Pure Mathematics 理论数学, 2025, 15(5), 36-49
Published Online May 2025 in Hans. https://www.hanspub.org/journal/pm
https://doi.org/10.12677/pm.2025.155151

Hans汉斯

GCGAN：基于生成对抗网络的时间序列Granger因果关系发现

吴光强

上海理工大学理学院，上海

收稿日期：2025年3月18日；录用日期：2025年4月25日；发布日期：2025年5月13日

摘　要

Granger因果关系在时间序列建模中具有重要意义，但传统方法存在难以捕捉复杂非线性关系等诸多局限性。本文在相关工作的基础上提出GCGAN模型，首次将生成对抗网络应用于时间序列中的Granger因果关系发现。通过设计多头生成器的架构建模目标序列，在训练时对生成器施加稀疏诱导惩罚项，并在提取因果关系矩阵时引入阈值方法，GCGAN能够精准找寻高维时间序列中的复杂Granger因果关系。本研究为时间序列Granger因果关系发现提供了新的深度学习范式。

关键词

时间序列分析，Granger因果关系，深度生成模型，生成对抗网络

GCGAN: Time Series Granger Causality Discovery Based on Generative Adversarial Networks

Guangqiang Wu

College of Science, University of Shanghai for Science and Technology, Shanghai

Received: Mar. 18th, 2025; accepted: Apr. 25th, 2025; published: May 13th, 2025

Abstract

Granger causality plays a significant role in time series modeling. However, traditional methods struggle with capturing complex nonlinear relationships among others limitations. Building on previous studies, this paper introduces the GCGAN model, marking the first application of Generative Adversarial Networks (GANs) to Granger causality discovery in time series. By designing a multi-

文章引用: 吴光强. GCGAN: 基于生成对抗网络的时间序列 Granger 因果关系发现[J]. 理论数学, 2025, 15(5): 36-49.
DOI: 10.12677/pm.2025.155151

head generator architecture to model target sequences, imposing sparse inducing penalties on the generator during training, and introducing a threshold method when extracting the causality matrix, GCGAN can accurately identify complex Granger causalities in high-dimensional time series. This study provides a novel deep learning paradigm for the discovery of Granger causality in time series.

Keywords

Time Series Analysis, Granger Causality, Deep Generative Models, Generative Adversarial Networks

Copyright © 2025 by author(s) and Hans Publishers Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
http://creativecommons.org/licenses/by/4.0/
Open Access

1. 引言

Granger 因果关系作为时间序列分析中的重要概念，自 Granger 在 1969 年提出以来[1]，在经济学、金融学、神经科学及环境科学等领域展现出独特的理论价值与应用潜力[2]-[6]。其核心在于通过统计检验判断一个时间序列变量是否包含对另一变量未来值的预测信息。通过 Granger 因果关系检验，我们可以确定一个时间序列是否包含了另一个时间序列未来值的有用信息，这种信息超越了仅依赖于该序列自身历史值所能提供的预测精度。在宏观经济学中，Granger 因果检验被广泛用于验证货币政策与通货膨胀、GDP 增长率与失业率等关键变量间的动态关系[5]；在金融市场分析中，分析 Granger 因果关系有助于识别出哪些经济指标能够更准确地预测股票市场波动，对于风险管理与投资策略制定具有至关重要的意义[3]。值得注意的是，Granger 因果关系虽然并不等同于真实因果关系，但其对预测性因果的识别能力为效果评估和复杂系统建模提供了不可替代的方法论支撑。

传统的 Granger 因果关系发现方法主要基于线性假设，这限制了其在处理非线性关系时的有效性[1][7]。例如，当使用向量自回归模型进行 Granger 因果关系检验时[8]，通常假定各时间序列间的影响是线性的。这意味着，对于那些存在复杂非线性相互作用的时间序列数据，如金融市场中的价格波动或生物系统中的基因表达模式，传统方法可能无法准确捕捉到这些非线性依赖关系[7]。此外，选择适当的时间滞后长度也是传统 Granger 因果关系发现中的一个重要挑战，若选择不当，可能会导致遗漏真实的因果关系或引入不必要的噪声。另一个局限性在于，随着研究对象维度的增加，即考虑更多时间序列及其历史值时，模型复杂度和计算负担也随之增大[9]。虽然稀疏正则化技术如 Lasso 可以帮助缓解这一问题，但它们并不能完全解决高维数据环境下的过拟合风险[10]。因此，尽管传统的 Granger 因果关系发现方法提供了一种有效的方式去揭示时间序列间的结构关系，但在处理非线性和高维度数据时，仍需谨慎对待其适用范围和局限性。

近年来，随着深度学习理论的提出及广泛应用[11]，神经网络在时间序列分析中展现了强大的能力，特别是在捕捉复杂的非线性关系方面。相较于传统的时间序列分析模型，神经网络例如多层感知器和长短期记忆网络[12] [13]，因其出色的非线性拟合能力和预测性能而受到青睐。为了解决传统的 Granger 因果关系发现方法存在的问题，不少研究者已经尝试将神经网络引入时间序列的 Granger 因果关系发现中。Nauta 等人提出了一种基于注意力机制的卷积神经网络方法来进行因果发现[14]，名为 Temporal Causal Discovery Framework (TCDF)，它能够从未处理的时间序列数据中识别变量间的因果关系，弥补了现有技术中未能充分利用深度学习表示能力的不足；Tank 等人提出了一种基于神经网络的方法 Neural Granger

Causality (NGC) [15]，该方法在训练神经网络时结合权重上的稀疏诱导惩罚来识别复杂的因果交互；Li 等人提出了一种名为因果递归变分自编码器(Causal Recurrent Variational Autoencoder, CRVAE)的因果发现方法[16]，首次将深度生成模型应用于时间序列的因果发现中。

生成对抗网络(Generative Adversarial Networks, GANs)作为一类经典的深度生成模型[17]，自提出以来便在图像领域引起了巨大的反响。鉴于 GANs 在图像生成任务上的成功，研究者们积极探索了其在其他类型数据上的应用潜力，尤其是时间序列数据。在时间序列分析中，GANs 已被用于时间序列预测、异常检测、填补缺失值以及数据增强等任务[18]-[21]，并取得了令人振奋的效果。

然而，在时间序列因果发现这一特定任务上，GANs 的应用却鲜有涉及。GANs 有能力模拟极其复杂的数据分布，理论上也应当能够在捕捉时间序列中的非线性依赖关系及潜在因果结构方面发挥重要作用。因此，探索如何将 GANs 应用于时间序列因果发现不仅是一个新颖的方向，而且有望挖掘出这种强大的生成模型在这类任务中未被充分认识的潜力，为解决因果推断问题提供新的视角和技术手段。因此，本文提出 GCGAN 模型，首次将 GANs 应用于时间序列中的 Granger 因果发现领域，主要贡献包括：

