CausalQueries: Make, Update, and Query Causal Models

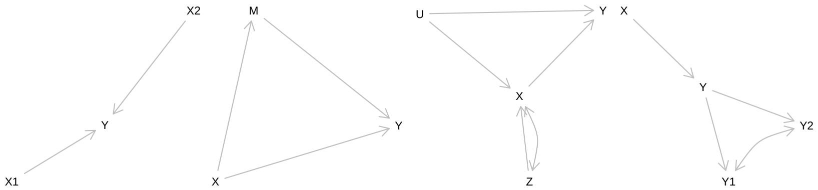
CausalQueries: Make, Update, and Query Binary Causal Models

Version:	0.0.3
Published:	2020-06-03
Author:	Clara Bicalho [ctb], Jasper Cooper [ctb], Macartan Humphreys [aut], Alan Jacobs [aut], Merlin Heidemanns [ctb], Lily Medina [aut, cre], Julio Solis [ctb], Georgiy Syunyaev [ctb]
Maintainer:	Lily Medina <lilymiru at gmail.com>
License:	MIT + file LICENSE

Outline

- Motivation
- Models and queries
- The package and what it does
- Some applications
  - Justifying case level inference (a "hoop" test)
  - Nothing from nothing
  - Case selection
  - Experimental and observational data
  - Censoring and Colliders: Police stopping
  - Democracy and Inequality
- Limitations and questions

Overview



`CausalQueries` make it easy to:

- `make_model`: **Define** classes of causal models given causal structures like these: figure out the implied parameter set
- `update_model`: **Update** given data and `stan`
- `query_model`: Pose any case- or pop-level causal **query**

Useful for:

- Learning about properties of models, properties of designs, including for quantities that are not identified
- Grounding qualitative inferences
- General estimation (without estimators)

Motivation

Background Motivation: Qualitative and Quantitative Inference

In Humphreys and Jacobs (2016) we presented a formalization of a Bayesian approach to “mixed methods” inference:

- Combined:
  - data on cross case variation
  - with within case “process” data
- $\rightarrow$ Single (“integrated”) conclusion

For instance: *Did this swamp cause malaria in this village?*

Quantitative strategy:

- Examine a large set of villages with and without swamps and with and without malaria
- Worry about confounding and all that
- Form a general view for this population about the chances that swamps cause malaria
- Apply the population inference to this case?

Qualitative strategy:

- Go to the village and gather within-case information
- Do mosquitoes breed in the swamp?
  - if I observe no breeding I conclude the swamp was not the cause;
  - if I observe breeding, I update towards believing the swamp caused malaria

Integrated

Procedure:

1. "Qual" part: For estimand, τ, specify $\Pr(K|\tau, X, Y)$
2. "Quant" part: We also have $\Pr(X, Y|\tau)$
3. Jointly update on τ given X, Y, K

Note:

- Critical thing for the qualitative part is that the inference depends on applying a *theory* of how the world works to a case at hand.
- *Inferences only make sense if you believe the theory*

The worry

We worried though that:

- there was no justification for *why* “clues” would be informative
- little or nothing on which to ground beliefs about their probative value
- not much process to the process tracing

Generalization

You can address all this using causal models that connect X, M, Y together in some way or another. Then learn about the whole model and draw inferences.

Then generalize this idea to:

1. Arbitrary causal models
2. Arbitrary causal queries
3. Partial data on different types of nodes

- Basic structures described by Pearl (2009) give conceptual basis and language to articulate possibly complex estimands
- Flexible `stan` structure used for estimation

Models

X, Y, Model

Say we believe, for binary nodes:

$$X \rightarrow Y$$

i.e. X possibly affects Y *and* X is as-if randomly assigned

Then:

- There are two possible "nodal types" for X: $\theta_X \in \{\theta_0, \theta_1\}$
- There are *four* "nodal types" for Y: four ways that Y might react to X for a unit:

$$\theta_Y \in \{\theta^Y_{00}, \theta^Y_{10}, \theta^Y_{01}, \theta^Y_{11}\}$$

Here θ^Y_{ij} means: $Y(0) = i, Y(1) = j$

A "nodal type" summarizes the values that a node takes for each of its parents' values.

