Contents lists available at ScienceDirect

International Journal of Forecasting

journal homepage: www.elsevier.com/locate/ijforecast

Granger causality detection in high-dimensional systems using feedforward neural networks

Hector Calvo-Pardo [a,b,c], Tullio Mancini [a], Jose Olmo [a,d,*,1]

[a] *Department of Economics, University of Southampton, Highfield Campus, SO17 1BJ, Southampton, UK*
[b] *Centre for Economic and Policy Research (CEPR), UK*
[c] *Centre for Population Change (CPC), UK*
[d] *Department of Economic Analysis, Universidad de Zaragoza, Gran Vía 2, 50005, Zaragoza, Spain*

ARTICLE INFO

Keywords:
Granger causality
Lasso penalty function
Mutual information
Neural networks
Sparsity

ABSTRACT

This paper proposes a novel methodology to detect Granger causality on average in vector autoregressive settings using feedforward neural networks. The approach accommodates unknown dependence structures between elements of high-dimensional multivariate time series with weak and strong persistence. To do this, we propose a two-stage procedure: first, we maximize the transfer of information between input and output variables in the network in order to obtain an optimal number of nodes in the intermediate hidden layers. Second, we apply a novel sparse double group lasso penalty function in order to identify the variables that have the predictive ability and, hence, indicate that Granger causality is present in the others. The penalty function inducing sparsity is applied to the weights that characterize the nodes of the neural network. We show the correct identification of these weights so as to increase sample sizes. We apply this method to the recently created Tobalaba network of renewable energy companies and show the increase in connectivity between companies after the creation of the network using Granger causality measures to map the connections.

1. Introduction

The concept of causality introduced by Wiener (1956) and Granger (1969) constitutes a basic notion for analyzing dynamic relationships between time series. When studying this type of statistical causality, predictability is the central issue, and is of great importance to economists, policymakers, and investors. A broad definition of Granger causality is based on detecting whether a variable or a group of variables helps to reduce the mean square forecast error of a univariate or multivariate prediction; see Geweke (1982, 1984), Dufour and Taamouti (2010), and, more recently, Song and Taamouti (2018) in a high-dimensional setting.

A natural parametric model to detect the presence of predictive ability in a multivariate setting is the family of Vector Autoregressive (VAR) models introduced by Sims (1980) in a seminal paper. The choice of this parametric approach has two inherent problems: the occurrence of overparametrization in large dimensions, and the incorrect specification of the relationship between the variables in the linear VAR model if the true data generating process that determines the interactions between the variables is nonlinear or, more generally, unknown. The different procedures suggested by the literature to overcome the "profligate parametrization" that can affect high-dimensional VARs are classified as dimensionality reduction and sparsity induction via convex regularizers.

[*] Corresponding author at: Department of Economics, University of Southampton, Highfield Campus, SO17 1BJ, Southampton, UK.

E-mail addresses: J.B.Olmo@soton.ac.uk, joseolmo@unizar.es (J. Olmo).

[1] Jose Olmo acknowledges financial support from Fundación Agencia Aragonesa para la Investigación y el Desarrollo (ARAID).

https://doi.org/10.1016/j.ijforecast.2020.10.004

In the first group, the literature tries to solve the overparametrization that may affect VARs by reducing the dimensionality of vector time series models such as canonical correlation analysis (Box & Tiao, 1977), factor models (Peña & Box, 1987), Bayesian models (e.g., Bańbura et al., 2010; Koop, 2013), principal component analysis (Stock & Watson, 2002), and generalized dynamic factor models (Forni et al., 2000), among many other statistical techniques. The main limitation of these approaches lies in the loss of interpretability due to the transformations involved in most of the methods that make impossible to track the Granger causal interactions between the "original" multivariate time series (Géron, 2017).

Recently, statistical and machine learning literature has instead focused on imposing sparsity in the estimated model coefficients through the use of convex loss functions such as the least absolute shrinkage and selector operator (hereafter referred to as Lasso; Tibshirani, 1996). The primitive version of this approach reduces the dimension of the problem by deleting individual regressors. More sophisticated versions such as the group lasso penalty (Yuan & Lin, 2006) reduce the dimension of the problem by jointly deleting groups of variables. None of these approaches, however, explicitly consider the structure of the dependence in multivariate time series processes; in particular, these methods do not consider the pivotal role that the correct specification of the order of the VARs plays in detecting Granger causal interactions. To overcome this limitation and accommodate penalty functions that explicitly consider the appropriate number of lags in the system, Nicholson et al. (2014) suggest a Hierarchical Group Lasso approach that allows not only for automatic variable selection but also for automatic lag selection. Another noteworthy example of Granger causality discovery in high-dimensional linear VARs is the paper by Skripnikov and Michailidis (2019), where the authors propose a generalized sparse fused lasso optimization criterion for jointly estimating multivariate VARs. The novel lasso-based optimization procedure developed by these authors not only enables the introduction of sparsity but also encourages similarities between transition matrices, ultimately allowing for both joint estimation and Granger causality detection in multiple VARs. The current literature has proposed robust procedures for the estimation of Granger causality in high-dimensional linear VARs through sparsity induction via convex regularizers; however, the identification of correct inferential procedures based on Granger causality testing is still not fully untangled.[2] A first step towards correct inferential procedures is proposed by Hecq et al. (2019) and extends the post-double selection approach of Belloni et al. (2014) to Granger causality testing in linear sparse high-dimensional VARs.[3] This procedure enables the retention of the correct size after the variable selection of the lasso and is shown to perform well in different data generating processes.

The presence of nonlinearities in the dynamic relationship between the variables is another problem that has not yet properly been studied in the analysis of Granger causality. Taamouti et al. (2014) propose nonparametric estimation and inference for conditional density-based Granger causality measures that quantify linear and nonlinear Granger causalities. These authors transform the Granger causality measures in terms of copula densities. More recently, Song and Taamouti (2018) propose model-free measures for Granger causality in the mean between random variables. Unlike the existing measures, these methods are able to detect and quantify nonlinear causal effects. The new measures are based on nonparametric regressions and are consistently estimated by replacing the unknown mean square forecast errors with their nonparametric kernel estimates.

Granger causality is a prediction problem. A powerful methodology for prediction in regression models and, more specifically, forecasting multivariate time series is neural networks. Empirical research shows that Artificial Neural Networks are characterized by high accuracy when used to forecast nonlinear multivariate time series (Chakraborty et al., 1992; Kaastra & Boyd, 1996).[4] More generally, deep learning methods based on training large neural networks have proven very successful in many high-dimensional problems such as pattern recognition, biomedical diagnosis, and others; see Schmidhuber (2015) and LeCun et al. (2015) for overviews of the topic. Athey and Imbens (2019) provide a recent literature review of applications and contributions to and from economics and econometrics.

The main impediment for neural network models to be considered as a standard tool for time series analysis is the lack of interpretation. This is due to the fact that the effects of inputs are difficult to quantify precisely due to the tangled web of interacting nodes between and across hidden layers. There is, however, some recent progress in this area. In particular, Scardapane et al. (2017) propose a methodology that adds interpretability to neural network structures by imposing a mapping between the "original" variables and the nodes of the first hidden layer of the neural network. An "original" variable in a model is considered irrelevant if the corresponding nodes that carry information from the variable to the neural network are pruned. Pruning nodes in the first layer of the neural network is equivalent to deleting variables from a regression model. Tank et al. (2018) were the first authors to apply this strategy to the specific time series problem of detecting Granger causality. These authors use a hierarchical group lasso penalty function (see Yuan & Lin, 2006) on the weights of the neural network, but their work is lacking in terms of the impact of the architecture of the network on Granger causality discovery.

The aim of the current paper is to propose a methodology based on neural networks to detect mean Granger

[2] Wilms et al. (2016) propose a bootstrap-based Granger causality test which ignores the uncertainty regarding the selection step and thus does not account for post-selection issues.

[3] Another important example that can be found in the literature is the research conducted by Song and Taamouti (2019). The authors propose correct inferential procedures for Granger causality testing in high-dimensional systems modelled by factor models as opposed to high-dimensional VARs.

[4] *Universal Approximation Theorem* by Cybenko (1989) analyzed by Hornik (1991).

causality for vectors of variables in which the dynamic dependence structure is unknown and can take very general forms that accommodate, in turn, linear and nonlinear VAR models with a potentially high-dimensional number of variables and lags. In contrast to most of the literature on neural networks, we add interpretability to the neural network by applying Scardapane et al.'s strategy (2017) to a time series setting. More specifically, we construct a neural network with an input layer given by the vector of *regressors* and an output layer given by the vector of *dependent* variables. The magnitude of the weights associated to the nodes in the first layer determines the presence of Granger causality between the input and output variables. More formally, the interpretability of the network is given by the existence of a mapping between the regressors and the nodes in the first hidden layer. A particular input variable will be relevant for predicting an output variable if there are connections from the corresponding input node to any node in the first hidden layer. The Granger causality of a variable involves checking the connections between all possible lags and all possible nodes in the first hidden layer such that a variable will not Granger cause another variable if there are no connections leaving from any of the input variables to any of the nodes in the first hidden layer. In contrast, the number of nodes in the intermediate hidden layers is not directly related to the definition and interpretation of Granger causality. The relevant intermediate nodes are obtained by optimizing the flow of information from the input variables to the output variables in the neural network, maximizing the mutual information transfer/minimizing information loss. More formally, we show that the optimal choice of the number of nodes in the intermediate hidden layers improves model selection and reduces type I and II errors in the Granger causality detection methods.

Our method allows for a large number of variables and lags. In this setting, the number of input nodes can be very large, rendering standard estimation and model selection methods unfeasible. We instead propose a novel sparse double group lasso penalty function that allows for the estimation of the weights that characterize the transfer of information through the neural network and model selection: Granger causality and lag selection. Our double group lasso penalty function considers all possible lags of a specific regressor and all possible nodes of the first hidden layer connecting to such a regressor as a first group. The second group considers separately all possible nodes of the first hidden layer that connect to a specific lag of a specific regressor. This is the proposed approach for detecting the optimal number of lags of a given input variable influencing each output variable.

Our sparse double group lasso penalty function extends the penalty functions proposed in Simon et al. (2013) for multivariate regression models and Scardapane et al. (2017) for neural networks. Both hierarchical and sparse group lasso procedures for detecting Granger causality allow specifying a different number of lags across variables in the vector. However, in contrast to the hierarchical group lasso, our novel objective function imposes a lower penalty function on the parameters of the model at the same time as guaranteeing model selection consistency. By doing so, we make sure we exclude those variables without the ability to predict the response variables without excluding important interactions between the variables once a group is not deleted; that is, once a variable is shown to cause another Granger variable.

To the best of our knowledge, this, together with Tank et al. (2018), is the first study that considers Granger causality in a very general setting - an unknown dependence structure between the variables - using neural networks. Our method differentiates from the Tank et al. (2018) study in two main ways. First, we propose an optimal network structure obtained from applying Montgomery and Eledath's algorithm (1995), and, second, we consider a different lasso penalty function that operates differently from the hierarchical group lasso; that is, in each hidden layer we only use those nodes that carry information from the input layer to the output layer, removing unnecessary nodes. The optimality of the neural network has a direct effect on the properties of our Granger causality procedure. In particular, we reduce the type I error, interpreted in this context as spurious Granger causality. An excessive number of nodes can lead to lasso-type penalty functions that spuriously identify non-existing interactions among the input nodes.

The paper also discusses results on parameter identification and model selection consistency as the sample size increases. We derive the conditions that determine the inclusion or non-inclusion of a parameter or group of parameters in the model. Our conditions for model selection coincide with those found in the literature on model selection consistency when the number of variables and the number of lags are fixed; in particular, we obtain $\lambda = o(1/T)$ with T being the sample size (see Fan & Li, 2001). Nevertheless, our procedure also achieves model selection consistency when the number of lags k increases with the sample size. In order to guarantee this, we impose $\lambda = o\left(\frac{1}{\sqrt{k_T T}}\right)$, with k_T being the number of lags of the input variables.

In Appendix A, we report a comprehensive Monte-Carlo simulation exercise that shows the performance of our methodology in the detection of Granger causality. First, we assess the type I and type II errors in our detection procedure in finite samples and compare it with a method that does not optimize the structure of the neural network. Second, we compare the performance of our proposed sparse double group lasso against the hierarchical group lasso. Both sets of results provide clear evidence of the outstanding performance of our methodology for detecting Granger causality in terms of the probability of type I and type II errors. These results hold for short- and long-range dependence and for linear and nonlinear VAR specifications. We also show the consistency of our approach for model selection for increasing sample sizes.

The suggested methodology is then applied to detect the interconnections between energy companies trading in the recently created Tobalaba network. The Tobalaba network is a test-net provided by the Energy Web Foundation (2018) that connects renewable energy companies via a blockchain platform. More specifically, we exploit recent work on social and financial networks that identifies the presence of connections in a network through the presence of Granger causality between their nodes; see

Billio et al. (2012) and Hecq et al. (2019). In our setting, we propose our two-stage neural network approach for detecting Granger causal relationships between the financial returns of the energy companies trading in the Tobalaba network.

