Appendix
Time Series Deconfounder: Estimating Treatment Effects over Time in the Presence of Hidden Confounders

Ioana Bica [1 2] **Ahmed M. Alaa** [3] **Mihaela van der Schaar** [2 3 4]

A. Proof for Theorem 1

Before proving Theorem 1, we introduce several definitions and lemmas that will aid with the proof. Note that the these are extended from the static setting in Wang & Blei (2019a). Remember that at each timestep t, the random variable $\mathbf{Z}_t \in \mathcal{Z}_t$ is constructed as a function of the history until timestep t: $\mathbf{Z}_t = g(\bar{\mathbf{H}}_{t-1})$, where $\bar{\mathbf{H}}_{t-1} = (\bar{\mathbf{Z}}_{t-1}, \bar{\mathbf{X}}_{t-1}, \bar{\mathbf{A}}_{t-1})$ takes values in $\bar{\mathcal{H}}_{t-1} = \bar{\mathcal{Z}}_{t-1} \times \bar{\mathcal{X}}_{t-1} \times \bar{\mathcal{A}}_{t-1}$ and $g : \bar{\mathcal{H}}_{t-1} \to \mathcal{Z}$. In order to obtain **sequential ignorable treatment assignment** using the substitutes for the hidden confounders $\mathbf{Z}_t$, the following property needs to hold:

$$\mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \perp\!\!\!\perp (A_{t1}, \ldots, A_{tk}) \mid \bar{\mathbf{X}}_t, \bar{\mathbf{A}}_{t-1}, \bar{\mathbf{Z}}_t, \tag{1}$$

$\forall \bar{\mathbf{a}}_{\geq t}$ and $\forall t \in \{0, \ldots, T\}$.

Definition 1. ***Sequential Kallenberg construction***
At timestep t, we say that the distribution of assigned causes $(A_{t1}, \ldots A_{tk})$ admits a sequential Kallenberg construction from the random variables $\mathbf{Z}_t = g(\bar{\mathbf{H}}_{t-1})$ and $\mathbf{X}_t$ if there exist measurable functions $f_{tj} : \mathcal{Z}_t \times \mathcal{X}_t \times [0,1] \to \mathcal{A}_j$ and random variables $U_{jt} \in [0,1]$, with $j = 1, \ldots, k$ such that:

$$A_{tj} = f_{tj}(\mathbf{Z}_t, \mathbf{X}_t, U_{tj}), \tag{2}$$

where U_{tj} marginally follow Uniform$[0,1]$ and jointly satisfy:

$$(U_{t1}, \ldots U_{tk}) \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{Z}_t, \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}, \tag{3}$$

for all $\bar{\mathbf{a}}_{\geq t}$.

Lemma 1. ***Sequential Kallenberg construction at each timestep $t \Rightarrow$ Sequential strong ignorability****. If at every timestep t, the distribution of assigned causes $(A_{t1}, \ldots A_{tk})$ admits a Kallenberg construction from $\mathbf{Z}_t = g(\bar{\mathbf{H}}_{t-1})$ and $\mathbf{X}_t$ then we obtain sequential strong ignorability.*

Proof. Assume that $\mathcal{A}_j$ for $j = 1, \ldots, k$ are Borel spaces. For any $t \in \{1, \ldots, T\}$ assume $\mathcal{Z}_t$ and $\mathcal{X}_t$ are measurable spaces and assume that $A_{tj} = f_{tj}(\mathbf{Z}_t, \mathbf{X}_t, U_{tj})$, where f_{tj} are measurable and

$$(U_{t1}, \ldots U_{tk}) \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{Z}_t, \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}, \tag{4}$$

for all $\bar{\mathbf{a}}_{\geq t}$. This implies that:

$$(\mathbf{Z}_t, \mathbf{X}_t, U_{t1}, \ldots U_{tk}) \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{Z}_t, \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}. \tag{5}$$

Since the A_{tj}'s are measurable functions of $(\mathbf{Z}_t, \mathbf{X}_t, U_{t1}, \ldots U_{tk})$ and $\bar{\mathbf{H}}_{t-1} = (\bar{\mathbf{Z}}_{t-1}, \bar{\mathbf{X}}_{t-1}, \bar{\mathbf{A}}_{t-1})$, we have that sequential strong ignorability holds:

$$(A_{t1}, \ldots A_{tk}) \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \bar{\mathbf{X}}_t, \bar{\mathbf{A}}_{t-1}, \bar{\mathbf{Z}}_t, \tag{6}$$

