Sequential Causal Effect Variational Autoencoder: Time Series Causal Link Estimation under Hidden Confounding

Violeta Teodora Trifunov
Department for Mathematics and Computer Science, Computer Vision Group
Friedrich Schiller University Jena
Jena, Germany
violetateodora.trifunov@uni-jena.de

Maha Shadaydeh
Department for Mathematics and Computer Science, Computer Vision Group
Friedrich Schiller University Jena
Jena, Germany

Joachim Denzler
Department for Mathematics and Computer Science, Computer Vision Group
Friedrich Schiller University Jena
Jena, Germany

Abstract—Estimating causal effects from observational data in the presence of latent variables sometimes leads to spurious relationships which can be misconceived as causal. This is an important issue in many fields such as finance and climate science. We propose Sequential Causal Effect Variational Autoencoder (SCEV AE), a novel method for time series causality analysis under hidden confounding. It is based on the CEV AE framework and recurrent neural networks. The causal link’s intensity of the confounded variables is calculated by using direct causal criteria based on Pearl’s do-calculus. We show the efficacy of SCEV AE by applying it to synthetic datasets with both linear and nonlinear causal links. Furthermore, we apply our method to real aerosol-cloud-climate observation data. We compare our approach to a time series deconfounding method with and without substitute confounders on the synthetic data. We demonstrate that our method performs better by comparing both methods to the ground truth. In the case of real data, we use the expert knowledge of causal links and show how the use of correct proxy variables aids data reconstruction.

Index Terms—time series, causality analysis, deep latent models

I. INTRODUCTION

Causal reasoning has been developed and continuously refined by many great thinkers striving to understand the causes of significant events of their time. However, despite the recent technological advances, modern causality analysis still faces certain limitations when tackling causal link estimation and inference from observational data in the presence of hidden confounders [1].

In an effort to mitigate this discrepancy, we propose a novel method for estimating causal effects of time series variables under hidden confounding, based on the Causal Effect Variational Autoencoder (CEV AE) [2]. This deep learning framework was first applied to analyzing effects of a binary treatment on the patients’ health outcomes. The causal Directed Acyclic Graph (DAG) representing hidden confounding with one proxy variable, as used by Louizos et al. [2], is depicted in Fig. 1. Y denotes the effect, W the cause, Z denotes the hidden confounder and X a proxy variable providing noisy views on Z. We note that the proxy can be multivariate, and both categorical and continuous. Although the DAG in Fig. [1] might appear restrictive, in many real-world applications, such as climate science for instance, there are unobserved confounders present, with multiple different proxy variables to describe them [3]. We assume that the causal link between the confounded variables is approximately directly identifiable through the proxy to the hidden confounder. We provide more
details on this and the identifiability theory in Section II-B.

In the work by Trifunov et al. [4], the CEVAE methodology was extended from modeling binary treatment variable to a more general continuous intervention setup [5] and applied to time series anomaly detection using a sliding window approach. Deep neural networks within CEVAE architecture allowed the authors to estimate not only linear, but nonlinear causal links as well. Apart from the nonlinear causal links, the problem of causal link estimation under hidden confounding is particularly challenging for time series because the data is non-stationary, and the causal link between confounded variables might be changing over time, thus introducing delayed causal effects.

To address these issues, we extend the CEVAE methodology to time series by using recurrent neural networks (RNNs), i.e. Long Short-Term Memory (LSTMs) [6] instead of the Multi-Layer Perceptrons (MLPs) [7] of the VAE [8] and a TARnet [9], used for parameterizing the probability distributions of observational data and distinguishing between binary treatment values, respectively. Moreover, we allow for the causal effect estimation to be performed on a single time series example of each observed variable. This is done by training the model on a multitude of windows of consecutive time steps instead of on multiple data samples. This means that one does not need to have multiple realizations of the variables on their disposal, or a large amount of patients’ records in the case of treatment effect estimation. This is an advantage since many fields such as finance and climate science only observe each variable once at a given time step. In the case of real data, by varying input intervention and proxy variables, one can determine which variable is causal to the outcome variable, thus tackling the problem of causal discovery using our method. The paper is structured as follows. In Section II we introduce the CEVAE framework and provide definition and discussion of causal identifiability, the assumptions under which our approach is justified, as well as a detailed description of our proposed method. In Section III we outline related work on treatment effect estimation and provide an overview of causality analysis under hidden confounding. Section IV describe the data we use, and V provides a detailed experimental setup along with our method’s the results. Section VI summarizes our approach and concludes the paper.