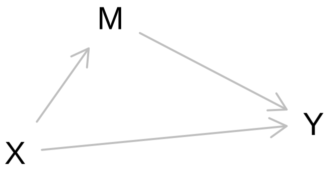
X, Y, Model

- Understanding the causal relations at the case-level means learning about $\theta = (\theta_X, \theta_Y)$, the case-level causal type
- Understanding the causal relations at a population level means learning about the distribution of θ_X, θ_Y.
  - We let λ^V denote the (categorical) distribution over θ^V
  - For instance: $\lambda^Y_{01} = \Pr(\theta^Y = \theta^Y_{01})$
  - We might have priors over λ^Y represented by a Dirichlet distribution (for example)

So, our challenge:

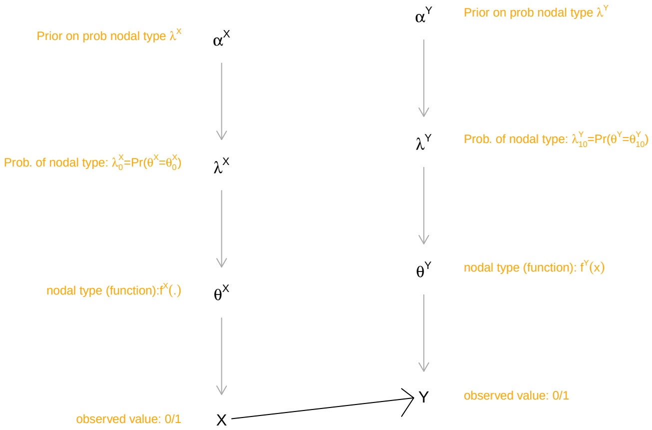
- can we learn about θ given data on X, Y?
- can we learn about λ given data on X, Y?

This generalizes like this



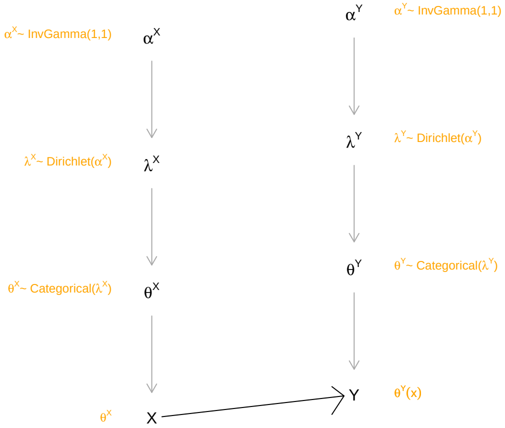
Model	$X \rightarrow Y$	$X \rightarrow M \rightarrow Y \leftarrow X$
Nodal types:	θ^X_i (2 types) θ^Y_{ij} (4 types)	θ^X_i (2 types) θ^M_{ij} (4 types) θ^Y_{hijk} (16 types)
Causal types:	$2 \times 4 = 8$	$2 \times 4 \times 16 = 128$
Priors on λ:	Beta, 4-Dirichlet	Beta, 4-Dirichlet, 16-Dirichlet