$\forall \bar{\mathbf{a}}_{\geq t}$ and $\forall t \in \{0, \ldots, T\}$. □

[1]University of Oxford, Oxford, United Kingdom [2]The Alan Turing Institute, London, United Kingdom [3]UCLA, Los Angeles, USA [4]University of Cambridge, Cambridge, United Kingdom. Correspondence to: Ioana Bica <ioana.bica@eng.ox.ac.uk>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

Lemma 2. ***Factor models for the assigned causes*** $\Rightarrow$ ***Sequential Kallenberg construction at each timestep*** t. *Under weak regularity conditions, if the distribution of assigned causes* $p(\bar{\mathbf{a}}_T)$ *can be written as the factor model* $p(\theta_{1:k}, \bar{\mathbf{x}}_T, \bar{\mathbf{z}}_T, \bar{\mathbf{a}}_T)$ *then we obtain a sequential Kallenberg construction for each timestep.*

Regularity condition: The domains of the causes $\mathcal{A}_j$ *for* $j = 1, \ldots, k$ *are Borel subsets of compact intervals. Without loss of generality, assume* $\mathcal{A}_j = [0, 1]$ *for* $j = 1, \ldots, k$.

The proof for Lemma 2 uses Lemma 2.22 in Kallenberg (2006) (kernels and randomization): Let μ be a probability kernel from a measurable space S to a Borel space T. Then there exists some measurable function $f : S \times [0, 1] \to T$ such that if ϑ is $U(0, 1)$, then $f(s, \vartheta)$ has distribution $\mu(s, \dot{})$ for every $s \in S$.

Proof. For timestep t, consider the random variables $A_{t1} \in \mathcal{A}_1, \ldots A_{tk} \in \mathcal{A}_k, \mathbf{X}_t \in \mathcal{X}_t, \mathbf{Z}_t = g(\bar{\mathbf{H}}_{t-1}) \in \mathcal{Z}_t$ and $\theta_j \in \Theta$. Assume sequential single strong ignorability holds. Without loss of generality, assume $\mathcal{A}_j = [0, 1]$ for $j = 1, \ldots, k$.

From Lemma 2.22 in Kallenberg (1997), there exists some measurable function $f_{tj} : \mathcal{Z}_t \times \mathcal{X}_t \times [0, 1] \to [0, 1]$ such that $U_{tj} \sim$ Uniform$[0, 1]$ and:

$$A_{tj} = f_{tj}(\mathbf{Z}_t, \mathbf{X}_t, U_{tj}) \tag{7}$$

and there exists some measurable function $h_{tj} : \Theta \times [0, 1] \to [0, 1]$ such that:

$$U_{tj} = h_{tj}(\theta_j, \omega_{tj}), \tag{8}$$

where $\omega_{tj} \sim$ Uniform$[0, 1]$ and $j = 1, \ldots, k$.

From our definition of the factor model we have that ω_{tj} for $j = 1, \ldots, k$ are jointly independent. Otherwise, $A_{tj} = f_{tj}(\mathbf{Z}_t, \mathbf{X}_t, h_{tj}(\theta_j, \omega_{tj}))$ would not have been conditionally independent given $\mathbf{Z}_t, \mathbf{X}_t$.

Since sequential single strong ignorability holds at each timestep t, we have that $A_{tj} \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}$ $\forall \bar{\mathbf{a}} \in \bar{\mathcal{A}}$, $\forall t \in \{0, \ldots, T\}$ and for $j = 1, \ldots, k$ which implies:

$$\omega_{tj} \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}, \tag{9}$$

$\forall \bar{\mathbf{a}}_{\geq t}$ and $\forall j \in \{1, \ldots, k\}$. Using this, we can write:

$$\begin{aligned}
p(\mathbf{Y}(\bar{\mathbf{a}}_{\geq t}), \omega_{t1}, \ldots, \omega_{tk} \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}) &= p(\mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}) \cdot p(\omega_{t1}, \ldots, \omega_{tk} \mid \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}), \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}) \\
&= p(\mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}) \cdot \prod_{j=1}^{k} p(\omega_{tj} \mid \omega_{t1}, \ldots, \omega_{t,j-1}, \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}), \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}) \\
&= p(\mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}) \cdot \prod_{j=1}^{k} p(\omega_{tj} \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}) \\
&= p(\mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}) \cdot p(\omega_{t1}, \ldots, \omega_{tk} \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1})
\end{aligned}$$

where the second and third steps follow form equation (9) and the fact that $\omega_{t1}, \ldots, \omega_{tk}$ are jointly independent. This gives us:

$$(\omega_{t1}, \ldots, \omega_{tk}) \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{X}_t, \bar{\mathbf{H}}_{t-1} \tag{10}$$