To the best of our knowledge, we are the first to propose a causal effect estimation method under hidden confounding for sequential data based on CEVAE.

II. METHODOLOGY

A. Causal effect variational autoencoder

Causal effect variational autoencoder (CEVAE) [2] is a deep learning method based on a VAE [8] and a TARnet [9]. Its underlying probabilistic graphical model is shown in Fig. 1. CEVAE methodology assumes all variables to be non-sequential. W denotes binary treatment, Y an outcome of this treatment, while the latent confounder Z, and its proxy X denote the socio-economic status and income of each patient, respectively. The central aim of treatment effect estimation is recovering the Individual Treatment Effect (ITE) and the Average Treatment Effect (ATE) defined in (1) and (2), respectively:

\[
ITE(x) := \mathbb{E}_Y(Y|X=x, do(W = w^1)) - \mathbb{E}_Y(Y|X=x, do(W = w^0))
\]

(1)

\[
ATE := \mathbb{E}_Y(ITE(x))
\]

(2)

These metrics are defined for each value \( x \) of variable \( X \), and by \( w^1 \) we denote applied treatment, while values of \( W \) when no treatment is applied are denoted by \( w^0 \). In the CEVAE framework, \( w^1 = 1 \), and \( w^0 = 0 \). ATE is easily calculated once we obtain the ITE, and for that we need to recover the joint distribution \( p(Z, X, W, Y) \), as stated in Theorem 1 by Louizos et al. [2].

**Theorem 1.** If CEVAE recovers \( p(Z, X, W, Y) \), then we can recover the ITE under the causal model in Fig. 1.

Distribution \( p(Z, X, W, Y) \) is obtained via CEVAE’s model network by approximating the true posterior over \( Z \) conditioned on \( X, W \) and \( Y \), whereas the prior \( p(Z) \) is modeled by the standard normal distribution. All estimated probability distributions are parameterized by MLPs. TARnet is used to infer the estimate of the posterior by branching for each of the two treatment groups in \( W \). Further, once all necessary distributions have been modeled, one can then construct a single objective for the inference and model networks, i.e. the variational lower bound:

\[
\mathcal{L} = \sum_{i=1}^{N} \mathbb{E}_{q(z_i|x_i, w_i, y_i)}(\log p(z_i) - \log q(z_i|x_i, w_i, y_i))
\]

(3)

\[+ \log p(x_i, w_i|z_i) + \log p(y_i|w_i, z_i)\]

of the causal graphical model from Fig. 1. By \( x_i \) we denote an input data point, by \( w_i \) the treatment assignment, by \( y_i \) the outcome of the specific treatment, by \( z_i \) the latent confounder, and by \( q(\cdot) \) we denote estimation of the probability distribution with the same arguments.

Finally, since it is necessary to know the intervention assignment \( W \) together with its outcome \( Y \) before inferring the posterior distribution over \( Z \), two auxiliary distributions are introduced, helping to predict \( w_i \) and \( y_i \) for new samples, so the variational lower bound becomes

\[
\mathcal{F}_{CEVAE} = \mathcal{L} + \sum_{i=1}^{N} \left\{ \log q(w_i = w_i^*|x_i^*) + \log q(y_i = y_i^*|x_i^*, w_i^*) \right\},
\]

(4)

where \( x_i^*, w_i^*, y_i^* \) are the observed values for the input, intervention, and outcome variables in the training set.

B. Causal identifiability

When attempting to estimate causal link intensity or perform causal discovery from observational data, one needs to establish if the underlying model is identifiable. If that is not the case, a set of assumptions under which the identifiability holds must be imposed. We introduce identifiability following recent work...
This means we can estimate the original latent variables up to point-wise transformations $T_i$ of $Z$. In certain cases of non-sequential data, having $A$ as a permutation matrix reduces the problem of indeterminacy of $Z^*$ to finding the point-wise transformations of $Z$. This is due to Eq. (6) then becoming $T_i^*(Z_i^*) = T_i'(Z_i')$ for a permuted index $i$.

Our deep latent variable model represented through parameters $\theta = (f, T_i, \lambda_i)$, as per [10], is

$$p_0(Y, W, Z | X) = p_f(Y, W | Z)p_{T_i, \lambda_i}(Z | X),$$

for a function $f$ parameterized by an LSTM and sufficient statistic $T_i$ with parameters $\lambda_i$, $i \in \{1, \ldots, N\}$. Some of the identifiability assumptions for a VAE as described in [10] are that function $f$ is injective, and that sufficient statistic $T_i$ is differentiable almost everywhere. Since our model is in principle a VAE, we rely on these assumptions and generate the synthetic data accordingly.