The World Bank Group (2018) argues that the decentralization, disintermediation, increase in information symmetry, and cost reduction via smart contracts will allow smaller participants to enter the market, increasing the number of bilateral transactions and ultimately diversifying the market structure. The objective of our application is to corroborate the World Bank Group's hypotheses by gauging the interconnectivity between energy firms before and after the introduction of Tobalaba. To do this, we construct two Granger causal networks (before and after the introduction of Tobalaba) and apply the proposed algorithm to each vertex. The empirical study reveals an increase in the number of connections among the members of the Tobalaba network after the introduction of the blockchain platform. We explore the implications of our methodology for forecasting purposes. To do this, we implement the Diebold–Mariano (1995) predictive ability test and find overwhelming empirical evidence supporting the outperformance of our approach in terms of the predictive ability compared to VAR models of different dimensions.

The rest of the paper is organized as follows: Section 2 presents the structure of the neural network and formulates the Granger causality detection procedure in this setting. Section 3 discusses the estimation and model selection using a two-stage procedure based on a novel sparse double group lasso penalty function. Section 4 discusses parameter identification and model selection consistency when the number of lags is fixed, as well as when it increases with the sample size. In Section 5, we apply our novel procedure for detecting Granger causality to the financial returns of the set of renewable energy firms trading in the recently created Tobalaba network. Section 6 concludes the paper. Following this, Appendix A presents a Monte-Carlo simulation exercise that provides empirical evidence in finite samples of the performance of our method in detecting Granger causality and model selection consistency for increasing sample sizes. Finally, Appendix B provides a formal parametric definition of Granger causality in a fully connected neural network framework.

2. Granger causality in neural networks

Let $\{\mathbf{x}_t \in \mathbb{R}^p\}_{t=1}^T$ denote a p-dimensional vector time series of length T. Our goal is to study Granger causality in mean. For this, the relevant loss function is the mean square forecast error. The vector of random variables $\mathbf{x}_t$ evolves according to the following dynamics, which are defined componentwise. Thus, for each x_{it}:

$$x_{it} = g_i(\mathbf{x}_{t-1}, \dots, \mathbf{x}_{t-k}) + \epsilon_{it}, \text{ for } i = 1, \dots, p, \tag{1}$$

where $g_i(\cdot)$ is a function that captures the dependence structure between the dependent variable x_{it} and the lags of the vector $\mathbf{x}_t$. The quantity ϵ_{it} is a martingale difference sequence satisfying $E[\epsilon_{it} \mid \Im_{t-1}] = 0$, with $\Im_{t-1}$ denoting the sigma-algebra containing all the information available to the individual at the time t. We further assume that the sigma-algebra $\Im_{t-1}$ can be approximated by the finite set $\mathbf{X}_{t-1}$, with $\mathbf{X}_{t-1} = [\mathbf{x}_{t-1} \dots \mathbf{x}_{t-k}]$ being a matrix of dimension $(p \times k)$ that contains the relevant information set. Then, $\Im_{t-1} \equiv \mathbf{X}_{t-1}$ such that $E[x_{it} \mid \Im_{t-1}] = g_i(\mathbf{X}_{t-1})$. In addition, it is also implicit that $k = k_i$ as we investigate the possibility of using different lags for different components.

There are different approaches to modelling the function $g_i(\mathbf{X}_{t-1})$ for $i = 1, \dots, p$. This paper builds on the recent literature on introducing techniques for adding interpretability to neural networks and proposes a feedforward neural network to model each function $g_i(\cdot)$ separately. Each feedforward neural network has N hidden layers, and the vector $\mathbf{h}_n$ denotes the values of the hidden layers obtained from z_n hidden nodes in the nth hidden layer. We use $g_i(\mathbf{X}_{t-1}; {}^i\mathbf{W}, \mathbf{z})$ to denote the function $g_i(\mathbf{X}_{t-1})$. By doing so, we incorporate, as additional arguments of the function, the matrix ${}^i\mathbf{W}$, which contains all the weights with the information carried through the nodes in the hidden layers for predicting the output variable x_{it}; and the vector $\mathbf{z} = (z_1, \dots, z_N)^\top$, which contains the number of nodes in each hidden layer.

In this framework, we propose $i = 1, \dots, p$ different neural networks to measure the relationship between each variable x_{it} and the matrix of the input variables $\mathbf{X}_{t-1}$. Furthermore, each submatrix ${}^i\mathbf{W}^1$ contains the weights associated with the nodes in the first hidden layer. These weights connect the vector of input variables to the first hidden layer with z_1 nodes. Let us focus on the output variable x_{it}. In this case, the matrix of the weights relevant for gauging Granger causality and characterizing the first hidden layer is:

$$\underset{(z_1 \times kp)}{{}^i\mathbf{W}^1} = \begin{bmatrix} {}^iw_{11}^{1(1)} & \dots & {}^iw_{11}^{1(k)} & & {}^iw_{1p}^{1(1)} & \dots & {}^iw_{1p}^{1(k)} \\ \dots & & \dots & \dots & \dots & & \dots \\ {}^iw_{z_1 1}^{1(1)} & \dots & {}^iw_{z_1 1}^{1(k)} & & {}^iw_{z_1 p}^{1(1)} & \dots & {}^iw_{z_1 p}^{1(k)} \end{bmatrix}.$$

For the intermediate hidden layers, the matrices ${}^i\mathbf{W}^n$, with $n = 2, \dots, N$, are constructed similarly; however, in this case, the connections are between a layer of z_{n-1} nodes and a layer of z_n nodes such that the dimensions of the matrix ${}^i\mathbf{W}^n$ need to be adapted. In what follows, we drop the superscript i and denote the matrix with the weights of the neural network as $\mathbf{W} = [\mathbf{W}^1; \dots; \mathbf{W}^N]$.

The information in a neural network flows across layers by means of activation functions θ. Let $\widetilde{\mathbf{x}}_{t-1}$ be a vector of dimension $kp \times 1$ that stacks all the elements of the matrix $\mathbf{X}_{t-1}$ from the input variables. For a given bias parameter $\mathbf{b}_1 \in \mathbb{R}^{z_1}$ and the activation vector-valued function $\boldsymbol{\theta}(\cdot) : \mathbb{R}^{z_1} \to \mathbb{R}^{z_1}$, the values at the first hidden layer $\mathbf{h}_1 \in \mathbb{R}^{z_1}$ are:

$$\begin{aligned} \mathbf{h}_1 &= \boldsymbol{\theta}\left(\mathbf{W}^1 \widetilde{\mathbf{x}}_{t-1} + \mathbf{b}_1\right) \\ &= \boldsymbol{\theta}\left(\begin{array}{c} \dots \\ \sum_{j=1}^{p} \sum_{l=1}^{k} w_{zj}^{1(l)} x_{j,t-l} + b_{1z} \\ \dots \end{array} \right). \end{aligned} \tag{2}$$

The values of the activation functions at the intermediate hidden layers, $\mathbf{h}_n \in \mathbb{R}^{z_n}$, are given by:

$$\mathbf{h}_n = \boldsymbol{\theta}\left(\mathbf{W}^n \mathbf{h}_{n-1} + \mathbf{b}_n\right), n = 2 \dots N, \tag{3}$$

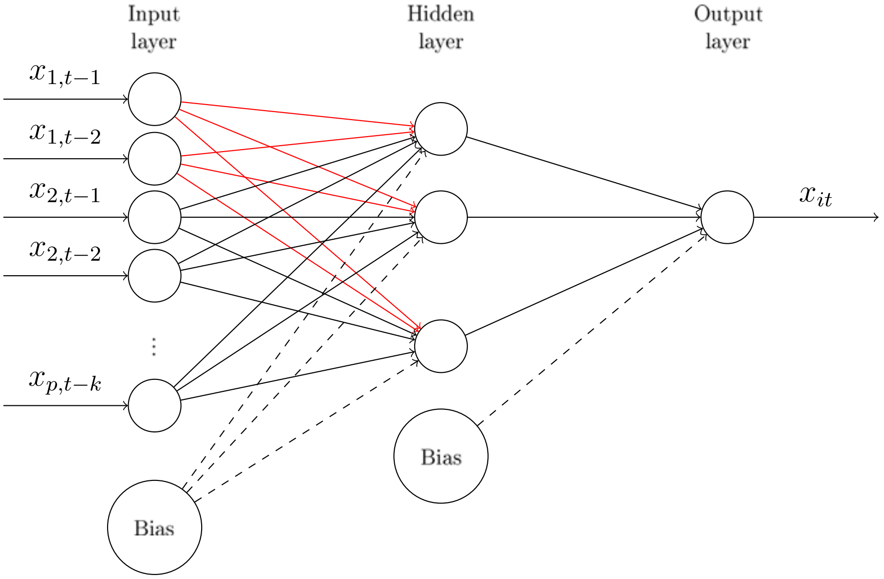


Fig. 1. Neural Granger causality.

where $\mathbf{b}_n \in \mathbb{R}^{z_n}$, $\mathbf{h}_{n-1} \in \mathbb{R}^{z_{n-1}}$ and hence $\mathbf{W}^n \in \mathbb{R}^{z_n} \times \mathbb{R}^{z_{n-1}}$ is the matrix of the weights in the hidden layer n with z_n rows and z_{n-1} columns. Since the vector-valued activation function $\boldsymbol{\theta}(\cdot)$ proceeds element-wise, we denote by $\theta_z(\cdot) : \mathbb{R} \to \mathbb{R}$ with $z = 1, \ldots, z_n$ each of its components that corresponds to each node in the hidden layer n. Therefore, the one-period-ahead forecast of the time series x_{it} is:

$$g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z}) = \boldsymbol{\omega}_O^\top \mathbf{h}_N + b_O, \tag{4}$$

where b_O is a constant, $\boldsymbol{\omega}_O \in \mathbb{R}^{z_N}$ is the vector of weights connecting the last hidden layer N to the output node, and $\mathbf{h}_N \in \mathbb{R}^{z_N}$ is the vector of values at the last hidden layer, which can be expressed in terms of the input data as:

$$\mathbf{h}_N = \boldsymbol{\theta}\left(\mathbf{W}^N \ldots \boldsymbol{\theta}\left(\mathbf{W}^1 \widetilde{\mathbf{x}}_{t-1} + \mathbf{b}_1\right) \ldots + \mathbf{b}_N\right), \tag{5}$$

where $\mathbf{W}^N \in \mathbb{R}^{z_N} \times \mathbb{R}^{z_{N-1}}$ and $\boldsymbol{\theta}(\cdot) : \mathbb{R}^{z_N} \to \mathbb{R}^{z_N}$. Hence,

$$\begin{aligned} &g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z}) \\ &\quad = \boldsymbol{\omega}_O^\top \boldsymbol{\theta}\left(\mathbf{W}^N \ldots \boldsymbol{\theta}\left(\begin{array}{c} \ldots \\ \sum_{j=1}^{p} \sum_{l=1}^{k} w_{zj}^{1(l)} x_{j,t-l} + b_{1z} \\ \ldots \end{array}\right) \cdots + \mathbf{b}_N\right) \\ &\qquad + b_O. \end{aligned} \tag{6}$$

Eq. (6) expresses the function $g_i(\cdot)$ in terms of a vector-valued activation function $\boldsymbol{\theta}(\cdot)$ applied to a linear combination of the input nodes. This equation adds interpretability to the neural network through the connections between the input variables and the nodes in the first hidden layer. A variable x_{jt} does not Granger-cause the variable x_{it} if all the weights connecting all the lags of x_{jt} in the model and all the nodes in the first hidden layer are zero. Appendix B provides a formal parametric definition of Granger causality in a fully connected neural network framework. The null hypothesis of no Granger causality of x_{jt} to x_{it} is:

$$\mathcal{H}_0 : w_{1j}^{1(1)} = \cdots = w_{1j}^{1(k)} = \cdots w_{z_1 j}^{1(1)} = \cdots = w_{z_1 j}^{1(k)} = 0, \tag{7}$$

and the alternative is:

$$\mathcal{H}_A : \text{some } w_{nj}^{1(l)} \neq 0, \text{ for } n = 1, \ldots, z_1 \text{ and } l = 1, \ldots, k. \tag{8}$$

Fig. 1 provides a visual representation of the intuition behind the test for Granger causality based on feedforward neural networks for a multivariate time series with one hidden layer. If the group lasso penalty penalizes the weights highlighted in red to zero, the variable x_{1t} does not Granger cause series x_{it}. Thus, Granger causality is inferred from the sparsity introduced in the first layer by the group lasso penalty.

3. Estimation and model selection

We propose a methodology to detect Granger causality using a neural network for each element of the vector $\mathbf{x}_t$. Our procedure is performed in two stages. For a given sample, we first obtain the optimal number of nodes in the feedforward neural network by minimizing the information loss across layers. We focus on the first hidden layer z_1 given that in the second stage we plug in the optimal quantity z_1 in a lasso-type function, penalizing the weights of the neural network. Minimization of the corresponding regularization problem has two objectives. First, it uncovers the input variables that are relevant for forecasting the output variables (Granger causality) and, second, it allows us to establish the optimal number of lags affecting the mean square forecast error.

Our approach differentiates from the recent literature on interpretable neural networks (see Hastie et al. (2005), Scardapane et al. (2017), or Tank et al. (2018)) in two main aspects. First, in our case, concretely, as per Algorithm 1, we choose the optimal number of nodes in all hidden layers that minimize the information loss through the neural network deep architecture. This has been done in the literature by maximizing the mutual information transfer between input and output nodes (see Schreiber (2000)) or, similarly, by minimizing the loss of information through the neural network (see De Veciana and Zakhor (1992), Montgomery and Eledath (1995), Reed et al. (1995), or more recently, Urban (2017)). In order to fully exploit these theories and construct an optimal architecture for the neural network that minimizes the information loss, we need to introduce uncertainty into the neural network. This is done by injecting noise into the model. In this case the optimal neural network is constructed through noise jittering.