Moreover, since the latent random variable $\mathbf{Z}_t$ is constructed without knowledge of $\mathbf{Y}(\bar{\mathbf{a}}_{\geq t})$, but rather as a function of the history $\bar{\mathbf{H}}_{t-1}$ we have:

$$(\omega_{t1}, \ldots, \omega_{tk}) \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{Z}_t, \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}. \tag{11}$$

$\theta_{1:k}$ are parameters in the factor model and can be considered point masses, so we also have that:

$$(\theta_1, \ldots, \theta_k) \perp\!\!\!\perp \mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \mid \mathbf{Z}_t, \mathbf{X}_t, \bar{\mathbf{H}}_{t-1}, \tag{12}$$

Since $U_{tj} = (h_{ij}(\theta_j, \omega_{tj}))$ are measurable functions of θ_j and ω_{tj} we have that:

$$(U_{t1}, \ldots, U_{tk}) \perp\!\!\!\perp \mathbf{Y}(\mathbf{a}_{\geq t}) \mid \mathbf{Z}_t, \mathbf{X}_t, \bar{\mathbf{H}}_{t-1} \tag{13}$$

We have thus obtained a sequential Kallenberg construction at timestep t. □

Theorem 1. *If the distribution of the assigned causes $p(\bar{\mathbf{a}}_T)$ can be written as the factor model $p(\theta_{1:k}, \bar{\mathbf{x}}_T, \bar{\mathbf{z}}_T, \bar{\mathbf{a}}_T)$ then we obtain sequential ignorable treatment assignment:*

$$\mathbf{Y}(\bar{\mathbf{a}}_{\geq t}) \perp\!\!\!\perp (A_{t1}, \ldots, A_{tk}) \mid \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t, \bar{\mathbf{A}}_{t-1}, \tag{14}$$

for all $\bar{\mathbf{a}}_{\geq t}$ and for all $t \in \{0, \ldots, T\}$.

Proof. Theorem 1 follows from Lemmas 1 and 2. In particular, using the proposed factor graph, we can obtain a sequential Kallenberg construction at each timestep and then obtain sequential strong ignorability. □

B. Implementation Details for the Factor Model

The factor model described in Section 5 was implemented in Tensorflow (Abadi et al., 2015) and trained on an NVIDIA Tesla K80 GPU. For each synthetic dataset (simulated as described in Section 6.1), we obtained 5000 patients, out of which 4000 were used for training, 500 for validation, and 500 for testing. Using the validation set, we perform hyperparameter optimization using 30 iterations of random search to find the optimal values for the learning rate, minibatch size (M), RNN hidden units, multitask FC hidden units and RNN dropout probability. LSTM (Hochreiter & Schmidhuber, 1997) units are used for the RNN implementation. The search range for each hyperparameter is described in Table 1.

The trajectories for the patients do not necessarily have to be equal. However, to be able to train the factor model, we zero-padded them such that they all had the same length. The patient trajectories were then grouped into minibatches of size M and the factor model was trained using the Adam optimizer (Kingma & Ba, 2014) for 100 epochs.

Table 1. Hyperparameter search range for the proposed factor model implemented using a recurrent neural network with multitask output and variational dropout.

Hyperparameter	Search range
Learning rate	0.01, 0.001, 0.0001
Minibatch size	64, 128, 256
RNN hidden units	32, 64, 128, 256
Multitask FC hidden units	32, 64, 128
RNN dropout probability	0.1, 0.2, 0.3, 0.4, 0.5

Table 2 illustrates the optimal hyperparameters obtained for the factor model under the different amounts of hidden confounding applied (as described by the experiments in Section 6.1). Since the results for assessing the Time Series Deconfounder are averaged across 30 different simulated datasets, we report here the optimal hyperparameters identified through majority voting. We note that when the effect of the hidden confounders on the treatment assignments and the outcome is large, more capacity is needed in the factor model to be able to infer them.