### C. Sequential CEVAE

We propose a novel method for time series causal link estimation under hidden confounding. It is based on CEVAE in which all distributions involved in the model and inference networks are modeled by LSTMs [6]. Furthermore, we do not use a TARnet to estimate $p(Y | X, w^0)$ and $p(Y | X, w^1)$, specialized for binary treatment, but rather model $p(Y | X, W)$ sequentially, and introduce branching in the decoder as depicted on the right of Fig. 2. This branching allows us to compare factual and counterfactual causal effects, in order to estimate the cause-effect intensity after intervention on the non-binary sequential cause variable.

We will now introduce the notation for sequential data and explain the proposed method in full detail.

Let $X = \{x_i\}_{i=1}^N$, $W = \{w_i\}_{i=1}^N$ and $Y = \{y_i\}_{i=1}^N$ be the time series of proxy, cause variable, and effect, for $N \in \mathbb{N}$, respectively. The hidden confounder $Z = \{z_i\}_{i=1}^N$ is also time series for each $z_t \in \mathbb{R}^d$, where $d$ is the dimension of $Z$ in the latent space at each time step $t$. The conditional distributions of these variables used by our method’s framework are depicted in Fig. 2. The LSTMs are denoted by $f_i$, and each is parameterized by its own parameters $\phi_i$, for $i \in \{1, 2, 3, 4\}$. For each time step $t$, we model $x_t, w_t, \text{and the prior of } z_t$ as follows:

$$p(z_t) = \mathcal{N}(0, 1)$$

$$p(w_t | z_t) = \mathcal{N}(\mu_{w_t}, \sigma_{w_t}^2), \quad \mu_{w_t}, \sigma_{w_t}^2 = f_1(z_t)$$

$$p(x_t | z_t) = \mathcal{N}(\mu_{x_t}, \sigma_{x_t}^2), \quad \mu_{x_t}, \sigma_{x_t}^2 = f_2(z_t).$$

This means that each variable is full time series, since each individual time step is modeled by a Gaussian distribution. The LSTMs are not bidirectional, hence they do not have access to
We weigh the regularization loss by

\[ \text{We indicate that in contrast to CEV AE, we do not fix variance in} \]

\[ \text{we apply to (SCEV AE, pronounced /see-V AE/).} \]

\[ \text{This approach Sequential Causal Effect Variational Autoencoder} \]

\[ \text{intervention, and outcome variables in the training set.} \]

\[ x \]

\[ \text{for} \]

\[ \text{from Eq. (3)} \]

\[ \text{becomes:} \]

\[ p(y_t \mid w_t, z_t) = \mathcal{N}(\mu_{y_t}, \sigma_{y_t}^2), \quad \mu_{y_t}, \sigma_{y_t}^2 = f_3(w_t, z_t) \]  

(10)

\[ \text{We indicate that in contrast to CEV AE, we do not fix variance in} \]

\[ \text{Eq. (10)} \]

\[ \text{but rather learn it from observational data. Since W is} \]

\[ \text{continuous and a time series, we model this distribution at each} \]

\[ \text{time step, thus avoiding branching our method’s architecture} \]

\[ \text{for different values of} \]

\[ \text{in contrast to CEV AE framework. Our approach is less restrictive, and can therefore be applied} \]

\[ \text{to a wider variety of real-world problems. The intervention} \]

\[ \text{we apply to W is setting it to Gaussian noise, while other} \]

\[ \text{intervention types such as knockoffs [12], which preserve the} \]

\[ \text{original distribution of W, will be tackled in the future work.} \]

\[ \text{According to the DAG in Fig.} \]

\[ \text{the posterior distribution of Z depends on X, Y, and W. We thus approximate it by:} \]

\[ q(z_t \mid x_t, y_t, w_t) = \mathcal{N}(\mu_{z_t}, \sigma_{z_t}^2), \quad \mu_{z_t}, \sigma_{z_t} = f_4(x_t, y_t, w_t). \]  

(11)

\[ \text{For estimating the parameters of the auxiliary distribution of} \]

\[ W \]

\[ Y \]

\[ \text{in Eq. (15), we use the following:} \]

\[ q(w_t \mid x_t) = \mathcal{N}(\mu_{w_t}, \sigma_{w_t}^2), \quad \mu_{w_t}, \sigma_{w_t}^2 = f_5(x_t) \]  

(12)

\[ q(y_t \mid x_t, w_t) = \mathcal{N}(\mu_{y_t}, \sigma_{y_t}^2), \quad \mu_{y_t}, \sigma_{y_t} = f_6(x_t, w_t) \]  

