DTW_KMod_Mod and DTW_KMod_Med tested on the ETF data set. Due to the smaller cross-section for this data set, we only consider the K values of 5 and 10. Both algorithms demonstrate fairly good performance for high alpha in the D_α strategy.

Table 12 presents the performance of the DTW_KMod_Mod and DTW_KMod_Med on the futures data set, with K ranging from 5 to 20 in increments of 5. It is observed that for this data set, the SR for the D_α strategy tends to be more sensitive to changes in K , and achieving profitability becomes challenging.

DTW_KMod_Mod	G_β strategy				D_α strategy			
	5	10	15	20	5	10	15	20
K								
E[Returns]	0.109	0.098	0.098	0.096	0.128	0.106	0.114	0.101
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.106	0.107	0.108	0.108	0.103	0.105	0.105	0.105
Maximum drawdown	-0.205	-0.192	-0.198	-0.235	-0.276	-0.27	-0.221	-0.272
Sortino ratio	1.026	0.92	0.908	0.891	1.238	1.007	1.084	0.961
Calmar ratio	0.531	0.512	0.497	0.408	0.463	0.392	0.514	0.372
Hit rate	0.514	0.512	0.516	0.514	0.52	0.519	0.522	0.519
Avg. profit / avg. loss	1.08	1.075	1.058	1.066	1.078	1.051	1.05	1.044
PnL per trade	4.317	3.9	3.906	3.805	5.073	4.201	4.511	4.012
Sharpe ratio	0.725	0.655	0.656	0.639	0.852	0.706	0.758	0.674
P-value	0.001	0.004	0.004	0.005	0	0.002	0.001	0.003

DTW_KMod_Med	G_β strategy				D_α strategy			
	5	10	15	20	5	10	15	20
K								
E[Returns]	0.115	0.097	0.083	0.092	0.14	0.116	0.099	0.112
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.107	0.107	0.109	0.109	0.102	0.104	0.105	0.104
Maximum drawdown	-0.188	-0.197	-0.226	-0.21	-0.297	-0.226	-0.241	-0.274
Sortino ratio	1.084	0.907	0.76	0.844	1.376	1.112	0.946	1.083
Calmar ratio	0.614	0.493	0.367	0.437	0.472	0.513	0.413	0.41
Hit rate	0.512	0.513	0.509	0.513	0.523	0.519	0.514	0.517
Avg. profit / avg. loss	1.1	1.069	1.07	1.065	1.078	1.067	1.067	1.068
PnL per trade	4.581	3.851	3.293	3.646	5.557	4.596	3.948	4.459
Sharpe ratio	0.77	0.647	0.553	0.612	0.934	0.772	0.663	0.749
P-value	0.001	0.004	0.015	0.007	0	0.001	0.003	0.001

Table 10: Equity data set: robustness analysis for K - rescaled to target volatility. The experiment has been set with the values $p = 7$, and $\delta = 7$.

DTW_KMod_Mod	G_β strategy		D_α strategy	
	5	10	5	10
K				
E[Returns]	0.021	0.058	0.117	0.07
Volatility	0.15	0.15	0.15	0.15
Downside deviation	0.111	0.116	0.093	0.098
Maximum drawdown	-0.507	-0.353	-0.256	-0.321
Sortino ratio	0.186	0.505	1.257	0.715
Calmar ratio	0.041	0.166	0.456	0.219
Hit rate	0.512	0.506	0.521	0.505
Avg. profit / avg. loss	0.98	1.059	1.056	1.067
PnL per trade	0.818	2.321	4.635	2.783
Sharpe ratio	0.137	0.39	0.779	0.468
P-value	0.619	0.161	0.004	0.088

DTW_KMod_Med	G_β strategy		D_α strategy	
	5	10	5	10
K				
E[Returns]	0.045	0.063	0.091	0.061
Volatility	0.15	0.15	0.15	0.15
Downside deviation	0.115	0.113	0.095	0.098
Maximum drawdown	-0.428	-0.415	-0.31	-0.388
Sortino ratio	0.394	0.561	0.955	0.621
Calmar ratio	0.106	0.153	0.293	0.157
Hit rate	0.513	0.508	0.517	0.507
Avg. profit / avg. loss	1.005	1.054	1.039	1.045
PnL per trade	1.799	2.514	3.608	2.413
Sharpe ratio	0.302	0.422	0.606	0.405
P-value	0.275	0.128	0.026	0.14

Table 11: ETF data set: robustness analysis for K - rescaled to target volatility. The experiment has been set with the values $p = 7$, and $\delta = 7$.

8 CONCLUSION AND FUTURE WORK

In this study, we introduce a Dynamic Time Warping (DTW) based approach for robustly detecting lead-lag relationships in high-dimensional multivariate time series, with a specific focus on lagged multi-factor models. Our proposed algorithms show promising Sharpe Ratios when applied to financial data sets, indicating their potential economic benefits compared to the benchmark.