Table 2. Optimal hyperparameters for the factor model when different amounts of hidden confounding are applied in the synthetic dataset. The parameter γ measures the amount of hidden confounding applied.

Hyperparameter	$\gamma = 0$	$\gamma = 0.2$	$\gamma = 0.4$	$\gamma = 0.6$	$\gamma = 0.8$
Learning rate	0.01	0.01	0.01	0.01	0.001
Minibatch size	64	64	64	64	128
RNN hidden units	32	64	64	128	128
Multitask FC hidden units	64	128	64	128	128
RNN dropout probability	0.2	0.2	0.1	0.3	0.3

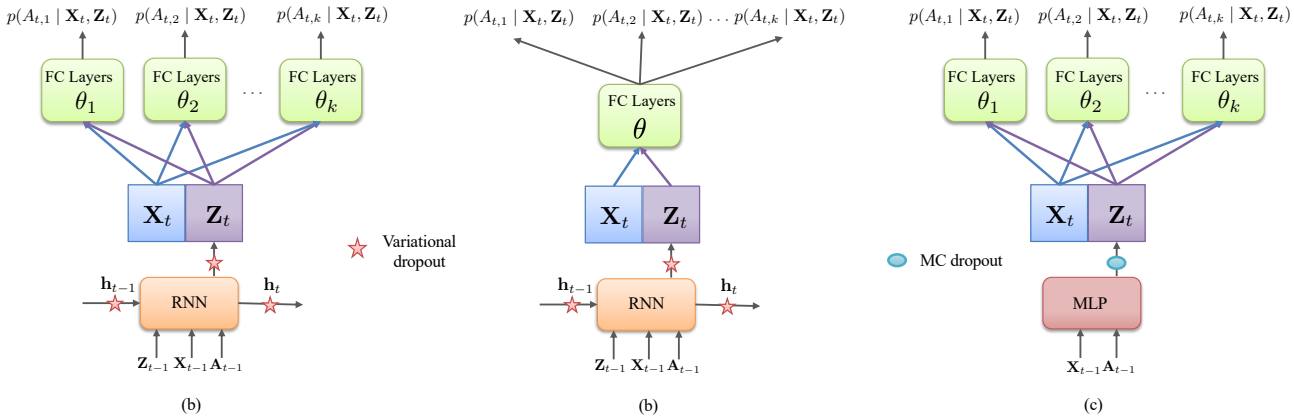

Figure 1. (a) Proposed factor model using a recurrent neural network with multitask output and variational dropout. (b) Alternative design without multitask output. (c) Factor model using an MLP (shared across timestep) and multitask output. This baseline does not capture time-dependencies. MC dropout (Gal & Ghahramani, 2016) is applied in the MLP to be able to sample from the substitutes for the hidden confounders.

C. Baselines for Evaluating Factor Model

Figure 1 illustrates the architecture at each timestep for our proposed factor model and the baselines used for comparison. Figure 1(a) represents our proposed architecture for the factor model consisting of a recurrent neural network with multitask output and variational dropout. We want to ensure that the multitask constraint does not cause a decrease in the capability of the network to capture the distribution of the assigned causes. To do so, we compare our proposed factor model with the network in Figure 1(b) where we predict the k treatment assignments by passing $\mathbf{X}_t$ and $\mathbf{Z}_t$ through a hidden layer and having an output layer with k neurons. Moreover, to highlight the importance of learning time-dependencies to estimate the substitutes for the hidden confounders, we also use as a baseline the factor model in Figure 1(c). In this case, a multilayer perceptron (MLP) is shared across the timesteps and it infers the latent variable Z_t using only the previous covariates and treatments. Note that in this case there is no dependency on the entire history.

The baselines were optimised under the same set-up described for our proposed factor model in Appendix B. Tables 3 and 4 describe the search ranges used for the hyperparameters in each of the baselines.

Table 3. Hyperparameter search range for factor model without multitask (Figure 1(b)).

