Causal Discovery Under a Confounder Blanket: Appendix

David S. Watson[1] **Ricardo Silva**[1]

[1]Department of Statistical Science, University College London, London, UK

A PROOFS

Before going through the derivations, we briefly summarize the CLOSURE function in pseudocode.

Algorithm 2 CLOSURE

```
Input: Ancestrality matrix $\mathbf{M}$
Output: Updated ancestrality matrix $\mathbf{M}$

for $i, j \in \{1, \dots, d_X\}$ such that $i > j$ do
  if $i \preceq_{\mathbf{M}} j \wedge i \succeq_{\mathbf{M}} j \vee i \sim_{\mathbf{M}} j$ then
    $\mathbf{M}_{ij} \leftarrow i \sim j$
  else if $i \prec_{\mathbf{M}} j$ then
    $\mathbf{M}_{ij} \leftarrow i \prec j$
  else if $j \prec_{\mathbf{M}} i$ then
    $\mathbf{M}_{ij} \leftarrow j \prec i$
  end if
end for
converged $\leftarrow$ FALSE
while not converged do
  converged $\leftarrow$ TRUE
  for $i, j, k \in \{1, \dots, d_X\}$ such that $i \neq j \neq k, i > k$ do
    if $i \prec_{\mathbf{M}} j \prec_{\mathbf{M}} k \wedge \mathbf{M}_{ik} \neq i \prec k$ then
      $\mathbf{M}_{ik} \leftarrow i \prec k$, converged $\leftarrow$ FALSE
    else if $k \prec_{\mathbf{M}} j \prec_{\mathbf{M}} i \wedge \mathbf{M}_{ik} \neq k \prec i$ then
      $\mathbf{M}_{ik} \leftarrow k \prec i$, converged $\leftarrow$ FALSE
    end if
  end for
end while
```

This subroutine is important because it allows us to draw correct inferences in some cases where (R1)-(R3) do not apply. Consider, for instance, the following graphs:

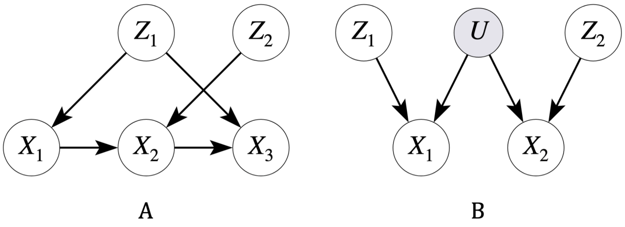

Figure 1: Example graphs illustrating how CBL-ORACLE exploits transitivity and antisymmetry to infer causal structure.

In Fig. 1(A), we may use (R1) to infer that $X_1 \prec X_2$ (since $Z_1 \perp\!\!\!\perp X_2 \mid Z_2 \cup [X_1]$) and again to infer that $X_2 \prec X_3$ (since $Z_2 \perp\!\!\!\perp X_3 \mid Z_1 \cup [X_2]$). However, this strategy will not allow us to infer that $X_1 \prec X_3$ due to the confounding signal from Z_1. Fortunately, the desired inference is easily drawn via transitivity ($X_1 \prec X_2 \wedge X_2 \prec X_3 \Rightarrow X_1 \prec X_3$).

In Fig. 1(B), the shaded node U denotes a latent variable. We use (R2) to infer that $X_1 \preceq X_2$ (since $Z_2 \not\perp\!\!\!\perp X_1 \mid Z_1 \cup [X_2]$) and again to infer that $X_2 \preceq X_1$ (since $Z_1 \not\perp\!\!\!\perp X_2 \mid Z_2 \cup [X_1]$). However, we cannot apply (R3) to learn that $X_1 \sim X_2$, since no set of observable non-descendants d-separates the two. Still, the desired inference can be drawn via antisymmetry ($X_1 \preceq X_2 \wedge X_1 \succeq X_2 \Rightarrow X_1 \sim X_2$).

A.1 PROOF OF THEOREM 1

CBL-ORACLE exhaustively applies (R1), (R2), and (R3) only to non-descendants that were confirmed to be such either by assumption ($\boldsymbol{Z}$) or by previous application of the rules, along with closure under transitivity and antisymmetry. Therefore, it boils down to the soundness of the rules themselves, as the starting set of non-descendants for all $X \in \boldsymbol{X}$ is $\boldsymbol{Z}$. As we mentioned in the main text, (R3) is a direct application of faithfulness since the conditioning set contains no descendants of X or Y. The correctness of (R1) and (R2) are a direct application of Lemma 1 of Magliacane et al. [2016] and the partial order knowledge.