- 设计了多头生成器的架构，每个生成器专注于单个目标序列的建模；
- 在生成器输入层引入了稀疏性诱导惩罚；
- 在提取 Granger 因果关系矩阵时，提出了一种有效确定阈值的方法；
- 对于提出的 GCGAN 模型，在多个数据集上验证了其卓越性。

2. 背景

2.1. 时间序列的 Granger 因果发现

Norbert Wiener 是首位在时间序列分析中引入“因果关系”概念的数学家[22] [23]。根据 Wiener 的观点，如果在一个统计学意义上，通过包含关于变量 x 的信息能够改进对变量 y 的预测，则可以说时间序列或变量 x 对另一个变量 y 产生了因果影响。这种定义强调了因果关系的信息理论基础。然而，直到 1969 年，Clive Granger 才进一步发展了这一思想，并提出了 Granger 因果性[1]。Granger 在其多元自回归模型(MVAR)框架下定义了因果性，通过比较预测误差方差来判断是否考虑 x 能够提高对 y 的预测精度。具体来说，在不考虑 x 的情况下，我们有模型：

$$y_t = \alpha + \sum_{i=1}^{p} \beta_i y_{t-i} + \epsilon_t,$$

其中，ϵ_t 是残差项。而当加入 x 后，模型变为：

$$y_t = \alpha' + \sum_{i=1}^{p} \beta_i' y_{t-i} + \sum_{j=1}^{q} \gamma_j x_{t-j} + \epsilon_t'.$$

如果 ϵ_t' 的方差显著小于 ϵ_t，则可以认为 x 对 y 存在 Granger 因果影响。

后来，Granger 因果性的基本理念不仅限于线性情况，还被扩展到了非线性领域。例如，Marinazzo 等人在 2008 年使用核方法将 Granger 因果性推广至非线性场景[24]，通过计算条件互信息量来评估因果关系。此外，Chen 等人在 2004 年提出了一种基于重构相空间的方法[25]，通过拟合局部线性模型来探索时间序列间的非线性因果关系。

近年来，深度学习技术也被应用于时间序列 Granger 因果发现。Nauta 等人在 2019 年提出的 Temporal Causal Discovery Framework (TCDF)利用带有扩张深度分离卷积网络中的注意力机制来学习复杂的非线性因果关系[14]。TCDF 不仅能够识别直接因果路径，还能揭示间接因果路径的存在。具体

而言，TCDF 采用残差网络结构以有效地处理长时间依赖性，并结合注意力机制来突出显示对预测目标变量至关重要的输入时间序列部分。此外，通过调整注意力得分，使得模型能够动态地强调或削弱不同输入特征的影响，从而提高因果发现的精度。Tank 等人在 2021 年提出的 Neural Granger Causality (NGC)是首个利用稀疏约束于自回归网络权重上以学习因果图的方法[15]。这种方法通过在训练过程中应用 L1 正则化来促使网络学习到一个稀疏的因果结构，从而更准确地发现重要的因果关系。其核心在于构建一个组件式神经网络模型，对于每个输出分量 $x_{t:t+\tau,i}$，采用单独的组件 f_i 来映射时间序列在过去时间点的输入：

$$x_{t:t+\tau,i} = f_i\left(x_{<t,1}, x_{<t,2}, \cdots, x_{<t,M}\right) + e_{t:t+\tau,i},$$

这里，f_i 代表针对第 i 个变量序列设计的神经网络，它定义了如何从过去的时间序列数据映射到序列 i。在这一框架下，Granger 非因果关系意味着 f_i 不依赖于序列 j 的任何过去滞后值 $x_{t-k,j}$。通过这种方式，不仅解决了标准神经网络难以解释的问题，而且允许每个时间序列根据实际需要选择不同的历史滞后，提高了模型的灵活性和准确性。此外，Li 等人在 2023 年提出了的 Causal Recurrent Variational Autoencoder (CRVAE)首次将深度生成模型应用于时间序列的因果发现中[16]，并将底层的因果机制融入数据生成过程中。

2.2. 生成对抗网络

生成对抗网络(Generative Adversarial Networks, GANs) [17]，由 Goodfellow 等人于 2014 年提出，是一类旨在生成与训练数据相似的新数据样本的深度学习模型。GAN 主要由两大部分组成：生成器(Generator, G)和判别器(Discriminator, D)。生成器接收一个随机噪声向量 z 作为输入，并输出试图模仿真实数据分布的合成数据样本。判别器则评估给定样本是真实的还是伪造的，力求准确区分真实数据与生成的数据。

GAN 的训练过程涉及生成器与判别器之间的对抗游戏。生成器努力产生与真实数据无异的样本，而判别器的目标是正确分类样本为真实或伪造。这种竞争促使双方不断改进。数学上，GAN 的目标函数可以表述为：

$$\min_G \max_D \mathbb{E}_{x\sim P_{data}}\left[\log D(x)\right] + \mathbb{E}_{z\sim P_z}\left[\log\left(1 - D\left(G(z)\right)\right)\right], \tag{1}$$

这里，x 表示从真实数据分布 P_{data} 中抽取的真实数据样本，z 是从先验分布 P_z 中采样的随机噪声向量。判别器 D 输出一个概率分数，指示给定样本为真实的可能性。生成器 G 的目标是最大化判别器将生成样本错误分类为真实样本的概率。通过这种对抗性训练，生成器学会生成高度逼真的数据，使得 GAN 成为包括图像生成、数据增强以及时间序列分析等多种应用的强大工具。

3. 方法

3.1. 问题定义

设 $S = \left[s_1, s_2, \cdots, s_T\right]^{\mathrm{T}} = \left[s^{(1)}, s^{(2)}, \cdots, s^{(M)}\right] \in \mathbb{R}^{T\times M}$ 表示一个多变量时间序列，其中 $s_t \in \mathbb{R}^M$ 表示第 t 个时间点的观测向量，包含了在同一时间点观测到的 M 个特征。第 j 个维度对应的单变量时间序列简称为序列 j。Granger 因果关系发现的目标是构建时间序列 S 的 Granger 因果关系矩阵 $A \in \{0,1\}^{M\times M}$，其中 $A_{ij} = 1$ 表示序列 j Granger 导致序列 i，记为 $j \xrightarrow{\text{Granger}} i$。关于 Granger 因果关系的定义如下：

定义[1]：设有序列 i 和序列 j，我们称序列 j Granger 导致序列 i，如果利用序列 j 的过去值可以提供对序列 i 未来值的额外预测信息，这种额外的信息是在仅已知序列 i 自身历史值的情况下所不能获得的。

换句话说，若在考虑了序列 i 自身的历史信息之后，进一步包含序列 j 的历史信息能够显著提高对序列 i 未来值的预测准确性。

3.2. GCGAN

为了得到时间序列中的 Granger 因果关系，我们需要先构建一个时间序列预测模型。生成对抗网络最初被设计用于图像生成任务，其中生成器接收随机噪声作为输入，通过学习真实数据集的分布特征来生成逼真的图像样本。对于图像生成任务，生成器和判别器的设计主要集中在捕捉和再现高维空间中的复杂模式，如图像的空间结构和纹理信息。然而，当转向时间序列这一特定应用场景时，我们需要对经典生成对抗网络的基本架构进行适当的调整以适应新的需求。