As a picture

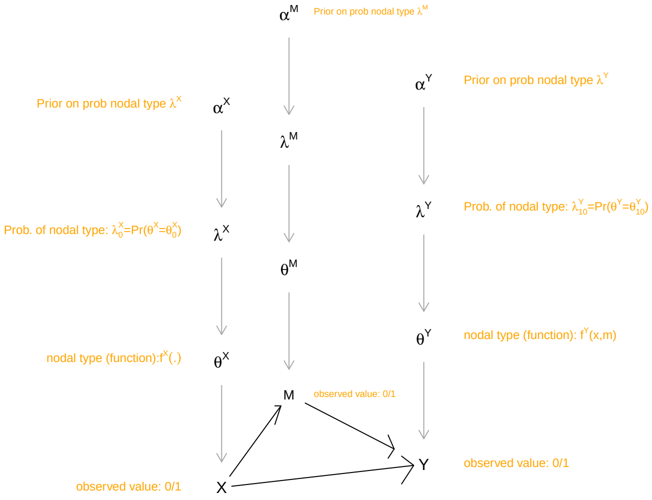


Prior on prob nodal type λ^X
α^X
Prob. of nodal type: $\lambda^X_0=Pr(\theta^X=\theta^X_0)$
λ^X
nodal type (function):$f^X(.)$
θ^X
observed value: 0/1
X
α^Y
Prior on prob nodal type λ^Y
λ^Y
Prob. of nodal type: $\lambda^Y_{10}=Pr(\theta^Y=\theta^Y_{10})$
θ^Y
nodal type (function): $f^Y(x)$
Y
observed value: 0/1

As a picture: Distributions

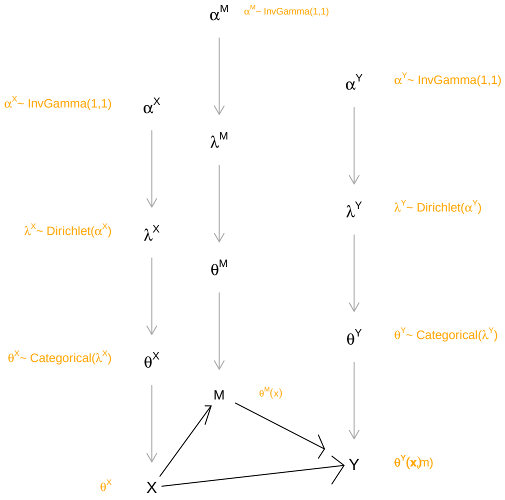


As a picture

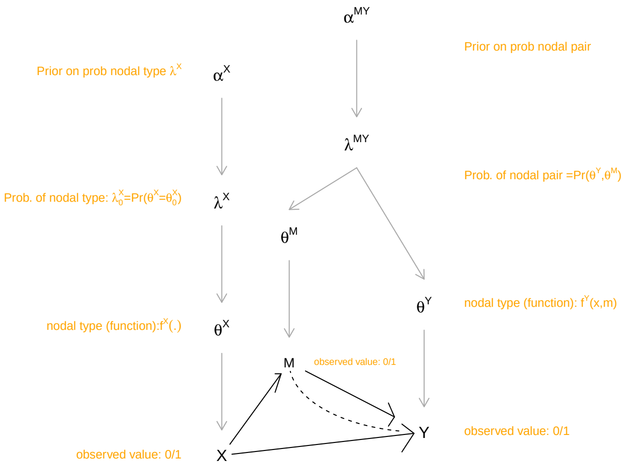


α^M
Prior on prob nodal type λ^M
α^Y
Prior on prob nodal type λ^Y
Prior on prob nodal type λ^X
α^X
λ^M
λ^Y
Prob. of nodal type: λ^Y_{10}=Pr(θ^Y=θ^Y_{10})
Prob. of nodal type: λ^X_0=Pr(θ^X=θ^X_0)
λ^X
θ^M
θ^Y
nodal type (function): f^Y(x,m)
nodal type (function):f^X(.)
θ^X
M
observed value: 0/1
Y
observed value: 0/1
observed value: 0/1
X

As a picture: Distributions



As a picture: with confounding



From data to inference

So the hard work is figuring out for any data profile such as $D = (X = 1, M = \text{NA}, Y = 0)$ and parameter vector $\lambda = (\lambda^X, \lambda^M, \lambda^Y)$, what is:

$$\Pr(D|\lambda)$$

We proceed by:

- figuring out the **data type** associated with each *causal type*
- figuring out the probability of each *causal type*
- summing up the probabilities of all *causal types* that imply this **data type**

For instance:

