Supplementary Material for Causal Mosaic

Pengzhou (Abel) Wu
The Graduate University for Advanced Studies

Kenji Fukumizu
The Institute of Statistical Mathematics

1 Proofs

Corollary 1

Proof. From A4, and substitute $\mathbf{X}_{\mathrm{p}}^{\mathrm{tr}} = \mathbf{f}(\mathbf{E}_{\mathrm{p}}^{\mathrm{tr}})$, we have $\mathbf{T}(\mathbf{E}_{\mathrm{p}}^{\mathrm{tr}})^{\mathrm{T}} = \mathbf{A}\mathbf{h}(\mathbf{f}(\mathbf{E}_{\mathrm{p}}^{\mathrm{tr}})) + \mathbf{b}$.

From A3, we know each $\mathbf{X}^{te}$'s support is contained in the support of $\mathbf{h}$. Thus, we can replace $\mathbf{E}_p^{tr}$ with $\mathbf{E}^{te}$ and the equality still holds, we get: $\mathbf{T}(\mathbf{E}^{\mathrm{te}}) = \mathbf{A}\mathbf{h}(\mathbf{f}(\mathbf{E}^{\mathrm{te}})) + \mathbf{b} = \mathbf{A}\mathbf{h}(\mathbf{X}^{\mathrm{te}}) + \mathbf{b}$. □

Theorem 2

Proof. Without loss of generality, assume after alignment cause variable for each training pair is input to $\mathbf{h}$ as the first argument. By A2 and Theorem 1, we will successfully learn $\mathbf{h}$ (Algorithm 1, line 1,2).

By A2 and Corollary 1, if the cause variable of $\mathbf{X}^{te}$ is input to **hICA** as the first argument, then its nonlinear ICA is realized (Algorithm 1, line 3,4). Denote the respective input permutation as α_r, then $C_{\alpha_r(1)} \perp\!\!\!\perp C_{\alpha_r(2)}$. While for the other input direction α_{1-r}, by A1, $C_{\alpha_{1-r}(1)} \not\perp\!\!\!\perp C_{\alpha_{1-r}(2)}$

Thus, we have $\texttt{dindep}(\mathbf{C}_{\alpha_r}) > \texttt{dindep}(\mathbf{C}_{\alpha_{1-r}})$, and $\alpha^* = \alpha_r$. □

Theorem 3

Proof. Similarly to the proof of Theorem 2, we know there is one and only one input direction α_r where nonlinear ICA is realized. We have $\mathbf{T}(\mathbf{E}^{\mathrm{te}}) = (C_{\alpha(1)}, C_{\alpha(2)})^T_{\alpha_r}$ where α is the unknown output permutation.

By A1 (which also implies rule (4)), we have $X_c^{te} \perp\!\!\!\perp C_{3-c,\alpha_r}$ where c is the cause index, but $X_i^{te} \not\perp\!\!\!\perp C_{j,\alpha}$ for all other i, j, α. Thus, $(i^*, j^*, \alpha^*) = (c, 3-c, \alpha_r)$ □

Proposition 1

Proof. From Definition 1, we write $\mathbf{X} = \mathbf{f}(\mathbf{E})$ and denote $\mathbf{g} = \mathbf{f}^{-1}$. And we have the relation of Jacobians $\mathbf{J_g} = \mathbf{J_f}^{-1}$, and:

$$\mathbf{J}_{\mathbf{f}}^{-1} = \begin{pmatrix} \frac{df_1}{dE_1} & 0 \\ \frac{\partial f_2}{\partial X_1}\frac{\partial f_1}{\partial E_1} & \frac{\partial f_2}{\partial E_2} \end{pmatrix}^{-1} = \begin{pmatrix} (\frac{df_1}{dE_1})^{-1} & 0 \\ -(\frac{\partial f_2}{\partial E_2})^{-1}\frac{\partial f_2}{\partial X_1}\frac{df_1}{dE_1}(\frac{df_1}{dE_1})^{-1} & (\frac{\partial f_2}{\partial E_2})^{-1} \end{pmatrix}$$