In order to obtain $\mathbf{M}_{ij} = i \prec j$, it must be the case that (R1) was used with some $\boldsymbol{A} = \boldsymbol{Z} \cup \big\{X \in \boldsymbol{X} \backslash \{X_i, X_j\} : X \preceq \{X_i, X_j\}\big\}$ to detect a minimal deactivation of the form $W \perp\!\!\!\perp X_j \mid \boldsymbol{A}_{\backslash W} \cup [X_i]$ for some $W \in \boldsymbol{A}$. Since this confirms that $X_i \prec X_j$, we satisfy the assumptions of Entner et al. [2013]'s (R1). Therefore, using the same arguments, we conclude that both $\boldsymbol{A}$ and $\boldsymbol{A}_{\backslash W}$ are valid adjustment sets for (X_i, X_j). $\square$

A.2 PROOF OF THEOREM 2

Without loss of generality, assume that foreground variables are indexed such that $X_i \preceq X_j$ for all $j > i$.

Let us start with $d_X = 2$, and assume that $X_1 \sim X_2$. Then we either have that $X_1 \perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}$ or $X_1 \not\perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}$. In the former case, condition (i) applies, and we learn the proper structure via (R3). If background variables do not d-separate foreground variables, however, we may still be able to learn that $X_1 \sim X_2$ so long as each node activates a path from one of its non-descendants to the other node. Graphical criteria for this are characterized by condition (ii), in which case (R2) applies twice. Since $X_1 \preceq X_2$ and $X_2 \preceq X_1$ implies $X_1 \sim X_2$, we learn the true structure via closure under antisymmetry.

Now assume that $d_X = 3$, and our pair of interest is $X_1 \prec X_3$. If some background variable $W \in \boldsymbol{Z}$ is d-separated from X_3 given $\boldsymbol{Z}_{\setminus W}$, then condition (iii) is satisfied and (R1) will fire. However, even if condition (iii) does not hold for this pair, we may still draw the proper inference provided that it does hold for both $X_1 \prec X_2$ and $X_2 \prec X_3$. In this case, condition (iv) is satisfied and we infer that $X_1 \prec X_3$ by the transitivity of ancestral relations.

Assume that we are able to solve all ancestral relations for $\mathcal{G}_X$. As X_1 and X_{d_X} satisfy the premises with ancestral set $\boldsymbol{Z}$, we will learn either $X_1 \prec X_{d_X}$ or $X_1 \sim X_{d_X}$. Hence, we will know all the common ancestors of X_2 and X_{d_X} that are in $\boldsymbol{X}$. We also know that conditioning on these ancestors will not create an active backdoor path between X_2 and X_{d_X}, since $\boldsymbol{Z}$ is a valid adjustment set for (X_1, X_2) if $X_1 \prec X_2$, and both $\boldsymbol{Z}$ and $\boldsymbol{Z} \cup \{X_1\}$ are valid adjustment sets for (X_2, X_3) if $X_2 \prec X_3$. We can therefore redefine a new $\boldsymbol{Z}$ by recursively adding shared ancestors, and iterate the argument. $\square$

A.3 PROOF OF THEOREM 3

Given that CBL-ORACLE performs all possible tests allowed for a lazy oracle algorithm (subject to the conditions of finding all possible non-descendants at each iteration), this boils down to guaranteeing that we are not failing to take into account further implications of (R1)-(R3) and that the non-ancestral relationships are built up properly as iterations progress.

We start with the case where $d_X = 2$, and then prove the general case by induction. The bipartite setting is similar to that of [Entner et al., 2013], except we do not assume a causal order between X_1 and X_2. This means that we will be unable to draw conclusions in some cases where their algorithm can. For instance, CBL-ORACLE cannot distinguish $Z \rightarrow X_1 \leftarrow X_2$ from $Z \rightarrow X_1 \leftrightarrow X_2$. Armed with the *a priori* knowledge that $X_1 \preceq X_2$, however, Entner et al. [2013] correctly infer that $X_1 \sim X_2$ in the latter case (more on this example below).

When $d_X = 2$, there is at most one possible modification that can be made to $\mathbf{M}$, so we only need to consider $\boldsymbol{Z}$ as the set of non-descendants of $\{X_1, X_2\}$. If some lazy oracle algorithm $\mathcal{A}$ dominates CBL-ORACLE, then there exists some DAG $\mathcal{G}'$ that is indistinguishable from $\mathcal{G}$ according to (R1)-(R3) and closure rules, but for which $\mathcal{A}$ draws a more informative inference. Assume, for concreteness, that CBL-ORACLE outputs $X_1 \preceq X_2$ but $\mathcal{A}$ outputs $X_1 \prec X_2$.

The completeness of our (R1) follows from the completeness of (R1) in Entner et al. [2013]. That proof makes no use of the presumed partial order between X_1 and X_2, except that their algorithm does not need to flip the roles of X_1 and X_2 when applying the rules. We do the flip, but as shown before, the rule remains sound.

Assume (R3) cannot be applied to $\mathcal{G}$. It is clear that it cannot be applied to $\mathcal{G}'$ either, since adding an edge to $\mathcal{G}$ (that is, $X_1 \rightarrow X_2$) cannot introduce further independencies.