- $\lambda^X_0 = .5, \lambda^X_1 = .5, \lambda^Y_{00} = .25, \lambda^Y_{10} = 0, \lambda^Y_{01} = .5, \lambda^Y_{11} = .25$
- The two causal types that yield $X = Y = 1$ arise with probability $\lambda^X_1 \lambda^Y_{01}$ and $\lambda^X_1 \lambda^Y_{11}$
- $\Pr(X = Y = 1|\lambda) = 3/8$

From data to inference

Say $X \to M \to Y$

Case 1: $X = M = Y = 1$:

$$\Pr(X = Y = M = 1|\lambda) = \lambda^X_1(\lambda^M_{01} + \lambda^M_{11})(\lambda^Y_{01} + \lambda^Y_{11})$$

Case 2: Data on X and Y only. We observe $X = Y = 1$:

$$\Pr(X = Y = 1|\lambda) = \sum_{m=0}^{1} \Pr(X = Y = 1, M = m|\lambda)$$

Joint probability: product of these probabilities (note between case independence assumption).

3 Important Wrinkles

1. **Nodal independence assumptions** may not be tenable: we need to allow for θ^X and θ^Y to have a joint distribution
2. **Data missingness**: in typical "mixed methods" cases we have data on some nodes for some units and other nodes for other units. For instance data on X, Y for all; data on K for some
3. **Restrictions**: researchers might want to place restrictions on possible functional relations, thus reducing the parameter space (for instance: "no police violence unless individuals are stopped"; "no defiers")

`CausalQueries` handles all of these

`CausalQueries` functions and procedures

Core functions

Functions to define models:

- `make_model`, `set_priors`, `set_restrictions`, `set_confounds`, `set_parameters`

Updating

- `update_model`

Inference

- `query_model`

Functions to inspect and use models:

- `plot`, `make_data`

Illustrations

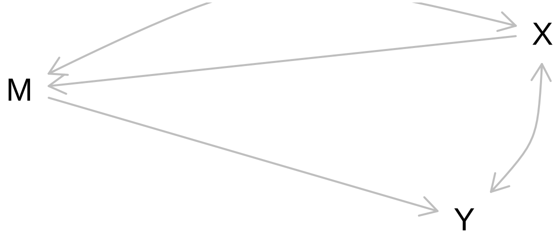
A hoop test

A hoop test

In qualitative inference a "hoop" test is a search for an observation that, if absent, greatly reduces confidence in a theory.

Define a model with X causing Y through M but with confounding.

```
model <- make_model("X -> M -> Y; X <-> Y; X<->M")

plot(model)
```



Simulate performance

We imagine a real world in which there are in fact monotonic effects and no confounding, though this is not known. (The data suggests a process in which X is necessary for M and M sufficient for Y)

```
data <- fabricate(N = 1000,
                  X = rep(0:1, N/2),
                  M = runif(N) < .7*X,
                  Y = runif(N) < (.3 + .7*M))
```

Then update:

```
model <- update_model(model, data)
```

Results

```
posterior_estimands <-

  query_model(
    model,
    queries = "c(Y[X=1] > Y[X=0])",
    given   = c("X==1 & Y==1",
                "X==1 & Y==1 & M==1",
                "X==1 & Y==1 & M==0"),
    using = "posteriors")
```

Table 2: Learning

Given	truth	prior	post.mean	sd
X==1 & Y==1	0.62	0.268	0.313	0.183
X==1 & Y==1 & M==1	0.70	0.250	0.354	0.206
X==1 & Y==1 & M==0	0.00	0.250	0.005	0.006

Hoop illustration implications

We see that we can find M informative about whether X caused Y in a case specifically when we see $M = 0$.