首先，在输入层面，生成器不再接受随机噪声作为输入，而是应当采用历史时间序列数据。这是因为时间序列具有显著的时间依赖性，生成器的任务是基于已有的时间序列数据预测未来值，而非创造全新的、无关联的数据点。这种变化要求生成器能够理解和模拟时间序列中的动态演变过程，从而产生既符合历史趋势又具备预测价值的未来数据。

其次，在输出方面，时间序列预测关注的是高度有序的数据流。这意味着生成器不仅要确保生成的时间序列数据在统计特性上与真实数据相匹配，还需要保持时间上的连贯性和一致性。为了实现这一点，生成器的学习目标需要进一步调整，以便更好地捕捉时间序列中的长期依赖关系和短期波动特性。

GCGAN 的整体架构如图 1 所示，包含了多头生成器 $G=\left[G^{(1)},G^{(2)},\cdots,G^{(M)}\right]$ 和判别器 D。在完成整个模型的训练后，我们将从生成器 G 的网络参数中提取时间序列的 Granger 因果关系矩阵 $\hat{A}$。

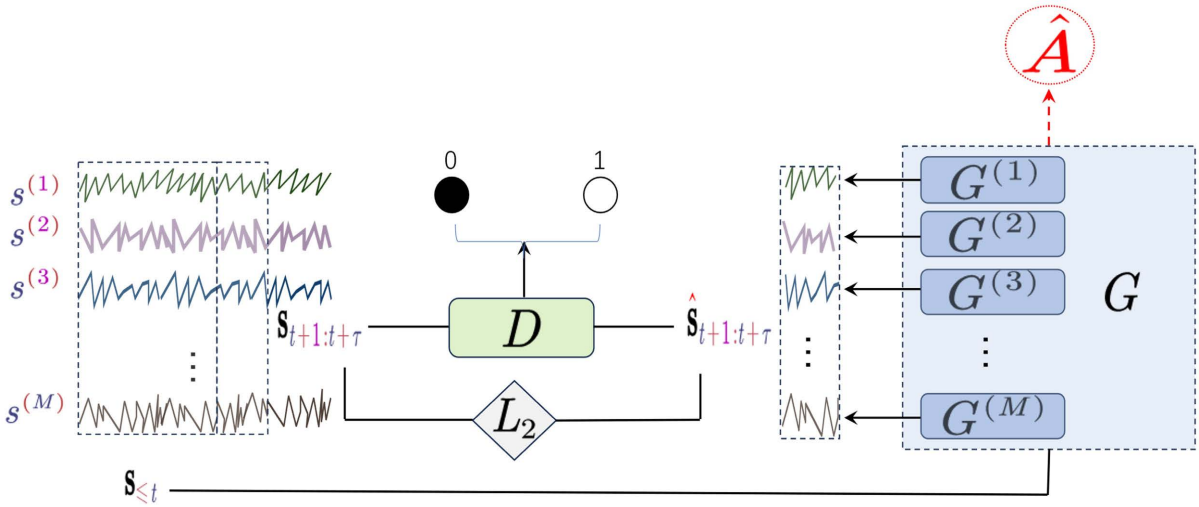

Figure 1. Diagram of the GCGAN model architecture
图 1. GCGAN 模型架构示意图

3.2.1. 多头生成器设计

在以往工作的生成对抗网络中，即使是用以时间序列预测，生成器也一般被设计为共享权重的单个神经网络模型。然而，这样的单一生成器结构可能不足以捕捉复杂系统内部各成分间复杂的相互作用，特别是在考虑 Granger 因果关系时。例如，$j \xrightarrow{\text{Granger}} i$ 但 $j \overset{\text{Granger}}{\nrightarrow} i+1$。为此，我们借鉴 NGC [15]的思想，提出了一种多头生成器(multi-head generator)的设计，旨在通过更细致地建模每个时间序列与其他序列之间的相互影响来提高 Granger 因果关系发现的准确性。

生成器的每一头 $G^{(i)}$ 专门负责一个特定的时间序列 $s^{(i)}$ 的预测任务。设预测窗口大小为 τ，当前时刻为 t，则

$$\hat{\boldsymbol{s}}_{t+1:t+\tau}^{(i)} = G^{(i)}\left(\boldsymbol{s}_{\le t}; W^{(i)}\right),$$

其中，$\hat{\boldsymbol{s}}_{t+1:t+\tau}^{(i)}$ 表示对时间序列 $s^{(i)}$ 在未来时间窗 $t+1$ 至 $t+\tau$ 的预测值，$\boldsymbol{s}_{\le t}$ 代表多变量序列 S 在前 t 个时间点的观测值集合，$W^{(i)}$ 是第 i 头生成器 $G^{(i)}$ 的网络参数集。

通过引入多头独立但协同工作的生成器，我们的方法能够更加灵活地学习不同序列间的动态依赖关系，进而有效识别那些真正具有因果意义的连接。在有了每一头生成器输出的预测值后，我们将其按列拼接作为整个生成器 $G = \left[G^{(1)}, G^{(2)}, \cdots, G^{(M)}\right]$ 的输出：

$$\hat{\boldsymbol{s}}_{t+1:t+\tau} = G\left(\boldsymbol{s}_{\le t}; W\right) = \left[\hat{\boldsymbol{s}}_{t+1:t+\tau}^{(1)}, \hat{\boldsymbol{s}}_{t+1:t+\tau}^{(2)}, \cdots, \hat{\boldsymbol{s}}_{t+1:t+\tau}^{(M)}\right], \tag{2}$$

其中，$\hat{\boldsymbol{s}}_{t+1:t+\tau}$ 表示对时间序列 S 在未来时间窗 $t+1$ 至 $t+\tau$ 的预测值，W 是整个生成器的网络参数集。

3.2.2. 生成器的网络结构

为了更好地捕捉时间序列中长程、非线性的依赖关系，时间序列建模中有以下常用的神经网络模型：多层感知器(MLP)和长短期记忆网络(LSTM) [12] [13]。本文中，我们采用长短期记忆网络对每一头生成器 $G^{(i)}$ 进行设计。

设当前时刻为 t_0，输入序列为 $\boldsymbol{s}_{\le t_0}$，我们先初始化隐藏状态 h_0 和细胞状态 c_0 为零张量，后按以下计算过程对 $t = 0, 1, \cdots, t_0$ 进行递推直到计算出最终的隐藏状态 h_{t_0}：

$$\begin{aligned}
f_t &= \sigma\left(W_f^{(i)} h_{t-1} + U_f^{(i)} s_t + b_f^{(i)}\right), \\
i_t &= \sigma\left(W_{in}^{(i)} h_{t-1} + U_{in}^{(i)} s_t + b_{in}^{(i)}\right), \\
\tilde{c}_t &= \tanh\left(W_c^{(i)} h_{t-1} + U_c^{(i)} s_t + b_c^{(i)}\right), \\
c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \\
o_t &= \sigma\left(W_o^{(i)} h_{t-1} + U_o^{(i)} s_t + b_o^{(i)}\right), \\
h_t &= o_t \odot \tanh\left(c_t\right),
\end{aligned} \tag{3}$$