Second, we propose a regularization function that extends the mean square error loss function for fitting a neural network by penalizing the weights associated with the nodes in the first hidden layer. In contrast to the literature, we propose a double group lasso regularization that penalizes Granger causal relations separately across groups and lag selection within groups.

3.1. Stage 1: Choosing the optimal neural network

In the first stage, we optimize the neural network by choosing a number of nodes per hidden layer that maximizes the transfer of information/minimizes information loss. To do this, we follow the above literature and inject noise into the neural network. In what follows, we adapt these methods to our setting. Our base loss function is the sample mean square error, which is defined as:

$$\frac{1}{T}\left\|x_{it}-g_i(\mathbf{X}_{t-1};\mathbf{W},\mathbf{z})\right\|_2^2 . \tag{9}$$

In order to be able to apply the different information criteria above, we introduce noise into the system by constructing noisy replicas of our sample, as in De Veciana and Zakhor (1992), Montgomery and Eledath (1995), Reed et al. (1995), or Urban (2017). This approach can be interpreted as a procedure to regularize the neural network applied to a population and not only to a given sample. Let $x_{jt}^* = x_{jt}+v_{jt}$, with v_{jt} an *iid* realization of a $\mathcal{N}(0,\sigma_v^2)$ random variable, and let $\mathbf{X}_t^*$ be the corresponding matrix. The objective function (9) applied to these iid random copies of the original observations becomes:

$$\frac{1}{T}\sum_{t=1}^{T}\left(x_{it}-g_i(\mathbf{X}_{t-1}^*;\mathbf{W},\mathbf{z})\right)^2 . \tag{10}$$

In what follows, we decompose the mean square error (10) into two components: a first component given by the mean square error of the original data, and a second component given by introducing noise into the model. To do this, we consider first the case of a single hidden layer $\mathbf{W}=\mathbf{W}^1$. Let the objective function be $g_i\left(\mathbf{X}_{t-1}^*;\mathbf{W},\mathbf{z}\right) = \boldsymbol{\omega}_0^\top \mathbf{h}_1^* + b_0$ with:

$$\mathbf{h}_1^* = \boldsymbol{\theta}\begin{pmatrix} \cdots \\ \sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}+\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}v_{j,t-l} \\ \cdots \end{pmatrix}. \tag{11}$$

For simplicity, we work with the activation function element-wise, such that $g_i\left(\mathbf{X}_{t-1}^*;\mathbf{W},\mathbf{z}\right) = \sum_{z=1}^{z_1}\omega_{0z}\theta_z \left(\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}+\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}v_{j,t-l}\right)+b_O$. Applying a Taylor expansion of the first order to each activation function $\theta_z(\cdot)$ around the deterministic component $\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}$, we obtain:

$$\begin{aligned}
&\theta_z\left(\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}+\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}v_{j,t-l}\right)\\
&\quad\approx \theta_z\left(\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}\right)\\
&\quad\quad+\dot{\theta}_z\left(\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}\right)\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}v_{j,t-l},
\end{aligned} \tag{12}$$

with $\dot{\theta}_z(\cdot)$ the first derivative of $\theta_z(\cdot)$. Note that for the standard activation functions proposed in the related literature, the second derivative of $\theta(\cdot)$ is close to zero along the support of the function; therefore, a first order expansion is sufficient to accurately approximate the activation function. Then, the objective function becomes:

$$\begin{aligned}
&g_i\left(\mathbf{X}_{t-1}^*;\mathbf{W},\mathbf{z}\right)\approx \boldsymbol{\omega}_0^\top h_1 + b_O\\
&\quad+\sum_{z=1}^{z_1}\omega_{0z}\dot{\theta}_z\left(\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}\right)\\
&\quad\times\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}v_{j,t-l},
\end{aligned}$$

and the loss function (10) can be decomposed as:

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\left(x_{it}-g_i(\mathbf{X}_{t-1}^*;\mathbf{W},\mathbf{z})\right)^2\\
&\quad+\frac{1}{T}\sum_{t=1}^{T}\left(\sum_{z=1}^{z_1}\omega_{0z}\dot{\theta}_z\left(\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}\right)\right.\\
&\quad\left.\times\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}v_{j,t-l}\right)^2\\
&\quad-\frac{2}{T}\sum_{t=1}^{T}\left(x_{it}-g_i(\mathbf{X}_{t-1};\mathbf{W},\mathbf{z})\right)\sum_{z=1}^{z_1}\omega_{0z}\\
&\quad\dot{\theta}_z\left(\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}x_{j,t-l}+b_{1z}\right)\sum_{j=1}^{p}\sum_{l=1}^{k} w_{zj}^{1(l)}v_{j,t-l}.
\end{aligned}$$

Note that the elements inside the outer sum over t are independent but not identically distributed. The randomness is introduced through v_t in all cases so that the mean is zero but the variance varies for each observation depending on the value of $\dot{\theta}(\cdot)$ and the weights $\boldsymbol{\omega}_O$. In this case, we can apply the law of large numbers for independent but not identically distributed random variables, and write the preceding function as the sum of the population mean square error and an additional regularization component. More specifically, as $T \to \infty$, the previous expression converges in probability to the following population quantity:

$$E\left[(x_{it} - g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z}))^2\right] + \sigma_v^2 \sum_{z=1}^{z_1} \omega_{0z} \dot{\theta}_z^2 \left(\sum_{j=1}^{p} \sum_{l=1}^{k} w_{zj}^{1(l)} x_{j,t-l} + b_{1z} \right) \times \sum_{j=1}^{p} \sum_{l=1}^{k} \left(w_{zj}^{1(l)} \right)^2 . \quad (13)$$

The variance of the innovation term, σ_v^2, can be interpreted as the tuning parameter of a regularization component given by the first derivative of the activation function and the magnitude of the weights.

The objective of this procedure is to minimize the noise transmitted through the neural network. This can be done in two ways: (i) by minimizing the nodes operating in the linear region of the activation function, and (ii) by minimizing the weight values in the network. If a node is operating in the saturation region then its output will not be affected as much by the noise. Fig. 2 illustrates the different regions. It is also clear that large weights, $\mathbf{W}$, will also amplify the noise of the output. Also, as noted by Montgomery and Eledath (1995), small weight values in the first hidden layer tend to keep nodes in the linear region, so we may only want to minimize the outgoing weights from each node. Furthermore, the choice of the activation function is another factor to consider. The choice of the *tanh* function, defined as:

$$\theta(x) = \frac{e^{2x} - 1}{e^{2x} + 1}, \quad (14)$$

guarantees that a node operating in the middle of the linear region has an average activation close to zero. Removing a hidden node in this case does not affect the training as much as under other activation functions.

We use these arguments and focus on the significance of each node in each hidden layer rather than on formally minimizing expression (13). In the second stage of our procedure, we will formally minimize the mean square error under Lasso regularization when a VAR structure is considered. In this stage, we assess, indirectly, the contribution of each node to the noise in the neural network by applying a version of the pruning algorithm known as Dynamic Node Removal, developed in Montgomery and Eledath (1995). This method removes hidden units as training progresses. The idea is to keep those nodes that contribute to transmitting information and delete those that transmit noise. The algorithm penalizes the nodes operating in the linear region (low confidence) of $\theta_z(\cdot)$ while accounting for the magnitude of the outgoing weights of the hidden nodes. In our setting, at each Epoch (defined as the pass that the machine learning algorithm has completed), the objective function is:

$$S_{1z} = \frac{\sigma_v^2}{pkT} \sum_{t=1}^{T} \left(\kappa \tanh^2 \left(b_{1z} + \sum_{j=1}^{p} \sum_{l=1}^{k} w_{zj}^{1(l)} x_{jt-l} \right) + \mu \sum_{\tilde{z}=1}^{z_2} \left(\mathbf{w}_{z,\tilde{z}}^2 \right)^2 \right), \quad (15)$$

with $\mathbf{w}_{z,\tilde{z}}^2$ denoting a vector of weights of dimension $(1 \times z_2)$ connecting nodes $z = 1, \ldots, z_1$ in the hidden layer 1 to nodes $\tilde{z} = 1, \ldots, z_2$ in the hidden layer 2. κ and μ are tuning parameters. The objective function is standardized by dividing by the number of observations and the number of nodes in the input layer.

This function measures the *quality* of the nodes in the neural network with regards to information transfer. Thus, if the magnitude of this function is lower than a given threshold χ, the hidden node is then pruned. The choice of the *tanh* activation function over other activation functions such as the *ReLu*, *Exponential ReLu*, or *sigmoidal* ensures that the *principle of minimum information loss* is satisfied. As we are considering a supervised neural network, the minimization of the entropy must ensure the minimization of the output error. Penalizing nodes that operate in the linear region of $\theta(.)$ (element-wise) is equivalent to penalizing nodes of which the output is approximately zero and, thus, nodes that have little impact on the final output of the feedforward neural network. However, as highlighted by Goodfellow et al. (2016) and by Géron (2017), sigmoidal activation functions saturate for high or low values of $\mathbf{h}_n$, incurring in the possible problem of increasing gradient and limiting the training of the neural network (Glorot & Bengio, 2010).[5]

Unfortunately, for neural networks comprising more than one hidden layer, minimizing the mutual information transfer by minimizing the mean square error of the *noisy* version of the data is even more challenging. In this case, we extend the Dynamic Node Removal of Montgomery and Eledath (1995) to higher layers:

$$S_{nz} = \frac{\sigma_v^2}{z_{n-1}} \left(\kappa \tanh^2 \left(b_{nz} + \mathbf{W}_z^n \mathbf{h}_{n-1} \right) + \mu \sum_{\tilde{z}=1}^{z_{n+1}} (\mathbf{w}_{z,\tilde{z}}^{n+1})^2 \right), \quad (16)$$

with $\mathbf{w}_{z,\tilde{z}}^{n+1}$ a row vector of dimension $(1 \times z_{n+1})$ of the matrix $\mathbf{W}^{n+1}$ connecting node $z = 1, \ldots, z_n$ in the hidden layer n to node $\tilde{z} = 1 \ldots z_{n+1}$ in the hidden layer $n+1$.[6] The algorithm that we propose for pruning the neural network is detailed in Algorithm 1.

[5] For saturated values of θ at 0 or 1, the derivative is extremely close to 0, leaving no gradient to propagate through the neural network (Géron, 2017).

[6] We should note that the time index is implicit in the objective function S_{nz}. All observations $\{\mathbf{x}_t\}_{t=1}^{T}$ are used to compute the objective function.

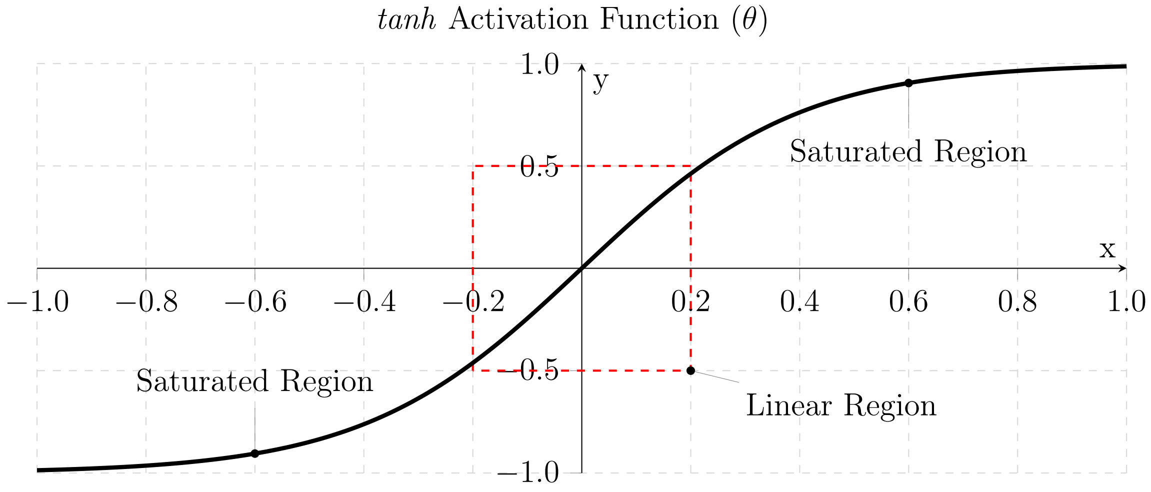


Fig. 2. Linear and Saturated Regions for a *tanh* Activation Function θ.

3.2. Stage 2: Model selection

In this section, we adapt the sparse group lasso proposed by Simon et al. (2013) to neural networks. Our penalty function extends Simon et al. (2013) by considering a double group lasso penalty function. In our case, the groups are defined not only by the nodes in the first hidden layer but also by the number of lags of each input variable.