Hyperparameter	Search range
Learning rate	0.01, 0.001, 0.0001
Minibatch size	64, 128, 256
Max gradient norm	1.0, 2.0, 4.0
RNN hidden units	32, 64, 128, 256
Multitask FC hidden units	32, 64, 128
RNN dropout probability	0.1, 0.2, 0.3, 0.4, 0.5

Table 4. Hyperparameter search range for MLP factor model. Figure 1(c))

Hyperparameter	Search range
Learning rate	0.01, 0.001, 0.0001
Minibatch size	64, 128, 256
MLP hidden layer size	32, 64, 128, 256
Multitask FC hidden units	32, 64, 128
MLP dropout probability	0.1, 0.2, 0.3, 0.4, 0.5

D. Outcome Models

After inferring the substitutes for the hidden confounders using the factor model, we implement outcome models to estimate the individualised treatment responses:

$$\mathbb{E}[\mathbf{Y}_{t+1}(\mathbf{a}_t) \mid \bar{\mathbf{A}}_{t-1}, \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t] = h(\bar{\mathbf{A}}_t, \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t) \tag{15}$$

We train the outcome models and evaluate them on predicting the treatment responses for each timestep, i.e. one-step-ahead predictions, for the patients in the test set. For training and tuning the outcome models, we use the same train/validation/test splits that we have used for the factor model. This means that the substitutes for the hidden confounders estimated using the fitted factor model on the test set are also used for testing purposes in the outcome models.

D.1. Marginal Structural Models

MSMs (Robins et al., 2000; Hernán et al., 2001) have been widely used in epidemiology to perform causal inference in longitudinal data. MSMs use inverse probability of treatment weighting during training to construct a pseudo-population from the observational data that resembles the one in a clinical trial and thus remove the bias introduced by time-dependent confounders (Platt et al., 2009). The propensity scores for each timestep are computed as follows:

$$\text{SW}_t = \frac{f(\mathbf{A}_t \mid \bar{\mathbf{A}}_{t-1})}{f(\mathbf{A}_t \mid \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t, \bar{\mathbf{A}}_{t-1})} = \frac{\prod_{j=1}^{k} f(A_{t,j} \mid \bar{\mathbf{A}}_{t-1})}{\prod_{j=1}^{k} f(A_{t,j} \mid \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t, \bar{\mathbf{A}}_{t-1})} \tag{16}$$

where $f(\cdot)$ is the conditional probability mass function for discrete treatments and the conditional probability density function for continuous treatments. We adopt the implementation in Hernán et al. (2001); Howe et al. (2012) for MSMs and estimate the propensity weights using logistic regression as follows:

$$f(A_{t,k} \mid \bar{\mathbf{A}}_{t-1}) = \sigma\Big(\sum_{j=1}^{k} \omega_k (\sum_{i=1}^{t-1} A_{t,j})\Big) \tag{17}$$

$$f(A_{t,k} \mid \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t, \bar{\mathbf{A}}_{t-1}) = \sigma\Big(\sum_{j=1}^{k} \phi_k (\sum_{i=1}^{t-1} A_{t,j}) + \mathbf{w}_1 \mathbf{X}_t + \mathbf{w}_2 \mathbf{X}_{t-1} + \mathbf{w}_3 \mathbf{Z}_t + \mathbf{w}_4 \mathbf{Z}_{t-1}\Big) \tag{18}$$

where $\omega_\star, \phi_\star$ and $\mathbf{w}_\star$ are regression coefficients and $\sigma(\cdot)$ is the sigmoid function.

For predicting the outcome, the following regression model is used, where each individual patient is weighted by its propensity score:

$$h(\bar{\mathbf{A}}_t, \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t) = \sum_{j=1}^{k} \beta_k (\sum_{i=1}^{t} A_{t,j}) + \mathbf{l}_1 \mathbf{X}_t + \mathbf{l}_2 \mathbf{X}_{t-1} + \mathbf{l}_3 \mathbf{Z}_t + \mathbf{l}_4 \mathbf{Z}_{t-1} \tag{19}$$

where $\beta_\star$ and $\mathbf{l}_\star$ are regression coefficients. Since MSMs do not require hyperparameter tuning, we train them on the patients from both the train and validation sets.

