On Machine Learning Methods for Time Series with Financial Applications

Stefanos Bennett
St Peter's College, University of Oxford

DPhil thesis
Centre for Doctoral Training in Modern Statistics and Statistical Machine Learning (StatML)

Michaelmas 2023

Abstract

This doctoral project investigates machine learning methods for time series that are motivated by challenges found in financial market time series data. In this thesis, three research projects are described. The first project, which is entitled "Lead–lag detection and network clustering for multivariate time series with an application to the US equity market", proposes a method for the extraction of clusters of leading and lagging time series in multivariate time series systems using directed network clustering. The second project, which is entitled "Time Series Prediction under Distribution Shift using Differentiable Forgetting", proposes a bi-level optimisation framework for updating time series prediction models in response to distribution shift. The third project, "Rethinking Neural Relational Inference for Granger Causal Discovery", studies the limitations of Neural Relational Inference, which is a graph-based variational auto-encoder model, in recovering the Granger Causal structure of multivariate time series. While a unifying theme of the thesis is that the methods developed were motivated by the characteristics of financial time series, the methods themselves can also be applied to non-financial data.

Acknowledgements

I would like to thank my supervisors, Mihai Cucuringu and Gesine Reinert, for their valuable guidance and their generosity in sharing their time and experience with me. I am greatly appreciative of their commitment and encouragement throughout my degree.

I would like to thank my Confirmation and Transfer of Status examiners, Xiaowen Dong and Frank Windmeijer, for their thoughtful feedback and guidance in refining my work. I would also like to thank Paolo Barucca for acting as my external examiner for my viva voce.

I acknowledge the vital financial support of the EPSRC CDT in Modern Statistics and Statistical Machine Learning and The Alan Turing Institute's Finance and Economics Programme. I would also like to thank the StatML CDT Management Team for providing an excellent environment for collaboration and learning, and, in particular, for their efforts during a tough period of Covid-19 social restrictions. Many thanks to Rose Yu for hosting me for a very enjoyable research visit to UCSD, and for her continued research guidance.

I would like to share my gratitude to friends and colleagues at the Department of Statistics, St Peter's College and the StatML CDT for their camaraderie and support, which greatly enriched my experience at Oxford.

Finally, I would like to express my special gratitude to my parents and my late grandfather, for their kindness, support and encouragement in pursuing my studies.

Contents

Chapter 1

Introduction

This doctoral project researches machine learning methods for time series that are motivated by challenges found in financial market time series data. In this thesis, three research projects are described, which relate to the problem of financial returns forecasting. This problem consists in the prediction of future returns for a single or multiple financial instruments over a given time period. The statistical modelling of financial market returns has important applications in portfolio management, risk monitoring and the development of statistical trading strategies. As a result, the problem has spawned a substantial literature [Petropoulos et al., 2022]. The statistical features of financial time series [Cont, 2001] provide challenges and opportunities for the study and development of novel time series methods. We begin our introduction by highlighting certain financial time series characteristics that jointly motivate the doctoral projects in Section 1.1.1. In light of these financial time series characteristics, we describe the modelling approaches that are shared across different projects in Section 1.1.2. This is followed by an account of the preliminary backgrounds and introductions to each of the doctoral projects.

1.1 Shared problem structures and modelling approaches

1.1.1 Financial time series characteristics

The five characteristics of financial time series data that motivate the projects presented in this thesis are as follows.

Financial return time series datasets are often highly multivariate and have complex dependency structures. Over 8000 equities can be traded on the New York Stock Exchange[1]; the US equity market can be viewed as a highly multivariate time series system. Further, this multivariate time series system is characterised by a complex structure [Wan et al., 2021, Mantegna and Stanley, 2007]. In particular, a complex structure of dependencies can be seen in the synchronous correlation structure of stock returns. Many sources of connectivity between stocks have been found to influence their correlation: these include shared business sector membership [Mantegna, 1999], common exposure to supply chain shocks [Onnela et al., 2003], or geographical proximity [Eckel et al., 2011]. Cohen et al. [2008] also show that social connections between mutual fund managers and corporate board members are an important source of information flow into stock prices.

Strong similarities can be found between the returns of financial instruments. Financial instruments are thought to exhibit common statistical properties, even across different markets.

[1] https://www.nyse.com/network/article/nyse-tapes-b-and-c

These phenomena on the empirical properties of asset returns have been widely studied and are known as stylised facts [Bouchaud, 2002, Cont, 2001]. In addition to these stylised facts concerning the statistical properties of returns, similarities can be found between the returns of certain financial instruments because they are driven by common factors. The contemporaneous financial returns of assets within the same market are often correlated. For instance, Avellaneda and Lee [2010] show that the number of eigenvectors needed to explain the variance of the correlation matrix of US equity returns varies between $7 - 33$ over time. This shows that equity returns are mostly driven by a common set of limited factors.

Financial returns forecasting tasks typically display low signal-to-noise ratios [Cont, 2001]. On liquid financial markets, asset returns exhibit little to no auto-correlation when measured on a time scale larger than the order of tens of minutes [Bouchaud, 2002]. These low effect sizes are challenging to detect with a limited sample size. Indeed, for daily or lower-frequency returns data, the power of methods to detect weak signals in forecasting returns is often hampered by low sample sizes. Furthermore, financial markets exhibit regime changes [Procacci and Aste, 2019]. This further limits the size of the number of relevant samples since each regime will, in general, need to be modelled separately.

Financial returns exhibit a degree of asynchronicity. Lo and MacKinlay [1990b] show that market frictions can slow information propagation between assets; new information tends to be reflected in the price of larger, more frequently traded assets before it is reflected in smaller capitalisation, less frequently traded stocks. This difference in the timing of information absorption manifests in asynchronicity between observed asset returns. In other words, financial instruments exhibit lead-lag effects. These lead-lag effects can vary through time and with market conditions, which makes estimation challenging [Ito and Sakemoto, 2020].

Time series of financial returns are often non-stationary. Time-variation in the distribution of financial returns is a well-established phenomenon [Procacci and Aste, 2019, Andersen et al., 2003]. An example of non-stationarity in financial returns can be seen by examining the correlation between stocks across time. Correlations are known to increase during volatile periods [Bouchaud, 2002]. This time-variation in the correlation between stocks has important implications for risk management: as correlations between stocks increase, the diversification of a portfolio holding these stocks will decrease.

1.1.2 Shared modelling approaches

These financial time series characteristics provide a modelling challenge and motivate the development of machine learning models that can meet these challenges. In response to these challenges, we describe the modelling approaches that we will use in the thesis.

Financial return time series datasets are often highly multivariate and have complex dependency structures. This motivates the use of graph-based modelling of multivariate time series in Chapter 2, "Lead–Lag Detection and Network Clustering for Multivariate Time Series with an Application to the US Equity Market" and Chapter 4, "Rethinking Neural Relational Inference for Granger Causal Discovery". In our graph-based modelling, time series are represented as nodes, and edges between nodes represent relationships between the underlying time series. This representation emphasises the connectivity structure and aggregate graph properties of the system. The question of how to construct the graph is key in such an analysis. In Chapter 2, we define a correlation-based relation between pairs of time series to construct the graph; such approaches are commonly used in

finance [Onnela et al., 2003]. In Chapter 3, we study an approach that treats the graph of relations between time series as a latent variable, which is then inferred based on the observation of the multivariate time series.

Strong similarities can be found between the returns of financial instruments. Financial time series similarity may be used as an inductive bias for creating new methods for modelling multiple financial time series. The problem of predicting multiple time series entails the estimation of a model for each of these time series. Given the strong similarity between the returns of certain financial instruments, the use of model parameter sharing to leverage these similarities across different forecasting tasks becomes an attractive proposal. The idea behind parameter sharing for forecasting multiple time series is to pool the inference of the parameters across time series; this increases the effective sample size used to estimate each model. Parameter sharing is an attractive option for financial time series returns forecasting, as there is typically little to no significant auto-correlation for the returns of each asset. The increase of effective sample size through parameter sharing therefore permits better estimation of any correlation effect in returns that does exist. More broadly, parameter sharing, also known as global modelling, underpins approaches that have been successful in the M4 [Makridakis et al., 2020] and M5 multi-domain time series forecasting challenges [Makridakis et al., 2022].

Chapter 2 includes a method that aims to leverage the similarities between time series using parameter sharing. Parameter sharing is used in the forecasting application presented in Chapter 2. In this forecasting application, we use a novel method to derive clusters of financial time series such that time series within a cluster have similar leading and lagging behaviours with respect to time series in other clusters. This property enables the use of cluster-level predictive modelling: there is a single predictive model for all time series within a cluster. Therefore, the forecasting application of Chapter 2 illustrates how our novel clustering method may provide a data-driven approach to determine which time series should share model parameters.

Parameter sharing is also inherent in the permutation invariance assumption underpinning the use of graph neural networks in Chapter 4. In the model studied in Chapter 4, the dynamics of the multivariate time series system, conditioned on the graph of relations between time series, are shared across multivariate time series samples. The inferred graph of relations between time series dictates the way information is aggregated between time series and modulates the degree of parameter sharing.

Financial returns forecasting tasks typically display low signal-to-noise ratios [Cont, 2001]. For daily or lower-frequency returns data, the power of methods to detect weak signals in forecasting tasks is often hampered by low sample sizes. In these cases, data-efficient methods are required. In two research projects described in this document, we use the inductive bias – which is described above – that there exists similarity between financial time series in order to increase the power of our methods to detect weak statistical effects. The project described in Chapter 2 aims to extract a latent cluster structure in the collection of inter-temporal relations between financial time series; this is accomplished using a network clustering approach that captures the similarities between the interactions of pairs of time series in order to extract a global clustering for the entire multivariate system. The project shows how the uncovered clustering can be used for challenging downstream forecasting tasks in the low signal-to-noise domain. The project described in Chapter 3 proposes a method to balance data relevancy with effective sample size to model time series data under distribution shift.

Financial returns exhibit a degree of asynchronicity. In Chapter 2, we review existing financial economic hypotheses for lead-lag relationships in asset returns. We identify the possibility of clustered

lead-lag relationships in the US equity market. Using a novel, data-driven method we investigate the structure of this clustering. The notion of a lead-lag relationship is captured using an associative modelling approach in Chapter 2. Namely, we use the cross-correlation between two time series to determine the presence of a lead-lag relation. As we discuss in Chapter 2, our definition of lead-lag relation precludes reasoning about the causality structure of time series. The question of whether there truly exist predictive causal relationships between pairs of time series is, in general, more challenging to answer. Granger causality is an approach to formally address this question [Shojaie and Fox, 2022]. Granger causality is a modelling approach that has been used in finance to model risk spillovers [Hong et al., 2009], the relative influence of stock market indices [Yong Tang and Zhang, 2019], as well as other phenomena [Shojaie and Fox, 2022]. In Chapter 4, we investigate a novel method for inferring Granger causal relations.

Time series of financial returns are often non-stationary. We see the impact of non-stationarity in the forecasting application of Chapter 2: the predictive performance of our model varies through time. This exemplifies the challenge of predicting non-stationary time series, or in other words, prediction under distribution shift. In fields such as economics and finance [Rossi, 2013], distribution shift often occurs at unknown times and is of unknown type; as a result, these fields provide a natural test-bed for time series distribution shift methods [McCarthy and Jensen, 2016, Kuznetsov and Mohri, 2020]. This motivates the need for a model-agnostic method that achieves competitive performance across a range of different distribution shift settings. In Chapter 3, "Time Series Prediction under Distribution Shift using Differentiable Forgetting", we develop a model-agnostic method for forecasting non-stationary time series. Amongst other economic and financial variable modelling tasks, we show how our method can be applied to improve financial risk modelling. Specifically, by learning how to vary the loadings on different equity driving factors through time, we can better explain the dynamic risk profile of a range of US equities.

Next, here are more detailed backgrounds and introductions to each of the three projects in this thesis.

1.2 Lead–Lag Detection and Network Clustering for Multivariate Time Series with an Application to the US Equity Market (Chapter 2)

In a multivariate time series system, we say that time series $\mathcal{A}$ leads time series $\mathcal{B}$ if $\mathcal{A}$'s past values are more strongly associated with $\mathcal{B}$'s future values than $\mathcal{A}$'s future values are with $\mathcal{B}$'s past values. Multivariate systems arising in domains such as earth science, biology and economics, often exhibit lead-lag relationships. For instance, Harzallah and Sadourny [1997] study the lead-lag relationship between the Indian summer monsoon and a grid of climate variables such as snow cover, sea surface temperature and geopotential height. In financial markets, lead-lag effects arise due to asynchronicity in prices [Lo and MacKinlay, 1990a]. There exists substantial evidence of lead-lag relations between financial instruments occurring at multiple scales [Badrinath et al., 1995, Brennan et al., 1993, Curme et al., 2015a]. In large-scale systems such as the US stock market, a lead-lag cluster structure may emerge. Previous works study the relative influence of *pre-defined* groups of equities, based on traits such as industry membership [Biely and Thurner, 2008, Liao et al., 2014] or geographical location [Sandoval, 2014]. However, no *data-driven* method has been proposed to directly extract latent lead-lag cluster structures based on multivariate time series observation. The number of works applying other forms of data-driven analysis, such as ranking, to financial lead-lag networks underscores the potential utility of a lead-lag clustering method [Wu et al., 2010, Basnarkov et al., 2019]. In Chapter 2, we propose a method for this task.

Problem description

The objective of our work is to cluster a multivariate time series system into communities that are, taken in pairs, mostly composed of either leading or lagging time series. This objective comprises two parts: lead-lag network construction and clustering of this network.

The first part can be addressed by defining a lead-lag relationship between two time series; the collection of all pairwise lead-lag relations can then be viewed as a network, in which the nodes correspond to time series and the edges represent the lead-lag relations. A question lies in the definition of the pairwise lead-lag relation: how does one mathematically define a metric to quantify a lead-lag relationship?

The second part of the objective is to cluster the network of lead-lag relations. We note that a lead-lag relation is associated with direction: either $\mathcal{A}$ leads $\mathcal{B}$ ($\mathcal{A} \to \mathcal{B}$) or $\mathcal{B}$ leads $\mathcal{A}$ ($\mathcal{B} \to \mathcal{A}$). In general, a lead-lag relation is also associated with a weight which indicates the strength of the relation. Hence, the lead-lag network clustering task can be formulated as a directed weighted network clustering task. Specifically, we wish to segment our multivariate system into communities which, taken in pairs, are mostly composed of either leaders or laggers. Mathematically, the usefulness of a clustering can be described through a notion of *cut imbalance*. In a directed graph with adjacency matrix A, the cut associated to two subsets of nodes $\mathcal{A}$ and $\mathcal{B}$, is given by

$$Cut(\mathcal{A}, \mathcal{B}) = \sum_{i \in \mathcal{A}, j \in \mathcal{B}} A_{ij}. \tag{1.1}$$

We refer to the difference $Cut(\mathcal{A}, \mathcal{B}) - Cut(\mathcal{B}, \mathcal{A})$ as the cut imbalance. A high cut imbalance between communities $\mathcal{A}$ and $\mathcal{B}$ indicates that variables in $\mathcal{A}$ are, on average, leaders of variables in $\mathcal{B}$. The key question in lead-lag network clustering therefore becomes: how can we identify clusters in the lead-lag network that have high pairwise cut imbalance?

Application to US equity market

Having introduced and tested a method to solve the problem of clustering a lead-lag network in Chapter 2, we then apply our method to obtain a clustering of the US equity market. The key questions that we wish to address in this application to US equities are as follows:

1. Does there exist a statistically significant cluster structure in the US equity market?
2. What is the nature of the clustering?
3. How does the cluster structure relate to previously discovered lead-lag mechanisms?
4. Can we leverage our clustering for downstream forecasting purposes?

We propose a permutation test to evaluate the statistical significance of our cluster structure. In order to investigate the nature of the clustering, we study cluster membership and inter-cluster cut imbalances in light of hypothesised lead-lag mechanism in the empirical finance literature [Biely and Thurner, 2008, Chordia and Swaminathan, 2000, Lo and MacKinlay, 1990a]. Further, we study the stability of the clusters through time.

In Chapter 2, we demonstrate how lead-lag clustering methods may be used within a multivariate time series forecasting pipeline for signal extraction and data-guided parameter pooling. A difficulty in forecasting high-dimensional systems is the identification of a suitable group of variables that can be used as predictors for other variables. The application of our method as a preliminary clustering step addresses this difficulty. Time series found within a cluster have similar leading and lagging behaviours with respect to time series found in other clusters. Therefore, by aggregating the

predictions into a cluster-level analysis, we reduce the number of parameters required to forecast each time series and reduce noise by averaging across time series. Similarly, Curme et al. [2015b] argue for the use of lead-lag networks to guide variable selection for downstream financial forecasting tasks.

1.3 Time series Prediction under Distribution Shift using Differentiable Forgetting (Chapter 3)

Time series exhibiting distribution shift appear in a wide range of domains such as industrial management and biomedical applications [Zliobaite et al., 2016]. As a result, prediction under distribution shift is a widely-occurring problem. In economics and finance, distribution shift often occurs at unknown times and is of unknown type [Rossi, 2013]. This motivates the need for a model-agnostic method that can deal with a range of different distribution shift settings.

We note that the problem of time series prediction under distribution shift falls under the wider category of transfer learning [Lu et al., 2021], a general learning setting in which the source distribution which we use to train a model is different from the target data distribution on which the model will be applied. We briefly summarise related sub-fields of transfer learning prior to discussing certain distinguishing features of the time series non-stationary prediction problem.

Fields related to non-stationary time series prediction

Areas related to the problem of prediction under distribution shift include:

- Domain adaptation: here, the training data distribution differs from the test data distribution, but both share the same feature space.
- Covariate shift adaptation: this is a form of domain adaptation where the distribution shift is limited to a change in the marginal distribution over the input variables.
- Meta-learning aims to learn a procedure to effectively train a model on a new task using a training set composed of different but related tasks [Finn et al., 2017]. A task in this instance consists of a prediction problem defined over a particular dataset.
- Continual or lifelong learning aims to learn from a stream of input data (and is therefore a branch of online learning). The key emphasis is on quickly adapting to new tasks while retaining the ability to perform previously seen tasks (i.e. avoiding the phenomenon of "Catastrophic Forgetting").

Whereas certain sub-fields of transfer learning such as domain adaptation and meta-learning deal with distribution shift *across* datasets, we are interested in the problem of distribution shift *within* a single dataset. Further, while the temporal nature of the distribution shift lies at the centre of non-stationary time series prediction, in standard domain adaptation and meta-learning settings, there is no explicit notion of time indexing. Methods in these fields must be adapted to handle time series data. Whereas online and continual learning address the problem of efficient adaptation under streaming data, our dataset size is fixed. Further, while one of the key goals of continual learning is maintaining performance on already observed tasks, we are only interested in forecasting performance at the current point in time. Therefore, previously observed tasks or regimes are only relevant insofar as they assist in training a model that will achieve strong predictive performance at the current point in time.

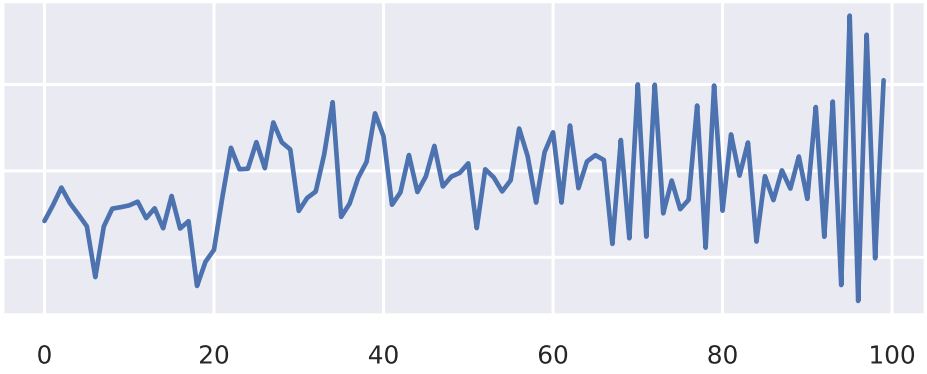


Figure 1.1: Example of a non-stationary times series using samples $t = 0, \ldots, 99$. The y-axis indicates the value of the time series and the x-axis indicates the time index t.

Motivating Example

As a motivating example of the problem of time series prediction under distribution shift, suppose that we are tasked with building a model to predict the next value of the univariate time series displayed in Figure 1.1.

The behaviour of the time series changes over time: at the start of the range, it exhibits trending behaviour (positive auto-correlation) while towards the end of the range, it exhibits mean-reverting (negative auto-correlation) behaviour. This indicates that the data generating process of the time series is time-varying i.e. that it undergoes distribution shift. Given this observation, how should we fit a model to predict the value of the time series at $t = 100$? A reasonable assumption is that the data generating process at $t = 100$ will be more strongly related to the data generating process at closer values of t compared to more distant values of t. Operating under this assumption, how do we decide what importance to assign to each time series sample during model estimation? When deciding what importance to assign to each sample, there is a trade-off between the relevancy and effective sample size that is used for the estimation of the predictive model. More generally, how do we predict under distribution shift settings where there has been an abrupt shift in the history of the time series or where there is a recurring regime?

Mathematical Problem Description

To frame the problem concretely, we study the supervised learning problem where we wish to learn a mapping f between inputs X_t and labels Y_t indexed by time t:

$$f(\cdot\,; \theta_t) : X_t \mapsto Y_t, \tag{1.2}$$

where the conditional distribution $Y_t|X \sim \pi_t(\cdot|X)$ is time-varying and the marginal distribution $X_t \sim \pi_t(\cdot)$ may also be time-varying. At a given time point $T-1$, we are interested in minimising the one-step-ahead path-dependent risk

$$R_T(\theta) = \mathbb{E}_{Y_T \sim \pi_T(\cdot|X_T)} \left[L\left(f(X_T; \theta), Y_T\right) | \{(X_\tau, Y_\tau)\}_{\tau=1}^{T-1} \right]. \tag{1.3}$$

We follow a weighted Empirical Risk Minimisation (ERM) approach in which the unknown true path-dependent risk (equation (1.3)) is estimated through the weighted Empirical Risk:

$$\hat{R}_T(\theta) = \sum_{\tau=1}^{T-1} \alpha_{|T-1-\tau|} L\left(f(X_\tau; \theta), Y_\tau\right). \tag{1.4}$$

The distribution shift in the data is accommodated through the values of the weights $\alpha_0, \ldots, \alpha_{T-2}$ in the risk estimator. The predictive model is then estimated by minimising the weighted Empirical Risk. The key question in this approach lies in the inference of the sample weights $\alpha_0, \ldots, \alpha_{T-2}$.

Data re-weighting literature

Data re-weighting is a well-established approach for dealing with distribution shift that has its roots in the principles of importance sampling [Kanamori et al., 2009]. Using the importance sampling principle [Shimodaira, 2000], we can obtain an unbiased estimate of R_T of the form

$$\hat{R}_T(\theta) = \frac{1}{T-1} \sum_{\tau=1}^{T-1} \frac{\pi_t(Y_\tau|X_\tau)}{\pi_\tau(Y_\tau|X_\tau)} L(f(X_\tau;\theta), Y_\tau). \tag{1.5}$$

In general, each ratio $\frac{\pi_t(Y_\tau|X_\tau)}{\pi_\tau(Y_\tau|X_\tau)}$ is unknown and this therefore provides a modelling challenge for distribution shift methods.

Data re-weighting has been widely studied for the problem of covariate shift adaptation [Kanamori et al., 2009]. Covariate shift adaptation is a specific instance of a distribution shift problem in which the marginal distribution of the inputs may vary from the training to the test set while the conditional distribution of the labels given the inputs remains fixed. A popular method for dealing with covariate shift adaptation is to use density ratio estimation approaches for estimating the importance sampling weights in equation (1.5) [Lu et al., 2021, Zhang et al., 2021]. The development of density ratio estimation and distribution matching approaches in the field of covariate shift adaptation has led to their application to the more general distribution shift setting in which *both* the conditional distribution of the labels and the marginal distribution of the inputs may vary from training to test set [Fang et al., 2020].

The application of data re-weighting approaches to the field of non-stationary time series prediction is more recent [Kuznetsov and Mohri, 2020, Masserano et al., 2022, Yusupova et al., 2022]. Kuznetsov and Mohri [2020] have the same focus as our work: they aim to minimise the one-step-ahead risk of their predictive model trained using a weighted empirical risk. They derive a learning guarantee on the one-step-ahead generalisation error that holds in the non-stationary time series setting. They then minimise this learning guarantee over the weights in their empirical risk and the parameters of their model. Masserano et al. [2022] treat the sample weights as hyper-parameters which are dealt with using Bayesian optimisation. Yusupova et al. [2022] consider a specific form of a linear predictive model and a single discount factor that determines the sample weights.

Proposed method

We propose a novel model-agnostic approach to data re-weighting for time series prediction under distribution shift. Specifically, we propose to infer the ERM weights using a generic parametric model which we call a "forgetting mechanism". The forgetting mechanism maps a time index to its sample weight. We propose a method for jointly learning the forgetting mechanism and predictive model parameters using bi-level optimisation. In contrast to previous work, we propose a gradient-based learning method for the parameters of the forgetting mechanism. This provides quick estimation of forgetting mechanism parameters and therefore facilitates more expressive forgetting mechanisms.

In Chapter 3, we situate our method theoretically using the bi-level optimisation literature and investigate the performance of our method against other state-of-the-art distribution shift methods based on data re-weighting. We evaluate our method on four synthetic datasets and seven real-world time series prediction tasks. The real-world predictive tasks span several domains including finance, economics, epidemics and energy modelling. A particular financial application of interest is that

of risk modelling using the Fama-French [Fama and French, 1993] decomposition. In this case, distribution shift manifests as time-variation in the risk factor loadings [McCarthy and Jensen, 2016].

Alternative methods in the non-stationary time series modelling literature

We briefly mention two alternative approaches for model adaptation under distribution shift that do not use data re-weighting. Ensemble methods [Shalizi et al., 2011] combine predictions from diverse models trained on different data subsets, thereby hedging against the impact of distribution shifts. Adaptive state-space methods treat parameters as latent variables and posit a dynamic update equation for these variables [Hamilton, 2010]. The parameter update equations impose an assumption on the form of distribution shift. In this case, the method used for parameter inference depends on the complexity of the dynamic update equation and prediction models.

1.4 Rethinking Neural Relational Inference for Granger Causal Discovery (Chapter 4)

Chapter 4 studies the performance of Neural Relational Inference [Kipf et al., 2018], a recently proposed method, on the task of recovering the Granger Causal structure of multivariate time series. We begin by introducing the task of Granger causal discovery.

An introduction to Granger Causality

Granger causal discovery is a widely studied problem with real-world applications in several fields such as neuroscience [Sporns, 2016], genetics [Fujita et al., 2010] and finance [Campbell et al., 1998]. Given an observed multivariate time series dataset $X_t^i \in \mathbb{R}$, $i = 1, \ldots, N, t = 1, \ldots, T$, Granger causal discovery aims to infer the underlying Granger causal relationships between all pairs of time series. Mathematically, time series X^i Granger causes [Eichler, 2012] time series X^j if, for some t, and some non-empty set S,

$$\mathbb{P}[X_{t+1}^j \in S|\mathcal{I}(t)] \neq \mathbb{P}[X_{t+1}^j \in S|\mathcal{I}_{-X^i}(t)]. \tag{1.6}$$

The sets $\mathcal{I}(t)$ and $\mathcal{I}_{-X^i}(t)$ respectively denote all information available as of time t and all information available as of time t excluding time series X^i. Intuitively, a Granger causal relation exists if X^i contains unique information that helps predict future values of X^j. Example applications of Granger causal discovery include estimating the interdependence of different brain regions based on regional activity measurements over time [Seth et al., 2015], estimating gene regulatory networks [Michailidis and d'Alché Buc, 2013] and understanding cross-country variations in electricity prices [Castagneto-Gissey et al., 2014].

Lechner [2010] discusses the relation of Granger causality to other notions of causality, such as the potential outcomes framework [Rubin, 2005]. In general, the Granger causality framework is distinct from the potential outcomes framework. Granger causality is inferred using purely observational data and does not rely on data from interventional studies. Causal discovery methods in the potential outcomes framework, such as NOTEARS [Zheng et al., 2018] and DAG-GNN [Yu et al., 2019], are not generally designed for temporal data. Additional conditions such as a lack of instantaneous effects and absence of unmeasured confounding variables [Maziarz, 2015] are required to draw an equivalence between the two frameworks. While an inferred Granger causal relation is not necessarily indicative of a true causal relation, it remains a popular framework for understanding temporal relations in multivariate dynamical systems. Indeed, in a literature review of causality in economics, Hoover [2006] states that Granger causality is "the most influential explicit approach to causality in economics".

Granger Causality methods

In the case of linear VAR modelling, the Granger causal relationship between two random variables, indexed by i and j, can be tested through the hypothesis that the coefficients relating the lagged values of time series i to the current value of time series j are jointly significantly different to 0. Granger causal discovery can be accomplished when N is small through an exhaustive hypothesis test search on the coefficients in the VAR model. Other variable selection methods such as sparsity-inducing regularisation or the graphical lasso may be used instead of hypothesis testing to estimate Granger Causal relations. Arnold et al. [2007] provide an overview of classical approaches for Granger causal discovery with linear modelling.

More recent work has extended the Granger causal discovery framework to account for non-Gaussian observations [Lanne et al., 2017], non-linear effects [Tank et al., 2021] or sub-sampled time series [Gong et al., 2015]. Shojaie and Fox [2022] provide an overview of recent advances within the field of Granger causal discovery. Neural network-based modelling approaches to Granger Causality [Khanna and Tan, 2019, Tank et al., 2021, Yin and Barucca, 2022] enable the modelling of non-linear dynamics and can accommodate more complex data sources. Neural network approaches such as that of Tank et al. [2021] and Khanna and Tan [2019] infer Granger causal relations through sparse estimation of network weights. Other neural network methods approaches [Orjuela-Cañón et al., 2020, Wang et al., 2018] test if time series j Granger causes time series i by evaluating the difference in prediction accuracy between a neural network model trained using the full dataset and a neural network trained excluding the time series j. More recent works apply a neural network approach to account for confounding variables in Granger causal estimation [Löwe et al., 2022, Yin and Barucca, 2022]. Another direction of research explores inductive neural network approaches to Granger causal modelling [Löwe et al., 2022, Chu et al., 2020]. Inductive approaches are useful in settings characterised by a large number of multivariate samples generated from related processes. Inductive approaches train a single model to infer Granger causal relations across all samples; they are therefore able to perform inference on unseen samples. An example application area for inductive approaches is the analysis of fMRI data from different subjects [Smith et al., 2011]. While different test subjects may have different patterns of brain connectivity, the same neurobiology underlies the brain activity of all subjects. Therefore, a reasonable strategy to pool inference is to train a single inductive model on the data observed across all subjects.

Background on Neural Relational Inference (NRI)

Neural Relational Inference (NRI), originally proposed in Kipf et al. [2018], is an encoder-decoder model for the auto-regressive modelling of multivariate time series with latent graph structures. NRI is an inductive approach which emphasises relational modelling through its graph-based encoder and decoder architectures. NRI has achieved state-of-the-art performance on benchmark relational modelling datasets in physics and biology [Kipf et al., 2018]. NRI has also been successfully applied in practice. For instance, Liu et al. [2023], Zhu et al. [2022] use NRI to model molecular dynamics and Li et al. [2020] use a model which is based on NRI to estimate causal graphs based on video data.

The NRI encoder is trained to map multivariate time series inputs to distributions over latent graph structures. The latent graph structure is encoded as the set of categorical relations between each pair of time series. Conditional on the inferred latent graph structures, the NRI decoder aims to predict the next-step values of each multivariate time series sample. Accordingly, conditional on the latent graph structure for each sample, the NRI model assumes shared dynamics across multivariate time series samples. The model can be viewed as a variational auto-encoder and is trained using amortised Variational Expectation-Maximisation [Kipf et al., 2018]. The training loss is given by the sum of a reconstruction loss, which judges the quality of the next-step multivariate time series

predictions, and a KL loss which penalises encoder graph outputs that deviate from the user-specified graph prior distribution.

The encoder and decoder NRI networks take the form of Graph Neural Networks (GNNs) [Zhou et al., 2020]. GNNs consist of a class of models which operate on graph-structured data by propagating information from node to node along the edges of a graph. GNNs have found success in relational reasoning [Santoro et al., 2017] and multi-agent modelling [Battaglia et al., 2016] tasks. The encoder GNN operates on a fully-connected graph in order to infer an embedding on all pairs of time series. The decoder GNN performs message passing along the edges of the latent graph inferred by the encoder.

Understanding the performance of NRI on Granger Causal Discovery

Löwe et al. [2022] recently proposed applying NRI to the task of Granger Causal Discovery. They empirically validate the performance of NRI on a simulated fMRI dataset as well as two physical systems. However, the conditions under which NRI is guaranteed to recover the true Granger causal graph underlying a multivariate time series are still unknown. Further, the performance of NRI has not yet been studied on economic and financial datasets; these datasets are characterised by lower signal-to-noise ratios than the highly structured datasets in physics and biology to which NRI has been previously applied. This creates a hurdle to applying NRI in practice as there are no theoretical guarantees or prior experimental results to suggest that it can recover the true causal structure in economic or financial applications.

NRI is a potentially attractive model to apply to economics and finance for three reasons. First, its empirical success in modelling physical systems makes it a promising candidate model to apply to new fields. Second, there is a prevalence of Granger causal inference problems in economics and finance. Third, graph-based modelling is known to be useful for analysing certain economic and financial systems. For example, Knight et al. [2020] find that the network structure aids prediction for gross domestic product growth rate time series across 35 OECD countries relative to linear vector auto-regressive (VAR) forecasting.

In Chapter 4, we investigate the question of when NRI recovers the true Granger causal graph. Since it is important to understand the performance of NRI on a simple benchmark prior to applying it to real-world data, we examine NRI in the context of the Generalised Network Autoregressive Process (GNAR) of Knight et al. [2020]. GNAR is a VAR data generating process with graph structure. The study of the performance of NRI under the GNAR data generating process, which is more strongly characteristic of economic and financial data compared to existing NRI benchmark datasets [Kipf et al., 2018], provides new perspectives on the performance of the model.

Works related to NRI

Generalised Network Autoregressive Process (GNAR)

The NRI model can be viewed as an extension of the GNAR model [Knight et al., 2020]. The GNAR model linearly propagates information between time series which are represented as nodes of an underlying graph. As such, it can be viewed as a graph-constrained form of a VAR. The GNAR model either treats this underlying graph as observed or treats it as a nuisance parameter (selected through random search). Extensions of GNAR have been proposed, such as GNAR-edge [Mantziou et al., 2023] which models time-variation in the edge weights. Neither, GNAR nor GNAR-edge can take into account uncertainty about the structure of the graph. In contrast, the NRI model infers graph structure using its encoder. The encoder incorporates information from its prior over graphs as well as the multivariate time series observations. Therefore, NRI can potentially incorporate hypothesised or real-world networks that are associated with each modelling task through the NRI model's prior distribution. This would permit the Granger causal discovery mechanism of the NRI encoder to be

guided by, but not constrained to, real-world networks that are thought to relate to the true Granger causal graph. This makes NRI an interesting proposal for the field of economics and finance where it is common to have a prior network which may be associated with, but not exactly correspond to, the ground-truth Granger causal underlying a time series system. For instance, a representation of the world trade network could be used as a prior for modelling GDP growth rates across time [Knight et al., 2020]. The approach of incorporating a real-world network into an NRI model through the prior has been successful in other fields such as traffic prediction [Tygesen et al., 2023]. Advances in the field of structural prior learning may prove useful in graph prior elicitation [Pu et al., 2021]. In addition to NRI's latent graph inference capacity, it is a more expressive model than GNAR as it is able to accommodate non-linear effects through its GNN decoder.

Latent-graph inference methods

NRI tackles the Granger causal discovery problem by inferring a latent graph that represents the collection of pairwise Granger causal relations in a multivariate time series. Here we briefly discuss other works that use latent graph modelling for multivariate time series forecasting. In most of the applications of graph deep learning for time series, graphs represent a spatial dimension [Cini et al., 2023a]. The latent graph models mentioned below were not designed to be used for Granger causal discovery: they typically treat the adjacency matrix as a spatial nuisance parameter. Furthermore, these latent graph methods typically seek to recover a single latent graph and are therefore non-inductive.

Several non-probabilistic approaches treat the edges of the latent graph as continuous variables representing similarities between pairs of nodes; in deep learning approaches, node-to-node similarities are modelled using the dot product between node embeddings that are learned during training [Oreshkin et al., 2021, Wu et al., 2019, BAI et al., 2020]. Thresholding is a commonly used non-probabilistic approach to obtain discrete adjacency matrix representations: in this case, the smallest node-to-node similarities are set to zero [Wu et al., 2020, Deng and Hooi, 2021].

Probabilistic approaches to learning the adjacency matrix typically deal with the non-inductive setting: they aim to learn a single graph which underpins all observations [Cini et al., 2023a]. Probabilistic approaches model the entries of the adjacency matrix using distributions, such as the Bernoulli distribution, that may be parameterised by node-to-node edge similarities, as described above. The key issue to overcome in learning the discrete adjacency matrix is to combine gradient estimation with discrete sampling. Gradient estimators which use the *re-parameterisation trick* [Maddison et al., 2016] introduce a continuous relaxation of the Bernoulli distribution [Shang et al., 2021]. Gradient estimators which are based on the score function approach [Williams, 1992] re-express the gradient of an expectation as the expectation of a related quantity, the latter of which can be estimated using Monte Carlo sampling [Cini et al., 2023b]. Franceschi et al. [2019] propose a bi-level optimisation method for jointly learning latent graph structures and graph neural networks operating on these structures.

1.5 Thesis overview

This thesis includes the three projects introduced above. Chapter 2 consists of a paper that was published in the Springer Machine Learning Journal entitled "Lead–Lag Detection and Network Clustering for Multivariate Time Series with an Application to the US Equity Market"; an abridged version of this paper was presented at the 7th SIGKKDD Workshop on Mining and Learning from Time Series (MiLeTS). Chapter 3 consists of a paper entitled "Time Series Prediction under Distribution Shift using Differentiable Forgetting" of which a preliminary version was presented at the 2022 ICML Workshop on Principles of Distribution Shift. Chapter 4 consists of a paper entitled "Rethinking

Neural Relational Inference for Granger Causal Discovery" of which a preliminary version was presented at the 2022 NeurIPS Workshop on Causality for Real-world Impact. Chapter 5 discusses avenues of future work and concludes the thesis.

Towards a comprehensive modelling of complex dependencies in non-stationary financial systems As outlined in Section 1.1.1, financial data are often characterised by complex dependency structures. It is important to understand the dependency structures between financial time series for the purposes of forecasting, risk management and portfolio optimisation. Chapters 2, 3 and 4 provide significant contributions to the problem of modelling complex dependencies in non-stationary financial systems. Chapter 2 develops a method that can be generally applied to understand the cluster structure of temporal dependencies in multivariate time series systems. Chapter 3 develops a model-agnostic method for parameter adaptation in the presence of distribution shift. Given the range of models that can be applied to capture financial time series interactions, as well as the challenge of non-stationarity in financial data, this model-agnostic distribution shift method is a valuable step in modelling non-stationary interactions. Chapter 4 investigates a model for temporal interactions in time series systems which aims to discover Granger causal interactions. This is a general-purpose model that aims to provide discovery of complex dependencies in time series systems without relying on domain expertise. The work of this thesis contributes towards a comprehensive modelling of complex dependencies in non-stationary financial systems.

A comparison of approaches based on correlation and causality In our work, we contrast two approaches to modelling complex dependency structures. The first approach, developed in Chapter 2, uses correlation-based modelling for capturing bivariate relations. The second approach of Chapter 4 uses Granger causality to model temporal dependencies. Correlation-based and causality-based methods provide two distinct approaches to modelling complex dependencies in financial time series systems. The method developed in Chapter 3 can be used in conjunction with both approaches.

Approaches based on correlation modelling provide a straightforward measure of the strength and direction of relationships between variables without making assumptions or inferences regarding causality. Correlation-based approaches are simpler, facilitate exploratory analysis, and make it easy to compare relationships across different pairs of variables. The downside of correlation-based approaches are that they are vulnerable to misinterpretations, as well as confounding or omitted variables effects.

Causality-based approaches can provide deeper insights into the underlying factors or variables driving the system. Causality approaches may provide more direct means of testing theories. Causality methods based on the potential outcomes framework enable counterfactual reasoning. Causal understanding can inform policymakers and investors about the potential impact of interventions or policy changes on financial markets. On the other hand, performing causal discovery and inference is more challenging than correlation-based modelling. Causal modelling often requires making strong assumptions about the relationships between variables and the absence of unobserved confounders. Building causal models may require expert knowledge to specify an appropriate model structure. Further, identifying true causal relationships from observational data alone may not always feasible due to the effects such as omitted variable bias. Interventional data are often unavailable in financial settings.

Chapter 2

Lead–Lag Detection and Network Clustering for Multivariate Time Series with an Application to the US Equity Market

The work of Chapter 2 develops a method for discovering clusters of time series with similar temporal interaction effects. This is a novel method for modelling dynamic interactions in multivariate time series which uses bivariate lead-lag interactions and graph clustering methods. We show that the method can be used for modelling the cluster structure of the US equity market which constitutes an important contribution towards the goal of comprehensively modelling complex dependencies in financial systems.

A note on the distinction between lead-lag and Granger causal graphs In Chapter 2, the dynamic interaction effects between time series are captured using a lead-lag metric. The collection of pairwise lead-lag interactions corresponds to a lead-lag network. This is not the only way to model dynamic interactions in time series systems. Indeed, we examine an alternative method in Chapter 4 that is based on Granger causal inference. We briefly explain the difference between lead-lag and Granger causal graph modelling.

The lead-lag graphs of Chapter 2 are simple, directed and weighted. There are no bidirectional edges between two nodes: the direction of the edge indicates which of two time series has a *net* leading effect on the other time series. The weight of each edge gives the net strength of the leading effect.

As we discuss in Chapter 2, the lead-lag metric defined between two time series *cannot* be interpreted as a causal effect. Unlike the lead-lag metric of Chapter 2, the Granger causal inference approach of Chapter 4 conditions on the entire history of the multivariate system. In particular, Granger causality takes into account the effect of auto-correlation and the effect of third variables (the lead-lag metric only captures bivariate effects). In contrast to the lead-lag graph, the Granger causal graph can be bidirectional: it is possible that time series A Granger causes time series B and that time series B Granger causes time series A. In contrast to the lead-lag graph, the Granger causal graph is unweighted.

The problem of Granger causal discovery is challenging, particularly in noisy, high-dimensional multivariate time series systems. Correlation-based methods may suffice and, indeed, may outperform Granger causality methods, on forecasting tasks when the objective is to purely minimise forecasting error. We demonstrate these considerations through our comparison of the performance of correlation-based and Granger causality-based forecasting of US equity data in Chapter 2. In this chapter, we

compare a forecasting model based on the outputs of lead-lag clustering method with a forecasting model based on a vector auto-regression (VAR) which is estimated using lasso regularisation. The VAR method can be seen as estimating a Granger causal graph: the non-zero coefficients of the inferred VAR matrix correspond to the estimated Granger causal connections in the multivariate time series.

Machine Learning
https://doi.org/10.1007/s10994-022-06250-4

Lead–lag detection and network clustering for multivariate time series with an application to the US equity market

Stefanos Bennett[1,2] · **Mihai Cucuringu**[1,2,3] · **Gesine Reinert**[1,2]

Received: 12 December 2021 / Revised: 30 June 2022 / Accepted: 15 September 2022

Abstract
In multivariate time series systems, it has been observed that certain groups of variables partially lead the evolution of the system, while other variables follow this evolution with a time delay; the result is a lead–lag structure amongst the time series variables. In this paper, we propose a method for the detection of lead–lag clusters of time series in multivariate systems. We demonstrate that the web of pairwise lead–lag relationships between time series can be helpfully construed as a directed network, for which there exist suitable algorithms for the detection of pairs of lead–lag clusters with high pairwise imbalance. Within our framework, we consider a number of choices for the pairwise lead–lag metric and directed network clustering model components. Our framework is validated on both a synthetic generative model for multivariate lead–lag time series systems and daily real-world US equity prices data. We showcase that our method is able to detect statistically significant lead–lag clusters in the US equity market. We study the nature of these clusters in the context of the empirical finance literature on lead–lag relations, and demonstrate how these can be used for the construction of predictive financial signals.

Keywords High-dimensional time series · Unsupervised learning · Lead–lag · Clustering · Financial markets · Directed networks · Flow imbalance

Editor: Joao Gama.

✉ Stefanos Bennett
stefanos.bennett@stats.ox.ac.uk

Mihai Cucuringu
mihai.cucuringu@stats.ox.ac.uk

Gesine Reinert
reinert@stats.ox.ac.uk

[1] Department of Statistics, University of Oxford, Oxford, UK

[2] The Alan Turing Institute, London, UK

[3] Mathematical Institute, University of Oxford, Oxford, UK

1 Introduction

Multivariate time series are ubiquitous in a wide range of domains, such as the physical sciences, medicine, and economics. Often, multivariate systems describing multiple processes or quantities are thought to exhibit lead–lag relationships (Podobnik et al., 2010). In this work, time series A is said to lead time series B if A's past values are more strongly associated with B's future values than A's future values are with B's past values. The study of lead–lag relationships in multivariate time series systems is of interest in fields such as earth science (Harzallah & Sadourny, 1997), biology (Runge et al., 2019) and economics (Wang et al., 2017a; Sornette & Zhou, 2005). For example, Harzallah and Sadourny (1997) study the lead–lag relationship between the Indian summer monsoon and a number of climate variables such as snow cover, sea surface temperature and geopotential height across a grid of locations on the Earth's surface. Wang et al. (2017a) examine the lead–lag dependence between the spot and futures markets for a Chinese stock market index.

In this paper, we examine systems of lead–lag relationships in time series data through the lens of directed network analysis. By constructing a network based on pairwise lead–lag metrics between variables, we are able to study overall properties of the web of lead–lag relationships via the tools of network analysis. Our specific interest lies in discovering clusters of different variables that exhibit strong lead–lag behaviour. To this end, we employ unsupervised directed network clustering and leverage recently developed algorithms (Cucuringu et al., 2020) that identify clusters with high imbalance in the flow of weighted edges between pairs of clusters.

While we expect our unsupervised learning method to be applicable to a number of multivariate time series domains, the particular application domain of interest in this study is the analysis of lead–lag clusters in financial time series data. Large financial markets, such as the US equity market, exhibit complex non-linear behaviour, often with a low signal-to-noise ratio (Cont, 2001). By using pairwise lead–lag detection and network analysis tools, we aim to extract clusters that capture the latent lead–lag relationships which may be present in such complex systems. Furthermore, persistent historical clusterings can be utilised for the challenging task of returns forecasting. As a result, our unsupervised learning method may prove to be a valuable component in certain financial forecasting pipelines. Beyond financial markets, this approach may lead to insights into the nature of lead–lag relationships in climate (Harzallah & Sadourny, 1997), social (Lin et al., 2013), biological (Runge et al., 2019) or economic systems (Iyetomi et al., 2020; Camilleri et al., 2019).

1.1 Problem description

In the context of multivariate time series systems, the problem of lead–lag detection consists in identifying random variables that lead or lag other random variables. There are a number of ways to mathematically define and extract the pairwise relationship between time series. Different lead–lag definitions are compared using a-priori considerations in Sect. 3.1 and synthetic experiments in Sect. 4.

Once we have chosen a metric to capture lead–lag relations, we can represent the uncovered relations using a directed weighted network. The nodes of our network correspond to different time series variables. A directed edge $A \to B$ exists between nodes A and B if time series A leads time series B. The weight of this edge is given by the magnitude of the

pairwise lead–lag metric, thus encoding the strength of the relation. We are thus able to study the properties of lead–lag relationships using the tools of network analysis.

