SUPPLEMENTAL DIGITAL CONTENT (corrected)

To accompany: Oates, Kasza, Simpson, Forbes. "Repair of partly misspecified causal diagrams"

This technical appendix supplements Oates et al. and provides full details for the PC-VET method. This appendix is organised as follows: In Sec. 1 the statistical properties of vetting are discussed at a high level. Sec. 2 introduces notation for the PC-VET algorithm that is described in detail in Sec. 3 and subjected to theoretical analysis in Sec. 4. In Sec. 5 we present additional results on the MCCS analysis from the main text and provide an auxiliary discussion of our results in Sec. 6. Then Sec. 7 contains algorithmic pseudo-code for PC-VET and Sec. 8 contains all auxiliary figures.

1. EFFICIENCY CONSIDERATIONS

In this work we restricted attention to the addition of edges to the expert's diagram, motivated by the fact that consistent estimation of the effect of the exposure on the outcome is possible if estimates are based on a DAG that contains at least those edges that are present in the true diagram. Our approach is "causally conservative", in the sense that we err on the side of a diagram with too many edges. However, "statistical conservatism" requires that the number of non-essential edges be limited in some way. Such control is necessary from an efficiency standpoint: using a "full" diagram containing all possible edges will lead to high variance estimates for the effect of the exposure of interest on the outcome, and vetting will trivially return the full diagram. Here there is an analogy with propensity score modelling, where a valid model must include all confounders of exposure on outcome but, for efficiency reasons, and because inclusion of such variables may increase bias in some situations, it is necessary to restrict the inclusion of variables that are independent of the outcome (1,2). In this paper the number of additional edges, above those given by the expert, is controlled by setting a threshold for the level of evidence in data required for edge addition. This implementation attempts to provide a sensible balance between the competing causal and statistical objectives.

A strong precedent for a two-stage approach to causal inference, such as the one proposed here, has been established by recent work in the high-dimensional statistics literature. In that work, the focus is estimation of causal effects. In the first stage, all variables are screened and a subset is selected, which are then included in a regression model in the second stage (3-5). The method proposed in this paper is also a two-stage approach, distinguished in its focus on integrating expert information.

2. NOTATION AND CAUSAL DIAGRAMS

Our study begins with a putative causal diagram elicited from an expert, represented by a DAG $G_0 = (V_0, E_0)$, where V_0 is an index set whose entries represent all variables relevant to causal inference, and $E_0 \subset V_0 \times V_0$ is the set of directed edges between these variables, characterising direct causal relationships. For example, if the expert believes that variable i is a direct cause of variable j, then in the expert's DAG there will be an edge from variable i to variable j: $(i, j) \in E_0$. Our interest in causal diagrams is motivated by their role in the selection of confounding variables to be adjusted for when estimating the effect of an exposure on an outcome, for example using a propensity score model or an outcome regression model.

If $G_{true} = (V_{true}, E_{true})$ is the "true" underlying DAG, then there are number of ways in which G_0 could differ from G. The most serious is the misspecification of the set of variables: $V_0 \neq V_{true}$. We assume that this does not occur, i.e. that the set of variables to be considered has been correctly specified (denoted by V). Additionally, we suppose that all variables are measured. That is, we assume the causal sufficiency assumption. This is a strong assumption, which we return to in Sec. 6.

Supposing that $V_0 = V$, G_0 and G_{true} will differ if and only if $E_0 \neq E_{true}$. This can occur in several ways that are highlighted in the main text. To recall: (C1) The expert is "essentially correct", providing a causal super-model of the truth: $E_{true} \subseteq E_0$. (C2) The expert is "weakly incorrect", meaning there exists a causal super-model containing both the truth and the expert's model: $E_{true} \nsubseteq E_0$, but E_0 can be extended, by adding edges, to a set E such that $E_{true} \subseteq E$ and E does not contain any cycles. (C3) The expert is "strongly incorrect", meaning that neither (C1) nor (C2) hold. Examples of cases (C1)-(C3) for a simple true DAG are provided in eFigure 1.

This paper restricts attention to cases (C1) and (C2): when the expert is essentially correct or weakly incorrect. We say that such an expert is "directionally informed". In (C1) the expert may over-specify the edge set: in addition to true direct causal relationships, the expert incorrectly believes there is a direct causal relationship between at least one pair of variables when there is no such relationship. In (C2) the expert overlooks at least one direct cause and may specify additional edges such that, if added to G, the resulting graph would be acyclic. The case (C3) can arise if the expert includes an incorrectly directed edge $i \rightarrow j$ in E_0 where $i \leftarrow j$ belongs to E_{true} or if the expert includes an edge such that a cycle would be induced were that edge included in E_{true}. (C3) is far more challenging as vetting in this case appears to require full *de novo* structure learning.