3.2.1. Sparse double group lasso penalty

The regularization component of our objective function is divided into two components. The first component comprises groups of size kz_1, with k being the number of lags[7] and z_1 the number of nodes in the first hidden layer. This component is specific to the Granger causality detection problem. This function penalizes, as a group, those weights that are associated with a specific input variable and all its lags. To do this, we use the Frobenius norm that extends the L_2 norm to matrices. The penalty function for a specific input variable takes a value of zero if the Frobenius norm is zero:

$$\sum_{j=1}^{p} \left\| \mathbf{W}_j^1 \right\|_F = \sum_{j=1}^{p} \left[\sum_{z=1}^{z_1} \sum_{l=1}^{k} \left(w_{zj}^{1(l)} \right)^2 \right]^{1/2}. \tag{19}$$

The second component is composed of groups of size z_1. This component is used for detecting the optimal number of lags. This function penalizes, as a group, those weights that are associated with a specific lag l of a given input variable. To do this, we use the L_2 norm, penalizing jointly all the nodes in the first hidden layer corresponding to that lag:

$$\sum_{j=1}^{p} \sum_{l=1}^{k} \left\| \mathbf{W}_j^{1(l)} \right\|_2 = \sum_{j=1}^{p} \sum_{l=1}^{k} \left[\sum_{z=1}^{z_1} \left(w_{zj}^{1(l)} \right)^2 \right]^{1/2}. \tag{20}$$

Algorithm 1 Optimal neural network - pruning method.

INPUT: Vector of all input variables, Gaussian noise $v \sim \mathcal{N}(0, \sigma_v^2)$
OUTPUT: Pruned Feedforward Neural Network that maximizes the mutual information transfer.

```
1: procedure N HIDDEN LAYER EXERCISE
2:
3:     Set $\chi = 0.001$ (see Montgomery and Eledath, 1995)
4:     For each epoch E, calculate the significance of the function $h_{n,z}$ as:
```

$$S_{1z} = \frac{\sigma_v^2}{pT} \sum_{t=1}^{T} \left(\kappa \, tanh^2 \left(b_{1z} + \sum_{j=1}^{p} \sum_{l=1}^{k} w_{zj}^{1(l)} x_{jt-l} \right) + \mu \sum_{\tilde{z}=1}^{z_2} \left(\mathbf{w}_{z,\tilde{z}}^2 \right)^2 \right), \tag{17}$$

if the feedforward neural network contains one hidden layer, and:

$$S_{nz} = \frac{\sigma_v^2}{z_{n-1}} \left(\kappa \, tanh^2 \left(b_{nz} + \mathbf{W}_z^n \mathbf{h}_{n-1} \right) + \mu \sum_{\tilde{z}=1}^{z_{n+1}} (\mathbf{w}_{z,\tilde{z}}^{n+1})^2 \right), \tag{18}$$

for multilayer neural networks.

```
5:     If $S_{nz} \le \chi$ for $n = 1, \dots, N$, remove $h_{nz}$
6:     $\chi = \chi * 0.0001$
7:     Repeat 4 – 6 until $E$ = Maximum Epochs
```

It is possible to illustrate the groupings determined by the first component (in blue) and second component (in

[7] We allow for different numbers of lags across input variables x_{jt}.

black):

$$\left[\begin{array}{ccc|ccc|c|ccc} {}^{i}w_{11}^{1(1)} & \cdots & {}^{i}w_{11}^{1(k)} & {}^{i}w_{12}^{1(1)} & \cdots & {}^{i}w_{12}^{1(k)} & & {}^{i}w_{1p}^{1(1)} & \cdots & {}^{i}w_{1p}^{1(k)} \\ \cdots & & \cdots & \cdots & & \cdots & \cdots & \cdots & & \cdots \\ {}^{i}w_{z_1 1}^{1(1)} & \cdots & {}^{i}w_{z_1 1}^{1(k)} & {}^{i}w_{z_1 2}^{1(1)} & \cdots & {}^{i}w_{z_1 2}^{1(k)} & & {}^{i}w_{z_1 p}^{1(1)} & \cdots & {}^{i}w_{z_1 p}^{1(k)} \end{array}\right]$$

In this setting, for a given output variable x_{it}, we propose the following regularization function:

$$P(\mathbf{W}^1, z_1; \lambda, \alpha) = \lambda(1-\alpha)\sqrt{kz_1}\sum_{j=1}^{p}\left\|\mathbf{W}_j^1\right\|_F + \lambda\alpha\sqrt{z_1}\sum_{j=1}^{p}\sum_{l=1}^{k}\left\|\mathbf{w}_j^{1(l)}\right\|_2, \qquad (21)$$

with z_1 being the optimal number of nodes. Based on the above penalty function, we propose the following objective function:

$$\min_{\mathbf{W}}\left\{\frac{1}{2T}\left\|\mathbf{Y}_i - g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})\right\|_2^2 + P(\mathbf{W}^1, z_1; \lambda, \alpha)\right\}, \qquad (22)$$

where $\mathbf{Y}_i \equiv [\mathbf{x}_{i,1} \ldots \mathbf{x}_{i,T}]$. Expression (22) has a 'sparse double group lasso' penalty because it contains features of both sparse lasso and group lasso penalty functions. The above discussion clearly shows that both penalty terms penalise different groups of variables differently. The second component introduces sparsity in the first component by adding shrinkage in each of the vectors comprising the matrices $\left\|\mathbf{W}_j^1\right\|_F$ before checking if all the parameters in the matrices are zero.

The level of sparsity induced by the group lasso depends on the level of λ; the higher the λ, the lower the number of groups selected. The parameter $\alpha \in [0, 1]$ is a tuning parameter that, like λ, defines the level of sparsity induced into the system. When $\alpha = 0$, the sparse double group lasso reduces to the group lasso. It is also possible to notice the relationship that subsists between the adapted sparse group lasso and the hierarchical group lasso when the number of lags is one and there is no lag selection. In particular, imposing $\alpha = 0$ suggests that no lag selection should be performed and thus that $k = 1$, implying equivalence between the objective function (22) and the hierarchical group lasso proposed by Nicholson et al. (2014) and Tank et al. (2018), as conveyed by the function (28) below.

In our context, it is important to note that the dimension of the groups, when the group lasso is applied in a feedforward neural network, is a quantity that is determined within the model. More specifically, the number of groups in the first hidden layer is established in stage 1 through the mutual information optimization procedure. The quantity z_1 is obtained from this stage. Given the soft thresholding of the group lasso, a higher number of nodes will lead to a lower level of sparsity for a fixed level of λ. In this setting, it is important to choose a suitable number of groups in the first hidden layer. Otherwise, a suboptimal choice $\underline{z}_1$ different from z_1 introduces two effects. First, if $\underline{z}_1 > z_1$, there are more nodes than are required and some of them do not carry information between the input variables and the output variables. In this case, the Granger causality procedures based on neural networks (see Tank et al. (2018)) will lead to spurious Granger causality as a result of the activation of more nodes than necessary, increasing the type I error. A second effect occurs due to introducing additional terms in the regularization component of the objective function $P(\mathbf{W}^1, \underline{z}_1; \lambda, \alpha)$ given by the difference between $\underline{z}_1$ and z_1. This effect can increase the severity of the penalty and lead to deleting weights that are indeed relevant in forecasting the output variable $x_{i,t}$. In this case, the suboptimal choice of the number of nodes in the first hidden layer can lead to an increase in type II errors, spuriously rejecting that a given variable Granger-causes another one, when in fact it does. These effects will be analyzed in a simulation study in Appendix A.

Before introducing the algorithm for the detection of Granger causality, we discuss the role of deeper layers in the correct detection of Granger causality. As the group lasso depends on the size of z_1 and not on the width of deeper architectures, it is expected that the layer-wise widths of z_N - for $N > 1$ - will not have an impact on the correct detection of Granger causality. Intuitively, when fitting a deep neural network, an increase in the number of hidden nodes in the first hidden layer leads to an increase in the assumed interactions among the different input nodes (and thus in the group sizes); conversely, an increase in depth leads to an improvement in the fit of the network by increasing the number of nonlinearities captured by the network without affecting the assumed interactions among input nodes. Therefore, the weights in the first layer take care of the potential relationships (Granger Causal interactions) between the variables in the system, and the weights in higher layers introduce further flexibility into the model and improve the goodness of fit.

3.2.2. Algorithm for the detection of Granger causality

In what follows, we present the algorithm that implements the above methodology for the detection of Granger causality when a feedforward neural network is fitted.

By applying Algorithm 2, this paper extends the current literature on Granger causality discovery via neural networks by defining a new objective function that enables not only the discovery of the Granger causal interactions but also of the optimal lag length. The combination of these two aspects should ensure low type I and type II errors when testing for Granger causality.[8]

The following subsection reviews two alternative penalty functions recently proposed in the neural network literature that introduce lasso penalty functions for model selection.

[8] Another important factor that could impact the interpretation of Granger causality from a neural network relies on the correct combination of the activation function and weight initialization. The exploding gradient problem can reduce the efficacy of the group lasso detection of Granger causality by limiting the training of $\mathbf{W}^1$. Similarly, if the rectified linear unit activation function is used, the "*dying ReLu*" problem could lead to spurious node selection. The impact of the correct engineering of the feedforward neural network in terms of the combination of the activation function and weight initialization is beyond the scope of this article.

Algorithm 2 Algorithm for the Detection of Granger Causality.

INPUT: Dependent and Independent variables.
OUTPUT: Predicted dependent variable, and Granger causal relations.

```
procedure BASED ON ALGORITHM 1

    Initialize an over-specified Deep Neural Network.

    Definition of the Optimal $\nu^2$.
    Divide the dataset in training and test set.
    Given the structure of the network, cross-validate the optimal $\nu^2$.

    Definition Optimal Structure of the Network
    $\chi = 0.001$.
    Set E = Maximum Epochs.
    while (Epoch < E) do
        Generate $v \sim \mathcal{N}(0, \sigma_v^2)$
        Calculate node significance applying Equation (18).
        Remove insignificant nodes.
        Update $\chi$.
    Algorithm 1 will return the pruned Neural Network

    Granger Causality x
    Define the number of hidden nodes and layers from previous steps.
    Fit the feedforward neural network with objective function (22).
```

3.3. Interpretable neural networks

Advances in neural networks have enabled us to propose a feedforward neural network for the detection of Granger causality in large systems. The main difference from previous models based on neural networks is the possibility of interpreting the intermediate steps when making the model predictions. To do this in a Granger causality setting, we interpret the connections between the nodes in the first hidden layer and the input variables. The interpretability of the neural network is made formal by adapting lasso-type regularization functions to a neural network setting. Rather than penalizing the parameters of standard regression models, we propose a model that penalizes the weights in the nodes of the first hidden layer of the neural network. The absence of Granger causality is interpreted as a lack of connections between a given input variable and the set of nodes in the first hidden layer.

Interpretable neural networks are briefly discussed in Hastie et al. (2005), in more detail in Scardapane et al. (2017), and adapted to the detection of Granger causality using a hierarchical lasso penalty function in Tank et al. (2018). To provide a suitable background to our above proposed regularization function, we discuss in this section the regulation functions proposed in these pioneering studies adapted to our VAR setting. In this way, we can compare our novel objective function to the related literature.

Scardapane et al. (2017): These authors propose the following penalty function:

$$P(\lambda, \mathbf{W}) = \lambda \sum_{i=1}^{p} \sqrt{kz_1} \sum_{j=1}^{p} \left\| \mathbf{W}_j^1 \right\|_F + \lambda \sum_{i=1}^{p} \sum_{n=2}^{N} \sum_{z=1}^{z_{n-1}} \sqrt{z_n} \left\| \mathbf{W}_z^n \right\|_2 + + \lambda \sum_{n=1}^{N} \sqrt{z_n} \left\| \mathbf{b}_n \right\|_2 + \lambda \left\| \mathbf{W} \right\|_1 . \tag{23}$$

This penalty term ‘equally’ weights each component of the penalty by $\lambda > 0$.[9] The first component:

$$\sum_{j=1}^{p} \left\| \mathbf{W}_j^1 \right\|_F = \sum_{j=1}^{p} \left[\sum_{z=1}^{z_1} \sum_{l=1}^{k} \left(w_{zj}^{1(l)} \right)^2 \right]^{1/2} , \tag{24}$$

is identical to the first component of the penalty function (22), penalizing the coefficients of the input layer ($n = 1$) across lags and nodes for each time series j. In our framework, there are two ‘groups’ of size kz_1 and z_1, respectively.

The second and third components penalize the ‘adaptable features’ of the neural network, $\{\mathbf{W}^n, \mathbf{b}_n\}_{n=1}^{N}$. The second function penalizes the vector of all outgoing connections from each node z_{n-1} in each hidden layer $n \neq 1$ such that:

$$\sum_{n=2}^{N} \sum_{z=1}^{z_{n-1}} \sqrt{z_n} \left\| \mathbf{W}_z^n \right\|_2 = \sum_{n=2}^{N} \sum_{z=1}^{z_{n-1}} \sqrt{z_n} \left(\sum_{\tilde{z}=1}^{z_n} \left(\mathbf{w}_{\tilde{z}z}^n \right)^2 \right)^{1/2} , \tag{25}$$

for each time series i. Intuitively, this corresponds to the column-wise penalization of column vectors of the matrix $\mathbf{W}^n, n \neq 1$. The third term penalizes the biases $\{\mathbf{b}_n\}_{n=1}^{N+1}$, where the $N+1 \equiv O$ (output node) of the neural network across layers n is:

$$\sum_{n=1}^{N} \sqrt{z_n} \left\| \mathbf{b}_n \right\|_2 = \sum_{n=1}^{N} \sqrt{z_n} \left(\sum_{z=1}^{z_n} b_{zn}^2 \right) . \tag{26}$$

Finally, the fourth term penalises the absolute value of the coefficients of the matrix $\mathbf{W} = [\ldots \left[{}^i\mathbf{W}^1 \ldots {}^i\mathbf{W}^N \right] \ldots]$ for all time series i, i.e:

$$\left\| \mathbf{W} \right\|_1 = \sum_{i=1}^{p} \sum_{n=1}^{N} \sum_{z=1}^{z_{n-1}} \sum_{j=1}^{p} \sum_{l=1}^{k} \left| {}^i w_{zj}^{n(l)} \right| . \tag{27}$$

Importantly, it is this last constraint that does not allow Scardapane et al.’s (2017) optimization problem to ‘decouple’ across the rows of the output variable $\mathbf{x}_t$

[9] Scardapane et al. (2017) argue that after experimenting with simulation results, weighting the components differently does not make a difference.

and to be solved 'in parallel'; specifically, we propose a different neural network for each of the p output variables in the system. For given network architectures, Farrell et al. (2018) obtain conditions for valid inference over $(\mathbf{W}, \mathbf{b})$ in non-regularized feedforward neural networks.