A key question in network analysis concerns community detection (Newman, 2018). Does there exist a clustering of nodes such that node similarity is, on average, stronger within clusters than between clusters? In the context of a directed network encoding lead–lag relations, the question of community detection can be framed in terms of identifying clusters that exhibit high pairwise cut imbalance, as follows. We regard the flow along a directed weighted edge $A \to B$ as a measure of the extent to which A leads B. In a directed graph with adjacency matrix A, the cut associated to two subsets of nodes $\mathcal{A}$ and $\mathcal{B}$, is given by $Cut(\mathcal{A}, \mathcal{B}) = \sum_{i \in \mathcal{A}, j \in \mathcal{B}} A_{ij}$, and we refer to the difference $Cut(\mathcal{A}, \mathcal{B}) - Cut(\mathcal{B}, \mathcal{A})$ as the *cut imbalance*. A high cut imbalance between communities $\mathcal{A}$ and $\mathcal{B}$ indicates that variables in $\mathcal{A}$ are, on average, leaders of variables in $\mathcal{B}$. Therefore, by identifying pairs of clusters with high imbalance, we segment our multivariate system into communities that, taken in pairs, are mostly composed of either leaders or laggers. In Sect. 3, we describe a Hermitian-based directed network clustering algorithm that is suited for this task following (Cucuringu et al., 2020).

The application domain studied in this paper is that of financial time series. In this domain, each time series corresponds to the return time series for a particular financial instrument. We investigate the lead–lag cluster structure of the US equity market. In particular, we are interested in four questions. Does there exist a statistically significant cluster structure in the US equity market? What is the nature of the data-driven clustering? How does the data-driven cluster structure relate to previously discovered lead–lag mechanisms? Can we leverage our clustering for downstream forecasting purposes?

1.2 Key contributions

Our primary contribution is the introduction of a principled method, which, to the best of our knowledge, is the first to address the problem of unsupervised clustering of leading and lagging variables in multivariate time series systems. We validate different components of our method on synthetic and real data sets. Our secondary contribution consists of an evaluation of novel pairwise lead–lag metrics using a new benchmark data generating process for multivariate time series systems with clustered lead–lag structure. Thirdly, the application of our method to US equity data provides insights into the structure of the US equity market. To the best of our knowledge, our work presents the first data-driven clustering of lead–lag networks in a financial market context. Finally, we construct a novel statistically significant trading signal for the US equity market—thus demonstrating how our method can be employed to extract valuable signals in the high-dimensional, low signal-to-noise data setting.

1.3 Paper outline

We discuss existing literature related to our work in Sect. 2. Section 3 describes our approach to solving the lead–lag extraction and clustering problems. In Sect. 4, we validate our method on synthetic data sets. We present the results of applying our algorithm to a universe of US equities in Sect. 5. In Sect. 6, we illustrate the use of our methodology in a financial forecasting application. Finally, we summarise our main findings in Sect. 7.

2 Related work

There exists substantial evidence of lead–lag relations at the scale of monthly, weekly and daily financial returns (Lo & MacKinlay, 1990; Badrinath et al., 1995; Brennan et al., 1993; Chordia & Swaminathan, 2000; Menzly & Ozbas, 2010; Cohen & Frazzini, 2008), as well as at higher frequencies (Huth, 2012; Wang et al., 2017a; Curme et al., 2015b, a). In addition, a number of studies have considered lead–lag relations from the point of view of networks (Curme et al., 2015a; Fiedor, 2014; Výrost et al., 2015; Liao et al., 2014; Sandoval, 2014; Billio et al., 2012; Wang et al., 2017b; Wu et al., 2010). Commonly studied questions in this financial lead–lag network literature concern the cluster structure of the lead–lag network (Sandoval, 2014; Billio et al., 2012; Liao et al., 2014; Wang et al., 2017b; Xia et al., 2018; Biely & Thurner, 2008). A number of papers consider the relative influence of different industry sectors within the lead–lag network (Biely & Thurner, 2008; Liao et al., 2014; Xia et al., 2018). The influence of various sub-sectors within the lead–lag network of financial institutions is also a particular question of concern (Billio et al., 2012; Wang et al., 2017b; Sandoval, 2014). For example, Billio et al. (2012) relate the lead–lag network to the systemic exposure of financial firms and sub-sectors, in order to understand their respective financial drawdowns during crisis periods. In addition, the effect of geography-based clusters has also been investigated (Sandoval, 2014).

A second commonly studied problem in the financial lead–lag literature is that of ranking. A number of lead–lag network papers focus on how network tools may be used to identify financial instruments that exhibit stronger tendencies to lead other instruments (Liao et al., 2014; Billio et al., 2012; Wu et al., 2010; Basnarkov et al., 2019; Stavroglou et al., 2017). For example, Wu et al. (2010) and Basnarkov et al. (2019), apply the PageRank algorithm (Google, 2012) to the lead–lag network in order to extract an ordering of equities in terms of their influence on the future values of other equities.

In addition to the literature on financial *lead–lag* correlation networks, there is also substantial literature on *synchronous* correlation networks (Tumminello et al., 2010; Namaki et al., 2011; Sandoval & Franca, 2012; Marti et al., 2019). The reader is referred to Marti et al. (2019) for an extensive review of clustering on (mostly) synchronous financial correlation networks.

Our empirical analysis is novel within the financial lead–lag literature since it is the first work to extract a data-driven clustering of the lead–lag network. In contrast, previous studies (Sandoval, 2014; Billio et al., 2012; Liao et al., 2014; Sandoval, 2014; Wang et al., 2017b; Xia et al., 2018; Biely & Thurner, 2008) are only able to capture the influence of predefined groups, which are given, for instance, by industry sector (Biely & Thurner, 2008; Liao et al., 2014; Xia et al., 2018) or geography (Sandoval, 2014), within the financial lead–lag network. We believe that the academic interest in our data-driven clustering approach is underscored by the plurality of papers (Marti et al., 2019) that apply data-driven clustering to synchronous correlation networks, as well as the number of papers that apply data-driven ranking methods to lead–lag networks (Liao et al., 2014; Billio et al., 2012; Wu et al., 2010; Basnarkov et al., 2019; Stavroglou et al., 2017). Furthermore, our work is the first to show that clustered lead–lag network structure can be successfully used for downstream out-of-sample prediction tasks.

3 Method

Our method is a pipeline consisting of three steps. First, we apply a pairwise lead–lag metric to capture the lead–lag relationship between each pair of time series; this results in a network of lead–lag relationships. Second, we apply a directed network clustering method to extract a partition of the multivariate system such that there is a large flow imbalance [net sum of weights of inter-cluster edges (Cucuringu et al., 2020)] between cluster pairs. The third step quantifies the *leadingness* of each cluster.

There are a number of choices for each of these components in our pipeline. In this section, we describe metrics that can be used to quantify lead–lag relations between pairs of time series, and available directed network clustering methods.

To introduce notation, let X_t^i denote the random value of the time series variable $i \in \{1, \ldots, p\}$ at time $t = 0, \ldots, T$. Further, define the first differences $Y_t^i = X_t^i - X_{t-1}^i$ for $i \in \{1, \ldots, p\}, t = 0, \ldots, T$.

In our application domain of US equities, X_t^i denotes the logarithm of the closing price for stock $i \in \{1, \ldots, p\}$ on day $t = 0, \ldots, T$. Hence Y_t^i provides the corresponding log-return for equity i from day $t-1$ to t. It is suitable to use log-returns for analysis as they exhibit closer to stationary properties, and log-returns are more mathematically tractable than linear or percentage returns in the computation of multi-horizon returns (Campbell et al., 1997).

3.1 Pairwise metrics of lead–lag relationship

In a complex, non-linear system such as the US stock market, determining a suitable way to define a metric to capture lead–lag relationships is challenging. Here we present some options.

3.1.1 Lead–lag metrics based on a functional of the cross-correlation

A commonly used approach to defining a lead–lag metric is to use a functional of a sample cross correlation function (ccf) between two time series. The general form of a *sample cross-correlation function* between time series i and j evaluated at lag $l \in \mathbb{Z}$ is given by

$$\mathrm{CCF}^{ij}(l) = \mathrm{corr}\Big(\{Y_{t-l}^i\}, \{Y_t^j\}\Big), \tag{1}$$

where corr denotes a choice of sample correlation function. The corresponding *lead–lag metric*, a measure of the extent to which i leads j, is then obtained by

$$S_{ij} = F(\mathrm{CCF}^{ij}), \tag{2}$$

where F is a suitable functional.

In this paper, we consider four choices for the sample correlation function corr, namely Pearson linear correlation, Kendall rank correlation (Kendall, 1938), distance correlation (Székely et al., 2007), and mutual Information based on discretised time series values (Fiedor, 2014). The four different sample correlation functions are able to detect different dependencies. Pearson correlation is able to detect linear dependencies, Kendall rank correlation is able to detect monotonic non-linear dependencies, while distance correlation

and mutual Information are able to detect general non-linear dependencies. The drawback of non-linear sample correlation functions is that they have lower power in the case of a true linear relationship.

Further, we consider two choices for the functional F, as follows

1. **ccf-lag1**: computes the difference of the cross-correlation function at $lag \in \{-1, 1\}$

$$S_{ij} = \mathrm{CCF}^{ij}(1) - \mathrm{CCF}^{ij}(-1),$$

2. **ccf-auc**: computes the signed normalised area under the curve (auc) of the cross-correlation function

$$S_{ij} = \frac{\mathrm{sign}(I(i,j) - I(j,i)) \cdot \max(I(i,j), I(j,i))}{I(i,j) + I(j,i)},$$

where $I(i,j) = \sum_{l=1}^{L} \left| \mathrm{corr}\left(\{Y^i_{t-l}\}, \{Y^j_t\} \right) \right|$ for a user-specified maximum lag L.

The **ccf-lag1** method used with Pearson correlation is a crude lead–lag indicator (Campbell et al., 1997). This lead–lag indicator is only designed to take into account positive cross-correlation. Indeed, like the signatures-based method described further below in Sect. 3.1.2, it is only able to correctly determine the direction of the lead–lag relationship under a positive cross-correlation association between time series. Thus, this lead–lag indicator should be restricted to domains such as US equity returns, where cross-correlations between time series variables are predominantly positive (Campbell et al., 1997).

The **ccf-auc** method accounts for both positive and negative associations across multiple lags $l \in \{-L, \ldots, L\}$. The maximum lag L can be chosen a-priori as the maximum time lag expected in the multivariate system, or by using cross-validation on some downstream validation criterion. The averaging approach **ccf-auc** presented here is similar to the lag aggregation methodology of Wu et al. (2010).[1]

Overall, we consider eight possible choices for lead–lag metrics based on functionals of the cross-correlation. This stems from four possible choices for correlation (Pearson, Kendall, distance correlation and mutual information) and two possible choices for the functional form (**ccf-lag1** and **ccf-auc**).

The functional cross-correlation approach is flexible and computationally simple. The flexibility of the framework permits the use of robust and non-linear correlation metrics. The use of such non-linear correlation metrics is particularly useful for the extraction of lead–lag relationships in the financial time series domain, where linear cross-correlations between returns are expected to be low. High information efficiency in US equity markets (Malkiel & Fama, 1970) implies that linear return cross-correlations are too low to be used to construct trading systems that have expected returns in excess of market equilibrium expected returns. On the other hand, a stylised feature of financial returns is volatility clustering (Cont, 2001); the size of the cross-correlation between the volatility of returns is expected to be larger than the cross-correlation between the raw returns themselves. A linear cross-correlation approach is unable to capture the relationship between the volatility of two instruments across time. Empirical studies have also found that stronger lead–lag

[1] We have also considered using a maximum aggregation approach, and have found similar qualitative results to the averaging-based approach presented in this paper; however, the maximum aggregation approach tends to perform slightly worse than the averaging-based approach.

relationships can be detected when taking into account volatility (Billio et al., 2012). Thus, when comparing the time-dependence in returns between two assets, we should allow for non-linear effects (Fiedor, 2014). In addition, the functional cross-correlation approach easily permits the use of correlation metrics that are robust to outliers. Since financial times series exhibit heavy tails (Cont, 2001), robustness constitutes an important feature for a lead–lag extraction method. In general, the functional cross-correlation component and, consequently, the entire pipeline will be robust to outliers if the choice of correlation metric is robust to outliers. For example, ordinal association correlation metrics such as Kendall correlation guarantee robustness to outliers.

The linear Granger causality approach that is often considered in financial lead–lag studies (Shojaie & Fox, 2021; Skoura, 2019) can be viewed as an extension of our functional linear cross-correlation-based approach that takes into account auto-correlation and also filters for statistical significance. General Granger causality methods may also use non-linear functional forms to capture the association between time series. These more general methods can be used as the lead–lag extraction component of our method. Following the vector auto-regressive modelling example of Skoura (2019), bi-variate modelling can be use to determine the existence and direction of a lead–lag relation between two pairs of time series. Thus, a vector auto-regressive modelling approach can be used to derive a lead–lag metric and therefore be used as the lead–lag extraction component of our model. For the purposes of demonstrating our lead–lag extraction and clustering method, simpler functional cross-correlation approaches will suffice. Since the combination of data auto-correlation and co-movement can produce lead–lag associations between time series variables using our method, one must be careful not to interpret resulting lead–lag associations as apparent causal influence estimates.

We contrast our approach, which is based on correlation networks, with causality-based approaches that attempt the more difficult problems of recovering the casual network underlying a multivariate time series system (Runge et al., 2019) and quantifying its causal influences (Janzing et al., 2013). For example, whereas Runge et al. (2019) attempt to estimate the causal network underlying the time-lagged dependency structure in a given multivariate time series system, our aim is estimating and clustering the association-network for the multivariate time series system. Association-based approaches are more common in the financial network lead–lag literature (Marti et al., 2019), since financial time series have very noisy returns and exhibit weak lead–lag effect sizes due to the informational efficiency of the market (Malkiel & Fama, 1970). These characteristics of financial returns make the problem of accurately estimating a lead–lag correlation network (let alone the causality network) challenging in itself.

3.1.2 Lead–lag metric based on signatures

The approach of using a functional of the cross-correlation function relies on the user to specify the choice of functional; this choice is not obvious in many cases. In particular, it is difficult to gauge the number of lags to incorporate into our lead–lag metric a-priori. An alternative approach draws on the idea of signatures from rough path theory (Levin et al., 2016), in order to construct a pairwise lead–lag metric. The signature of a continuous path with finite 1-variation (Levin et al., 2016) $X : [a, b] \to \mathbb{R}^d$, denoted by $S(X)_{a,b}$, is the collection of all the iterated integrals of X, namely $S(X)_{a,b} = (1, S(X)^1_{a,b}, \ldots, S(X)^d_{a,b}, S(X)^{1,1}_{a,b}, S(X)^{1,2}_{a,b}, \ldots)$, where the iterated integrals are given by

$$S(X)^{i_1,\ldots,i_k}_{a,t} = \int_{a<t_k<t} \cdots \int_{a<t_1<t_2} dX^{i_1}_{t_1} \ldots dX^{i_k}_{t_k}.$$

Based on the proposal in Levin et al. (2016), the signatures-based pairwise measure of the lead–lag relation between two stocks i and j over the time period $[t-m, t]$ is given by

$$S_{ij}(t-m,t) = \iint_{t-m<u<v<t} (dX^i(u)dX^j(v) - dX^j(u)dX^i(v)). \tag{3}$$

This is the difference in the cross-terms of the second level of the time series signature of the log-prices. Theoretical results in rough path theory (Levin et al., 2016) have established that a signature is essentially unique to the path it describes, and that the truncated signature (i.e. the lower order terms) can efficiently describe the path. Chevyrev and Kormilitzin (2016) provide an interpretation of the signature lead–lag metric (3). The signature lead–lag metric is positive and grows larger whenever increases (resp. decreases) in X^i are followed by increases (resp. decreases) in X^j. If the relative moves of X^i and X^j are in the opposite directions, then the signature lead–lag measure is negative. Note that a downside of this method is that it is not able to tell the difference between

1. $i \to j$ with negative association,
2. $i \leftarrow j$ with positive association.

As a result, we do not expect the method to perform well when there is significant negative association in the lead–lag data generating process.

When analysing price data observed at discrete time points, we transform the data stream into a piecewise linear continuous path and calculate the second order signatures (Reizenstein & Graham, 2018). From this, we may calculate the lead–lag relation using the difference in second order signature cross-terms (3). We refer the reader to Gyurkó et al. (2014) for additional details on signatures and their application in a financial context, along with an interpretation in terms of second order areas and interplay with lead–lag relationships. In practice, when comparing the signature lead–lag metrics across different pairs of time series, we recommend the normalisation of the price data prior to computation of the lead–lag metric, since the absolute value of the metric is increasing in the volatility of the underlying price series.

3.1.3 Alternative lead–lag metrics

The lead–lag extraction approaches mentioned in this section are by no means exhaustive. Indeed, alternative methods can be found within the financial time series lead–lag literature (Wang et al., 2017a). Furthermore, the functional cross-correlation framework presented in this paper is agnostic to the choice of the correlation metric used within it. As such, it is able to draw on a wide array of non-linear correlation metrics such as target/forget dependence coefficient (Marti et al., 2016), maximal information coefficient (Reshef et al., 2011) or maximum mean discrepancy (Gretton et al., 2012).

The detection of lead–lag relations can be attempted in the frequency-domain as well as the time-domain (Skoura, 2019). Wavelet techniques do not rely on time series stationarity and have been shown to provide a more nuanced understanding of lead–lag relations when used in conjunction with time-domain analysis (Skoura, 2019). However, the wavelet

coherence approach (Skoura, 2019) does not straightforwardly provide a single lead–lag metric between two time series since this would require a method for aggregating across wavelet location and scale parameters. Further, the wavelet coherence method also takes into account synchronous correlation between two time series: this is not desirable for the computation of a lead–lag relation metric. Further work is required to develop a single lead–lag metric based on wavelet coherence that could be used in our lead–lag extraction and clustering pipeline.

3.2 Algorithms for clustering directed networks

Let S_{ij} denote the user-defined lead–lag metric that quantifies how much time series variable i leads j. The value S_{ij} can be positive or negative, and satisfies $S_{ij} = -S_{ji}$. The lead relationships between all pairs of time series is encoded by the asymmetric matrix $A_{ij} = \max(S_{ij}, 0)$. We apply directed network clustering algorithms to the weighted and directed network G, where each node corresponds to a time series variable and the adjacency matrix is A. In this section, we present different relevant clustering methods for such directed networks.

Note that as a pre-processing step for any of the clustering methods described below, it is possible to filter the pairwise measurements S_{ij} when constructing the network A. For example, Curme et al. (2015a) apply significance thresholding, whereby an edge exists between two nodes only if the corresponding lead–lag metric is sufficiently large in magnitude.

3.2.1 Naive symmetrisation clustering

Popular undirected network clustering methods, such as spectral clustering (Shi & Malik, 2000), cannot be immediately applied to directed networks, since directed networks with asymmetric adjacency matrices have complex spectra. Traditional approaches for directed network clustering have applied spectral analysis to a symmetrised version of the directed network adjacency matrix (Sussman et al., 2012; Pentney and Meila, 2005). We consider a commonly used naive symmetrisation-based directed clustering method as a baseline (Satuluri & Parthasarathy, 2011). This naive method applies a standard spectral clustering algorithm (Shi & Malik, 2000) to the undirected network with adjacency matrix $\tilde{A} = A + A^T$. In this paper, the spectral clustering algorithm applied to the derived undirected networks uses k-means clustering on a projection onto the first k non-trivial eigenvectors of the random-walk normalised graph Laplacian (we drop the first eigenvector since for connected networks it is always the unit vector). The value of k, corresponding to the desired number of clusters, is a hyperparameter of the algorithm.

3.2.2 Bibliometric symmetrisation clustering

Naive symmetrisation methods produce a clustering that only takes into account edge density and not edge direction. As a result, they are unable to target clusterings with high pairwise flow imbalance between clusters. Satuluri and Parthasarathy (2011) propose the degree-discounted bibliometric symmetrisation that is able to take into account edge direction information. In the degree-discounted bibliometric symmetrisation, spectral clustering is applied to the adjacency matrix

$$\tilde{A} = D_o^{-1/2} A D_i^{-1/2} A^T D_o^{-1/2} + D_i^{-1/2} A^T D_o^{-1/2} A D_i^{-1/2},$$

where D_i is the diagonal matrix of weighted in-degrees and D_o is the diagonal matrix of weighted out-degrees. The degree-discounted bibliometric symmetrisation applies degree-discounting to a symmetrisation that sums the number of common in- and out-links between two pairs of nodes. Therefore, clusters produced by this method are expected to group together nodes that have a relatively large number of parent (sender) and children (receiver) nodes in common (Satuluri & Parthasarathy, 2011). Degree-discounting is a technique that has been found to work well in tasks on graphs with highly skewed degree distributions.

3.2.3 DI-SIM co-clustering

Rohe et al. (2016) propose a co-clustering algorithm for directed networks. The co-clustering algorithm first computes a regularised graph Laplacian using A; this initial step is performed so that the algorithm may deal with heterogeneous and sparse data. Then, co-clustering is performed by applying k-means on the k-largest of each of the left and right normalised singular vectors of the Laplacian. This co-clustering identifies two partitions of nodes: one partition groups together vertices with similar sending behaviour, while the other partition groups together vertices with similar receiving behaviour. In this paper, we denote the clustering obtained using the left singular vectors as **DI-SIM-L** and the clustering obtained using the right singular vectors as **DI-SIM-R**. We consider both choices of clustering in our experiments.

3.2.4 Hermitian clustering

The Hermitian clustering procedure (Cucuringu et al., 2020) for clustering directed networks considers the spectrum of the complex matrix $\tilde{A} \in \mathbb{C}^{p \times p}$, which is derived from the directed network adjacency matrix as $\tilde{A} = i(A - A^T)$. Since $\tilde{A}$ is Hermitian, it has p real-valued eigenvalues which we order by magnitude $|\lambda_1| \geq \ldots \geq |\lambda_p|$. The eigenvector associated with λ_j is denoted by $g_j \in \mathbb{C}^p$ where, in Euclidean norm, $\| g_j \| = 1$ for $1 \leq j \leq p$.

Algorithm 1 describes the procedure for clustering the directed network G. In our implementation, we set the number of top eigenvectors used to $l = k$.

Algorithm 1 Hermitian clustering algorithm.

Input: A directed graph $G = (V, E)$ with Hermitian adjacency matrix $\tilde{A}$; number of clusters $k \geq 2$; $\epsilon > 0$

1. Compute all the eigenvalue/eigenvector pairs of $\tilde{A}$ $\{(\lambda_1, g_1), (\lambda_2, g_2), \ldots, (\lambda_l, g_l)\}$ satisfying $\| g_j \| = 1$ and $|\lambda_j| > \epsilon$, $\forall j \in \{1, \ldots, l\}$
2. $P \leftarrow \sum_{j=1}^{l} g_j g_j^T$
3. Apply a k-means algorithm with input rows of P
4. Return a partition of V corresponding to the output of k-means

Note that in practice, for scalability purposes, one can bypass the computation of the entire $n \times n$ matrix P in order to directly cluster using the embedding given by the top l eigenvectors.

Cucuringu et al. (2020) study the performance of the algorithm theoretically and experimentally under data generated from a directed version of a stochastic block model that embeds latent structure in terms of flow imbalance between clusters. They show that the algorithm is able to discover cluster structures based on directed edge imbalance. This contrasts with previous spectral clustering methods that detect clusters based purely on the edge-density of symmetrised networks. The Hermitian clustering algorithm is particularly suited to our setting of clustering lead–lag networks, since we aim to extract pairs of clusters with high flow imbalance. In addition, as a pre-processing step for this algorithm, we apply random-walk normalisation to the adjacency matrix $\tilde{A}$, so that the method is robust to heterogeneous degree distributions (Cucuringu et al., 2020); we refer to the resulting algorithm as the **Hermitian RW** algorithm.

3.3 Alternative clustering algorithms

State-of-the-art modularity clustering algorithms such as the Leiden algorithm (Traag et al., 2019) may be used on directed graphs using a directed modularity metric (Dugué & Perez, 2015). Dugué and Perez (2015) optimise a modularity metric that compares the number of edges within clusters to the expected number of edges under a null model. However, such modularity-based algorithms, which return clusters based on edge density, are not suited to our goal of uncovering clusters of leading and lagging variables based on flow imbalance between clusters.

An adaptation of the Hermitian clustering method has been proposed in Laenen and Sun (2020). The method presented in Laenen and Sun (2020) aims to discover a clustering that maximises a normalised flow metric between communities using the spectrum of a normalised Hermitian Laplacian matrix. As such, it could be used as an alternative to the Hermitian RW clustering algorithm considered in this paper. Also recently, Underwood et al. (2020) proposed an algorithm for clustering weighted directed networks that employs motif-based spectral clustering to uncover flow imbalance relationships between pairs of clusters.

Lastly, we draw attention to a recent approach introduced in He et al. (2021) that extends the Hermitian-based clustering algorithm (Cucuringu et al., 2020). This recent method departs from standard approaches in the literature, and treats edge directionality not as a nuisance but rather as the main latent signal. It does so by introducing a graph neural network framework for obtaining node embeddings for directed networks in a self-supervised manner, while accounting for node-level covariates.

3.4 The leadingness metric

We introduce the concept of a *meta-flow graph* in order to capture the aggregate weighted flow between pairs of clusters. The total *flow* between any two clusters is given by the net of the normalised weights between all edges directed from one cluster to another. The skew-symmetric matrix that encodes this information is dubbed the *meta-flow* matrix, which we denote by F. Mathematically

$$F_{ij} = \frac{1}{|C_i|\,|C_j|} \sum_{l \in C_i, m \in C_j} \left[A_{lm} - A_{ml}\right],$$

where C_a denotes the set of all nodes in cluster $a \in \{1, \ldots, k\}$, and $i, j \in \{1, \ldots, k\}$, $i \neq j$. The diagonal of F consists of zeros: $F_{ii} = 0$, $\forall i \in 1, \ldots, k$. We also define a metric for the *leadingness* of each cluster $i \in \{1, \ldots, k\}$,

$$L(i) := \frac{1}{|C_i|} \sum_{l \in C_i, m \in \{1,\ldots,p\}} \left[A_{lm} - A_{ml}\right]. \tag{4}$$

Thus, $L(i)$ averages the row-sums of the skew-symmetric matrix $A - A^T$ for nodes within the cluster i; the row-sums of the lead–lag matrix provide a measure of the total tendency of the equity corresponding to the row to be a leader (Huber, 1962). From this metric, we obtain a ranking of the clusters from the most leading cluster (largest row-sum value), which we will label 0, to the most lagging cluster (smallest row-sum value), which has the largest numeric label $k - 1$. In this paper, all data-driven clustering results will be presented using this labelling. The RowSum Ranking (Huber, 1962; Gleich & Lim, 2011) algorithm is an instance of a ranking method that recovers a latent ordering of variables given variable pairwise comparisons. There exists a rich literature on ranking from pairwise comparisons. The goal in this literature is to infer the strength $\ell_i, i = 1, \ldots, p$ or ranking of p items given a (potentially incomplete) set of pairwise comparisons which encode a noise proxy for $\ell_i - \ell_j$. Alternative ranking algorithms that could be employed for defining the leadingness of a cluster include (Fogel et al., 2016; Cucuringu, 2016; De Bacco et al., 2018; d'Aspremont et al., 2021; Bradley & Terry, 1952; Page et al., 1998), as well as Chau et al. (2020) for rankings that incorporate any available node level covariates.

3.5 Algorithmic complexity of the method

Let us denote by ψ the cost of the pairwise lead–lag metric of choice. The cost of the lead–lag network construction step amounts to $O(p^2\psi)$, where p is the number of time series. For example, for the linear Pearson correlation $O(\psi) = O(TL)$, where L is the number of lags, and T is the sample size. The cost of a spectral clustering algorithm for k clusters, such as Hermitian clustering (Cucuringu et al., 2020), is $O(kp^2) < O(p^3)$. Therefore, the overall complexity amounts to $O(p^2TL + kp^2)$.

In the large p setting, the above pipeline can become computationally prohibitive. One approach to alleviate this amounts to subsampling m pairs of time series out of the $\binom{p}{2}$ choices. This will lead to a comparison lead–lag matrix with only m nonzero entries; for example, the choice of sampling each edge with probability $\frac{\log p}{p}$ renders $m = O(p \log p)$. Since computing the leading eigenvectors of a sparse matrix via an iterative power method-based approach can be performed in a running time that is linear in the number of nonzero entries in the matrix, this step takes $O(p \log p)$ time. Thus, the approximate method is almost linear in the number of edges in the comparison graph. If the underlying pairwise comparison graph is weakly connected, which in practice will be the case because correlations will not be zero, then a choice of sampling probability of $O(\frac{\log p}{p})$ results in a pairwise comparison graph that is weakly connected with high probability. In such a situation we would expect the clustering in the sampled network to be a reasonable reflection of the clustering in the true network. We refer the reader to Batson et al. (2013) for spectral

algorithms and theoretical considerations of the closely related graph sparsification literature, and to Hu and Lau (2013) for a survey and taxonomy of graph sampling techniques.

4 Synthetic data experiments

The purpose of this section is to validate our method on synthetic experiments in which the ground truth lead–lag relationships and clusters are known. This approach will also give an indication of the relative performance of each of our lead–lag metrics and clustering components, under different data generating settings.

4.1 Synthetic data generating process

We introduce five different lagged latent variable synthetic generating processes to test our method. The general form of these synthetic generating processes is a latent variable model whereby the lagged dependence on the latent variable Z induces the clustering amongst the different times series $\{Y_t^i\}$. Mathematically, the synthetic data generating processes take the form

$$\begin{aligned} &Z_t \overset{i.i.d.}{\sim} F_Z \, \forall t \in \{1, \dots, T\}, \quad Z_t := 0 \quad \forall\, t \leq 0, \\ &Y_t^i = g_{l_i}(Z_{t-l_i}) + \epsilon_t^i, \quad \epsilon_t^i \overset{i.i.d.}{\sim} N(0, \sigma_\epsilon^2) \; \forall t \in \{1, \dots, T\}, \; i \in \{1, \dots, p\}, \end{aligned} \tag{5}$$

where the lag corresponding to time series variable i is $l_i \in L$ and L is the set of lag values. The choice of the shared latent variable distribution F_Z and the functional dependencies $g_l, l \in L$ on the latent variable Z determines the data generating process. The factor-based form of the synthetic data generation is motivated by our application to US equities (Fama & French, 1993; Jegadeesh & Titman, 1995). For instance, early work by Jegadeesh and Titman (1995) studies a lagged factor model in the context of lead–lag effects. See also Sect. 5 for a discussion of hypothesised clustered lead–lag return structures in the US equity market. The synthetic data generating process considered in this section is a toy model that is designed to test whether our method can correctly detect and cluster time series in a factor-driven scenario.

The five particular forms of (5) that we will consider are as follows.

1. *Linear*

$$F_Z = N(0, 1) \quad \text{and} \quad Y_t^i = Z_{t-l_i} + \epsilon_t^i. \tag{6}$$

2. *Cosine*

$$F_Z = U(-\pi, \pi) \quad \text{and} \quad Y_t^i = \frac{1}{\sqrt{\pi}} \cos(l_i \cdot Z_{t-l_i}) + \epsilon_t^i. \tag{7}$$

3. *Legendre*

$$F_Z = U(-1, 1) \quad \text{and} \quad Y_t^i = P_{l_i+1}^L(Z_{t-l_i}) + \epsilon_t^i. \tag{8}$$

4. *Hermite*

$$F_Z = N(0,1) \quad \text{and} \quad Y_t^i = \frac{1}{\sqrt{l_i!}} P_{l_i+1}^H(Z_{t-l_i}) + \epsilon_t^i. \tag{9}$$

5. *Heterogeneous*

$$Z \in \mathbb{R}^K,\ F_Z = N_{K\times K}(0, I_{K\times K}) \quad \text{and} \quad Y_t^i = Z_{t-l_i}^{f_i} + \epsilon_t^i. \tag{10}$$

Here, P_l^L and P_l^H are respectively the Legendre polynomial of degree l and the Hermite polynomial of degree l. In the heterogeneous case, the superscript $f_i \in \{1, \ldots, K\}$ indicates on which component of the multivariate factor Z the time series i depends.

In these five data generating process scenarios, by design, the cross-covariance at lag $k \in \mathbb{N}$ between any two time series $i, j \in \{1, \ldots, p\}$ is

$$\mathbb{E}\left[(Y_{t-k}^i - \mathbb{E}[Y_{t-k}^i])(Y_t^j - \mathbb{E}[Y_t^j])\right] = 0$$

whenever $k \neq l_j - l_i$ due to the independence of z_t across time. In the linear data generating case, setting (6), when $k = l_j - l_i$, then we have that $\mathbb{E}\left[(Y_{t-k}^i - \mathbb{E}[Y_{t-k}^i])(Y_t^j - \mathbb{E}[Y_t^j])\right] = \mathbb{E}[(Z_{t-l_j})^2] \geq 0$. This induces a linear dependence between time series i and time series j through the single non-zero value in the cross-covariance function between these two time series. Considering the whole network of lead–lag relations, we find that $i \to j$ (i is a leader of j) if and only if $l_i < l_j$. Since multiple time series share the same lag, this network is clustered: time series i and j share the same cluster if and only if $l_i = l_j$. Our synthetic experiments test our method's ability to correctly detect lead–lag relationships and recover the underlying ground-truth clustering structure of the lead–lag network.

The non-linear data generating settings (7)–(9) engender additional challenges for our lead–lag extraction method. Due to the respective orthogonality of the cosine functions $\{\cos(mx)\}_{m\in\mathbb{N}}$, Legendre polynomials $\{P_m^L(x)\}_{m\in\mathbb{N}}$ and Hermite polynomials $\{P_m^L(x)\}_{m\in\mathbb{N}}$, the linear cross-covariance evaluated at lag k between two time series i and j is zero even when $k = l_j - l_i$. Thus we expect metrics based on linear or cross-covariance methods to perform poorly in these settings. Non-linear lead–lag metrics are required in order to detect a non-linear dependence of time series j on time series i at lag $k = l_j - l_i$.

The heterogeneous data generating process setting adds a further independence condition on the relationship between two time series. In this case, the cross-covariance at lag k is nonzero if and only if both $k = l_j - l_i$ and $f_i = f_j$ are satisfied. The additional factor component equality condition implies that time series i and time series j share the same cluster if and only if $l_i = l_j$ and $f_i = f_j$.

In our simulation studies, we consider the performance of different configurations of our method as the noise level σ of our idiosyncratic error increases. The following experiment parameter choices are considered:

- Number of data points per time series: $T = 250$
- Number of time series: $p = 100$
- The standard deviation of the idiosyncratic noise: $\sigma \in \{0, 0.2, 0.4, 0.6, 0.8, 1, 2, 3, 4\}$
- Latent variable lag dependence for each time series by experiment setting:
  - *Linear*: $l_i = \lfloor \frac{i-1}{10} \rfloor$ for $i = 1, \ldots, 100$
  - *Cosine*: $l_i = \lfloor \frac{i-1}{10} \rfloor + 1$ for $i = 1, \ldots, 100$

- *Legendre and Hermite*: $l_i = \lfloor \frac{i-1}{10} \rfloor + 2$ for $i = 1, \ldots, 100$
- *Heterogeneous*: $f_i = \lfloor \frac{i-1}{50} \rfloor$ for $i = 1, \ldots, 100$ while $l_i = \lfloor \frac{i-1}{5} \rfloor$ for $i = 1, \ldots, 50$ and $l_i = \lfloor \frac{i-51}{5} \rfloor$ for $i = 51, \ldots, 100$.

The lag and factor structure implies that there are 10 clusters in the Linear, Cosine, Legendre and Hermite settings, while in the Heterogeneous setting there are 20 clusters. In each configuration of our method, we set the clustering algorithm hyperparameter corresponding to the number of clusters to be equal to the ground truth number of clusters. The remaining hyperparameter choices for the different method configuration components are:

- *ccf-auc*: the maximum cross-covariance lag: $L = 5$
- *DI-SIM co-clustering*: the regularisation parameter is set equal to the average row sum of the adjacency matrix (Rohe et al., 2016) and the number of singular vectors used in the co-clustering is set equal to the ground truth number of clusters in each synthetic data generating setting.
- *Naive, Bibliometric and Hermitian RW clustering*: the number of eigenvectors used in the respective spectral clustering projections is set equal to the ground truth number of clusters.

4.2 Performance metrics

We employ different performance criteria to evaluate both components—the lead–lag detection component and the clustering component—of our method. In order to evaluate the lead–lag detection component, we calculate the proportion of correctly classified edges in the true underlying lead–lag network (i.e. the accuracy of correctly classifying the direction of the lead–lag relationship between two time series). In order to evaluate the clustering component, we calculate the Adjusted Rand Index (ARI) between the ground-truth clustering and the clustering recovered by our method. The Adjusted Rand Index (Hubert & Arabie, 1985; Gates & Ahn, 2017) is a popular metric of success for a clustering algorithm which calculates its propensity to allocate of pairs of nodes that belong to the same (resp. different) cluster in the ground truth partition to the same (resp. different) cluster in the partition recovered by the algorithm.

4.3 Results

4.3.1 Marginal results over lead–lag extraction and clustering

We present the results for the lead–lag metric and clustering stages separately. In this section, we present results for the linear and cosine synthetic data generating settings; results for the other synthetic data generating settings can be found in Appendix Sections A.1 and A.2. For each experimental setting, we have generated 48 samples from the synthetic data generating process and applied our method to each one.

We display the average value and confidence interval for the lead–lag component detection classification accuracy over the 48 samples in the linear setting in Fig. 1 and the cosine setting in Fig. 2. The confidence interval is a 95% Gaussian for the classification accuracy computed on a sample from the data-generating process.

Figure 1 shows that the proposed lead–lag detection components are able to detect linear lead–lag associations and that their performance decreases to random chance performance

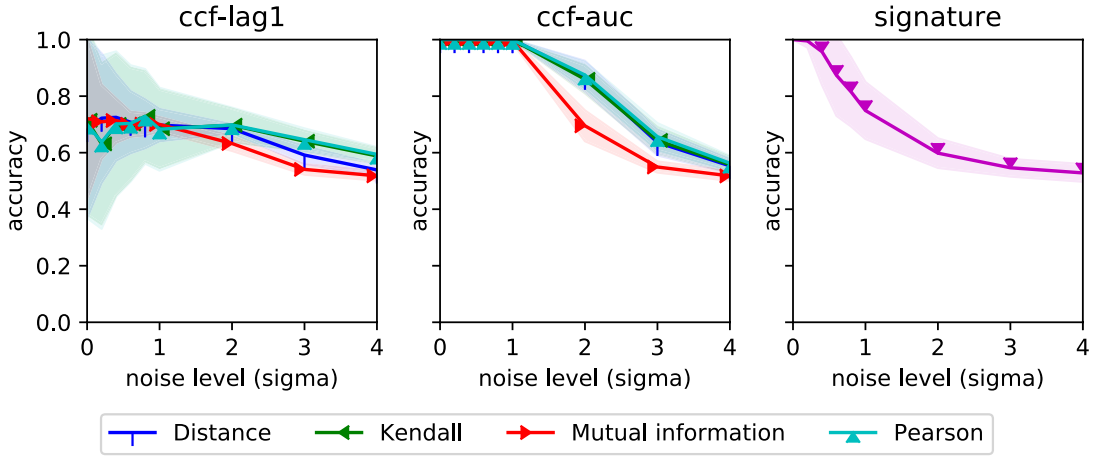


Fig. 1 Average and confidence interval for the classification accuracy by lead–lag detection method in the linear setting

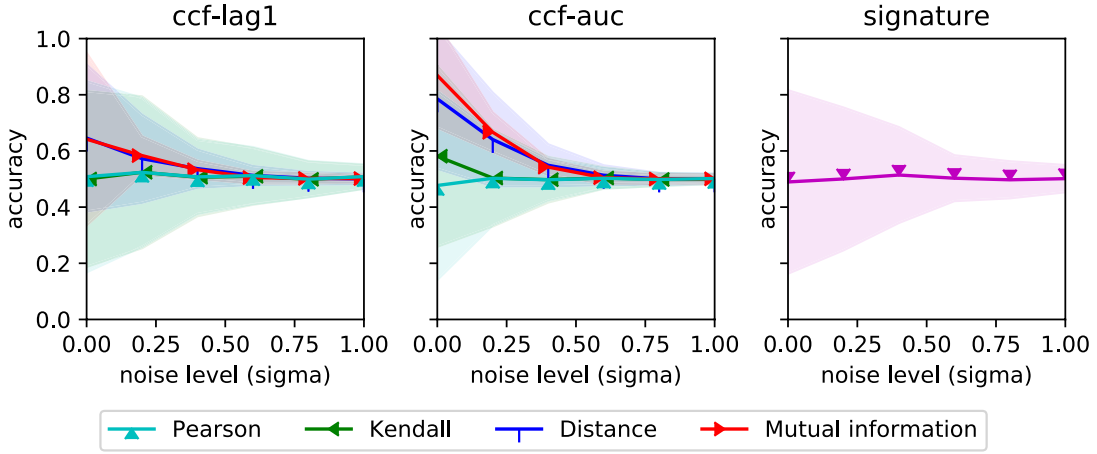


Fig. 2 Average and confidence interval for classification accuracy by lead–lag detection method in the cosine setting

as the level of noise in the synthetic data experiment increases. The **ccf-auc** and signature methods work best in this setting. Within the **ccf-auc** method, the non-linear Kendall and distance correlation metrics are able to maintain similar performance to the linear metric. The outperformance of the **ccf-auc** method over the **ccf-lag1** method shows the advantage of considering a larger number of lags in the cross-correlation function when pairs of time series depend on each other through large lag values.

The performance of the methods decreases in the cosine setting: the noise level at which the performance of all methods drops to that of random chance is about is around $\sigma = 0.5$ (compared with $\sigma = 4$ in the linear setting). In particular, the **ccf-lag1** and signature methods perform poorly; this is not a surprise since this method cannot deal with negative associations. The **ccf-auc** method using mutual information or distance correlation is able to achieve the highest accuracy; this illustrates the use of methods that are able to take into account negative and non-linear associations.

In order to compare the performance of different clustering methods, we compute, for each clustering method and experimental repetition, the marginal of ARI over the different lead–lag detection metrics. The mean and confidence interval for the ARI values over the experimental repetitions are shown in Figs. 3 and 4.

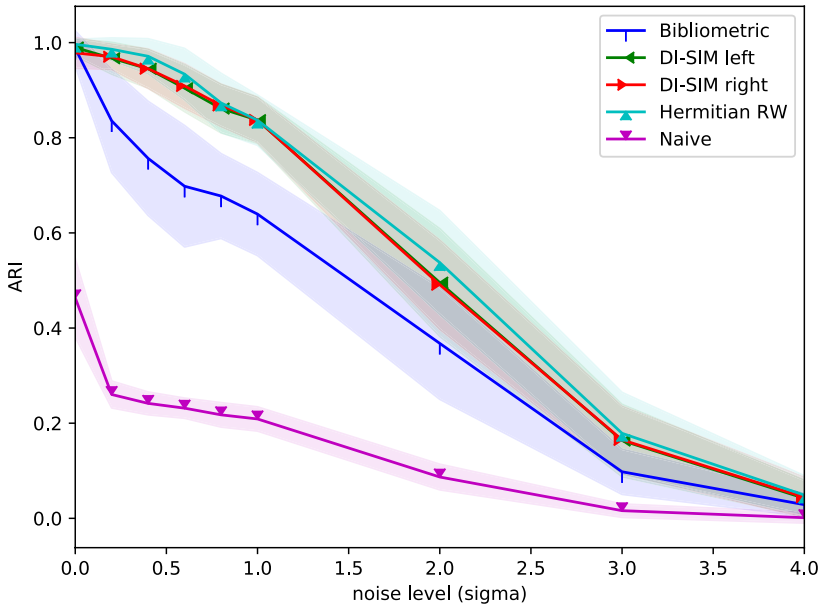


Fig. 3 Average and confidence interval for the ARI by clustering method in the linear setting

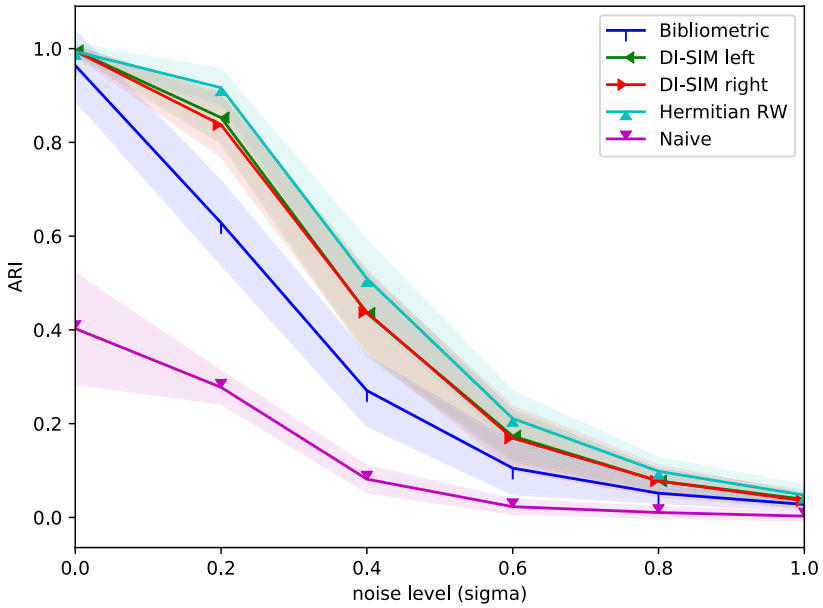


Fig. 4 Average and confidence interval for the ARI by clustering method in the cosine setting

We observe that the (non-naive) implementations of our method are able to recover almost perfectly (ARI of 1) the clustering in both settings (1) and (2) when σ is low. As expected, the performance of our methods decrease as σ increases; the performance in the cosine setting decreases faster than in the linear setting. The Hermitian RW and the DI-SIM clustering methods perform best in the settings considered. The Hermitian RW method targets clusters with high imbalance (Cucuringu et al., 2020) and is therefore particularly suited to the task of clustering time series according to directed imbalances in their lead–lag relations. The importance of edge direction is illustrated by the

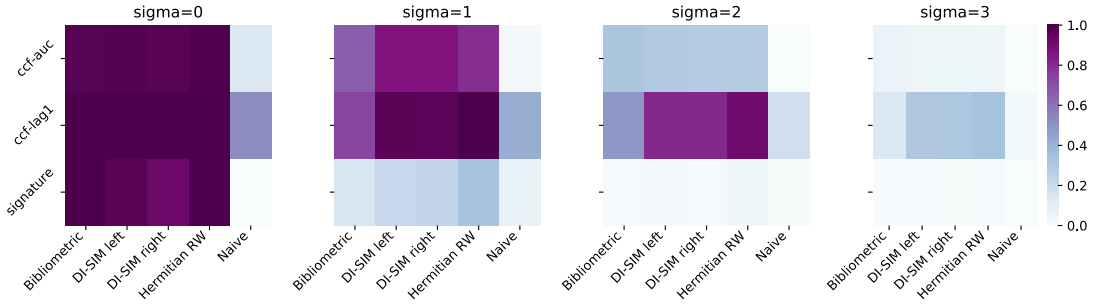


Fig. 5 Average ARI by lead–lag and clustering method in the linear setting

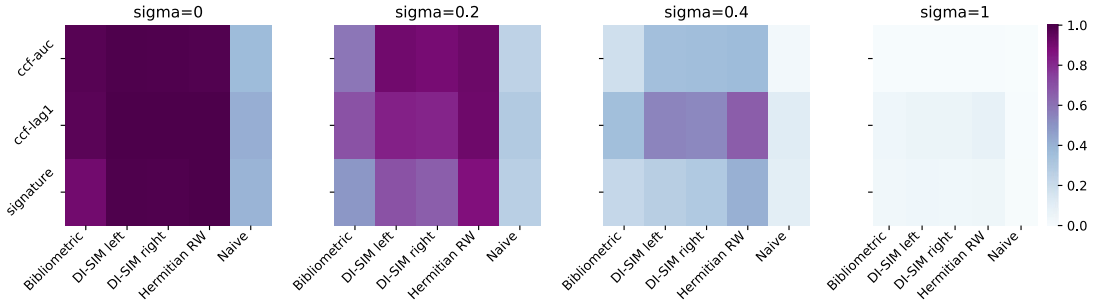


Fig. 6 Average ARI by lead–lag extraction and clustering method in the cosine setting

relatively poor performance of the naive method, which relies solely on the magnitude and not the direction of the edges.