其中，

- f_t、i_t 和 o_t 分别是遗忘门、输入门和输出门的激活值；
- $\tilde{c}_t$ 是候选细胞状态；
- c_t 是当前时间步的细胞状态；
- h_t 是当前时间步的隐藏状态；
- σ 表示 Sigmoid 函数；
- $\odot$ 表示逐元素乘法；
- $W_f^{(i)}, W_{in}^{(i)}, W_c^{(i)}, W_o^{(i)}$ 是权重矩阵，用于处理隐藏状态；
- $U_f^{(i)}, U_{in}^{(i)}, U_c^{(i)}, U_o^{(i)}$ 是权重矩阵，用于处理输入序列；
- $b_f^{(i)}, b_{in}^{(i)}, b_c^{(i)}, b_o^{(i)}$ 是偏置向量。

在完成递推计算后，最终的隐藏状态 h_{t_0} 包含了时间序列中迄今为止的所有信息。最后，为了映射到目标输出空间，我们对其进行线性变换，输出预测值：

$$G^{(i)}\left(\boldsymbol{s}_{\le t_0}; W^{(i)}\right) = W_{out}^{(i)} h_{t_0} + b_{out}^{(i)},$$

其中，$W_{out}^{(i)}$ 和 $b_{out}^{(i)}$ 是输出层的权重矩阵和偏置向量。

3.2.3. 判别器的网络结构

在我们的模型中，判别器的设计至关重要。判别器接收的时间序列窗口为 $\boldsymbol{s}_{t+1:t+\tau}$ 或 $\hat{\boldsymbol{s}}_{t+1:t+\tau}$，我们可以将其视为单通道图像，从而利用图像处理领域常用的神经网络架构来构建判别器。通过使用二维卷积层[26]，可以在一次操作中覆盖整个时间序列窗口及其所有特征维度，使得模型能够一次性提取全局特征，而不是逐步处理每个时间步。这种全局视角有助于捕捉长程依赖关系和复杂模式，这对于判别器区分真实数据与生成数据尤为重要。

设判别器输入的时间序列窗口为 $\boldsymbol{s}_{t+1:t+\tau} \in \mathbb{R}^{\tau\times M}$，其中 τ 代表预测时间窗口长度，M 表示时间序列的特征数量。二维卷积核(权重矩阵)为 $W \in \mathbb{R}^{K_h\times K_w}$，偏置项为 $b \in \mathbb{R}$。判别器中经过二维卷积操作后，在位置 (i,j) 的输出特征图元素可以表示为：

$$o_{i,j} = a\left(\sum_{m=0}^{K_h-1}\sum_{n=0}^{K_w-1} W_{m,n}\cdot \boldsymbol{s}_{i+m,j+n} + b\right),$$

其中，

- $o_{i,j}$ 表示输出特征图在位置 (i,j) 的值；
- $W_{m,n}$ 是卷积核中位置 (m,n) 处的权重；
- $\boldsymbol{s}_{i+m,j+n}$ 是输入时间序列窗口相应位置处的值；
- a 表示激活函数(例如 ReLU)。

对于整个输出特征图 O，有：

$$O = \left[o_{i,j}\right]_{\substack{i=1,\cdots,\tau-K_h+1\\ j=1,\cdots,M-K_w+1}}.$$

为了使判别器输出最终的概率值，需要将输出特征图进一步通过全连接层处理。首先，将输出特征图 O 展平成一维向量，展平后的向量可以表示为：

$$\mathrm{vec}(O) = \left[o_{1,1}, o_{1,2}, \cdots, o_{1,M-K_w+1}, o_{2,1}, \cdots, o_{\tau-K_h+1,M-K_w+1}\right].$$

接着，我们将这个向量通过一个全连接层。设全连接层的权重向量为 $W_f \in \mathbb{R}^{1\times(\tau-K_h+1)(M-K_w+1)}$，偏置项为 $b_f \in \mathbb{R}$。经过线性变换并应用 Sigmoid 激活函数来获得判别器的最终输出：

$$D(\boldsymbol{s}_{t+1:t+\tau}) = \sigma\left(W_f\cdot \mathrm{vec}(O) + b_f\right) = \frac{1}{1+\mathrm{e}^{-\left(W_f\cdot \mathrm{vec}(O)+b_f\right)}}.$$

3.2.4. 目标函数

根据经典生成对抗网络的对抗损失(1)，我们容易得到 GCGAN 的对抗损失：

$$\min_G \max_D \sum_{t=1}^{T-\tau}\left(\log D(\boldsymbol{s}_{t+1:t+\tau}) + \log\left(1-D(\hat{\boldsymbol{s}}_{t+1:t+\tau})\right)\right),$$

其中，$\hat{\boldsymbol{s}}_{t+1:t+\tau}$ 已在式(2)中定义。

然而，在我们的模型中，仅依赖对抗损失进行训练可能无法保证生成样本的精确性与可靠性。对抗损失主要确保生成器输出的数据在分布层面接近真实数据集的统计特性，但这一策略可能导致生成器产出虽能“迷惑”判别器却偏离实际值的样本。为了克服此局限并提升预测准确性，我们引入了预测损失作为额外误差项[27]。具体而言，通过计算生成器每一头 $G^{(i)}$ 的生成样本与真实样本之间的 L_2 距离来量化预测损失，使得生成器不仅需模仿原始时序数据的整体分布特征，还需尽可能准确地预测未来值。复合损失函数为：

$$\min_G \max_D \sum_{t=1}^{T-\tau}\left[\log D\left(\boldsymbol{s}_{t+1:t+\tau}\right)+\log\left(1-D\left(\hat{\boldsymbol{s}}_{t+1:t+\tau}\right)\right)+\lambda\sum_{i=1}^{M}\left\|\boldsymbol{s}_{t+1:t+\tau}^{(i)}-\hat{\boldsymbol{s}}_{t+1:t+\tau}^{(i)}\right\|_2^2\right], \tag{4}$$

其中，λ 是调节对抗损失和预测损失之间权重的超参数。通过调节 λ，可以在对抗损失与预测损失之间取得平衡，既允许生成器生成符合真实数据分布特性的样本，又确保其数值预测的精确度。因此，把对抗损失和预测损失结合起来，不仅增强了模型对复杂数据分布的学习能力，同时也显著提高了对时间序列预测的准确性和稳定性，有效解决了生成对抗网络中生成器过于关注分布匹配而忽视数值精度的问题。