Hierarchical group lasso in neural networks: Tank et al. (2018) also base their definition of Granger causality on the invariance of the neural network to x_{jt}. In particular, these authors adapt a hierarchical group lasso objective function that had previously been proposed for high-dimensional linear VAR models by Nicholson et al. (2014). Tank et al. (2018) propose a hierarchical group lasso function that allows for Granger causality detection and also for automatic lag selection. The objective function is:

$$\min_{\mathbf{W}} \frac{1}{2T} \|\mathbf{Y}_i - g_i(\mathbf{X}_{t-1}; \mathbf{W})\|_2^2 + \lambda \sum_{j=1}^{p} \sum_{l=1}^{k} \|\mathbf{w}_j^{1(l:k)}\|_F, \tag{28}$$

where $\mathbf{w}_j^{1(l:k)} = [w_{ij}^{1(l)} \ldots w_{ij}^{1(k)}]$. This is a hierarchical group penalty in the sense that if $\mathbf{w}_{ij}^{1(l:k)} = \mathbf{0}$ then for all $l' > l, \mathbf{w}_{ij}^{1(l':k)} = \mathbf{0}$. We use $g_i(\mathbf{X}_{t-1}; \mathbf{W})$ in (28) to differentiate from our multilayer perceptron function $g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})$ that chooses the number of nodes z_n strategically. In the hierarchical group lasso setting, the quantity $\mathbf{z}$ is a vector of nuisance parameters that is taken as given in the optimization problem.

Therefore, the methodology proposed by Tank et al. (2018) differs from the methodology introduced in our paper not only in terms of the regularization considered but also in terms of the identification of the parameters affecting Granger causality via a feedforward neural network. In particular, as discussed previously, the underestimation or overestimation of the number of hidden nodes in the first hidden layer due to the exogenous dimension of $\mathbf{h}_1$ can lead to an increase in either aggregate type I or type II errors. Also, expression (22) allows a lag selection strategy to be performed in a similar way to Tank et al. (2018) but with a lower level of penalty given by not using the hierarchical structure. For each group, the optimal lag will be identified by the highest non-zero lag length l'. Lag lengths higher than l' will have the L_2 norms equal to zero and can be considered jointly non-significant.

4. Parameter identification and model selection

The aim of this section is to assess the correct identification of the parameters characterizing the objective function (22) under the null hypothesis of no Granger causality. To do this, we explore the conditions obtained from our objective function that leads us to delete irrelevant weights (nodes and input variables). We consider these conditions and, in particular, the role of λ and α in introducing sparsity into the regularization problem.

There are kz_1p parameters in the objective function (22) indexed by $w_{zj}^{1(l)}$, with $z = 1, \ldots, z_1$, $j = 1, \ldots, p$, and $l = 1, \ldots, k$. These parameters constitute a group, denoted by the matrix $\mathbf{W}_j^1$, of size kz_1. The objective function is convex, implying that the solution $\widehat{\mathbf{W}}_j^1$ to the minimization problem is characterized by the first-order conditions of the problem. These conditions are given by:

$$\frac{1}{T} \frac{\partial g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})}{\partial \mathbf{W}_j^1} (\mathbf{Y}_i - g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z}))$$
$$= \lambda(1-\alpha)\sqrt{z_1 k}\mathbf{u}_1 + \lambda\alpha\sqrt{z_1}\mathbf{u}_2, \tag{29}$$

where $\frac{\partial g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})}{\partial \mathbf{W}_j^1}$ is the first derivative of the function $g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})$ with respect to the parameters in matrix $\mathbf{W}_j^1$; $\mathbf{u}_1 = \frac{\widehat{\mathbf{W}}_j^1}{\left\|\widehat{\mathbf{W}}_j^1\right\|_F}$ if $\widehat{\mathbf{W}}_j^1 \neq 0$, and $\mathbf{u}_1$ is a matrix inside a unit ball such that $\|\mathbf{u}_1\|_F \leq 1$ if $\widehat{\mathbf{W}}_j^1 = 0$. The definition of $\mathbf{u}_2$ is similar, but replaces the matrix $\mathbf{W}_j^1$ with the vector $\mathbf{w}_j^{1(l)}$ and the Frobenius norm with the L_2 norm.

The null hypothesis H_0 of no Granger causality of x_{jt} to x_{it} corresponds to $\mathbf{W}_j^1 = 0$. The corresponding estimate from the objective function (22) must be zero in order for the lasso penalty to delete the parameter. The first-order conditions with $\widehat{\mathbf{W}}_j^1 = 0$ must satisfy the condition:

$$\frac{1}{T} \left\| \frac{\partial g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})}{\partial \mathbf{W}_j^1} (\mathbf{Y}_i - g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})) \right\|_F$$
$$\leq \lambda(1-\alpha)\sqrt{z_1 k} + \lambda\alpha\sqrt{z_1}. \tag{30}$$

This inequality shows the contribution of λ and α to the condition that keeps a group inactive; that is, the condition that allows us to assume $\mathbf{W}_j^1 = 0$, and, hence, not rejecting the null hypothesis that the variable x_{jt} does not Granger cause x_{it}.

For problems in which the number of lags k is fixed, it is sufficient to impose $\lambda = o(1/T)$ in order for this condition to be satisfied for increasing sample sizes (see Fan and Li (2001)). In this scenario, our model selection strategy is consistent; that is, it deletes those weights that do not influence the output variables. For high-dimensional problems in which the number of lags also grows to infinity with the sample size, $k = k_T$, we must impose a tighter convergence of the tuning parameter λ. In our problem, it is sufficient to have $\lambda = o\left(\frac{1}{\sqrt{k_T T}}\right)$, with $k_T/T \to 0$. Alternatively, we can assume that α also converges to zero such that for high-dimensional problems we have $\lambda = o(1/T)$ and $1 - \alpha = o(1/\sqrt{T})$. These two conditions are sufficient to guarantee the correct selection of the parameters as $T \to \infty$. In practice, for a given sample size, these parameters are optimized by cross-validation.

The first-order conditions of the objective function (22) can also provide some insight into the sparsity of the vector $\mathbf{w}_j^{1(l)}$ within the matrix $\mathbf{W}_j^1$ when some of the elements of the matrix are nonzero. In this case, the group corresponding to the input variable x_{jt} is not rejected and the question of interest is to select those lags that influence the forecasting of x_{it} and, by doing so, the optimal number of lags that should be included in the model. Sparsity in this case is provided by subsets of parameters in the matrix $\mathbf{W}_j^1$ that are actually zero. The aim of our objective function is to be able to delete these parameters.

More formally, if $\mathbf{W}_j^1 \neq 0$, the corresponding first-order conditions of the objective function (22), if $x_{j,t-l}$ does not influence x_{it}, must satisfy, for $\widehat{\mathbf{w}}_j^{1(l)} = 0$, the following condition:

$$\left\| \frac{\partial g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})}{\partial \mathbf{w}_j^{1(l)}} (\mathbf{Y}_i - g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})) \right\|_2 \leq T\lambda\alpha\sqrt{z_1}. \quad (31)$$

In this case, it is sufficient to assume that $\lambda = o(1/T)$ and α are constant for the model to delete those parameters that are zero within a larger group and to achieve model selection consistency.

There is another source of sparsity within groups. Thus, we can consider parameters that are zero within the group of parameters that determine the relevance of a lag. More specifically, we can have $\mathbf{w}_j^{1(l)} \neq 0$, but some parameters of this vector are equal to zero. To address this case, one possibility is to extend the objective function (22) to include a further penalty function $\|\mathbf{W}^1\|_1$ inducing sparsity at the individual level. Although this additional penalty would allow us to correctly detect those weights that are zero if the vector $\mathbf{w}_j^{1(l)}$ is at least partially nonzero, this would increase the computational complexity of the method. More importantly, the marginal benefit of including those terms would be negligible from the point of view of model selection since the parameters that would be rightly identified as zero would correspond to connections between input nodes and specific nodes in the first hidden layer. The interpretation of these connections is not important once we accept that for some nodes in the first hidden layer the weights are different from zero, $\mathbf{w}_j^{1(l)} \neq 0$, and, therefore, carry information relevant for Granger causality and lag selection. For this reason, we shall not further pursue the identification of these parameters.

Finally, we should mention that in contrast to lasso penalty functions expanding least squares procedures and well-behaved likelihood functions (see Yuan and Lin (2006), Zhou and Zhu (2010), Simon et al. (2013), and Nicholson et al. (2014, among other leading examples), the objective function (22) is highly nonlinear due to the presence of nonlinear activation functions θ in each hidden layer that characterize the function $g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})$. This implies that it is not possible to derive, in closed form, the estimates of the weights different from zero that characterize the objective function (22).

5. Empirical analysis: Tobalaba network

The Energy Web Foundation provides the energy sector with a blockchain-based test network with a Proof-of-Authority[10] consensus mechanism: the Tobalaba test network (Energy Web Foundation, 2018). The World Bank Group (2018) highlights the benefit of a distributed ledger technology over the traditional centralized ledgers: it enables decentralization and disintermediation, it guarantees information symmetry due to the verifiable audit of transactions of both physical and digital assets, and it ensures the cost reduction and the associated increase in speed of the stipulation of contracts via smart contracts. J.P. Morgan (2018) argues that the automation and the disintermediation arising from the application of blockchain technologies will automate the functions necessary to participate in the market, also expanding the access to smaller participants. The Energy Web Foundation (2018) states that smart contracts, by automating bilateral transactions, will allow for a greater diversity of market structure. The resulting information symmetry will allow tracing in the carbon and renewable energy market and credit ownership with lower costs and higher accuracy (Energy Web Foundation, 2018).

Using these arguments, one should expect an increase in the stipulation of contracts, and thus an increase in the interactions among the members of the Tobalaba network. The increase in the connections (unilateral or bilateral) is justified by a higher transparency of credit and asset ownership, by the automation of the execution of smart contracts (not feasible for centralized ledgers), and by the reduction of transaction costs due to disintermediation. The aim of this section is to explore this empirically. To do this, we use recent work on social and financial networks (see Billio et al. (2012) and Hecq et al. (2019)) that establish connections in a network through the presence of Granger causality between the variables characterizing the nodes. These authors explore Granger causality between pairs of variables. In this application, we broaden the analysis of Granger causality to define a network, and consider equation (4), reproduced here again:

$$x_{it} = g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z}) + \epsilon_{it}, \quad \text{for } i = 1, \ldots, p,$$

where $g_i(\mathbf{X}_{t-1}; \mathbf{W}, \mathbf{z})$ captures the multivariate dynamic structure between the percentage cumulative log-returns over a 30-minute time window for the firms below. We model the multivariate dependence component-wise using a neural network for each company $i = 1, \ldots, p$.

5.1. Data

Intra-day prices in 30-minute intervals for the companies reported in Table 1 over the period 09/05/2016 to 10/05/2019 are collected from Bloomberg. Of the 70 companies belonging to the Tobalaba Network, only those reported in Table 1 are considered.[11] We exclude companies listed in different time zones and nonlisted companies. Our dataset is divided in two periods: before the introduction of Tobalaba (09/05/2016–29/03/2018) and after the creation of Tobalaba (26/10/2018 - 10/05/2019). A time interval between the two subsets is left in order to allow the creation of the connections between the members of the Energy Web blockchain.

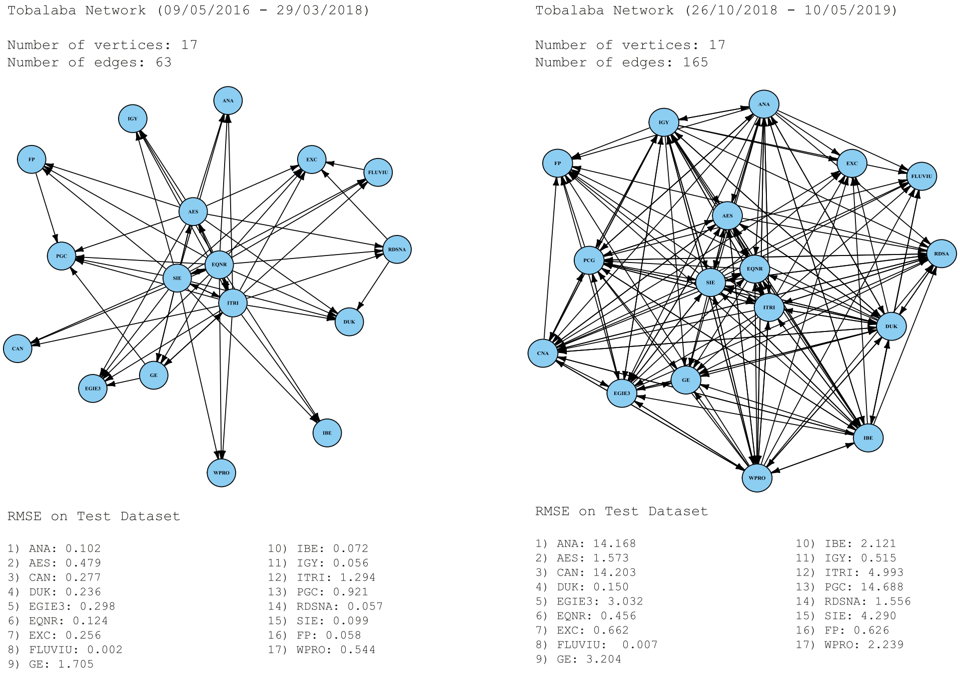
The missing values in the dataset are completed using the MissForest algorithm (Stekhoven, 2013). The maximum number of trees to be grown in each forest is set equal to 500, the maximum number of nodes for each tree is equal to 100, and the maximum number of iterations is 50. The MissForest algorithm does not make any assumption about the distribution of the variables

[10] Transactions are validated through validators, reducing the energy impact.