Note that even as the number of lags considered in the cross-correlation function by the **ccf-lag1** and **ccf-auc** component method (1 and 5 lags, respectively) is lower than the largest lag dependence between any two pairs of time series (e.g. $l_{100} - l_1 = 9$ in the linear setting), our overall two-stage pipeline using these component methods is still able to leverage enough similarities in the dependence structure between the time series to correctly recover the ground-truth clustering. We are able to successfully cluster in this case since $\max_{i\in\{1,\ldots,p\}} \min_{j\in\{1,\ldots,p\}} |l_i - l_j| = 1$, which is less than or equal to the number of lags considered by the cross-correlation function methods.

Our experimental observations are robust to the other synthetic data generating processes reported in Appendix A. Similar results are also observed when performing simulation studies for a smaller number of time series and smaller sample sizes.

4.3.2 Interaction of lead–lag and clustering components

In this section, we investigate the joint dependence of the pipeline on the lead–lag and clustering components. The performance of the pipeline, measured by ARI averaged over the different Monte Carlo repetitions, is shown for linear and cosine synthetic data settings in Figs. 5 and 6; the other synthetic data settings are presented in Appendix A.3. For each synthetic data setting, we select a range of noise levels σ that are representative of the different levels of overall ARI significance. In these figures, for the pipelines using **ccf-lag1** and **ccf-auc** components, we show the ARI averaged across the 4 different choices of sample correlation function described in Sect. 3.1.1.

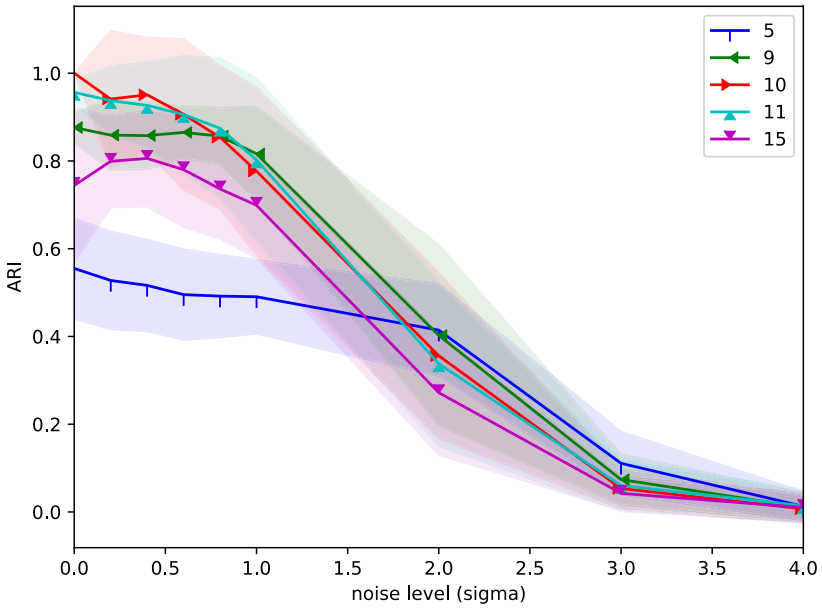


Fig. 7 Average and confidence interval for the ARI by different levels of the hyperparameter corresponding to the number of clusters in the linear setting

We observe that the pipelines that use **ccf-lag1** or **ccf-auc** lead–lag extraction components with **DI-SIM** or **Hermitian RW** as the clustering component tend to perform best. For small values of σ, the performance of each of these pipelines tends to be quite similar. For larger σ values, the relative performance difference between the different lead–lag extraction components tends to increase, with the performance of the DI-SIM and Hermitian RW components within a pipeline using **ccf-lag1** or **ccf-auc** appearing to be quite correlated. Eventually, the performance of every pipeline drops to 0 as σ increases.

4.3.3 Ablation study: varying hyperparameter corresponding to the number of clusters

We perform an ablation study to examine the sensitivity of the pipeline to the hyperparameter controlling the number of clusters returned by the clustering algorithm. In Fig. 7, we present the results for the typical linear synthetic data generating setting using a pipeline of **ccf-auc** with distance correlation and Hermitian RW clustering. Results for the cosine, Legendre and Hermite data settings are shown in the Appendix A.4.

The true underlying number of clusters in the linear data setting is 10 (see Sect. 4). In Fig. 7, we see that the performance of the pipeline is robust to small variations in the hyperparameter corresponding to the number of clusters around the true underlying number of clusters. Further, we find that using a large hyperparameter value for the number of clusters results in a large decay in the ARI of the pipeline.

4.3.4 Summary of synthetic data experiment results

To summarise this section, we have validated our pipeline on five synthetic data generating processes. While the choice of particular correlation components should be driven by the application in mind, the **ccf-auc** method using distance correlation achieves relatively strong performance both in the linear and in the cosine synthetic data generating settings.

The clustering component methods that were found to perform best were the DI-SIM and Hermitian RW methods.

5 US equity data experiment

It is well known that US equity returns exhibit a cross-sectional factor structure (Fama & French, 1993). Some of the prominent factors, for example the factors representing industry membership, can exhibit cluster membership. This induces a clustering structure in the synchronous cross-sectional equity returns (Farrell, 1974). In addition to this synchronous clustering structure, we conjecture that there exists a clustering structure in US equities due to inter-temporal relations in equity returns. In this section, our method is applied to construct and cluster a lead–lag network on a US equity universe, and investigate the resulting data-driven clustering. On the basis of a-priori considerations and performance under the synthetic data experiments, a lead–lag metric that computes distance correlation (Székely et al., 2007) between the shifted time series and a directed clustering method that uses the spectrum of a Hermitian adjacency matrix are suitable components for the application of our method to US equity returns. We will use **ccf-auc** with lags $l \in \{-5, \dots, 5\}$ with the distance correlation as our lead–lag metric, and Hermitian RW clustering as our clustering step. This method has the potential to capture non-linear lead–lag relations between returns on the scale of up to a week. The range of lag values is a hyperparameter of our method and in general, can be chosen using a-priori considerations or empirically selected using cross validation on a downstream loss. In our case, we set the range of lag values to $l \in \{-5, \dots, 5\}$, which allows us to capture lead–lag relations on daily and weekly scales (see Sect. 2). We set the number of clusters, a hyperparameter of our algorithm, to 10 in order to facilitate comparison with the industry-sector clustering of equities.

5.1 Data description

We consider the universe of 5325 NYSE equities spanning from 04-01-2000 to 31-12-2019 from Wharton's CRSP database (Wharton Research Data Service, 2020)—restricting our attention to equities trading on the same exchange to avoid spurious lead–lag effects due to non-synchronous trading (Campbell et al., 1997). The data consists of daily closing prices from which we compute daily log-returns. We also compute the average daily dollar volume that is traded for each equity. We subset to the equities that have the largest average volume (largest 500 equities in average volume) and the least number of missing values (at least 2.5 years' worth of non-missing data). This results in a data set of 434 equities. Filtering to the most traded equities with the least number of missing prices reduces the risk of spurious lead–lag effects due to non-synchronous trading (Campbell et al., 1997). Any remaining missing prices are forward-filled prior to the calculation of log-returns.

5.2 Data analysis

5.2.1 Illustration of US equity lead–lag matrix

Figure 8 shows a sorted skew-symmetric lead–lag matrix encoding the measurement between each pair of stocks. Positive entries in the matrix correspond to a leading relationship between the stock depicted on the vertical axis with respect to the stock depicted on

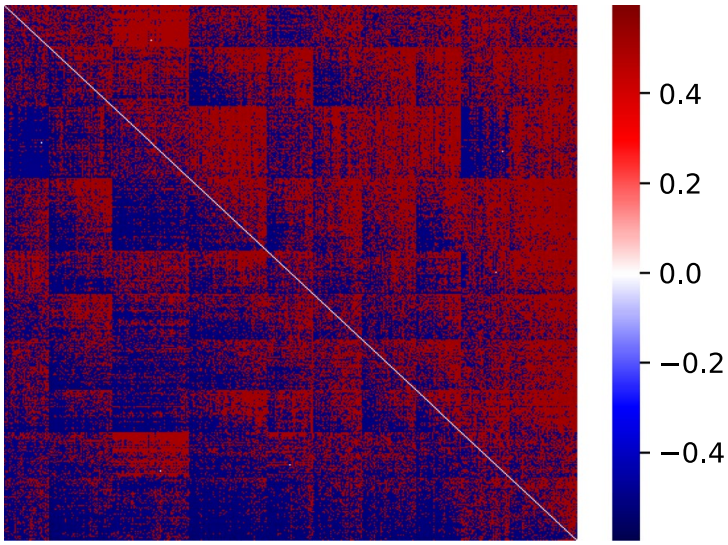


Fig. 8 Heatmap of the double-sorted lead–lag $p \times p$ matrix $A - A^T$. The rows and columns of the matrix index the $p = 434$ equities, and are categorised by cluster membership [labelled by the leadingness metric (4)]. Within each cluster, we sort the equities by their respective row-sum in $A - A^T$, a proxy for their individual leadingness

the horizontal axis. Similarly, negative values indicate that the horizontal axis stock leads the vertical axis one. The skew-symmetric matrix $A - A^T$ depicted in Fig. 8 is double-sorted by the leadingness metric (4) for each cluster and then, within each cluster, by the rowsum $\sum_{j=1}^{p} \left[A_{ij} - A_{ji}\right]$ of each equity i that is a member of the cluster. A block structure is apparent, with the last block being a highly lagging cluster.

5.2.2 Statistical significance testing for lead–lag clusters

We test whether there is a statistically significant time dependence in daily US equity returns using a permutation test on the spectrum of the Hermitian adjancency matrix $\tilde{A} = i(A - A^T)$. Under the null hypothesis that there is no time dependence, the ordering of the rows of the daily returns matrix $Y \in \mathbb{R}^{T \times p}$ is drawn uniformly at random from the set of all permutations on $\{1, \ldots, T\}$, $\sigma \in S_T$. Therefore, under the hypothesis of no time dependence, the spectrum of the observed lead–lag matrix should be consistent with the distribution over the spectra of matrices $\{\tilde{A}_\sigma\}_{\sigma \in S_T}$ computed using row-permuted returns matrices $Y_{\sigma(t),j}, t = 1, \ldots, T, j = 1, \ldots, p$. Since lead–lag cluster structure is associated with the largest eigenvalues of the Hermitian matrix $\tilde{A}$ (Cucuringu et al., 2020), our permutation test statistic is set to be the largest eigenvalue of $\tilde{A}$. We use 200 Monte Carlo samples from the null distribution. Under the null hypothesis, the Monte Carlo probability that the largest eigenvalue is greater than or equal to the observed largest eigenvalue is 1/201. We thus reject the null hypothesis with p-value $p < 0.005$, and conclude that there is significant temporal structure in US equity markets.

Note that a rejection of the null implies either

1. Significant auto-correlation
2. Significant cross-correlation
3. Some combination of 1. and 2.

It is not possible to resolve the identification issue between these three cases using our method. However, since our test statistic is a summary statistic of the lead–lag matrix spectrum, which encodes cross-correlations between time series and relates to the clustering structure (Cucuringu et al., 2020), a rejection of the null *suggests* that there is significant

Table 1 Number of equities in each SIC industry sector

Retail	90
Manufacturing	67
Construction	66
Mining	58
Trans., Util. & other	54
Fin., Ins. & RE	46
Wholesale	43
Services	9
Agri., Forest. & Fish.	1

cluster structure in the lead–lag matrix. Our statistically significant results when using our method for downstream prediction tasks (which relies solely on cross-equity prediction and not auto-correlation) in Sect. 6 provide further evidence for significant clustered lead–lag structure in the US equities.

5.2.3 Comparing data-driven clustering with known lead–lag mechanisms

We investigate whether our data-driven lead–lag extraction and clustering results can be explained by three potential mechanisms in the empirical finance lead–lag literature.

1. Sector membership induces clustered lead–lag effects. Biely and Thurner (2008) find associations between sector membership and lead–lag structure on the high-frequency scale of returns.
2. Equities with higher trading volume are hypothesised to lead lower volume equities. The disparities in trading volume across equities can lead to non-synchronous trading lead–lag effects (Chordia & Swaminathan, 2000; Campbell et al., 1997). Clustering structure may be induced by ordering equities based on quantiles of average trading volume.
3. Larger capitalisation equities are hypothesised to lead lower capitalisation equities (Lo & MacKinlay, 1990). This market capitalisation mechanism can produce lead–lag effects partly via non-trading effects and partly via other channels (Campbell et al., 1997). Conrad et al. (1991) also find that large stocks may lead small stocks via volatility spillovers. Clustering structure may be induced by ordering equities based on quantiles of market capitalisation.

Comparison of data-driven clustering with industry membership clustering

We compute the Jaccard similarity coefficient between the data-driven Hermitian RW clustering and the clustering due to industry membership. We use the first level of the Standard Industrial Classification (SIC) (Wharton Research Data Service, 2020) code for the firm corresponding to each equity in order to assign the equity to an industry. Table 1 counts the number of equities that are a member of each SIC sector. Most sectors have a relatively large number of equities, with *Agriculture, Forestry and Fisheries* and *Services* being quite small.

For comparison, the number of equities in each of the Hermitian RW clusters is shown in Table 2. The Hermitian RW algorithm leads to clusters of approximately equal size.

Table 2 Number of equities in each Hermitian RW cluster

0	1	2	3	4	5	6	7	8	9
37	49	57	58	35	35	42	34	32	55

Cluster ID is shown in the top row

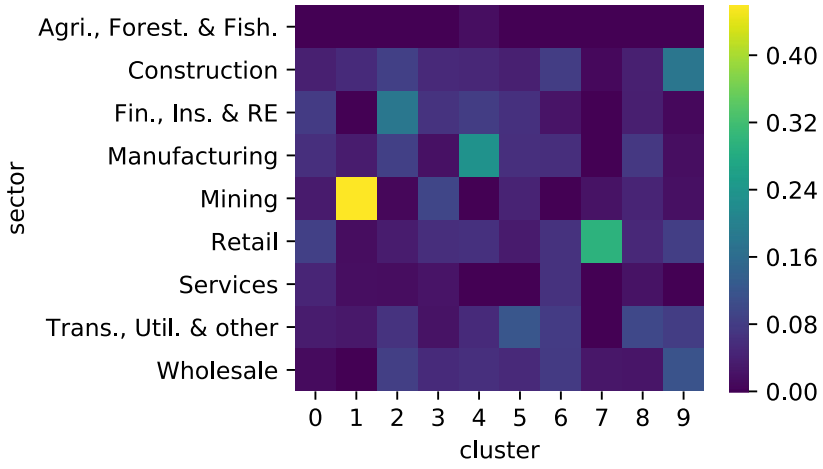


Fig. 9 The Jaccard similarity coefficient between the Hermitian RW clusters and industry clusters (SIC)

Figure 9 displays the Jaccard similarity between each pair of Hermitian RW and industry clusters. Overall, given the low values of the Jaccard similarity coefficients, the clustering seems to recover a structure that goes beyond simple industry sectors.

However, there does appear to be some association between certain SIC sectors and Hermitian RW clusters. We observe that the Mining sector seems to be strongly associated with cluster 1 (the second most leading cluster). The Finance, Insurance and Real Estate sector is also associated with a relatively leading cluster (cluster 2). These observations are consistent with the findings of Biely and Thurner (2008) that the finance and energy sectors have strong participation in the significant eigenvalues of the lead–lag matrix.[2] Xia et al. (2018) also find that the Financial and Real Estate sectors are associated with leading equities in the Chinese equity market. These associations between SIC code and Hermitian RW membership provide a partial interpretation for the links of the meta-flow network corresponding to the Hermitian RW clustering. The meta-flow network is depicted in Fig. 10. For example, we see that one of the strongest flows is from cluster 4 to 9—which are associated with Manufacturing and Construction respectively.

Figure 11 displays a histogram of the edge weights of two meta-flow networks: one corresponding to Hermitian RW clustering and the other corresponding to SIC clustering. We see that the distribution of meta-flow network edge weights obtained through the Hermitian RW clustering appears to be shifted to the right of the distribution of edge weights for the industry-based clustering. Since edge weights in the meta-flow network measure flow imbalance between pairs of clusters, this suggests that the data-driven Hermitian RW clustering results in larger flow between pairs of clusters than an

[2] While Biely and Thurner (2008) use GICS sector classification in their analysis, the GICS Energy sector has substantial overlap with the Mining SIC sector.

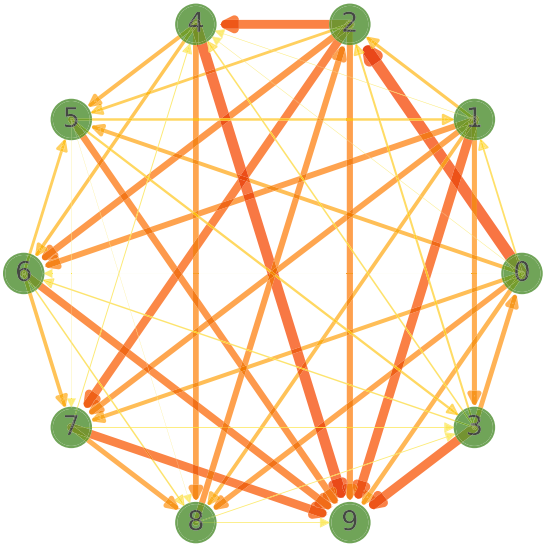


Fig. 10 Meta-flow network for Hermitian RW clusters; clusters are represented by nodes and larger edge weights are depicted by bolder colours and thicker lines

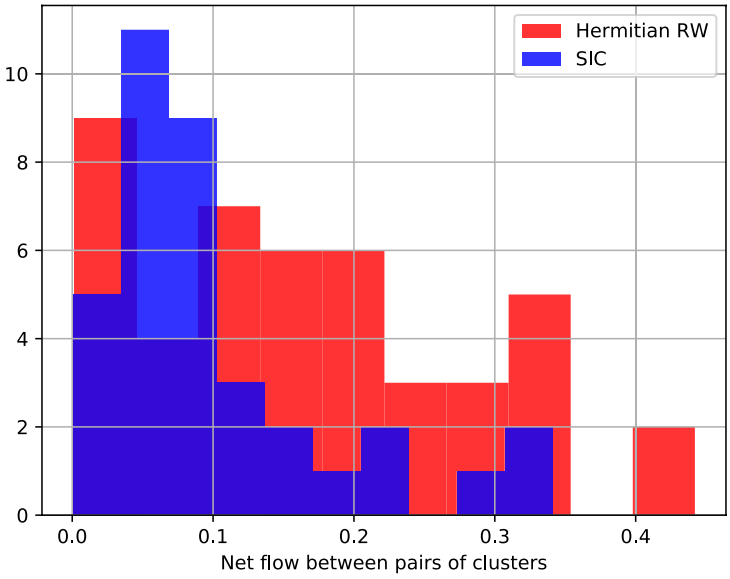


Fig. 11 Histogram of Hermitian RW and SIC clustering meta-flow network edge weights. The edge colours are layered in a semi-transparent fashion

industry-based clustering. This demonstrates the efficacy of our method in retrieving pairs of clusters with high flow imbalance.

Comparing data-driven clustering with market capitalisation and volume-based explanations

Figures 12 and 13 display the average daily dollar volume and market capitalisation averaged across all stocks in a given cluster. We observe that the leading clusters (clusters labelled 0–3) do not appear to have larger average daily dollar volume or market capitalisation.

In order to examine the association between the tendency for an equity to lead and its daily dollar volume or market capitalisation at a sub-cluster level, we compute the Spearman correlation between the row-sums of the lead–lag matrix—which provides a metric for the tendency of each cluster to lead—and these equity characteristics (trading

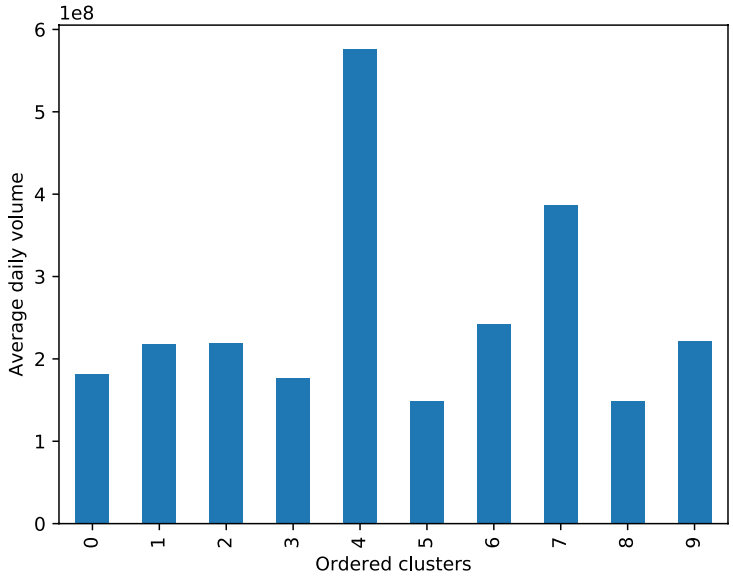


Fig. 12 Average daily dollar volume by Hermitian RW cluster

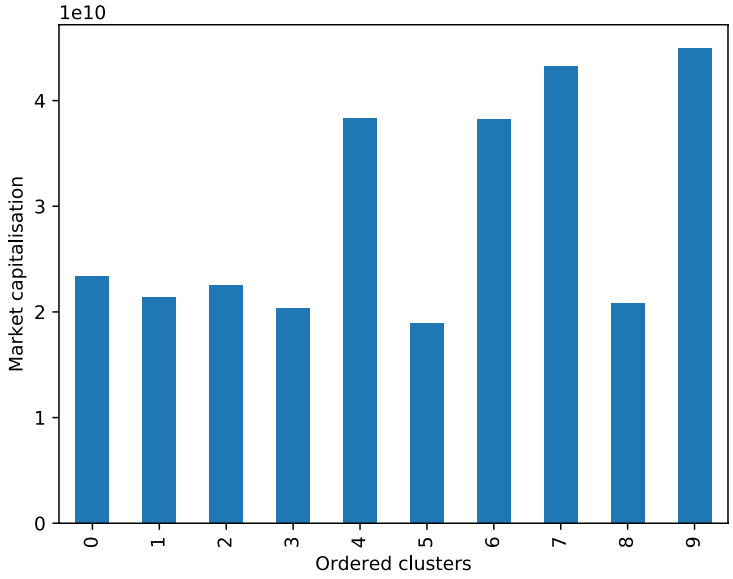


Fig. 13 Average market capitalisation by Hermitian RW cluster

volume and market capitalisation). This results in a Spearman correlation of 0.01 and −0.15 between the lead–lag row-sums and the equity trading volume and market capitalisation, respectively. These results are not consistent with a positive association between a cluster's tendency to lead and the trading volume or market capitalisation of its constituents.

Therefore, the results obtained by our data-driven clustering method cannot be explained by the three previously hypothesised mechanisms outlined in Sect. 5.2.3. Our proposed method may prove to be useful in the exploration of novel lead–lag mechanisms in the empirical finance community.

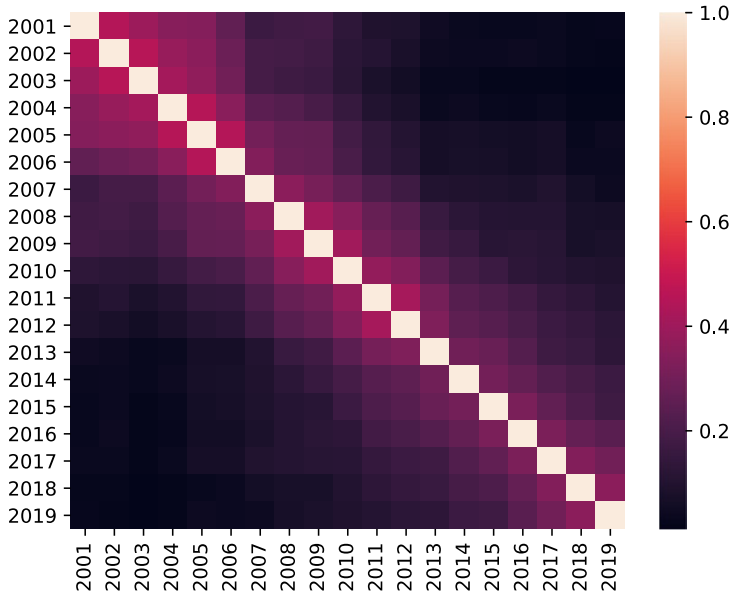


Fig. 14 Adjusted Rand index between clusters computed on yearly snapshots of data

5.3 Time-variation in clusters

To investigate the time-variation in the clustering obtained from our method, we recompute the clustering year-by-year using only data from the retrospective year to do so.[3] In order to compare the similarity in clusterings across time, we calculate the Adjusted Rand Index (ARI) between each pair of yearly clusterings. The results are illustrated in Fig. 14.

The relatively low ARI values between pairs of clusters indicates some—albeit low—persistence in year-to-year lead–lag structure. Biely and Thurner (2008) find that there is significant persistence in lead–lag structures across time. Xia et al. (2018) agree with our observations and find that the lead–lag phenomenon between two stocks is not constant but emerges during certain periods. They find that, on average, individual lead–lag relationships tend to last for around a year.

Further, we see that higher ARI values occur in earlier years: this suggests that there is a decrease in persistence between clusterings as time increases. Nevertheless, in Sect. 6, we show that there is sufficient persistence in the lead–lag cluster relationships in order for a dynamically updated clustering to be useful for forecasting purposes on a daily scale.

5.4 Limitations and implications of the empirical analysis

Our novel lead–lag extraction and clustering method yields clusters that cannot be explained by three previously considered mechanisms for lead–lag structure in US equity markets. Below, we discuss the limitations of our empirical analysis and its implications for understanding lead–lag structure in US equity markets.

First, a caveat of our empirical analysis is the instability of the lead–lag structure across time. In Sect. 5.3, we observe that the lead–lag structure does not exhibit high overall persistence. Since the lead–lag structure is not stable year-to-year, it is possible that the

[3] If an equity time series does not have sufficient data during a year then its missing entries in that year's lead–lag matrix are set to 0.

lead–lag results can be partially explained by the three mechanisms on a subset of the data. However, we have repeated our empirical clustering analysis on the relatively stable range[4] 2000–2006 and have found that the industry-based clustering is unable to fully explain the resulting data-driven clustering on this subset of the data. Furthermore, we have repeated the Spearman correlation analysis that was described in Sect. 5.2.3 using yearly snapshots of data. Appendix Figs. 27 and 28 display the Spearman correlation between an equity's tendency to be a leader (which is given by its lead–lag matrix row-sum) and its market capitalisation or trading volume. As these figures suggest, the association between an equity's tendency to be a leader and its market capitalisation or trading volume is not stable throughout time. There appear to be some periods when the sign of the association is consistent with the positive association predicted by the trading volume and market capitalisation lead–lag mechanisms. Nevertheless, the general sign and transience of the association across time does not support trading volume and market capitalisation as mechanisms which can explain the observed lead–lag structure.

A second caveat for the interpretation of our results concerns the relevancy of the market capitalisation mechanism. As explained in Sect. 5.1, we have restricted our attention to large capitalisation equities in order to avoid non-synchronous trading effects. Therefore, any interpretation of our empirical results must be conditioned by the large capitalisation of our equity universe. In particular, the market capitalisation mechanism may not be relevant under the condition that we restrict attention to the largest equities. In addition, previous papers that have found that smaller cap equities are able to lead larger cap equities if these smaller cap equities receive more news coverage (Scherbina & Schlusche, 2015). Thus, the hypothesised market capitalisation mechanism can be modulated by other information diffusion channels. This implies that the market capitalisation mechanism does not necessarily manifest itself in a positive association between market capitalisation and the tendency of an equity to be a leader.

Thirdly, the lead–lag literature contains other mechanisms that could potentially explain our results (Badrinath et al., 1995; Brennan et al., 1993; Menzly & Ozbas, 2010; Cohen & Frazzini, 2008). For example, cross-firm information flows through supplier networks have been hypothesised as lead–lag mechanisms (Menzly & Ozbas, 2010; Cohen & Frazzini, 2008). Testing these and other hypothesised mechanisms as sources for our observed lead–lag results remains further work.

Finally, given the novelty of our method and the fact that the resulting lead–lag structure cannot be explained through the three hypotheses that we have tested, our method may prove to be useful in the exploration of new mechanisms. The use of non-linear lead–lag metrics and effective algorithms for clustering directed networks (such as the distance correlation lead–lag metric and Hermitian RW algorithm) may illuminate lead–lag structures in US equity markets that cannot be explained by existing lead–lag mechanisms in the empirical finance literature.

6 Financial forecasting application

A difficulty in the modelling of high-dimensional systems is the identification of a suitable group of variables that can be used as predictors for other variables. This is related to the problem of variable selection in high-dimensional predictive modelling. On the one hand, the selection of too few conditioning variables can result in poor predictive power due to

[4] Cf relatively large values of ARI displayed during 2000–2006 in Fig. 14.

not capturing the temporal dependence between the response variable and relevant omitted variables. On the other hand, conditioning on too many variables can lead to the inclusion of many irrelevant variables; this dilutes the predictive power of the model (Runge et al., 2019).

In general, our unsupervised learning method can be used as a preliminary step to inform the choice, or design of, potential target and feature variables in a predictive model. This is achieved by using clusters with large net inflows (lagging clusters) to guide the choice of *target variables*, and clusters with large net outflows (leading clusters) guide the choice of *feature variables*. For example, in the latent variable synthetic data generating model presented in Sect. 4, the method identifies clusters of variables sharing the same lagged dependence on the latent variable z. By averaging the time series variables within each cluster, $\frac{1}{|C_i|}\sum_{j\in C_i} Y_t^j$, $\forall i \in \{1,\ldots,k\}$, the leading latent function $g_1(Z_t)$ at time t (the average of time series values in the most leading cluster) and the lagged latent functions $g_l(Z_{t-l})$, $l = 2,\ldots,k$ (the average of time series values in lagging clusters) can be recovered for each $t = 1,\ldots,T$ thanks to the reduction in observation noise resulting from the averaging procedure. By fitting models that capture the relations between the average value of the lagging clusters (target variable) and the average value of the leading cluster (feature variable), the latent variable dynamics can be captured, allowing the user to make predictions on the subsequent values of the lagging clusters. In Sect. 6.1, we illustrate the use of our lead–lag detection and clustering method for target and feature variable extraction in a financial forecasting application.

When a downstream model is built to capture the relationships between such target and feature variables, it is likely to exhibit stronger predictive power since our method has screened potential explanatory variables. Our method identifies predictable response variables and diminishes the risk of conditioning on irrelevant variables when used in downstream predictive modelling in high-dimensional time series systems. This approach to identifying groups of target and feature variables is useful for the application of returns forecasting in the US equity universe, since this is a highly noisy multivariate time series system where statistical lead–lag effect sizes are weak.[5]

We assess the predictive power of our lead–lag extraction and spectral clustering approach by evaluating the out-of-sample performance of a trading signal that was constructed using our method. The risk-adjusted returns of our trading signal will be evaluated using the Sharpe ratio. In order to test whether the signal's Sharpe ratio is significantly different to 0, we use a hypothesis test (Opdyke, 2007) that holds asymptotically under the general conditions of stationary and ergodic signal returns.

Our approach to quantifying the predictive performance of our method by studying the risk-adjusted performance of a portfolio constructed using our method is common in the quantitative finance literature (Asness et al., 2013). The task of constructing a statistically significant trading signal using only publicly available price data in a highly liquid market such as the US equity market is a challenging task due to the informational efficiency of such markets (Malkiel & Fama, 1970). The weak-form of the Efficient Markets Hypothesis states (Malkiel & Fama, 1970) that markets fully reflect all historical price data; this implies that it is not possible to make economic profits in excess of market equilibrium profits by trading on the basis of such historical price data. The number of empirical studies (Malkiel & Fama, 1970) in strong support of the weak-form of the Efficient Markets

[5] due to the Efficient Markets Hypothesis (Malkiel & Fama, 1970).

Hypothesis underlines the informational efficiency of US equity markets and hence the challenge of constructing a statistically significantly profitable trading signal.

Similarly, Curme et al. (2015b) argue for the use of lead–lag networks to guide variable selection for downstream financial forecasting tasks. However, our results are stronger as we test the performance of our lead–lag network method for variable subset selection in a rolling out-of-sample evaluation.

6.1 Signal construction

We keep the trading signal relatively simple in order to effectively assess the predictive performance of the underlying signal derived from our lead–lag extraction and clustering methodology. Our trading signal forecasts lagging cluster returns using smoothed leading cluster returns. In order to evaluate the out-of-sample performance of our method, we compute the clustering $C_1, \ldots, C_k$ and flow graph F on a rolling basis using a 2-month update period and yearly look-back window. Further, using the same update frequency and yearly look-back window, we fit a separate linear model for each pair of clusters. In particular, for each ordered pair of clusters $i, j \in \{1, \ldots, k\}$, we fit a linear model to forecast the mean daily return for lagging cluster j

$$Y_t^{(j)} = \frac{1}{|C_j|} \sum_{n \in C_j} Y_t^{(n)},$$

using an exponentially weighted moving average of the mean returns for cluster i as the covariate (input variable to the linear regression)

$$X_t^{(i)} = \frac{1}{|C_i|} \sum_{n \in C_i} \sum_{l=1}^{t} (1-\alpha)^{l-1} Y_{t-l}^{(n)}.$$

The choice of exponential parameter $\alpha = 0.4$ assigns 92% of the total weight of the exponential sum $\sum_{l=1}^{\infty} (1-\alpha)^{l-1}$ to the first 5 lags $l = 1, \ldots, 5$. Thus, the exponential moving average mainly captures lead–lag effects on the scale of approximately up to 1 week, while emphasising higher-frequency daily lead–lag effects. The coefficient θ_{ij} of the linear model $Y^{(j)} = \theta_{ij} X_t^i$ is fitted using ordinary least squares.[6]

For every day $t = 1, \ldots, T$, we compute the predictive signal from cluster i to j for each ordered pair $i, j \in \{1, \ldots, k\}$ of clusters

$$\hat{Y}_t^{(j)} = \theta_{ij} X_t^{(i)}.$$

These predictive signals are aggregated using a thresholded flow graph $\tilde{F}$ where $\tilde{F}_{ij} = \mathbb{1}\{F_{ij} > c\}$ where c is the 90% quantile of the edge weights of the flow graph F. Thus, the flow graph ensures that only the cluster-to-cluster relationships that have shown the

[6] Note that unbiasedness and consistency do not hold in general for this ordinary least squares estimation due to network effects within the residual error structure.

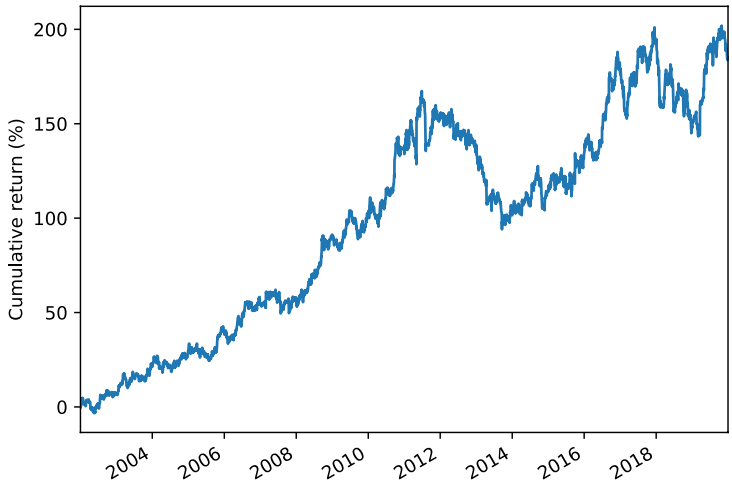


Fig. 15 Cumulative return for the financial forecasting signal; the signal is scaled to target a 10% yearly volatility

greatest historical flow are included in the construction of the signal. Mathematically, the predictive signal S_t for cluster $j \in \{1, \ldots, k\}$ is given by

$$S_t(j) = \text{sign}\left(\sum_{n=1}^{k} \tilde{F}_{ij} \hat{Y}_t^{(j)} \right).$$

The signal for a specific equity $m \in \{1, \ldots, p\}$ on day t is set to be the signal for its cluster C_m i.e. $S_t(C_m)$.

Finally, the signals for each equity are normalised by a 21-day historical rolling estimator of the overall signal's volatility. This rolling normalisation ensures that the overall position size is dynamically adjusted to target a constant 10% annual volatility. Assuming that the Sharpe ratio of our signal is constant throughout time, this procedure can be seen as targeting an optimal Kelly criterion (Thorp, 2011) for the signal on a rolling basis. Further, volatility normalisation tends to bring our daily trading returns closer to stationarity while decreasing their absolute skew and kurtosis; this makes the analysis of our trading returns more reliable.

6.2 Evaluation metric

We evaluate the performance of the signal constructed using our method by its risk-adjusted return. Since the return of the signal on equity m at time t is given by $S_t(C_m) \cdot Y_t^{(m)}$, the total return of the signal at time t is given by

$$\sum_{m=1}^{p} S_t(C_m) \cdot Y_t^{(m)}$$

The metric that we use for the risk-adjusted return is the Sharpe ratio (Campbell et al., 1997) of the total signal return.

6.3 Results

The cumulative total return of the signal across time is displayed in Fig. 15.

The trading signal results in an annualised Sharpe ratio of 0.62 with a corresponding significant one-sided p-value of $p < 0.004$ (Opdyke, 2007). We compare this with the

Sharpe ratio of 0.40 for the S &P500 market return on the same period. Further, the trading signal exhibits a low correlation (0.04) with the market return. This suggests that the trading signal cannot be explained by market equilibrium returns. The mean daily return of the trading signal is 2.4 basis points.[7]

We observe in Fig. 15 that there is a decay in the performance of the signal after 2012; this can be compared with the reduction in clustering persistence observed after 2012 in Fig. 14, and with the observation in the work of Curme et al. (2015a) that the informational efficiency of the market appears to increase in 2012 relative to earlier years.

Ablation study

We conduct an ablation study in order to test the importance of the lead–lag clustering structure on the observed performance of the trading signal. Specifically, under the null hypothesis that there is no lead–lag cluster structure in US equity returns, the clustering for the US equities is drawn uniformly at random from the set of permutations on cluster labels. Therefore, under the hypothesis of lead–lag cluster structure, the Sharpe ratio of the trading signal described in Sect. 6.1 should be consistent with the distribution over the Sharpe ratios of trading signals that are computed with permuted cluster labels. We use 200 Monte Carlo samples from the null distribution that computes the Sharpe ratio of the same trading signal pipeline described in Sect. 6.1 but with any clustering in this pipeline drawn uniformly at random from S_p. Under the null hypothesis, the Monte Carlo probability that the Sharpe ratio is greater than or equal to the observed Sharpe ratio of 0.62 is 1/201. We thus reject the null hypothesis with p-value $p < 0.005$, and conclude that the lead–lag cluster structure is significant in the construction of the predictive trading signal.

For comparison, we also implement a LASSO-VAR model (Friedman et al., 2010) that fits a multivariate linear model for each equity's next-day return using the 5 previous lagged returns across all equities. Rolling validation and volatility normalisation was performed as described in Sect. 6.1. Specifically, we use an update period of 2 months and a yearly look-back window to fit the LASSO-VAR model on a rolling basis. At each update period, the L1 regularisation hyperparameter is selected using 5-fold cross validation on the rolling year's worth of data across the grid of values $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. The signals for each equity are normalised by a 21-day historical rolling estimator of the overall signal's volatility. The LASSO-VAR signal yields a Sharpe ratio of 0.27 with a one-sided p-value $p > 0.13$; the relatively poor performance of the LASSO-VAR baseline model highlights the informational efficiency of the US equity market as well as the difficulty of variable selection in equity returns forecasting problems. Further, the signal constructed using our cluster-based method has a low correlation of -0.005 with the LASSO-VAR signal indicating that the predictive signals of our method cannot be captured using a simple baseline LASSO-VAR model.

A caveat to our results is that we do not take into account transaction costs when calculating the profit of our signal. These may be significant in practice given the basis point size of the average daily returns. On the other hand, the turnover of the trading signal, which is based on a weekly smoothing of lagged returns, is relatively low. Regardless of the economic significance of the signal, it is clear that the clustered lead–lag structure is statistically strong enough to be used as a predictive signal for equity returns.

[7] Cf a mean daily market return of 3.0 basis points.

7 Conclusion

We propose a methodology for the problem of data-driven detection of leading and lagging clusters of time series. Our unsupervised learning method can capture general, non-linear lead–lag correlations and leverages a state-of-the-art directed network clustering algorithm which is able to detect clusters with high flow imbalance. When applied to US equity data, our method produces a clustering that is statistically significant but that cannot be explained by three prominent lead–lag hypotheses in the empirical finance literature; this suggests that our methodology is a useful tool for the exploration of novel lead–lag mechanisms in the discipline of empirical finance. Furthermore, we find that our method can be employed for challenging downstream forecasting tasks in noisy, high-dimensional settings. In particular, we show how our method can be used for the construction of a statistically significant, parsimonious trading signal in the US equity market.

In addition to the financial domain, the applicability of our proposed methodology extends to other areas—such as economics, medicine and earth sciences—that are characterised by large multivariate time series data which exhibit a latent lead–lag structure. Finally, our network approach to time series, which is able to infer global clustering structure based on local pairwise interactions, can be applied to general pairwise directed interaction data between time series variables. Thus, our framework may be generalised beyond *lead–lag* interactions, in order to discover cluster structure in high-dimensional time-series systems based on *general* directed interactions.

Appendices

A Additional numerical experiments

A.1 Synthetic data experiment: lead–lag results

Figures 16 and 17 display the lead–lag metric classification accuracy for the Legendre (8) and Hermite (9) synthetic data generating settings, respectively. We observe that the **ccf-auc** method with the distance correlation performs best in these non-linear settings.

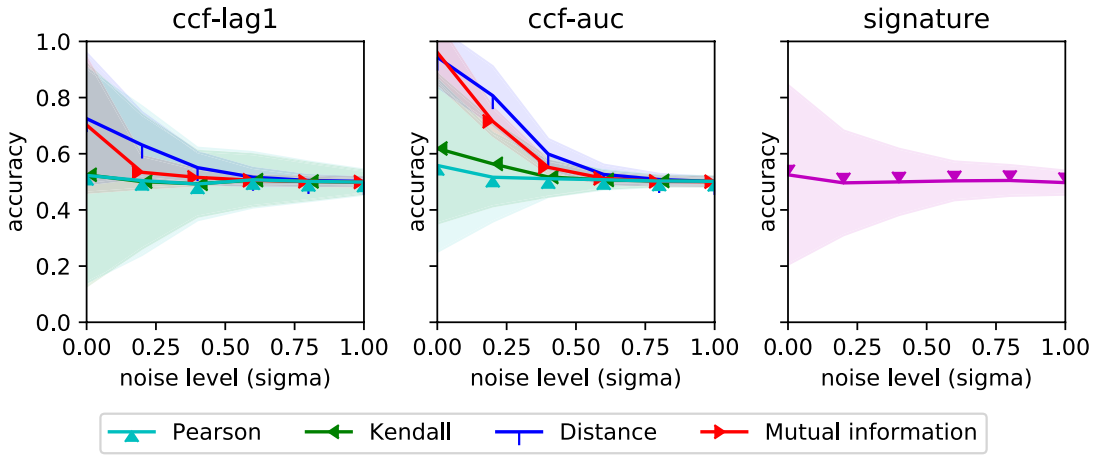


Fig. 16 Average and confidence interval for classification accuracy by lead–lag detection method in the Legendre setting (8)

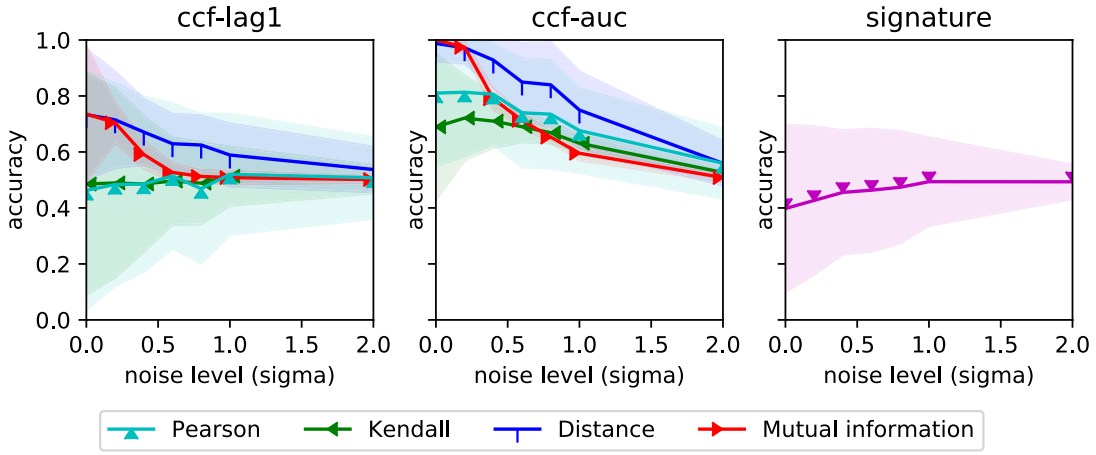


Fig. 17 Average and confidence interval for classification accuracy by lead–lag detection method in the Hermite setting (9)

A.2 Synthetic data experiment: clustering results

Figures 18, 19 and 20 display the ARI of our pipeline in the Legendre (8), Hermite (9) and Heterogeneous (10) synthetic data generating settings, respectively. The pipeline performs best on average using the Hermitian RW clustering component in these settings.