D.2. Recurrent Marginal Structural Networks

R-MSNs, implemented as descried in Lim et al. (2018)[1], use recurrent neural networks to estimate the propensity scores and to build the outcome model. The use of RNNs is more robust to changes in the treatment assignment policy. Moreover, R-MSNs represent the first application of deep learning in predicting time-dependent treatment effects. The propensity weights are estimated using recurrent neural networks as follows:

$$f(A_{t,k} \mid \bar{\mathbf{A}}_{t-1}) = \text{RNN}_1(\bar{\mathbf{A}}_{t-1}) \qquad f(A_{t,k} \mid \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t, \bar{\mathbf{A}}_{t-1}) = \text{RNN}_2(\bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t, \bar{\mathbf{A}}_{t-1}) \tag{20}$$

For predicting the outcome, the following prediction network is used:

$$h(\bar{\mathbf{A}}_t, \bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t) = \text{RNN}_3(\bar{\mathbf{X}}_t, \bar{\mathbf{Z}}_t, \bar{\mathbf{A}}_t), \tag{21}$$

[1]We used the publicly available immlementation from https://github.com/sjblim/rmsn_nips_2018.

where in the loss function, each patient is weighted by its propensity score. Since the purpose of our method is not to improve predictions, but rather to assess how well the R-MSNs can be deconfounded using our method, we use the optimal hyperparameters for this model, as identified by Lim et al. (2018). R-MSNs are then trained on the combined set of patients from the training and validation sets.

Table 5. Hyperparameters used for R-MSN.

Hyperparameter	Propensity networks		Prediction network
	$f(\mathbf{A}_t \mid \bar{\mathbf{A}}_{t-1})$	$f(\mathbf{A}_t \mid \bar{\mathbf{H}}_t)$	
Dropout rate	0.1	0.1	0.1
State size	6	16	16
Minibatch size	128	64	128
Learning rate	0.01	0.01	0.01
Max norm	2.0	1.0	0.5

R-MSNs (Lim et al., 2018), can also be used to forecast treatment responses for an arbitrary number of steps in the future. In our paper we focus on one-step ahead predictions of the treatment responses. However, the Time Series Deconfounder can also be applied to estimate the effects of a sequence of future treatments.

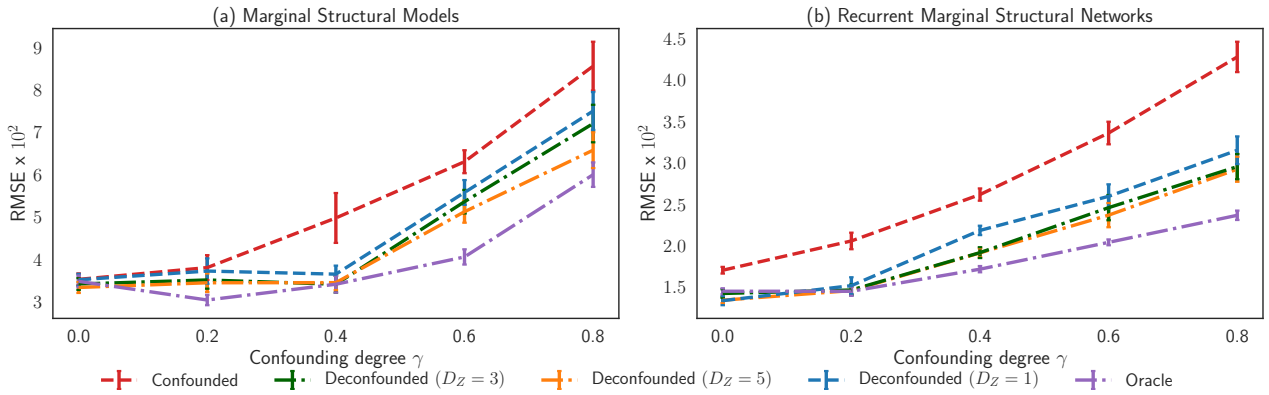

Figure 2. Results for deconfounding one-step ahead estimation of treatment responses in two outcome models: (a) Marginal Structural Models (MSM) and (b) Recurrent Marginal Structural Networks (R-MSN). The simulated (true) size of the hidden confounders is $D_Z = 3$. The average RMSE and the standard error in the results are computed for 30 dataset simulations for each different degree of confounding, as measured by γ.