[11] The rest of the companies are excluded for two main reasons: because they are either not public or have a high number of missing random observations.

Table 1
Names, market, and a short description of the core activities of the companies considered in this study.

Company name	Market	Core	Tick
Acciona	Spain	Renewable energy	ANA
Aes Corp	USA	Electricity sell, mines coal, alternative source of energy	AES
Centrica	London	Home and business energy solution	CAN
Duke Energy	USA	Manage portfolio of natural gas supply and delivery	DUK
Engie Brasil Energia	Brasil	Exploration, production and trading of electricty and natural gas	EGIE3
Equinor Asa	Oslo	Develops oil, gas, wind and solar energy projects	EQNR
Exelon Corp	USA	Distributes energy to Illinois and Pennsylvania	EXC
Fluvius	Luxemburg	Renewable energy distribution Network	FLUVIU
General Electric	USA	Diversified technology	GE
Iberdrola	Spain	Generates, distributes, trade electricity	IBE
Innogy	Germay	Manages plans to generate power from renewable energy	IGY
Itron	USA	Collecting, communicating analysing electric data	ITRI
PG&E Corp	USA	Holding company that provides natural gas and electricyt	PCG
Royal Dutch Shell	London	Explores, produces, refines petroleum	RDSA
Siemens	Germany	Engineering and manufacturing company	SIE
Total Sa	Euronext Paris	Explores for producers, refines, transports, and market oil and natural gas	FP
Wipro	India	E-commerce, data warehousing, system administration	WPRO



```
Tobalaba Network (09/05/2016 - 29/03/2018)

Number of vertices: 17
Number of edges: 63

RMSE on Test Dataset

1) ANA: 0.102                 10) IBE: 0.072
2) AES: 0.479                 11) IGY: 0.056
3) CAN: 0.277                 12) ITRI: 1.294
4) DUK: 0.236                 13) PGC: 0.921
5) EGIE3: 0.298               14) RDSNA: 0.057
6) EQNR: 0.124                15) SIE: 0.099
7) EXC: 0.256                 16) FP: 0.058
8) FLUVIU: 0.002              17) WPRO: 0.544
9) GE: 1.705
```

```
Tobalaba Network (26/10/2018 - 10/05/2019)

Number of vertices: 17
Number of edges: 165

RMSE on Test Dataset

1) ANA: 14.168                10) IBE: 2.121
2) AES: 1.573                 11) IGY: 0.515
3) CAN: 14.203                12) ITRI: 4.993
4) DUK: 0.150                 13) PGC: 14.688
5) EGIE3: 3.032               14) RDSNA: 1.556
6) EQNR: 0.456                15) SIE: 4.290
7) EXC: 0.662                 16) FP: 0.626
8) FLUVIU:  0.007             17) WPRO: 2.239
9) GE: 3.204
```

Fig. 3. Granger causal networks before and after the introduction of Tobalaba. The out-of-sample RMSE is reported for each company.

as it involves estimating the missing values by fitting a random forest trained on the observed values. The Out-Of-Bag (OOB) estimates of the imputation error in terms of the Normalized Root Mean Squared Error (NRMSE) for the two subsamples are 0.01438 and 0.012984, respectively. The returns are then computed from the intra-day prices.

Table 2 reports the exploratory data analysis conducted for both subsamples for each individual series considered. There is a general increase in the mean and standard deviation of the returns for each company. In all cases, the Dickey-Fuller test rejects the null hypothesis of the unit root at a 0.05 significance level, and we fail to reject the null hypothesis of the stationarity of the KPSS test at a 0.05 significance level, showing that, for both subsamples, all series considered are stationary.

5.2. *Empirical results*

Fig. 3 shows the network topologies induced by fitting model (4) to detect Granger causality between the $p = 17$ firms considered in our sample before and after the introduction of Tobalaba. The method to detect Granger causality is based on using an optimized neural network and the objective function (22). Thus, the network is constructed by fitting 17 feedforward neural networks with a double group lasso penalty function to the weights that connect input nodes to the nodes in the first hidden layer. Each component-wise feedforward neural network has company x_{it} in the output layer and the lagged values of companies x_{jt} with $j \neq i$ in the input layer. The edges for each vertex are identified by the Granger causal

Table 2
Exploratory data analysis of the series considered. Due to the number of observations, the Kolmogorov-Sminrov test for normality is adopted in the first sub-sample. Moreover, the test statistics and associated *p*-values of the Dickey-Fuller and the KPSS tests for stationarity are reported.

(09/05/2016–29/03/2018)																	
	ANA	AES	CAN	DUK	EGIE3	EQNR	EXC	FLUVIU	GE	IBE	IGY	ITRI	PGC	RDSNA	SIE	FP	WPRO
Mean	−0.0015	−0.0005	−0.0045	−0.0005	0.0000	0.0032	0.0013	0.0003	−0.0092	−0.0003	−0.0007	0.0054	0.0063	0.0017	0.0013	0.0009	0.0007
Std. Deviation	0.3127	0.7086	0.4612	0.5685	0.7603	0.4184	0.6126	0.0667	1.0939	0.2951	0.5256	1.6895	1.5639	0.3069	0.3286	0.3069	1.1897
Min	−8.3657	−8.5695	−17.1965	−6.6416	−9.6321	−3.6688	−5.2387	−0.6217	−13.8826	−12.8402	−7.6234	−34.4226	−32.8006	−8.5474	−8.7049	−9.5013	−11.4161
Max	3.7182	10.7099	14.3815	6.5628	10.5759	4.2803	5.4795	0.6585	21.9190	3.3580	7.5691	34.7628	33.9859	5.2412	6.7774	4.9290	10.5179
Skweness	−1.7878	−0.1906	−5.8680	−0.2275	0.0911	0.0493	−0.3339	0.3446	0.6017	−9.3001	−0.0698	0.8649	0.7876	−1.5988	−1.8681	−2.7896	−0.3433
Kurtosis	72.9088	35.5819	423.4997	28.9009	36.5251	15.1058	17.3413	36.5411	66.8865	419.7467	53.2526	101.3114	122.7723	93.5393	129.5589	119.0220	19.4230
Kolm. t. stat.	0.2872	0.2576	0.2696	0.2835	0.2342	0.2718	0.2648	0.4399	0.2652	0.3070	0.2935	0.2257	0.2221	0.3017	0.3033	0.2940	0.2167
Kolm. p-value	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
DF t. stat.	−47.2078	−50.6489	−46.6261	−50.4284	−50.6923	−57.6236	−52.4630	−54.1766	−48.6654	−50.2723	−51.1224	−47.7949	−48.3805	−48.1052	−49.8299	−49.9683	−50.6796
DF p-value	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100
KPSS stat	0.1414	0.0155	0.1606	0.0630	0.0626	0.0605	0.0343	0.0539	0.3718	0.1060	0.0176	0.0387	0.0589	0.0456	0.3180	0.0333	0.0472
KPSS p-value	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.0893	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
(26/10/2018–10/05/2019)																	
	ANA	AES	CNA	DUK	EGIE3	EQNR	EXC	FLUVIU	GE	IBE	IGY	ITRI	PCG	RDSA	SIE	FP	WPRO
Mean	0.0079	0.0016	−0.0145	−0.0006	0.0060	−0.0022	0.0022	0.0003	0.0104	0.0053	−0.0007	0.0055	−0.0111	0.0027	0.0028	−0.0013	−0.0031
Std. Deviation	2.1697	1.5338	2.4323	0.8389	2.8989	1.3800	0.8406	0.0656	3.8024	1.3659	0.5781	2.2506	11.7503	1.2577	1.3416	1.1393	2.0681
Min	−27.0106	−17.2372	−24.6358	−6.1559	−24.1935	−12.3153	−8.8194	−0.4471	−41.8479	−11.1166	−5.4102	−21.9921	−116.7995	−9.2351	−11.5270	−9.5761	−23.3458
Max	26.1996	18.7566	24.8999	6.9390	22.7915	12.3872	8.6189	0.3615	40.9288	11.0580	5.2053	22.4000	85.3738	9.6994	11.5836	9.9737	25.3803
Skweness	−0.2296	0.7305	−0.0618	−0.3039	−0.1741	0.2325	0.5157	−0.2896	0.0196	0.0515	−0.2343	−0.2159	−1.1079	0.1642	0.0447	0.2635	−1.1434
Kurtosis	47.4203	52.9197	34.5855	21.1393	30.9132	37.4122	31.7782	15.4785	33.8791	29.2221	27.9577	26.2839	24.8550	16.2953	25.5945	22.2098	49.7950
Shapiro t. stat.	0.3710	0.4670	0.3945	0.5968	0.4429	0.4528	0.5746	0.6277	0.5099	0.4557	0.5111	0.5600	0.4822	0.5319	0.4741	0.5222	0.4525
Shapiro p-value	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
DF t. stat.	−24.4804	−24.8217	−22.9151	−24.9950	−23.7368	−22.8723	−24.7452	−18.4441	−24.4323	−23.9530	−23.4678	−23.4181	−24.3258	−23.0250	−22.8387	−22.7410	−23.9251
DF p-value	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100
KPSS t. stat.	0.0192	0.0497	0.0646	0.0171	0.0145	0.0176	0.0192	0.0640	0.0141	0.0208	0.0191	0.0356	0.0162	0.0101	0.0316	0.0275	0.0343
KPSS p-value	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000

interactions, defined by the sparsity induced in the objective function (22).

To train the component-wise feedforward neural networks, we apply the Adam optimizer with a constant learning rate. Different learning rates (0.0001, 0.001, 0.01, and 0.1) are tuned and the optimal learning rate for both before and after the introduction of Tobalaba is 0.1. The initial number of hidden nodes is set equal to $\underline{z}_1 = 30$ and $\underline{z}_2 = 35$. Following Montgomery and Eledath (1995), the tuning parameters of the algorithm are $\kappa = 1$, $\mu = 0.2$, $\chi = 0.000001$, and the number of epochs is 7000. The same parameters are adopted for both subsamples. The optimal combination of α and λ is obtained by cross-validation. The domain of the two hyper-parameters was discussed earlier. As in the simulation exercise (see Appendix A), we consider 75% of the dataset to train the network and 25% of the sample to obtain the out-of-sample RMSE associated with each combination.

Fig. 3 shows an increase in the number of edges in the network after the introduction of Tobalaba (from 63 to 165) and, in particular, there is an increase in bi-directional edges. This finding clearly reveals the increase in connections after the introduction of the platform and can be justified by the introduction of the distributed ledgers and smart contracts that allow significantly more contracts to be stipulated due to the reduction of transaction costs, the increase in information, and the absence of intermediaries.

5.2.1. Centrality measures

In this section, we study different centrality measures to interpret the results with respect to the importance of the firms in the Tobalaba platform. Table 3 reports different measures of the degree centrality for each network, the betweenness centrality, the eigen centrality, the page rank, and the in-degree and out-degree centrality reported below. The different measures reported in Table 3 are used to identify the central nodes in the two uncovered networks; the different centrality measures allow the absence of a general definition of centrality to be overcome (Rodrigues, 2019).

Looking at degree centrality (defined as the number of connections relative to each node), we observe an increase in the number of links for each vertex after the introduction of Tobalaba. Before the introduction of Tobalaba, the companies *AES*, *EQNR*, *ITRI*, and *SIE* were the central nodes, whereas after the introduction of Tobalaba the number of central nodes increases drastically to 10.[12] However, as pointed out by Rodrigues (2019), degree centrality should be considered as a local centrality measure that does not take into account the density of the links among different nodes. In Table 3 we also report the in-degree and out-degree centrality statistics that are relevant in directed networks. The in-degree centrality defines how prominent a node is and the out-degree centrality measures the centrality of a node in the network. The reported out-degree centrality measures confirm the conclusions drawn from the other centrality measures analyzed: the introduction of the new blockchain platform increases the number of central nodes from 4 to 12. More interestingly, the directed measures of degree centrality provide useful insights regarding the interactions among the members of the network. After the introduction of Tobalaba, all nodes become more receptive due to an increase in the in-degree centrality for all the analyzed companies. However, the out-degree centrality for *EXC*, *FLUVIU*, *GE*, and *FP* is still zero, which is in contrast to all the other members of the network that increase the out-degree statistic after the introduction of Tobalaba. Looking at the core activities of the members of the network, we note that *EXC* and *FLUVIU* are retail distributors of energy and, as such, are expected to receive a high number of incoming transactions from companies that are either producers of energy or of the infrastructures used for distribution.