3. STRUCTURE LEARNING AND VETTING DAGS

Structure learning algorithms have been widely studied (6,7). Our implementation of vetting proceeds through the application of a structure learning algorithm to the data set of interest, restricting the search space to the subspace of all DAGs $G = (V, E)$ satisfying $E_{true} \subseteq E$. That is, only super-graphs of G_0 are considered. To remain agnostic to the causal problem of ultimate interest, we focus on structure learning algorithms that are model-free, being based on tests of conditional independence. The aim is to determine if any DAGs with additional edges provide a better agreement with the independence relationships in the data than the expert-elicited DAG.

In general, it is not possible to identify graphical structure from conditional independence, as multiple DAGs may encode identical conditional independence relationships (8). Such graphs are called "Markov equivalent" (9). The set of DAGs can thus be partitioned into classes of Markov equivalent DAGs. For our application we took inspiration from the PC algorithm (6), which (in its unrestricted form) returns a Markov equivalence class of DAGs. The PC algorithm is agnostic to the statistical problem that is of ultimate interest, since no functional or distributional assumptions are required for implementation. Additionally, the PC-algorithm is readily available in the TETRAD software (10) and in the pcalg package in R (11).

The "PC-VET" algorithm (oracle version), described in full in Sec. 3, is summarised as consisting of two stages:

I. Begin with a fully connected undirected graph: all variables are connected to all other variables with undirected edges. Each pair of variables i and j is then considered in turn. If an edge between i and j is not in the expert's edge set E_0, and there exists a set of variables $S \subseteq V \setminus \{i, j\}$ such that i is conditionally independent of j given S, the edge $i - j$ is deleted.

II. Second, each undirected edge $i - j$ is replaced by a directed edge $i \to j$ whenever that directed edge belongs to E_0: the expert's directed edges are included in the graph. Once this is completed, an attempt is made to orient as many of the remaining undirected edges as possible, using both information from the sets S from Stage I and deductive logic, based on the algebraic structure of conditional independence.

The first stage of the algorithm builds an undirected skeleton for the DAG, by removing edges whenever there exists a set of variables that explains the co-variation of i and j, when the expert does not deem such an edge necessary. In practice we do not have access to an oracle for conditional independence, so a judgement is made on the basis of a hypothesis test applied to the available data. Following the literature on the PC algorithm, for binary data we used a chi-squared test of conditional independence at a significance level $\alpha = 0.05$. The sensitivity of the final equivalence class of DAGs to the significance level can be assessed by running PC-VET with different values of α. Alternatively, a modification of the PC algorithm to

allow control of the false discovery rate is available (12), though we do not consider it here. The second stage of the algorithm performs deductive reasoning on the basis of both the expert information and the data-driven conditional independence information, by directing the expert's edges as specified, and ensuring that the directions of new edges comply with those directions. The full version of the PC-VET algorithm and the theoretical properties of the algorithm are provided in Sec. 7.

The following theoretical properties are established in Sec. 4 below: In cases (C1-2), the oracle version of PC-VET returns a partially directed acyclic graph that characterises a vetting equivalence class. Moreover, under an additional assumption (A3) in case (C2), this equivalence class contains the DAG with edge set $E_{true} \cup E_0$. All members of this vetting equivalence class (i) contain the expert's edges, and (ii) are valid for causal inference, being super-sets of the true causal graph.

For a given causal quantity of interest, different DAGs in the vetting equivalence class may require different adjustment sets of variables. Depending on the application, it may be more appropriate to average the resulting estimates of the causal effect over all members of the vetting equivalence class or use the graph that results in the most conservative inferences (13). The number of DAGs in a Markov equivalence class approaches about 3.7 as the number of vertices increases (14), and hence provides an asymptotic upper bound on the size of a vetting equivalence class.

4. THEORETICAL PROPERTIES

In this section we provide a full formal definition of the PC-VET algorithm and establish its theoretical properties.