至此，我们建立了一个完整的时间序列预测模型，接下来考虑如何让模型在预测过程中倾向于寻找时间序列的 Granger 因果关系。对于预测序列 i 的 $G^{(i)}$ 来说，我们希望在预测过程中它能够筛除掉那些 $j \overset{\text{Granger}}{\nrightarrow} i$ 的序列 j，在神经网络上的体现就是 $G^{(i)}$ 在训练过程中对输入序列中的序列 j 分配越来越小且逐渐趋于 0 的权重。根据式(3)，用于处理输入序列的权重矩阵 $U_f^{(i)},U_{in}^{(i)},U_c^{(i)},U_o^{(i)}$ 的第 j 列都是序列 j 的系数向量，如果这些系数向量均为零向量，则序列 j 对于序列 i 的预测毫无关联，即 $j \overset{\text{Granger}}{\nrightarrow} i$。因此，记

$$\tilde{W}^{(i)}=\begin{bmatrix}U_f^{(i)}\\U_{in}^{(i)}\\U_c^{(i)}\\U_o^{(i)}\end{bmatrix}, \tag{5}$$

并在式(4)的基础上添加惩罚项，得到最终的目标函数：

$$\min_G \max_D \sum_{t=1}^{T-\tau}\left[\log D\left(\boldsymbol{s}_{t+1:t+\tau}\right)+\log\left(1-D\left(\hat{\boldsymbol{s}}_{t+1:t+\tau}\right)\right)+\lambda\sum_{i=1}^{M}\left\|\boldsymbol{s}_{t+1:t+\tau}^{(i)}-\hat{\boldsymbol{s}}_{t+1:t+\tau}^{(i)}\right\|_2^2\right]+\mu\sum_{i=1}^{M}R\left(\tilde{W}^{(i)}\right), \tag{6}$$

其中，$R(\cdot)$ 表示一个正则化函数(如 L_1 范数)，μ 是正则化系数，用于控制惩罚项对总损失的影响。

3.2.5. 优化方法

在目标函数(6)中，理想的正则化函数 $R(\cdot)$ 应选择为 L_0 范数，因为它精确地衡量了非零元素的数量，刚好能够达成我们稀疏性诱导惩罚的目的。然而，在神经网络框架内优化基于 L_0 范数的目标函数颇具难度。因此，我们采用 L_1 范数作为替代方案，从而使目标函数转化为标准的 Lasso 问题进行处理。

为了优化这个目标函数，我们仅对权重矩阵 $\tilde{W}^{(i)}$ 实施迭代收缩阈值算法(ISTA, Iterative Shrinkage-Thresholding Algorithm) [28]，而对其它所有参数则采用随机梯度下降(SGD) [29]。对于目标函数中不含惩罚性的凸部分损失 L，ISTA 通过以下步骤更新生成器每一头 $G^{(i)}$ 对应的权重矩阵 $\tilde{W}^{(i)}$：

$$\tilde{W}_k^{(i)}=\operatorname{prox}_{\gamma_k\mu R}\left(\tilde{W}_{k-1}^{(i)}-\gamma_k\nabla L\left(\tilde{W}_{k-1}^{(i)}\right)\right), \tag{7}$$

其中，$\operatorname{prox}_{\gamma_k\mu R}(\cdot)$ 是与正则化项 R 相关的近端算子，γ_k 是步长参数，$\nabla L\left(\tilde{W}^{(i)}\right)$ 表示损失函数 L 关于 $\tilde{W}^{(i)}$ 的梯度。对于 L_1 正则化，即 $R(\cdot)=|\cdot|_1$，其近端算子是软阈值操作(soft-thresholding operator)，定义如下：

$$\operatorname{prox}_{\gamma_k\mu R}(x)=S_{\gamma_k\mu}(x)=\operatorname{sign}(x)\max\left(|x|-\gamma_k\mu,0\right).$$

通过这种方式，我们可以有效地得到稀疏矩阵 $\tilde{W}^{(1)},\tilde{W}^{(2)},\cdots,\tilde{W}^{(M)}$，同时保持模型对时间序列预测的准确性。

3.2.6. Granger 因果关系矩阵的提取

在训练后，生成器中的权重矩阵 $\tilde{W}^{(1)},\tilde{W}^{(2)},\cdots,\tilde{W}^{(M)}$ 成为了稀疏矩阵。这些矩阵表示了不同时间序列之

间的潜在 Granger 因果关系。为了从这些权重矩阵中提取出 Granger 因果关系矩阵，我们定义：

$$A_0 = \left[\left\| \tilde{W}_{:j}^{(i)} \right\|_2 \right]_{M \times M},$$

其中，$\tilde{W}_{:j}^{(i)}$ 表示矩阵 $\tilde{W}^{(i)}$ 的第 j 列，$\left\| \tilde{W}_{:j}^{(i)} \right\|_2$ 表示该列向量的 L_2 范数，代表了输入序列中的序列 j 对输出序列 i 的影响强度。

然而，在实际训练过程中我们发现，尽管采用了促进稀疏性的正则化方法，实现绝对的稀疏化仍然具有挑战性，尤其是处理复杂时间序列的时候。具体来说，真正存在 Granger 因果关系的连接在矩阵 A_0 中往往保持较大的值，而不存在 Granger 因果关系的连接虽然理论上应为零，但在实践中有时会表现为较小但非零的值，这种现象可能是由于模型复杂性、数据噪声或优化过程中的局部极小值等因素导致的。

鉴于此，我们引入了一个阈值方法来区分显著的 Granger 因果关系与非显著的关系。设 A_0 的最大值和最小值分别为 M_0 和 m_0，我们定义阈值 γ 如下：

$$\gamma = \frac{M_0 + m_0}{2}.$$

接下来，我们将原始矩阵 A_0 与这个阈值进行比较，生成最终的二值化 Granger 因果关系矩阵 A：

$$A_{ij} = \begin{cases} 1 & \text{当} (A_0)_{ij} > \gamma \\ 0 & \text{其他} \end{cases},$$

其中，A_{ij} 表示矩阵 A 的 (i, j) 元。

最终得到的矩阵 A 是一个 $M \times M$ 的二值矩阵，$A_{ij} = 1$ 表示序列 j 对序列 i 存在显著的 Granger 因果关系。通过这种阈值方法，我们不仅能够有效地识别出显著的 Granger 因果关系，还能避免因训练过程中未能实现绝对稀疏化而导致的误判。

3.3. 算法总结

我们将 GCGAN 的训练和提取 Granger 因果关系矩阵的算法总结表 1 为和表 2：

Table 1. Algorithm for training the GCGAN Model
表 1. GCGAN 的训练算法

输入： $S \in \mathbb{R}^{T \times M}$：时间序列数据。
λ, μ：预测损失项和惩罚项的系数。
N：最大迭代次数。
N_d：判别器每轮更新的训练轮次。

```
初始化生成器 G 和判别器 D 的参数
for n = 1 to N do:
    根据式(6)计算目标函数
    for N_d = 1 to N_d do:
        使用 SGD 更新判别器 D 的参数
    end for
    for i = 1 to M do:
        根据式(7)使用 ISTA 更新 G^(i) 对应的特定权重矩阵 W̃^(i)
        使用 SGD 更新 G^(i) 的其余参数
    end for
end for
```