The betweenness centrality (unweighted) quantifies the importance of a node when connecting to other vertices (Bloch et al., 2017). Table 3 also shows that the degree of centrality, with the exception of EXC, *FLUVIU*, *GE*, and *FP*, changes after the introduction of Tobalaba. Before the introduction of Tobalaba, the betweenness centrality identifies *ITRI* as the primary central node, implying that a removal of *ITRI* from the network would have implied a disruption of the overall network activity. Conversely, after the introduction of Tobalaba, the primary central node is *PCG*. It is also interesting to see, after the introduction of Tobalaba, how the number of nodes that influence the flow of information circulating through the network increases. The betweenness centrality for the majority of the vertices in the network is zero before the introduction of Tobalaba and increases, in most cases, after the introduction of the blockchain platform.

Eigenvector centrality (Bonancich, 1987) takes into account not only the connections of the particular node but also how many links the connected neighbours have. In other words, it measures the "*prestige*" (Bloch et al., 2017) of a node. Before the introduction of Tobalaba, the eigenvector centrality confirms the conclusions drawn from the betweenness centrality. After the introduction of Tobalaba, *DUK* is identified as the primary central node. Also, in this case, it is possible to note how the degree of centrality increases for all the members of the network, with the companies *ANA*, *AES*, *CNA*, *IBE*, *IGY*, *ITRI*, *PCG*, *SIE*, and *WPRO* characterized by an eigenvalue of centrality close to unity.

Finally, the page rank is also analyzed. The page rank is a variant of the eigenvector centrality that also takes into account the directions of the different links. Before the introduction of Tobalaba, the page rank identifies *PCG* and *EXC* as central nodes; after the introduction of Tobalaba, the degree of centrality of the different nodes becomes more uniform, reducing the spread in page rank across the different members.

To summarize, the different centrality measures confirm an increase in the degree of centrality associated with each vertex of the Granger causal network after the introduction of Tobalaba. The increase in the number of central nodes can be associated with an increase in the number of bilateral transactions due to the newly

[12] Degree centrality higher than 20.

Table 3
Centrality Measures: Degree centrality, betweenness centrality, eigen centrality, and page rank before and after the introduction of the Tobalaba network.

	ANA	AES	CNA	DUK	EGIE3	EQNR	EXC	FLUVIU	GE	IBE	IGY	ITRI	PCG	RDSA	SIE	FP	WPRO
(09/05/2016–29/03/2018)																	
Degree	4	13	3	5	5	18	6	4	7	3	4	19	6	5	17	4	3
In-Degree	4	3	3	5	5	2	6	3	4	3	4	4	6	3	2	3	3
Out-Degree	0	10	0	0	0	16	0	1	3	0	0	15	0	2	15	1	0
Betweennes	0.0000	1.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	19.3333	0.0000	0.0000	0.6667	0.0000	0.0000
Eigen	0.3533	0.7329	0.2802	0.3878	0.4069	0.9191	0.4199	0.3221	0.5372	0.2802	0.3533	1.0000	0.4366	0.3451	0.8904	0.2971	0.2802
Page Rank	0.0515	0.0475	0.0475	0.0724	0.0661	0.0451	0.1127	0.0475	0.0515	0.0475	0.0515	0.0626	0.1069	0.0490	0.0450	0.0480	0.0475
(26/10/2018–10/05/2019)																	
Degree	23	20	22	27	20	12	8	9	11	25	25	25	24	18	25	11	25
In-Degree	7	10	10	11	10	10	8	9	11	9	9	9	11	11	10	11	9
Out-Degree	16	10	12	16	10	2	0	0	0	16	16	16	13	7	15	0	16
Betweennes	0.7100	1.1445	3.0056	5.2286	1.0215	0.3270	0.0000	0.0000	0.0000	2.1262	2.4052	3.5334	17.3520	0.9917	4.5448	0.0000	2.6096
Eigen	0.8644	0.8216	0.8305	1.0000	0.7846	0.4841	0.3453	0.3879	0.4423	0.9353	0.9337	0.9306	0.9055	0.7214	0.9254	0.4586	0.9301
Page Rank	0.0448	0.0561	0.0593	0.0648	0.0561	0.0554	0.0459	0.0513	0.0829	0.0533	0.0533	0.0563	0.0830	0.0609	0.0606	0.0635	0.0526

adopted blockchain technology, thereby reducing the overall network reliance on a single central vertex, increasing its activity, robustness, and reliability.[13]

5.2.2. Structure of neural network

Table 4 reports the optimal α, λ, and structure of the component-wise feedforward neural networks fitted to construct the networks reported in Fig. 3. These results highlight the sensitivity of the structure of the neural network to the amount of information transmitted through it, and, hence, the importance of constructing an optimal neural network prior to uncovering the presence of the predictive ability between the variables. Before the introduction of Tobalaba, the optimal number of nodes in the first and second hidden layers is lower than the number of hidden nodes after the introduction of Tobalaba. After the introduction of Tobalaba, the larger number of hidden nodes captures the higher number of interactions that arise between firms due to the increase in the number of bilateral (decentralized) transactions, which increases the interdependencies between operating firms, naturally leading to a 'more dense' network architecture to capture them. The optimal construction of the neural network obtained from applying the algorithm from Montgomery and Eledath (1995) guarantees that the network does not propagate noise through the neural network, instead only considering meaningful information for the analysis of Granger causality and the predictive ability of the variables.

To add robustness to the results of this exercise, we also consider a reduced dataset. In particular, both subsamples are reduced to the first 25% of the observations. In these cases, the number of edges observed before the introduction of Tobalaba is 63, and the number after the introduction of Tobalaba is 143. These results corroborate previous findings: there is no change in the number of connections and interactions between the firms before the introduction of Tobalaba; conversely, after the introduction of Tobalaba, when the first 25% of the dataset is used, we observe a reduction in the number of edges compared to Fig. 3, showing that the number of interactions between the firms has increased over time.

5.2.3. Forecast accuracy

The original definition of lagged causality (Granger, 1969), $x_{j,t-l} \Rightarrow x_{it}$, involves an increase in the forecast accuracy of time series x_{it} given the lagged values of the time series x_{jt}. The edges of the Granger causal network reported above are identified by the Granger causal interactions discovered by the objective function (22). Once the parameters are estimated and the weights penalized, it is possible to forecast out of sample. Consequentially, the Granger causal network that we have uncovered in this empirical exercise can be justified as a framework for improving the forecasts of a multivariate time series of the log returns of 17 firms.

We formalize this claim by comparing the component-wise forecast accuracy of the feedforward neural network against several benchmark models. To obtain the one-step-ahead forecasts, a rolling window approach is implemented. We compare the predictive ability of the constructed Tobalaba network against the different benchmark models by applying a one-sided Diebold–Mariano test (1995). The hypothesis of the predictive ability can be written in terms of the mean square forecast error (MSFE) between both predictive models. For each i, we have:

$$\mathcal{H}_0 : MSFE^i_{nn} \geq MSFE^i_{VAR}, \tag{32}$$

and the alternative is:

$$\mathcal{H}_A : MSFE^i_{nn} < MSFE^i_{VAR}, \tag{33}$$

with $MSFE^i_{nn}$ denoting the mean square forecast error for the prediction obtained from the neural networks, and $MSFE^i_{VAR}$ the corresponding quantity obtained from the alternative models. Table 5 reports the test statistics and the p-value of the one-sided Diebold–Mariano test (1995) for different benchmark models.

In the absence of a relevant model for an unknown data generating process, the linear VAR(K) is chosen as the first benchmark as it can be considered the best linear approximation of a process that may be nonlinear. Moreover, Plagborg-Møller and Wolf (2020) show how, for a large number of lags, a VAR(K) is as robust to nonlinearities as the linear projection. The maximum lag length allowed in the VAR(K) is 10; the optimal lag is selected using the AIC scores. The top panel in Table 5 corresponds to the period before the introduction of the Tobalaba network and the bottom panel corresponds to the period afterwards. For the first period, we fail to reject the null hypothesis of equal forecast ability in six cases, at a 0.05 significance level. However, after the introduction of Tobalaba, the null hypothesis is rejected in all cases. The higher forecast accuracy of the feedforward neural network for each vertex is a result of including nonlinear interactions between the variables and also a potentially larger persistence compared to the VAR(10) model.

As a robustness exercise, we also propose three different benchmark models that compete against our neural network specification. First, we select the optimal lag of our VAR(K) model using the BIC score instead of the AIC score.[14] In addition to the model obtained from the BIC score, we also consider two alternative benchmarks given by a linear VAR(K) model estimated using a component-wise hierarchical group lasso, as in Nicholson et al. (2014). This benchmark defines the best linear VAR alternative that can be considered in such high-dimensional multivariate time series. This model is, therefore, a suitable alternative strategy in large dimensions for our feedforward neural network. Finally, we also consider an ARIMA(p, d,

[13] It is worth noting that the increase in the number of central nodes, and thus in the number of critical companies and connections, may have a significant impact on the study of cascading failures based on the dependency risk methodology of Kotzanikolaou et al. (2013). The study of cascade failure and risk transmission will be the subject of future research.

[14] Lütkepohl (1985) shows in his simulation study with VAR models that the BIC outperforms other model selection criteria by choosing the correct autoregressive order and by returning the smallest mean squared forecast error from one-step-ahead forecasts

Table 4
Optimal λ and α returned from cross-validation for each of the fitted Feedforward Neural Networks. The structure of the Network selected by the Algorithm of Montgomery and Eledath (1995) is also reported.

	ANA	AES	CNA	DUK	EGIE3	EQNR	EXC	FLUVIU	GE	IBE	IGY	ITRI	PCG	RDSA	SIE	FP	WPRO
(09/05/2016–29/03/2018)																	
z_1	3	6	2	1	1	11	3	2	7	2	3	20	3	3	1	5	11
z_2	2	3	1	1	1	4	1	2	1	1	2	4	2	1	1	1	3
α	0.1	0.3	0.2	0.2	0.3	0.4	0.1	0.3	0.3	0.1	0.1	0.2	0.3	0.3	0.4	0.3	0.3
λ	0.3	0.2	0.7	1	0.8	0.3	0.4	0.5	0.3	0.9	0.5	0.4	0.9	0.5	0.2	0.7	0.6
(26/10/2018–10/05/2019)																	
z_1	28	28	17	24	24	19	15	2	29	19	2	27	30	20	17	21	18
z_2	7	8	3	7	6	4	4	2	11	5	1	7	11	5	4	5	4
α	0.2	0.4	0.2	0.3	0.2	0.2	0.2	0.2	0.2	0.1	0.3	0.1	0.2	0.1	0.1	0.3	0.3
λ	0.4	0.9	0.4	0.3	0.5	0.6	0.9	0.5	1	0.1	0.1	0.1	0.1	0.4	0.2	1	0.3

q) process; this model does not accommodate any feedback effect from other input variables and, hence, fails to incorporate Granger causal interactions. The predictive ability of these models is compared, as before, using the Diebold and Mariano (1995) tests in Table 5. The choice of these benchmarks allows us to understand different aspects regarding the performance of the proposed methodology in uncovering Granger causal relations.

The results show that the forecasts of the VAR(K) model obtained using the BIC score have a similar predictive ability to the VAR(K) model using the AIC score. Both models fare poorly with respect to the feedforward neural network in terms of predictive ability. The second benchmark is given by a high-dimensional VAR - optimized over one-step ahead forecasts - with the componentwise hierarchical group lasso proposed by Nicholson et al. (2014). Despite both procedures - Nicholson et al. (2014)'s VAR benchmark and our methodology - being able to accommodate high-dimensional systems, the results in Table 5 report statistically significant differences in one-step-ahead forecasts between the two models. These results provide evidence that the neural network is able to outperform state-of-the-art high-dimensional linear VARs with induced sparsity via convex regularizers. The main reason for this is the ability of feedforward neural network models to capture the nonlinearities in the underlying data generating process. Finally, an ARIMA(p, d, q) is used to further validate the Granger causal network discovered with our novel methodology against a time series linear model that exhibits no Granger causality. The results reported in the bottom panels of Table 5 provide further empirical support of the feedforward neural network model.

6. Conclusions

This paper has proposed a new methodology for the detection of Granger causality in a vector autoregressive setting using feedforward neural networks. To do this, we have constructed an optimal neural network that maximizes the mutual information transfer between input and output nodes. In a second stage, we propose a novel objective function that introduces sparsity in high-dimensional systems and controls for the number of connections between the input variables and the nodes in the first hidden layer. The newly proposed objective function detects the Granger causal interactions between the variables and also the optimal lag length associated with each input variable, allowing different lag orders for each endogenous time series.

The simulation study shows, in finite samples, the importance of using an optimal network structure to reduce type I and type II errors. In particular, we show that the number of nodes in the first hidden layer has a significant impact on the correct detection of Granger causal interactions. Our simulations also show the consistency of the algorithm used to detect the optimal number of nodes in each hidden layer as the sample size increases. We compare the performance of our approach against a hierarchical group lasso penalty function. The results show clear evidence that our method outperforms others in detecting Granger causality.