Our analysis proceeds under the assumption of a causally sufficient set of observable random variables $\{X_i\}_{i\in V}$, with finite index set V. We write $G=(V,E)$ for a general directed acyclic graph (DAG) whose vertex set is V and whose edge set is $E \subset V \times V$. All DAGs below share the same vertex set V. A motif of the form $i \to j \leftarrow k$ where i and k are non-adjacent is known as a "v-structure". Define a "path" $P=(i_1,\ldots,i_m) \in E \times \ldots \times E$ in a DAG G to be a sequence of vertices that are connected by edges, agnostic of their direction. A node i_k, $1<k<m$, on P is said to be a "collider" if the motif $i_{k-1} \to i_k \leftarrow i_{k+1}$ occurs. A path $P=(i_1,\ldots,i_m)$ between i_1 to i_m in a DAG G is said to be "blocked" by a set $S \subseteq V \setminus \{i_1,i_m\}$ whenever there is a node i_k on P such that one of the following hold: (i) i_k is not a collider on P and $i_k \in S$; (ii) i_k is a collider on P and neither i_k nor any of its descendants are in S. Given disjoint subsets $A,B,C \subset V$ we say that A and B are d-separated by C in the DAG G if every path between vertices in A and B is blocked by S. We will use the shorthand $A \subseteq B$ for DAGs $A=(V,E_A)$ and $B=(V,E_B)$ whenever $E_A \subseteq E_B$; i.e. all of the edges that

are present in A are also present in B. Write $\mathrm{Pa}_G(X_i) \subseteq (X_j)_{j \in V \setminus \{i\}}$ for the parents of node i in G. For $G = (V, E)$ and $G' = (V, E')$, write $G \cup G'$ for the DAG with edge set $E \cup E'$.

Write $\mathbb{P}$ for the (observational) joint distribution of the random variables $\{X_i\}_{i \in V}$. To limit scope we consider data which are independent samples $\mathbf{x}_m = \{x_i^m\}_{i \in V}$ generated from $\mathbb{P}$ and write $\mathbf{x} = \{\mathbf{x}_m\}_{m=1}^n$ for the full dataset. Denote the empirical distribution of the random variables based on data $\mathbf{x}$ by $\mathbb{P}_{\mathbf{x}}$.

Our vetting approach is straight-forward to describe. In brief, given data $\mathbf{x}$ generated from a true (but unknown) DAG G_{true}, we proceed as follows: Given the expert's DAG G_0 we perform structure learning over the set

$$\mathcal{D}(G_0) = \{G : G \text{ is a DAG}, G_0 \subseteq G\}$$

to determine whether the element $\hat{G} \in \mathcal{D}(G_0)$ that provides the best agreement with the independence structure in the data $\mathbf{x}$. Then we base our causal inferences on $\hat{G}$ rather than G_0 (with the possibility that $\hat{\mathrm{G}} = G_0$). In situations where certain edges E_{untrue} do not make sense on physical grounds, the search space may be further constrained as

$$\mathcal{D}(G_0, E_{untrue}) = \{G : G = (V, E) \text{is a DAG}, G_0 \subseteq G, E \cap E_{untrue} = \emptyset\}.$$

For simplicity of presentation we assume that $E_{untrue} = \varnothing$ below.

It is well-known that structure learning methods based only on conditional independence information cannot, in general, identify a unique DAG (15). The same holds true for vetting, under a modified notion of equivalence that accounts for expert information:

Definition. (Vetting equivalence). Given G_0, write $G \sim G'$ whenever G, G' are DAGs such that (i) $G_0 \subset G, G'$, (ii) G, G' have the same edges (agnostic to direction) and (iii) G, G' contain the same v-structures. The relation $\sim$ defines an equivalence class structure $\mathcal{D}(G_0)/\sim$ and we refer to elements $\mathbb{G} \in \mathcal{D}(G_0)/\sim$ as `vetting equivalence classes'.

This equivalence relationship does *not* coincide with Markov equivalence, since expert information can sometimes be used to orient some edges that cannot otherwise be oriented, with knock-on consequences for other parts of the graph due, for example, to the global acyclicity constraint. Nevertheless, as with Markov equivalence, a vetting equivalence class $\mathbb{G}$ can be characterised by a partially directed acyclic graph (PDAG):

Proposition 1. Given G_0, a vetting equivalence class $\mathbb{G} \in \mathcal{D}(G_0)/\sim$ corresponds to a unique PDAG $P = P(\mathbb{G})$, defined such that the directed edges in P are those which appear in all members of $\mathbb{G}$ and the undirected edges in P are those which appear in both orientations among members of $\mathbb{G}$.