Assume (R1) cannot be applied to $\mathcal{G}$. Now, given $X_1 \not\perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}$ (that is, (R3) does not apply to $\mathcal{G}$), there is at least one path $\mathcal{P} := X_1 \leftarrow \cdots \rightarrow X_2$ which is active given $\boldsymbol{Z}$ in $\mathcal{G}$. We will show that no $W \in \boldsymbol{Z}$ exists such that $W \perp\!\!\!\perp_{\mathcal{G}'} X_2 \mid \boldsymbol{Z}_{\setminus W} \cup [X_1]$, i.e. (R1) does not apply to $\mathcal{G}'$ either. To see this, assume that (R1) does apply to $\mathcal{G}'$. Then, $W \perp\!\!\!\perp_{\mathcal{G}'} X_2 \mid \boldsymbol{Z}_{\setminus W} \cup \{X_1\}$, which implies $W \perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}_{\setminus W} \cup \{X_1\}$, since adding edge $X_1 \rightarrow X_2$ cannot create any new independence. It is not possible that $W \not\perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}_{\setminus W}$, otherwise (R1) would fire and incorrectly imply that $X_1 \rightarrow X_2$ in $\mathcal{G}$. Hence, $W \perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}_{\setminus W}$. Given that $X_1 \perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}_{\setminus W} \cup \{W\}$ by hypothesis, W cannot be in $\mathcal{P}$. Since $X_1 \rightarrow X_2$ is not in the equivalent $\mathcal{P}$ path in $\mathcal{G}'$, W must be connected to X_1 via some other path $\mathcal{P}'$ into X_1 that is active given $\boldsymbol{Z}_{\setminus W}$ in order for X_1 to d-separate X_2 from W. But the concatenation of $\mathcal{P}'$ with $\mathcal{P}$ in $\mathcal{G}'$ will contradict $W \perp\!\!\!\perp_{\mathcal{G}'} X_2 \mid \boldsymbol{Z}_{\setminus W} \cup \{X_1\}$. By symmetry with the case where $X_2 \rightarrow X_1$ is added, (R1) cannot be applied to either $\mathcal{G}$ nor $\mathcal{G}'$.

Assume (R2) cannot be applied to $\mathcal{G}$. Now, assume there exists some $W \in \boldsymbol{Z}$ such that $W \not\perp\!\!\!\perp_{\mathcal{G}'} X_2 \mid \boldsymbol{Z}_{\setminus W} \cup [X_1]$, i.e. (R2) applies to $\mathcal{G}'$. This will also imply $W \perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}_{\setminus W}$, as adding the edge could not have created a new independence. Therefore, it must be the case that $W \perp\!\!\!\perp_{\mathcal{G}} X_2 \mid \boldsymbol{Z}_{\setminus W} \cup \{X_1\}$, otherwise (R2) would fire for $\mathcal{G}$. This again results in a path $\mathcal{P}'$ which, concatenated with $X_1 \rightarrow X_2$, will contradict $W \perp\!\!\!\perp_{\mathcal{G}'} X_2 \mid \boldsymbol{Z}_{\setminus W}$.

Entner et al. [2013] include an extra inference rule with no analogue in our setup. They show that if $X_1 \preceq X_2$, and there exists some $W \in \boldsymbol{Z}$ such that $W \not\perp\!\!\!\perp X_1 \mid \boldsymbol{Z}_{\setminus W}$ and $W \perp\!\!\!\perp X_2 \mid \boldsymbol{Z}_{\setminus W}$, then $X_1 \sim X_2$. (This is how they learn $\mathcal{G}_X$ in the example above, $Z \rightarrow X_1 \leftrightarrow X_2$.) Call this pattern "cross-activation", as opposed to the (de)activation signatures characterized by (R1) and (R2). We have no need for cross-activation, since it is either redundant or inapplicable for a lazy oracle. (Note that cross-activation may be informative with classical oracles, where conditioning sets are unrestricted.) Redundancy is obvious when identifiability conditions (i) or (ii) of Thm. 2 are satisfied, as CBL-ORACLE will also infer that $X_1 \sim X_2$ in this case. Yet even when these conditions fail, lazy oracle cross-activation adds no new information. To see this, we must articulate a partial identification condition implicit in Thm. 2, indexed to be continuous with that theorem's list:

(v) If $X_i \preceq X_j$, then some $V \in \boldsymbol{X}_{\preceq j}$ is d-connected to X_i given $\boldsymbol{X}_{\preceq j} \backslash \{V\}$.

Cross-activation is inapplicable when no pair satisfies condition (v), since in that case we could not use (R2) to learn that $X_1 \preceq X_2$ in the first place (remember, this partial order is not "learned" by Entner et al. [2013]'s algorithm but assumed upfront). However, if (v) holds for all partially ordered pairs, then we may infer $X_1 \sim X_2$ via the closure of (R2). Thus, to be useful, cross-activation requires a setting in which $X_1 \preceq X_2$ may be learned through activation but $X_2 \preceq X_1$ cannot.