By comparing the 1st row of $\mathbf{J_g}$ and $\mathbf{J_f}^{-1}$, we have $\frac{\partial g_1}{\partial X_2} = 0$ which indicates g_1 is not a function of X_2, and $\frac{dg_1}{dX_1} = (\frac{df_1}{dE_1})^{-1}$ which, by inverse function theorem, implies f_1 is invertible and $g_1 = f_1^{-1}$. □

2 Combining graphical search methods

There are already some studies that successfully combine cause-effect inference methods with graphical search methods; for example, cause-effect inference methods can be directly employed to infer the undirected edges output by search methods (Monti et al., 2019; Zhang and Hyvärinen, 2009), and overlapping datasets can be integrated using bivariate causal discovery to give more precise output class (Dhir and Lee, 2020). Our method can easily be applied in the same way to help multivariate causal discovery under confounding.

3 Invertibility requirement in Definition 1

Our method is still valid if there exists a transformation $\boldsymbol{\tau}(\mathbf{E}) \coloneqq (\tau_1(E_1), \tau_2(E_2))$ such that the transformed SCM satisfies the assumptions of Theorem 1 (e.g., $\mathbf{X} = \mathcal{F}(\boldsymbol{\tau}(\mathbf{E}))$ and $\mathcal{F}$ is invertible). By Theorem 1, the TCL followed by linear ICA can successfully output the sufficient statistics of $\tau_i(E_i)$, which plays the same role as E_i when testing independence. Note that now the mixing function $\mathbf{f} = \mathcal{F} \circ \boldsymbol{\tau}$ can be *non*-invertible. We believe that the existence of such $\boldsymbol{\tau}$ should prevail in practice, and the results on real world benchmark datasets suggest this. We can go a step further to say $\boldsymbol{\tau}(\mathbf{E})$ are *the* exogenous variables,

since, by definition, exogenous variables are unknown and the only requirement is that they are independent of each other.

Another note is that, Definition 1 does *not* mean that the function relating X_1 and X_2 should be invertible. Quite oppositely, take analyzable SCM (1), f_2 is a function from $\mathbf{R}^2$ to $\mathbf{R}$, which is always non-invertible. Moreover, even if $E_2 = e_2$ is given, the deterministic relation $X_2 = f_2^{e_2}(X_1) := f_2(X_1, e_2)$ could still be non-invertible.

4 Nonlinear ICA violates causal faithfulness assumption

Causal Markov and faithfulness assumptions are common in causal discovery literature, and we also require them in our theorem. However, we should note that causal faithfulness assumption is violated for a realized bivariate nonlinear ICA, because $X_1 \not\perp\!\!\!\perp X_2$ and the nonlinear ICA procedure necessarily has one of the following graphical models:

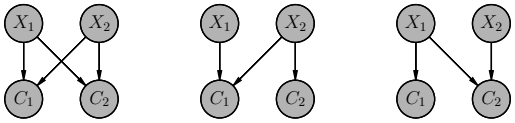

Figure 1: Graphs of nonlinear ICA procedure.

None of them induce $C_1 \perp\!\!\!\perp C_2$ under causal faithfulness assumption.

5 Choice of independence test

HSIC is a widely used independence test in causal discovery literature, but it has several drawbacks. First, its test statistic is not normalized for different testing pairs, and thus not comparable[1]. Second, although p-value of the test is comparable, it does not directly measure the degree of independence. Most importantly, as mentioned in Mooij et al. (2016, sec. 2.2), standard threshold of the test would be too tight for our purpose. This is because in causal discovery we often want to test the independence between an observed variable and an estimation *from* observed data, and there always exists small dependence with finite sample and other real world limitations. For the same reason, the flexibility of HSIC to detect dependence can do harm, not benefit, to causal discovery.

Unlike HSIC[2], dCor value is always in [0, 1], and equals to 0 if and only if the pair under test are independent. Thus, the value 1 - dCor works as a comparable degree of independence. As a bonus, dCor is much faster than HSIC when testing independence between univariate real-valued variables, particularly when sample size is large [3].

Hence, we suggest dCor rather than HSIC as the default choice to measure degree of independence for cause-effect inference, and try HSIC when you can afford the time, both for tuning and running.

6 Caveats on structural MLP

1) While one might think that we need to make MLP1 invertible since g_1 is invertible, we should *not* impose it; the sufficient statistics $\mathbf{T}$ are also learned as part of MLP, and they are in general non-invertible. 2) The structural MLP works only when there is a direct causal effect, as required by SCM (2). 3) Since node i_1 corresponds to the cause, we need to input the cause variable to i_1 for training the asymmetric MLP properly. This requires knowledge on the causal directions of training pairs, and thus, we can only apply it with `inferule1`.