(13)

\[ \text{To make sure that variances are positive, we apply a softplus} \]

\[ \text{activation function softplus}(u) = \ln(1 + e^u), \text{for } u \in \mathbb{R} \text{ to the output of a given LSTM cell used to parameterize the variance.} \]

\[ \text{We weigh the regularization loss by} \lambda \in \mathbb{R} \text{ in order to obtain more stable effect predictions, so the variational lower bound from Eq. (3) becomes:} \]

\[ \hat{\mathcal{L}} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{q(z_t \mid x_t, w_t, y_t)}(\lambda \cdot (\log p(z_t) - \log q(z_t \mid x_t, w_t, y_t)) \]

\[ + \log p(x_t, w_t, z_t) + \log p(y_t \mid w_t, z_t)). \]

\[ \text{The variational lower bound used by our method is then:} \]

\[ \mathcal{F}_{\text{SCAEVAE}} = \hat{\mathcal{L}} + \frac{1}{N} \sum_{i=1}^{N} (\log q(w_t = w_t^* \mid x_t^*) + \log q(y_t = y_t^* \mid x_t^*, w_t^*)) \]

\[ \text{for } x_t^*, w_t^*, y_t^* \text{ being the observed values for the input,} \]

\[ \text{intervention, and outcome variables in the training set.} \]

\[ \text{Our method’s encoder is depicted in Fig. 2 on the left, and} \]

\[ \text{the decoder can be seen in the same figure on the right. We call} \]

\[ \text{this approach Sequential Causal Effect Variational Autoencoder} \]

\[ \text{(SCEVAE, pronounced /see-VAE/).} \]

\[ \text{III. Related Work} \]

\[ \text{Treatment effect estimation in the static setting is a very} \]

\[ \text{active field of research [13, 14, 15]. Under the assumption of} \]

\[ \text{no hidden confounding, methods such as VCNet [16] and its} \]

\[ \text{extensions by state-of-the-art transformer networks [17] were} \]

\[ \text{recently applied to the task of continuous treatment estimation} \]

\[ \text{by Zhang et al. [18] and Melnychuk et al. [19].} \]

\[ \text{In a more realistic scenario when there are unobserved} \]

\[ \text{confounders present, Causal Effect Variational Autoencoder} \]

\[ \text{(CEV AE) [2] was one of the first such deep learning methods.} \]

\[ \text{It relies on the existence of one or more proxies to directly} \]

\[ \text{estimate the latent confounder. Nevertheless, in contrast to} \]

\[ \text{above-mentioned transformer-based treatment estimation meth-} \]

\[ \text{ods, CEV AE focuses on binary treatment. Research endeavors} \]

\[ \text{such as those by Rissanen and Marttinen [20], Im et al. [21],} \]

\[ \text{and Trifunov et al. [4] either extend CEV AE to a uniform} \]

\[ \text{or continuous treatment, or analyse this deep latent variable} \]

\[ \text{model’s capabilities. However, until now, to the best of our} \]

\[ \text{knowledge, CEV AE has never been applied to time series.} \]

\[ \text{Taking a different approach towards treatment effect esti-} \]

\[ \text{mation, Bica et al. [22] and Hatt and Feuerriegel [23] propose} \]

\[ \text{deconfounding methods for sequential data using neural networks. Specifically, in [22], the authors develop Time} \]

\[ \text{Series Deconfounder (TSdeconf), a method built upon RNNs} \]

\[ \text{with multitask output to produce a factor model over time and} \]

\[ \text{estimate latent variables. These latent variable estimates are} \]

\[ \text{then used for causal inference as proxies. The limitation of this} \]

\[ \text{approach, in contrast to ours, is that it requires many patients} \]

\[ \text{i.e. samples and is not suitable for processing long time series} \]

\[ \text{(e.g. } N \geq 1000). \text{Another shortcoming of TSdeconf is that} \]

\[ \text{it cannot be applied to observational data with one or more} \]

\[ \text{treatments assigned at each time step. This was addressed} \]

\[ \text{in [23], where the authors introduced a similar time series} \]

\[ \text{deconfounding method called Sequential Deconfounder, this} \]

\[ \text{time based on a Gaussian process latent variable model.} \]

\[ \text{These methods are useful when there are no proxies} \]

\[ \text{available. In contrast to our method, this may introduce more} \]

\[ \text{approximation error and requires multiple realizations of each} \]

\[ \text{variable i.e. many independent samples for training. Although} \]