Proof: Our task is to prove that the map from $\mathbb{G}$ to $P(\mathbb{G})$ is injective. i.e. Given $\mathbb{G}, \mathbb{G}' \in \mathcal{D}(G_0)/\sim$ with $\mathbb{G} \neq \mathbb{G}'$, our task is to show that $P(\mathbb{G}) \neq P(\mathbb{G}')$. By definition all members of $\mathbb{G}$ (resp. $\mathbb{G}'$) share the same edge set E (resp. E). The skeleton of $P(\mathbb{G})$ is identical to the skeleton of E. Thus if $E \neq E'$ then $P(\mathbb{G}) \neq P(\mathbb{G}')$. So it remains only to consider the case where $\mathbb{G}, \mathbb{G}'$ have the same skeleton but different v-structures. Suppose all members of $\mathbb{G}$ include the v-structure $i \to j \leftarrow k$, but that this is not the case for $\mathbb{G}'$. From the definition, $P(\mathbb{G})$ must contain $i \to j \leftarrow k$. Now, if $P(\mathbb{G}')$ also contains $i \to j \leftarrow k$ then from the definition of $P(\mathbb{G}')$ it follows that all members of $\mathbb{G}'$ contain $i \to j \leftarrow k$, which contradicts our supposition. This proves that the map from $\mathbb{G}$ to $P(\mathbb{G})$ is injective, as required.

Assuming that we are in case (C1) or, under an additional assumption (A3), in case (C2), and assuming that we have access to oracle information on the conditional independence structure of the distribution $\mathbb{P}$, any logically correct structure learning algorithm that is well-defined on $\mathbb{G} \in \mathcal{D}(G_0)/\sim$ will, under standard assumptions (A1-2) below, provide a vetting equivalence class $\mathbb{G} \in \mathcal{D}(G_0)/\sim$ that satisfies $G_{\text{true}} \subseteq G$ for every element $G \in \mathbb{G}$. Below we establish that the PC-VET algorithm is well-defined (Proposition 2) and logically correct (Proposition 3). Outside the oracle setting, we hope that data $\mathbf{x}$ lead us to choose $\mathbb{G}$ such that, with high probability, $G_{\text{true}} \subseteq G$ for every element $G \in \mathbb{G}$. Thus the output $\mathbb{G}$ of vetting will be an equivalence class of DAGs such that (with high probability) each constituent DAG is valid for causal inference. Empirical experiments in the main text suggest that the PC-VET algorithm remains effective outside the oracle setting.

Our method, summarised in the main text, is based on a restriction of the PC algorithm so as to enforce edges that are prescribed by the expert. It is described in detail in Algorithm 1.

To describe the oracle performance of PC-VET, we make the following standard structural assumptions (6):

(A1) Markov property: For all disjoint subsets $A, B, C \subset V$, if A and B are *d*-separated by C in the true graph G_{true}, then $X_A \perp X_B | X_C$, where $\perp$ denotes conditional independence.

(A2) Faithfulness property: For all disjoint subsets $A, B, C \subset V$, if $X_A \perp X_B | X_C$, then A and B are *d*-separated by C in the true graph G_{true}.

The following oracle properties, whose proofs are only sketched for brevity below, are consequences of established results on correctness of the standard PC algorithm.

Proposition 2. (PC-VET is well-defined). Given an empirical distribution $\mathbb{P}_{\boldsymbol{x}}$, the PC-VET algorithm returns a PDAG $P = P(\mathbb{G})$ corresponding to a unique vetting equivalence class $\mathbb{G} \in \mathcal{D}(G_0)/\sim$.

Sketch: The aim is to show that the output P of the PC-VET algorithm corresponds to a PDAG P of the form $P(\mathbb{G})$ for some $\mathbb{G} \in \mathcal{D}(G_0)/\sim$ and hence defines a unique vetting equivalence class $\mathbb{G} \in \mathcal{D}(G_0)/\sim$ (Proposition 1). Firstly, observe that the output of Stage I is an undirected

graph G containing G_0. At the start of Stage II the edges E_0 in G are oriented in a way that G is acyclic, since G_0 is itself a DAG. Subsequent operations indexed by $k \notin S(i,j)$ cannot introduce cycles, so that prior to application of (R1-4), the graph G is a PDAG. The main result then follows from Meek (1995) (16) who proved that (R1-4) were sufficient to find all logically implied orientations while preserving acyclicity. Thus application of (R1-4) produces a PDAG P of the form $P(\mathbb{G})$ for some vetting equivalence class $\mathbb{G} \in \mathcal{D}(G_0)/\sim$.