To summarize, we require a structure $W_1 \rightarrow \cdots \rightarrow X_1 \leftarrow \cdots \rightarrow X_2 \leftarrow \cdots \leftarrow W_2$, where:

(a) $X_1 \sim X_2$.
(b) $\{W_1, W_2\} \subseteq \boldsymbol{Z} \preceq \{X_1, X_2\}$.
(c) $W_2 \not\perp\!\!\!\perp X_1 \mid \boldsymbol{Z}_{\backslash W_2} \cup [X_2]$. This is how we infer $X_1 \preceq X_2$.
(d) $W_1 \not\perp\!\!\!\perp X_1 \mid \boldsymbol{Z}_{\backslash W_1} \wedge W_1 \perp\!\!\!\perp X_2 \mid \boldsymbol{Z}_{\backslash W_1}$. This is the hypothesis of cross-activation.
(e) $X_1 \not\perp\!\!\!\perp X_2 \mid \boldsymbol{Z}$. This prevents us from inferring the true structure via (R3).
(f) $W_1 \not\perp\!\!\!\perp X_2 \mid \boldsymbol{Z}_{\backslash W_1} \vee W_1 \perp\!\!\!\perp X_2 \mid \boldsymbol{Z}_{\backslash W_1} \cup \{X_1\}$. This prevents us from inferring the true structure via (R2).

These desiderata are inconsistent. Observe that the second conjunct of (d) is the negation of the first disjunct of (f). Thus we infer that $W_1 \perp\!\!\!\perp X_2 \mid \boldsymbol{Z}_{\backslash W_1} \cup \{X_1\}$. By (a), (d), and (e), we know that X_1 must be a collider on the path from W_1 to X_2. Thus conditioning on X_1 will activate this path, which we know to be unblocked by $\boldsymbol{Z}$ given (e). Thus it must be that $W_1 \not\perp\!\!\!\perp X_2 \mid \boldsymbol{Z}_{\backslash W_1} \cup \{X_1\}$. But this contradicts (f)'s second disjunct. Therefore if (a)-(d) are true, then it must be the case that either (e) is false, and we can learn $\mathcal{G}_X$ via (R3), or (f) is false, and we can learn $\mathcal{G}_X$ via (R2).

Hence, for $d_X = 2$, any failure to infer a structural relationship present in $\mathcal{G}$ using (R1)-(R3) and closure rules also means that there exists some different graph, with the same answers to the oracle, where that structural feature is not present. Assume this is the case for all $\mathcal{G}$ with $d_X = n$, for some $n > 2$. We will show this is also the case for $d_X = n + 1$.

Let X_d be any foreground vertex in $\mathcal{G}$ that has no children. The graph implied by the removal of X_d, $\mathcal{G}_{\backslash d}$, will have all of its lazy oracle-identifiable pairwise ancestral relationships in $\mathbf{M}$ resolved by the induction hypothesis and the fact that no descendants of a pair $\{X_i, X_j\}$ can be used in the conditioning set of a query. Assuming there exists some t by which we find all possible non-descendants of $X_i \in \boldsymbol{X} \backslash X_d$ (which is the case for the exhaustive search done by CBL-ORACLE), the oracle can then be invoked to sort all pairwise relationships involving X_d for iterations greater than t. □

A.4 PROOF OF THEOREM 4

As noted in the text, this is simply an instance of Shah and Samworth [2013]'s Eq. 8, adapted for our modified target, which is a conjunction of inclusion and exclusion statements rather than a single selection event. The arguments from their proof go through just the same so long as empirical rates $\hat{r}_\phi(Z)_\psi$ for all $Z \in \boldsymbol{L}_{\theta,\phi,\psi}$ are $-1/4$-concave distributed. See Fig. 2, Appx. B for empirical evidence supporting this assumption.