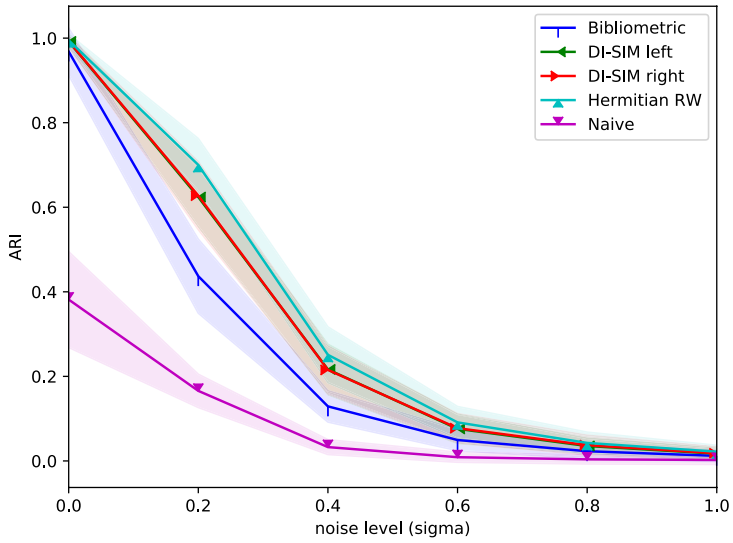


Fig. 18 Average and confidence interval for the ARI by clustering method in the Legendre setting

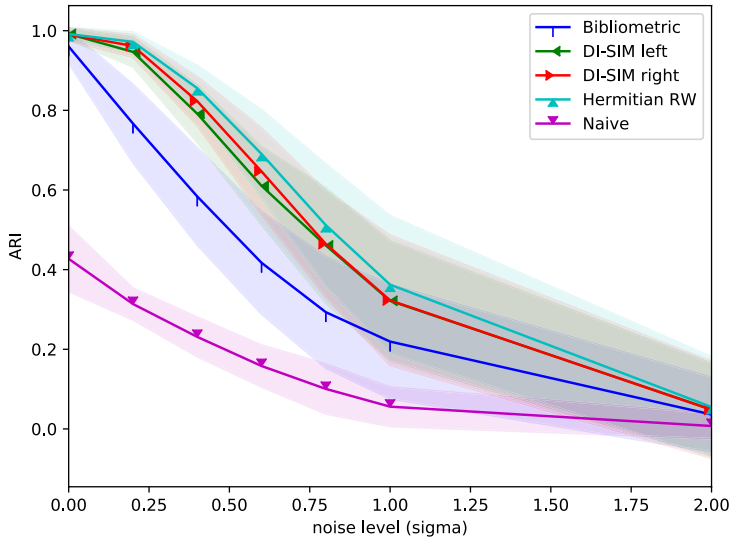


Fig. 19 Average and confidence interval for the ARI by clustering method in the Hermite setting

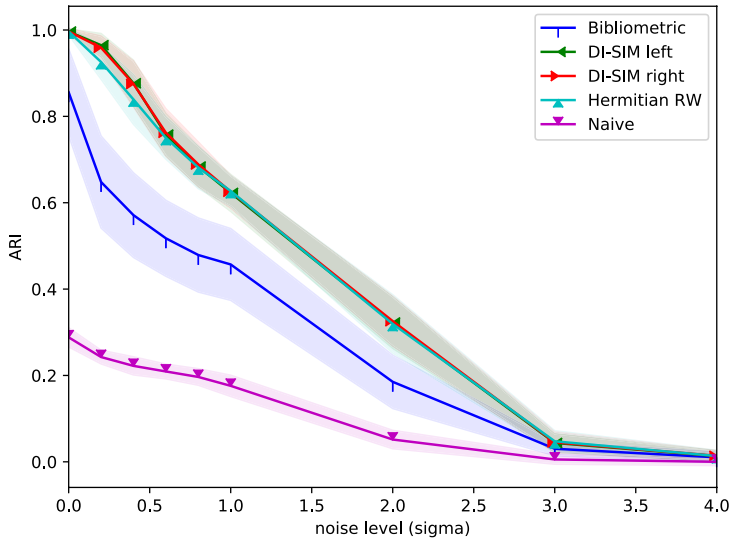


Fig. 20 Average and confidence interval for the ARI by clustering method in the heterogeneous setting

A.3 Synthetic data experiment: interaction of lead–lag and clustering components

Figures 21, 22 and 23 display the ARI of the pipeline for each choice of lead–lag extraction and clustering components.

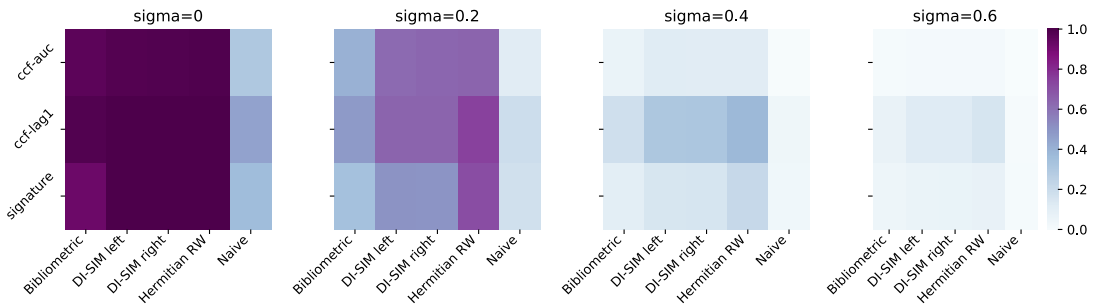


Fig. 21 Average ARI by lead–lag and clustering component in the Legendre setting

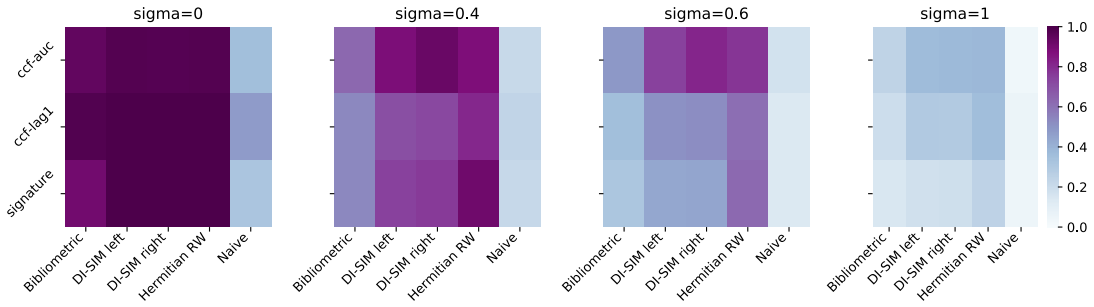


Fig. 22 Average ARI by lead–lag and clustering method in the Hermite setting

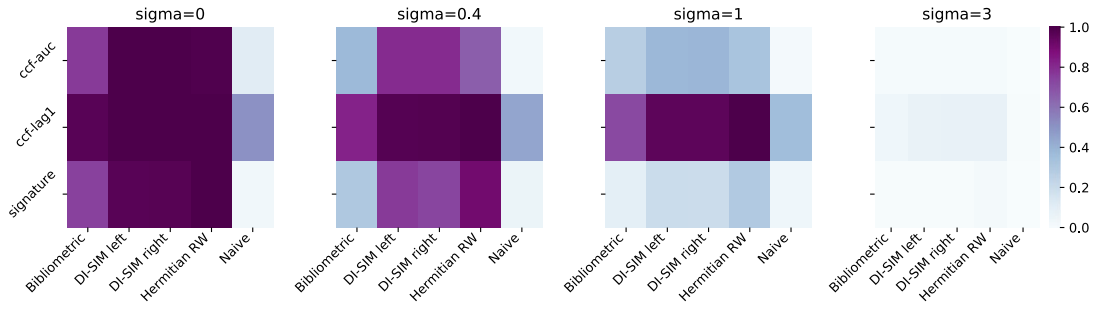


Fig. 23 Average ARI by lead–lag and clustering method in the heterogeneous setting

A.4 Synthetic data ablation study: varying the hyperparameter corresponding to the number of clusters

In Figs. 24, 25 and 26 we display the average and confidence interval for the ARI across different hyperparameter levels for the number of clusters used in the clustering component

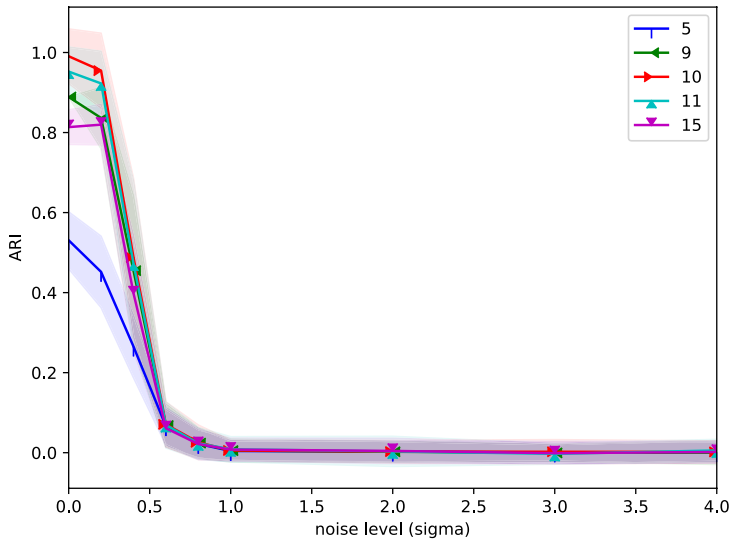


Fig. 24 Average and confidence interval for the ARI by different levels of the hyperparameter corresponding to the number of clusters in the cosine setting

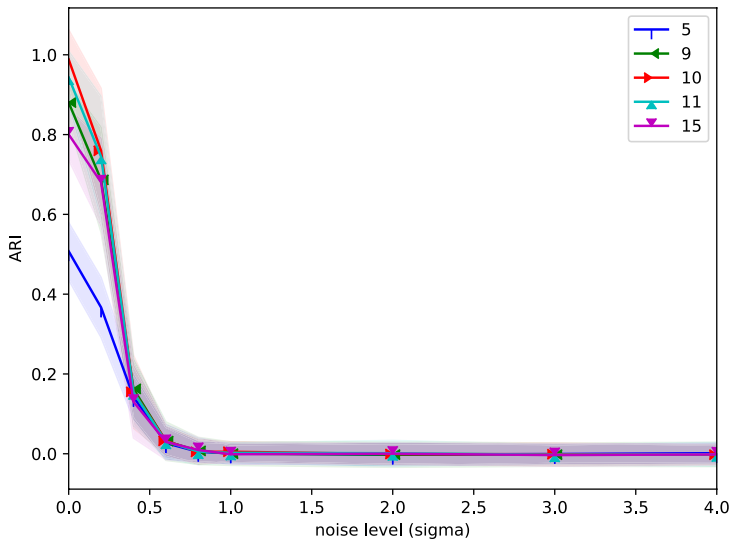


Fig. 25 Average and confidence interval for the ARI by different levels of the hyperparameter corresponding to the number of clusters in the Legendre setting

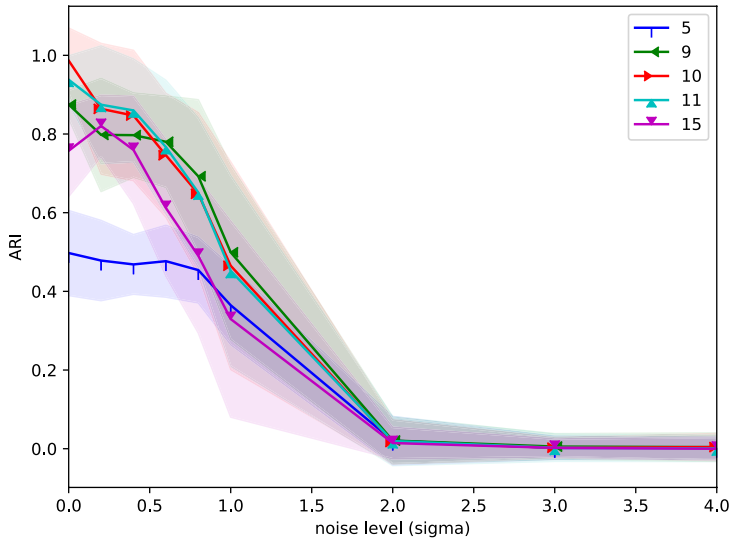


Fig. 26 Average and confidence interval for the ARI by different levels of the hyperparameter corresponding to the number of clusters in the Hermite setting

of the pipeline.

A.5 Real data experiment: time-variation in results

Figures 27 and 28 display the temporal variation in Spearman correlation between the US equity lead–lag matrix row-sums and a given characteristic (*average daily trading volume* in Fig. 27 and *market capitalisation* in Fig. 28) of each equity. We observe that there is

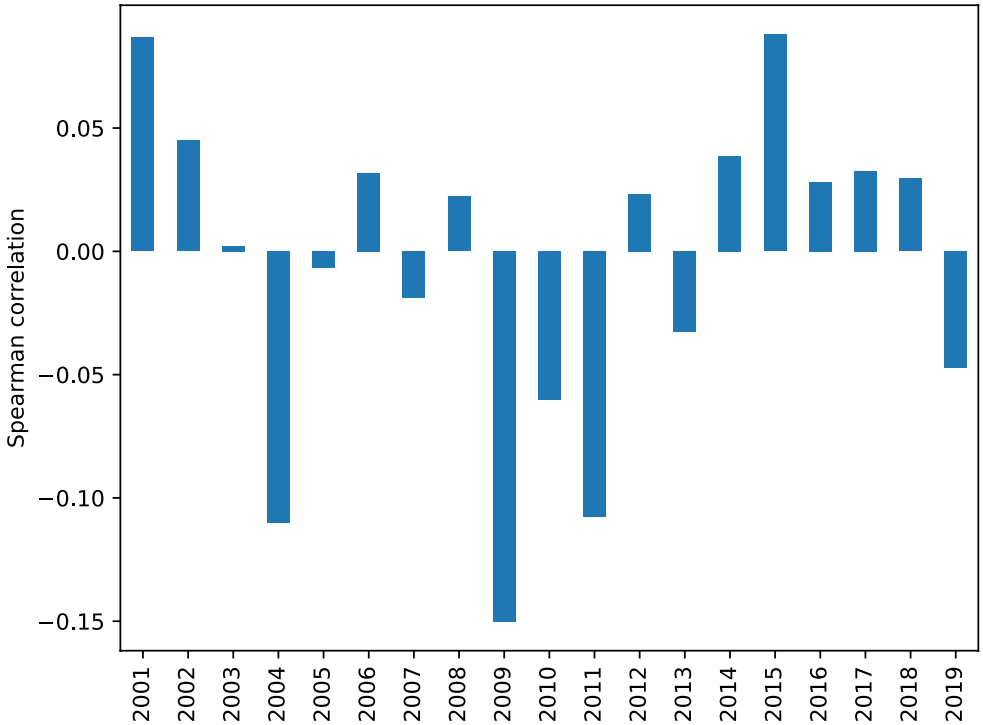


Fig. 27 Spearman correlation between the lead–lag matrix row-sums and *average daily trading volume* for each equity, using yearly snapshots of data

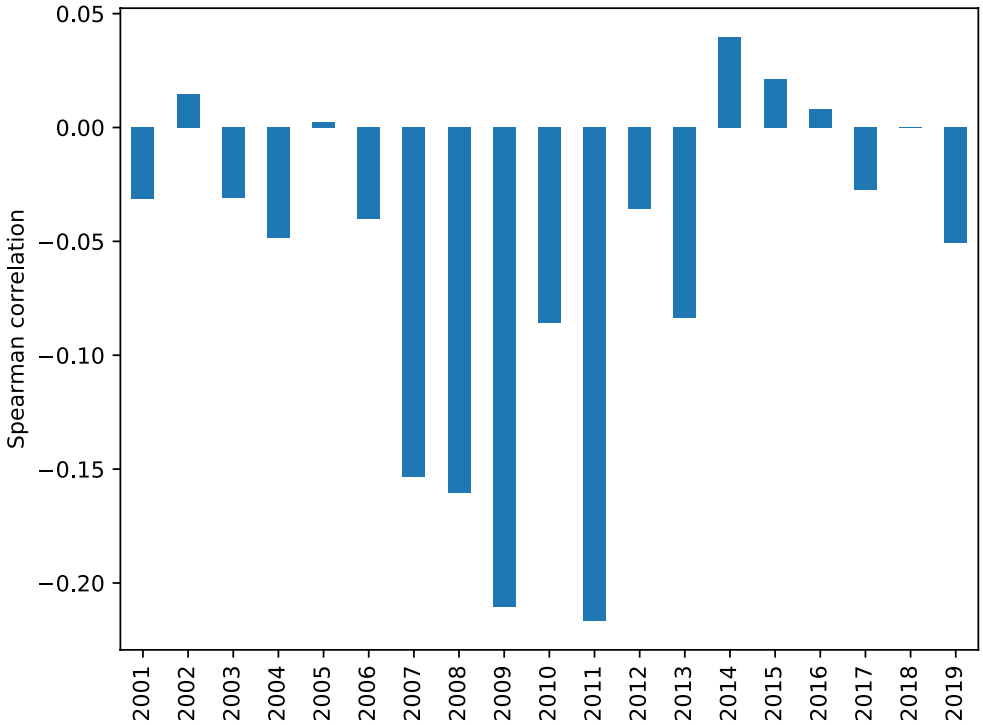


Fig. 28 Spearman correlation between lead–lag matrix row-sums and *market capitalisation* for each equity, using yearly snapshots of data

substantial temporal variation in each equity's tendency to be a leader (as measured by its lead–lag matrix row-sum) and its underlying characteristic.

Acknowledgements The authors would like to acknowledge the anonymous referees for their helpful feedback.

Author contributions All authors made substantial contributions to the work.

Funding SB is supported by the EPSRC CDT in Modern Statistics and Statistical Machine Learning (EP/S023151/1) and The Alan Turing Institute's Finance and Economics Programme. GR is funded in part by EPSRC Grants EP/T018445/1 and EP/R018472/1. All authors acknowledge support from the EPSRC Grant EP/N510129/1 at The Alan Turing Institute.

Data availability All data is available through a WRDS subscription https://wrds-www.wharton.upenn.edu/.

Code availability (software application or custom code) All code is available at https://github.com/stefanosbennett/mlj-lead-lag.

Declarations

Conflict of interest The authors have no conflicts of interest to declare.

Ethics approval The study did not involve human or animal participants.

Consent to participate The study did not involve human or animal participants.

Consent for publication The study did not involve sensitive or confidential data.

References

Asness, C. S., Moskowitz, T. J., & Pedersen, L. H. (2013). Value and momentum everywhere. *Journal of Finance, 68*(3), 929–985. https://doi.org/10.1111/jofi.12021

Badrinath, S. G., Jayant, R. K., & Thomas, H. N. (1995). Of Shepards, Sheep and the cross-autocorrelations in equity returns. *The Review of Financial Studies, 8*(2), 401.

Basnarkov, L., Stojkoski, V., Utkovski, Z., & Kocarev, L. (2019). Lead–lag relationships in foreign exchange markets. arXiv https://doi.org/10.1016/j.physa.2019.122986, arXiv:1906.10388

Batson, J., Spielman, D. A., Srivastava, N., & Teng, S. H. (2013). Spectral sparsification of graphs: Theory and algorithms. *Communications of the ACM, 56*(8), 87–94. https://doi.org/10.1145/2492007.2492029

Biely, C., & Thurner, S. (2008). Random matrix ensembles of time-lagged correlation matrices: Derivation of eigenvalue spectra and analysis of financial time-series. *Quantitative Finance, 8*(7), 705–722. https://doi.org/10.1080/14697680701691477arxiv:abs/0609053 [physics].

Billio, M., Getmansky, M., Lo, A. W., & Pelizzon, L. (2012). Econometric measures of connectedness and systemic risk in the finance and insurance sectors. *Journal of Financial Economics, 104*(3), 535–559. https://doi.org/10.1016/j.jfineco.2011.12.010

Bradley, R. A., & Terry, M. E. (1952). Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika, 39*, 324–345.

Brennan, M. J., Narasimhan, J., & Swaminathan, B. (1993). Investment analysis and the adjustment of stock prices to common information source. *The Review of Financial Studies, 6*(4), 799–824.

Camilleri, S. J., Scicluna, N., & Bai, Y. (2019). Do stock markets lead or lag macroeconomic variables? Evidence from select European countries. *The North American Journal of Economics and Finance, 48*, 170–186. https://doi.org/10.1016/j.najef.2019.01.019

Campbell, J. Y., Lo, A. W., & MacKinlay, A. C. (1997). *The econometrics of financial markets*. Princeton University Press. https://doi.org/10.1515/9781400830213-004

Chau, S. L., Cucuringu, M., & Sejdinovic, D. (2020). Spectral ranking with covariates. arXiv preprint arXiv:2005.04035

Chevyrev, I., & Kormilitzin, A. (2016). A primer on the signature method in machine learning. arXiv arXiv:1603.03788v1

Chordia, T., & Swaminathan, B. (2000). Trading volume and cross-autocorrelations in stock returns. *The Journal of Finance, LV*(2), 913–935.

Cohen, L., & Frazzini, A. (2008). Economic links and predictable returns. *Journal of Finance, 63*(4), 1977–2011. https://doi.org/10.1111/j.1540-6261.2008.01379.x

Conrad, J., Gultekin, M., & Kaul, G. (1991). Asymmetric predictability of conditional variances. *The Review of Financial Studies, 4*(4), 597–622.

Cont, R. (2001). Empirical properties of asset returns: Stylized facts and statistical issues. *Quantitative Finance, 1*(2), 223–236. https://doi.org/10.1080/713665670

Cucuringu, M. (2016). Sync-Rank: Robust ranking, constrained ranking and rank aggregation via eigenvector and semidefinite programming synchronization. *IEEE Transactions on Network Science and Engineering, 3*(1), 58–79.

Cucuringu, M., Li, H., Sun, H., & Zanetti, L. (2020). Hermitian matrices for clustering directed graphs: Insights and applications. AISTATS pp 1–19. arXiv:1908.02096

Curme, C., Tumminello, M., Mantegna, R. N., Stanley, H. E., & Kenett, D. Y. (2015a). Emergence of statistically validated financial intraday lead–lag relationships. *Quantitative Finance, 15*(8), 1375–1386. https://doi.org/10.1080/14697688.2015.1032545arXiv:1401.0462

Curme, C., Tumminello, M., Mantegna, R. N., & Stanley, H. E. (2015b). *How lead–lag correlations affect the intraday pattern of collective stock dynamics*. Office of Financial Research Working Paper Series https://doi.org/10.2139/ssrn.2648490

d'Aspremont, A., Cucuringu, M., & Tyagi, H. (2021). Ranking and synchronization from pairwise measurements via SVD. *Journal of Machine Learning Research, 22*(19), 1–63.

De Bacco, C., Larremore, D. B., & Moore, C. (2018). A physical model for efficient ranking in networks. *Science Advances, 4*(7), 1–10.
Dugué, N., & Perez, A. (2015). Directed Louvain: Maximizing modularity in directed networks. In *HAL archives ouvertes* (pp. 0–14). https://hal.archives-ouvertes.fr/hal-01231784
Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics, 33*(1), 3–56. https://doi.org/10.2469/dig.v36.n3.4225
Farrell, J. (1974). Analyzing covariation of returns to determine homogeneous stock groupings. *Journal of Business, 47*(2), 186–207.
Fiedor, P. (2014). Information-theoretic approach to lead–lag effect on financial markets. *European Physical Journal B*. https://doi.org/10.1140/epjb/e2014-50108-3arXiv:1402.3820
Fogel, F., d'Aspremont, A., & Vojnovic, M. (2016). Spectral ranking using seriation. *Journal of Machine Learning Research, 17*(88), 1–45.
Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software, 33*(1), 1–22. arxiv:0908.3817
Gates, A. J., & Ahn, Y. Y. (2017). The impact of random models on clustering similarity. *Journal of Machine Learning Research, 18*(87), 1–28.
Gleich, D. F., & Lim, L. H. (2011). Rank aggregation via nuclear norm minimization. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM* (pp. 60–68).
Google. (2012). The PageRank citation ranking: Bringing order to the web January. In *Proceedings of the 2012 IEEE international symposium on workload characterization. IISWC* (Vol. 2012, pp. 111–112). https://doi.org/10.1109/IISWC.2012.6402911
Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., & Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research, 13*, 723–773.
Gyurkó, L. G., Lyons, T., Kontkowski, M., & Field, J. (2014). Extracting information from the signature of a financial data stream (pp 1–22). arXiv arXiv:1307.7244
Harzallah, A., & Sadourny, R. (1997). Observed lead–lag relationships between Indian summer monsoon and some meteorological variables. *Climate Dynamics, 13*(9), 635–648. https://doi.org/10.1007/s0038 20050187
He, Y., Reinert, G., & Cucuringu, M. (2021). Digrac: Digraph clustering with flow imbalance. arXiv arxiv: 2106.05194 [stat.ML]
Hu, P., & Lau, W. C. (2013). A survey and taxonomy of graph sampling. arxiv:1308.5865
Huber, P. J. (1962). Pairwise comparison and ranking: Optimum properties of the row sum procedure. *The Annals of Mathematical Statistics, 34*, 511.
Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of classification, 2*(1), 193–218.
Huth, N. (2012). High frequency lead/lag relationships empirical facts. *Journal of Empirical Finance, 26*(March 2014), 41–58.
Iyetomi, H., Aoyama, H., Fujiwara, Y., Souma, W., Vodenska, I., & Yoshikawa, H. (2020). Relationship between macroeconomic indicators and economic cycles in US. *Scientific Reports, 10*(1), 1–12.
Janzing, D., Balduzzi, D., Grosse-Wentrup, M., & Schölkopf, B. (2013). Quantifying causal influences. *Annals of Statistics, 41*(5), 2324–2358. https://doi.org/10.1214/13-AOS1145arXiv:1203.6502
Jegadeesh, N., & Titman, S. (1995). Overreaction, delayed reaction, and contrarian profits. *The Review of Financial Studies, 8*(4), 973–993.
Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika, 30*(1), 81–93.
Laenen, S., & Sun, H. (2020). Higher-order spectral clustering of directed graphs. In *Advances in neural information processing systems 2020 (NeurIPS)*. arXiv:2011.05080
Levin, D., Lyons, T., & Ni, H. (2016). Learning from the past, predicting the statistics for the future, learning an evolving system (pp 1–40). arXiv arXiv:1309.0260
Liao, C., Huang, Y., Shi, X., & Jin, X. (2014). Mining influence in evolving entities: A study on stock market. In: *DSAA 2014—Proceedings of the 2014 IEEE international conference on data science and advanced analytics* (pp. 244–250). https://doi.org/10.1109/DSAA.2014.7058080
Lin, Z., Ding, W., Yan, G., Yu, C., & Giua, A. (2013). Leader–follower formation via complex Laplacian. *Automatica, 49*, 1900–1906.
Lo, A. W., & MacKinlay, A. C. (1990). When are contrarian profits due to stock market overreaction. *The Review of Financial Studies, 3*(2), 175–205.
Malkiel, B. G., & Fama, E. (1970). Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*. https://doi.org/10.2307/2325488
Marti, G., Andler, S., & Nielsen, F., & Donnat, P. (2016). Exploring and measuring non-linear correlations: Copulas, lightspeed transportation and clustering. arXiv arXiv:1610.09659

Marti, G., Nielsen, F., & Bińkowski, M., & Donnat, P. (2019). A review of two decades of correlations, hierarchies, networks and clustering in financial markets (pp. 1–34). arXiv arXiv:1703.00485

Menzly, L., & Ozbas, O. (2010). Market segmentation and cross-predictability of returns. *Journal of Finance, 65*(4), 1555–1580. https://doi.org/10.1111/j.1540-6261.2010.01578.x

Namaki, A., Shirazi, A. H., Raei, R., & Jafari, G. R. (2011). Network analysis of a financial market based on genuine correlation and threshold method. *Physica A: Statistical Mechanics and its Applications, 390*(21–22), 3835–3841. https://doi.org/10.1016/j.physa.2011.06.033

Newman, M. (2018). *Networks* (2nd ed.). Oxford University Press.

Opdyke, J. D. (2007). Comparing sharpe ratios: So where are the *p*-values? *Journal of Asset Management, 8*(5), 308–336. https://doi.org/10.1057/palgrave.jam.2250084

Page, L., Brin, S. Motwani, R., & Winograd, T. (1998). The PageRank citation ranking: Bringing order to the Web. In *Proceedings of the 7th international world wide web conference* (pp 161–172).

Pentney, W., & Meila, M. (2005). Spectral clustering of biological sequence data. *Proceedings of the National Conference on Artificial Intelligence, 2*, 845–850.

Podobnik, B., Wang, D., Horvatic, D., Grosse, I., & Stanley, H. (2010). Time-lag cross-correlations in collective phenomena. *EPL, 90*, 68001. https://doi.org/10.1209/0295-5075/90/68001

Reizenstein, J. & Graham, B. (2018). The iisignature library: Efficient calculation of iterated-integral signatures and log signatures (pp. 1–18). arXiv:1802.08252

Reshef, D. N., Reshef, Y. A., Finucane, H. K., Grossman, S. R., McVean, G., Turnbaugh, P. J., Lander, E. S., Mitzenmacher, M., & Sabeti, P. C. (2011). Detecting novel associations in large datasets. *Science, 334*(6062), 1518–1524. https://doi.org/10.1126/science.1205438.Detecting

Rohe, K., Qin, T., & Yu, B. (2016). Co-clustering directed graphs to discover asymmetries and directional communities. *Proceedings of the National Academy of Sciences of the United States of America, 113*(45), 12679–12684. https://doi.org/10.1073/pnas.1525793113

Runge, J., Nowack, P., Kretschmer, M., Flaxman, S., & Sejdinovic, D. (2019). Detecting causal associations in large nonlinear time series datasets. *Science Advances, 5*(11), 1–15.

Sandoval, L. (2014). Structure of a Global Network of financial companies based on transfer entropy. *Entropy, 16*(8), 4443–4482. https://doi.org/10.3390/e16084443

Sandoval, L., & Franca, I. D. P. (2012). Correlation of financial markets in times of crisis. *Physica A: Statistical Mechanics and its Applications, 391*(1–2), 187–208. https://doi.org/10.1016/j.physa.2011.07.023arXiv:1102.1339

Satuluri, V., & Parthasarathy, S. (2011). Symmetrizations for clustering directed graphs. In *ACM international conference proceeding series* (pp 343–354). https://doi.org/10.1145/1951365.1951407

Scherbina, A. D., & Schlusche, B. (2015). Cross-firm information flows and the predictability of stock returns. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.2263033

Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/ICIP.2014.7025680

Shojaie, A., & Fox, E. B. (2021). Granger causality: A review and recent advances. arXiv arXiv:2105.02675

Skoura, A. (2019). Detection of lead–lag relationships using both time domain and time-frequency domain; An application to wealth-to-income ratio. *Economies, 7*(2), 28–60. https://doi.org/10.3390/economies7020028

Sornette, D., & Zhou, W. X. (2005). Non-parametric determination of real-time lag structure between two time series: The 'optimal thermal causal path' method. *Quantitative Finance, 5*(6), 577–591. https://doi.org/10.1080/14697680500383763

Stavroglou, S., Pantelous, A., Soramaki, K., & Zuev, K. (2017). Causality networks of financial assets. *The Journal of Network Theory in Finance, 3*(2), 17–67. https://doi.org/10.21314/jntf.2017.029

Sussman, D. L., Tang, M., Fishkind, D. E., & Priebe, C. E. (2012). A consistent adjacency spectral embedding for stochastic blockmodel graphs. *Journal of the American Statistical Association, 107*(499), 1119–1128. https://doi.org/10.1080/01621459.2012.699795arXiv:1108.2228

Székely, G. J., Rizzo, M. L., & Bakirov, N. K. (2007). Measuring and testing dependence by correlation of distances. *Annals of Statistics, 35*(6), 2769–2794. https://doi.org/10.1214/009053607000000505

Thorp, E. O. (2011). The Kelly criterion in blackjack sports betting, and the stock market. In *The Kelly capital growth investment criterion (Chapter 9)*. World Scientific Book https://doi.org/10.1016/s1872-0978(06)01009-x

Traag, V. A., Waltman, L., & van Eck, N. J. (2019). From Louvain to Leiden: Guaranteeing well-connected communities. *Scientific Reports, 9*, 1–12. https://doi.org/10.1038/s41598-019-41695-zarXiv:1810.08473

Tumminello, M., Lillo, F., & Mantegna, R. N. (2010). Correlation, hierarchies, and networks in financial markets. *Journal of Economic Behavior and Organization, 75*(1), 40–58. https://doi.org/10.1016/j.jebo.2010.01.004arXiv:0809.4615

Underwood, W. G., Elliott, A., & Cucuringu, M. (2020). Motif-based spectral clustering of weighted directed networks. *Applied Network Science, 5*(62), 1–14.

Výrost, T., Lyócsa, Š, & Baumöhl, E. (2015). Granger causality stock market networks: Temporal proximity and preferential attachment. *Physica A: Statistical Mechanics and its Applications, 427*, 262–276. https://doi.org/10.1016/j.physa.2015.02.017

Wang, D., Tu, J., Chang, X., & Li, S. (2017). The lead–lag relationship between the spot and futures markets in China. *Quantitative Finance, 17*(9), 1447–1456. https://doi.org/10.1080/14697688.2016.1264616

Wang, G. J., Xie, C., He, K., & Stanley, H. E. (2017). Extreme risk spillover network: Application to financial institutions. *Quantitative Finance, 17*(9), 1417–1433. https://doi.org/10.1080/14697688.2016.1272762

Wharton Research Data Service (2020) Center for Research in Security Prices (CRSP)

Wu, D., Ke, Y., & Yu, J. X., Chen, L.(2010). Detecting leaders from correlated time series. In *International conference on database systems for advanced applications 5981 LNCS* (pp. 352–367). https://doi.org/10.1007/978-3-642-12026-8_28

Xia, L., You, D., Jiang, X., & Chen, W. (2018). Emergence and temporal structure of lead–lag correlations in collective stock dynamics. *Physica A: Statistical Mechanics and its Applications, 502,* 545–553. https://doi.org/10.1016/j.physa.2018.02.112

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	Lead–Lag Detection and Network Clustering for Multivariate Time Series with an Application to the US Equity Market
Publication Status	☒ Published □ Accepted for Publication □ Submitted for Publication □ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Published in Springer Machine Learning (2022)

Student Confirmation

Student Name:	Stefanos Bennett		
Contribution to the Paper	This is my work under the supervision of my advisors, Profs. Mihai Cucuringu and Gesine Reinert.		
Signature		Date	02/01/2024

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Mihai Cucuringu		
Supervisor comments Stefanos made a substantial contribution to the publication, as indicated above.		
Signature 	Date	5 January 2024

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 3

Time Series Prediction under Distribution Shift using Differentiable Forgetting

We observe the effects of financial time series non-stationarity in the year-to-year variation of the US equity lead-lag clusters in Chapter 2. This is consistent with prior research indicating that financial lead-lag relationships vary with market conditions [Ren et al., 2019, Xiao et al., 2023]. In the results of Chapter 2, we also observe that the forecasting performance of an equity return model based on lead-lag clustering diminishes with time. We expect forecasting performance to decrease when market conditions change more frequently, trading frictions decrease or market efficiency increases. When market conditions change more frequently, the true lead-lag network at any given point in time will change rapidly. In this case, it may be advantageous to apply distribution shift methods in the estimation of the lead-lag network. For instance, placing greater emphasis on a subset of the most recent data may yield a more accurate estimate of the true lead-lag network at any given point in time. This illustrates the potential applicability of a distribution shift method using data re-weighting on the task of interaction effect modelling.

Non-stationarity affects additional interaction modeling problems, including synchronous-correlation modeling. Chapter 3 develops a method for generally adapting time series models in the presence of distribution shift. Model-agnostic distribution shift methods are valuable tools for financial practitioners, considering the wide range of financial datasets that demonstrate non-stationarity and the diversity of financial models. We demonstrate the performance of our novel distribution shift adaptation method on an interaction modelling task in Chapter 3. Specifically, we apply our distribution shift method to the estimation of a cross-sectional factor model. The factor model relates the return of an equity to the contemporaneous return of three risk factors. Factor modelling is a sub-problem of modelling the synchronous interactions between equity returns. The factor loadings can be viewed as the edge weights in a bipartite network representing the links between equity and factor returns. This application demonstrates the utility of our method in comprehensively modelling non-stationary financial systems.

Time Series Prediction under Distribution Shift using Differentiable Forgetting

Stefanos Bennett [1 2] **Jase Clarkson** [1 2]

Abstract

Time series prediction is often complicated by distribution shift which demands adaptive models to accommodate time-varying distributions. We frame time series prediction under distribution shift as a weighted empirical risk minimisation problem. The weighting of previous observations in the empirical risk is determined by a forgetting mechanism that controls the trade-off between the relevancy and effective sample size that is used for the estimation of the predictive model. In contrast to previous work, we propose a gradient-based learning method for the parameters of the forgetting mechanism. This speeds up optimisation and therefore allows more expressive forgetting mechanisms. We theoretically situate our method and demonstrate its efficacy on synthetic and real-world datasets.

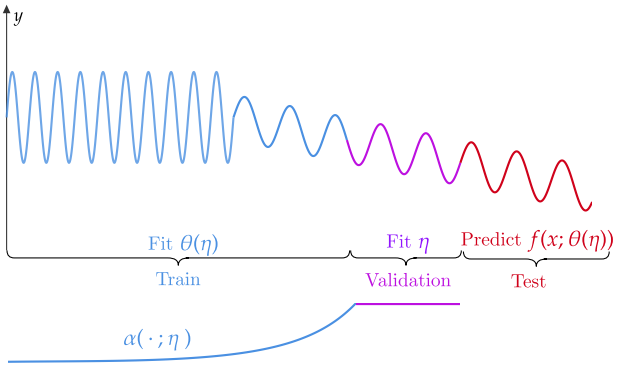


Figure 1: Method visualisation: time series training samples (top) are weighted by the forgetting mechanism $\alpha(\cdot;\eta)$ (bottom). The forgetting mechanism, which is parameterised by η and optimises predictive performance on the validation set data, assigns weight to the most recent, representative training samples.

1. Introduction

We introduce a new method for the problem of predicting time series that exhibit distribution shift, also known as *concept drift* in the domain of engineering (Lu et al., 2019) or *forecasting under unstable environments* in economics (Rossi, 2013). Time series exhibiting distribution shift appear in a wide range of domains (Zliobaite et al., 2016) from industrial management to biomedical applications. Financial time series distribution shift has been studied in the concept drift literature (Harries & Horn, 1996), as well as more recently in the statistics (McCarthy & Jensen, 2016) and learning theory (Kuznetsov & Mohri, 2020) literatures. In fields such as economics and finance (Rossi, 2013), distribution shift often occurs at unknown times and is of unknown type. This motivates the need for a model-agnostic method that achieves competitive performance across a range of different distribution shift settings.

In this work, we introduce a model-agnostic distribution shift method for time series that uses a generic forgetting mechanism to infer sample importance weights. The key novel contribution of our work is a gradient-based learning method for directly optimising the validation performance of differentiable forgetting mechanisms. The main advantages of our method are that it is easy-to-use with any machine learning model fit with empirical risk minimisation, provides quick estimation of forgetting mechanism hyperparameters – thereby facilitating expressive forgetting mechanisms – and has competitive performance compared to popular baseline methods. Our method is able to model a range of synthetically generated distribution shifts, from gradual drift to regime switching behaviour, while maintaining strong performance in the stationary case. Through experiments on real-world forecasting tasks, we show that our method can outperform distribution shift baselines which use data re-weighting. A link to our code is available in the references under Anon (2023).

The remainder of this paper is structured as follows. We start with the problem definition in Section 2. Section 3 is a description of our proposed method for tackling distribution shift in time series. Section 4 details related work. Section 5 describes our synthetic and real-world experimental settings. Section 6 presents experimental results and baseline comparisons. Section 7 discusses the sensitivity of our method to the forgetting mechanism form and dataset size. Finally, we provide connections to active research areas in Section 8 and suggest future work directions.

2. Problem Definition

Consider a sequential supervised learning problem where for each $t = 1, \ldots, T$, we wish to learn a mapping $f(\,\cdot\,; \theta_t) : X_t \mapsto Y_t$ between features X_t and labels Y_t where $\theta_t \in \Theta$, for some parameter space Θ. We use a train-validation-test split of

$$D_{\text{train}} = \{(X_t, Y_t)\}_{t=1}^{t^*-1},$$
$$D_{\text{val}} = \{(X_t, Y_t)\}_{t=t^*}^{T_{\text{test}}-1},$$
$$D_{\text{test}} = \{(X_t, Y_t)\}_{t=T_{\text{test}}}^{T},$$

for a choice of t^*, which indexes the start of the validation set, and T_{test}, which indexes the start of the test set ($1 \leq t^* \leq T_{test} < T$).

The labels, conditional on the features, follow a time-varying distribution $Y_t|X \sim \pi_t(\cdot|X)$. At each test set time index $t = T_{test}, \ldots, T$, we are interested in minimising the one-step-ahead path-dependent risk $R_t(\theta) = \mathbb{E}_{Y_t \sim \pi_t(\cdot|X_t)} \left[L\left(f(X_t; \theta), Y_t\right) | \{(X_\tau, Y_\tau)\}_{\tau=1}^{t-1} \right]$, over $\theta \in \Theta$, for a given loss function L. We follow an Empirical Risk Minimisation (ERM) (Kuznetsov & Mohri, 2020) approach in which we estimate the optimal model parameters θ_t by minimising the risk estimator $\hat{R}_t(\theta)$.

3. Model Description

3.1. Model Adaptation to Distribution Shift via Weighted Empirical Risk Minimisation

Our approach to accommodate distribution shift in the ERM formulation is through the choice of the one-step-ahead risk estimator which aims to estimate the risk over the test set using a weighted sum of training points:

$$\hat{R}_{t^*}(\theta) = \sum_{\tau=1}^{t^*-1} \alpha(t^* - 1 - \tau; \eta) L\left(f(X_\tau; \theta), Y_\tau\right) \quad (1)$$

where the weights $\{\alpha(i; \eta)\}_{i=0}^{t^*-2}$ are the outputs of a forgetting mechanism parameterised by η.

The purpose of re-weighting empirical loss samples is to find a linear combination of training set sample losses that yields a model with improved generalisation. We can encode inductive biases by choosing the functional form of the forgetting mechanism. For instance, a monotonically decreasing forgetting mechanism encodes the hypothesis that more recent samples are more relevant for training. We introduce examples of specific functional forms in Section 3.4. The learning of forgetting mechanism parameters η controls the trade-off between the adaptivity of the forecaster and the statistical efficiency of the learning procedure: forgetting mechanisms that assign high weight to a few points can capture more relevant data at the cost of reducing the effective sample size that is used to fit the underlying model.

3.2. Differentiable Bi-Level Optimisation

We jointly learn the predictive model and forgetting mechanism parameters (θ, η) used in Equation (1) by solving the bi-level optimisation problem

$$\min_{\eta} \; g^U(\eta, \hat{\theta}) \; \textit{such that} \; \hat{\theta} \in \operatorname*{argmin}_{\theta} g^L(\eta, \theta),$$

where g^U and g^L represent the upper and lower level objective functions

$$g^U(\eta, \theta) := \sum_{(X_t, Y_t) \in D_{valid}} L(f(X_t; \theta), Y_t)$$
$$g^L(\eta, \theta) := \sum_{(X_\tau, Y_\tau) \in D_{train}} \alpha(t^* - \tau; \eta) L(f(X_\tau; \theta), Y_\tau).$$

As described in Section 2, D_{valid} denotes the validation set, which is selected to be the most recent $T_{test} - t^*$ data samples, for some choice of t^*. The assumption underlying this choice of validation set is that the distribution $\pi_t(\cdot|X)$ does not, on average, substantially change in any given small time span (Kuznetsov & Mohri, 2020). We discuss the robustness of the method to the choice of t^* in Section 6 and Section 7.

Applying the implicit function theorem to bi-level optimisation, we are able to perform gradient-based learning of the forgetting mechanism parameters η. Specifically, Gould et al. (2016) provide the following Lemma for differentiating through the argmin in the lower level problem:

Lemma 3.1. *Let $g^L : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ be an element of $C^2(\mathbb{R}^m, \mathbb{R}^n)$, the set of twice continuously differentiable functions in each argument, and let $\hat{\theta}(\eta) = \operatorname*{argmin}_{\theta} g^L(\eta, \theta)$. Then the derivative of $\hat{\theta}$ with respect to η_i, $i = 1, \ldots, m$ is given by*

$$\frac{\partial \hat{\theta}(\eta)}{\partial \eta_i} = -\left(\nabla^2_\theta g^L(\eta, \theta)\big|_{\hat{\theta}}\right)^{-1} \frac{\partial}{\partial \eta_i} \nabla_\theta g^L(\eta, \theta)\big|_{\hat{\theta}}. \quad (2)$$

Therefore, by the chain rule, the total derivative of $g^U(\eta, \hat{\theta}(\eta))$ with respect to η_i is given by:

$$\frac{d}{d\eta_i} g^U(\eta, \hat{\theta}(\eta)) = \frac{\partial g^U}{\partial \eta_i} + \left(\nabla_\theta g^U\right)^T \left.\frac{\partial \hat{\theta}(\eta)}{\partial \eta_i}\right|_{\hat{\theta}} \quad (3)$$

where we use the gradient expression in equation (2). Equation (3) gives us the required expression to apply gradient descent in η to the upper-level objective function:

$$\eta \leftarrow \eta - \lambda \frac{d}{d\eta} g^U(\eta, \hat{\boldsymbol{\theta}}(\eta)). \quad (4)$$

where $\lambda \in \mathbb{R}$ is the step size.

Theoretical underpinnings The approach of loss re-weighting is well-established and theoretically supported in the distribution shift literature (Kuznetsov & Mohri, 2020; Lu et al., 2021). The foundation of the loss re-weighting approach is based on importance sampling (Kanamori et al., 2009; Zhang et al., 2021). Using the importance sampling principle (Shimodaira, 2000), we can obtain an unbiased estimator of the risk R_t at any test set point indexed by t, using

$$\hat{R}_{t^*}(\theta) = \frac{1}{t^* - 1} \sum_{\tau=1}^{t^*-1} \frac{\pi_t(y_\tau|x_\tau)}{\pi_\tau(y_\tau|x_\tau)} L(f(x_\tau;\theta), y_\tau), \quad (5)$$

where $\pi(\cdot)$ is defined in Section 2. In general, each ratio $\frac{\pi_t(y_\tau|x_\tau)}{\pi_\tau(y_\tau|x_\tau)}$ is unknown. Our approach can be seen as estimating these ratios by modelling them with a parametric weighting function in τ, which we call a forgetting mechanism. In Appendix F, we derive an example forgetting mechanism based on the functional form of the optimal importance sampling ratio under a simple data generating process.

The computational convergence of bi-level optimisation to a stationary point of the upper level problem is given under the smoothness assumptions described in Appendix E.

3.3. Bi-level Optimisation Algorithm

Our proposed method, which we call GF, for the bi-level optimisation of a forgetting mechanism and a predictive model is detailed in Algorithm 1 of Appendix A. The warm initialisation of the predictive model parameters θ_{ERM} is given by the solution to the uniformly-weighted ERM problem. We then alternately minimise the upper and lower level objectives using gradient descent with Equation (4). The number of iterations L for the lower objective loop is set to 10% of the total iterations that were originally used to train the model in the warm initialisation step[1]. Since the upper level problem is generally non-convex with multiple local minima, we propose restarting the procedure R times to explore different local solutions. We mean-aggregate the predictive models obtained from these multiple restarts to return a final model.

Computational and space complexity The overhead relative to standard uniformly-weighted ERM lies in the computation of the hyper-gradient given in Equation (2). The Normal Conjugate Gradient method (Ren et al., 2022) for hyper-gradient computation is $O(p^2)$ in both memory and computation, where p is the number of parameters of the predictive model. The computational and memory cost of the hyper-gradient updates can be further reduced through the use of a Neumann series approximation to the Hessian (Yang et al., 2021; Lorraine et al., 2020). Lorraine et al. (2020) study this approximation and compare it to a number of approximate schemes that have been proposed in the hyper-gradient optimisation literature. These approximate schemes provide computationally feasible hyper-parameter optimisation for large predictive models. Lorraine et al. (2020) find that the Neumann series approximation is stable and can recover the performance of exact Hessian computation using as few as 5-20 Neumann series iterations. To illustrate this application on our method, we provide the results for the Neumann series approximate solver with a transformer predictive model in Appendix H. All other experiments are run using the Normal Conjugate Gradient method (Ren et al., 2022).

[1]However, we find that we can obtain reasonably accurate results using just 5% of the warm initialisation iteration budget.

3.4. Forgetting Mechanism Forms

We consider four instances of forgetting mechanisms in our experiments.