This is striking because:

- We have a non experimental design in which the effect of X on Y is not identified
- The probability that Y is caused by X is also not identified.
- We placed no restrictions on functional forms except those implied by the DAG (notably X works via M)
- We did not build "hoopiness" into the model, we learned about it from the model

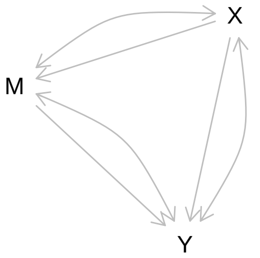
(The intuition is that if we observe $M = 0$ we will infer that X did not cause $M = 0$–despite confounding, since *conditional* on $X = 1$ we see no evidence of negative effects— but then M did not cause $Y = 1$ either)

Nothing from nothing

Nothing from nothing

- Say you had access to large amounts of observational data on X, Y and M
- You know the temporal order of X, Y, M only.
- Can you figure out whether M is informative for X causes Y from this data?
- Assume a world, like above, where *in fact* $X \rightarrow M \rightarrow Y$, all effects strong (80%, 80%).

```
agnostic_XMY <- make_model("X ->  M; M ->  Y; X -> Y;
                            X <-> M; M <-> Y; X <-> Y")
```



Updating gives:

Table 3: Conditional inferences from an updated agnostic model given a true model in which X causes M and M causes Y

Query	Given	Using	mean	sd
Q 1	-	posteriors	0.4	0.09
Q 1	M==0	posteriors	0.4	0.12
Q 1	M==1	posteriors	0.4	0.13

This negative result holds even if we can exclude $X \rightarrow Y$

This example illustrates the Cartwight idea of no causes in, no causes out.

Case selection

Case selection

In qualitative methods there is *lots* of advice on case selection:

- on the regression line!
- off the line!
- select on X! select on Y! $X = Y = 1$ cases only please! never $X = Y = 0$!
- most likely case, least likely case, representative cases, random,

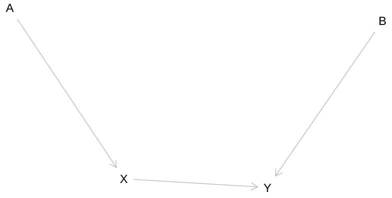
Remarkably, we think, much of this advice:

- Does not stipulate the estimand
- Does not stipulate the inferential strategy once you have a case
- Is not sensitive to the structure of the causal model

Case selection

Basic ideas from causal models give guidance about what you want to look for:

- If new within-case evidence is *d*-separated from your query then you won't learn anything from the within case evidence
- Here: if you already know X and Y, then B might be informative for the effect of X on Y but A won't be



- If you can learn about A in one case and B in another, choose the other

General procedure

- Make your model (updated with data)
- For a given strategy, figure out the probability of each possible data realization
- Figure out what your posterior variance would be if that realization arose
- Calculated expected posterior variance
- Select strategy with lowest expected posterior variance