7 Details and notes for artificial experiments

Training and testing data As mentioned, under multi-environment setting, the pairs are for both training and testing. Under multi-pair setting, these same pairs are again used for testing. But for training, we generate another set of pairs with random parameters, while the mixing functions and pair number for each mixing function are the same as testing pairs. For each pair, we always generate 512 data points.

Hyperparameters For the MLP in TCL, we use the same number of layers as data-generating MLP, and each hidden layer has same number of units (4 or 40 in the experiments) with the maxout activation. The two output units have the absolute value function as activation. For the asymmetric MLP (Figure 3, right), we use same width for both sub-MLPs, and keep the sum of the widths the same as fully-connected MLP. Note that the asymmetric MLP has much less parameters than the fully-connected one, since the sub-MLPs are disconnected. We use Momentum optimizer with momentum 0.9 and initial learning rate 0.01, and the batch size is 32.

[1] If we use the default Gaussian kernel and median heuristic for kernel bandwidth (Gretton et al., 2005). And this is also the most common way it is used in bivariate causal discovery (Mooij et al., 2016; Hu et al., 2018)

[2] We noticed that distance covariance is an instance of HSIC for certain choice of kernels (Sejdinovic et al., 2013). But again, this is not default for HSIC.

[3] We use Huo and Székely (2016) for dCor and Zhang et al. (2018) for HSIC, the implementation can be found at https://github.com/vnmabus/dcor and https://github.com/oxmlcs/kerpy, respectively.

MLP width The experimental results show that we need large enough MLP to fit more pairs. Note in particular that the MLP of width 4 performs almost always worse than that of width 40. If we use asymmetric MLP, this tendency is more drastic since it has much less parameters. When the MLP width is 4, the accuracy often decreases w.r.t the number of training pairs. When the MLP width is 40, the accuracy usually increases w.r.t the number of training pairs, but when the pair size is larger than 30, it increases slowly or even slightly drops.

Training pair number We observe better performance as the pair size grows (under the MLP width 40). Under the multi-pair setting, this implies that TCL learns more thoroughly the shared mechanism. Under multi-environment setting, we have one more reason: majority voting performs better with more voters (pairs).

Transferability To confirm the transferability of TCL, we also try inferring directions for individual pairs without voting under multi-environment setting (Figure 4, dashed lines). The results from the two settings are similar, meaning the transferability. The slight drop of performance under multi-pair setting should come from the two input trials needed.

8 Experiments without assuming direct causal effect

We also experiment without assuming direct causal effect necessarily exists, and allow "inconclusive" outputs when the assumption is possibly violated. The purpose here is mainly to conform the problem mentioned in S.3 above, and to show how our method can address it to a large extent. When applying the inference rules, now we need to set a threshold or alpha value for the independence tests. For clearer comparisons, we apply Theorem 3 and also use HSIC, though Theorem 2 or other independence tests can also be applied. Then our method only differs with NonSENS by inferring for each environment and then majority voting.

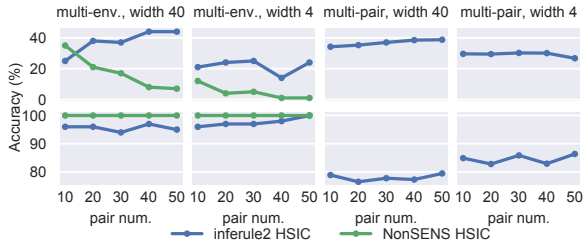

Figure 2: Performance without assuming direct causal effect. 1st/2nd row is results on direct casual data/purely confounded data respectively.

Similarly to Monti et al. (2019), we evaluate on two datasets: 1) all pairs are direct causal (1st row). 2) all pairs are purely confounded (simply use a fully connected MLP) (2nd row). On direct causal pairs, we can see NonSENS' accuracy decreases drastically w.r.t pair number and is nearly always below 10% when MLP width is 4. On the other hand, on purely confounded pairs, it always reports 100% inconclusive.