输出： 优化后的生成器 G_{θ^*} 和判别器 D_{φ^*}。

Table 2. Extracting the granger causality matrix using GCGAN
表 2. 使用 GCGAN 提取 Granger 因果关系矩阵

输入：经过训练的生成器 G。

根据式(5)从生成器 G 中取出特定权重矩阵 $\tilde{W}^{(1)},\tilde{W}^{(2)},\cdots,\tilde{W}^{(M)}$
for $i=1$ to M do:
  for $j=1$ to M do:
    $A_0(i,j)=\left\|\tilde{W}_{:j}^{(i)}\right\|_2$
  end for
end for
计算阈值 $\gamma=\dfrac{M_0+m_0}{2}$，其中 M_0 和 m_0 分别是 A_0 的最大值和最小值
for $i=1$ to M do:
  for $j=1$ to M do:
    if $A_0(i,j)>\gamma$ then
      $A(i,j)=1$
    else
      $A(i,j)=0$
    end if
  end for
end for

输出：Granger 因果关系矩阵 A。

4. 实验

我们将实验相关代码开源在 Github 链接：GCGAN。

4.1. 数据集

在本次实验中，我们使用了三种不同的动态系统模型生成数据集来评估我们的方法：耦合 Hénon 映射[30]、Lorenz-96 模型[31]以及向量自回归模型 VAR (3) [8]。

Hénon：采用了六个相互关联的 Hénon 混沌映射作为研究对象，这种模型被广泛用于探索复杂系统的因果关系结构。在实验中，我们采用映射 $\boldsymbol{s}^{(i-1)}\to\boldsymbol{s}^{(i)}$ 模拟一个简单的动态系统，并生成了 2001 个时间序列数据点。

Lorenz-96：采用了由 Edward Lorenz 提出的 Lorenz-96 模型(Lorenz, 1996)。这个模型被广泛应用于气象学和气候科学中，用于模拟大气动力学行为：

$$\frac{\mathrm{d}\boldsymbol{s}_t^{(i)}}{\mathrm{d}t}=\left(\boldsymbol{s}^{(i+1)}-\boldsymbol{s}^{(i-2)}\right)\boldsymbol{s}^{(i-1)}-\boldsymbol{s}^{(i)}+F,$$

其中 $\boldsymbol{s}^{(-1)}=\boldsymbol{s}^{(M-1)}$，$\boldsymbol{s}^{(0)}=\boldsymbol{s}^{(M)}$ 和 $\boldsymbol{s}^{(M+1)}=\boldsymbol{s}^{(1)}$，$F$ 是外部强迫力(在我们的例子中设为 10)。该模型的关键特征在于其非线性项 $\left(\boldsymbol{s}^{(i+1)}-\boldsymbol{s}^{(i-2)}\right)\boldsymbol{s}^{(i-1)}$，它引入了序列之间的复杂相互作用。此外，边界条件确保了系统的周期性和连续性。在实验中，我们选择了 $M=10$ 维度，即系统中有 10 个序列。每个序列都遵循上述微分方程，但彼此之间通过非线性项相互影响。我们使用数值积分方法生成了 1000 个时间序列数据点。

VAR (3)：对具有 $M=10$ 个变量的向量自回归模型 VAR (3)进行了仿真。每个时间序列不仅包含其自身的时间滞后影响(即一阶、二阶和三阶滞后)，还通过随机选择若干其他变量作为依赖项来构建稀疏连接矩阵。具体来说，对于任意一个时间序列 $\boldsymbol{s}^{(i)}$，除了考虑它自身的过去值外，还会从剩余的 $M-1$ 个时间序列中随机挑选出若干时间序列。如果时间序列 $\boldsymbol{s}^{(i)}$ 受到时间序列 $\boldsymbol{s}^{(j)}$ 的影响，则对应的连接矩阵元素设

置为 β(在我们的例子中设为 1)，而所有其他元素则设为零。通过这种方式，能够有效地模拟一个稀疏但有实际因果关系结构的线性动态系统。在我们的实验中，我们生成了 1000 个时间序列数据点。

我们将三个数据集对应的多变量时间序列和真实 Granger 因果关系矩阵展示在图 2 和图 3：

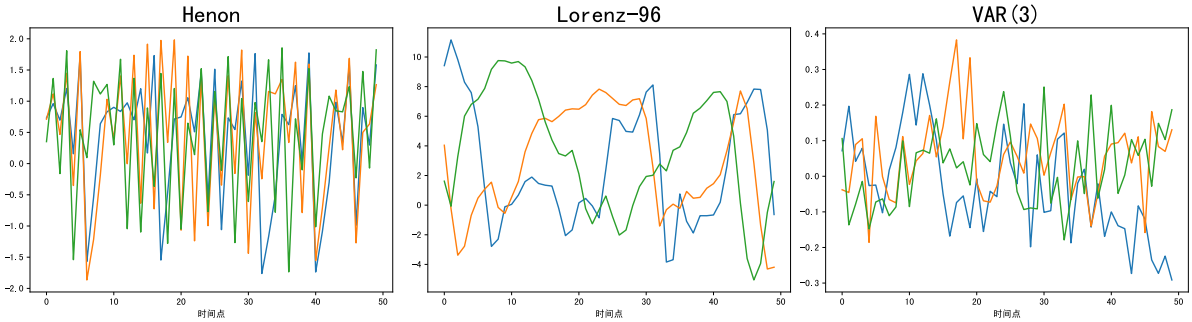

Figure 2. Three datasets: Hénon, Lorenz-96, and VAR (3). For ease of visualization, only the first three sequences and the first fifty time points are displayed

图 2. 三个数据集：Hénon，Lorenz-96，VAR (3)。为了显示方便，仅展示前三个序列和前五十个时间点

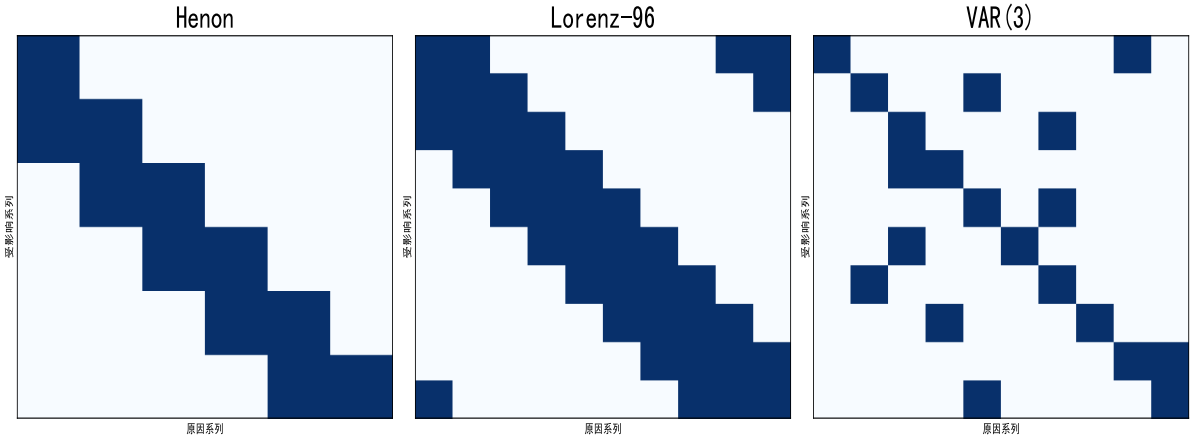

Figure 3. The ground-truth Granger causality matrices for the three datasets are shown, with dark-colored cells indicating a value of 1 and light-colored cells indicating a value of 0 in the matrix