General procedure

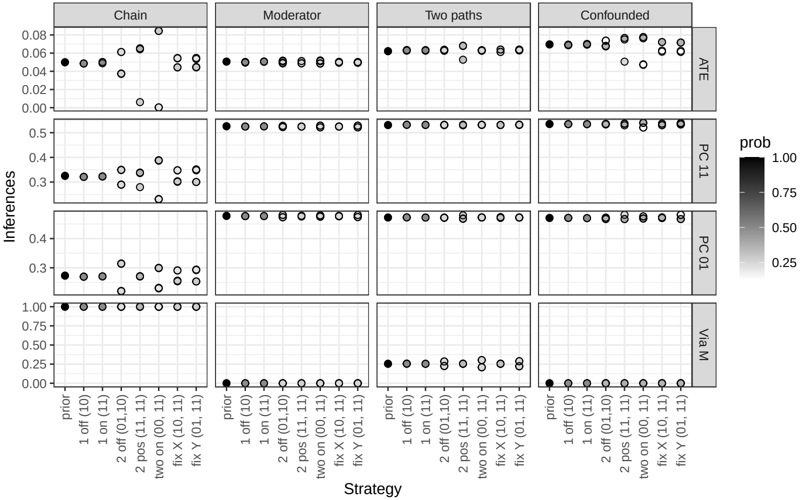


Figure 1: Inferences given observations

Variance reduction

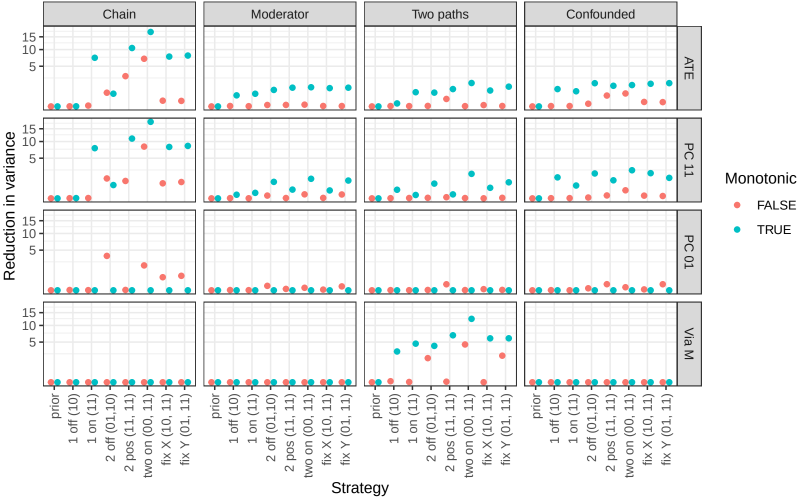


Figure 2: Reduction in variance on ATE given strategies

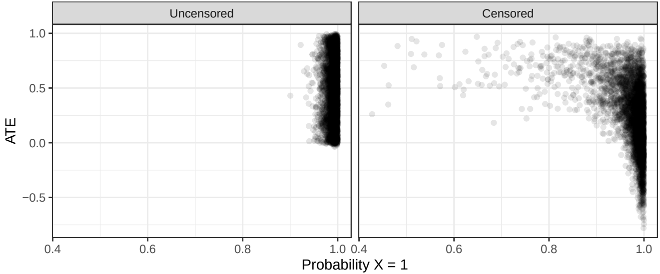
Missingness

Missingness

- Easy enough: you gather data on X only for cases in which $Y = 1$. The likelihood principle sees us through.
- More difficult: you only observe cases as a function of their realizations: you only see data for cases where $Y = 1$

Imagine:

You know $X \to Y$, you observe 100 cases all with $X = 1$, $Y = 1$



Inferences depend on knowledge of censoring

- Negative effects still entertained
- Posterior correlation between beliefs about the probability of X and beliefs about effects