\[ \text{we rely on access to proxies, it is not a limitation in many} \]

\[ \text{fields such as environmental and climate science, where many} \]

\[ \text{observed variables can be used to describe a latent one [3].} \]

\[ \text{Similarly to our method, work by Yin and Barucca [24] relies} \]

\[ \text{on RNNs to estimate the causal link of the variables under} \]

\[ \text{influence of a hidden confounder. However, it does not employ} \]

\[ \text{do-calculus [5], but rather Granger causality [25] to determine} \]

\[ \text{the presence of a non-spurious causal link. Due to this choice} \]

\[ \text{of causality analysis tools, it cannot detect instantaneous causal} \]

\[ \text{links.} \]

\[ \text{IV. Data} \]

\[ \text{Ground truth is usually not available when conducting causal} \]

\[ \text{effect estimation, so it is an important standard practice to} \]

\[ \text{validate any new method for that purpose on synthetic data.} \]
The hidden confounder and the cause variable are initialized approximately. Cloud and aerosol observation data are obtained twice daily at spectroradiometer’s measurements as intervention of the cause variable, and its effect at time step $t$, respectively.

$$z_t = a \cdot z_{t-1} + \epsilon_1$$
$$x_t = b_1 \cdot \tanh(z_{t-\tau}) + s \cdot \epsilon_2$$
$$w_t = c_1 \cdot w_{t-1} + c_2 \cdot z_t + \epsilon_3$$
$$\tilde{w}_t \sim N(0, \sigma^2)$$
$$y_t = d_1 \cdot y_{t-1} + d_2 \cdot z_t + g \cdot e^{h \cdot \tilde{w}_t} + \epsilon_4$$

For determining how our method performs for estimating linear causal effects, we alter Eq. (20) as follows:

$$y_t = d_1 \cdot y_{t-1} + d_2 \cdot z_t + g \cdot w_t + \epsilon_4.$$  

In Eqs. (16)-(21), $a, c_1, d_1, g, h, s, \sigma^2, \sigma_2^2 \in \mathbb{R}$, $i = 1, 2, \tau \in \mathbb{N}$, and noise terms $\epsilon_j \sim N(0, \sigma_j^2)$, for $j = 1, 2, 3, 4$, are mutually independent. The time-varying proxy parameter $b_1$ is generated by setting $b_1 = \frac{\epsilon_1}{\tau+1}$ for $t = 1, \ldots, \tau$ and symmetrically setting $b_{\tau+1} := b_{\tau-1}$ for $t = \tau+2, \ldots, N$.

The counterfactual outcome $\tilde{y}_t$ for data with nonlinear and linear causal link is obtained by substituting $w_t$ by $\tilde{w}_t$ in Eq. (20) and Eq. (21), respectively. The parameters used in our experiments are $a = 0.88, s = 0.5, \tau = 100, c_1 = 0.6, c_2 = 0.2, d_1 = 0.4, d_2 = 0.8, g = 0.5, h = 0.5, \sigma^2 = 0.6, \sigma_2^2 = 1$, and $\sigma_3^2 = 0.5, \sigma_4^2 = 0.8, \sigma_5^2 = 0.7$, and $\sigma_6^2 = 0.5$. The hidden confounder and the cause variable are initialized with $z_0 = 0.1$ and $w_0 = 0.1$, whereas $y_0 = 0$.

### B. Cloud and aerosol observation data

We demonstrate our method’s applicability in real-world scenarios by applying it to cloud and aerosol observations dataset provided by Jesson et al. (26). It consists of the $1^\circ \times 1^\circ$ gridded version of the Moderate Resolution Imaging Spectroradiometer’s measurements obtained twice daily at approximately 1 km $\times$ 1 km resolution as the Aqua satellite revolved around the Earth from 2004 until 2019. The region where the data stems from is located in the Pacific Basin off the coast of South America. As per (27), we use the daily average of all variables to make the observations more homogeneous. Moreover, we use cloud optical depth (COD) as the outcome variable $Y$ and aerosol optical depth (AOD) as intervention variable $W$. In accordance with (27), we use meteorological variables sea surface temperature (SST), estimated inversion strength (EIS), lower tropospheric stability (LTS), vertical motion at 500mb (w500), relative humidity at 700mb, 850mb, and 900mb (RH700, RH850, RH900) as proxies. All the variables are normalized before using them for training SCEVAE.

We note that one can vary the outcome and the intervention variables as well as the proxies to find the variable configuration with the highest causal link intensity towards tackling the problem of causal discovery using our method.