图 3. 三个数据集对应的真实 Granger 因果关系矩阵，其中深色部分代表矩阵中对应值为 1，浅色部分则代表矩阵中对应值为 0

4.2. 模型训练

我们在三个数据集上，依据算法表 1 对模型进行训练，并采用了“早停”策略以便对训练给予终止[32]。我们将模型的训练轨迹(即生成器和判别器的对抗损失)展示在图 4。值得注意的是，我们的模型训练几乎不受模式崩溃和梯度消失的影响，这是因为我们的生成器不仅仅要从分布层面生成符合真实数据分布特性的样本，而且还要从数值层面生成高精度的预测值，这一点不同于经典的生成对抗网络。

4.3. 实验结果

为了评估 GCGAN 模型的有效性，我们将其与两个前沿的基准模型进行了比较：NGC [28]和 CRVAE [15]。其中，为了更显著地对比，我们同样选用了长短期记忆网络作为对比实验中 NGC 的神经网络，即原文里的模型实例 cLSTM。

在实验中，我们采用 AUROC (Area Under the Receiver Operating Characteristic Curve，受试者操作特

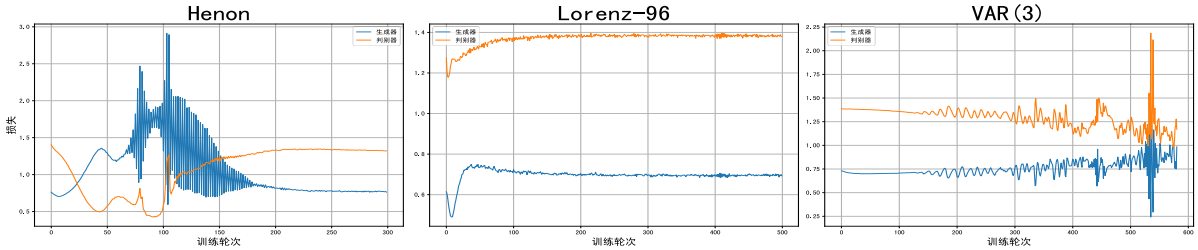

Figure 4. Training trajectories of GCGAN on three datasets
图 4. 三个数据集上 GCGAN 的训练轨迹

征曲线下面积)和准确率(Accuracy)作为时间序列 Granger 因果关系发现模型的评估指标。其中 AUROC 是通过绘制 ROC 曲线(不同阈值下真阳率和假阳率的关系曲线)并计算曲线下面积来衡量模型区分能力的一种方法，而准确率则是判断正确的样本占总样本的比例。

最后，我们将模型的评估指标对比以及模型估计的 Granger 因果关系矩阵对比分别展示在表 3 和图 5，可以看到 GCGAN 在三个数据集上的表现均超越了基准模型，并在相对简单的数据集上达到了完美的精度。

Table 3. Comparison of evaluation metrics for various models on three datasets
表 3. 各模型在三个数据集上的评估指标对比

模型 \ 数据集	Hénon		Lorenz-96		VAR (3)	
	AUROC	准确率	AUROC	准确率	AUROC	准确率
NGC	**1.000**	97%	0.987	97%	0.961	95%
CRVAE	**1.000**	**100%**	0.986	98%	0.872	94%
GCGAN	**1.000**	**100%**	**1.000**	**100%**	**0.997**	**98%**

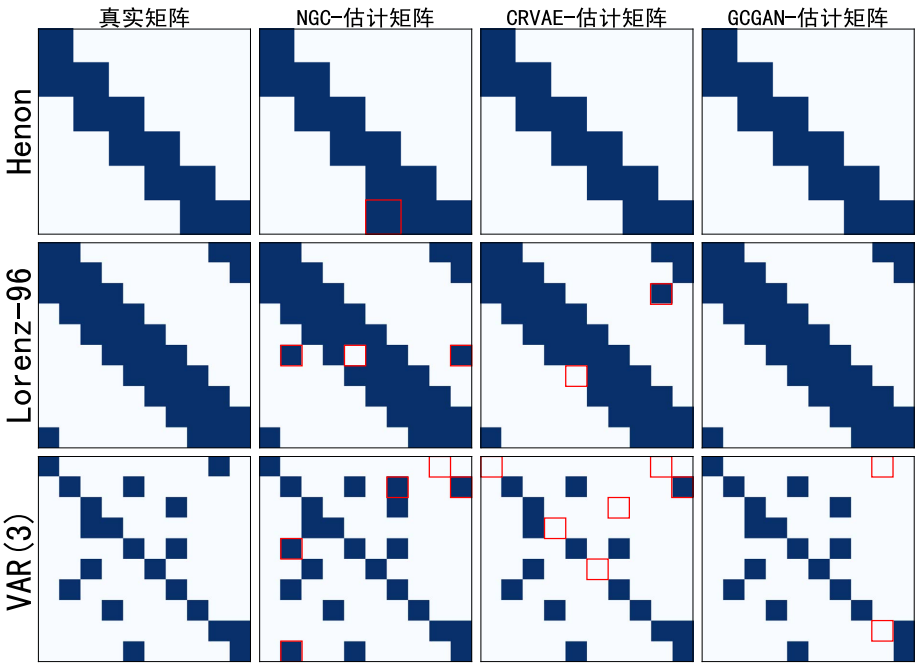

Figure 5. Comparison of the estimated granger causality matrices by different models, with red-marked squares indicating model misjudgments
图 5. 各模型估计的 Granger 因果关系矩阵对比，红色标注的方块代表模型的误判

5. 总结

通过设计多头生成器的架构建模目标序列，在训练时对生成器施加稀疏诱导惩罚项，并在提取因果关系矩阵时引入阈值方法，本文提出的 GCGAN 模型成功将生成对抗网络应用于时间序列的 Granger 因果发现。实验结果表明，与现有的方法相比 GCGAN 模型展现了其卓越的性能。展望未来的工作，我们将继续探索更广泛、更复杂的生成模型在时间序列分析中的潜力，并扩展到更多的实际场景中。

参考文献

[1] Granger, C.W.J. (1969) Investigating Causal Relations by Econometric Models and Cross-Spectral Methods. *Econometrica*, **37**, 424-438. https://doi.org/10.2307/1912791

[2] Shahbaz, M., Lean, H.H. and Shabbir, M.S. (2012) Environmental Kuznets Curve Hypothesis in Pakistan: Cointegration and Granger Causality. *Renewable and Sustainable Energy Reviews*, **16**, 2947-2953. https://doi.org/10.1016/j.rser.2012.02.015

[3] Sharpe, W.F., Alexander, G.J. and Bailey, J. (1995) Investments Prentice Hall. Prentice Hall.