1. GF-EXP:

$$\alpha(\tau;\eta) = \exp(-\eta_1\tau),\ \eta_1 \geq 0.$$

2. GF-MIX:

$$\alpha(\tau;\eta) = \exp\big(-\eta_1\tau - \eta_2\tau^2 - \eta_3\log(\tau+1)\big),$$

$$\eta_1, \eta_2, \eta_3 \geq 0.$$

3. GF-MLP:

$$\alpha(\tau;\eta) = \mathrm{MLP}(\tau;\eta),$$

where MLP is a multi-layer perceptron with a single hidden layer of 32 units, LeakyReLU activation, layer-normalisation and a sigmoid output activation function.

4. GF-MLP2:

$$\alpha(\tau;\eta) = \mathrm{MLP2}(\tau;\eta),$$

where MLP2 is defined similarly to MLP in the definition of GF-MLP but has two hidden layers.

These four instances allow us to contrast two approaches to forgetting mechanism selection.

Our first approach considers low-variance forgetting mechanisms (GF-EXP and GF-MIX) which enforce the strong inductive bias that "more recent samples are more relevant". Forgetting mechanism GF-EXP corresponds to exponential decay. GF-MIX corresponds to a mixture of various functional forms of decay. We motivate these mechanisms in Appendix F by drawing on the simple example of a linear

model with coefficient drift and deriving a heuristic approximation to the optimal importance sampling ratios.

Our second approach to selecting a forgetting mechanism uses MLPs to accommodate non-monotonic distribution shift scenarios. Since the forgetting mechanism maps a univariate input (time index) to output (sample weight), we propose to use simple architectures as a proof-of-concept for this low-dimensional modelling problem. Our proposed architectures are in line with the literature on sample weight modelling based on a univariate input: Shu et al. (2019) use a single-layer MLP for the related problem of mapping a loss value to a sample weight. Furthermore, the MLP architectures GF-MLP and GF-MLP2 have found success as implicit neural representation models (Sitzmann et al., 2020), which build representations from coordinate inputs, and have also been successfully applied to time-index forecasting tasks (Woo et al., 2022).

For all forgetting mechanisms, the time-index input τ is normalised by its maximum value on the time series to which it is applied.

4. Related Work

Origins Recursive least squares (Haykin, 1986; Rossi, 2013; Harvey, 1990) provides an early example of the use of a simple forgetting mechanism for model adaptation. Since then, the concept drift literature has studied the use of forgetting mechanisms for time series distribution shift on an algorithm-by-algorithm basis (Koychev, 2000; Klinkenberg, 2004). For instance, Klinkenberg (2004) proposes several methods to handle concept drift using support vector machines with a simple forgetting mechanism that is selected with grid-search.

Non-stationary time series forecasting McCarthy & Jensen (2016); Masserano et al. (2022); Kuznetsov & Mohri (2020); Yusupova et al. (2022) have also considered the use of forgetting mechanisms for dealing with non-stationary time series forecasting. McCarthy & Jensen (2016) propose an alternative Bayesian formulation to our ERM-based approach. In parallel work to ours, Masserano et al. (2022) use Bayesian optimisation to learn the parameters for a simple forgetting mechanism for time series prediction under distribution shift. Masserano et al. (2022) conclude "While we find Bayesian optimization to be a powerful technique to find good adaptive sampling parameters, this approach is computationally expensive (because many parallel trials need to be evaluated) and could degrade as the number of parameters increases." Kuznetsov & Mohri (2020) propose an estimator for the one-step-ahead risk that is based on a weighted empirical risk. They derive a learning guarantee on the one-step-ahead generalisation error that holds in the non-stationary time series setting. They then minimise this learning guarantee over the weights in their empirical risk and the parameters of their model. The first key difference between their work and ours is that we model the weights using a parametric forgetting mechanism. The second key difference is that they optimise an upper bound on the generalisation error to derive sample weights whereas we directly minimise the validation-set prediction error to estimate these weights.

Hyperparameter optimisation Pedregosa (2016) provide a comparison of gradient-based to grid search and Bayesian hyper-parameter optimisation approaches. Faster optimisation allows for more complex, richly-parameterised forgetting mechanisms that would be too computationally burdensome to fit using grid search. Our method connects time series prediction under distribution shift to the growing field of automated machine learning (Hutter et al., 2018), allowing us to leverage powerful modern automatic differentiation libraries for efficient implementation (Ren et al., 2022).

Beyond time series prediction The idea of re-weighting empirical loss to address distribution shift has been considered in the context of importance weighting for robust deep learning (Shu et al., 2019). Motivated by the problem of data noise and class imbalance, Shu et al. (2019) propose to train a neural network that outputs the ERM importance weight of a training sample given that sample's loss. The authors propose to use a MAML-style (Finn et al., 2017) iterative algorithm for jointly optimising the importance weights and prediction model. This contrasts with our approach to solving the bi-level optimisation problem using implicit differentiation.

Iterative and implicit hyper-gradient optimisation Grazzi et al. (2020) compare the convergence behaviour of iterative and implicit hypergradient methods theoretically and experimentally. They find that implicit hypergradient methods using conjugate gradient calculations are preferable due to a potentiality better approximation of the hypergradient and lower space complexity. The work of Shu et al. (2019) and Jenni & Favaro (2018) illustrates the interest in applying differentiable bi-level optimisation methods to the problem of generalisation in non-temporal data settings.

Importance sampling Viewing weighted empirical risk (Equation (1)) through the lens of importance sampling has led to the development of density ratio estimation approaches for fitting the sampling weights (Lu et al., 2021). These are popular approaches in the field of covariate shift adaptation (Kanamori et al., 2009; Zhang et al., 2021). Fang et al. (2020) introduce an importance weight estimation method for distribution shift which is compatible with deep predictive models. In their work, they dynamically learn a

non-linear transformation of the data which is used to obtain the importance weights through distribution matching. The non-linear data transformation is iteratively updated alongside the predictive model.

Comparison to existing non-stationary time series methods Our work distinguishes itself from the existing concept drift and non-stationary time series forecasting literature as it provides gradient-based learning for a generic forgetting mechanism and ERM-based learning model. A key novelty of our work is gradient-based minimisation of the validation error over the parameters of the forgetting mechanism. Gradient-based approaches are known to be much faster (Hutter et al., 2018) than grid search, which has typically been used to estimate the forgetting mechanism parameters in the concept drift (Koychev, 2000; Klinkenberg, 2004) and non-stationary time series literature (McCarthy & Jensen, 2016). Our hyper-gradient optimisation proposal is, therefore, a solution to the infeasibility of high-dimensional random search approaches described in Masserano et al. (2022).

5. Experimental Settings

To examine the efficacy of our proposed method, we evaluate our method on synthetic and real-world datasets which feature temporal distribution shift. Our evaluation metric is out-of-sample predictive performance. We evaluate the effect of our proposed method on five time series forecasting models.

5.1. Baselines

We compare our proposed method to five model-agnostic baselines for handling distribution shift which use sample re-weighting. These baselines range from popular, naive approaches (UNIFORM, WINDOW) to state-of-the-art distribution shift methods (DBF, META and DIW).

The UNIFORM method assumes no underlying distribution shift in the time series; it fits predictive models using uniform weighting on all samples. The popular WINDOW method applies uniform weights to a fixed-length interval of the most recent samples. DBF is an implementation of the two-step, discrepancy-based forecasting algorithm from Kuznetsov & Mohri (2020). META is the Meta-Weight-Net algorithm of Shu et al. (2019), which trains a neural network to re-weight examples based on their training loss. DIW is the loss-value transformation version of the dynamic importance weighting algorithm of Fang et al. (2020).

Implementation details and hyper-parameters for our own method and all baselines are reported in the Appendix C.

5.2. Synthetic Data

Our synthetic data experiments are based on those from Kuznetsov & Mohri (2020). Four distribution shift settings are considered: regularly occurring change points, gradual distribution drift, irregularly occurring change points and no distribution shift. Data are generated by the following equations for $t = 1, \dots, 3000$ and $\epsilon_t \overset{\text{iid}}{\sim} N(0, 0.05^2)$:

FixedRegime: $Y_t = \theta_t Y_{t-1} + \epsilon_t$, where $\theta_t = -0.9$ if $t \in [1000, 2000]$ and 0.9 otherwise.

RandomWalk: $Y_t = \theta_t Y_{t-1} + \epsilon_t$, where $\theta_t = 1 - (t/1500)$.

RandomRegime: $Y_t = \theta_{i_t} Y_{t-1} + \epsilon_t$, where $\theta_1 = -0.5$, $\theta_2 = 0.9$ and i_t is the stochastic process on $\{1, 2\}$ such that $\mathbb{P}(i_{s+t} = i | i_{s+t-1:s} = i, i_{s-1} \neq i) = (0.99998255)^t$.

Stationary: $Y_t = -0.5 Y_{t-1} + \epsilon_t$.

Predictive model The task in each setting is to forecast the time series value Y_t using a linear model with a feature that consists of the last value of the time series Y_{t-1}. The train-validation-test split for the synthetic data experiments is $D_{train}, D_{val}, D_{test} = \{(X_t, Y_t)\}_{t=1}^{2875}, \{(X_t, Y_t)\}_{t=2876}^{2975}, \{(X_t, Y_t)\}_{t=2976}^{3000}$. For each synthetic setting, we fit the linear model using each distribution shift method under consideration. We report the test MSE loss of the inferred model averaged over 20 Monte Carlo runs in Table 1.

5.3. Real Data

Datasets and tasks We evaluate our method on seven popular real-world time series prediction tasks. These consist of two financial modelling tasks (FACTOR and IR), an economic variable prediction task (FRED-MD), two epidemiological forecasting tasks (COVID and ILI), an energy prediction task (SOLAR) and the M4 forecasting competition. The FACTOR dataset corresponds to a cross-sectional modelling task; the task consists in linearly decomposing equity returns using the Fama-French three-factor model (Fama & French, 1993) to minimise the out-of-sample squared error of residual returns. The other six real-world datasets correspond to auto-regressive modelling tasks. The task for each time series i in dataset $\mathcal{D}$ is to forecast time series value Y_t using a feature that consists of the most recent $L_{\mathcal{D}}$ values of the time series $X_t = (Y_{t-1} \dots, Y_{t-L_{\mathcal{D}}})$. We use a single-split single-step evaluation approach in order to be consistent with the M4 time-series forecasting benchmark (Makridakis et al., 2020). A full description of each dataset and forecasting task is provided in Appendix B; we also

provide references to prior work that studies the problem of distribution shift on these datasets.

Predictive models The FACTOR cross-sectional prediction task uses linear modelling. For the other six forecasting tasks, we study the performance of the distribution shift methods applied to four auto-regressive predictive models: these are a linear auto-regressive model denoted LINEAR, a MLP model, a dilated-causal CNN model named WAVENET, and a transformer encoder model named TF-ENC. The details on all predictive models are provided in Appendix D.

Evaluation metric We use the popular MASE loss (Hyndman & Koehler, 2006) for performance evaluation on the real-world auto-regressive forecasting tasks. This evaluates the out-of-sample predictive performance of an auto-regressive model that has been fit on a given time series using a specific distribution shift method.

Suppose that dataset $\mathcal{D}$ contains N time series indexed by $i = 1, \ldots, N$. Having fit predictive model f on time series i using a given distribution shift method, we evaluate its MASE on time series i through

$$\text{MASE}_i = \frac{|D_{ts}|^{-1} \sum_{(X_\tau, Y_\tau) \in D_{ts}} |Y_\tau - f(X_\tau)|}{|D_{tr} \cup D_{val}|^{-1} \sum_{Y_t \in D_{tr} \cup D_{val}} |Y_t - Y_{t-1}|} \tag{6}$$

where D_{tr}, D_{val}, D_{ts} are respectively the training, validation and test sets. Finally, the dataset-level MASE is computed by averaging across all time series in dataset $\mathcal{D}$:

$$\text{MASE} = \frac{1}{N} \sum_{i=1}^{N} \text{MASE}_i \,. \tag{7}$$

5.4. Validation set selection

Using the notation defined above, we choose the validation set to be the most recent $t = t^*, \ldots, T_{test} - 1$ samples. The assumption underlying this choice is that the distribution $\pi_t(\cdot|X)$ does not, on average, substantially change in any given small time span (cf bounded discrepancy (Kuznetsov & Mohri, 2020)). This is a weak assumption for a large enough $t^*, t^* < T_{test}$: indeed if it does hold then the out-of-sample distribution can be arbitrarily different to the training set rendering model adaptation impossible.

The heuristic we apply in selecting the value of $t*$ (see Table 3) is as follows. When T is large (over 900), we select a moderate validation set size (100 - 366). When T is small, we aim to maintain a reasonable ratio of points in the training and validation set. Across the real world experiments, the fraction of points in the validation set ranges 0.03–0.43. This allows us to experimentally evaluate the validity of our validation set size heuristic as well as the sensitivity of our results to different validation set fractions. Further, in Section 7, we describe the results of an experiment that varies the size of the validation set to examine our method's sensitivity to t^*.

6. Experimental Results

6.1. Synthetic Data Results

We see from Table 1 that the GF methods perform best on all synthetic settings. The outperformance of GF-MLP and GF-MLP2 on the FIXEDREGIME and RANDOMREGIME datasets illustrates the advantage of more expressive differentiable forgetting mechanisms which would be too computationally burdensome to fit using grid search. In Figure 2, we display the sample weights inferred on a Monte Carlo run of the FIXEDREGIME dataset. The more flexible GF-MLP and GF-MLP2 forgetting mechanisms are better equipped to capture the sudden regime change at the time index 2000. Furthermore, GF-MLP2 and, to some extent, GF-MLP, are able to capture the non-monotonic character of the distribution shift by placing non-zero weight on the samples in the first regime $t \in [0, 999]$.

6.2. Real Data Results

Table 2 displays the experimental results on the real-world tasks[2]. These results demonstrate the competitive predictive performance that can be achieved with the use of a

[2]Due to the poor performance of DBF on the synthetic experiments and its computationally expensive run-time, we did not run it on real-world datasets.

Table 1: This table reports the Mean Squared Error (MSE $\times 10^3$) for each method averaged across 20 Monte Carlo repetitions on the four synthetic data settings. For each data set, the MSE of the best-performing method is shown in bold.

DATASET	UNIFORM	META	DIW	WINDOW	DBF	GF-EXP	GF-MIX	GF-MLP	GF-MLP2
FIXEDREGIME	9.15	10.27	11.00	3.75	15.79	3.51	2.98	2.94	**2.91**
RANDOMWALK	44.98	57.42	16.26	12.16	85.84	13.45	8.43	9.08	**6.66**
RANDOMREGIME	3.90	4.04	4.60	4.00	6.61	3.89	3.93	**3.76**	3.78
STATIONARY	2.86	2.90	2.87	**2.85**	3.77	**2.85**	**2.85**	2.86	**2.85**

DATASET	MODEL	UNIFORM	META	DIW	WINDOW	GF-EXP	GF-MIX	GF-MLP	GF-MLP2
FACTOR	LINEAR	1.46	1.83	2.11	1.44	1.31	**1.30**	1.31	**1.30**
COVID	LINEAR	1.61	1.37	1.76	0.46	0.35	0.35	0.35	**0.34**
	MLP	1.89	1.85	2.43	**0.53**	0.61	0.61	0.66	0.69
FRED-MD	LINEAR	1.15	1.64	1.77	1.15	**1.11**	**1.11**	1.13	1.12
	MLP	1.16	2.14	2.38	1.17	1.14	**1.13**	1.16	1.14
ILI	LINEAR	0.75	0.78	0.84	0.76	**0.73**	**0.73**	**0.73**	**0.73**
	MLP	0.78	0.76	0.77	0.78	0.78	0.78	**0.74**	0.77
IR	LINEAR	0.64	0.63	0.79	0.64	0.63	0.62	**0.61**	**0.61**
	MLP	0.69	0.72	0.98	0.69	**0.66**	**0.66**	0.72	0.69
M4	LINEAR	1.05	1.14	1.36	1.02	1.00	0.99	1.02	**0.98**
	MLP	1.08	1.24	1.52	1.07	1.07	1.07	**1.06**	1.08
SOLAR	LINEAR	0.61	0.61	0.66	0.60	**0.59**	0.60	**0.59**	**0.59**
	MLP	0.47	0.48	0.56	0.46	0.46	0.47	0.46	**0.45**

Table 2: This table reports the average test MASE for each combination of dataset and predictive model by distribution shift method (shown across the columns) on each real-world dataset. Results for the FACTOR dataset are reported in MSE $\times 10^4$. For each dataset and predictive model pair, the loss of the best-performing distribution shift method is shown in bold.

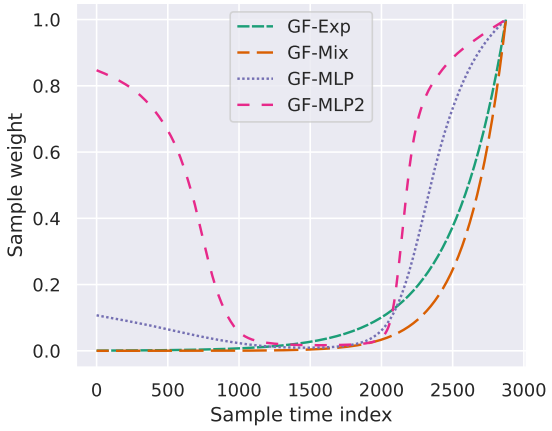


Figure 2: Example of the inferred sample weights by each forgetting mechanism on a single instance of the FIXEDREGIME dataset.

forgetting mechanism which is trained using gradient-based bi-level optimisation on a range of real-world tasks. Relative to UNIFORM, GF-MLP has an average reduction in loss across all datasets of 15% for the LINEAR predictive model and an average reduction of 12% for the MLP predictive model. Similar average reduction statistics are true for GF-EXP, GF-MIX and GF-MLP2. The reduction in test loss relative to UNIFORM is largest for the COVID, a dataset in which there are pronounced distribution shifts (Arik et al., 2022). Based on the average rank in performance of each distribution-shift method and predictive model pair computed across datasets, we find that seven out of the top eight pairs use GF as a forgetting mechanism.

Full experimental results including those using predictive models WAVENET and TF-ENC can be found in Appendix H.

7. Sensitivity Analysis

Validation length We investigate the sensitivity of the GF method to the choice of validation length on the FIXEDREGIME dataset. The performance of GF-MIX and GF-MLP as a function of validation set size is shown in Figure 3.

Using a validation set size of 75 to 1000, GF-MLP is able to achieve near-optimal forecasting error; note that the Bayes rule MSE for the synthetic settings is 0.0025. The corresponding optimal validation set size range for GF-MIX is 75 to 500. We see that the GF-MLP is more robust to the choice of validation set size since it is able to place weight on samples in the first regime $t \in [0, 999]$ even as the training set reduces in size with increasing validation length. This figure also shows the limitations of the GF method: as the set size increases past 1000, the validation interval increasingly contains samples from a different regime to the test distribution; therefore, test forecasting performance degrades. Nonetheless, we note that even at the extreme ends $t = 5, 1500$ of the validation length range, the performance of the GF methods match UNIFORM.

Across the real-world experiments, the fraction of points in the validation set ranges 0.03–0.43: we see strong performance across this range. In Appendix H, we include a breakdown of the M4 results by time series sample frequency. As discussed in Appendix H, we find that the performance of GF drops when the time series, and therefore the validation

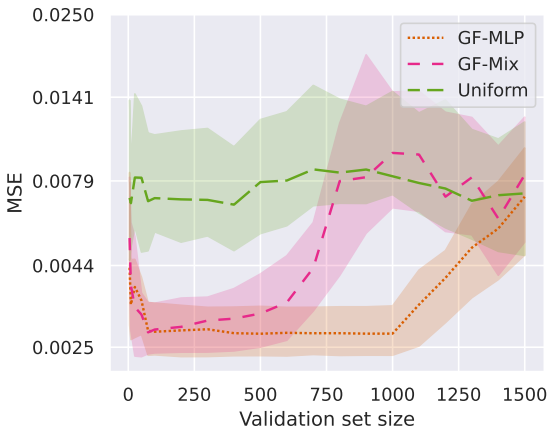


Figure 3: An ablation study showing the sensitivity of UNIFORM, GF-MIX and GF-MLP to the validation set size on the FIXEDREGIME dataset. The out-of-sample MSE median and inter-quartile range across the 20 Monte Carlo samples for each method are shown on the vertical axis using a log scale. The validation set size is varied while the test set size of 25 and combined training and validation length of 2975 remains constant.

set, has few samples. In these cases, inferences based on the limited validation set will have high variance and simpler distribution shift methods such as UNIFORM or WINDOW are more likely to succeed.

Choice of forgetting mechanism On the synthetic data generating processes (Table 1), which have a clear and prominent distribution shift structure, we find that it is beneficial to use the more expressive GF-MLP2 architecture.

On real data, the inductive bias that "more recent samples are more relevant" helps: the performance of GF-EXP and GF-MIX is in line with that of GF-MLP and GF-MLP2. The reader is referred to Appendix G for an illustration of the inferred weights on a time series in the FACTOR dataset. As seen in 2, the best performing GF scheme varies from setting to setting: this is not surprising given the breadth of experimentation. Nevertheless, all mechanisms achieve strong performance compared to competitor distribution shift methods.

Comparing the real data results for GF-MLP and GF-MLP2 in Table 2, we find that the performance of MLP-based forgetting mechanisms is robust to the choice of the number of hidden layers. In results not reported, we experiment using an MLP with three hidden layers: this results in similar performance, however, training times are increased.

8. Future Work

We have shown that our proposed differentiable bi-level optimisation method can model a range of distribution shift scenarios across synthetic and real-world datasets. While we have focused on univariate forecasting tasks to simplify the exposition of our method, our approach could be extended to multivariate forecasting. The evaluation of our method in the multivariate forecasting context is subject to future work.

Forgetting mechanism extension Our method allows the possibility of using more richly-parameterised forgetting mechanisms for modelling distribution shift. A future avenue of work could be to draw on advances from the field of implicit neural representations (Sitzmann et al., 2020; Woo et al., 2022) to improve the mapping from time-index to sample weight. Extending the time-index input with a sample loss feature, as in Shu et al. (2019), could be a useful strategy to equip the forgetting mechanism to deal with data corruption or noise. More generally, it would be interesting to extend the feature inputs for the forgetting mechanism using the sample inputs and labels. In this context, sample input and label dimensionality reduction are crucial for importance ratio estimation (Maia Polo & Vicente, 2022; Stojanov et al., 2019). Joint feature extraction for the predictive model and forgetting mechanism may be useful for the purposes of this dimensionality reduction (Zhang et al., 2021).

Online learning extension Another interesting research extension would be to adapt our method to the online learning setting so that the sample weights and the predictive model can be updated in response to streaming data. The perturbation-based approach of Ren et al. (2018) provides an example of how sample re-weighting methods may be adapted to the streaming setting.

9. Conclusion

In summary, we propose learning forgetting mechanisms for time series prediction under distribution shift using gradient-based optimisation. Our approach is flexible, fast and can achieve competitive performance to baseline methods on synthetic and real-world datasets. More broadly, our paper motivates further research on the topics of distribution shift, time series prediction and hyper-parameter learning.

References

Anon. Project code repository. https://anonymous.4open.science/r/icml_tsds_24-F2F5, 2023.

Arik, S. O., Yoder, N. C., and Pfister, T. Self-adaptive forecasting for improved deep learning on non-stationary time-series. *arXiv preprint arXiv:2202.02403*, 2022.

Borovykh, A., Bohte, S., and Oosterlee, C. W. Conditional time series forecasting with convolutional neural networks. *arXiv preprint arXiv:1703.04691*, 2017.

Fama, E. F. and French, K. R. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1):3–56, 1993. ISSN 0046-9777.

Fang, T., Lu, N., Niu, G., and Sugiyama, M. Rethinking importance weighting for deep learning under distribution shift. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11996–12007, 2020.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks, 2017.

Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. MIT press, 2016.

Gould, S., Fernando, B., Cherian, A., Anderson, P., Cruz, R. S., and Guo, E. On Differentiating Parameterized Argmin and Argmax Problems with Application to Bi-level Optimization. *arXiv preprint arXiv:1607.05447*, 2016.

Grazzi, R., Franceschi, L., Pontil, M., and Salzo, S. On the iteration complexity of hypergradient computation. *37th International Conference on Machine Learning*, (2): 3706–3716, 2020.

Harries, M. and Horn, K. Detecting concept drift in financial time series prediction using symbolic machine learning. *AI'95 Proceedings*, 1996.

Harvey, A. C. *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge University Press, 1990.

Haykin, S. *Adaptive Filter Theory*. Prentice Hall, 1986.

Hong, M., Wai, H.-T., Wang, Z., and Yang, Z. A two-timescale stochastic algorithm framework for bilevel optimization: Complexity analysis and application to actor-critic. *SIAM Journal on Optimization*, 33(1):147–180, 2023.

Hutter, F., Kotthoff, L., and Vanschoren, J. *Automated Machine Learning: Methods, Systems, Challenges*. Springer, 2018.

Hyndman, R. J. and Koehler, A. B. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679–688, 2006. ISSN 0169-2070.

Jenni, S. and Favaro, P. Deep bilevel learning. *arXiv preprint arXiv:1809.01465*, 2018.

Ji, K., Yang, J., and Liang, Y. Bilevel optimization: Convergence analysis and enhanced design. In *International conference on machine learning*, pp. 4882–4892. PMLR, 2021.

Kanamori, T., Hido, S., and Sugiyama, M. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10:1391–1445, 2009. ISSN 15324435.

Kim, T., Kim, J., Tae, Y., Park, C., Choi, J.-H., and Choo, J. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International Conference on Learning Representations*, 2021.

Klinkenberg, R. Learning drifting concepts: Example selection vs. example weighting. *Intelligent Data Analysis*, 8 (3):281–300, 2004.

Koychev, I. Gradual Forgetting for adaptation to concept drift. *Proceedings of ECAI 2000 Workshop on Current Issues in Spatio-Temporal Reasoning*, 2000.

Kuznetsov, V. and Mohri, M. Discrepancy-Based Theory and Algorithms for Forecasting Non-Stationary Time Series. *Annals of Mathematics and Artificial Intelligence*, 88(4):367–399, 2020.

Liu, Y., Wu, H., Wang, J., and Long, M. Non-stationary transformers: Rethinking the stationarity in time series forecasting. In *Advances in Neural Information Processing Systems*, volume 35, pp. 9881–9893. Curran Associates, Inc., 2022.

Lorraine, J., Vicol, P., and Duvenaud, D. Optimizing millions of hyperparameters by implicit differentiation. In Chiappa, S. and Calandra, R. (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 1540–1552. PMLR, 26–28 Aug 2020.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., and Zhang, G. Learning under concept drift: A review. *IEEE Transactions on Knowledge and Data Engineering*, 31(12): 2346–2363, 2019.

Lu, N., Zhang, T., Fang, T., Teshima, T., and Sugiyama, M. Rethinking importance weighting for transfer learning. *arXiv preprint arXiv:2112.10157*, pp. 1–44, 2021.

Maia Polo, F. and Vicente, R. Effective sample size, dimensionality, and generalization in covariate shift adaptation. *Neural Computing and Applications*, 2022. ISSN 14333058.

Makridakis, S., Spiliotis, E., and Assimakopoulos, V. The M4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1): 54–74, 2020. ISSN 0169-2070.

Masserano, L., Rangapuram, S. S., Kapoor, S., Nirwan, R. S., Park, Y., and Bohlke-Schneider, M. Adaptive sampling for probabilistic forecasting under distribution shift. In *NeurIPS 2022 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2022.

McCarthy, D. and Jensen, S. T. Power-weighted densities for time series data. *The Annals of Applied Statistics*, 10 (1):305–334, 2016.

McCracken, M. W. and Ng, S. FRED-MD: A monthly database for macroeconomic research. *Journal of Business & Economic Statistics*, 34(4):574–589, 2016.

Pedregosa, F. Hyperparameter optimization with approximate gradient. In *Proceedings of the 33nd International Conference on Machine Learning, ICML*, 2016.

Petropoulos, F., Apiletti, D., Assimakopoulos, V., Babai, M. Z., Barrow, D. K., Taieb, S. B., Bergmeir, C., Bessa, R. J., Bijak, J., Boylan, J. E., et al. Forecasting: theory and practice. *International Journal of Forecasting*, 2022.

Ren, J., Feng, X., Liu, B., Pan, X., Fu, Y., Mai, L., and Yang, Y. TorchOpt: An Efficient Library for Differentiable Optimization. *OPT2022: 14th Annual Workshop on Optimization for Machine Learning*, 2022.

Ren, M., Zeng, W., Yang, B., and Urtasun, R. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pp. 4334–4343. PMLR, 2018.

Rossi, B. Advances in forecasting under instability. In *Chapter 21*, volume 2 of *Handbook of Economic Forecasting*, pp. 1203–1324. Elsevier, 2013.

Shimodaira, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

Shu, J., Xie, Q., Yi, L., Zhao, Q., Zhou, S., Xu, Z., and Meng, D. Meta-Weight-Net: Learning an Explicit Mapping for Sample Weighting. *Advances in Neural Information Processing Systems*, 32:1–23, 2019.

Sitzmann, V., Martel, J., Bergman, A., Lindell, D., and Wetzstein, G. Implicit neural representations with periodic activation functions. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 7462–7473. Curran Associates, Inc., 2020.

Stojanov, P., Gong, M., Carbonell, J., and Zhang, K. Low-dimensional density ratio estimation for covariate shift correction. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 3449–3458. PMLR, 2019.

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Wang, R., Maddix, D., Faloutsos, C., Wang, Y., and Yu, R. Bridging physics-based and data-driven modeling for learning dynamical systems. In *Learning for Dynamics and Control*, pp. 385–398. PMLR, 2021.

Wang, R., Dong, Y., Arik, S. O., and Yu, R. Koopman neural operator forecaster for time-series with temporal distributional shifts. In *The Eleventh International Conference on Learning Representations*, 2023.

Williams, R. J. and Zipser, D. Gradient-based learning algorithms for recurrent. *Backpropagation: Theory, Architectures, and Applications*, 433:17, 1995.

Woo, G., Liu, C., Sahoo, D., Kumar, A., and Hoi, S. Deeptime: Deep time-index meta-learning for non-stationary time-series forecasting. *arXiv preprint arXiv:2207.06046*, 2022.

Yang, J., Ji, K., and Liang, Y. Provably faster algorithms for bilevel optimization. *Advances in Neural Information Processing Systems*, 34:13670–13682, 2021.

Yusupova, A., Pavlidis, N. G., and Pavlidis, E. G. Dynamic linear models with adaptive discounting. *International Journal of Forecasting*, 2022. ISSN 01692070.

Zhang, T., Yamane, I., Lu, N., and Sugiyama, M. A One-Step Approach to Covariate Shift Adaptation. *SN Computer Science*, 2(4):65–80, 2021. ISSN 26618907.

Zliobaite, I., Pechenizkiy, M., and Gama, J. *An Overview of Concept Drift Applications*, pp. 91–114. Springer International Publishing, 2016.

A. Bi-level Optimisation Algorithm

Algorithm 1 Training Algorithm for GF

Require: Number of restarts R, number of weighting scheme updates U, number of model updates L, learning rates λ_1, λ_2, training set D_{train}, validation set D_{val}, batch sizes B_1, B_2, first validation index t^*, random initialisation distribution $\eta \sim H$, predictive model family f, forgetting mechanism α.

```
% Warm initialise using ERM solution
```
$\theta_{ERM} = \arg\min_{\theta} \sum_{(X_i,Y_i)\in D_{train}} L(f(X_i;\theta), Y_i)$.
```
% Start Bi-level optimisation
```
for $r = 1$ to R **do**
```
  % Re-initialise bi-level problem
```
  $\theta^* = \theta_{ERM}, \eta \sim H$

```
  % Upper objective loop
```
  for $u = 1$ to U **do**

```
    % Lower objective loop
```
    for $l = 1$ to L **do**
```
      % Sample training mini-batch
```
      $\{(X_i, Y_i, \tau_i)\}_{i=1}^{B_1} \sim D_{train}$
```
      % Compute mini-batch loss
```
      $g^L(\eta, \theta) = \sum_{i=1}^{B_1} \alpha(t^* - \tau_i; \eta) L(f(X_i;\theta), Y_i)$
```
      % Update the predictive model
```
      $\theta \leftarrow \theta - \lambda_1 \nabla_\theta g^L(\eta, \theta)$
    end for

```
    % Sample validation mini-batch
```
    $\{(X_i, Y_i)\}_{i=1}^{B_2} \sim D_{val}$
```
    % Compute validation loss
```
    $g^U(\eta, \theta) \leftarrow \frac{1}{B_2} \sum_{i=1}^{B_2} L(f(X_i;\theta), Y_i)$
```
    % Update forgetting mechanism
```
    $\eta \leftarrow \eta - \lambda_2 \nabla_\eta g^U(\eta, \theta)$
```
    % Checkpoint model parameters
```
    $\theta^* \leftarrow \arg\min_\theta \left\{ g^U(\eta, \theta) | \, \theta \in \{\theta^*, \theta\} \right\}$
  end for

```
  % Save theta solution
```
  $\theta_r \leftarrow \theta^*$
end for

```
% Ensemble solutions
```
$f(\cdot) \leftarrow \frac{1}{R} \sum_{r=1}^{R} f(\cdot\,; \theta_r)$

output $f(\cdot)$

B. Dataset Details

All time series except those found in the synthetic and FACTOR datasets were standardised to have mean zero and unit variance as a pre-processing step. An overview of all datasets can be found in Table 3. In the rest of this section of the Appendix, we describe each dataset. Links to download all datasets can be found in the project GitHub repository. All experiments in this paper were run using an Intel(R) Xeon(R) Gold 6240 CPU.

M4 This dataset was introduced in the M4 forecasting competition (Makridakis et al., 2020). The task is auto-regressive prediction of the next time series value. We use time series from four different frequencies (hourly, daily, monthly and yearly). For each frequency, we sample 100 time series from the M4 repository. This gives 4 sub-datasets M4-HOURLY, M4-DAILY, M4-MONTHLY and M4-YEARLY. The experimental result reported as M4 in Table 2 are the mean MASE results across the M4-HOURLY, M4-DAILY, M4-MONTHLY and M4-YEARLY experiment. This dataset was used as a non-stationary time series forecasting test-bed in Kim et al. (2021); Wang et al. (2023).

COVID This dataset consists of country-level daily Covid-19 death counts from the Johns Hopkins Coronavirus Resource Center. The data span January 2020 to February 2023. We drop territories with death counts that are constant in the last year of data. We then sample 100 time series from the repository. The task is auto-regressive prediction of the cumulative number of deaths. We use a validation set size of 10 months, and the last two months are test data. This dataset is used in (Wang et al., 2021; Arik et al., 2022) in the context of non-stationary time series forecasting.

SOLAR This dataset consists of 137 time series each representing the solar power production at different locations within Alabama during 2006. The data were re-sampled to hourly frequency. The task is auto-regressive prediction of solar power produced in the next hour, given the production in the previous 24 hours. We use two weeks of data for validation, and the final week as the test set. Masserano et al. (2022) evaluate their non-stationary time series method on this dataset.

FRED-MD This is a database of 137 macro-economic variables sampled at monthly frequency McCracken & Ng (2016). The task consists of auto-regressive prediction of each economic variable's next monthly value given its previous 6 monthly values. We use 12 years of data for validation, and the last 18 months for the test data.

ILI This single-time series dataset records the weekly proportion of patients with Influenza-Like-Illness (ILI). It is

provided by the United States' Centers for Disease Control and Prevention and spans 2002 to 2021. We use two years of data for validation, and two years for test. We consider the task of predicting next week's ILI proportion using its value in the previous four weeks (Woo et al., 2022; Liu et al., 2022).

IR This Bank for International Settlements dataset of interest rates records central bank policy rates across different countries. We re-sample each time series of rates to monthly frequency as typically rates do not vary much day-to-day. Target rates for each central bank differ across countries and time. Some of the time series include artefacts due to changing target rates: this poses an additional challenge to the distribution shift methods in terms of data corruption and noise. An overview of interest rate prediction tasks can be found in Petropoulos et al. (2022). The year in which data are first recorded for each time series varies by country: the shortest time series starts in 2005 while the longest starts in 1946. All time series end in 2022. We use 5 years of data as validation and 4 years for test.

FACTOR We consider the universe of NYSE equities spanning from 04-01-2000 to 31-12-2019 from Wharton's CRSP database. The data consists of daily closing prices from which we compute daily log-returns. We also compute the average daily dollar volume that is traded for each equity. We subset to the 50 equities that have the largest average daily dollar volume. Any missing prices are forward-filled prior to the calculation of log-returns. Factor returns were downloaded from Kenneth French's data repository.

The task is to model the risk of each equity in terms of three risk factors. Specifically, we linearly decompose each equity return $Y_t^{(i)}$, $i = 1, \ldots, 50$ in excess of the risk-free rate RF_t using a three-factor model parameterised by $\theta_t^{(i)} \in \mathbb{R}^4$:

$$Y_t^{(i)} - \mathrm{RF}_t = \theta_t^{(i)T} X_t \textit{ where } X_t := (1, \mathrm{MR}_t, \mathrm{SB}_t, \mathrm{HL}_t)^T$$

The three Fama-French factors correspond to market risk (MR), the risk factor related to size (SB) and the risk factor related to book-to-market equity (HL) (Fama & French, 1993). We use 150 days for validation and 150 days for testing.

C. Distribution Shift Method Details

Each distribution shift method is grid-searched over its hyper-parameters to minimise validation loss. We categorise each dataset as either Small, Medium or Large in Table 3. This categorisation is used to guide the hyper-parameter range of each distribution shift method by dataset. For example, on smaller datasets we perform more epochs of gradient descent to maintain an adequate number of total gradient descent steps. In the rest of this section of the Appendix, we provide details for each method's hyper-parameter grid and implementation details.

GF We use stochastic gradient descent with the AdamW (Loshchilov & Hutter, 2019) optimiser for both the upper and lower level problems. We perform $R = 5$ restarts of bi-level optimisation. We use a batch size of 64 for the upper level problem and a batch size of 256 for the lower level problem. Optimisation hyper-parameters that vary across datasets are displayed in Tables 4 and 5. In these tables, multiple comma-separated values indicate that a grid search was performed on the validation set.

UNIFORM The UNIFORM method fits predictive models using uniform weighting on all samples. Stochastic gradient optimisation is performed using AdamW with a learning rate of 0.01 and a batch size of 256. The number of epochs and range of weight decay hyper-parameters are identical to those we use in the lower-level optimisation for GF – these are indicated in Table 4 or 5.

WINDOW The WINDOW method fits each predictive model to a uniformly-weighted interval of the most recent samples. For each dataset size category, we use a batch size of 256 and a learning rate of 0.01. The window length and weight decay hyper-parameter ranges are shown in Table 6.

DIW This is the loss-value transformation version of the dynamic importance weighting algorithm of Fang et al. (2020). We use the implementation given by the original authors with two minor modifications. Firstly, we add gradient clipping (Goodfellow et al., 2016) to ensure that the optimisation is stable. Secondly, we re-compute the kernel width every 10 epochs rather than every epoch; we find that the kernel width does not change much during training and is costly to re-compute. For all data size categories, we use a batch size of 256 for the training data and 64 for the validation data. We clip gradients to have a maximum norm of 1. The variable hyper-parameters are the number of epochs and weight decay, which are the same as those we use to fit the lower level problems in GF, given in Table 4.

META is the Meta-Weight-Net algorithm of Shu et al. (2019). This is also a bi-level approach, but instead of selecting weights as a function of the time index, uses the loss values on the training set as input to a neural network which outputs new weights for each training datapoint. We use the original implementation given by Shu et al. (2019), copying over as many hyper-parameters as we could. As in the original work, we use a 1-layer meta-net with 32 hidden units, and use SGD with momentum, with a meta-learning rate of $1e^{-3}$ and momentum value 0.9. We used a batch size of 256 for both the upper and lower level problems, and

Table 3: A numerical overview of the real-world datasets and tasks considered in this paper. The columns are, in order from left to right, the number of lags used for auto-regressive prediction, the number of time series in the dataset, the average length of time series in a dataset, size category (which is used for hyper-parameter tuning as explained in Section C), validation set size and test set size.

DATASET	# LAGS	# TIME SERIES	AVG LENGTH	SIZE	VAL SIZE	TEST SIZE
FIXEDREGIME	1	20	3000	MEDIUM	100	25
RANDOMWALK	1	20	3000	MEDIUM	100	25
RANDOMREGIME	1	20	3000	MEDIUM	100	25
STATIONARY	1	20	3000	MEDIUM	100	25
COVID	14	100	1112	MEDIUM	305	70
SOLAR	24	100	8760	LARGE	366	168
FRED-MD	6	100	728	MEDIUM	120	18
ILI	4	1	1321	MEDIUM	104	104
IR	12	38	449	MEDIUM	72	48
FACTOR	N/A	50	4695	MEDIUM	150	150
M4-HOURLY	24	100	906	MEDIUM	366	48
M4-DAILY	7	100	2506	MEDIUM	100	14
M4-MONTHLY	12	100	234	SMALL	48	18
M4-YEARLY	3	100	35	SMALL	10	6

Table 4: Variable hyper-parameters for GF-EXP and GF-MIX.

SIZE	NUM EPOCHS		WEIGHT DECAY		LEARNING RATE	
	UPPER	LOWER	UPPER	LOWER	UPPER	LOWER
SMALL	10	2000	10^{-1}	$10^{1}, 10^{0}, 10^{-1}$	1	10^{-1}
MEDIUM	10	500	10^{-1}	$10^{-1}, 10^{-2}, 10^{-3}$	1	10^{-1}
LARGE	3	100	10^{-1}	$10^{-2}, 10^{-3}, 10^{-4}$	1	10^{-1}

Table 5: Variable hyper-parameters for GF-MLP and GF-MLP2.

SIZE	NUM EPOCHS		WEIGHT DECAY		LEARNING RATE	
	UPPER	LOWER	UPPER	LOWER	UPPER	LOWER
SMALL	15	2000	1	$10^{1}, 10^{0}, 10^{-1}$	10^{-2}	10^{-2}
MEDIUM	15	500	1	$10^{-1}, 10^{-2}, 10^{-3}$	10^{-2}	10^{-2}
LARGE	3	100	1	$10^{-2}, 10^{-3}, 10^{-4}$	10^{-2}	10^{-2}

Table 6: Variable Hyper-Parameters for WINDOW

SIZE	NUM EPOCHS	WEIGHT DECAY	WINDOW LENGTH
SMALL	2000	$10^{1}, 10^{0}, 10^{-1}$	100, 200, 300
MEDIUM	500	$10^{-1}, 10^{-2}, 10^{-3}$	1600, 3200, 5000
LARGE	100	$10^{-2}, 10^{-3}, 10^{-4}$	10000, 20000, 30000

used the same number of epochs and weight decay values as were used for GF (shown in Table 4).

DBF is an iterative two-stage procedure that iteratively fits the predictive model and sample weights in the ERM problem (see Kuznetsov & Mohri (2020), Section 7, Equation 14). Their procedure is run for 5 iterations. The look-back parameter, s, corresponding to the window in which the *discrepancy*, a measure of distribution shift defined in Kuznetsov & Mohri (2020), is thought to be bounded is set to the value used in the original paper $s = 20$. The DBF algorithm has two penalisation hyper-parameters λ_1 and λ_2, which correspond to regularisation on the sample weights and model parameters. We grid search both of these parameters over $[0.1, 0.01, 0.001]$.

D. Predictive Model Details

On all real-world datasets, we apply RevIN (Kim et al., 2021) normalisation to the model inputs and de-normalisation to the outputs. The use of instance-wise normalisation has been shown to be important for modelling non-stationary time series (Kim et al., 2021).

LINEAR This is a standard multivariate linear regression. Note we do not apply RevIN normalisation on the FACTOR modelling dataset, as the problem is cross-sectional regression: the model inputs are contemporaneous returns.

MLP We use an MLP with 1 hidden layer, consisting of 32 units and ReLU activation functions.

WAVENET We use the discriminative version of the WAVENET model (Borovykh et al., 2017; van den Oord et al., 2016). We use a kernel size of 2, with two output channels for each convolutional layer. We automatically select the number of layers so that the receptive field size is approximately equal to the number of time series lags used for each task (see Table 3 for a list of these lag numbers).

TF-ENC The TF-ENC model uses a transformer encoder with a sinusoidal positional encoding as described in Vaswani et al. (2017). We apply a linear layer to the embedding of the final element in the sequence to generate a forecast for the next time step. We use a single transformer layer and attention head which has 16 hidden units in the self-attention layer and 32 in the subsequent MLP feed-forward layer.

A note on using GF with an RNN predictive model Gradient-based learning of the forgetting mechanism interacts poorly with Recurrent Neural Networks (RNNs) models trained using back-propagation through time. RNNs are known to be unstable when trained using backpropagation through time (Goodfellow et al., 2016). This instability is magnified when computing the higher-order derivatives used for hyperparameter updates (Equation (3)). In experiments not shown in this work, we find that this problem is exacerbated as the number of time series lags used for forecasting increases. Future work to make RNN models compatible with gradient-based hyperparameter learning could use truncated back-propagation through time to mitigate training instability (Williams & Zipser, 1995).

E. Convergence

We can frame our bi-level optimisation method as an instance of the HOAG algorithm which is introduced in Pedregosa (2016). Thus, we benefit from the same convergence guarantees described in their work. We state sufficient conditions for convergence in this Appendix. These three conditions are:

1. The gradient ∇g^U with respect to θ, η of the upper-level objective and the Hessian $\nabla^2 g^L$ with respect to θ, η of the lower-level objective are Lipschitz continuous.

2. The Hessian of the lower-level objective with respect to θ, $\nabla^2_\theta g^L$, is invertible at $(\eta, \hat{\theta}(\eta))$ for all $\eta \in E$.

3. The domain of η is convex, non-empty and compact.

The HOAG algorithm consists of the following steps for gradient-based bi-level optimisation. For each iteration $k = 1, \ldots, n$,

1. Solve g^L in θ_k to tolerance $\epsilon_k^{(1)}$. That is, find a solution $\hat{\theta}(\eta_k) \in \underset{\theta}{\operatorname{argmin}}\, g^L(\eta, \theta)$, so that

$$||\hat{\theta}(\eta_k) - \theta_k|| \leq \epsilon_k^{(1)}.$$

2. Solve the linear system $\nabla^2_\theta g^L(\eta_k, \theta_k) q_k = \nabla_\theta g^U(\eta_k, \theta_k)$ to tolerance $\epsilon_k^{(2)}$; i.e. such that

$$||\nabla^2_\theta g^L(\eta_k, \theta_k) q_k - \nabla_\theta g^U(\eta_k, \theta_k)|| \leq \epsilon_k^{(2)}.$$

3. Define the approximate gradient p_k as

$$p_k = \nabla_\eta g^U(\eta_k, \theta_k) - \nabla^2_{\eta,\theta} g^L(\eta_k, \theta_k) q_k.$$

4. Update the weighting scheme parameters as

$$\eta_{k+1} = \eta_k - \lambda p_k.$$

Defining the tolerance sequence as $\epsilon_k = \max\{\epsilon_k^{(1)}, \epsilon_k^{(2)}\}$, the convergence result (Theorem 2, Pedregosa (2016)) states that if the tolerance sequence $(\epsilon_i)_{i=1}^{\infty}$ satisfies

$$\sum_{i=1}^{\infty} \epsilon_i < \infty$$

then

$$\lim_{k \to \infty} ||\nabla g^U(\eta_k, \hat{\theta}(\eta_k))|| = 0$$

i.e. the approximate iterates converge towards a stationary point of the upper level problem.

These convergence conditions are satisfied in our method by the combination of the LINEAR predictive model with the GF-EXP forgetting mechanism, although our experimental results suggest the method still performs well when the assumptions do not necessarily hold. Similar convergence results when using *stochastic* gradient descent on the upper and lower level problems can be achieved with technical assumptions on the random sampling behaviour, see Ji et al. (2021); Hong et al. (2023).