For case (C2) we require an additional assumption:

(A3) No triplets (i,j,k) can be formed such that

I. $j \leftarrow k \in G_0$ and $j \leftarrow k \notin G_{true}$

II. $i \rightarrow j \notin G_0$ and $i \rightarrow j \in G_{true}$

III. $i \rightarrow k, i \leftarrow k \notin G_0$

Proposition 3. (Oracle behaviour: PC-VET is logically correct). Consider cases (C1) and (C2). Given oracle information $\mathbb{P}$, under (A1-3), the PC-VET algorithm returns a PDAG $P(\mathbb{G})$ corresponding to the unique vetting equivalence class $\mathbb{G} \in \mathcal{D}(G_0)/\sim$ generated by $G_0 \cup G_{\text{true}}$ where G_{true} is the true DAG.

Sketch: Note that (C1) and (C2) each imply G_0 can be extended to a DAG $\hat{G}$ satisfying $G_{\text{true}} \subseteq \hat{G}$. The unique minimal choice for $\hat{G}$, in the sense of containing the fewest possible edges, is $\hat{G} = G_0 \cup G_{\text{true}}$. The PC-VET algorithm starts, in Stage I, with the complete undirected graph and removes precisely those edges that do not appear (in directed form) in the DAG $\hat{G}$, due to (A1-2) and minor modification of the standard analysis of the PC algorithm (6). The result is an undirected graph, equal to the skeleton of $\hat{G}$, which is subsequently passed to Stage II of the PC-VET algorithm. In Stage II the directed edges E_0 are imposed, then repeated application of (R1-4) is necessary and sufficient to establish all logically implied edge orientations (16). In the case of (C2), assumption (A3) ensures that the orientation phase of the PC algorithm is not affected by v-structures that are incorrectly introduced due to the inclusion of incorrect edges in G_0. The result is a PDAG P where (using Proposition 1) we have $P = P(\mathbb{G})$ such that $\hat{G} \in \mathbb{G}$.

Proposition 3 demonstrates that in the oracle setting, or equivalently given unlimited data, the PC-VET algorithm returns a vetting equivalence class of DAGs, each element of which is a valid DAG for the purpose of causal inference. In practice the algorithm operates with finite data and the error rate in testing of conditional independences must be controlled to ensure that the output of the empirical version of the PC-VET algorithm is a set of DAGs that are each valid for the inference problem.

This performance of the PC-VET algorithm at finite n can be explored theoretically, under additional assumptions, via slight modification to the theory provided in (17). This was,

however, beyond the scope of the present paper. Similarly, there are numerous extensions to the PC algorithm in the literature that could be employed with minor modification in the context of vetting; these were also beyond the scope of this paper.

5. ADDITIONAL RESULTS FOR THE MCCS ANALYSIS

For the MCCS application, a contingency table of Living Alone against Alcohol Intake makes clear that the expert DAG was incorrect in its assumption of independence of these two variables:

Alcohol Intake \ Living Alone	No	Yes
0-39 g/day	6733 (85%)	844 (80%)
40+ g/day	1215 (15%)	208 (20%)

Restricting to participants who lived alone, the proportion who reported high Alcohol Intake was 0.20, while for participants who did not live alone the corresponding proportion was 0.15. It seems that in this case the expert has made a mistake and should reconsider their causal diagram.

6. DISCUSSION

Through simulation we demonstrated how vetting can confer more accurate estimation of causal effects. The strong observed performance occurs since inclusion of overlooked edges can change the set of variables to be adjusted for when estimating the effect of an exposure on the outcome and reduce bias in the estimation of the effect.

Vetting is an easy-to-implement, independent pre-processing step that can be used to detect and avoid under-specification of causal models. However, the vetting procedure is unable to exclude or reverse the direction of any edges that are included in the expert-elicited DAG. Contradicting expert-supplied edges appears to require *de novo* learning of the entire DAG structure from data. For the purposes of vetting, we would encourage the expert to avoid the strongly incorrect case by providing information only on edges about which they are certain of the directionality.

This work proceeded under the (strong) assumptions that all relevant variables were included in the expert-elicited DAG and that all variables were measured. Our approach could be extended to “partial ancestral graphs”, which permit the inclusion of latent variables and can be learned from data using the fast causal inference algorithm (1,2). Using such an approach would permit the inclusion of unmeasured variables in the expert-elicited DAG, but would be more data-intensive.

In practice we expect that vetting will be used to aid the expert elicitation of causal structure, as a tool to allow an expert to converge on a suitable graphical structure. In this respect it may be desirable to perform pre-processing and estimation on separate subsets of the data, to prevent any suggestion of "using the data twice". We took a similar approach in our application of vetting to the MCCS data: we used a random subset of the data to vet the DAG, and then estimated odds ratios using the entire data set (which was itself a random subsample of the MCCS dataset, and thus the final estimates of the total causal effect are for illustrative purposes only and should not be interpreted substantively). Finally, we advise that the output of vetting should ideally be verified against independent data sources, where possible.