[4] Sheikhattar, A., Miran, S., Liu, J., Fritz, J.B., Shamma, S.A., Kanold, P.O., *et al*. (2018) Extracting Neuronal Functional Network Dynamics via Adaptive Granger Causality Analysis. *Proceedings of the National Academy of Sciences*, **115**, E3869-E3878. https://doi.org/10.1073/pnas.1718154115

[5] Sims, C.A. (1980) Macroeconomics and Reality. *Econometrica*, **48**, 1-48. https://doi.org/10.2307/1912017

[6] Stokes, P.A. and Purdon, P.L. (2017) A Study of Problems Encountered in Granger Causality Analysis from a Neuroscience Perspective. *Proceedings of the National Academy of Sciences*, **114**, E7063-E7072. https://doi.org/10.1073/pnas.1704663114

[7] Hiemstra, C. and Jones, J.D. (1994) Testing for Linear and Nonlinear Granger Causality in the Stock Price-Volume Relation. *The Journal of Finance*, **49**, 1639-1664. https://doi.org/10.1111/j.1540-6261.1994.tb04776.x

[8] Lutkepohl, H. (2005) New Introduction to Multiple Time Series Analysis. Springer Science & Business Media.

[9] Fan, J. and Li, R. (2001) Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties. *Journal of the American Statistical Association*, **96**, 1348-1360. https://doi.org/10.1198/016214501753382273

[10] Tibshirani, R. (1996) Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society Series B*: *Statistical Methodology*, **58**, 267-288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

[11] Hinton, G.E. and Salakhutdinov, R.R. (2006) Reducing the Dimensionality of Data with Neural Networks. *Science*, **313**, 504-507. https://doi.org/10.1126/science.1127647

[12] Graves, A. (2012) Supervised Sequence Labelling. In: *Studies in Computational Intelligence*, Springer, 5-13. https://doi.org/10.1007/978-3-642-24797-2_2

[13] Raissi, M., Perdikaris, P. and Karniadakis, G.E. (2018) Multistep Neural Networks for Data-Driven Discovery of Nonlinear Dynamical Systems.

[14] Nauta, M., Bucur, D. and Seifert, C. (2019) Causal Discovery with Attention-Based Convolutional Neural Networks. *Machine Learning and Knowledge Extraction*, **1**, 312-340. https://doi.org/10.3390/make1010019

[15] Tank, A., Covert, I., Foti, N., Shojaie, A. and Fox, E.B. (2021) Neural Granger Causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44, 4267-4269. https://doi.org/10.1109/tpami.2021.3065601

[16] Li, H., Yu, S. and Principe, J. (2023) Causal Recurrent Variational Autoencoder for Medical Time Series Generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, **37**, 8562-8570. https://doi.org/10.1609/aaai.v37i7.26031

[17] Goodfellow, I., Pouget-Abadie, J., Mirza, M., *et al*. (2014) Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, **27**, 2672-2680.

[18] Geiger, A., Liu, D., Alnegheimish, S., Cuesta-Infante, A. and Veeramachaneni, K. (2020) Tadgan: Time Series Anomaly Detection Using Generative Adversarial Networks. 2020 *IEEE International Conference on Big Data* (*Big Data*), Atlanta, 10-13 December 2020, 33-43. https://doi.org/10.1109/bigdata50022.2020.9378139

[19] Iglesias, G., Talavera, E., González-Prieto, Á., Mozo, A. and Gómez-Canaval, S. (2023) Data Augmentation Techniques in Time Series Domain: A Survey and Taxonomy. *Neural Computing and Applications*, **35**, 10123-10145. https://doi.org/10.1007/s00521-023-08459-3

[20] Koochali, A., Dengel, A. and Ahmed, S. (2021) If You Like It, GAN It—Probabilistic Multivariate Times Series Forecast with Gan. *The 7th International Conference on Time Series and Forecasting*, **5**, Article 40. https://doi.org/10.3390/engproc2021005040

[21] Luo, Y.H., Cai, X.R., Zhang, Y., *et al*. (2018) Multivariate Time Series Imputation with Generative Adversarial Networks. *Advances in Neural Information Processing Systems*, **31**, 1-12.

[22] Bressler, S.L. and Seth, A.K. (2011) Wiener-Granger Causality: A Well-Established Methodology. *NeuroImage*, **58**, 323-329. https://doi.org/10.1016/j.neuroimage.2010.02.059

[23] Wiener, N. and Masani, P. (1957) The Prediction Theory of Multivariate Stochastic Processes: I. The Regularity Condition. *Acta Mathematica*, **98**, 111-150. https://doi.org/10.1007/bf02404472

[24] Marinazzo, D., Pellicoro, M. and Stramaglia, S. (2008) Kernel Method for Nonlinear Granger Causality. *Physical Review Letters*, **100**, Article 144103. https://doi.org/10.1103/physrevlett.100.144103

[25] Chen, Y., Rangarajan, G., Feng, J. and Ding, M. (2004) Analyzing Multiple Nonlinear Time Series with Extended Granger Causality. *Physics Letters A*, **324**, 26-35. https://doi.org/10.1016/j.physleta.2004.02.032

[26] Zhang, W., Tanida, J., Itoh, K. and Ichioka, Y. (1988) Shift-Invariant Pattern Recognition Neural Network and Its Optical Architecture. *Proceedings of Annual Conference of the Japan Society of Applied Physics*, **564**, 2147-2151.

[27] Zhang, K., Zhong, G., Dong, J., Wang, S. and Wang, Y. (2019) Stock Market Prediction Based on Generative Adversarial Network. *Procedia Computer Science*, **147**, 400-406. https://doi.org/10.1016/j.procs.2019.01.256

[28] Beck, A. and Teboulle, M. (2009) A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences*, **2**, 183-202. https://doi.org/10.1137/080716542

[29] Amari, S. (1993) Backpropagation and Stochastic Gradient Descent Method. *Neurocomputing*, **5**, 185-196. https://doi.org/10.1016/0925-2312(93)90006-o

[30] Kugiumtzis, D. (2013) Direct-Coupling Information Measure from Nonuniform Embedding. *Physical Review E*, **87**, Article 062918. https://doi.org/10.1103/physreve.87.062918

[31] Lorenz, E.N. (1996) Predictability: A Problem Partly Solved. *Proceedings of Seminar on Predictability*, **1**, 1-18.

[32] Prechelt, L. (1998) Early Stopping, But When? In: *Lecture Notes in Computer Science*, Springer, 55-69. https://doi.org/10.1007/3-540-49430-8_3