F. A Heuristic Derivation of the Forgetting Mechanism Form in a Linear Model with Random Walk Parameter Drift

Using the importance sampling principle (Shimodaira, 2000), we can obtain an unbiased estimator of the risk R_t at any test set point indexed by t, using

$$\hat{R}_{t^*}(\theta) = \frac{1}{t^* - 1} \sum_{\tau=1}^{t^*-1} \frac{\pi_t(y_\tau|x_\tau)}{\pi_\tau(y_\tau|x_\tau)} L(f(x_\tau; \theta), y_\tau) \quad (8)$$

where $\pi(\cdot)$ is defined in Section 2. The forgetting mechanism approximates the ratio $\frac{\pi_t(y_\tau|x_\tau)}{\pi_\tau(y_\tau|x_\tau)}$ using a parametric

function of τ. In this section, we provide some intuition for a heuristic form of the forgetting mechanism using the data generating process of Yusupova et al. (2022).

We suppose that the true data generating process $Y_t|X_t = x$ takes the form of a linear random walk model,

$$Y_t = \beta_t x_t + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, 1) \tag{9}$$

with $\beta_t = \beta_{t-1} + \eta_t$, where $\eta_t \sim \mathcal{N}(0, \sigma_\eta^2)$. In this case, the log-importance ratio is given by the following expression, up to a constant in x, y:

$$\log\left(\frac{\pi_t(y|x)}{\pi_\tau(y|x)}\right) = -\frac{1}{2}(y - \beta_t x)^2 + \frac{1}{2}(y - \beta_\tau x)^2 \tag{10}$$

$$= y(\beta_t - \beta_\tau)x - \frac{1}{2}(\beta_t x)^2 + \frac{1}{2}(\beta_\tau x)^2. \tag{11}$$

Notice that $\beta_t - \beta_\tau = \sum_{i=0}^{t-\tau} \eta_i = O_p(|t - \tau|)$. Let $k = t - \tau$ and define $S_k := \sum_{i=0}^{k} \eta_i$. Then $S_k = O_p(k)$.

Using the decomposition $\beta_t = \beta_\tau + S_k$ we can re-write Equation (11) as

$$y(S_k)x - \frac{1}{2}(\beta_\tau x + S_k x)^2 + \frac{1}{2}(\beta_\tau x)^2. \tag{12}$$

Expanding and collecting terms gives the form

$$(y - \beta_\tau x)(S_k x) - \frac{1}{2}(S_k x)^2 = O_p(k + k^2). \tag{13}$$

Hence the log-likelihood ratio takes the approximate functional form

$$\log\left(\frac{\pi_t(y|x)}{\pi_\tau(y|x)}\right) = \alpha_1 k + \alpha_2 k^2 \implies \frac{\pi_t(y|x)}{\pi_\tau(y|x)} = e^{\alpha_1 k + \alpha_2 k^2}. \tag{14}$$

This suggests a form for the forgetting mechanism given by the exponential of a quadratic function of the lag to a time point. The forgetting mechanism GF-MIX builds on this heuristic by incorporating a further power-decay term while GF-EXP studies the case of a single exponential decay. Here α_1 and α_2 are constants that depend on the true regression function (in particular the noise variance σ_η^2 and the true regression parameter β_τ). In this example, they correspond to the hyper-parameters η of the forgetting mechanism.

G. Figure of inferred sample weights for the FACTOR experiment

In Figure 4, we display the inferred sample weights for the time series of IBM stock returns in the FACTOR experiment. For this example, we see that all inferred forgetting mechanisms tend to place more weight on more recent data. This illustrates that using the inductive bias that "more recent samples are more relevant" can help on real world prediction tasks.

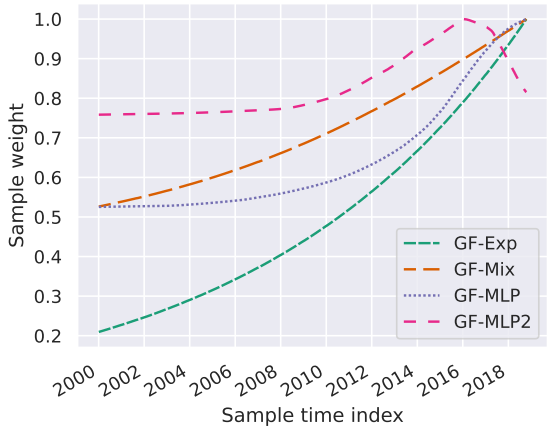


Figure 4: Inferred sample weights by each forgetting mechanism for IBM on the FACTOR dataset.

H. Full Experimental Results

We display the full real-data experimental results in Table 7. Relative to UNIFORM, the mean reduction in test loss for the predictive models LINEAR, MLP and WAVENET averaged across the GF methods and datasets is considerably high at 12.5%. On the other hand, the same figure for the TF-ENC model is a moderate 1.1%. We conjecture that the performance of GF on TF-ENC could be improved through a more careful tuning of the Neumann Series approximation used in the hyper-gradient calculations.

On the datasets with the smallest validation set sizes (10 for M4-YEARLY and 48 for M4-MONTHLY), the performance of GF relative to other methods is not as strong. Indeed, from Figure 3 we see that the GF methods under-perform when the validation set size is very small. In these cases, using the simpler WINDOW approach with a limited choice of window size may work better.

Table 7: This table reports the average test MASE for each combination of dataset and predictive model by distribution shift method (shown across the columns). Results for the FACTOR dataset are reported in MSE $\times 10^4$. For each dataset and predictive model pair, the loss of the best-performing distribution shift method is shown in bold. The META implementation is not compatible with TF-ENC so these results are missing.

DATASET	MODEL	UNIFORM	META	DIW	WINDOW	GF-EXP	GF-MIX	GF-MLP	GF-MLP2
FACTOR	LINEAR	1.46	1.83	2.11	1.44	1.31	**1.30**	1.31	**1.30**
COVID	LINEAR	1.61	1.37	1.76	0.46	0.35	0.35	0.35	**0.34**
	MLP	1.89	1.85	2.43	**0.53**	0.61	0.61	0.66	0.69
	TF-ENC	1.01	–	1.87	**0.77**	0.95	0.83	0.88	0.88
	WAVENET	1.26	2.45	1.60	1.00	0.87	**0.85**	1.17	1.02
FRED-MD	LINEAR	1.15	1.64	1.77	1.15	**1.11**	**1.11**	1.13	1.12
	MLP	1.16	2.14	2.38	1.17	1.14	**1.13**	1.16	1.14
	TF-ENC	1.23	–	2.31	1.25	1.26	1.36	**1.21**	1.24
	WAVENET	1.46	1.70	1.77	1.51	**1.29**	1.32	1.34	1.30
ILI	LINEAR	0.75	0.78	0.84	0.76	**0.73**	**0.73**	**0.73**	**0.73**
	MLP	0.78	0.76	0.77	0.78	0.78	0.78	**0.74**	0.77
	TF-ENC	0.81	–	0.86	0.84	**0.80**	0.85	0.82	0.84
	WAVENET	0.82	**0.76**	0.87	0.87	0.77	0.79	0.77	0.79
IR	LINEAR	0.64	0.63	0.79	0.64	0.63	0.62	**0.61**	**0.61**
	MLP	0.69	0.72	0.98	0.69	**0.66**	**0.66**	0.72	0.69
	TF-ENC	0.86	–	1.53	**0.73**	0.84	0.87	0.90	0.84
	WAVENET	1.15	1.45	1.94	1.17	**1.06**	**1.06**	1.10	1.09
M4	LINEAR	1.05	1.14	1.36	1.02	1.00	0.99	1.02	**0.98**
	MLP	1.08	1.24	1.52	1.07	1.07	1.07	**1.06**	1.08
	TF-ENC	1.33	–	1.70	**1.13**	1.33	1.58	1.18	1.23
	WAVENET	1.25	1.46	1.51	1.22	**1.21**	1.23	**1.21**	1.22
M4-DAILY	LINEAR	**1.08**	1.12	1.32	**1.08**	1.09	**1.08**	1.09	**1.08**
	MLP	1.10	1.17	1.47	1.10	1.10	1.11	**1.09**	1.10
	TF-ENC	1.13	–	1.67	1.14	**1.09**	1.14	1.11	1.11
	WAVENET	1.35	1.53	1.72	1.36	1.35	1.36	1.35	**1.34**
M4-HOURLY	LINEAR	0.41	0.42	0.47	0.40	0.38	**0.37**	**0.37**	**0.37**
	MLP	0.35	0.38	0.46	0.35	**0.34**	**0.34**	**0.34**	0.35
	TF-ENC	0.37	–	0.71	0.36	**0.35**	**0.35**	0.36	**0.35**
	WAVENET	0.63	0.64	0.75	0.60	0.54	0.54	0.55	**0.53**
M4-MONTHLY	LINEAR	1.10	1.36	1.35	1.07	1.06	**1.02**	1.06	1.04
	MLP	1.19	1.72	1.62	**1.15**	1.16	1.19	1.17	1.16
	TF-ENC	1.19	–	1.87	1.18	**1.15**	1.27	1.16	1.18
	WAVENET	1.47	1.92	1.88	**1.42**	1.45	1.46	**1.42**	1.45
M4-YEARLY	LINEAR	1.63	1.65	2.28	1.53	1.49	1.49	1.58	**1.45**
	MLP	1.67	1.69	2.53	1.66	1.70	1.67	**1.64**	1.71
	TF-ENC	2.62	–	2.53	**1.82**	2.73	3.54	2.10	2.30
	WAVENET	1.54	1.73	1.69	**1.50**	1.53	1.55	**1.50**	1.55
SOLAR	LINEAR	0.61	0.61	0.66	0.60	**0.59**	0.60	**0.59**	**0.59**
	MLP	0.47	0.48	0.56	0.46	0.46	0.47	0.46	**0.45**
	TF-ENC	**0.48**	–	0.99	**0.48**	0.49	**0.48**	0.49	0.49
	WAVENET	1.01	1.00	1.17	0.94	**0.83**	0.84	0.88	0.84

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

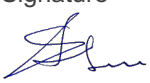
Title of Paper	Time Series Prediction under Distribution Shift using Differentiable Forgetting
Publication Status	☒Published ☐Accepted for Publication ☐Submitted for Publication ☐Unpublished and unsubmitted work written in a manuscript style
Publication Details	A preliminary version of this work was presented at the 2022 ICML Workshop on Principles of Distribution Shift.

Student Confirmation

Student Name:	Stefanos Bennett		
Contribution to the Paper	The idea of learning a forgetting mechanism with gradient-based bi-level optimisation was conceived by SB. The initial codebase development and writing of the ICML workshop paper was joint work with DPhil student Jase Clarkson. The additional experimentation and writing of the full chapter were done by SB.		
Signature		Date	02/01/24

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Mihai Cucuringu		
Supervisor comments Stefanos made a substantial contribution to the publication, as indicated above.		
Signature 	Date	5 January 2024

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 4

Rethinking Neural Relational Inference for Granger Causal Discovery

NRI, which is the subject of study in Chapter 4, is a promising model for the goal of comprehensively modelling complex dependencies in financial time series. NRI takes a Granger causal approach towards modelling temporal interactions between time series. NRI aims to overcome the limitations of traditional linear Granger causal discovery techniques by accommodating inductive modelling and non-linear effects.

In Chapter 2, we observe that our prior expectations concerning temporal interactions are not always satisfied. When comparing the inferred US equity lead-lag clusters with an industry clustering, we find that, broadly, the inferred clusters do not correspond to an industry clustering. However, we do observe that certain leading clusters are associated with the finance and energy sectors: this agrees with our prior expectations concerning leading sectors based on previous empirical finance studies. Hence, we see that it is important to take into account, but not be constrained to, prior expectations concerning temporal interactions in financial systems. The NRI model provides a principled approach for incorporating prior knowledge into the modelling of temporal interactions. This is a useful step towards comprehensively modelling complex temporal dependencies in multivariate systems.

Rethinking Neural Relational Inference for Granger Causal Discovery

Stefanos Bennett*
University of Oxford and
The Alan Turing Institute
stefanos.bennett@stats.ox.ac.uk

Mihai Cucuringu
University of Oxford and
The Alan Turing Institute
mihai.cucuringu@stats.ox.ac.uk

Gesine Reinert
University of Oxford and
The Alan Turing Institute
reinert@stats.ox.ac.uk

Rose Yu
University of California, San Diego
roseyu@ucsd.edu

Abstract

Granger causal discovery aims to infer the underlying Granger causal relationships between pairs of variables in a multivariate time series system. Recent work has proposed using Neural Relational Inference (NRI) [Kipf et al., 2018] – a latent graph inference model – for Granger causal discovery. However, the conditions under which NRI succeeds in recovering the true Granger causal graph remain unknown. In this work, we show how the mean-field approximation inherent in NRI can hinder its ability to recover the Granger causal structure in multivariate time series. We illustrate this point theoretically and experimentally using a linear vector autoregressive model – an important benchmark in economic and financial studies. Further, we examine the performance of previous NRI adaptations and propose a new NRI extension, based on a continuous relaxation for sampling dependent binary variables, for the task of Granger causal discovery.

1 Introduction

Granger causal discovery is a widely studied problem with real-world applications in a number of fields such as neuroscience [Sporns, 2016], genetics [Fujita et al., 2010] and finance [Campbell et al., 1998]. Given an observed multivariate time series data set, Granger causal discovery aims to infer the underlying Granger causal relationships between pairs of time series. Examples of the application of Granger causal discovery include the estimation of the influence of different brain regions based on fMRI data measuring regional activity over time [Seth et al., 2015], the estimation of gene regulatory networks [Michailidis and d'Alché Buc, 2013], or to understand cross-country variations of electricity prices [Castagneto-Gissey et al., 2014]. In recent years, deep learning methods [Tank et al., 2021, Löwe et al., 2022] have been proposed for Granger causal discovery which aim to provide more flexibility over traditional approaches [Granger, 1969] for Granger causal discovery. In this work, we examine the approach of Löwe et al. [2022] on a simple data generating process.

Löwe et al. [2022] propose a method named Amortized Causal Discovery, which infers Granger causal relationships using Neural Relational Inference (NRI) [Kipf et al., 2018]. The Granger causal structure of a multivariate time series system can be viewed as a directed graph in which nodes represent time series and edges represent Granger causal relations. NRI is a graph-based variational autoencoder that infers latent edge relations. This model is an exciting proposal for Granger causal

*Denotes corresponding author; authors are listed in alphabetical order

discovery as it is fully inductive, which means that it can be applied across several multivariate time series samples, and achieves competitive performance on several benchmark data sets.

However, the conditions under which NRI succeeds in recovering the Granger causal graph underlying a multivariate time series system are not known. This creates a hurdle to its use on real-world data as there are no theoretical guarantees or strong experimental results to suggest that it can recover the true causal structure in any given application.

In this work, we examine NRI in the context of a vector autoregressive (VAR) data generating process with graph structure. VAR is a synthetic benchmark that is commonly used in financial and economic domains. Since Granger causal discovery is a widely studied question and certain financial and economic autoregressive processes can be modeled using graph-based approaches [Knight et al., 2020], NRI is a potentially attractive method for the domains of economics and finance. Therefore, it is of interest to understand the performance of NRI on a simple VAR benchmark, prior to applying it to real-world data. In contrast to the highly structured biological and physical systems to which NRI has been previously applied [Liu et al., 2023, Zhu et al., 2022, Löwe et al., 2022], economic and financial data often have lower signal-to-noise ratios. Thus, the study of the performance of NRI under the VAR data generating process, which is more strongly characteristic of economic and financial data compared to existing NRI benchmark data sets [Kipf et al., 2018], may provide new insights into the performance of the model.

The remainder of this paper is structured as follows. We start by briefly describing the task of Granger causal discovery and the NRI approach that will be the object of study in Section 2. Then, in Section 3, we introduce the graph-based VAR benchmark. By comparing the true posterior distribution of the edge relations with the form of the GNN encoder used in NRI, we argue that the mean-field approximation inherent in NRI poses a fundamental limitation on the model for Granger causality discovery. We construct an indicator which we call a *difficulty indicator* that theoretically predicts the specific graph structures that NRI will struggle to infer in this simple VAR setting. In Section 4 this difficulty indicator is validated using synthetic data experiments. In Section 5 we propose an novel encoder for NRI that relaxes the mean-field assumption. This extension draws on a normalising flow method for sampling dependent binary variables which uses a continuous relaxation to permit gradient estimation. Using our difficulty indicator developed in Section 3, we construct difficult synthetic examples on which we expect that NRI will struggle. In Section 6 we test the performance of standard NRI, popular extensions of NRI, and our novel extension on the task of Granger causal discovery using these difficult synthetic examples. Section 7 provides a brief conclusion. Derivations of theoretical results and additional experiments, as well as details for the experiments, are deferred to the Appendix.

Our primary contributions can be summarized as follows:

1. We provide the first study to understand the conditions under which NRI can successfully recover the Granger causal structure of a multivariate time series system.
2. We propose an indicator that predicts when NRI will fail to recover the Granger causal structure on a linear graph autoregressive process.
3. We empirically validate our indicator using synthetic data experiments[2].
4. We highlight the limitations of existing extensions.
5. We propose an adaptation that shows a degree of progress in the task of Granger causal discovery.

2 Background

2.1 Granger Causality

Let $X_t^i \in \mathbb{R}$ denote the value of time series $i = 1, \ldots, N$ at time t. Then time series X^i Granger causes [Eichler, 2012] time series X^j if, for some t, and some non-empty set S, it holds that

$$\mathbb{P}[X_{t+1}^j \in S | \mathcal{I}(t)] \neq \mathbb{P}[X_{t+1}^j \in S | \mathcal{I}_{-X^i}(t)]. \tag{1}$$

[2]Code used in experiments can be found here `https://anonymous.4open.science/r/rethinking_nri_granger-076F/README.md`

The sets $\mathcal{I}(t)$ and $\mathcal{I}_{-X^i}(t)$, respectively, denote all information available as of time t and all information available as of time t excluding time series X^i. Granger causal discovery aims to infer the Granger causal relation between each pair of time series in a multivariate time series data set. While an inferred Granger causal relation is not necessarily indicative of a true causal relation [Maziarz, 2015], it remains a popular framework for understanding temporal relations in multivariate dynamical systems.

In the case of linear VAR modeling, the Granger causal relationship between two random variables can be tested through the hypothesis that the coefficients relating the lagged values of time series i to the current value of time series j are jointly significantly different from 0. Granger causal discovery can be accomplished when N is small through an exhaustive hypothesis test search on the coefficients in the VAR model [Arnold et al., 2007].

2.2 Neural Relational Inference for Granger causal discovery

Löwe et al. [2022] propose to use NRI [Kipf et al., 2018], an encoder-decoder latent variable model, to infer Granger causal relations. NRI is an autoregressive model for multivariate time series with latent graph structures. For each observed multivariate time series input, the latent graph structure is encoded as the set of categorical relations between each pair of time series. The NRI encoder is trained to map multivariate time series inputs to distributions over latent graph structures. Conditional on the inferred latent graph structures, the NRI decoder aims to predict the next-step values of each multivariate time series sample. The model can be construed as a variational auto-encoder, and is trained using amortised Variational Expectation-Maximisation [Kipf et al., 2018].

The NRI encoder takes the form of a Graph Neural Network (GNN), which is a neural network that processes data with a graph structure [Liu and Zhou, 2022]. The NRI encoder learns to approximate the posterior distribution over latent graph relations; it accomplishes this by propagating information across a fully connected graph in which each node corresponds to a variable in the multivariate system and a node feature embeds its respective variable's time series. The NRI encoder models the posterior probability for $\mathbf{z}_{ij}$, the type (category) of edge $i \to j$, with

$$\boldsymbol{\psi}_{ij} = f_{\text{enc}}(\mathbf{X}_{1:T})_{ij}, \quad q(\mathbf{z}_{ij}|\mathbf{X}_{1:T}) = \text{Softmax}(\boldsymbol{\psi}_{ij}/\tau) \tag{2}$$

where f_{enc} is a GNN encoder, detailed in equation (3) below, and τ is a temperature parameter for the Softmax activation function. The variables $\boldsymbol{\psi}_{ij}$ and $\mathbf{z}_{ij}$ are K-dimensional, where K is the number of edge categories chosen by the user. The variable $\mathbf{z}_{ij}$ is a one-hot vector: if the k-th element of $\mathbf{z}_{ij}$, which is denoted by $z_{ij,k}$, is equal to 1, then this indicates an inferred category of k for edge $i \to j$. We write $\mathbf{X}_{1:T}$ to denote the dataset $\left(X_t^i\right)_{t=1,\ldots,T}^{i=1,\ldots,N}$, and we define $\mathbf{X}_{1:T}^i := (X_1^i, \ldots, X_T^i)^T \in \mathbb{R}^T$. In the general NRI formulation, the variable X_t^i may also be multivariate, $\mathbf{X}_t^i \in \mathbb{R}^m$, $m \geq 1$.

Mathematically, the GNN encoder f_{enc} first embeds the time series $j = 1, \ldots, N$ using an MLP, f_{emb}, resulting in $\mathbf{h}_j^{(1)} = f_{\text{emb}}\left(\mathbf{X}_{1:T}^j\right)$. Then, these initial embeddings are propagated using the following message passing steps

$$\begin{aligned}
v \to e: \quad & \mathbf{h}_{(i,j)}^{(1)} = f_e^{(1)}\left(\left[\mathbf{h}_i^{(1)}, \mathbf{h}_j^{(1)}\right]\right), \\
e \to v: \quad & \mathbf{h}_j^{(2)} = f_v^{(1)}\left(\sum_{i:\, i \neq j} \mathbf{h}_{(i,j)}^{(1)}\right), \\
v \to e: \quad & \mathbf{h}_{(i,j)}^{(2)} = f_e^{(2)}\left(\left[\mathbf{h}_i^{(2)}, \mathbf{h}_j^{(2)}\right]\right).
\end{aligned} \tag{3}$$

The result of the last node-to-edge step gives the final embedding $\boldsymbol{\psi}_{ij} = \mathbf{h}_{(i,j)}^{(2)}$ prior to the softmax calculation. The functions $f_e^{(1)}$, $f_v^{(1)}$, $f_e^{(2)}$ in the equations expressed in equation (3) are distinct MLPs.

The NRI decoder – another GNN – models the multivariate time series conditional on the graph and latent edge relations returned by the encoder. Each edge relation type coincides with a unique message passing function in the decoder. The variant of NRI proposed in the Amortized Causal

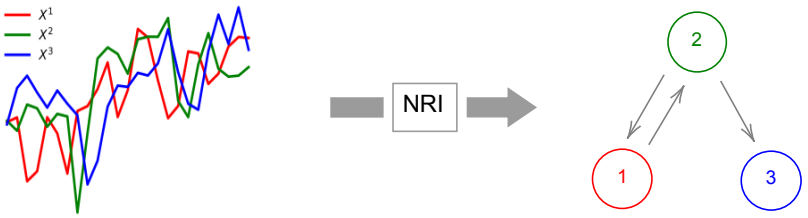


Figure 1: In the Amortized Causal Discovery method of Löwe et al. [2022], NRI is proposed as a Granger causal discovery method. NRI aims to infer the Granger causal graph (right) that corresponds to an observed multivariate time series (left).

Discovery method of Löwe et al. [2022] introduces a "zero" edge type: this edge type has its decoder message passing function hard-coded to return zero. An inferred zero-edge implies no Granger causation between two variables. Inferred edges that are not of a zero type imply a Granger causal relationship between the two variables. Apart from the addition of a single zero-edge type, Löwe et al. [2022] use the same neural architecture as in the original NRI work. In the remainder of the paper, we shall refer to NRI as the model variant which includes a zero-edge type.

Mathematically, conditional on the edges sampled by the encoder, the decoder generates the following prediction for time series j

$$\begin{aligned} v \to e: \quad & \tilde{\mathbf{h}}^t_{(i,j)} = \sum_{k=0}^{K-1} z_{ij,k} \tilde{f}^k_e \left(\left[\mathbf{X}^i_t, \mathbf{X}^j_t \right] \right), \\ e \to v: \quad & \boldsymbol{\mu}^j_{t+1} = \mathbf{x}^j_t + \tilde{f}_v \left(\sum_{i:\, i \neq j} \tilde{\mathbf{h}}^t_{(i,j)} \right), \\ & p\left(\mathbf{X}^j_{t+1} \mid \mathbf{X}_t, \mathbf{z} \right) = \mathcal{N}\left(\boldsymbol{\mu}^j_{t+1}, \sigma^2 \mathbf{I} \right). \end{aligned} \tag{4}$$

Here, σ^2 is a prior variance. In general, the functions $\tilde{f}^k_e$, $k = 0, \ldots, K-1$ and $\tilde{f}_v$ in equation (4) are given by neural networks. When the edge (i, j) is of type k, the message is $\tilde{\mathbf{h}}^t_{(i,j)} = \tilde{f}^k_e \left(\left[\mathbf{X}^i_t, \mathbf{X}^j_t \right] \right)$. By defining the message-passing function corresponding to edge type $k = 0$, $\tilde{f}^0_e$, to be the zero-function, no information can propagate between time series in the decoder whenever a zero-edge is inferred between them.

In contrast to existing methods, NRI has a fully inductive encoder for the Granger causal graph. This means that a trained model can be applied to an out-of-sample multivariate time series data set with a different Granger causal structure but the same shared dynamics conditional on the Granger causal graph. The inductive nature of the model means that it can leverage the information that is shared across multivariate time series samples. Its potential ability to learn across multivariate time series samples and its competitive performance on three synthetic data sets considered by Löwe et al. [2022] renders NRI an exciting model for Granger causal discovery.

3 Theoretical limitations of NRI for Granger causal discovery

Despite its empirical success on structured data sets in the domain of physics and biology [Löwe et al., 2022], the theoretical properties of NRI on the task of Granger causal discovery are not fully understood. In this section, we examine this issue by investigating the performance of NRI on a simple synthetic data generating process, motivated by the Generalized Network Autoregressive Process (GNAR) of Knight et al. [2020]. Under this data generating process, the true Granger causal

graph is known and the exact posterior distribution under the true likelihood and a Bernoulli prior can be derived analytically. This allows us to analytically compare the true posterior distribution to the NRI approximation.

Consider the following generative process for a multivariate time series of size T with N variables

$$\mathbf{X}_t = cA^\intercal \mathbf{X}_{t-1} + \boldsymbol{\epsilon}_t, \tag{5}$$

where $\mathbf{X}_t := \left(X_t^1, \ldots, X_t^N\right)^\intercal$, $\boldsymbol{\epsilon}_t \in \mathbb{R}^N$, $X_0^i := 0$ and $\epsilon_t^i \sim N(0,1)$ independently for $t = 1, \ldots, T$, $i = 1, \ldots, N$. The matrix $A \in \{0,1\}^{N \times N}$ is the adjacency matrix corresponding to the Granger causal graph, and c is a constant that determines the signal-to-noise ratio of the problem. The adjacency matrix is prespecified for each experiment. If $A_{ij} = 1$ then time series i Granger causes time series j; otherwise, if $A_{ij} = 0$, there is no causal relation. We set the diagonal elements of A to 1 so that there is an autoregressive dependence for each time series.

Under a Bayesian model in which the entries of A are distributed independently at random with uniform probability p of being 1 under the prior, the log-posterior distribution of entry A_{ij} conditional on the observed data set and the other entries of A is given, up to a term which is a constant in A_{ij}, by

$$\begin{aligned}
&\log \mathbb{P}\left(A_{ij} | \mathbf{X}_{1:T}, (A_{kl})_{(k,l)\neq(i,j)}\right) = \\
&\frac{c}{2} A_{ij} \Bigg[2(\mathcal{L}\mathbf{X}_{1:T}^i) \cdot \mathbf{X}_{1:T}^j - c(\mathcal{L}\mathbf{X}_{1:T}^j) \cdot (\mathcal{L}\mathbf{X}_{1:T}^i) \\
&- c(\mathcal{L}\mathbf{X}_{1:T}^i) \cdot (\mathcal{L}\mathbf{X}_{1:T}^i) - c \sum_{k \neq i,j} A_{kj} (\mathcal{L}\mathbf{X}_{1:T}^i) \cdot (\mathcal{L}\mathbf{X}_{1:T}^k) \Bigg] \\
&+ [A_{ij} \log p + (1 - A_{ij}) \log(1-p)]
\end{aligned} \tag{6}$$

where $\mathbf{X}_{1:T}^i := (X_1^i, \ldots, X_T^i)^\intercal \in \mathbb{R}^T$, $\mathcal{L}$ denotes the lag operator, so that $\mathcal{L}\mathbf{X}_{1:T}^i := (X_0^i, X_1^i, \ldots, X_{T-1}^i)^\intercal \in \mathbb{R}^T$ for all $i = 1, \ldots, N$, and $\cdot$ denotes the vector dot product. The derivation of equation (6) can be found in Appendix A.

Under the NRI model, the posterior is approximated using a mean-field approximation where the marginal posterior probability for A_{ij} is given by the result of the GNN encoder, $q(\mathbf{z}_{ij} | \mathbf{X}_{1:T})$ in equation (2). Here, there are only two categories for each edge – present or absent. The two-dimensional variable $\mathbf{z}_{ij}$ corresponds to A_{ij} in this case, as the ground truth data generating process has $K = 2$ edge types. We make the following remark, by the definition of the mean-field assumption.

Remark 1. *In the mean-field approximating model of NRI, the posterior distribution for each edge indicator, A_{ij}, is **independent** of the other edges.*

However, by examining the true posterior distribution given in equation (6), we note:

Remark 2. *Under the true likelihood and a Bernoulli edge prior, the exact posterior distribution over the edge indicator A_{ij} is potentially **dependent** on the edges $A_{kj}, k \in \{1, \ldots, N\} \setminus \{i, j\}$.*

Indeed, equation (6) indicates that there will be significant negative posterior correlation between A_{ij} and A_{kj} whenever $(\mathcal{L}\mathbf{X}_{1:T}^i) \cdot (\mathcal{L}\mathbf{X}_{1:T}^k)$ is large. Intuitively, if variables i and k are positively correlated, then the true posterior distribution will place negative correlation on the edges originating from either node and having a common target node: if variable k predicts j, then it is less likely, a-posteriori, that i also predicts j since k and i partially explain the same variance in j.

Since the NRI encoder is unable to capture such posterior dependence between edges, we expect it will incorrectly classify the edge $i \to j$ whenever X^k and X^i have significant positive correlation and there is an edge $k \to j$ but there is no edge $i \to j$ in the true underlying Granger graph. These cases can be mathematically characterised, as follows.

Algebraic formulation In order to validate this hypothesis concerning the limitations of NRI to estimate the Granger relations in certain graph structures, we construct a "difficulty indicator" for each edge.

Definition 1. *The difficulty indicator $D_{ij} \in [0, 1]$ for each edge is defined to be*

$$D_{ij} = \frac{|C_{ij}|}{|C_{ij}|+|M_{ij}|}(1 - A^*_{ij}), \tag{7}$$

where

$$C_{ij} = c \sum_{k\in\{1,\dots,N\}\setminus\{i,j\}} A^*_{kj}(\mathcal{L}\mathbf{X}^i_{1:T}) \cdot (\mathcal{L}\mathbf{X}^k_{1:T}),$$
$$M_{ij} = 2(\mathcal{L}\mathbf{X}^i_{1:T}) \cdot \mathbf{X}^j_{1:T} - c(\mathcal{L}\mathbf{X}^j_{1:T}) \cdot (\mathcal{L}\mathbf{X}^i_{1:T}) - c(\mathcal{L}\mathbf{X}^i_{1:T}) \cdot (\mathcal{L}\mathbf{X}^i_{1:T}).$$

Here, A^* denotes the ground truth adjacency matrix. We set $0/0 := 0$. While D_{ij} is dependent on T, $D_{ij} = D_{ij}(T)$, we suppress this dependence in our notation.

The difficulty indicator gives the size $|C_{ij}|$ of the true posterior dependence of A_{ij} on other edges A_{kj} relative to the size $|M_{ij}|$ of the component of the posterior edge probability in equation (6) which does not depend on other edges. Note that we ignore the prior term $A_{ij} \log p + (1 - A_{ij}) \log(1 - p)$ in equation (6) as this will become insignificant as T increases: in probability, the dot-product terms from the likelihood are $O(T)$ in T whereas the prior is $O(1)$ in T.

The difficulty indicators D_{ij} can be estimated from observations, and from the true Granger causal structure $A^\intercal$, by replacing the contemporaneous and lagged covariances in the definitions of C_{ij} and D_{ij} (given below equation (7)) with their empirical counterparts. Alternatively, the edges with non-zero difficulty indicators can be identified by examining the connectivity structure of the Granger causal graph, $A^\intercal$.

Topological interpretation We propose an interpretation for the posterior dependence structure in equation (6) based on the connectivity structure of the Granger causal graph. This is provided in Lemma 1, the proof of which can be found in Appendix B. The result of Lemma 1 provides intuition for when the numerator of the difficulty indicator is non-zero.

Lemma 1. *Assume that the VAR process $(\mathbf{X}_t)_t$ in (5) is covariance-stationary. Then for any distinct $i, j \in \{1, \dots, N\}$, the expectation $\mathbb{E}[C_{ij}(1 - A^*_{ij})]$ is non-zero if and only if the following conditions both hold:*

1. *j has a parent $k \neq i, j$ such that k has a common ancestor with i. Mathematically, that is to say $\exists k, m \in \{1, \dots, N\} \setminus \{i, j\}$ such that $k \to j$, with $m \to \dots \to k$ and $m \to \dots \to i$.*
2. *i is not a parent of j. Mathematically, $A^*_{ij} = 0$*

Based on the argument given in this section, the encoder is unable to capture the negative posterior correlation of A_{ij} and A_{kj} when $D_{ij} > 0$. It will tend to overestimate the probability of $A_{ij} = 1$ and therefore misclassify edge $i \to j$. In summary, we expect that:

Hypothesis 1. *NRI will tend to misclassify the Granger Causal relations between time series in a data set whenever the data set has edges with large difficulty indicator values, D_{ij}, which indicates that the true posterior has relatively large posterior dependencies between edges.*

We test Hypothesis 1 using synthetic data in section 4.

4 Evaluating NRI for Granger causal discovery

4.1 Synthetic data generation

We generate multivariate time series data sets of size $T = 200$, $N = 3$ using equation (5). We consider three causal graph structures. These are illustrated in Figure 2.

We note that using a suite of t-tests on the estimated coefficients of an ordinary least squares fit of a VAR model with 1 lag to data sets of this size results in 100% classification accuracy for the entries in the VAR autoregressive matrix with a single multivariate time series sample of size $T = 1000$, $N = 3$. This illustrates that the task of causal graph discovery is solvable on these three graph structures. In addition, an exact MCMC inference procedure [Robert and Casella, 1999] can achieve perfect graph recovery in the cases shown.

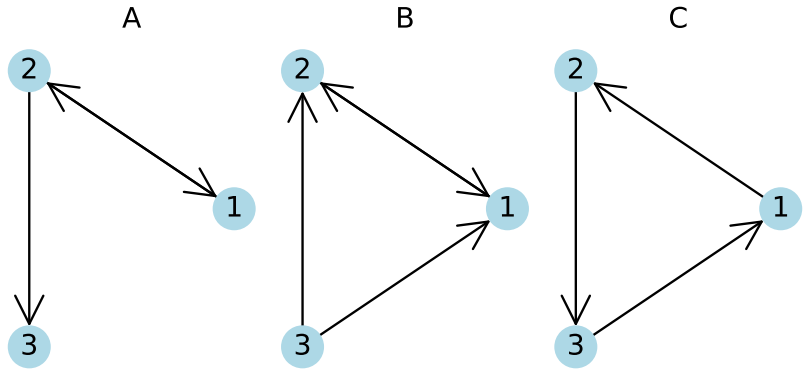


Figure 2: The three Granger causal graph structures used in the experiments in this section.

The learning paradigm of NRI uses samples of multivariate time series data sets [Löwe et al., 2022]. We match this learning paradigm and generate 1000 different training, validation and test multivariate time series for each graph structure. Details of the NRI implementation are similar to those of Löwe et al. [2022] and can be found in Appendix C. The training hyperparameters that were used across experiments are given in Appendix E.

The performance of NRI in recovering the Granger causal structure is evaluated using the classification accuracy of the encoder on the off-diagonal entries of the adjacency matrix on test multivariate time series data. The Area Under the Receiver Operating Curve (AUC) and the F1 harmonic mean of the recall and precision for the task of Granger Causal Recovery are reported for each method. In addition, we also record the predictive performance of each method using mean squared error (MSE) averaged across each time series variable. We note for reference that the Bayes rule predictor will have an MSE error of 1 (since the residual error has variance of 1 for each variable) on our examples.

4.2 NRI model variants

The following encoders are used in the experiments in this section:

- **RefMLP**: this is the standard MLP encoder used in Löwe et al. [2022], Kipf et al. [2018].
- **Unshared**: this is a transductive encoder that has a unique parameter vector for each edge which gives the log-probabilities of that edge being in either category (present or absent). Since it is a transductive rather than inductive encoder, it is not a "true" NRI model variant, however, we train it using the same variational inference procedure as NRI. We expect this model to perform best in experiments when there is a single Granger causal graph underlying the multivariate time series samples. However, the transductive approach cannot work if the training set is drawn from a data generating process consisting of a mixture of graphs.

The following decoders are used in the experiments in this section:

- **Linear**: this decoder consists of a single round of linear message passing. The message passing function applies a linear map (with no additive constant term) to each of the sender node features (the current time series value). This corresponds to setting $\tilde{f}_e^{(1)}\left(\left[\mathbf{X}_t^i, \mathbf{X}_t^j\right]\right) = x_t^i$ and $\boldsymbol{\mu}_{t+1}^j = \tilde{c}\left(\mathbf{X}_t^j + \sum_{i:\, i\neq j} \tilde{\mathbf{h}}_{(i,j)}^t\right)$ in equation (4) for a parameter $\tilde{c}$ to be inferred. Since this is the correctly specified decoder functional form for the VAR data generating process in equation (5), any potential shortcomings of the NRI model on the experimental benchmarks cannot be attributed to decoder misspecification.
- **GNN**: We have also validated our hypothesis using the GNN decoder used in the standard NRI formulation [Löwe et al., 2022]. These results are shown in Table 5 of Appendix G.

The value of the autoregressive constant c in equation (5) is chosen so that, for each graph used, the largest eigenvalue of the re-scaled adjacency matrix cA is equal to 0.9. This ensures that the time series process is stationary [Hamilton, 1994] while having a high signal-to-noise ratio.

4.3 Results

We report the performance of the NRI model variants on each of the three causal graph structures in Table 1. We see that the performance of NRI using the RefMLP encoder varies across the three

Graph	Encoder	AUC	F1	MSE	1-Edge Acc (%)	0-Edge Acc (%)
A	RefMLP	0.51 (0.01)	0.67 (0.00)	1.12 (0.03)	99.9 (0.1)	0.1 (0.1)
	Unshared	1.00 (0.00)	1.00 (0.00)	1.01 (0.02)	99.8 (0.0)	99.9 (0.0)
B	RefMLP	1.00 (0.00)	1.00 (0.00)	1.00 (0.03)	99.9 (0.1)	99.9 (0.1)
	Unshared	1.00 (0.00)	1.00 (0.00)	1.00 (0.03)	99.8 (0.0)	99.7 (0.1)
C	RefMLP	0.66 (0.16)	0.70 (0.04)	1.14 (0.03)	87.9 (12.0)	35.8 (36.1)
	Unshared	1.00 (0.00)	1.00 (0.00)	1.00 (0.02)	99.8 (0.1)	99.9 (0.0)

Table 1: NRI prediction loss and graph recovery classification accuracy on test samples. The NRI model is implemented with the Linear decoder. The prediction loss is expressed in MSE. "Edge Acc", "0-Edge Acc" and "1-Edge Acc" respectively give the encoder's classification accuracy on all of the test edges, accuracy on the test edges with ground truth 0 type (edge absent) and accuracy on the test edges with ground truth 1 type (edge present). AUC and F1 give the classification scores across all edges. All metrics are expressed as mean (standard deviation) across 24 Monte Carlo repetitions.

graph structures. In particular, it achieves perfect edge recovery on Graph B and achieves poor edge recovery on Graphs A and C.

In contrast, the Unshared encoder achieves perfect edge recovery on all three graphs. This shows that a typical transductive encoding approach which infers the Granger causal structure using variable selection learned through variational inference can work well on the problems considered. Further, the strong performance of the Unshared encoder suggests that the poor performance of the NRI RefMLP encoder is not due to latent variable non-identifiability [Wang et al., 2021]. We note that the Unshared encoder pools the inference of the Granger causal graph across all of the multivariate time series inputs: in this case, the posterior distribution will be highly peaked around the true Granger causal structure and approximating it with the Unshared encoder suffices. In the setting where there is a single Granger causal structure and the number of samples is large, the posterior will converge to a distribution with a single peaked mode (with little correlation structure to capture). This means that there will be little difference in performance between methods that place a point mass at the maximum a-posteriori and methods that capture the whole distribution [Wang and Blei, 2019]. We refer the reader to Geffner et al. [2022] for conditions on the asymptotic causal graph recovery of a variational expectation-maximization approach in the single-graph setting under the structural equation modeling scenario. However, for the finite sample case of Granger causal graph recovery, the issue of posterior dependence is at play, as we observe by studying the performance of the inductive RefMLP encoder.

The inconsistent performance of the NRI model across the three different graph types can be explained by examining the difficulty indicators for each graph in Figure 3.

Graph A:

$$\begin{pmatrix} 0 & 0 & \mathbf{0.42} \\ 0 & 0 & 0 \\ \mathbf{0.43} & \mathbf{0.43} & 0 \end{pmatrix}$$

Graph B:

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Graph C:

$$\begin{pmatrix} 0 & 0 & \mathbf{0.49} \\ \mathbf{0.49} & 0 & 0 \\ 0 & \mathbf{0.49} & 0 \end{pmatrix}$$

Figure 3: For each Granger causal graph used in the experiments, we display the difficulty indicators for its edges. Entry i, j in each matrix refers to the directed edge from node i to node j.

From Figure 3, we observe that the low classification scores in Table 1 align with those graphs with high difficulty indicators. Graphs A and C both have 3 non-zero difficulty indicators, while Graph C has no non-zero difficulty indicators.

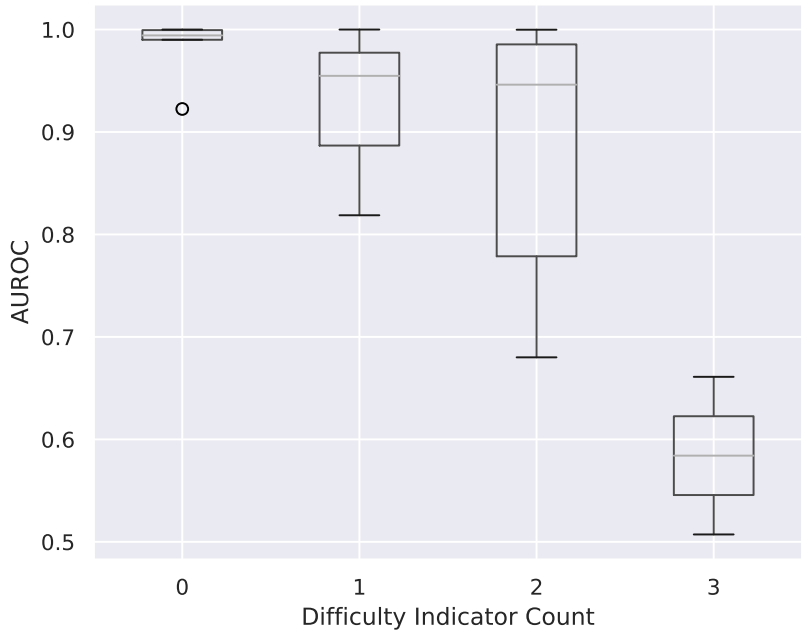


Figure 4: Barplot showing the mean AUC (AUROC) scores for the NRI model on all 16 directed Graphs on 3 nodes. The graphs are grouped by their "Difficulty Indicator Count" which counts the number of edges with non-zero difficulty indicators for each graph.

While we illustrate our theoretical argument with three graphs, we have experimentally tested our hypothesis on all 16 directed graphs on 3 vertices. These further experiments agree with the illustrative cases. The mean AUC/AUROC scores for these graphs are shown in Figure 4. The average AUC score for NRI is 0.99 for graphs with a Difficulty Indicator (DI) Count of 0 and 0.58 for graphs with a DI Count of 3. There is a -0.72 correlation of DI count with the AUC score and a linear regression of AUC onto the average difficulty indicator is significant at the $p < 0.01$ level. Since the difficulty indicator is constructed to identify edges with high relative posterior edge dependence effects, this validates our hypothesis that the NRI model is unable to capture specific Granger causal relations due to its mean-field approximation.

Applicability of findings to other data generating processes The theoretical argument presented in Section 3 applies to linear VAR data generating processes in the general case with N variables. While we have performed experiments on $N = 3$ variables, we expect that these theoretical limitations of NRI will manifest empirically with poor performance in experiments with a larger number of variables. Indeed, from equation (6), we see that posterior dependence effects between edges will exist so long as there are pairwise correlation effects between triplets of variables in the multivariate time series system. We have replicated our results on two graphs on $N = 4$ nodes; results are shown in Appendix H. The number of directed graphs on N nodes grows rapidly, which prevents exhaustive computational verification.

Intuitively, we expect that NRI will fail whenever there is a large dependence between edge indicators under the true posterior distribution. This can happen in the general non-linear VAR case as well. On more complex real-world data sets and models for which the posterior is not analytically tractable, we do not know whether there exists significant posterior dependence between variables without numerically simulating the posterior – this would be computationally costly and negate the purpose of using a variational approximation such as NRI. As a result, the arguments of this paper imply that NRI should be used on real-world Granger causal discovery problems with caution.

5 Relaxed Multivariate Bernoulli (RMVB) sampling and a novel NRI encoder

According to our preceding analysis, in order to improve the performance of the NRI model for Granger causal discovery, its mean-field variational approximation ought to be relaxed. In this section, we propose a relaxation of mean-field sampling that is theoretically able to capture posterior dependence.

Towards an edge-dependent sampling extension To our knowledge, no work has been proposed that alleviates the core bottleneck in NRI's capacity to recover the Granger Causal Graph – its mean-field posterior approximation. In response, we adapt a technique from the field of normalising flows to propose an NRI extension that is capable of modelling posterior correlation. Normalising flows are a class of invertible transformations that can be used to map samples from a simple distribution to a more complex one. In the context of VAEs, this allows for a distribution over latent variables that can capture posterior dependencies. Therefore, the inferred approximate latent variable distribution may better capture the true posterior distribution [Rezende and Mohamed, 2015]. In particular, by transforming a multivariate latent Gaussian sample with independent components through a series of normalising flow layers, the resulting distribution of the sample can be non-Gaussian with dependent components. In our case, we are interested in approximating the posterior distribution over Granger causal graphs by sampling dependent binary random variables that indicate edge presence. In the mean-field setting, the binary random variables are sampled using the so-called, concrete distribution [Maddison et al., 2016], a continuous relaxation of categorical random variables that permits gradient computation during training. Wang and Yin [2020] propose an extension, named Relaxed Multivariate Bernoulli (RMVB) sampling, which is a continuous relaxation of dependent binary random variable sampling. In the following section, we explain how RMVB sampling may be used as a building block in extending NRI.

In detail, we propose an extension to mean-field NRI which uses an edge sampling procedure based on RMVB (Relaxed Multivariate Bernoulli sampling). The RMVB method applies a simple linear transformation to a Gaussian sample with independent components; this enables Gaussian copula modelling. The multivariate Gaussian sample is then transformed into a relaxed Multivariate Bernoulli sample through a series of deterministic transformations. Thus, the sampling procedure can capture correlation structure in the latent posterior distribution. The dependence between the observed multivariate time series sample and the correlation structure in the latent posterior distribution is modelled through the use of an additional inference neural network. The use of a neural network to infer the correlation structure between latent variables in a RMVB model is analogous to the use of the encoder neural network to model the mean (marginal probability) for each edge given a multivariate time series in mean-field modelling setting.