7. PC-VET ALGORITHM

Below we present pseudo-code for the PC-VET algorithm. Here $\mathrm{Ne}_G(X_i) \subseteq (X_j)_{j \in V \setminus \{i\}}$ is used to denote the neigobours of node i in an undirected graph G.

Stage I: Obtaining the skeleton

$G \leftarrow$ complete undirected graph on vertex set V

$m \leftarrow 0$

while $\exists j \in V : | \mathrm{Ne}_G(j) | > m$ **do**

for $i \in V$ such that $i - j \in G$ and $i \rightarrow j$, $i \leftarrow j \notin G_0$ **do**

test if $\exists S \subseteq \mathrm{Ne}_G(j) \setminus \{i\}$ such that $|S| = m$ and $X_i \perp\!\!\!\perp X_j \,|\, (X_k)_{k \in S}$

if true **then**

store the set $S(i, j) \leftarrow S$

delete the edge $i - j$ from G

end

end

$m \leftarrow m + 1$

end

Stage II: Orienting the edges

Orient $i - j$ into $i \rightarrow j$ whenever $i - j \in G$ and $i \rightarrow j \in G_0$

for $i, j \in V$ non-adjacent with common neighbour k **do**

if $k \notin S(i, j)$ **then**

if $i \to k - j \in G$ **then**

Replace $i \to k - j$ with $i \to k \leftarrow j$, provided no cycles are created

end

if $i - k \leftarrow j \in G$ **then**

Replace $i - k \leftarrow j$ with $i \to k \leftarrow j$, provided no cycles are created

end

end

end

repeat

R1: orient $j - k$ into $j \to k$ whenever there is an arrow $i \to j$ such that i and k are non-adjacent.

R2: orient $i - j$ into $i \to j$ whenever there is a chain $i \to k \to j$.

R3: orient $i - j$ into $i \to j$ whenever there are two chains $i - k \to j$ and $i - l \to j$ such that k and l are non-adjacent.

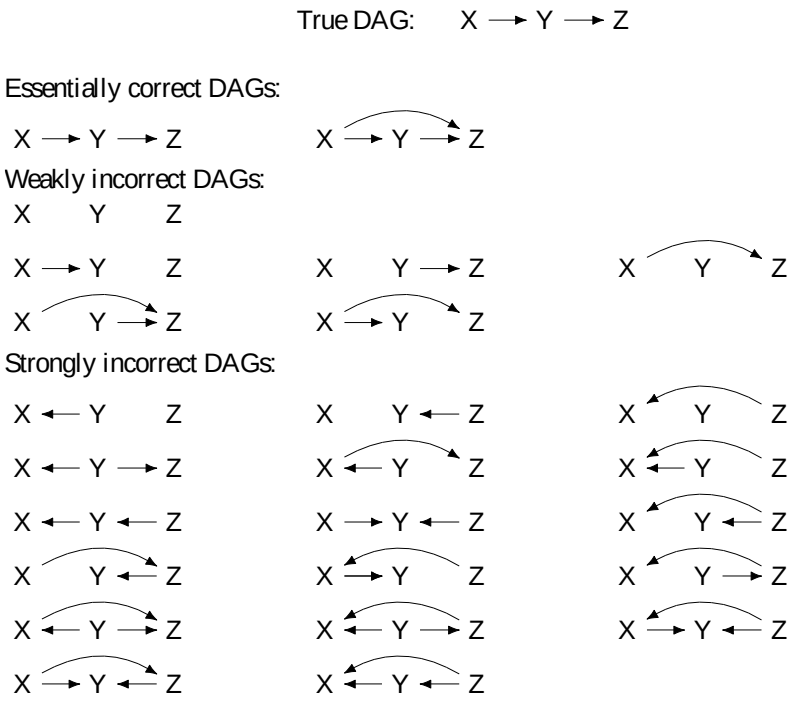
R4: orient $i - j$ into $i \to j$ whenever there are two chains $i - k \to l$ and $i \to l \to j$ such that j and k are non-adjacent.

until no further application of R1-R4 is possible

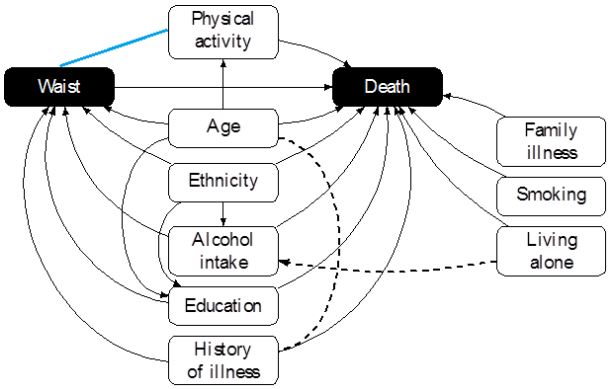
return a partially directed acyclic graph $P = G$.