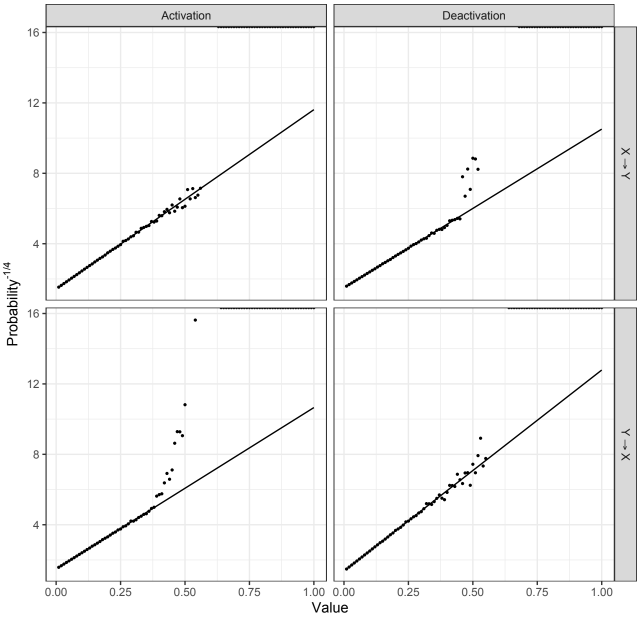

Figure 2: Example probability mass functions of $\hat{r}_\phi(Z)_\psi$, $Z \in \boldsymbol{L}_{\theta,\phi,\psi}$ (black points), alongside the $-1/4$-concave distribution (smooth line), which has maximum tail probability beyond 0.4. Differences in expected values are due to variation in θ across rates. See [Shah and Samworth, 2013, Appx. A.4].

B STABILITY SELECTION

The following definition of r-concavity is from Shah and Samworth [2013, Sect. 3.3]. To define the r-concave distribution, recall that the r^{th} generalized mean $M_r(a, b; \lambda)$ of $a, b \geq 0$ is given by

$$M_r(a, b; \lambda) = \{(1 - \lambda)a^r + \lambda b^r\}^{1/r}$$

for $r > 0$. This is also well-defined for $r < 0$ if we take $M_r(a, b; \lambda) = 0$ when $ab = 0$, and define $0^r = \infty$. In addition, we may define

$$M_0(a, b; \lambda) := \lim_{r \to 0} M_r(a, b; \lambda) = a^{1-\lambda} b^\lambda$$
$$M_{-\infty}(a, b; \lambda) := \lim_{r \to -\infty} M_r(a, b; \lambda) = \min(a, b).$$

We may now define r-concavity.

Definition 1. A non-negative function f on an interval $I \subset \mathbb{R}$ is *r-concave* if, for every $x, y \in I$ and $\lambda \in (0, 1)$, we have:

$$f((1 - \lambda)x + \lambda y) \geq M_r(f(x), f(y); \lambda).$$

Definition 2. A probability mass function f supported on $\{0, {}^1\!/_B, {}^2\!/_B, \ldots, 1\}$ is *r-concave* if the linear interpolant to $\{(i, f({}^i\!/_B)) : i = 0, 1, \ldots, B\}$ is r-concave.

We find that the empirical probability mass functions of (de)activation rates for low-rate features are approximately $-1/4$-concave. The fit is tightest for the lowest rates, which occur with the greatest frequency. See Fig. 2, especially the bottom left quadrants of all four panels, which account for the vast majority of probability mass. Missing observed probabilities (typically at x-axis values above 0.6) correspond to zero-frequency events, e.g. no feature is deactivated in 80% of training/validation splits.

Let d_A be the number of conditioning variables for a given regression. Following the recommendations of Shah and Samworth [2013, Sect. 3.4], we use $B = 50$ complementary pairs; fix $\hat{\theta} := d_A^{-1} \sum_{k=1}^{d_A} \hat{r}_\phi(A_k)_\psi$ for each $\phi \in \{d, a\}$, $\psi \in \{i \preceq j, j \preceq i\}$; and plug in d_A as a conservative estimate of $|\hat{\boldsymbol{L}}_{\hat{\theta},\phi,\psi}|$. See Alg. 3.

Algorithm 3 StabilitySelection

```
Input: Empirical rates $\hat{R}_{\phi,\psi}$, lower bound $\epsilon$, number of complementary subsamples $B$
Output: One of $\{i \prec j, i \preceq j, j \prec i, j \preceq i, \texttt{NA}\}$

Initialize: $\texttt{pos} \leftarrow \{0_b\}_{b=1}^{2B}$, $\texttt{out} \leftarrow \texttt{NA}$
$d_A \leftarrow |\hat{R}_{\phi,\psi}|$
$\hat{\theta} \leftarrow d_A^{-1} \sum_{k=1}^{d_A} \hat{r}_\phi(A_k)_\psi$
for $\tau \in \{1/2B, 1/B, \dots, 1\}$ s.t. $\tau \geq \epsilon$ do
  $\mathrm{hi}_\tau \leftarrow \min\{D(\hat{\theta}^2, 2\tau - 1, B, -1/2), D(\hat{\theta}, \tau, 2B, -1/4)\} d_A$
  $\hat{\boldsymbol{H}}_{\tau,\phi,\psi} \leftarrow \{k : \hat{r}_\phi(A_k)_\psi \geq \tau\}$
  $b \leftarrow 2B\tau$
  if $|\hat{\boldsymbol{H}}_{\tau,\phi,\psi}| > \mathrm{hi}_\tau$ then
    $\texttt{pos}_b \leftarrow 1$
  end if
end for
if $\sum_{b=1}^{2B} \texttt{pos}_b > 0$ then
  if $\phi = d \ \wedge \ \psi = i \preceq j$ then
    $\texttt{out} \leftarrow i \prec j$
  else if $\phi = a \ \wedge \ \psi = i \preceq j$ then
    $\texttt{out} \leftarrow i \preceq j$
  else if $\phi = d \ \wedge \ \psi = j \preceq i$ then
    $\texttt{out} \leftarrow j \prec i$
  else if $\phi = a \ \wedge \ \psi = j \preceq i$ then
    $\texttt{out} \leftarrow j \preceq i$
  end if
end if
return $\texttt{out}$
```