Algorithm 1 Sampling from a Relaxed Multivariate Bernoulli distribution (RMVB)

Input: dimension of the distribution d, location vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_d)$, positive definite covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$ with σ_j^2 as the j-th diagonal element, temperature λ

Output: Relaxed Multivariate Bernoulli sample $\boldsymbol{z} = (z_1, \ldots, z_d) \in (0,1)^d$

Draw a standard normal sample: $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$
Compute $\mathbf{L} = \text{CholeskyDecomposition}\,(\Sigma)$
Generate a multivariate Gaussian vector: $\boldsymbol{g} = \mathbf{L}\boldsymbol{\eta}$
Apply element-wise Gaussian CDF Φ_{σ_j} with mean zero and variance σ_j^2:
    $U_j = \Phi_{\sigma_j}(g_j)$
Apply inverse CDF of the logistic distribution:
    $l_j = \log(\alpha_j) + \log(U_j) - \log(1 - U_j)$
Apply the sigmoid function:
    $z_j = \frac{1}{1+\exp(-l_j/\lambda)}$
return $\boldsymbol{z}$

In Algorithm 1 we display the procedure from Wang and Yin [2020] for sampling a relaxed Multivariate Bernoulli random variable of dimension d. In our case, $d = N(N-1)$, the number of edges in the Granger Causal Graph. The location vector $\boldsymbol{\alpha}$ for each multivariate time series sample is inferred

by a GNN encoder $\alpha_i \sim q_\alpha(\cdot|\mathbf{X}_{1:T})$ with the same architecture as the standard RefMLP encoder given above. A separate GNN encoder is used to infer the covariance matrix $\Sigma \in \mathbb{R}^{d\times d}$ for each multivariate time series input, $\Sigma_{ij} \sim q_\Sigma(\cdot|\mathbf{X}_{1:T})$. The architecture of the covariance matrix encoder network q_Σ is based on a RefMLP network with an output of size $\frac{d(d-1)}{2}$; the output is reshaped into a covariance matrix of size $d \times d$ by reshaping, symmetrisation ($\Sigma \leftarrow \Sigma + \Sigma^\intercal$), and normalisation to ensure that $|\Sigma_{ij}| \leq 1$. The diagonal elements of Σ are constrained to be 1 to avoid the additional standardisation step on line 5 of Algorithm 1. The outputs of the RMVB sampling procedure are the inferred latent edge categories which are used by the decoder in equation (4), as in the standard NRI formulation. We note that the RMVB sampling method is limited to sampling $K = 2$ edge types: present ($z = 0$) or absent ($z = 1$). This dual categorical encoding suffices for our experiments on the VAR model with a single lag but may be a limitation for modelling more complex interactions [Kipf et al., 2018]. For further details of the implementation of RMVB sampling see Appendix D.

The memory and computational cost of our proposed extension method is $O(N^4)$ in the number of nodes N. This is an unavoidable cost when modelling all the pairwise interactions between edges using a normalising flow [Madhawa et al., 2019]. As such, the NRI extension must be limited in application to small and moderate-sized graphs.

6 Evaluating NRI extensions

6.1 NRI extension Baselines

We have identified the limitations of the original NRI formulation on the task of Granger causal discovery and have proposed a novel extension, RMVB, to address these limitations. In this section, we evaluate the RMVB encoder as well as existing NRI extensions on the task of Granger causal discovery. We begin by giving an overview of relevant NRI extensions. Two plausible NRI extensions may yield improvements, as follows.

6.1.1 GAT encoder extension

In the standard NRI formulation, the encoder and decoder networks are given by Graph Neural Networks (GNNs) [Zhou et al., 2020]. The specific GNN architecture used in NRI consists of a message passing GNN which accommodates interaction between the message sender and receiver embeddings [Gilmer et al., 2017, Battaglia et al., 2016]. More recent GNN architectures such as Graph Attention Networks (GAT) [Veličković et al., 2017] augment the message-passing step by incorporating an attention mechanism into the neighbour message aggregation step; this mechanism modulates the strength of node-to-mode message-passing in the graph. In this paper, we investigate the effects of replacing the standard message passing network encoder with a GAT model. This may yield improvements in encoder expressivity which could translate to the task of Granger Causal Recovery. Gong et al. [2021] propose an extension of NRI based on using attention in conjunction with memory pools for enhanced long-range dynamic interaction modelling. More broadly, beyond their use in conjunction with NRI, attention mechanisms have been used for modelling continuous relations [Hoshen, 2017] as well as interactions between clustered objects Van Steenkiste et al. [2018].

6.1.2 MPM: Efficient Message Passing extension

Recent papers [Chen et al., 2021, Alet et al., 2019] have proposed NRI extensions that could potentially capture posterior dependencies amongst edges. Chen et al. [2021] stack an edge-to-edge message passing module on top of the standard NRI encoder which updates each edge's hidden representation prior to the sampling of its categories. Alet et al. [2019] propose a modular meta-learning approach which uses Gibbs sampling to jointly learn edge categories. However, Chen et al. [2021] find that the modular meta-learning approach is computationally costly and performs experimentally worse than their own method.

We consider the use of the stacked NRI encoder of Chen et al. [2021], which we call MPM in our experiments. However, under this proposal, edges are still sampled independently under the encoder. Therefore, we expect that MPM will not fully solve the problem of posterior edge independence.

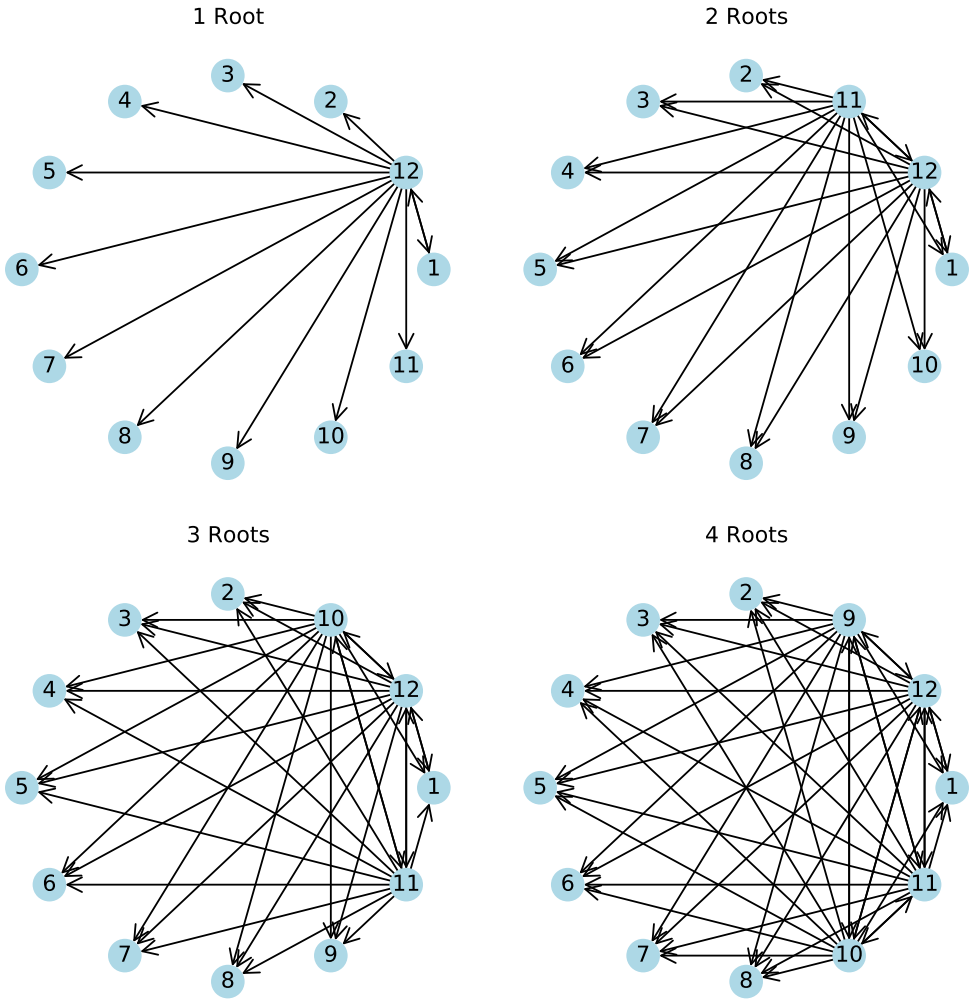


Figure 5: Four Granger causal graph structures on twelve nodes that have a high difficulty indicator count. The number of non-zero difficulty indicators on the graphs with 1, 2, 3 and 4 roots is respectively given by 120, 109, 98 and 87.

6.2 Experimental setting: high difficulty indicator test cases

We evaluate the NRI extensions on the experimental cases introduced in Section 4 and illustrated in Figure 2. In addition, we evaluate the performance of the NRI extensions on further constructed examples of Granger Causal Graphs which display a high degree of difficulty.

Using the topological interpretation of the difficulty indicator provided in Lemma 1, we construct examples of Granger Causal Graphs that should have a large degree of posterior dependence between edges. In particular, we consider the four Granger Causal Graphs displayed in Figure 5. The high-level motivation behind these graphs is to induce a contemporaneous correlation between time series by giving all time series at least one common ancestor while minimising the total number of pairwise Granger Causal relations. The four Granger Causal Graphs differ in the number of common ancestors, which we call "roots", that are shared by all nodes; the number of roots varies from 1 to 4 across the four Granger Causal Graphs. Out of the 132 possible edges in a directed graph on 12 nodes, the number of non-zero difficulty indicators on the graphs with 1, 2, 3 and 4 roots is respectively given by 120, 109, 98 and 87.

We consider two experimental settings on 12 nodes:

1. Single-graph setting: the performance of each NRI extension is evaluated separately on each of the four Granger Causal Graph settings displayed in Figure 5.
2. Multi-graph setting: here, the experimental dataset of multivariate samples contains data generated using an equal mixture of all four Granger Causal Graphs.

The details of the NRI extension implementations can be found in Appendix D.

6.3 Experimental results

Results on single-graph settings on three nodes In Table 2, we present the results of the NRI extensions on the three-node graphs A, B and C from Figure 2. We denote by MPM the NRI encoder which stacks the edge-to-edge message passing module of Alet et al. [2019] on top of the standard NRI encoder.

We see that RMVB Encoder achieves similar performance as RefMLP with respect to F1 score, AUC and MSE. The MPM encoder performance lags behind that of RefMLP on graph B while the GAT encoder lags behind RefMLP on all graph settings.

Graph	Encoder	AUC	F1	MSE	1-Edge Acc (%)	0-Edge Acc (%)
A	**RMVB**	**0.51 (0.02)**	**0.67 (0.00)**	**1.12 (0.03)**	**99.7 (0.4)**	**0.6 (1.3)**
	GAT	0.50 (0.00)	0.36 (0.34)	1.23 (0.12)	53.7 (50.5)	46.2 (50.5)
	MPM	0.50 (0.01)	0.67 (0.00)	1.12 (0.03)	99.9 (0.0)	0.1 (0.1)
	RefMLP	**0.51 (0.01)**	**0.67 (0.00)**	**1.12 (0.03)**	**99.9 (0.1)**	**0.1 (0.1)**
B	**RMVB**	**1.00 (0.00)**	**1.00 (0.00)**	**1.00 (0.03)**	**99.9 (0.1)**	**99.8 (0.2)**
	GAT	0.83 (0.24)	0.72 (0.43)	1.18 (0.22)	73.8 (43.6)	91.2 (23.8)
	MPM	0.82 (0.02)	0.80 (0.03)	1.28 (0.05)	74.8 (7.3)	75.3 (7.0)
	RefMLP	**1.00 (0.00)**	**1.00 (0.00)**	**1.00 (0.03)**	**99.9 (0.1)**	**99.9 (0.1)**
C	RMVB	0.50 (0.00)	0.67 (0.00)	1.13 (0.02)	99.8 (0.1)	0.3 (0.2)
	GAT	0.50 (0.00)	0.50 (0.29)	1.20 (0.11)	74.1 (43.7)	25.9 (43.7)
	MPM	0.50 (0.00)	0.67 (0.00)	1.13 (0.02)	99.9 (0.1)	0.1 (0.1)
	RefMLP	**0.66 (0.16)**	**0.70 (0.04)**	**1.14 (0.03)**	**87.9 (12.0)**	**35.8 (36.1)**

Table 2: NRI prediction loss and graph recovery classification accuracy on test samples on $N = 3$ nodes for the graphs in Figure 2. The NRI models are implemented with the Linear decoder. All metrics are expressed as mean (standard deviation) across 24 Monte Carlo repetitions. The NRI used encoder is shown in the column Encoder. The results with the highest mean AUC scores for each graph for shown in bold.

Results on high difficulty single-graph settings on twelve nodes The experimental results of the NRI extensions on the single-graph settings on $N = 12$ nodes are shown in Table 3. We see that RMVB sampler achieves stronger performance than RefMLP on all four graphs. The performance of the GAT and MPM Encoders is similar to that of RefMLP. We note that the AUC score of RefMLP is close to 0.5 — indicating that it struggles to do better than random chance on the task of Graph recovery on these challenging high difficulty indicator settings. These results are consistent with the performance of RefMLP on the high difficulty indicator graphs with three nodes, A and C, as displayed in Table 2.

Results on high difficulty multi-graph settings on ten and twelve nodes The experimental results of the NRI extensions on the multi-graph settings on $N = 12$ nodes are shown in Table 4. Alongside the results on $N = 12$ nodes, we also display the results of reducing the number of nodes to $N = 10$ while maintaining an equal mixture of 1-, 2-, 3- and 4- root graphs. We display the Granger causal graph structures on $N = 10$ in Figure 6 of Appendix F. Further, we display the results of increasing the AR coefficient strength from 0.9 to 0.99 for each of the $N = 10$ and $N = 12$ multi-graph settings. The AR coefficient was denoted by c in equation (5). In reporting the value of c in Table 4, we compute the leading eigenvalue for each of the four Granger causal AR matrix $A^{\intercal}$ in the multi-graph setting. Then, we normalise each AR matrix $A^{\intercal}$ by the maximum leading eigenvalue across the four

Number of Roots	Encoder	AUC	F1	MSE
1	**RMVB**	**0.72 (0.07)**	**0.22 (0.05)**	**1.30 (0.02)**
	GAT	0.50 (0.00)	0.07 (0.07)	1.35 (0.05)
	MPM	0.49 (0.02)	0.13 (0.00)	1.32 (0.02)
	RefMLP	0.53 (0.05)	0.08 (0.07)	1.35 (0.04)
2	**RMVB**	**0.65 (0.05)**	**0.31 (0.05)**	**1.31 (0.02)**
	GAT	0.50 (0.00)	0.06 (0.10)	1.42 (0.07)
	MPM	0.51 (0.01)	0.22 (0.00)	1.32 (0.02)
	RefMLP	0.52 (0.03)	0.14 (0.11)	1.38 (0.07)
3	**RMVB**	**0.67 (0.06)**	**0.39 (0.07)**	**1.27 (0.02)**
	GAT	0.50 (0.00)	0.14 (0.14)	1.39 (0.10)
	MPM	0.51 (0.00)	0.28 (0.00)	1.29 (0.02)
	RefMLP	0.55 (0.05)	0.15 (0.15)	1.38 (0.09)
4	**RMVB**	**0.63 (0.06)**	**0.39 (0.05)**	**1.23 (0.01)**
	GAT	0.50 (0.00)	0.13 (0.17)	1.36 (0.10)
	MPM	0.50 (0.00)	0.33 (0.00)	1.25 (0.02)
	RefMLP	0.54 (0.04)	0.25 (0.15)	1.31 (0.08)

Table 3: NRI prediction loss and graph recovery classification accuracy on single-graph test samples on $N = 12$ nodes. The NRI models are implemented with the Linear decoder. All metrics are expressed as mean (standard deviation) across 24 Monte Carlo repetitions. The NRI used encoder is shown in the column Encoder. The four graphs used are displayed in Figure 5 and are indicated by their number of "roots" in this Table. The results with the highest mean AUC scores for each graph for shown in bold.

graphs prior to re-scaling each AR matrix by c. Therefore, the values of c reported in Table 4 refer to the magnitude of the leading eigenvalue of each AR matrix $cA^\intercal$.

From Table 4, we see that RMVB sampler achieves stronger AUC performance than RefMLP on all multi-graph settings apart from the case $N = 10, c = 0.9$, where the performance between the two encoders is similar. We see that the relative gap in performance between RMVB and RefMLP performance is larger when we increase the number of nodes from $N = 10$ to $N = 12$ and when we increase the value of c, the AR coefficient. We find that the performance of the GAT and MPM Encoders is often worse than that of RefMLP.

Summary of the experimental results on the NRI extensions As expected by their mean-field restrictions, we find that the GAT and MPM models offer little improvement over RefMLP on the task of Granger causal graph recovery.

Summarising the performance of RMVB against the mean-field approximation RefMLP on the task of Granger Causal Graph discovery, we find that there is:

- No improvement in graph recovery performance when using RMVB sampling on graphs with a smaller number of nodes ($N = 3$).
- An improvement in graph recovery performance when using RMVB sampling on high difficulty indicator graphs with a larger number of nodes ($N = 10, 12$).

Therefore, we find some, albeit limited, improvements in performance when using the RMVB sampler. Moreover, RMVB tends to perform at least similarly well as RefMLP. Thus, these results make the RMVB sampler an interesting alternative to RefMLP. The next subsection provides a discussion of the current limitations of RMVB and potential avenues for improved Granger causal discovery with NRI using improved dependent variable sampling.

Number of nodes	AR coef	Encoder	AUC	F1	MSE
10	**0.9**	**RMVB**	**0.65 (0.00)**	**0.29 (0.01)**	**1.14 (0.01)**
		GAT	0.57 (0.08)	0.18 (0.13)	1.18 (0.04)
		MPM	0.55 (0.01)	0.24 (0.01)	1.17 (0.01)
		RefMLP	**0.65 (0.00)**	**0.29 (0.01)**	**1.14 (0.01)**
10	**0.99**	**RMVB**	**0.64 (0.00)**	**0.41 (0.00)**	**1.16 (0.01)**
		GAT	0.37 (0.01)	0.00 (0.00)	1.59 (0.01)
		MPM	0.53 (0.03)	0.34 (0.03)	1.25 (0.03)
		RefMLP	0.60 (0.09)	0.29 (0.17)	1.30 (0.21)
12	**0.9**	**RMVB**	**0.65 (0.00)**	**0.25 (0.01)**	**1.14 (0.01)**
		GAT	0.54 (0.08)	0.12 (0.12)	1.19 (0.04)
		MPM	0.55 (0.02)	0.21 (0.01)	1.17 (0.01)
		RefMLP	0.53 (0.11)	0.11 (0.12)	1.19 (0.04)
12	**0.99**	**RMVB**	**0.63 (0.00)**	**0.37 (0.00)**	**1.17 (0.01)**
		GAT	0.38 (0.01)	0.00 (0.00)	1.60 (0.01)
		MPM	0.54 (0.03)	0.29 (0.01)	1.27 (0.03)
		RefMLP	0.57 (0.11)	0.24 (0.16)	1.32 (0.20)

Table 4: NRI prediction loss and graph recovery classification accuracy on multi-graph test samples. The NRI models are implemented with the Linear decoder. All metrics are expressed as mean (standard deviation) across 24 Monte Carlo repetitions. The encoder used in NRI is shown in the column Encoder. The multi-graph setting is an equal-weighted mixture of the four graphs displayed in Table 5 in the $N = 12$ node case and of the four graphs displayed in Figure 6 in the $N = 10$ case. We also display the performance when varying the coefficient "AR coef" which modulates the strength of the auto-regressive effect. The results with the highest mean AUC scores for each graph for shown in bold.

6.4 Why does RMVB sampling only offer limited improvements in Granger causal discovery performance?

In this section, we discuss various hypotheses for the limited improvements observed when using RMVB over RefMLP.

High variance in encoder inference There are two sources of variation in the latent variable samples produced by any encoder. The first source is test-to-test sample variation due to the amortised inference encoder mapping different multivariate inputs to different posterior latent variable distributions. The second source is variance within a latent variable distribution for any given multivariate input. When analysing the latent variable samples produced by the RMVB encoder on the single-graph settings, we note that the first source of variation is quite pronounced: there is a high degree of test-to-test latent distribution variation indicating high variance in the encoder network inference for each multivariate sample. This issue is likely compounded in the high difficulty indicator case in which the true posterior distribution for a given sample (as given in equation (6)) has a larger variance; the encoder may struggle to distinguish the two sources of variation in this case.

We implemented two approaches for reducing the variance of the NRI encoder. These are general approaches for regularising the encoder in VAEs. The first is the noise-perturbation regularisation approach of Denoising Variational Autoencoders [Shu et al., 2018]. The second is the contractive regularisation approach of Rifai et al. [2011]. However, both methods yielded no improvement in the performance of the RefMLP or RMVB encoders (results not shown).

It is possible that a more sample-efficient encoder network may be able to achieve superior Granger causal graph recovery by reducing the encoder inference variance.

Low posterior correlation signal It may be the case that the posterior correlation signal obtained in our experiments is too low for the NRI encoder to pick up. This may add to the issue described above leading to the high variance in the inferred distributions. Indeed on the $N = 3$ graphs, the posterior correlation signal may be lower than on the high difficulty indicators on $N = 12$ – which

may explain the lack of RMVB outperformance in the $N = 3$ case. Further, from Table 4, we see that increasing the AR coefficient c and the number of nodes N in the graph increases the RMVB's relative performance; this may be due to an increased posterior correlation signal. Nevertheless, in limited further experiments not shown, we find that increasing the number of nodes to $N = 14$ does not yield further improvements. Alternative synthetic settings beyond the VAR model may be useful in investigating the effect of posterior correlation strength.

VAE optimisation issues VAEs are prone to optimisation difficulties [Fu et al., 2019]. A common source of failure, known as *KL vanishing*, occurs when the encoder produces posteriors identical to the Gaussian prior and the decoder learns to ignore the latent variable samples. This issue is less likely to occur when the decoder has limited expressivity, as in our experiments with the linear decoder, since in this case, the decoder will achieve poor training loss when ignoring the encoder. Nonetheless, we experimented with KL-annealing, an approach for mitigating KL vanishing which varies the influence of the KL-prior term, during training. We found that this yielded little improvement in experimental performance for NRI when using an MLP or a linear decoder.

Another source of optimisation failure occurs when the encoder parameters become stuck in a local stationary point during training. In the case of NRI, the GNN encoder may converge to a local optimum which corresponds to a latent graph mode that differs from the true graph. An approach to mitigating this local optimisation issue is to gradually anneal the temperature of the concrete distribution (referred to as τ in equation (2)). This approach permits the exploration of many latent graph structures at the start of training. Another common approach to avoid local optimisation uses cyclical learning rate annealing schemes [Smith, 2017]; a spike in the learning rate may permit the encoder parameters to jump out of a local minimum.

We found that starting at a sufficiently large temperature (we use $\tau = 0.5$ in experiments) was important for avoiding local minima at the start of training. However, learning rate and temperature annealing schemes did not yield improvements in model training.

Alternative normalising flow methods Our RMVB extension can be viewed as a linear autoregressive normalising flow. Superior results for modelling posterior correlation may be achieved by using more flexible normalising flow transformations. The requirement that the latent distribution is over discrete edge variables with a dependence structure restricts the applicability of normalising flow methods. We briefly discuss some approaches in the field which may be adapted to create a novel NRI encoder. We note that these normalising flow methods are designed to generate samples rather than encode them; therefore they would have to be inverted to function as VAE encoders [Rezende and Mohamed, 2015].

Within the context of normalising flow methods that use the "re-parameterisation trick" [Rezende and Mohamed, 2015] along with a bijective transformation to obtain differentiable gradient estimates, the GraphNVP [Madhawa et al., 2019] method may be relevant. GraphNVP proposes a flow-based model for generating an adjacency matrix for graphs. The method introduces a coupling layer that transforms a single row of the adjacency matrix while keeping the other rows constant. A de-quantisation operation is applied to the discrete adjacency matrix input to transform it into a continuous latent representation. The approach of [Honda et al., 2019] is similar but it uses a residual rather than a coupling layer for a bijective transformation. Approaches based on Discrete Flows [Tran et al., 2019, Luo et al., 2021] use autoregressive or bi-partite normalisation flow transformations on a modulo scale to perform categorical sampling. An alternative approach to modelling dependent latent variables in a VAE framework uses an LSTM to sample the latent variables, and score-function gradient estimators to obtain unbiased gradient estimates [Annadani et al., 2021].

The downside of normalising flow approaches is that a single flow layer requires $O(N^4)$ parameters in the encoder network. This makes inference computationally challenging and the model becomes prone to overfitting.

Alternative limiting factors Our work is limited to the analysis of a single graph-based data generating process. On more complex data generating processes, the performance of NRI may be inhibited by some aspect other than failure to capture posterior edge dependence. For instance, in cases with low posterior edge dependence, the ability of the NRI encoder architecture to approximate the marginal edge posterior distribution may be its limiting factor in Granger causal discovery.

Understanding other cases in which NRI fails to recover the ground truth graph is another interesting direction of further work.

7 Conclusion

By theoretical and experimental arguments, we have shown that the mean-field posterior approximation inherent in NRI poses a challenge to its application to Granger causal discovery. The limitation of NRI is apparent even on a simple benchmark data generating process, for which we are able to approximately characterise the cases in which NRI fails to achieve satisfactory Granger causal graph recovery.

In light of the limitations of mean-field inference for Granger causal discovery, we examine the performance of three extensions of NRI. We find that the GAT encoder and the stacked message passing encoder of Chen et al. [2021] fail to improve on graph recovery tasks. This is consistent with their mean-field assumptions. On the contrary, we find that our novel extension, which uses RMVB sampling to overcome the mean-field approximation, is able to achieve moderate outperformance compared to the baseline NRI model on examples that demonstrate a high degree of posterior dependence. As RMVB typically matches the performance of the NRI's standard RefMLP encoder but can yield better performance, it is an interesting alternative to RefMLP for small graphs. We hope that our work will encourage future research in adapting NRI to meet the challenge of Granger causal discovery.

Acknowledgments

SB is supported by the EPSRC CDT in Modern Statistics and Statistical Machine Learning (EP/S023151/1) and The Alan Turing Institute's Finance and Economics Programme. G.R is supported in part by EPSRC grants EP/T018445/1, EP/W037211/1, EP/V056883/1, and EP/R018472/1. RY was supported in part by the U.S. Army Research Office under Army-ECASE award W911NF-07-R-0003-03, the U.S. Department Of Energy, Office of Science, IARPA HAYSTAC Program, NSF Grants #2205093, #2146343, and #2134274.

References

Ferran Alet, Erica Weng, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Neural relational inference with fast modular meta-learning. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Yashas Annadani, Jonas Rothfuss, Alexandre Lacoste, Nino Scherrer, Anirudh Goyal, Yoshua Bengio, and Stefan Bauer. Variational causal networks: approximate Bayesian inference over causal structures. *arXiv preprint arXiv:2106.07635*, 2021.

Andrew Arnold, Yan Liu, and Naoki Abe. Temporal causal modeling with graphical Granger methods. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 66–75, 2007.

Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics. *Advances in Neural Information Processing Systems*, 29, 2016.

John Y. Campbell, Andrew W. Lo, A. Craig MacKinlay, and Robert F. Whitelaw. The econometrics of financial markets. *Macroeconomic Dynamics*, 2(4):559–562, 1998.

Giorgio Castagneto-Gissey, Mario Chavez, and Fabrizio De Vico Fallani. Dynamic Granger-causal networks of electricity spot prices: A novel approach to market integration. *Energy Economics*, 44: 422–432, 2014. ISSN 0140-9883.

Siyuan Chen, Jiahai Wang, and Guoqing Li. Neural relational inference with efficient message passing mechanisms. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8):7055–7063, May 2021.

Michael Eichler. *Causal Inference in Time Series Analysis*. Wiley Online Library, 2012.

Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. Cyclical annealing schedule: A simple approach to mitigating KL vanishing. *arXiv preprint arXiv:1903.10145*, 2019.

André Fujita, Patricia Severino, João Ricardo Sato, and Satoru Miyano. Granger causality in systems biology: Modeling gene networks in time series microarray data using vector autoregressive models. In *Advances in Bioinformatics and Computational Biology*, pages 13–24. Springer Berlin Heidelberg, 2010. ISBN 978-3-642-15060-9.

Tomas Geffner, Javier Antoran, Adam Foster, Wenbo Gong, Chao Ma, Emre Kiciman, Amit Sharma, Angus Lamb, Martin Kukla, Nick Pawlowski, Miltiadis Allamanis, and Cheng Zhang. Deep end-to-end causal inference. *arXiv preprint arXiv:2202.02195*, 2022.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, pages 1263–1272. PMLR, 2017.

Dong Gong, Frederic Z Zhang, Javen Qinfeng Shi, and Anton Van Den Hengel. Memory-augmented dynamic neural relational inference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11843–11852, 2021.

Clive WJ Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: Journal of the Econometric Society*, pages 424–438, 1969.

James Douglas Hamilton. *Time Series Analysis*. Princeton University Press, 1994.

Shion Honda, Hirotaka Akita, Katsuhiko Ishiguro, Toshiki Nakanishi, and Kenta Oono. Graph residual flow for molecular graph generation. *arXiv preprint arXiv:1909.13521*, 2019.

Yedid Hoshen. Vain: Attentional multi-agent predictive modeling. *Advances in Neural Information Processing Systems*, 30, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2688–2697. PMLR, 10–15 Jul 2018.

Marina Knight, Kathryn Leeming, Guy Nason, and Matthew Nunes. Generalized network autoregressive processes and the GNAR package. *Journal of Statistical Software*, 96(5):1–36, 2020.

Ling Liu, Yan Cheng, Zhigang Zhang, Jing Li, Yichao Geng, Qingsong Li, Daxian Luo, Li Liang, Wei Liu, Jianping Hu, and Weiwei Ouyang. Study on the allosteric activation mechanism of SHP2 via elastic network models and neural relational inference molecular dynamics simulation. *Phys. Chem. Chem. Phys.*, 25:23588–23601, 2023.

Zhiyuan Liu and Jie Zhou. *Introduction to Graph Neural Networks*. Springer Nature, 2022.

Sindy Löwe, David Madras, Richard Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. In *Proceedings of the First Conference on Causal Learning and Reasoning*, volume 177 of *Proceedings of Machine Learning Research*, pages 509–525. PMLR, 11–13 Apr 2022.

Youzhi Luo, Keqiang Yan, and Shuiwang Ji. GraphDF: A discrete flow model for molecular graph generation. In *International Conference on Machine Learning*, pages 7192–7203. PMLR, 2021.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The Concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Kaushalya Madhawa, Katushiko Ishiguro, Kosuke Nakago, and Motoki Abe. GraphNVP: An invertible flow model for generating molecular graphs. *arXiv preprint arXiv:1905.11600*, 2019.

Mariusz Maziarz. A review of the Granger-causality fallacy. *The Journal of Philosophical Economics*, 8(2):6, 2015.

George Michailidis and Florence d'Alché Buc. Autoregressive models for gene regulatory network inference: Sparsity, stability and causality issues. *Mathematical Biosciences*, 246(2):326–334, 2013. ISSN 0025-5564.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pages 1530–1538. PMLR, 2015.

Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *Proceedings of the 28th International Conference on Machine Learning*, pages 833–840, 2011.

Christian P Robert and George Casella. *Monte Carlo Statistical Methods*. Springer, 1999.

Anil K Seth, Adam B Barrett, and Lionel Barnett. Granger causality analysis in neuroscience and neuroimaging. *Journal of Neuroscience*, 35(8):3293–3297, 2015.

Rui Shu, Hung H Bui, Shengjia Zhao, Mykel J Kochenderfer, and Stefano Ermon. Amortized inference regularization. *Advances in Neural Information Processing Systems*, 31, 2018.

Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 464–472. IEEE, 2017.

Olaf Sporns. *Networks of the Brain*. MIT press, 2016.

Alex Tank, Ian Covert, Nicholas Foti, Ali Shojaie, and Emily B Fox. Neural Granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021.

Dustin Tran, Keyon Vafa, Kumar Agrawal, Laurent Dinh, and Ben Poole. Discrete flows: Invertible generative models of discrete data. *Advances in Neural Information Processing Systems*, 32, 2019.

Sjoerd Van Steenkiste, Michael Chang, Klaus Greff, and Jürgen Schmidhuber. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. *arXiv preprint arXiv:1802.10353*, 2018.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Xi Wang and Junming Yin. Relaxed multivariate Bernoulli distribution and its applications to deep generative models. In *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124 of *Proceedings of Machine Learning Research*, pages 500–509. PMLR, 03–06 Aug 2020.

Yixin Wang and David M. Blei. Frequentist consistency of variational Bayes. *Journal of the American Statistical Association*, 114(527):1147–1161, 2019.

Yixin Wang, David Blei, and John P Cunningham. Posterior collapse and latent variable non-identifiability. In *Advances in Neural Information Processing Systems*, volume 34. Curran Associates, Inc., 2021.

Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81, 2020. ISSN 2666-6510.

Jingxuan Zhu, Juexin Wang, Weiwei Han, and Dong Xu. Neural relational inference to learn long-range allosteric interactions in proteins from molecular dynamics simulations. *Nat Commun*, 13: 1661, 2022.

A Derivation of the posterior for the linear autoregressive model with graph structure

In the derivation below, we use $D_{i=1,\ldots,4}$ to denote variables that are constant in $(A_{ij})_{i,j\in\{1,\ldots,N\}}$.

Under the prior distribution, the graph edges are modelled using independent Bernoulli random variables,

$$\mathbb{P}\left((A_{ij})_{i,j\in\{1,\ldots,N\},i\neq j}\right) = \prod_{i,j\in\{1,\ldots,N\},i\neq j} p^{A_{ij}}(1-p)^{1-A_{ij}}. \tag{8}$$

Using an autoregressive factorisation, the likelihood is given by

$$\mathbb{P}\left(\mathbf{X}_{1:T}\middle|(A_{ij})_{i,j\in\{1,\ldots,N\},i\neq j}, c\right) = \prod_{t=1}^{T}\prod_{i=1}^{N}\mathbb{P}\left(X_t^i\middle|\left(X_{t-1}^j\right)^{j=1,\ldots,N}, (A_{ij})_{i,j\in\{1,\ldots,N\},i\neq j}, c\right) \tag{9}$$

where $X_0^i := 0$, $i = 1,\ldots,N$. The logarithm of the posterior distribution over edges is therefore given by

$$\begin{aligned}\log\mathbb{P}\left((A_{ij})_{i,j\in\{1,\ldots,N\},i\neq j}\middle|\mathbf{X}_{1:T}\right) = \\ \sum_{t=1}^{T}\sum_{i=1}^{N}\log\mathbb{P}\left(X_t^i\middle|\left(X_{t-1}^j\right)^{j=1,\ldots,N}, (A_{ij})_{i,j\in\{1,\ldots,N\},i\neq j}, c\right) \\ + \sum_{i=1}^{N}\sum_{j\neq i}\left[A_{ij}\log p + (1-A_{ij})\log(1-p)\right].\end{aligned} \tag{10}$$

Using the data generating process given by equation (5) and the Gaussian probability density function for the residual errors, we find that by expanding the square,

$$\begin{aligned}&\log\mathbb{P}\left(X_t^i\middle|\left(X_{t-1}^j\right)^{j=1,\ldots,N}, (A_{ij})_{i,j\in\{1,\ldots,N\},i\neq j}, c\right) \\ &= -\frac{1}{2}\left(X_t^i - cX_{t-1}^i - c\sum_{j\neq i}A_{ji}X_{t-1}^j\right)^2 + D_1 \\ &= \frac{c}{2}\sum_{j\neq i}A_{ji}\left[(2X_t^i - cX_{t-1}^i)X_{t-1}^j - c(X_{t-1}^j)^2 - c\sum_{k\in\{1,\ldots,N\}\setminus\{i,j\}}A_{ki}X_{t-1}^jX_{t-1}^k\right] + D_2.\end{aligned} \tag{11}$$

Substituting this expression in equation (10) gives

$$\begin{aligned}\log\mathbb{P}\left((A_{ij})_{i,j\in\{1,\ldots,N\},i\neq j}\middle|\mathbf{X}_{1:T}\right) = \\ \frac{c}{2}\sum_{i=1}^{N}\sum_{j\neq i}A_{ij}\Big[2(\mathcal{L}\mathbf{X}_{1:T}^i)\cdot\mathbf{X}_{1:T}^j - c(\mathcal{L}\mathbf{X}_{1:T}^j)\cdot(\mathcal{L}\mathbf{X}_{1:T}^i) \\ - c(\mathcal{L}\mathbf{X}_{1:T}^i)\cdot(\mathcal{L}\mathbf{X}_{1:T}^i) - c\sum_{k\neq i,j}A_{kj}(\mathcal{L}\mathbf{X}_{1:T}^i)\cdot(\mathcal{L}\mathbf{X}_{1:T}^k)\Big] \\ + \sum_{i=1}^{N}\sum_{j\neq i}\left[A_{ij}\log p + (1-A_{ij})\log(1-p)\right] + D_3\end{aligned} \tag{12}$$

where $\mathbf{X}^i_{1:T} := (X^i_1, \ldots, X^i_T)^\intercal \in \mathbb{R}^T$, $\mathcal{L}\mathbf{X}^i_{1:T} := (X^i_0, X^i_1, \ldots, X^i_{T-1})^\intercal \in \mathbb{R}^T$ for all $i = 1, \ldots, N$ and $\cdot$ denotes the vector dot product. Therefore, by Bayes' rule

$$\begin{aligned}
&\log \mathbb{P}\left(A_{ij} | \mathbf{X}_{1:T}, \{A_{kl}\}_{(k,l)\neq(i,j)}\right) = \\
&\frac{c}{2} A_{ij} \Bigg[2(\mathcal{L}\mathbf{X}^i_{1:T}) \cdot \mathbf{X}^j_{1:T} - c(\mathcal{L}\mathbf{X}^j_{1:T}) \cdot (\mathcal{L}\mathbf{X}^i_{1:T}) \\
&- c(\mathcal{L}\mathbf{X}^i_{1:T}) \cdot (\mathcal{L}\mathbf{X}^i_{1:T}) - c \sum_{k \neq i,j} A_{kj} (\mathcal{L}\mathbf{X}^i_{1:T}) \cdot (\mathcal{L}\mathbf{X}^k_{1:T}) \Bigg] \\
&+ [A_{ij} \log p + (1 - A_{ij}) \log(1-p)] + D_4
\end{aligned} \tag{13}$$

which is the conditional posterior for A_{ij} given in equation (6).

B Proof of Lemma 1, which provides a topological interpretation of the difficulty indicator

Proof. The expected value $\mathbb{E}[C_{ij}(1 - A^*_{ij})] = (1 - A^*_{ij})\mathbb{E}[C_{ij}]$ is non-zero if and only if $\mathbb{E}[C_{ij}] \neq 0$ and $A^*_{ij} = 0$. The second condition, $A^*_{ij} = 0$, is equivalent to saying that i is not a parent of j, giving the second condition of Lemma 1. From now on assume that $A^*_{ij} = 0$ and use A to denote the true value of the adjacency matrix A^*, for simplicity.

Using the definition of C_{ij},

$$\mathbb{E}[C_{ij}] = c \sum_{k \in \{1,\ldots,N\} \setminus \{i,j\}} A_{kj}\, \mathbb{E}\left[(\mathcal{L}\mathbf{X}^i) \cdot (\mathcal{L}\mathbf{X}^k)\right],$$

where $(\mathcal{L}\mathbf{X}^i) \cdot (\mathcal{L}\mathbf{X}^k) = \sum_{t=1}^{T-1} X^i_t X^k_t$. We assume that the process X is covariance stationary with unconditional contemporaneous covariance matrix Σ [Hamilton, 1994]. Then, we have $\mathbb{E}\left[X^i_t X^k_t\right] = \mathbb{E}\left[X^i_0 X^k_0\right] = \Sigma_{ik}$, and therefore,

$$\mathbb{E}[C_{ij}] = c(T-1) \sum_{k \in \{1,\ldots,N\} \setminus \{i,j\}} A_{kj}\, \Sigma_{ik}. \tag{14}$$

Using the definition given in equation (5) for the autoregressive process, we can derive the recursion

$$\mathbb{E}[\mathbf{X}_T \mathbf{X}_T^\top] = c^2 A^\top \mathbb{E}[\mathbf{X}_{t-1} \mathbf{X}_{t-1}^\top] A + I,$$

where $\mathbf{X}_T \in \mathbb{R}^N$ is the multivariate value of the time series at time T and I is the $N \times N$ identity matrix. By covariance-stationary, this is equivalent to

$$\Sigma = c^2 A^\top \Sigma A + I.$$

Recursively applying this expression, we obtain the following equation for any $m \geq 1$,

$$\Sigma = c^{2m} (A^\top)^m \Sigma A^m + \sum_{s=0}^{m-1} c^{2s} (A^\top)^s A^s,$$

where $A^0 := I$. Since X is covariance stationary, the magnitude of the largest eigenvalue of $cA^\top$ is less than 1 [Hamilton, 1994]. Therefore, for any $\epsilon > 0$, there exists a value m_1, such that $\forall m > m_1$ and for all pairs i, j, $\left|c^{2m}(A^\top)^m \Sigma A^m\right|_{i,j} < \epsilon$. Therefore, as $m \to \infty$, we find that

$$\Sigma = \sum_{t=0}^{\infty} c^{2t} \left(A^\top\right)^t (A)^t. \tag{15}$$

Since $c > 0$ and $A \in \{0,1\}^{N \times N}$, we see that $\Sigma_{i,k} \geq 0$ for any i, k. It follows from expression (14) that $\mathbb{E}[C_{ij}] \neq 0$ holds if and only if there exists $k \in \{1, \ldots, N\} \setminus \{i, j\}$ such that

$$A_{kj} \Sigma_{i,k} \neq 0,$$

i.e. $A_{kj} \neq 0$ and $\Sigma_{i,k} \neq 0$. From the form of Σ given by the infinite series in equation (15), the condition $\Sigma_{i,k} \neq 0$ is equivalent to the existence of a value of $t \in \{0, 1, 2, \ldots\}$ such that $\left[\left(A^\intercal\right)^t (A)\right]_{i,k} \neq 0$. This occurs if and only if there exists a sequence of nodes $p_1, p_2, \ldots, p_{2t-1}$ such that $p_t \to p_{t-1} \to \ldots \to p_1 \to i$ and $p_t \to p_{t+1} \to \ldots \to p_{2t-1} \to k$. Equivalently, $\exists m \in \{1, \ldots, N\} \setminus \{i, j\}$ such that $m \to \ldots \to k$ and $m \to \ldots \to i$. Therefore equation (14) holds if and only if j has a parent $k \neq i, j$ such that k has a common ancestor with i, giving the first condition of Lemma 1. □

C NRI implementation

C.1 RefMLP Encoder

The implementation of the RefMLP encoder follows that of Löwe et al. [2022] and Kipf et al. [2018]. The specifics of the implementation are:

- We use the last 200 values of each time series as input into the first layer of the encoder.
- The encoder MLPs f_{emb}, $f_e^{(1)}$, $f_v^{(1)}$ have two layers, each with 32 hidden units and ELU (Exponential Linear Unit) activation.
- The final node-to-edge step $f_e^{(2)}$ consists of a 2-layer MLP with 32 hidden units which contains an additional skip connection to the initial edge representations. This is passed through a single-layer MLP with 32 hidden units to obtain the final edge embeddings.

Under the prior distribution, each edge is sampled uniformly at random from a Bernoulli distribution; the probability of class "no edge" is set to 0.95. This ensures that under the null hypothesis of no Granger causal relations, the type I error rate for each edge is 0.05.

C.2 MLP decoder

The MLP decoder used in the experiments reported in Appendix G has the same architecture as that of Löwe et al. [2022] and Kipf et al. [2018], except that we use 32 hidden units in each MLP layer for our experiments.

D Implementation details for NRI extensions

D.1 MPM encoder

We follow the implementation of Chen et al. [2021] for the inter- and intra- final edge-to-edge message passing layers. The hidden unit size for the intra- and inter- GRUs is 32. We use a linear layer to unify the results for the intra- and inter- layers.

D.2 GAT encoder

We use the same architecture for the initial time series embedding MLP f_{emb} as was used in the RefMLPEncoder described above. However, we replace the RefMLP Encoder message-passing steps described above with three graph attentional layers. These layers of hidden unit size 32 follow the description provided in Veličković et al. [2017]. As in the RefMLP implementation, we use a single-layer MLP to obtain the final edge embeddings.

D.3 RMVB encoder

The RMVB encoder consists of a location network that infers edge locations for each multivariate time series observation (given by α in Algorithm 1), and a covariance network which infers edge correlations for each observation (given by Σ in Algorithm 1). The location network is identical to that of the RefMLP encoder with a hidden unit size of 32, which is described above. The architecture of the covariance network is based on a RefMLP network with an output of size $\frac{d(d-1)}{2}$ and a hidden unit size of 16. As described in Section 5, the output of the covariance network is processed into

a covariance matrix of size $d \times d$ with unit diagonal elements by reshaping, symmetrisation and normalisation.

E Model training details

In all experiments, NRI was trained using stochastic gradient descent with a batch size of 100 and with ADAM optimisation [Kingma and Ba, 2014]. We monitor loss curves to verify that convergence occurs during training. The ELBO performance on the validation set is used for model selection. The initial ADAM learning rate is selected from the choices $\{0.05, 0.005\}$. The temperature τ of the concrete distribution is set to $\tau = 0.5$. In the experiments with graphs containing $N = 10$ or $N = 12$ nodes, we use 200 epochs of training. Otherwise, when the graph contains $N = 3$ nodes, we use 50 epochs of training.

F Figure displaying the four graphs on ten nodes with high difficulty indicator counts

Figure 6 displays four Granger causal graph structures on $N = 10$ nodes that have high difficulty indicator count. These graphs are the ten-node analogues of the graphs displayed in Figure 5.

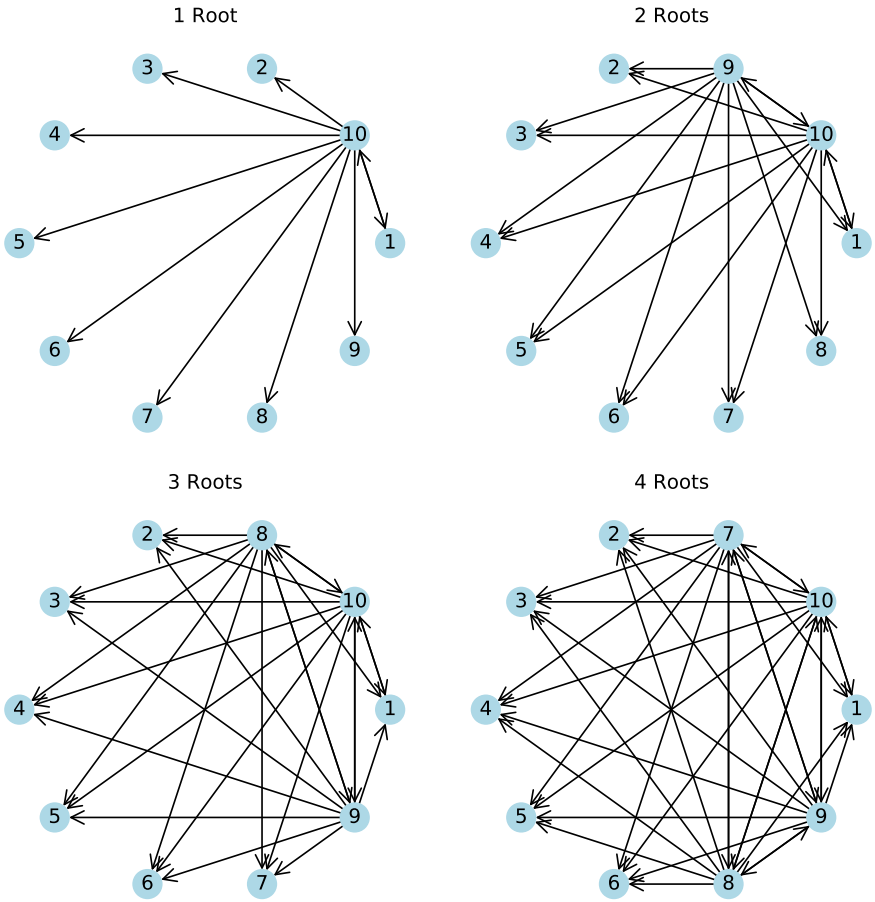


Figure 6: Four Granger causal graph structures on ten nodes that have a high difficulty indicator count.