E. Additional Results

E.1. Experiments on Synthetic Data

We considered an additional experimental set-up where we have simulated hidden confounders of dimension $D_Z = 3$. In Figure 2 we illustrate the root mean squared error (RMSE) for one-step-ahead estimation of treatment responses for patients in the test set without adjusting for the bias from the hidden confounders (Confounded), when using the simulated hidden confounders (Oracle) and after applying the Time Series Deconfounder with different model specifications (Deconfounded). We notice that the Time Series Deconfounder can still account for the bias from hidden confounders when the true size for the hidden confounders is underestimated in the factor model and set to ($D_Z = 1$). The performance is improved when setting D_Z to the true number of hidden confounders or when overestimating the number of hidden confounders.

E.2. Model of Tumour Growth

To show the applicability of our method in a more realistic simulation, we use the pharmacokinetic-pharmacodynamic (PK-PD) model of tumor growth under the effects of chemotherapy and radiotherapy proposed by Geng et al. (2017). The tumor volume after t days since diagnosis is modeled as follows:

$$V(t) = \Big(1 + \underbrace{\rho \log(\frac{K}{V(t-1)})}_{\text{Tumor growth}} - \underbrace{\beta_c C(t)}_{\text{Chemotherapy}} - \underbrace{\big(\alpha_r d(t) + \beta_r d(t)^2\big)}_{\text{Radiotherapy}} + \underbrace{e_t}_{\text{Noise}}\Big) V(t-1) \tag{22}$$

where $K, \rho, \beta_c, \alpha_r, \beta_r, e_t$ are sampled as described in Geng et al. (2017). $C(t)$ is the chemotherapy drug concentration and $d(t)$ is the dose of radiation. Chemotherapy and radiotherapy prescriptions are modeled as Bernoulli random variables that depend on the tumor size. Full details about treatments are in Lim et al. (2018).

Table 6. Average RMSE $\times 10^2$ (normalised by the maximum tumour volume) and the standard error in the results for predicting the effect of chemotherapy and radiotherapy on the tumour volume.

Outcome model	MSM	R-MSN
Confounded	7.29 ± 0.14	5.31 ± 0.16
Deconfounded ($D_Z = 1$)	6.47 ± 0.16	4.76 ± 0.17
Deconfounded ($D_Z = 5$)	6.25 ± 0.14	4.79 ± 0.19
Deconfounded ($D_Z = 10$)	6.31 ± 0.11	4.54 ± 0.17
Oracle	6.92 ± 0.19	5.00 ± 0.15

To account for patient heterogeneity due to genetic features (Bartsch et al., 2007), the prior means for β_c and α_r are adjusted according to three patient subgroups as described in Lim et al. (2018). The patient subgroup $S^{(i)} \in \{1, 2, 3\}$

represents a confounder because it affects the tumor growth and subsequently the treatment assignments. We reproduced the experimental set-up in Lim et al. (2018) and simulated datasets with 10000 patients for training, 1000 for validation, and 1000 for testing. We simulated 30 datasets and averaged the results for testing the MSM and R-MSN outcome models without the information about patient types (confounded), with the true simulated patient types, as well as after applying the Time Series Deconfounder with $D_Z \in \{1, 5, 10\}$.

The results in Table 6 indicate that our method can infer substitutes for static hidden confounders such as patient subgroups which affect the treatment responses over time. By construction, $\bar{\mathbf{Z}}_t$ also captures time dependencies which help with the prediction of outcomes. This is why the performance of the deconfounded models is slightly better than of the oracle model which uses static patient groups.

E.3. MIMIC III

We performed an additional experiment using the dataset extracted from the MIMIC III database where we have removed 3 patient covariates from the dataset (temperature, glucose, hemoglobin). In Table 7 we report the results for estimating the effects of antibiotics, vasopressors, and mechanical ventilator on the patient's white blood cell count when including all variables, after removing these 3 patient covariates (which we notice that further confound the results) and after applying the Time Series Deconfounder with different settings for D_Z.

Table 7. Average RMSE $\times 10^2$ and the standard error in the results for predicting the effect of antibiotics, vasopressors, and mechanical ventilator on white blood cell count. The results are for 10 runs.