C SAMPLE ALGORITHM

The sample version of our CBL-ORACLE algorithm is provided in pseudocode below. The SAMPLE function draws uniformly without replacement from its first argument, producing a set of size equal to its second argument. The feature selection method s is a function with three arguments: a matrix of predictors, a vector of responses, and a set of row indices on which to operate. The output is an active set of features, as determined by the chosen method (e.g., lasso or stepwise regression). $\mathbf{R}$ is a $|\boldsymbol{A}| \times 4$ matrix storing empirical (de)activation rates. $\mathbf{R}_{[k:]}$ denotes the kth row of $\mathbf{R}$, while $\mathbf{R}_{[:\phi,\psi]}$ denotes the column of rates for a given $\phi \in \{d, a\}, \psi \in \{i \preceq j, j \preceq i\}$. The CONSISTENT function checks for inconsistent inferences at a candidate threshold τ (see Alg. 5).

Algorithm 4 CBL-SAMPLE

```
Input: Background variables $\boldsymbol{Z}$, foreground variables $\boldsymbol{X}$, feature selection method $s$, number of complementary subsamples $B$, omission threshold $\gamma$
Output: Ancestrality matrix $\mathbf{M}$

Initialize: converged $\leftarrow$ FALSE, $\mathbf{M} \leftarrow [\texttt{NA}]$
while not converged do
  converged $\leftarrow$ TRUE
  for $X_i, X_j \in \boldsymbol{X}$ such that $i > j,\ \mathbf{M}_{ij} = \texttt{NA}$ do
    $\boldsymbol{A} \leftarrow \boldsymbol{Z} \cup \big\{X \in \boldsymbol{X} \backslash \{X_i, X_j\} : X \preceq_{\mathbf{M}} \{X_i, X_j\}\big\}$
    for $b \in [B]$ do
      $\mathcal{D}_{2b-1} \leftarrow$ SAMPLE$([n], \lfloor n/2 \rfloor)$
      $\mathcal{D}_{2b} \leftarrow [n] \backslash \mathcal{D}_{2b-1}$
    end for
    for $b \in [2B]$ do
      $\hat{\boldsymbol{S}}_i^0(\mathcal{D}_b) \leftarrow s(\boldsymbol{A}, X_i, \mathcal{D}_b), \hat{\boldsymbol{S}}_i^1(\mathcal{D}_b) \leftarrow s(\boldsymbol{A} \cup X_j, X_i, \mathcal{D}_b)$
      $\hat{\boldsymbol{S}}_j^0(\mathcal{D}_b) \leftarrow s(\boldsymbol{A}, X_j, \mathcal{D}_b), \hat{\boldsymbol{S}}_j^1(\mathcal{D}_b) \leftarrow s(\boldsymbol{A} \cup X_i, X_j, \mathcal{D}_b)$
    end for
    $r_0 \leftarrow \#\{b : X_j \notin \hat{\boldsymbol{S}}_i^1(\mathcal{D}_b) \vee X_i \notin \hat{\boldsymbol{S}}_j^1(\mathcal{D}_b)\}$
    if $r_0 > \gamma$ then
      $\mathbf{M}_{ij} \leftarrow i \sim j$, converged $\leftarrow$ FALSE
    else
      $\mathbf{R} \leftarrow [\texttt{NA}]$
      for $k \in \{1, \dots, |\boldsymbol{A}|\}$ do
        $\hat{r}_d(A_k)_{i \preceq j} \leftarrow \#\{b : A_k \in \hat{\boldsymbol{S}}_j^0(\mathcal{D}_b) \wedge A_k \notin \hat{\boldsymbol{S}}_j^1(\mathcal{D}_b)\}/2B$
        $\hat{r}_a(A_k)_{i \preceq j} \leftarrow \#\{b : A_k \notin \hat{\boldsymbol{S}}_i^0(\mathcal{D}_b) \wedge A_k \in \hat{\boldsymbol{S}}_i^1(\mathcal{D}_b)\}/2B$
        $\hat{r}_d(A_k)_{j \preceq i} \leftarrow \#\{b : A_k \in \hat{\boldsymbol{S}}_i^0(\mathcal{D}_b) \wedge A_k \notin \hat{\boldsymbol{S}}_i^1(\mathcal{D}_b)\}/2B$
        $\hat{r}_a(A_k)_{j \preceq i} \leftarrow \#\{b : A_k \notin \hat{\boldsymbol{S}}_j^0(\mathcal{D}_b) \wedge A_k \in \hat{\boldsymbol{S}}_j^1(\mathcal{D}_b)\}/2B$
        $\mathbf{R}_{[k:]} \leftarrow (\hat{r}_d(A_k)_{i \preceq j}, \hat{r}_a(A_k)_{i \preceq j}, \hat{r}_d(A_k)_{j \preceq i}, \hat{r}_a(A_k)_{j \preceq i})$
      end for
      $\epsilon \leftarrow \min\big\{\tau \in \{1/2B, 1/B, \dots, 1\} : \textsc{Consistent}(\mathbf{R}, \tau) = \texttt{TRUE}\big\}$
      for $\phi \in \{d, a\}, \psi \in \{i \preceq j, j \preceq i\}$ do
        $\mathbf{M}_{ij} \leftarrow \mathbf{M}_{ij} \wedge \textsc{StabilitySelection}(\mathbf{R}_{[:\phi,\psi]}, \epsilon, B)$
        if $\mathbf{M}_{ij} \neq \texttt{NA}$ then
          converged $\leftarrow$ FALSE
        end if
      end for
    end if
  end for
  $\mathbf{M} \leftarrow \textsc{Closure}(\mathbf{M})$
end while
```