Note that the output of PC-VET is dependent on the order of the operations that are carried out in Stage I. This particular feature is also present in the standard PC algorithm, where it is known that this order-dependence is not problematic in low dimensional settings. The standard PC algorithm can be made order-independent following the methods in (18) and these can also be applied to the PC-VET algorithm. However, this was beyond the scope of the present research.

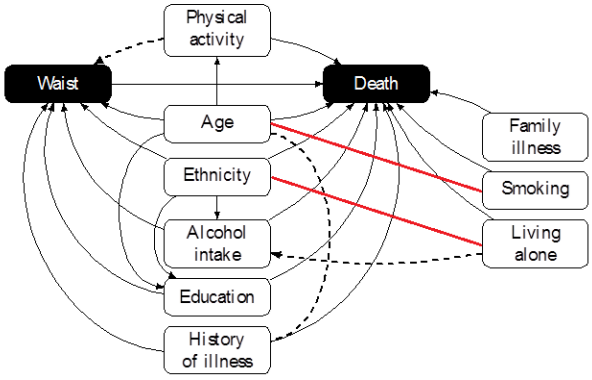
8. ADDITIONAL FIGURES



eFigure 1: Here we enumerate all essentially correct, weakly incorrect, and strongly incorrect DAGs G_0 that could be proposed by an expert, where correctness is defined relative to the true DAG G_{true} of the form $X \to Y \to Z$.