### V. Experiments

Here we expound our experimental setup. We generate 1000-time-steps long multivariate time series and use the last 10% of each variable as test data without shuffling in order to preserve inter-temporal dependencies. The last 10% of training data
is used for validation. We normalize the data by subtracting the mean of each variable and then dividing it by its standard deviation. We use batch size of 100 and randomly select that many consecutive time steps as batches of the training data. The optimizer used is Adam [28] with learning rate of $10^{-5}$, and a weight decay of $10^{-3}$. We model the hidden confounder as five-dimensional in the latent space in all but one experiment where we explicitly investigate the influence of its dimensionality to causal link estimation. The LSTMs of SCEVAE consist of two LSTM cells i.e. layers. Since the LSTMs that we use are not bidirectional, they can only look at the past time steps of a given time series. The hidden states and cell states of each LSTM have 32 dimensions. We set the regularization parameter $\lambda$ from Eq. (14) to 0.1.

### A. Results

1) Synthetic data experiments: To demonstrate the efficacy of our proposed method by comparing its results to the ground truth causal link intensity, we apply it to synthetic data with either linear or nonlinear causal link between $W$ and $Y$. Our method’s reconstruction of the effect variable $\hat{Y}$ is shown in Fig. 3 on the left, whereas the reconstruction of the counterfactual effect variable $\hat{Y}$ is depicted in Fig. 3 on the right. We qualitatively observe that the reconstructions (orange) are quite close to the corresponding observed values (blue). These variables are generated according to Eq. (20) using $W$ and $\hat{W}$, respectively. The quantitative results of SCEVAE’s causal analysis are shown in Table I. The causality scores used are ITE with a slight abuse of notation, denoting the difference between the ground truth and the predicted ITE as per Eq. (1), factual Root Mean Squared Error (RMSE), and counterfactual RMSE. The latter two scores measure the discrepancy between the observed $Y$ and the estimated $E(Y|W, Z)$ at each time step $t$, and the discrepancy between $\hat{Y}$ and $E(Y|\hat{W}, Z)$ at each time step $t$, respectively. In the linear case, as per Table I the estimated causal link compared to the ground truth is $\text{ITE}= 0.52$ when proxy parameter $b$ is time-varying and $\text{ITE}= 0.57$ for a fixed value of parameter $b = 0.95$. In the nonlinear case, for a time-varying parameter $b$ $\text{ITE}= 0.56$, and for a fixed $b = 0.95$ we obtain $\text{ITE}= 0.59$. This means that our method can estimate the ground truth causal link intensity between $Y$ and $W$ similarly well in the nonlinear case as in the linear case but it performs better when the proxy parameter $b$ is time-varying.

We compare our results on synthetic data to those of Time Series Deconfounder (TSdeconf) [22] either without taking hidden confounding into account or when the substitutes for the hidden confounder are generated and used as proxies. We note that TSdeconf only outputs RMSE between the ground truth and the predicted outcome, so this is the main comparison metric. Furthermore, since TSdeconf is not suitable for long time series, we generate 100 100-time-steps-long training samples for fairness. This is due to the fact that SCEVAE is trained on 100 100-long epochs randomly chosen from our 1000-long sequential variables.

We note that in the case of the linear causal link, TSdeconf performs almost as well as our method but with much higher variance, making our method much more stable. To obtain more stable results using TSdeconf, one would need to use much more training data. In the case of the nonlinear causal link, our method’s superior performance becomes clearer for both time-varying and fixed proxy parameter $b = 0.95$.

The counterfactual RMSE metric shows that SCEVAE similarly estimates the counterfactual effect variable $\hat{Y}$ for both linear and nonlinear causal links. It is higher than the factual RMSE since $\hat{Y}$ is sampled from the learned distribution of $Y$ using $\hat{W}$ and was not learned individually.

The results of causal link estimation using SCEVAE on synthetic data with nonlinear causal link are depicted in Fig. 4. We see that our method converges to the ground truth ATE both during training and testing phases. The ground truth ATE values for both training and testing are shown in green and red dashed lines, respectively. Furthermore, it is worth to mention that the difference between the ground truth ATE during training and test stems from the non-stationarity of the observational data and the fact that we choose last 10% of the original time series for testing.