Data	X1	ATE	sd(ATE)
Uncensored	0.99	0.48	0.28
Censored	0.97	0.24	0.32

Police stopping

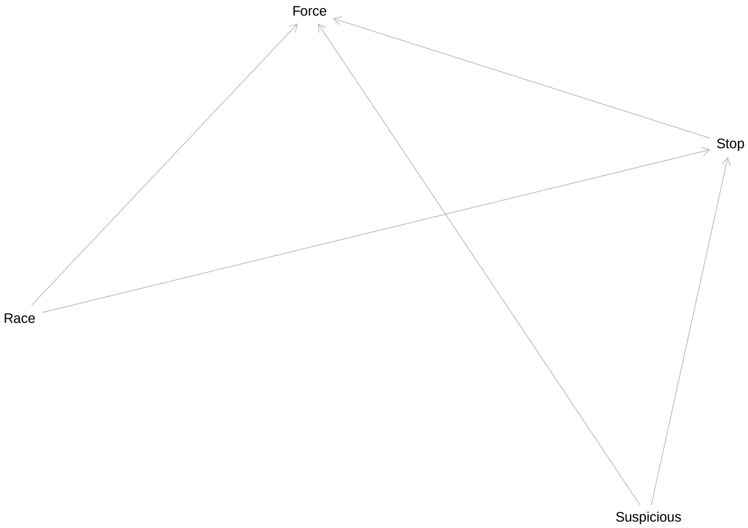
Model

Five "qualitative" restrictions in addition to causal structure:

```
discrimination <-

 make_model("Race -> Stop -> Force <- Race;
             Stop <- Suspicious -> Force") %>%

 set_restrictions("Force[Stop=0] == 1") %>%

 set_restrictions("Force[Race=0] > Force[Race=1]") %>%

 set_restrictions("Stop[Race=0] > Stop[Race=1]") %>%

 set_restrictions("Stop[Suspicious=0] > Stop[Suspicious=1]"

 set_restrictions("Force[Suspicious=0] > Force[Suspicious=

 set_priors(node = "Race", alpha = c(10, 10))
```

Model

```
plot(discrimination)
```



Model

Imagine in truth that there are strong complementarities between race and suspiciousness for stopping probability.

```
discrimination <- discrimination %>%

 set_parameters(param_type = define,
                node = "Stop",
                label = "0001",
                parameters = .75)
```

Stopping estimands

```
discrimination %>%
  query_model(
   queries = list(
    ATE  ="Force[Race=1]-Force[Race=0]",
    ATE_M="Force[Race=1]-Force[Race=0]",
    CDE_M="Force[Race=1, Stop=1]-Force[Race=0, Stop=1]"),
  given = c(TRUE, "Stop==1", "Stop==1"),
  using = "parameters") %>%
  kable(caption = "estimands", digits = 2)
```

Table 5: estimands

Query	Given	Using	Case.estimand	mean
ATE	-	parameters	FALSE	0.39
ATE_M	Stop==1	parameters	FALSE	0.65
CDE_M	Stop==1	parameters	FALSE	0.33

Regression results

In this case regression results *over*estimate the CDE (because race and suspiciousness are strong complements for stops)

```
df <- discrimination %>% make_data(n = 100000) %>%
  select(-Suspicious) %>%
  filter(Stop == 1)
```

Table 6: Regression results

term	estimate	std.error
(Intercept)	0.42	0.01
Race	0.37	0.01

Stopping estimands

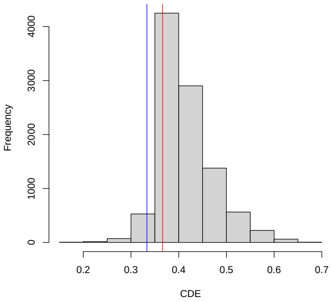


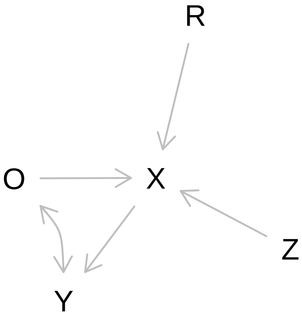
Figure 3: OLS estimates (red) are too large (as are values to the right of it). Bayesian estimates are also too large but the posteriors exhibit a wide distribution despite large data.

- We have an identification problem. We can (a) ignore it (b) impose more assumptions to bound estimates (c) calculate directly what the data imply given priors.

A model with mixed experimental and observational data

The model

```
model <-

 make_model("R -> X -> Y; O -> X <- Z; O <-> Y") %>%

 set_restrictions("(X[R=1, Z=0]!=0)|(X[R=1, Z=1]!=1)|
                    (X[R=0, O=0]!=0)|(X[R=0, O=1]!=1)")

plot(model)
```



Estimates

The `CausalQueries` estimates are:

Table 7: Effects of X conditional on X for units that were randomly assigned or not. Effects of X do not depend on X in the experimental group, but they do in the observational group becuase of seld selection.