This function checks for two types of errors. Internal errors occur when inconsistent inferences are drawn for a single non-descendant; external errors occur when inconsistent inferences are drawn across multiple non-descendants.

Algorithm 5 CONSISTENT

```
Input: Empirical rate matrix $\mathbf{R}$, inference threshold $\tau$
Output: One of $\{$TRUE, FALSE$\}$

$d_A \leftarrow$ NROW($\mathbf{R}$)
int_error_vec $\leftarrow \{0_k\}_{k=1}^{d_A}$
for $k \in \{1, \ldots, d_A\}$ do
  if $\#\big\{j \in \{1, \ldots, 4\} : \mathbf{R}_{kj} \geq \tau\big\} > 1$ then
    int_error_vec$_k \leftarrow 1$
  end if
end for
if $\sum_{k=1}^{d_A}$ int_error_vec$_k > 0$ then
  int_error $\leftarrow$ TRUE
else
  int_error $\leftarrow$ FALSE
end if
d_ij $\leftarrow \#\big\{k \in \{1, \ldots, d_A\} : \mathbf{R}_{k1} \geq \tau\big\} > 0$
a_ij $\leftarrow \#\big\{k \in \{1, \ldots, d_A\} : \mathbf{R}_{k2} \geq \tau\big\} > 0$
d_ji $\leftarrow \#\big\{k \in \{1, \ldots, d_A\} : \mathbf{R}_{k3} \geq \tau\big\} > 0$
a_ji $\leftarrow \#\big\{k \in \{1, \ldots, d_A\} : \mathbf{R}_{k4} \geq \tau\big\} > 0$
if $\big($d_ij $\wedge$ (d_ji $\vee$ a_ji)$\big) \vee \big($d_ji $\wedge$ (d_ij $\vee$ a_ij)$\big)$ then
  ext_error $\leftarrow$ TRUE
else
  ext_error $\leftarrow$ FALSE
end if
if int_error $\vee$ ext_error then
  out $\leftarrow$ FALSE
else
  out $\leftarrow$ TRUE
end if
return out
```

D EXPERIMENTS

Complete code for all experiments can be found at `https://github.com/dswatson/cbl/paper`. Our simulation design is as follows:

- Background variables $\boldsymbol{Z}$ are drawn from a multivariate normal distribution $\mathcal{N}(\mathbf{0}, \Sigma)$, where $\mathbf{0}$ denotes a length-d_Z vector of 0's and Σ is a Toeplitz matrix with autocorrelation $\rho = 0.25$. Variance for each $Z \in \boldsymbol{Z}$ is fixed at $1/d_Z$.
- In nonlinear settings, we create a new matrix $\tilde{\boldsymbol{Z}}$ in which 80% of background variables undergo some nonlinear transformation. Specifically, the following transformations are applied with equal probability:
  - **Quadratic**: $\tilde{Z} = Z^2$
  - **Square root**: $\tilde{Z} = \sqrt{|Z|}$
  - **Softplus**: $\tilde{Z} = \log(1 + \exp(Z))$
  - **ReLU**: $\tilde{Z} = \max(0, Z)$
- Edges from $\boldsymbol{Z}$ to $\boldsymbol{X}$ are randomly generated with some fixed probability $(1 -$ sparsity$)$. In the linear case, foreground variables are given by a linear combination of parents, $X_i = \sum_{j<i} \beta_{ij} A_j + \epsilon_i$, where $A_j \in \boldsymbol{A} = \boldsymbol{Z} \cup \boldsymbol{X}_{\prec i}$. In the nonlinear case, we substitute $\tilde{A}$ for A, with transformations described above. Nonzero weights $\beta \neq 0$ are Rademacher distributed, i.e. drawn uniformly from $\{-1, 1\}$. Residuals ϵ_i are independent Gaussians with variance selected to ensure the target SNR.