In Fig. 6 in Appendix we show how dimensionality of $Z$ in the latent space and the confounding coefficient $d_2$ influence the factual and counterfactual outcome’s reconstruction. We
We validated our method’s results on synthetic data with both with and without substitutes of the hidden confounder. Moreover, through when comparing RMSE of the outcome predictions with and we introduced SCEV AE, a novel deep learning method for time series deconfounding linear and nonlinear causal links. Moreover, we compared our analysis instead of using many independent samples for training. Single-variable causality in the decoder to compare factual to counterfactual outcome variables. Our method allows for single-variable causality modeled by LSTMs but significantly differs in the sense of architecture branching. We do not rely on a TARnet since our method to the benchmark method on time series deconfounding on a CEV AE framework when all probability distributions are tested latent dimensions $D_Z \in \{1, 5, 10, 20\}$, and coefficients $d_2 \in \{0.8, 1, 1.2, 1.6\}$. We observe that RMSE values in both factual and counterfactual cases are lowest when $d_2 = 0.8$ and tend to increase as the rate of hidden confounding $d_2$ increases regardless of $D_Z$. As the dimensionality of $Z$ increases, the uncertainty of the prediction becomes higher.

The factual reconstruction RMSE during training and test can be found in Appendix Fig. 5

2) Cloud and aerosol data experiments: In the experiments on real aerosol-cloud-climate observations, we demonstrate the importance of choosing good proxy variables. Since now we do not have the ground truth causal link intensity values, we cannot use the ITE error. Instead, we use ATE of the predicted factual and counterfactual outcome variables $Y$ and $\hat{Y}$. In both experiments from Table II we use COD as outcome and AOD as intervention variable. In the first experiment we choose meteorological proxies as per [27], while in the other (marked by *) we set the proxy variable $X$ to uniform $\mathcal{U}(0, 1)$ noise. We note that using meteorological proxies yields lower RMSE and drastically lower standard error of all metrics in contrast to the case where we used uniform noise as the proxy. This demonstrates the importance of choosing the right proxy variables that possibly contain information about the hidden confounder.

TABLE II: Causal effect estimation metrics for cloud-aerosol dataset when using COD as the outcome variable and AOD as intervention variable. In the unmarked row we used meteorological proxies EIS, SST, w500, RH700, RH850, and RH900. In the row marked by * we set proxy $X$ to uniform $\mathcal{U}(0, 1)$ noise. The values are averaged over five replications and shown with standard error.

| Outcome | ATE train | ATE test | RMSE |
|---------|-----------|----------|------|
| COD     | 0.06 ± 0.05 | 0.09 ± 0.08 | 0.72 ± 0.001 |
| *COD    | 0.21 ± 0.18 | 0.24 ± 0.14 | 0.79 ± 0.14 |

Causal link intensity estimation is a challenging task, especially in the presence of latent confounders. In this paper we introduced SCEVAE, a novel deep learning method for time series causality analysis under hidden confounding. It is based on a CEVAE framework when all probability distributions are modeled by LSTMs but significantly differs in the sense of architecture branching. We do not rely on a TARnet since our intervention variable is sequential and only introduce branching in the decoder to compare factual to counterfactual outcome predictions. Our method allows for single-variable causality analysis instead of using many independent samples for training. We validated our method’s results on synthetic data with both linear and nonlinear causal links. Moreover, we compared our method to the benchmark method on time series deconfounding and obtained better and more stable results on synthetic data when comparing RMSE of the outcome predictions with and without substitutes of the hidden confounder. Moreover, through the experiments on real cloud-aerosol observational data, we indicated our method’s applicability to real-world problems and illustrated how the use of the proxy contributes to causal identifiability of our method.

ACKNOWLEDGMENT

Funding for this study was provided by the European Research Council (ERC) Synergy Grant “Understanding and Modelling the Earth System with Machine Learning (USMILE)” under the Horizon 2020 research and innovation programme (Grant agreement No. 855187) and by the Carl Zeiss Foundation within the scope of the program line “Breakthroughs: Exploring Intelligent Systems” for “Digitization — explore the basics, use applications”.

APPENDIX

![Fig. 5: RMSE during training (blue) and test (orange) for the synthetic data with nonlinear causal link.](image)

REFERENCES

[1] J. Runge, S. Bathiany, E. Boltt, et al., “Inferring causation from time series of earth system sciences,” Nature Communications, vol. 10, p. 2553, 2019.
[2] C. Louizos, U. Shalit, J. Mooij, D. Sontag, R. Zemel, and M. Welling, “Causal effect inference with deep latent-variable models,” in Advances in Neural Information Processing Systems 30, pp. 6446–6456, Curran Associates Inc., 2017.
[3] S. Samarasinghe, E. A. Barnes, and I. Ebert-Uphoff, “Causal discovery in the presence of confounding latent variables for climate science,” 2018.
[4] V. T. Trifunov, M. Shadaydeh, J. Runge, V. Eyring, M. Reichstein, and J. Denzler, “Nonlinear causal link estimation under hidden confounding with an application to time-series anomaly detection,” in German Conference on Pattern Recognition (GCPR), pp. 261–273, Springer-Verlag, 2019.
[5] J. Pearl, Causality: Models, Reasoning, and Inference. Cambridge University Press, 2nd ed., 2009.
[6] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[7] I. J. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2019.
Fig. 6: Influence of latent dimension and coefficient of the latent confounder to the outcome forecast. Factual RMSE is shown on the left and counterfactual RMSE on the right with standard deviation after five replications.