Query	Given	mean	sd
ATE	R==1 & X==0	0.20	0.03
ATE	R==1 & X==1	0.20	0.03
ATE	R==0 & X==0	-0.18	0.03
ATE	R==0 & X==1	0.59	0.05

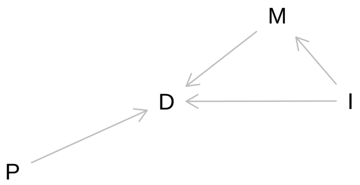
Democracy and Inequality

Democracy and Inequality

- Scholars have provided theoretical arguments linking inequality, `I`, to democratization `D` (e.g. Acemoglu and Robinson, Boix)
- Qualitative scholars have examined cases with $I = D = 1$ to see whether there is evidence of mobilization `M` or other external pressures to democratize `P` (Haggard and Kaufman)
- Can we combine the theory with case data to draw inferences about whether I caused (or prevented) D in a given case

Model

```
make_model("I -> M -> D <- P; I -> D") %>%
  set_restrictions(c(
       "(M[I=1] < M[I=0])",
       "(D[I=1] > D[I=0]) | (D[M=1] < D[M=0]) |
        (D[P=1] < D[P=0])")) %>%
  plot()
```



Inferences given restricted model

Table 8: Inferences from restricted model

Query	Given	Using	mean	sd
Q 1	-	posteriors	-0.23	0.05
Q 1	I==1 & D==1	posteriors	0.06	0.04
Q 1	I==1 & D==1 & M==1	posteriors	0.09	0.06

M is certainly informative given a model that builds in theoretical constraints.

Inferences given unrestricted model

Table 9: Inferences from restricted model

Query	Given	Using	mean	sd
Q 1	-	priors	0.00	0.03
Q 1	-	posteriors	-0.01	0.02
Q 1	I==1 & D==1	priors	0.50	0.03
Q 1	I==1 & D==1	posteriors	0.50	0.02
Q 1	I==1 & D==1 & M==1	priors	0.50	0.03
Q 1	I==1 & D==1 & M==1	posteriors	0.50	0.03

But these theoretical constraints in fact do essentially all of the work: without them the data patterns are not enough to justify these inferences.

Uses, Limitations and Questions

Uses: Many implications

You could use this kind of model, for example, if you have both X, Y data on many cases and qualitative data, M, (e.g. process data) on some.

With a model in hand you can ask:

- *how many* cases should I look at?
- what types of *cases* should I look at?
- what *evidence* should I look at

Uses

Richer questions become possible. . .

- e.g. identification not needed for progress. Estimands frequently not identified, but bounded, as for example with the "causes of effects" estimand or the "average treatment effect" given confounding.

Take the good not the bad:

- Colliders confuse classical estimators. But they still contain information. You can use them.

Weakesses: Complexity

- Things get terribly slow once models get moderately complex
- A node j with k_j parents has $\prod_j 2^{\left(2^{k_j}\right)}$ values that uniquely determine what data will be observed for a type under all possible interventions

Weakesses: Scope

- Binary nodes: this is not a deep limitation but complexity issues become much more severe once we move away from binary
- Under construction: Hierarchy. Current package has no hierarchical components, so e.g. one cannot introduce random effects or similar directly. However we have implemented hierarchical models of this form.
- Under construction: Missingness. Current package has controlled missingness for cases when likelihood principle can be invoked but not otherwise.

Questions

Searching for a theorem:

- A nice feature is: we don't have to develop estimators. Every analysis follows the same procedure: `make_model`, `update_model`, `query_model`
- Is it the case that if there is a consistent estimator for τ (and so τ is identified) then are we guaranteed that posterior uncertainty on queries following model updating collapses as n grows?
- Is there a result on the converse: a consistent estimator might not exist even if τ is identified, but should posterior variance nevertheless collapse?

Fin

CausalQueries: Make, Update, and Query Binary Causal Models

Version:	0.0.3
Published:	2020-06-03
Author:	Clara Bicalho [ctb], Jasper Cooper [ctb], Macartan Humphreys [aut], Alan Jacobs [aut], Merlin Heidemanns [ctb], Lily Medina [aut, cre], Julio Solis [ctb], Georgiy Syunyaev [ctb]
Maintainer:	Lily Medina <lilymiru at gmail.com>
License:	MIT + file LICENSE

Figure 4: Credits