	White blood cell	
Outcome model	MSM	R-MSN
All patient covariates	3.90 ± 0.00	2.91 ± 0.05
Removed 3 covariates	4.12 ± 0.00	3.11 ± 0.03
Deconfounded ($D_Z = 1$)	3.98 ± 0.02	3.05 ± 0.05
Deconfounded ($D_Z = 3$)	3.91 ± 0.03	2.87 ± 0.08
Deconfounded ($D_Z = 5$)	3.85 ± 0.04	2.81 ± 0.03

F. Discussion

The Time Series Deconfounder firstly builds a factor model to infer substitutes for the multi-cause hidden confounders. If Assumption 3 holds and the fitted factor model captures well the distribution of the assigned causes, which can be assessed through predictive checks, the substitutes for the hidden confounders help us obtain sequential strong ignorability (Theorem 1). Then, the Time Series Deconfounder uses the inferred substitutes for the hidden confounders in an outcome model that estimates individualized treatment responses. The experimental results show the applicability of the Time Series Deconfounder both in a controlled simulated setting and in a real dataset consisting of electronic health records from patients in the ICU. In these settings, the Time Series Deconfounder was able to remove the bias from hidden confounders when estimating treatment responses conditional on patient history.

In the static causal inference setting, several methods have been proposed to extend the deconfounder algorithm in Wang & Blei (2019a). For instance, Wang & Blei (2019b) augment the theory in the deconfounder algorithm in Wang & Blei (2019a) by extending it to causal graphs and show that by using some of the causes as proxies of the shared confounder in the outcome model one can identify the effects of the other causes. D'Amour (2019) also suggests using proxy variables to obtain non-parametric identification of the mean potential outcomes (Miao et al., 2018). Additionally, Kong et al. (2019) proves that identification of causal effects is possible in the multi-cause setting when the treatments are normally distributed and the outcome is binary and follows a logistic structural equation model.

For the Time Series Deconfounder, similarly to Wang & Blei (2019a), identifiability can be assessed by computing the uncertainty in the outcome model estimates, as described in Section 4.2. When the treatment effects are non-identifiable, the Time Series Deconfounder estimates will have high variance. Thus, future work could explore building upon the results in Wang & Blei (2019b) and D'Amour (2019) and using proxy variables in the outcome model to prove identifiability of causal estimates in the multi-cause time-series setting.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/. Software available from tensorflow.org.

Bartsch, H., Dally, H., Popanda, O., Risch, A., and Schmezer, P. Genetic risk profiles for cancer susceptibility and therapy response. In *Cancer Prevention*, pp. 19–36. Springer, 2007.

D'Amour, A. On multi-cause approaches to causal inference with unobserved counfounding: Two cautionary failure cases and a promising alternative. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 3478–3486, 2019.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059, 2016.

Geng, C., Paganetti, H., and Grassberger, C. Prediction of treatment response for combined chemo-and radiation therapy for non-small cell lung cancer patients using a bio-mathematical model. *Scientific reports*, 7(1):13542, 2017.

Hernán, M. A., Brumback, B., and Robins, J. M. Marginal structural models to estimate the joint causal effect of nonrandomized treatments. *Journal of the American Statistical Association*, 96(454):440–448, 2001.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Howe, C. J., Cole, S. R., Mehta, S. H., and Kirk, G. D. Estimating the effects of multiple time-varying exposures using joint marginal structural models: alcohol consumption, injection drug use, and hiv acquisition. *Epidemiology (Cambridge, Mass.)*, 23(4):574, 2012.

Kallenberg, O. *Foundations of modern probability*. Springer Science & Business Media, 2006.

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

Kong, D., Yang, S., and Wang, L. Multi-cause causal inference with unmeasured confounding and binary outcome. *arXiv preprint arXiv:1907.13323*, 2019.

Lim, B., Alaa, A., and van der Schaar, M. Forecasting treatment responses over time using recurrent marginal structural networks. In *Advances in Neural Information Processing Systems*, pp. 7493–7503, 2018.

Miao, W., Geng, Z., and Tchetgen Tchetgen, E. J. Identifying causal effects with proxy variables of an unmeasured confounder. *Biometrika*, 105(4):987–993, 2018.

Platt, R. W., Schisterman, E. F., and Cole, S. R. Time-modified confounding. *American journal of epidemiology*, 170(6): 687–694, 2009.

Robins, J. M., Hernan, M. A., and Brumback, B. Marginal structural models and causal inference in epidemiology, 2000.

Wang, Y. and Blei, D. M. The blessings of multiple causes. *Journal of the American Statistical Association*, (just-accepted): 1–71, 2019a.

Wang, Y. and Blei, D. M. Multiple causes: A causal graphical view. *arXiv preprint arXiv:1905.12793*, 2019b.