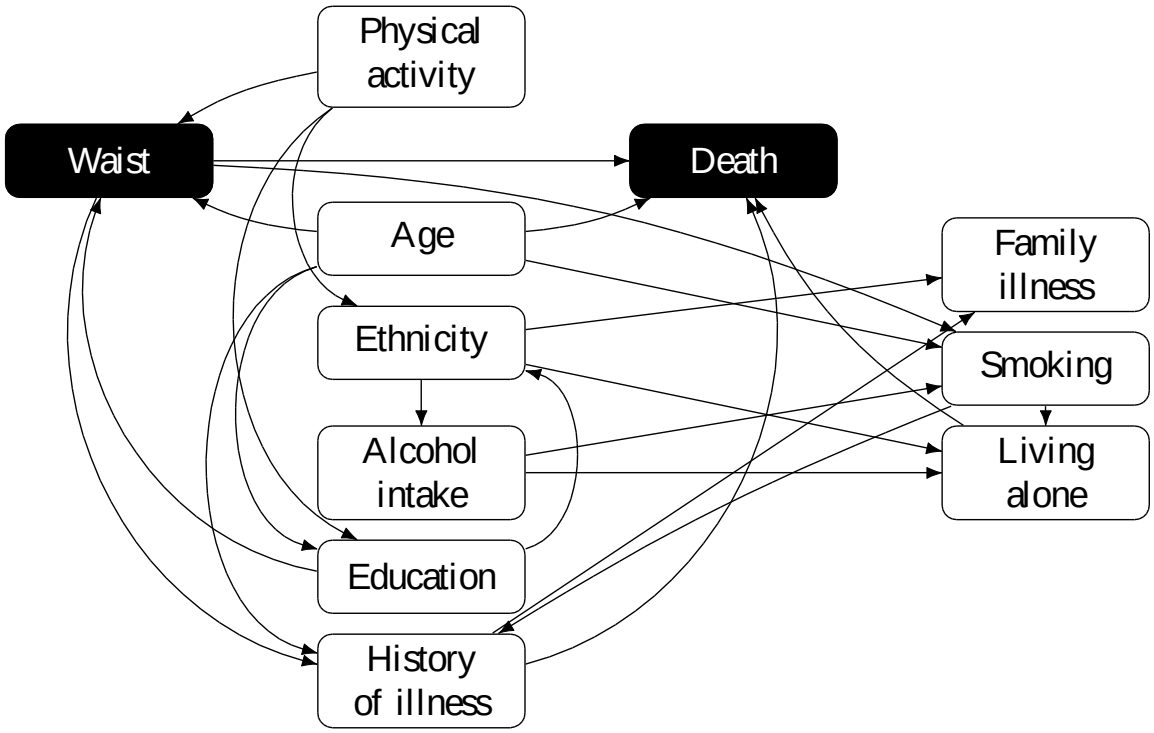


(a) $\alpha = 0.025$



(b) $\alpha = 0.1$

eFigure 2: Here we investigate sensitivity of the results for the Melbourne Collaborative Cohort Study, reported in Figure 3 of the main text, to the choice of significance threshold $\alpha = 0.05$. (a) For $\alpha = 0.025$ we no longer have statistical power to detect an edge from Physical Activity to Waist (blue line). (b) For $\alpha = 0.1$ we report, in addition, undirected edges between Age and Smoking, and between Ethnicity and Living Alone (red lines). Context (time ordering) implies that these edges operate as Age → Smoking and Ethnicity → Living Alone.



eFigure 3: The directed acyclic graph returned by applying the unrestricted PC algorithm to the Melbourne Collaborative Cohort Study data. This results in many edges that do not appear to have reasonable subject-matter interpretations, for example the edges from Physical activity and Education to Ethnicity do not appear to have reasonable interpretations.

ACKNOWLEDGEMENT

The authors are extremely grateful to Marloes Maathuis and Emilija Perkovic for bringing to our attention the need for assumption (A3). The red text represents a correction to the electronic supplement that was originally published.

REFERENCES

1. Brookhart MA, Schneeweiss S, Rothman KJ, et al. Variable selection for propensity score models. *Am J Epidemiol.* 2006;163(12):1149-56.
2. Pearl J. On a class of bias-amplifying variables that endanger effect estimates. *Proc. Twenth-sixth Conference on Uncertainty in Artificial Intelligence (UAI 2010)*. Corvallis, OR: Association for Uncertainty in Artifical Intelligence. 2010;425-32.
3. Bühlmann P, Rütimann P, Kalisch M. Controlling false positive selections in high-dimensional regression and causal inference. *Stat Methods Med Res.* 2013;22(5):466-92.
4. Bühlmann P, Mandozzi J. High-dimensional variable screening and bias in subsequent inference, with an empirical comparison. *Comput Stat.* 2014;29(3-4):407-30.
5. Bühlmann P, Peters J, Ernest J. CAM: Causal additive models, high-dimensional order search and penalized regression. *Ann Stat.* 2014;42:2526-56.
6. Spirtes P, Glymour CN, Scheines R. *Causation, prediction, and search*. MIT Press; 2000.
7. Koller D, Friedman N. *Probabilistic graphical models: principles and techniques*. MIT Press; 2009.
8. Chickering DM. A transformational characterization of equivalent Bayesian network structures. *Proc. Eleventh Conference on Uncertainty in Artificial Intelligence (UAI '95)*. 97-98.
9. Verma TS, Pearl J. Equivalence and synthesis of causal models. *Proc. Sixth Conference on Uncertainty in Artificial Intelligence (UAI '90)*. 220-227.
10. Scheines R, Spirtes P, Glymour C, et al. The TETRAD Project: Constraint based aids to causal model specification. *Multivariate Behav Res.* 1998;33(1):65-117.
11. Kalisch M, Mächler M, Colombo D, et al. Causal inference using graphical models with the R package pcalg. *J Stat Softw.* 2012;47(11):1-26.
12. Li J, Wang ZJ. Controlling the false discovery rate of the association/causality structure learned with the PC algorithm. *J Mach Learn Res*. 2009;10(Feb):475-514
13. Maathuis MH, Kalisch M, Buhlmann P. Estimating high-dimensional intervention effects from observational data. *Ann. Stat.* 2009;37:3133–64.
14. Gillispie SB, Perlman MD. The size distribution for Markov equivalence classes of acyclic digraph models. *Artificial Intelligence.* 2002;141(1–2):137-55.
15. Pearl J. *Causality*. 2nd ed. Cambridge University Press; 2009.
16. Meek C. Causal inference and causal explanation with background knowledge. *Proc. Eleventh Conference on Uncertainty in Artificial Intelligence (UAI '95)*. 87-98.
17. Uhler C, Raskutti G, Buhlmann P, et al. Geometry of the faithfulness assumption in causal inference. *Ann Stat.* 2013;41(2):436-63.
18. Colombo D, Maathuis MH. Order-independent constraint-based causal structure learning. *J Mach Learn Res*. 2014;15(Nov):3921-62.