causaleffect: An R package for identification of causal effects

Santtu Tikka and Juha Karvanen
Department of Mathematics and Statistics, University of Jyväskylä

ABSTRACT

Do-calculus is concerned with estimating the interventional distribution of an action from the observed joint probability distribution of the variables in a given causal structure.

All identifiable causal effects can be derived using the rules of do-calculus, but the rules themselves do not give any direct indication whether the effect in question is identifiable or not. An algorithm for identifying joint interventional distributions in causal models which contain unobserved variables and induce directed acyclic graphs was constructed by Shpitser and Pearl [2]. This algorithm, known as the ID algorithm, can be seen as a repeated application of the rules of do-calculus and known properties of probabilities, and it ultimately either derives an expression for the causal distribution, or fails to identify the effect, in which case the effect is non-identifiable.

The *causaleffect* package implements the ID algorithm and its variations for other similar problems in the field of causality such as transportability, which is the problem of identifying causal effects when data from multiple domains are available. In some cases experimental data on some auxiliary variables can also be accessed.

INTRODUCTION

Causal models can be depicted by using graphs where vertices correspond to variables and edges correspond to functional relationships. Out focus lies in so called semi-Markovian causal models that satisfy additional assumptions. In these models we assume the latent variables in our model to be mutually independent and that each latent variable is a parent of exactly two observed variables, denoted by a bidirected arc. Furthermore, the induced graph of the model is assumed to be acyclic, rendering the induced graph a directed acyclic graph (DAG). The main difference between causal models and other graphical models is the notion of intervention. An intervention on a variable in a causal model assigns the variable a constant value instead of letting it be determined by its parents. In the graph this corresponds to removing all incoming edges to the vertex. An intervention on a variable X is denoted by do($X = x$), also known as the do-operator The intervention generates an interventional distribution, also known as the causal effect. The question of interest is know whether the causal effect is identifiable, that is whether it can be uniquely computed from the probability distribution of the observed variables.

DO-CALCULUS

Our primary tool to operate on interventional distributions is do-calculus, which consists of three inference rules [1]. If the causal effect is identifiable, it can be shown that do-calculus can be used to derive an expression for it that is free of the do-operator. The rules themselves however do not give any clear indication as to which rules should be applied in which order to reach the desired expression. This makes do-calculus difficult to apply directly in practice. Fortunately, efficient algorithmic solutions such as presented in [2] are readily available and implemented in the R package *causaleffect*. Details of the implementation and usage of the package are thoroughly described in [3]. The package is also available on CRAN (https://cran.r-project.org/web/packages/causaleffect/index.htm).

DEFINING GRAPHS

To analyze identifiability of causal effects, we must first be able to construct graphs of interest. We use the *igraph* package to define and manipulate graphs.
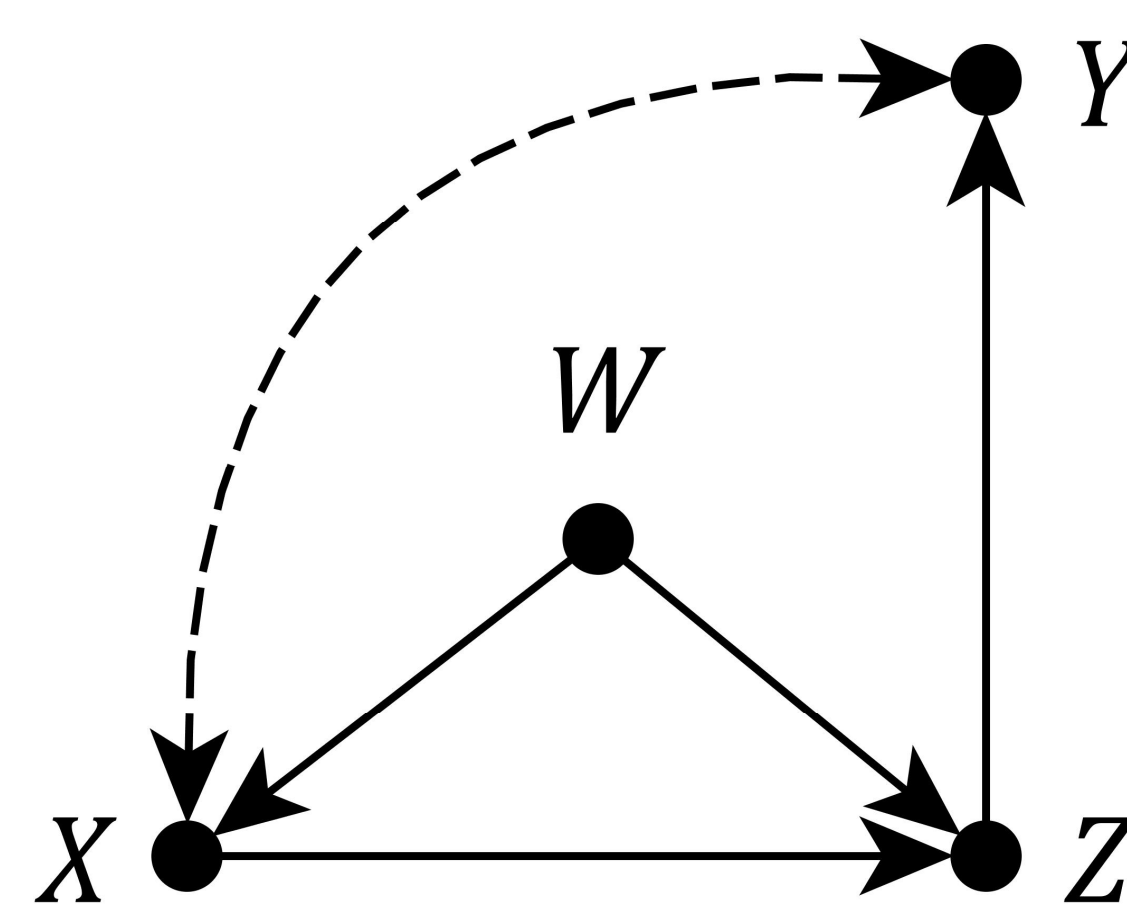