For our bivariate experiments, we draw 100 random graphs according to each data generating process (with expected sparsity $1/2$ and SNR = 2) and record results with lasso regression (for linear systems) and gradient boosting (for nonlinear systems). The data generating process for the unidentifiable setting (c) is identical to (b)'s, except that half the shared parents of X and Y are removed from $\boldsymbol{Z}$.

CBL begins by splitting the data intro training and test sets, with the conventional ratio 4:1. For both lasso and GBM, features are selected according to model performance on the test set. For the former, this requires a sequence of values for the regularization parameter λ, automatically generated by the `glmnet` package [Friedman et al., 2010]. For the latter, we train a forest of up to 3500 trees with early stopping if performance does not improve after 10 rounds. This is efficiently implemented via the `lightgbm` package [Ke et al., 2017]. Features never selected for splits are automatically discarded, which is how GBMs naturally accommodate sparse model fits [Bühlmann and Yu, 2003].

To implement Entner et al. [2013]'s constraint-based benchmark, we follow the authors' advice in using conditional independence tests to infer both conditional *dependence* (for low p-values) and *independence* (for high p-values). We set decision thresholds of $p < 0.1$ and $p > 0.5$ for these respective tasks. We sample 1000 variable-subset pairs for linear data and 500 for nonlinear, as the testing subroutine in the latter case is more computationally intensive. Since instances of $W \perp\!\!\!\perp Y \mid \boldsymbol{A}_{\backslash W} \cup [X]$ proved far more elusive than instances of $X \perp\!\!\!\perp Y \mid \boldsymbol{A}$—i.e., the method finds separating sets with much higher frequency than it does minimal deactivators—we used different thresholds for these two cases. Specifically, we declare $X \to Y$ if minimal deactivations are detected in just 0.5% of all trials, while we infer $X \sim Y$ if separating sets are found in 20% of trials. These parameters were established after considerable experimentation, as the authors provide little guidance on such matters. Different data generating processes will likely require different thresholds to guarantee reasonable results.

For the score-based benchmark, we fit our model quartet with either lasso (for linear data) or GBM (for nonlinear data) and compute the proportion of variance explained on a test set of m samples via the formula

$$\text{PVE} = 1 - \frac{\sum_{i=1}^m \epsilon_i^2}{\sum_{i=1}^m (y_i - \overline{y})^2},$$

where ϵ_i denotes the model residual for sample i and Y is the outcome variable with empirical mean $\overline{y}$. Then, using the same indexing as above, we score the hypotheses as follows:

$$\begin{aligned}
X \to Y &: \ \hat{g}_1 = (\text{PVE}_X^0 + \text{PVE}_Y^1)/2 \\
X \leftarrow Y &: \ \hat{g}_2 = (\text{PVE}_Y^0 + \text{PVE}_X^1)/2 \\
X \sim Y &: \ \hat{g}_3 = (\text{PVE}_X^0 + \text{PVE}_Y^0)/2.
\end{aligned}$$

We evaluate potential dependencies between residuals and predictors via Pearson correlation with $\alpha = 0.1$.

Multivariate experiments were run using the RFCI and GES implementations in the pcalg package [Kalisch et al., 2012]. The SNP and mRNA data for *S. cerevisiae*, originally gathered by Brem and Kruglyak [2005], are distributed with the trigger package [Chen et al., 2007].

References

Rachel B. Brem and Leonid Kruglyak. The landscape of genetic complexity across 5,700 gene expression traits in yeast. *Proc. Natl. Acad. Sci.*, 102(5):1572–1577, 2005.

Peter Bühlmann and Bin Yu. Boosting with the L_2 loss. *J. Am. Stat. Assoc.*, 98(462):324–339, 2003.

Lin S Chen, Frank Emmert-Streib, and John D Storey. Harnessing naturally randomized transcription to infer regulatory relationships among genes. *Genome Biol.*, 8(10):R219, 2007.

Doris Entner, Patrik Hoyer, and Peter Spirtes. Data-driven covariate selection for nonparametric estimation of causal effects. In *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics*, pages 256–264, 2013.

Jerome H. Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.*, 33(1):1–22, 2010.

Markus Kalisch, Martin Mächler, Diego Colombo, Marloes H. Maathuis, and Peter Bühlmann. Causal inference using graphical models with the R package pcalg. *J. Stat. Softw.*, 47(11):1–26, 2012.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Sara Magliacane, Tom Claassen, and Joris M. Mooij. Ancestral causal inference. In *Advances in Neural Information Processing Systems 30*, page 4473–4481, 2016.

Rajen D. Shah and Richard J. Samworth. Variable selection with error control: Another look at stability selection. *J. R. Statist. Soc. B*, 75(1):55–80, 2013.