The empirical application shows that after the introduction of the Tobalaba network there is an increase in the number of edges among the 17 companies studied. Moreover, the centrality measures obtained show an increase in the number of central nodes in the network after the introduction of the new platform. Our results demonstrate how the introduction of the blockchain platform has changed the structure of the connections between the firms trading in the platform due to the introduction of smart contracts and disintermediation. The application of the Diebold–Mariano test (1995) shows that the Granger causal network constructed using the algorithm proposed in this paper outperforms, in terms of forecast accuracy, several linear VAR(K) models in low and high dimensions, and provides empirical evidence of the importance of our nonlinear method for forecasting.

Since the paper focused on Granger causality, forecasting, estimation, and inference within regularized neural networks were thus not analyzed. The recent contribution by Hecq et al. (2019) develops an LM test for Granger causality in high-dimensional VAR models based on penalized least squares estimations. To obtain a test that retains an appropriate size after the variable selection performed by the lasso, these authors propose a post-double selection procedure to partially remove the effects of nuisance variables and establish its uniform asymptotic validity. Although the method performs very well in high-dimensional settings, it is devised for linear parametric settings. In contrast, the method presented in this paper based on detecting Granger causality through sparsity

Table 5
Test statistics and *p*-values for the one sided Diebold–Mariano (1995) test.

	ANA	AES	CNA	DUK	EGIE3	EQNR	EXC	FLUVIU	GE	IBE	IGY	ITRI	PCG	RDSA	SIE	FP	WPRO
(09/05/2016–29/03/2018)																	
VAR(10) - AIC																	
DM t-stat	0.7109	3.5557	−0.3475	4.1242	3.4742	2.7946	1.8170	1.7894	3.3899	−0.1324	5.9087	4.2003	3.7315	0.6025	1.1613	0.5325	2.8867
P-value	0.2398	0.0003	0.6354	<.0001	0.0005	0.0034	0.0367	0.0390	0.0006	0.5526	<.0001	<.0001	0.0002	0.2744	0.1248	0.2980	0.0025
VAR(10) - SC/BIC																	
DM t-stat	0.8314	1.4418	−0.3375	5.0691	3.2742	1.8498	3.7063	−0.0756	1.2011	−0.2029	5.1596	4.0094	4.4937	0.6554	0.5775	0.97435	2.4512
P-value	0.2043	0.0770	0.6316	<.0001	0.0009	0.0343	0.0002	0.5301	0.1169	0.5801	<.0001	<.0001	<.0001	0.2572	0.2827	0.1666	0.0008
VAR(10) - H. Lasso																	
DM t-stat	0.5715	2.8837	3.2669	4.6904	5.5815	1.8014	4.3471	−1.3607	1.8002	−0.4502	5.9700	5.7816	2.2178	−0.3449	1.0025	−0.7035	6.7251
P-value	0.2848	0.0026	0.0008	<.0001	<.0001	0.038	<.0001	0.9110	0.0381	0.6730	<.0001	<.0001	0.0149	0.6344	0.1598	0.7580	<.0001
ARIMA																	
DM t-stat	0.2950	1.6813	−0.1464	4.0966	1.3429	0.6443	4.2751	−1.1433	3.3477	3.1960	1.9294	2.4249	1.5788	1.7826	1.9389	1.5764	4.4666
P-value	0.3844	0.04862	0.558	<.0001	0.0918	0.2608	<.0001	0.8761	0.0006	0.0011	0.0289	0.0009	0.0598	0.0395	0.0283	0.0598	<.0001
(26/10/2018–10/05/2019)																	
VAR(10) - AIC																	
DM t-stat	4.4708	5.4614	4.4851	5.3019	7.1449	5.2264	5.3268	3.7235	6.3085	4.6958	4.6995	4.0900	5.8294	6.0783	3.7364	5.5485	5.7272
P-value	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0002	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0002	<.0001	<.0001
VAR(10) - SC/BIC																	
DM t-stat	5.5601	3.5373	4.1454	3.5823	4.2906	2.3727	4.8180	3.5271	5.7450	4.8650	3.6427	4.6288	4.4794	4.9785	3.3436	4.1410	5.3268
P-value	<.0001	0.0004	<.0001	0.0003	<.0001	0.0102	<.0001	0.0004	<.0001	<.0001	0.0003	<.0001	<.0001	<.0001	0.0007	<.0001	<.0001
VAR(10) - H. Lasso																	
DM t-stat	4.1515	−1.3573	2.6058	0.0533	3.8638	2.5957	1.6365	−0.36215	−0.29011	3.9205	−1.5090	7.9074	2.1622	1.7082	3.9536	−0.5101	3.7499
P-value	<.0001	0.9105	0.0056	0.4788	0.0001	0.0057	0.0531	0.6408	0.6137	0.0001	0.9321	<.0001	0.01703	0.0460	<.0001	0.6942	0.0002
ARIMA																	
DM t-stat	−0.6447	1.3833	1.2962	3.2208	2.3887	1.6395	3.3682	0.87364	4.8382	−0.4006	2.6907	2.2173	0.7802	1.8874	0.9494	0.3256	3.4211
P-value	0.7394	0.0855	0.0996	0.0009	0.0098	0.0523	0.0006	0.1927	<.0001	0.6550	0.0044	0.0149	0.2190	0.0316	0.1729	0.3728	0.0005

induction presents a suitable alternative that works in more general settings. Future research will extend the current work to the derivation of nonasymptotic bounds for regularized and non-regularized neural networks (Farrell et al., 2018), as well as the limiting distributions for the two-step estimator proposed in this paper.

Another potential extension of the current research is to couple our methodology to detect Granger causality with the graphic theory introduced by Eichler (2007), Eichler and Didelez (2012). The main advantages of the analysis of Granger causality in graph theory are the possibility of visualizing the complex dependence structures that may underline multivariate time series and a definition of Granger causality that can be applied to multivariate time series with nonlinear dependencies. Therefore, by analyzing the matrix of the weights and error terms of a VAR defined by our feedforward neural networks approach, it may be possible to define the directed and undirected edges in a mixed path diagram in high-dimensional and potentially nonlinear time series.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.ijforecast.2020.10.004.

References

Athey, S., & Imbens, G. W. (2019). Machine learning methods that economists should know about. *Annual Review of Economics*, 11.

Bańbura, M., Giannone, D., & Reichlin, L. (2010). Large Bayesian vector auto regressions. *Journal of Applied Econometrics*, *25*(1), 71–92.

Belloni, A., Chernozhukov, V., & Kato, K. (2014). Uniform post-selection inference for least absolute deviation regression and other z-estimation problems. *Biometrika*, *102*(1), 77–94.

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.

Bloch, F., Jackson, M. O., & Tebaldi, P. (2017). Centrality measures in networks. Available at SSRN 2749124.

Bonancich, P. (1987). Power and centrality: A family of measures. *American Journal of Sociology*, *92*(5), 1170–1182.

Box, G. E., & Tiao, G. C. (1977). A canonical analysis of multiple time series. *Biometrika*, *64*(2), 355–365.

Chakraborty, K., Mehrotra, K., Mohan, C. K., & Ranka, S. (1992). Forecasting the behavior of multivariate time series using neural networks. *Neural Networks*, *5*(6), 961–970.

Cybenko, G. (1989). Approximation by superposition of a sigmoidal function. *Mathematics of Control, Signals and Systems*, *2*(4), 303–314.

De Veciana, G., & Zakhor, A. (1992). Neural net based continuous phase modulation receivers. *IEEE Transactions on Communications*, *40*(8), 1396–1408.

Diebold, F. C., & Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, *13*(3), 253–263.

Dufour, J. M., & Taamouti, A. (2010). Short and long run causality measures: Theory and inference. *Journal of Econometrics*, *154*(1), 42–58.

Eichler, M. (2007). Granger causality and path diagrams for multivariate time series. *Journal of Econometrics*, *137*(2), 334–353.

Eichler, M., & Didelez, V. (2012). Causal reasoning in graphical time series models. arXiv preprint arXiv:1206.5246.

Energy Web Foundation (2018). The energy web chain: Accelerating the energy transition with an open-source, decentralized blockchain platform. Available at: http://www.energyweb.org/papers/the-energy-web-chain.

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, *96*(456), 1348–1360.

Farrell, M. H., Liang, T., & Misra, S. (2018). Deep neural networks for estimation and inference: Application to causal effects and other semiparametric estimands. arXiv preprint arXiv:1809.09953.

Forni, M., Hallin, M., Lippi, M., & Reichlin, L. (2000). The generalized dynamic-factor model: Identification and estimation. *Review of Economics and Statistics*, *82*(4), 540–554.

Géron, A. (2017). *Hans-on machine learning with scikit-learn & tensorflow*. O'Reilly.

Geweke, J. (1982). Measurement of linear dependence and feedback between multiple time series. *Journal of the American Statistical Association*, *77*(378), 304–313.

Geweke, J. (1984). Inference and causality in economic time series models. *Handbook of Econometrics*, *2*, 1101–1144.

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteen international conference on artificial intelligence and statistics* (pp. 249–256).

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, *37*(3), 424–438.

Hastie, T., Tibshirani, R., & Friedman, J. (2005). *The elements of statistical learning*. Springer.

Hecq, A., Margaritella, L., & Skeekes, S. (2019). Granger causality testing in high-dimensional VARs: a post-double-selection procedure. arXiv preprint arXiv:1902.10991v3.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, *4*(2), 251–257.

Kaastra, I., & Boyd, M. (1996). Designing a neural network for forecasting financial and economic time series. *Neurocomputing*, *10*(3), 215–236.

Koop, G. M. (2013). Forecasting with medium and large Bayesian VARs. *Journal of Applied Econometrics*, *28*(2), 177–203.

Kotzanikolaou, P., Theoharidou, M., & Gritzalis, D. (2013). Assessing n-order dependencies between critical infrastructures. *International Journal of Critical Infrastructures*, *6*(9(1-2)), 93–110.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Lütkepohl, H. (1985). Comparison of criteria for estimating the order of a vector autoregressive process. *Journal of Time Series Analysis*, *6*(1), 35–52.

Montgomery, M. C., & Eledath, J. K. (1995). Maximum information transfer in feedforward neural networks. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.27.330&rep=rep1&type=pdf.

Morgan, J. P. (2018). *Blockchain and the decentralization revolution* (pp. 1–21).

Nicholson, W. B., Wilms, I., Bien, J., & Matteson, D. S. (2014). High dimensional forecasting via interpretable vector autoregression. arXiv preprint arXiv:1412.5250.

Peña, D., & Box, G. E. (1987). Identifying a simplifying structure in time series. *Journal of the American statistical Association*, *82*(399), 836–843.

Plagborg-Møller, M., & Wolf, C. K. (2020). Local projections and vars estimate the same impulse responses. *Econometrica*, forthcoming.

Reed, R., Marks, R. J., & Oh, S. (1995). Similarities of error regularization, sigmoid gain scaling, target smoothing, and training with jitter. *IEEE Transactions on Neural Networks*, *6*(3), 529–538.

Rodrigues, F. A. (2019). Network centrality: an introduction. arXiv preprint arXiv:1901.07901v1.

Scardapane, S., Comminiello, D., Hussain, A., & Uncini, A. (2017). Group sparse regularization for deep neural networks. *Neurocomputing*, *241*, 81–89.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117.

Schreiber, T. (2000). Measuring information transfer. *Physical Review Letters*, *85*(2), 165–461.

Simon, N., Friedman, J., Hastie, T., & Tibshirani, R. (2013). A sparse-group lasso. *Journal of Computational and Graphical Statistics*, *22*(2), 231–245.

Sims, C. A. (1980). Macroeconomics and reality. *Econometrica: Journal of the Econometric Society*, *48*(1), 1–48.

Skripnikov, A., & Michailidis, G. (2019). Joint estimation of multiple network Granger causal models. *Econometrics and Statistics*, *10*, 120–133.

Song, X., & Taamouti, A. (2018). Measuring nonlinear granger causality in mean. *Journal of Business & Economic Statistics*, *36*(2), 321–333.

Song, X., & Taamouti, A. (2019). A better understanding of granger causality analysis: A big data environment. *Oxford Bulletin of Economics and Statistics*, *81*(4), 911–936.

Stekhoven, D. J. (2013). missForest: Nonparametric missing value imputation using random forest, R package version 1.4.0.

Stock, J. H., & Watson, M. W. (2002). Forecasting using principal components from a large number of predictors. *Jounral of the American Statistical Association*, *97*(460), 1167–1179.

Taamouti, A., Bouezmarni, T., & El Ghouch, A. (2014). Nonparametric estimation and inference for conditional density based Granger causality measures. *Journal of Econometrics*, *180*(2), 251–264.

Tank, A., Covert, I., Foti, N., Shojaie, A., & Fox, E. (2018). Neural Granger causality for time series. arXiv preprint arXiv:1802.05842.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, *58*(1), 267–288.

Urban, S. (2017). *Neural network architectures and activation functions: A Gaussian process approach* (Doctoral dissertation), Technische Universität München.

Wiener, N. (1956). *The theory of prediction in modern mathematics for engineers*. New York: McGraw-Hill.

Wilms, I., Gelper, S., & Croux, C. (2016). The predictive power of the business and bank sentiment of firms: A high-dimensional granger causality approach. *European Journal of Operational Research*, *254*(1), 138–147.

World Bank (2018). Distributed ledger technology (DLT) and blockchain. *FinTech Note*, *1*, 1–60.

Yuan, M., & Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, *68*(1), 49–67.

Zhou, N., & Zhu, J. (2010). Group variable selection via a hierarchical lasso and its oracle property. arXiv preprint arXiv:1006.2871.