G Experimental results using NRI with an MLP decoder

In Section 4, results using NRI with an MLP decoder were briefly described. Here are the details.

The experimental results of NRI using the MLP decoder on the graphs with three nodes from Figure 2 are shown in Table 5. We observe that the results are similar to the Linear decoder case in Table 2. We recall that classic Granger causal inference techniques [Arnold et al., 2007] result in perfect recovery in this case.

Graph	Encoder	AUC	F1	MSE	1-Edge Acc (%)	0-Edge Acc (%)
A	**RMVB**	**0.69 (0.02)**	**0.69 (0.02)**	**1.13 (0.03)**	**82.2 (5.0)**	**44.6 (6.2)**
	GAT	0.50 (0.00)	0.66 (0.01)	1.13 (0.03)	96.0 (3.1)	4.1 (3.0)
	MPM	0.50 (0.02)	0.66 (0.00)	1.13 (0.04)	98.0 (1.3)	2.2 (1.4)
	RefMLP	0.66 (0.07)	0.68 (0.02)	1.13 (0.03)	83.2 (9.7)	38.0 (18.7)
B	**RMVB**	**1.00 (0.00)**	**1.00 (0.00)**	**1.00 (0.03)**	**99.9 (0.1)**	**99.8 (0.1)**
	GAT	0.82 (0.19)	0.89 (0.10)	1.09 (0.08)	99.8 (0.3)	47.7 (48.3)
	MPM	0.76 (0.04)	0.83 (0.00)	1.11 (0.04)	83.5 (0.1)	66.9 (0.1)
	RefMLP	**1.00 (0.00)**	**1.00 (0.00)**	**1.00 (0.03)**	**99.9 (0.1)**	**99.9 (0.1)**
C	RMVB	0.59 (0.15)	0.69 (0.04)	1.13 (0.03)	91.8 (11.7)	22.1 (33.7)
	GAT	0.50 (0.00)	0.66 (0.01)	1.13 (0.03)	96.9 (2.2)	3.2 (2.1)
	MPM	0.50 (0.00)	0.66 (0.00)	1.13 (0.03)	98.5 (1.4)	1.3 (1.1)
	RefMLP	**0.83 (0.02)**	**0.76 (0.01)**	**1.12 (0.03)**	**76.9 (2.4)**	**73.5 (3.0)**

Table 5: NRI prediction loss and graph recovery classification accuracy on test samples for different encoders. The NRI models are implemented with the MLP decoder. The prediction loss is expressed in MSE. "Edge Acc", "0-Edge Acc" and "1-Edge Acc" respectively give the classification accuracy of the encoder on all of the test edges, accuracy on the test edges with ground truth 0 type (edge absent) and accuracy on the test edges with ground truth 1 type (edge present). AUC and F1 give the classification scores across all edges. All reported metrics are expressed as mean (standard deviation) across 24 Monte Carlo repetitions. The results with the highest mean AUC scores for each graph for shown in bold.

H Further experimental results using graphs with 4 nodes

In this Appendix, we report the results of experiments that were run on two graphs with four nodes. The graphs are shown in Figure 7.

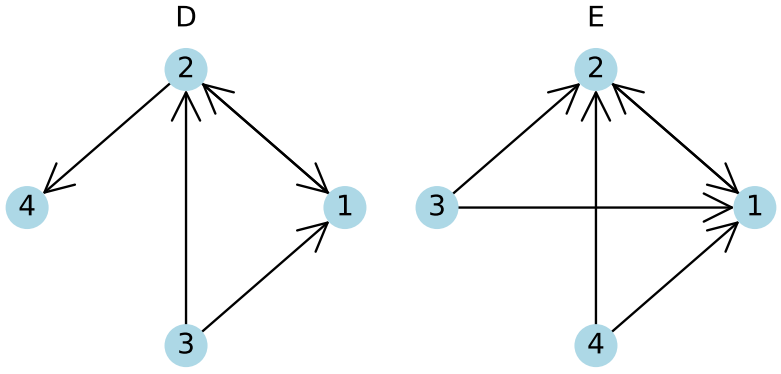


Figure 7: The two Granger causal graph structures on four nodes used in experiments.

We display the difficulty indicators for each of the two graphs on four nodes in Figure 8. We see that graph D has four non-zero difficulty indicators whereas all the difficulty indicators for graph E are zero.

Graph D:

$$\begin{pmatrix} 0 & 0 & 0 & \mathbf{0.39} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{0.59} \\ \mathbf{0.36} & \mathbf{0.36} & 0 & 0 \end{pmatrix}$$

Graph E:

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Figure 8: For each four-node Granger causal graph used in the experiments, we display the difficulty indicators for its edges. Entry i, j in each matrix refers to the directed edge from node i to node j.

The results of the experiments are displayed in Table 6. We use the experimental settings that were described in section 4.

Graph	Encoder	AUC	F1	MSE	1-Edge Acc (%)	0-Edge Acc (%)
D	**RMVB**	**0.90 (0.04)**	**0.81 (0.02)**	**1.15 (0.04)**	**82.9 (4.8)**	**83.6 (9.0)**
	GAT	0.66 (0.12)	0.49 (0.30)	1.33 (0.16)	65.0 (41.1)	58.6 (24.7)
	MPM	0.63 (0.01)	0.56 (0.01)	1.40 (0.03)	65.9 (3.5)	50.0 (3.1)
	RefMLP	0.81 (0.08)	0.78 (0.03)	1.15 (0.03)	92.4 (8.0)	66.2 (16.1)
E	**RMVB**	**1.00 (0.00)**	**1.00 (0.00)**	**1.00 (0.02)**	**99.9 (0.1)**	**100.0 (0.1)**
	GAT	0.63 (0.22)	0.32 (0.41)	1.45 (0.19)	30.8 (41.2)	92.0 (12.2)
	MPM	0.87 (0.02)	0.71 (0.03)	1.42 (0.04)	59.8 (6.1)	90.9 (2.3)
	RefMLP	**1.00 (0.00)**	**1.00 (0.00)**	**1.00 (0.02)**	**99.9 (0.1)**	**100.0 (0.0)**

Table 6: NRI prediction loss and graph recovery classification accuracy on test samples on $N = 4$ nodes for the graphs in Figure 7. The NRI models are implemented with the Linear decoder. All metrics are expressed as mean (standard deviation) across 24 Monte Carlo repetitions. The encoder used with NRI is shown in the column Encoder. The results with the highest mean AUC scores for each graph for shown in bold.

The results presented in Table 6 are consistent with Hypothesis 1. We see that RefMLP is able to perfectly recover graph E, for which all difficulty indicators are zero. RefMLP has a lower graph recovery accuracy on graph D, which has four non-zero difficulty indicators. Further, we find that RMVB achieves a higher graph recovery accuracy than RefMLP on graph D.

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	Rethinking Neural Relational Inference for Granger Causal Discovery
Publication Status	☒ Published □ Accepted for Publication □ Submitted for Publication □ Unpublished and unsubmitted work written in a manuscript style
Publication Details	A preliminary version of the work was presented at the 2022 NeurIPS Workshop on Causality for Real-world Impact

Student Confirmation

Student Name:	Stefanos Bennett		
Contribution to the Paper	This is my work, which started under the supervision of Prof Rose Yu during a research exchange and continued under the joint supervision of Profs. Rose Yu, Mihai Cucuringu and Gesine Reinert.		
Signature		Date	02/01/24

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Mihai Cucuringu		
Supervisor comments Stefanos made a substantial contribution to the publication, as indicated above.		
Signature 	Date	5 January 2024

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 5

Conclusion

5.1 Summary of Findings

This thesis comprises three research projects, each addressing specific challenges in machine learning for time series. We briefly recapitulate the key findings here.

Lead–lag detection and network clustering for multivariate time series Chapter 2 introduces a novel method, which, to the best of our knowledge, is the first to address the problem of unsupervised clustering of leading and lagging variables in multivariate time series systems. Our method can capture non-linear lead-lag correlations and leverages a state-of-the-art directed network clustering algorithm which is able to detect clusters with high flow imbalance. The specific sub-components of our method, as well as its overall performance, are validated on a range of synthetic test cases for unsupervised clustering. When applied to US equity data, our method produces a statistically significant clustering that cannot be explained by three prominent lead-lag hypotheses in the empirical finance literature. An exploratory analysis provides insights into the structure of the US equity market. Further, we show how our clustering method can be used for the construction of a statistically significant trading signal in the US equity market; this demonstrates the usefulness of our clustering method for signal extraction in noisy, high-dimensional multivariate time series systems.

Time series prediction under distribution shift Chapter 3 introduces a novel model-agnostic distribution shift method for time series that uses a generic forgetting mechanism to infer sample importance weights. The forgetting mechanism controls the trade-off between the relevancy and effective sample size that is used for the estimation of the predictive model. We propose learning forgetting mechanisms for time series prediction under distribution shift using gradient-based optimisation. In contrast to previous work, we propose a gradient-based learning method for the parameters of the forgetting mechanism. This speeds up optimisation and therefore allows more expressive forgetting mechanisms. We theoretically situate our method and demonstrate its efficacy on synthetic and real-world datasets. Our method is able to model a range of synthetically generated distribution shifts, from gradual drift to regime switching behaviour, while maintaining strong performance in the stationary case. Through experiments on real-world forecasting tasks, we show that our method is able to outperform distribution shift baselines.

NRI for Granger Causal Discovery Chapter 4 explores the limitations of neural relational inference (NRI) in Granger causal discovery and presents a novel extension overcoming these challenges. We show that the mean-field posterior approximation inherent in NRI poses a challenge for its application to Granger causal discovery. The limitation of NRI is apparent even on a simple linear autoregressive data generating process, for which we are able to approximately characterise the cases in which NRI

fails to achieve satisfactory Granger causal graph recovery. Motivated by a theoretical argument, we propose an indicator that predicts when NRI will fail to recover the Granger causal structure on the linear benchmark process. We empirically validate our indicator using synthetic data experiments.

Our argument concerning the limitations of mean-field modelling also applies to existing extensions of NRI. We find that the GAT encoder and the stacked message passing encoder of Chen et al. [2021] do *not* yield improved performance on Granger causal recovery tasks. We propose a novel NRI extension that uses Relaxed Multivariate Bernoulli sampling to overcome the mean-field approximation. This novel extension is able to achieve moderate outperformance compared to the baseline NRI model on examples that demonstrate a high degree of posterior dependence.

Towards a comprehensive modelling of complex dependencies in non-stationary financial systems Chapters 2, 3 and 4 provide significant contributions to the problem of modelling complex dependencies in non-stationary financial systems. The lead-lag clustering method developed in Chapter 2 can be applied to understand the cluster-based temporal dependency structure in multivariate time series systems. The work of Chapter 2 develops a general method for adapting time series models in the presence of distribution shift; in particular, this method may be applied towards dynamically adapting interaction models in response to non-stationary conditions. Chapters 2 and 3 provide flexible tools that may be used to analyse complex dependencies in financial systems without relying on extensive domain knowledge. Chapter 4 investigates a model for temporal interactions in time series systems which aims to discover Granger causal interactions. Modelling causality has traditionally relied on domain expertise to specify model structure. Novel approaches such as NRI aim to provide general-purpose modelling of complex dependencies in time series systems which decreases reliance on domain expertise in the process of causal discovery. We show in Chapter 4 that NRI must be adapted to fulfil the promise of comprehensive Granger causal discovery.

5.2 Practical applications

In this section, we highlight current and future areas of application for the methods that we have developed in each chapter.

Lead–lag detection and network clustering for multivariate time series The application of our method to investigate the structure of the US equity market shows that our methodology is a useful tool for the exploration of novel lead-lag mechanisms in the discipline of empirical finance. Our method can be applied to derive a data-driven clustering of other financial instruments whenever there is a hypothesised lead-lag relationship between time series. New potential areas for the application of our method include investigating clustered lead-lag relationships on a higher-frequency scale [Ito and Sakemoto, 2020], on other markets [Castagneto-Gissey et al., 2014], or between different markets [Yong Tang and Zhang, 2019].

The forecasting application presented in Chapter 2 demonstrates that our method can be employed for challenging downstream forecasting tasks in noisy, high-dimensional settings. Our method recovers clusters of financial time series such that time series within a cluster have similar leading and lagging behaviours with respect to time series in other clusters. This enables signal extraction and the use of cluster-level predictive modelling. Our method may be applicable as an unsupervised step within a forecasting pipeline in other financial time series contexts as mentioned above.

In addition to the financial domain, the applicability of our proposed clustering methodology extends to other areas – such as biology [Michailidis and d'Alché Buc, 2013], medicine [Seth et al., 2015] and earth sciences [Harzallah and Sadourny, 1997] – that are characterised by multivariate time series system consisting of related entities that exhibit a latent lead-lag structure.

Further, sub-components of our method may be applied in isolation. For instance, the lead-lag network construction component of our method has been used in the work of Mantziou et al. [2023]. This work studies an aggregate network of UK business payments. The network construction sub-component of our method is used by Mantziou et al. [2023] to perform network thinning and sparsification.

Time series prediction under distribution shift Our model-agnostic method for predicting time series under distribution shift can be applied to a range of domains. Indeed in the experimental section of our work in Chapter 3, we apply our method to economic variable, energy and epidemic forecasting tasks, among other areas.

We briefly highlight our application to financial risk modelling. We show how our method can better explain the dynamic risk profile of a range of US equities. The time-variation in the risk loadings of equities plays an important role in financial risk management. The correlations between assets usually increase in periods of distress [Bouchaud, 2002]; this poses a challenge to portfolio diversification. Our method is able to dynamically adapt the risk loadings for a range of equities on the Fama-French three-factor model [Fama and French, 1993], providing a more accurate representation of the risk profile of a portfolio of stocks. An interesting area of application for our method would be to use it in conjunction with more sophisticated risk model.

NRI for Granger Causal Discovery The arguments of the work presented in Chapter 4 imply that NRI should be used with caution on real-world Granger causal discovery problems. In particular, NRI is prone to fail on Granger causal recovery tasks in settings characterised by low-to-moderate sample sizes or which have low signal-to-noise ratios.

5.3 Future work

We now discuss certain avenues of future work for each of three projects.

Lead–lag detection and network clustering for multivariate time series Our proposed method infers a clustering structure based on lead-lag relations between time series. In general, the clustering component of our method may be used with any pairwise directed relation defined between time series. Thus, our framework may be generalised beyond *lead-lag* interactions, in order to discover cluster structure in high-dimensional time-series systems based on *general* directed interactions.

Time series prediction under distribution shift In Chapter 3, we have focused on one-step univariate forecasting tasks to simplify the exposition of our method. The extension to multi-step or multivariate prediction tasks is, in theory, straightforward by simply using a multi-step or multivariate loss function. Often, a multi-step or multivariate loss function decomposes into a sum of separate univariate loss functions: for example, an N-step MSE loss decomposes into a sum of N different univariate loss functions. In this case, the forgetting mechanism should, in principle, output a separate weight to model the distribution shift for each of these separate loss functions. However, it may be the case that a degree of weight sharing is beneficial if the nature of the distribution shift simultaneously affects all prediction tasks. The evaluation of our method on multi-step and multivariate forecasting tasks is subject to future work.

A second avenue of future work would be to explore more complex forgetting mechanisms. In order to improve the mapping from time-index to sample weight, we could draw on advances from the field of implicit neural representations [Sitzmann et al., 2020, Woo et al., 2022]. Extending the time-index input with a sample loss feature, as in Shu et al. [2019], could be a useful strategy to

equip the forgetting mechanism to deal with data corruption or noise. More generally, it would be interesting to extend the feature inputs for the forgetting mechanism using the sample inputs and labels. In this context, sample input and label dimensionality reduction are crucial for importance ratio estimation [Maia Polo and Vicente, 2022, Stojanov et al., 2019]. Joint feature extraction for the predictive model and the forgetting mechanism may be useful in this case [Zhang et al., 2021].

A third research extension would be to adapt our method to the online learning setting. In this case, the sample weights and the predictive model would be updated on the fly in response to streaming data. A basic approach to handling streaming data using the stochastic gradient descent algorithm proposed in Chapter 3 would be to take a gradient update whenever a new batch of data is available. The perturbation-based approach of Ren et al. [2018] provides an example of how sample re-weighting methods may be adapted to the streaming setting.

NRI for Granger Causal Discovery In Chapter 4, we discuss various avenues for improving the performance of our NRI extension. In particular, we discuss the possibility of reducing the high-variance in the test-to-test latent distribution through a more sample-efficient NRI encoder. A sample-efficient NRI encoder could be helpful in the fields of finance and economics, in which the signal-to-noise ratios of financial and economic prediction tasks tend to be lower than the signal-to-noise ratios of the highly structured physical and biological systems on which NRI has traditionally been applied [Liu et al., 2023, Zhu et al., 2022]. In Chapter 4, we also discuss several alternative normalising flow methods that may be adapted to improve the posterior correlation-modelling capacity of NRI [Madhawa et al., 2019, Honda et al., 2019, Annadani et al., 2021].

Chapter 4 is limited to the analysis of a single graph-based data generating process. Understanding when NRI fails to recover the ground truth graph on other data generating processes is an interesting direction of further work. Alternative synthetic settings beyond the GNAR model may be useful in investigating the effect of posterior correlation strength. In addition, we may find that the performance of NRI on more complex data generating processes is inhibited by some aspect other than failure to capture posterior edge dependence. For instance, in cases with low posterior edge dependence, the NRI encoder architecture's ability to approximate the marginal edge posterior distribution may be its limiting factor in Granger causal discovery.

In this work, we specifically examine the question of NRI's performance in Granger Causal Discovery, an important inference task in the fields of finance and economics. Given the economic and financial data characteristics described in Section 1.1.1 of the introduction, we expect that the application of NRI to these domains will pose a number of challenges. Yet, certain challenges in the domain of economics and finance are shared with other fields. The success of NRI in its application to the biological and physical sciences [Liu et al., 2023, Zhu et al., 2022] has led to several extensions which could be utilised in these cases. We briefly outline two shared modelling challenges in relation to NRI and its extensions. Firstly, the graph permutation invariance assumption underpinning the use of a GNN encoder and decoder does not hold in cases when the time series entities are highly heterogeneous. NRI adaptations, such as Seq2VAR [Pineau et al., 2019] or the work of Banijamali [2022] and Ramos et al. [2021], that mediate between time series-specific auto-regressive effects and graph time series structure, are relevant in such cases. Secondly, many financial and economic time series settings are characterised by non-stationarity. In these settings, NRI extensions that are able to accommodate time-varying graphs are relevant [Graber and Schwing, 2020, Xiao et al., 2020].

Towards a comprehensive modelling of complex dependencies in non-stationary financial systems Beyond the specific extensions described above, there exist broader challenges in comprehensively modelling complex dependencies in financial systems. Many of the most expressive models, including deep learning approaches, tend to lack interpretability. Interpretability in financial modelling is a desirable trait for high-stakes applications. Training highly expressive models often requires

large datasets to be effective, which can be a limitation in certain financial settings. Additionally, scaling these models to handle very high-dimensional datasets efficiently remains a challenge.

5.4 Outlook

We finish with a discussion of the relationship of the work in this thesis to current research trends.

Lead–lag detection and network clustering for multivariate time series Since the publication of our work in Chapter 2, more recent research has continued the data-driven analysis of financial systems using lead-lag networks. For instance, Cartea et al. [2023] explore the rankings of equity based on lead-lag networks constructed using the signatures and cross-correlation approach that we describe in our work. Zhang et al. [2023] investigate clustering lead-lag networks in the context of a lagged multi-factor data generating process. Additionally, research has pursued more complex methods for recovering lead-lag relations. Cheng et al. [2022] use a graph neural network in conjunction with multiple data modalities (historical returns, news and a knowledge graph of linked entities) to recover an equity lead-lag network. Shi et al. [2023] posit a latent variable model which underpins lead-lag relations and is estimated using multi-reference alignment.

Time series prediction under distribution shift Our proposed method for forgetting mechanism inference using bi-level optimisation is part of a growing number of works which apply ideas from bi-level optimisation to machine learning [Sinha et al., 2017]. Early applications include model selection or hyperparameter search [Bennett et al., 2006]. More recently, bi-level optimisation has been applied to the field of meta-learning [Rajeswaran et al., 2019]. Of particular interest is a recent work [Broderick et al., 2023] which uses bi-level optimisation to investigate the robustness of statistical hypothesis testing conclusions to the removal of a portion of the data samples. While the problem under study – robustness to small data subsets – is different, Broderick et al. [2023] use bi-level optimisation within a similar framework to us: data samples are associated with weights which are then probed with gradient bi-level optimisation methods to determine the portion of the data to which the conclusions are most sensitive.

The interaction between time series prediction and hyper-parameter optimisation, presented in Chapter 3, is part of a growing research trend towards automated machine learning (AutoML) for time series. AutoML aims to provide tools to build machine learning systems without much requirement for domain expertise or human input. Traditional AutoML methods have focused on the tabular data setting [LeDell and Poirier, 2020, He et al., 2021]. Increasingly, there has been an interest in extending these frameworks to the time series setting [Shchur et al., 2023]. The automated building of machine learning models that can handle time-series characteristics, such as distribution shift, lies at the centre of such efforts.

NRI for Granger Causal Discovery Our investigation of NRI for Granger causality contributes to the growing application of deep learning methods for Granger causal discovery [Chu et al., 2020, Tank et al., 2021, Yin and Barucca, 2022]. Graph deep learning approaches are also increasingly being used for causal discovery under the potential outcomes framework [Vowels et al., 2022]. The potential outcomes framework imposes further constraints, for example, acyclicity, on the causal graph estimates. Typically, causal graphs are inferred on a non-inductive basis: model re-training needs to be performed for new observations [Geffner et al., 2022, Ng et al., 2019]. More recent work has proposed inductive modelling to map from observations directly to causal graphs [Ke et al., 2022]. Beyond causal inference, there is a growing literature on GNN-based forecasting of multivariate time series [Cini et al., 2023a].

Outlook The work in this thesis is motivated by the characteristics of financial time series, which include complex dependencies, low signal-to-noise ratios, asynchronicity, and non-stationarity. Each of the methods developed in this thesis addresses one or more of these features. However, there is still a need for methods that can jointly address these characteristics in an automated fashion. More broadly, there is no shortage of problems in the fields of time series forecasting, financial analysis, and latent relation inference. We hope that our research will motivate the development of machine learning methods for time series to address these problems.

Bibliography

Torben G Andersen, Tim Bollerslev, Francis X Diebold, and Paul Labys. Modeling and forecasting realized volatility. *Econometrica*, 71(2):579–625, 2003.

Yashas Annadani, Jonas Rothfuss, Alexandre Lacoste, Nino Scherrer, Anirudh Goyal, Yoshua Bengio, and Stefan Bauer. Variational causal networks: approximate Bayesian inference over causal structures. *arXiv preprint arXiv:2106.07635*, 2021.

Andrew Arnold, Yan Liu, and Naoki Abe. Temporal causal modeling with graphical Granger methods. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 66–75, 2007.

Marco Avellaneda and Jeong-Hyun Lee. Statistical arbitrage in the US equities market. *Quantitative Finance*, 10(7):761–782, 2010.

S. G. Badrinath, R. Kale Jayant, and H. Noe Thomas. Of shepards, sheep and the cross-autocorrelations in equity returns. *The Review of Financial Studies*, 8(2), 1995.

LEI BAI, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive graph convolutional recurrent network for traffic forecasting. In *Advances in Neural Information Processing Systems*, volume 33, pages 17804–17815. Curran Associates, Inc., 2020.

Ershad Banijamali. Neural relational inference with node-specific information. In *International Conference on Learning Representations*, 2022.

Lasko Basnarkov, Viktor Stojkoski, Zoran Utkovski, and Ljupco Kocarev. Lead-lag Relationships in Foreign Exchange Markets. *arXiv preprint arXiv:1906.10388*, 2019.

Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics. *Advances in neural information processing systems*, 29, 2016.

Kristin P Bennett, Jing Hu, Xiaoyun Ji, Gautam Kunapuli, and Jong-Shi Pang. Model selection via bilevel optimization. In *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pages 1922–1929. IEEE, 2006.

Christoly Biely and Stefan Thurner. Random matrix ensembles of time-lagged correlation matrices: Derivation of eigenvalue spectra and analysis of financial time-series. *Quantitative Finance*, 8(7): 705–722, 2008. ISSN 14697696.

Jean-Philippe Bouchaud. An introduction to statistical finance. *Physica A: Statistical Mechanics and its Applications*, 313(1):238–251, 2002. ISSN 0378-4371.

Michael J . Brennan, Jegadeesh Narasimhan, and Bhaskaran Swaminathan. Investment analysis and the adjustment of stock prices to common information source. *The Review of Financial Studies*, 6 (4):799–824, 1993.

Tamara Broderick, Ryan Giordano, and Rachael Meager. An automatic finite-sample robustness metric: When can dropping a little data make a big difference? *arxiv preprint arXiv:2011.14999*, 2023.

John Y. Campbell, Andrew W. Lo, A. Craig MacKinlay, and Robert F. Whitelaw. The econometrics of financial markets. *Macroeconomic Dynamics*, 2(4):559–562, 1998.

Álvaro Cartea, Mihai Cucuringu, and Qi Jin. Detecting lead-lag relationships in stock returns and portfolio strategies. *Available at SSRN*, 2023.

Giorgio Castagneto-Gissey, Mario Chavez, and Fabrizio De Vico Fallani. Dynamic Granger-causal networks of electricity spot prices: A novel approach to market integration. *Energy Economics*, 44: 422–432, 2014. ISSN 0140-9883.

Siyuan Chen, Jiahai Wang, and Guoqing Li. Neural relational inference with efficient message passing mechanisms. *Proceedings of the AAAI conference on artificial intelligence*, 35(8):7055–7063, May 2021.

Dawei Cheng, Fangzhou Yang, Sheng Xiang, and Jin Liu. Financial time series forecasting with multi-modality graph neural network. *Pattern Recognition*, 121:108218, 2022. ISSN 0031-3203.

Tarun Chordia and Bhaskaran Swaminathan. Trading Volume and Cross-Autocorrelations in Stock Returns. *The Journal of Finance*, LV(2):913–935, 2000.

Yunfei Chu, Xiaowei Wang, Jianxin Ma, Kunyang Jia, Jingren Zhou, and Hongxia Yang. Inductive Granger causal modeling for multivariate time series. In *2020 IEEE International Conference on Data Mining (ICDM)*, pages 972–977. IEEE, 2020.

Andrea Cini, Ivan Marisca, Daniele Zambon, and Cesare Alippi. Graph deep learning for time series forecasting. *arXiv preprint arXiv:2310.15978*, 2023a.

Andrea Cini, Daniele Zambon, and Cesare Alippi. Sparse graph learning from spatiotemporal time series. *Journal of Machine Learning Research*, 24(242):1–36, 2023b.

Lauren Cohen, Andrea Frazzini, and Christopher Malloy. The small world of investing: Board connections and mutual fund returns. *Journal of Political Economy*, 116(5):951–979, 2008.

Rama Cont. Empirical properties of asset returns: Stylized facts and statistical issues. *Quantitative Finance*, 1(2):223–236, 2001. ISSN 14697696.

Chester Curme, Michele Tumminello, Rosario N. Mantegna, H. Eugene Stanley, and Dror Y. Kenett. Emergence of statistically validated financial intraday lead-lag relationships. *Quantitative Finance*, 15(8):1375–1386, 2015a. ISSN 14697696.

Chester Curme, Michele Tumminello, Rosario N. Mantegna, H. Eugene Stanley, and Dror Y. Kenett. How lead-lag correlations affect the intraday pattern of collective stock dynamics. *Office of Financial Research Working Paper Series*, 2015b.

Ailin Deng and Bryan Hooi. Graph neural network-based anomaly detection in multivariate time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4027–4035, 2021.

Stefanie Eckel, Gunter Löffler, Alina Maurer, and Volker Schmidt. Measuring the effects of geographical distance on stock market correlation. *Journal of Empirical Finance*, 18(2):237–247, 2011. ISSN 0927-5398.

Michael Eichler. *Causal inference in time series analysis*. Wiley Online Library, 2012.

Eugene F Fama and Kenneth R French. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1):3–56, 1993. ISSN 0046-9777.

Tongtong Fang, Nan Lu, Gang Niu, and Masashi Sugiyama. Rethinking importance weighting for deep learning under distribution shift. In *Advances in Neural Information Processing Systems*, volume 33, pages 11996–12007, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arxiv preprint arXiv:1703.03400*, 2017.

Luca Franceschi, Mathias Niepert, Massimiliano Pontil, and Xiao He. Learning discrete structures for graph neural networks. In *International Conference on Machine Learning*, pages 1972–1982. PMLR, 2019.

André Fujita, Patricia Severino, João Ricardo Sato, and Satoru Miyano. Granger causality in systems biology: modeling gene networks in time series microarray data using vector autoregressive models. In *Advances in Bioinformatics and Computational Biology*, pages 13–24. Springer Berlin Heidelberg, 2010. ISBN 978-3-642-15060-9.

Tomas Geffner, Javier Antoran, Adam Foster, Wenbo Gong, Chao Ma, Emre Kiciman, Amit Sharma, Angus Lamb, Martin Kukla, Nick Pawlowski, Miltiadis Allamanis, and Cheng Zhang. Deep end-to-end causal inference. *arXiv preprint arXiv:2202.02195*, 2022.

Mingming Gong, Kun Zhang, Bernhard Schoelkopf, Dacheng Tao, and Philipp Geiger. Discovering temporal causal relations from subsampled data. In *International Conference on Machine Learning*, pages 1898–1906. PMLR, 2015.

Colin Graber and Alexander Schwing. Dynamic neural relational inference for forecasting trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1018–1019, 2020.

James D Hamilton. Regime switching models. In *Macroeconometrics and Time Series Analysis*, pages 202–209. Springer, 2010.

A. Harzallah and R. Sadourny. Observed lead-lag relationships between Indian summer monsoon and some meteorological variables. *Climate Dynamics*, 13(9):635–648, 1997. ISSN 14320894.

Xin He, Kaiyong Zhao, and Xiaowen Chu. AutoML: A survey of the state-of-the-art. *Knowledge-Based Systems*, 212:106622, 2021.

Shion Honda, Hirotaka Akita, Katsuhiko Ishiguro, Toshiki Nakanishi, and Kenta Oono. Graph residual flow for molecular graph generation. *arXiv preprint arXiv:1909.13521*, 2019.

Yongmiao Hong, Yanhui Liu, and Shouyang Wang. Granger causality in risk and detection of extreme risk spillover between financial markets. *Journal of Econometrics*, 150(2):271–287, 2009.

Kevin D Hoover. Causality in economics and econometrics. *Available at SSRN 930739*, 2006.

Katsuya Ito and Ryuta Sakemoto. Direct estimation of lead–lag relationships using multinomial dynamic time warping. *Asia-Pacific Financial Markets*, 27(3):325–342, 2020.

Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10:1391–1445, 2009. ISSN 15324435.

Nan Rosemary Ke, Silvia Chiappa, Jane Wang, Anirudh Goyal, Jorg Bornschein, Melanie Rey, Theophane Weber, Matthew Botvinic, Michael Mozer, and Danilo Jimenez Rezende. Learning to induce causal structure. *arXiv preprint arXiv:2204.04875*, 2022.

Saurabh Khanna and Vincent YF Tan. Economy statistical recurrent units for inferring nonlinear Granger causality. *arXiv preprint arXiv:1911.09879*, 2019.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2688–2697. PMLR, 10–15 Jul 2018.

Marina Knight, Kathryn Leeming, Guy Nason, and Matthew Nunes. Generalized network autoregressive processes and the GNAR package. *Journal of Statistical Software*, 96(5):1–36, 2020.

Vitaly Kuznetsov and Mehryar Mohri. Discrepancy-based theory and algorithms for forecasting non-stationary time series. *Annals of Mathematics and Artificial Intelligence*, 88(4):367–399, 2020.

Markku Lanne, Mika Meitz, and Pentti Saikkonen. Identification and estimation of non-gaussian structural vector autoregressions. *Journal of Econometrics*, 196(2):288–304, 2017. ISSN 0304-4076.

Michael Lechner. The relation of different concepts of causality used in time series and microeconometrics. *Econometric Reviews*, 30(1):109–127, 2010.

Erin LeDell and Sebastien Poirier. H2O AutoML: Scalable automatic machine learning. *7th ICML Workshop on Automated Machine Learning (AutoML)*, July 2020.

Yunzhu Li, Antonio Torralba, Anima Anandkumar, Dieter Fox, and Animesh Garg. Causal discovery in physical systems from videos. *Advances in Neural Information Processing Systems*, 33:9180–9192, 2020.

Chang Liao, Yinfei Huang, Xibin Shi, and Xin Jin. Mining influence in evolving entities: A study on stock market. *DSAA 2014 - Proceedings of the 2014 IEEE International Conference on Data Science and Advanced Analytics*, pages 244–250, 2014.

Ling Liu, Yan Cheng, Zhigang Zhang, Jing Li, Yichao Geng, Qingsong Li, Daxian Luo, Li Liang, Wei Liu, Jianping Hu, and Weiwei Ouyang. Study on the allosteric activation mechanism of SHP2 via elastic network models and neural relational inference molecular dynamics simulation. *Phys. Chem. Chem. Phys.*, 25:23588–23601, 2023.

Andrew W. Lo and A. Craig MacKinlay. When are contrarian profits due to stock market overreaction? *The Review of Financial Studies*, 3(2):175–205, 1990a.

Andrew W Lo and A Craig MacKinlay. An econometric analysis of nonsynchronous trading. *Journal of Econometrics*, 45(1-2):181–211, 1990b.

Sindy Löwe, David Madras, Richard Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. In *Proceedings of the First Conference on Causal Learning and Reasoning*, volume 177 of *Proceedings of Machine Learning Research*, pages 509–525. PMLR, 11–13 Apr 2022.

Nan Lu, Tianyi Zhang, Tongtong Fang, Takeshi Teshima, and Masashi Sugiyama. Rethinking importance weighting for transfer learning. *arXiv preprint arXiv:2112.10157*, 2112.10157:1–44, 2021.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The Concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Kaushalya Madhawa, Katushiko Ishiguro, Kosuke Nakago, and Motoki Abe. GraphNVP: An invertible flow model for generating molecular graphs. *arXiv preprint arXiv:1905.11600*, 2019.

Felipe Maia Polo and Renato Vicente. Effective sample size, dimensionality, and generalization in covariate shift adaptation. *Neural Computing and Applications*, 2022. ISSN 14333058.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The M4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1):54–74, 2020. ISSN 0169-2070.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. M5 accuracy competition: Results, findings, and conclusions. *International Journal of Forecasting*, 38(4):1346–1364, 2022. ISSN 0169-2070.

Rosario N Mantegna. Hierarchical structure in financial markets. *The European Physical Journal B-Condensed Matter and Complex Systems*, 11:193–197, 1999.

Rosario N. Mantegna and H. Eugene Stanley. *Introduction to Econophysics: Correlations and Complexity in Finance*. Cambridge University Press, 1 edition, July 2007. ISBN 0521039878.

Anastasia Mantziou, Mihai Cucuringu, Victor Meirinhos, and Gesine Reinert. The GNAR-edge model: A network autoregressive model for networks with time-varying edge weights. *arxiv preprint arxiv:2305.16097*, 2023.

Luca Masserano, Syama Sundar Rangapuram, Shubham Kapoor, Rajbir Singh Nirwan, Youngsuk Park, and Michael Bohlke-Schneider. Adaptive sampling for probabilistic forecasting under distribution shift. In *NeurIPS 2022 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2022.

Mariusz Maziarz. A review of the Granger-causality fallacy. *The Journal of Philosophical Economics*, 8 (2):6, 2015.

Daniel McCarthy and Shane T. Jensen. Power-weighted densities for time series data. *The Annals of Applied Statistics*, 10(1):305–334, 2016.

George Michailidis and Florence d'Alché Buc. Autoregressive models for gene regulatory network inference: Sparsity, stability and causality issues. *Mathematical Biosciences*, 246(2):326–334, 2013. ISSN 0025-5564.

Ignavier Ng, Shengyu Zhu, Zhitang Chen, and Zhuangyan Fang. A graph autoencoder approach to causal structure learning. *arXiv preprint arXiv:1911.07420*, 2019.

J.-P. Onnela, A. Chakraborti, K. Kaski, J. Kertész, and A. Kanto. Dynamics of market correlations: Taxonomy and portfolio analysis. *Physical Review E*, 68(5), November 2003. ISSN 1095-3787.

Boris N Oreshkin, Arezou Amini, Lucy Coyle, and Mark Coates. FC-GAGA: Fully connected gated graph architecture for spatio-temporal traffic forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 9233–9241, 2021.

Alvaro D Orjuela-Cañón, Jan A Freund, Andres Jutinico, and Alexander Cerquera. Granger causality analysis based on neural networks architectures for bivariate cases. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE, 2020.

Fotios Petropoulos, Daniele Apiletti, Vassilios Assimakopoulos, Mohamed Zied Babai, Devon K. Barrow, Souhaib Ben Taieb, Christoph Bergmeir, Ricardo J. Bessa, Jakub Bijak, John E. Boylan, Jethro Browell, Claudio Carnevale, Jennifer L. Castle, Pasquale Cirillo, Michael P. Clements, Clara Cordeiro, Fernando Luiz Cyrino Oliveira, Shari De Baets, Alexander Dokumentov, Joanne Ellison, Piotr Fiszeder, Philip Hans Franses, David T. Frazier, Michael Gilliland, M. Sinan Gönül, Paul Goodwin, Luigi Grossi, Yael Grushka-Cockayne, Mariangela Guidolin, Massimo Guidolin, Ulrich Gunter, Xiaojia Guo, Renato Guseo, Nigel Harvey, David F. Hendry, Ross Hollyman, Tim Januschowski, Jooyoung Jeon, Victor Richmond R. Jose, Yanfei Kang, Anne B. Koehler, Stephan Kolassa, Nikolaos Kourentzes, Sonia Leva, Feng Li, Konstantia Litsiou, Spyros Makridakis, Gael M. Martin, Andrew B. Martinez, Sheik Meeran, Theodore Modis, Konstantinos Nikolopoulos, Dilek Önkal, Alessia Paccagnini, Anastasios Panagiotelis, Ioannis Panapakidis, Jose M. Pavía, Manuela Pedio, Diego J. Pedregal, Pierre Pinson, Patrícia Ramos, David E. Rapach, J. James Reade, Bahman Rostami-Tabar, Michał Rubaszek, Georgios Sermpinis, Han Lin Shang, Evangelos Spiliotis, Aris A. Syntetos, Priyanga Dilini Talagala, Thiyanga S. Talagala, Len Tashman, Dimitrios Thomakos, Thordis Thorarinsdottir, Ezio Todini, Juan Ramón Trapero Arenas, Xiaoqian Wang, Robert L. Winkler, Alisa Yusupova, and Florian Ziel. Forecasting: theory and practice. *International Journal of Forecasting*, 38(3):705–871, jul 2022.

Edouard Pineau, Sebastien Razakarivony, and Thomas Bonald. Seq2VAR: multivariate time series representation with relational neural networks and linear autoregressive model. In *AALTD workshop, ECML/PKDD : 4th Workshop on Advanced Analytics and Learning on Temporal Data*, Würzburg, Germany, 2019.

Pier Francesco Procacci and Tomaso Aste. Forecasting market states. *Quantitative Finance*, 19(9): 1491–1498, July 2019. ISSN 1469-7696.

Xingyue Pu, Tianyue Cao, Xiaoyun Zhang, Xiaowen Dong, and Siheng Chen. Learning to learn graph topologies. *Advances in Neural Information Processing Systems*, 34:4249–4262, 2021.

Aravind Rajeswaran, Chelsea Finn, Sham Kakade, and Sergey Levine. Meta-learning with implicit gradients. *arxiv preprint arxiv:1909.04630*, 2019.

Joao A Candido Ramos, Lionel Blondé, Stéphane Armand, and Alexandros Kalousis. Conditional neural relational inference for interacting systems. *arXiv preprint arXiv:2106.11083*, 2021.

Fei Ren, Shen-Dan Ji, Mei-Ling Cai, Sai-Ping Li, and Xiong-Fei Jiang. Dynamic lead–lag relationship between stock indices and their derivatives: A comparative study between chinese mainland, hong kong and us stock markets. *Physica A: Statistical mechanics and Its applications*, 513:709–723, 2019.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pages 4334–4343. PMLR, 2018.

Barbara Rossi. Advances in forecasting under instability. In *Chapter 21*, volume 2 of *Handbook of Economic Forecasting*, pages 1203–1324. Elsevier, 2013.

Donald B Rubin. Causal inference using potential outcomes. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

Leonidas Sandoval. Structure of a global network of financial companies based on transfer entropy. *Entropy*, 16(8):4443–4482, 2014. ISSN 10994300.

Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. *Advances in Neural Information Processing Systems*, 30, 2017.

Anil K Seth, Adam B Barrett, and Lionel Barnett. Granger causality analysis in neuroscience and neuroimaging. *Journal of Neuroscience*, 35(8):3293–3297, 2015.

Cosma Rohilla Shalizi, Abigail Z Jacobs, Kristina Lisa Klinkner, and Aaron Clauset. Adapting to non-stationarity with growing expert ensembles. *arXiv preprint arXiv:1103.0949*, 2011.

Chao Shang, Jie Chen, and Jinbo Bi. Discrete graph structure learning for forecasting multiple time series. *arXiv preprint arXiv:2101.06861*, 2021.

Oleksandr Shchur, Caner Turkmen, Nick Erickson, Huibin Shen, Alexander Shirkov, Tony Hu, and Yuyang Wang. AutoGluon-TimeSeries: AutoML for probabilistic time series forecasting. *arXiv preprint arXiv:2308.05566*, 2023.

Danni Shi, Jan-Peter Calliess, and Mihai Cucuringu. Multireference alignment for lead-lag detection in multivariate time series and equity trading. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pages 507–515, 2023.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

Ali Shojaie and Emily B Fox. Granger causality: A review and recent advances. *Annual Review of Statistics and Its Application*, 9:289–319, 2022.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-Weight-Net: Learning an explicit mapping for sample weighting. *Advances in Neural Information Processing Systems*, 32:1–23, 2019.

Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2): 276–295, 2017.

Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems*, volume 33, pages 7462–7473. Curran Associates, Inc., 2020.

Stephen M Smith, Karla L Miller, Gholamreza Salimi-Khorshidi, Matthew Webster, Christian F Beckmann, Thomas E Nichols, Joseph D Ramsey, and Mark W Woolrich. Network modelling methods for fMRI. *Neuroimage*, 54(2):875–891, 2011.

Olaf Sporns. *Networks of the Brain*. MIT press, 2016.

Petar Stojanov, Mingming Gong, Jaime Carbonell, and Kun Zhang. Low-dimensional density ratio estimation for covariate shift correction. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3449–3458. PMLR, 2019.

Alex Tank, Ian Covert, Nicholas Foti, Ali Shojaie, and Emily B Fox. Neural Granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021.

Mathias Niemann Tygesen, Francisco Camara Pereira, and Filipe Rodrigues. Unboxing the graph: Towards interpretable graph neural networks for transport prediction through neural relational inference. *Transportation Research Part C: Emerging Technologies*, 146:103946, 2023. ISSN 0968-090X.

Matthew J Vowels, Necati Cihan Camgoz, and Richard Bowden. D'ya like DAGs? a survey on structure learning and causal discovery. *ACM Computing Surveys*, 55(4):1–36, 2022.

Xingchen Wan, Jie Yang, Slavi Marinov, Jan-Peter Calliess, Stefan Zohren, and Xiaowen Dong. Sentiment correlation in financial news networks and associated market movements. *Scientific Reports*, 11(1):3062, 2021.

Yueming Wang, Kang Lin, Yu Qi, Qi Lian, Shaozhe Feng, Zhaohui Wu, and Gang Pan. Estimating brain connectivity with varying-length time lags using a recurrent neural network. *IEEE Transactions on Biomedical Engineering*, 65(9):1953–1963, 2018.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi. Deeptime: Deep time-index meta-learning for non-stationary time-series forecasting. *arXiv preprint arXiv:2207.06046*, 2022.

Di Wu, Yiping Ke, Jeffrey Xu Yu, Philip S. Yu, and Lei Chen. Detecting leaders from correlated time series. *International Conference on Database Systems for Advanced Applications*, 5981 LNCS: 352–367, 2010. ISSN 03029743.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph WaveNet for deep spatial-temporal graph modeling. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 1907–1913. International Joint Conferences on Artificial Intelligence Organization, 7 2019.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 753–763, 2020.

Ru Xiao, Chaoqun Ma, and Xianhua Mi. The time-varying lead-lag relationship between index futures and the cash index and its factors. *Economic research-Ekonomska istraživanja*, 36(1):1549–1569, 2023.

Ruichao Xiao, Manish Kumar Singh, and Rose Yu. Dynamic relational inference in multi-agent trajectories. *arxiv preprint arxiv:2007.13524*, 2020.

Zexuan Yin and Paolo Barucca. Deep recurrent modelling of Granger causality with latent confounding. *Expert Systems with Applications*, 207:118036, 2022. ISSN 0957-4174.

Yong Luo Yong Tang, Jason Jie Xiong and Yi-Cheng Zhang. How do the global stock markets influence one another? Evidence from finance big data and Granger causality directed network. *International Journal of Electronic Commerce*, 23(1):85–109, 2019.

Yue Yu, Jie Chen, Tian Gao, and Mo Yu. DAG-GNN: DAG structure learning with graph neural networks. In *International Conference on Machine Learning*, pages 7154–7163. PMLR, 2019.

Alisa Yusupova, Nicos G. Pavlidis, and Efthymios G. Pavlidis. Dynamic linear models with adaptive discounting. *International Journal of Forecasting*, 2022. ISSN 01692070.

Tianyi Zhang, Ikko Yamane, Nan Lu, and Masashi Sugiyama. A one-step approach to covariate shift adaptation. *SN Computer Science*, 2(4):65–80, 2021. ISSN 26618907.

Yichi Zhang, Mihai Cucuringu, Alexander Y Shestopaloff, and Stefan Zohren. Dynamic time warping for lead-lag relationships in lagged multi-factor models. *arXiv preprint arXiv:2309.08800*, 2023.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. DAGs with NO TEARS: Continuous optimization for structure learning. *Advances in Neural Information Processing Systems*, 31, 2018.

Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81, 2020. ISSN 2666-6510.

Jingxuan Zhu, Juexin Wang, Weiwei Han, and Dong Xu. Neural relational inference to learn long-range allosteric interactions in proteins from molecular dynamics simulations. *Nat Commun*, 13: 1661, 2022.

Indre Zliobaite, Mykola Pechenizkiy, and Joao Gama. *An Overview of Concept Drift Applications*, pages 91–114. Springer International Publishing, 2016.