Here the results conform that the default alpha value (0.05) for independence test is way too tight. Specifically, the problem here is that, with more pairs (which means more sample points for NonSENS), HSIC is more sensitive to small dependence between estimated noise and observed cause. This means we must train TCL very optimally to avoid the unwanted dependence.

Our method performs much better than NonSENS, especially with large pair number. The reason is that, it is easier to get rid of unwanted dependence by looking at each environment, since if any one of the environments shows dependence, then the pooled data tested in NonSENS will be dependent.

9 Alternative ensemble scorings

Without loss of generality, assume X_1 is input to the same node when calculating $w_{ns,1}$, as cause variable is when training. Then we have $Direction_{ns} = \mathbb{I}(w_{ns,1} > w_{ns,2}) - \mathbb{I}(w_{ns,1} < w_{ns,2})$ where indicator function $\mathbb{I}$ maps $true/false$ to $1/0$.

Now the ensemble score in Algorithm 4 line 10 becomes:

$$\begin{aligned} Score_s = &\sum_{n \in TSR_s} w_{ns,1} w_n \mathbb{I}(w_{ns,1} > w_{ns,2}) \\ &- \sum_{n \in TSR_s} w_{ns,2} w_n \mathbb{I}(w_{ns,1} < w_{ns,2}) \end{aligned} \tag{1}$$

But since $\mathbb{I}(w_{ns,1} > w_{ns,2})$ and $\mathbb{I}(w_{ns,1} < w_{ns,2})$ just reflect the relative value of $w_{ns,1}$ and $w_{ns,2}$, the following simplification is reasonable:

$$Score_s = \sum_{n \in TSR_s} w_n (w_{ns,1} - w_{ns,2}) \tag{2}$$

And on the same line of reasoning, we can alternatively disregard $w_{ns,1}, w_{ns,2}$ and have:

$$\begin{aligned} Score_s &= \sum_{n \in TSR_s} w_n \mathbb{I}(w_{ns,1} > w_{ns,2}) \\ &\quad - \sum_{n \in TSR_s} w_n \mathbb{I}(w_{ns,1} < w_{ns,2}) \\ &= \sum_{n \in TSR_s} w_n Direction_{ns} \end{aligned} \tag{3}$$

This is just the weighted average of prediction by each $\mathbf{h}_n$. And finally, since $\mathbf{h}_n$ with small w_n is unlikely to produce large $w_{ns,i}$, we can further disregard w_n in (2). This gives:

$$Score_s = \sum_{n \in TSR_s} (w_{ns,1} - w_{ns,2}) \tag{4}$$

We compared these scoring equations and found the last one is stably the best.

References

Anish Dhir and Ciarán M Lee. Integrating overlapping datasets using bivariate causal discovery. In *Thirty-Fourth AAAI conference on artificial intelligence*, 2020.

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pages 63–77. Springer, 2005.

Shoubo Hu, Zhitang Chen, Vahid Partovi Nia, Laiwan Chan, and Yanhui Geng. Causal inference and mechanism clustering of a mixture of additive noise models. In *Advances in Neural Information Processing Systems*, pages 5206–5216, 2018.

Xiaoming Huo and Gábor J Székely. Fast computing for distance covariance. *Technometrics*, 58(4):435–447, 2016.

Ricardo Pio Monti, Kun Zhang, and Aapo Hyvärinen. Causal discovery with general non-linear relationships using non-linear ICA. In *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019*, page 45, 2019.

Joris M Mooij, Jonas Peters, Dominik Janzing, Jakob Zscheischler, and Bernhard Schölkopf. Distinguishing cause from effect using observational data: methods and benchmarks. *The Journal of Machine Learning Research*, 17(1):1103–1204, 2016.

Dino Sejdinovic, Bharath Sriperumbudur, Arthur Gretton, Kenji Fukumizu, et al. Equivalence of distance-based and rkhs-based statistics in hypothesis testing. *The Annals of Statistics*, 41(5):2263–2291, 2013.

Kun Zhang and Aapo Hyvärinen. On the identifiability of the post-nonlinear causal model. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 647–655. AUAI Press, 2009.

Qinyi Zhang, Sarah Filippi, Arthur Gretton, and Dino Sejdinovic. Large-scale kernel methods for independence testing. *Statistics and Computing*, 28(1):113–130, 2018.