Figure 1: A graph depicting a causal model that consists of four variables: **X, Y,** *Z and* **W**.

We define the graph of Figure 1 in R as follows:

```
library(igraph)

# Defining the vertices and edges
G <- graph.formula(w -+ x, w -+ z, x -+ z, z -+ y,
  x -+ y, y -+ x, simplify = FALSE)

# Setting an edge attribute to distinguish
# the bidirected edge from regular edges
G <- set.edge.attribute(graph = G,
  name = "description", index = 5:6,
  value = "U")
```

We note that bidirected edges are defined in the package using two directed edges, which are then given a special description attribute with the value U (unobserved) which sets them apart from the other directed edges. The option *simplify = FALSE* allows the same edge to appear twice before setting the attribute.

USING THE CAUSALEFFECT PACKAGE

After having defined the graph, we can now attempt to identify causal effects of interest. The primary function of the *causaleffect* package is *causal.effect* which takes as input the set of outcomes Y, the set of interventions X and the graph depicting the causal model G. The default output of the function is a string for the expression of the causal effect in LaTeX syntax in the case that the effect is identifiable. If the effect is not identifiable, a problematic graph structure known as a hedge is returned instead. If a string is not desired, a list-type object depicting the interventional probability distribution can also be returned as output for manual parsing purposes. There are also additional arguments to control the form of the output among others. In many situations, the output is unnecessarily complicated even when a simpler expression for the causal effect exists. Simplification methods such as described in [4] have been developed and implemented in the *causaleffect* package.

A more detailed description on the functionality can be found in the package documentation on CRAN. We identify the causal effect of X on Y in the graph of Figure 1 in R as follows:

```
library(causaleffect)

# Deriving an expression for the causal effect of
# X on Y
ce <- causal.effect(y = "y", x = "x", G = G)

# Output the expression for the effect
cat(ce)
```

We find that the causal effect is identifiable and the output of the last line as interpreted in LaTeX is:

$$\sum_{w,z} P(w)P(z|w,x)\left(\sum_{x'} P(y|w,x',z)P(x'|w)\right)$$

We are not limited to singletons when setting the outcomes and interventions. We can also use the package for example to identify the causal effect of X and W on Z and Y in the graph of Figure 1:

```
# Deriving an expression for the causal effect of
# X and W on Z and Y
ce2 <- causal.effect(y = c("z", "y"),
  x = c("x", "w"), G = G)

# Output the expression for the effect
cat(ce2)
```

The causal effect in question is again identifiable and the output of the last line is:

$$P(z|w,x)\left(\sum_{x'} P(y|w,x',z)P(x'|w)\right)$$

The package can also be used to derive expressions for conditional causal effects and causal effects with auxiliary information, i.e. effects where experiments on a set separate from X and Y are available.

CONCLUSIONS

• Manual derivations of causal effects using do-calculus are difficult and unavailable for automated processing

• Methods for causal inference such as identification of causal effects using the ID algorithm are implemented in the R package *causaleffect*, which is available on CRAN

• Graphs depicting causal models can be easily constructed using the R package *igraph*

REFERENCES

[1] Pearl, J., Causality: Models, Reasoning and Inference, Cambridge Univeristy Press, New York, 2nd Edition, 2009.
[2] Shpitser, I. and Pearl, J., Identification of Joint Interventional Distributions in Recursive Semi-Markovian Causal Models,
    In *Proceedings of the 21st National Conference on Artificial Intelligence*, pp. 1219-1226, 2006.
[3] Tikka, S. and Karvanen, J., Identifying Causal Effects with the R-package causaleffect,
    Journal of Statistical Software, 76(12):1–30, 2017.
[4] Tikka, S. and Karvanen, J., Simplifying Probabilistic Expressions in Causal Inference,
    Journal of Machine Learning Research, 18(36):1–30, 2017, http://jmlr.org/papers/v18/16-166.html.

UNIVERSITY OF JYVÄSKYLÄ