[8] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in Proceedings of the 2nd International Conference on Learning Representations (ICLR), 2014.

[9] U. Shalit, F. Johansson, and D. Sontag, “Estimating individual treatment effect: Generalization bounds and algorithms,” in Proceedings of the 34th International Conference on Machine Learning, vol. 70, p. 3076–3085, JMLR.org, 2017.

[10] I. Khemakhem, D. P. Kingma, R. P. Monti, and A. Hyvärinen, “Variational autoencoders and nonlinear ica: A unifying framework,” in AISTATS 2020 - 23rd International Conference on Artificial Intelligence and Statistics, 2020.

[11] R. A. Fisher, “A mathematical Examination of the Methods of determining the Accuracy of Observation by the Mean Error, and by the Mean Square Error,” Monthly Notices of the Royal Astronomical Society, vol. 80, no. 8, pp. 758–770, 1920.

[12] R. F. Barber and E. J. Candès, “Controlling the false discovery rate via knockoffs,” The Annals of Statistics, vol. 43, no. 5, 2015.

[13] A. M. Alaa and M. van der Schaar, “Bayesian inference of individualized treatment effects using multi-task gaussian processes,” in Advances in Neural Information Processing Systems, vol. 30, Curran Associates, Inc., 2017.

[14] K. Jiang and Y. Ning, “Treatment effect estimation with unobserved and heterogeneous confounding variables,” 2022.

[15] Z. Qian, Y. Zhang, I. Bica, A. Wood, and M. van der Schaar, “Systcwin: Treatment effect estimation with longitudinal outcomes,” in Advances in Neural Information Processing Systems, vol. 34, pp. 3178–3190, Curran Associates, Inc., 2021.

[16] L. Nie, M. Ye, Q. Liu, and D. Nicolae, “Vcnet and functional targeted regularization for learning causal effects of continuous treatments,” 2021.

[17] T. Wolf, L. Debut, V. Sanh, et al., “Transformers: State-of-the-art natural language processing,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38–45, Association for Computational Linguistics, 2020.

[18] Y. Zhang, H. Zhang, Z. C. Lipton, L. E. Li, and E. P. Xing, “Exploring transformer backbones for heterogeneous treatment effect estimation,” CoRR, vol. abs/2202.01336, 2022.

[19] V. Melnychuk, D. Frauen, and S. Feuerriegel, “Causal transformer for estimating counterfactual outcomes,” 2022.

[20] S. Rissanen and P. Marttinen, “A critical look at the consistency of causal estimation with deep latent variable models,” in Advances in Neural Information Processing Systems, vol. 34, pp. 4207–4217, Curran Associates, Inc., 2021.

[21] D. J. Im, K. Cho, and N. Razavian, “Causal effect variational autoencoder with uniform treatment,” CoRR, vol. abs/2111.08656, 2021.

[22] I. Bica, A. M. Alaa, and M. van der Schaar, “Time series deconfounder: Estimating treatment effects over time in the presence of hidden confounders,” 2019.

[23] T. Hatt and S. Feuerriegel, “Sequential deconfounding for causal inference with unobserved confounders,” vol. abs/2104.09323, 2021.

[24] Z. Yin and P. Barucca, “Deep recurrent modelling of granger causality with latent confounding,” Expert Systems with Applications, vol. 207, p. 118036, 2022.

[25] C. W. Granger, “Investigating causal relations by econometric models and cross-spectral methods,” Econometrica, vol. 37, no. 3, pp. 424–438, 1969.

[26] A. Jesson, P. Manshausen, A. Douglas, et al., “Using non-linear causal models to study aerosol-cloud interactions in the southeast pacific,” 2021.

[27] A. Jesson, A. Douglas, P. Manshausen, et al., “Scalable sensitivity and uncertainty analyses for causal-effect estimates of continuous-valued interventions,” arXiv preprint arXiv:2204.10022v3, 2022.

[28] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2015.