To enhance the methodology further, a possible future direction could involve exploring dynamic selection of the number of clusters

DTW_KMod_Mod	G_β strategy				D_α strategy			
	5	10	15	20	5	10	15	20
K								
E[Returns]	0.047	0.051	0.064	0.055	0.029	0.042	0.048	0.04
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.103	0.102	0.103	0.104	0.107	0.105	0.105	0.105
Maximum drawdown	-0.316	-0.412	-0.421	-0.338	-0.424	-0.468	-0.459	-0.455
Sortino ratio	0.453	0.496	0.618	0.528	0.269	0.395	0.457	0.378
Calmar ratio	0.147	0.123	0.152	0.163	0.068	0.089	0.105	0.087
Hit rate	0.506	0.5	0.501	0.508	0.498	0.5	0.5	0.499
Avg. profit / avg. loss	1.033	1.065	1.08	1.04	1.042	1.05	1.058	1.055
PnL per trade	1.848	2.015	2.534	2.19	1.139	1.649	1.908	1.577
Sharpe ratio	0.31	0.338	0.426	0.368	0.191	0.277	0.321	0.265
P-value	0.159	0.124	0.053	0.095	0.387	0.21	0.147	0.231

DTW_KMod_Med	G_β strategy				D_α strategy			
	5	10	15	20	5	10	15	20
K								
E[Returns]	0.036	0.05	0.061	0.055	0.021	0.033	0.053	0.039
Volatility	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Downside deviation	0.104	0.104	0.105	0.106	0.107	0.105	0.106	0.105
Maximum drawdown	-0.376	-0.376	-0.428	-0.365	-0.502	-0.465	-0.475	-0.479
Sortino ratio	0.347	0.48	0.583	0.521	0.193	0.313	0.505	0.374
Calmar ratio	0.096	0.133	0.143	0.151	0.041	0.071	0.112	0.082
Hit rate	0.505	0.502	0.505	0.506	0.501	0.499	0.504	0.499
Avg. profit / avg. loss	1.023	1.055	1.057	1.048	1.022	1.045	1.052	1.054
PnL per trade	1.427	1.978	2.436	2.184	0.822	1.305	2.115	1.556
Sharpe ratio	0.24	0.332	0.409	0.367	0.138	0.219	0.355	0.261
P-value	0.278	0.132	0.063	0.096	0.533	0.321	0.108	0.237

Table 12: Futures data set: robustness analysis for K - rescaled to target volatility. The experiment has been set with the values $p = 7$, and $\delta = 7$.

K . Additionally, another interesting direction would be to delve into the more intricate mixed membership model described in Section 3, which poses a more challenging task. Investigating intraday lead-lag relationships using, for example, minutely data, could be another fruitful area of research.

REFERENCES

- [1] John Aach and George M Church. 2001. Aligning gene expression time series with time warping algorithms. *Bioinformatics* 17, 6 (2001), 495–508.
- [2] Jakob Albers, Mihai Cucuringu, Sam Howison, and Alexander Y Shestopaloff. 2021. Fragmentation, price formation and cross-impact in bitcoin markets. *Applied Mathematical Finance* 28, 5 (2021), 395–448.
- [3] Afonso S Bandeira, Moses Charikar, Amit Singer, and Andy Zhu. 2014. Multireference alignment using semidefinite programming. In *Proceedings of the 5th conference on Innovations in theoretical computer science*. 459–470.
- [4] Richard Bellman and Robert Kalaba. 1959. On adaptive control processes. *IRE Transactions on Automatic Control* 4, 2 (1959), 1–9.
- [5] Stefanos Bennett, Mihai Cucuringu, and Gesine Reinert. 2022. Lead-lag detection and network clustering for multivariate time series with an application to the US equity market. *Machine Learning* 111, 12 (2022), 4497–4538.
- [6] Donald J Berndt and James Clifford. 1994. Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd international conference on knowledge discovery and data mining*. 359–370.
- [7] Giuseppe Buccheri, Fulvio Corsi, and Stefano Peluso. 2021. High-frequency lead-lag effects and cross-asset linkages: a multi-asset lagged adjustment model. *Journal of Business & Economic Statistics* 39, 3 (2021), 605–621.
- [8] Jay Cao, Jacky Chen, and John Hull. 2020. A neural network approach to understanding implied volatility movements. *Quantitative Finance* 20, 9 (2020), 1405–1413.
- [9] Alvaro Cartea, Ryan Donnelly, and Sebastian Jaimungal. 2018. Enhancing trading strategies with order book signals. *Applied Mathematical Finance* 25, 1 (2018), 1–35.
- [10] Rama Cont. 2001. Empirical properties of asset returns: stylized facts and statistical issues. *Quantitative finance* 1, 2 (2001), 223.
- [11] Yan Cui, Jun Yang, and Zhou Zhou. 2021. State-domain change point detection for nonlinear time series regression. *Journal of Econometrics* (2021).
- [12] Giovanni De Luca and Federica Pizzolante. 2021. Detecting Leaders Country from Road Transport Emission Time-Series. *Environments* 8, 3 (2021), 18.
- [13] Felix Drinkall, Stefan Zohren, and Janet B Pierrehumbert. 2022. Forecasting COVID-19 Caseloads Using Unsupervised Embedding Clusters of Social Media Posts. *arXiv:2205.10408* (2022).
- [14] Dariu M Gavrilă, Larry S Davis, et al. 1995. Towards 3-d model-based tracking and recognition of human movement: a multi-view approach. In *International workshop on automatic face-and gesture-recognition*, Vol. 3. Citeseer, 272–277.

- [15] David F Gleich and Lek-heng Lim. 2011. Rank aggregation via nuclear norm minimization. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. 60–68.
- [16] Kartikay Gupta and Niladri Chatterjee. 2020. Examining lead-lag relationships in-depth, with focus on FX market as Covid-19 crises unfolds. *arXiv:2004.10560* (2020).
- [17] Clint Howard, Talis J Putninš, and Vitali Alexeev. 2022. To lead or to lag? Measuring asynchronicity in financial time-series using dynamic time warping. (2022).
- [18] Peter J Huber. 1963. Pairwise comparison and ranking: optimum properties of the row sum procedure. *The annals of mathematical statistics* (1963), 511–520.
- [19] Katsuya Ito and Ryuta Sakemoto. 2020. Direct estimation of lead-lag relationships using multinomial dynamic time warping. *Asia-Pacific Financial Markets* 27, 3 (2020), 325–342.
- [20] Eamonn Keogh and Chotirat Ann Ratanamahatana. 2005. Exact indexing of dynamic time warping. *Knowledge and information systems* 7 (2005), 358–386.
- [21] Yongli Li, Chao Liu, Tianchen Wang, and Baiqing Sun. 2021. Dynamic patterns of daily lead-lag networks in stock markets. *Quantitative Finance* 21, 12 (2021), 2055–2068.
- [22] Bryan Lim, Stefan Zohren, and Stephen Roberts. 2019. Enhancing time-series momentum strategies using deep neural networks. *The Journal of Financial Data Science* 1, 4 (2019), 19–38.
- [23] Yutong Lu, Gesine Reinert, and Mihai Cucuringu. 2022. Trade co-occurrence, trade flow decomposition, and conditional order imbalance in equity markets. *arXiv:2209.10334* (2022).
- [24] Yutong Lu, Gesine Reinert, and Mihai Cucuringu. 2023. Co-trading networks for modeling dynamic interdependency structures and estimating high-dimensional covariances in US equity markets. *arXiv:2302.09382* (2023).
- [25] Deborah Miori and Mihai Cucuringu. 2022. Returns-Driven Macro Regimes and Characteristic Lead-Lag Behaviour between Asset Classes. *arXiv:2209.00268* (2022).
- [26] Jakob Runge, Peer Nowack, Marlene Kretschmer, Seth Flaxman, and Dino Sejdic. 2019. Detecting and quantifying causal associations in large nonlinear time series datasets. *Science advances* 5, 11 (2019), eaau4996.
- [27] Matthew D Schmill, Tim Oates, and Paul R Cohen. 1999. Learned models for continuous planning. In *AISTATS*. Citeseer.
- [28] Pavel Senin. 2008. Dynamic time warping algorithm review. *Information and Computer Science Department University of Hawaii at Manoa Honolulu, USA* 855, 1-23 (2008), 40.
- [29] Alik Sokolov, Jonathan Mostovoy, Eduard Losing, Marco Ceccarelli, Qingyang Emma Zhang, Yichi Zhang, Kurtis Laurion, Baker Jackson, and Luis Seco. 2022. Assessing the Impact of Sustainability on Fund Flows: An Excess Information Approach and US-Based Case Study. *The Journal of Impact and ESG Investing* (2022).
- [30] Johannes Stübinger and Dominik Walter. 2022. Using multi-dimensional dynamic time warping to identify time-varying lead-lag relationships. *Sensors* 22, 18 (2022), 6884.
- [31] Konstantinos Tolikas. 2018. The lead-lag relation between the stock and the bond markets. *The European Journal of Finance* 24, 10 (2018), 849–866.
- [32] Milena Vuletić, Felix Prenzel, and Mihai Cucuringu. 2023. Fin-GAN: Forecasting and Classifying Financial Time Series via Generative Adversarial Networks. Available at SSRN 4328302 (2023).
- [33] Kieran Wood, Stephen Roberts, and Stefan Zohren. 2022. Slow momentum with fast reversion: A trading strategy using deep learning and changepoint detection. *The Journal of Financial Data Science* 4, 1 (2022), 111–129.
- [34] Di Wu, Yiping Ke, Jeffrey Xu Yu, Philip S Yu, and Lei Chen. 2010. Detecting leaders from correlated time series. In *International Conference on Database Systems for Advanced Applications*. Springer, 352–367.
- [35] Can-Zhong Yao and Hong-Yu Li. 2020. Time-varying lead-lag structure between investor sentiment and stock market. *The North American Journal of Economics and Finance* 52 (2020), 101148.
- [36] Yichi Zhang, Mihai Cucuringu, Alexander Y Shestopaloff, and Stefan Zohren. 2023. Robust Detection of Lead-Lag Relationships in Lagged Multi-Factor Models. *arXiv:2305.06704* (2023).