Stability of causal inference

Leonard J. Schulman & Piyush Srivastava

California Institute of Technology

UAI 2016
Causal identification: Experimental intervention

Observation

Genetic factors

Smoking \( X \) → Lung disease \( Y \)

Experimental intervention on \( X \)
Causal identification: Experimental intervention

Observation

Intervention on X

Genetic factors

Smoking $\rightarrow$ Lung disease

Genetic factors

Smoking $\rightarrow$ Lung disease
Directed graphical models

- A directed acyclic graph $G = (V, E)$ whose nodes are random variables.
- Absent edges represent conditional independence assumptions.

$$P(X, Y, Z) = P(X)P(Y|X)P(Z|X, Y)$$
$$= P(X)P(Y)P(Z|X, Y), \text{ due to model constraints}$$
Directed graphical models

- A directed acyclic graph $G = (V, E)$ whose nodes are random variables
- Absent edges represent conditional independence assumptions

\[
P(X, Y, Z) = P(X)P(Y|X)P(Z|X, Y) \\
= P(X)P(Y|X)P(Z|Y), \text{ due to model constraints}
\]
Semi-Markovian models

- A Markovian model with some nodes hidden
- Hidden nodes have no parents

\[
P(X, Y, Z) := \sum_u P(U = u)P(X|U = u)P(Z|X)P(Y|X, U = u)
\]
Semi-Markovian models

- A Markovian model with some nodes hidden
- Hidden nodes have no parents

Observed distribution

\[
P(X, Y, Z) := \sum_u P(U = u)P(X|U = u)P(Z|X)P(Y|X, U = u)
= P(X)P(Z|X)P(Y|Z, X)
\]
Interventions without experiments [Pearl, 1995]

Observational distribution

\[ P(X, Y) \]
\[
\sum_u P(U = u)P(X|u)P(Y|X, u)
\]
Interventions without experiments [Pearl, 1995]

Observational distribution

\[ P(X, Y) \]
\[ \sum_{u} P(U = u)P(X|u)P(Y|X, u) \]

Intervention distribution

\[ P(Y | \text{do}(X = x)) \]
\[ \sum_{u} P(U = u)P(Y|X = x, u) \]

Identification problem [Pearl, 1995]

When is \( P(Y = y | \text{do}(X = x)) \) computable given the observed distribution \( P \)?
Interventions without experiments [Pearl, 1995]

Observational distribution

\[ P(X, Y) \]
\[ \sum_u P(U = u)P(X|u)P(Y|X, u) \]

Intervention distribution

\[ P(Y | \text{do}(X = x)) \]
\[ \sum_u P(U = u)P(Y|X = x, u) \]

Identification problem [Pearl, 1995]

When is \( P(Y = y | \text{do}(X = x)) \) computable given the observed distribution \( P \)?

Not always!
But sometimes it is... 

$$P(Y \mid \text{do}(X = x)) = \sum_z P(Z = z \mid X = x) \cdot \sum_{x'} P(X = x') \cdot P(Y = y \mid Z = z, X = x').$$
Deciding identifiability

A long line of work culminated in the following striking result

**Complete Identification** [Huang and Valtorta, 2008; Shpitser and Pearl, 2006, …]

An efficient algorithm with the following characteristics exists:

**Input:** Semi-Markovian graph $G = (V, E, U, D)$, disjoint subsets $X, Y$ of $V$

**Output:** Either

- A rational map

\[
ID(G, X, Y) : P(V) \mapsto P(Y \mid \text{do}(X)),
\]

- A certificate of non-existence of such a map

**Note**

- The observed distribution $P$ is **not** an input to the algorithm
- The output is not numerical, but a symbolic, **exact** description of the map
Deciding identifiability

A long line of work culminated in the following striking result

Complete Identification  [Huang and Valtorta, 2008; Shpitser and Pearl, 2006, …]

An efficient algorithm with the following characteristics exists:

**Input:** Semi-Markovian graph $G = (V, E, U, D)$, disjoint subsets $X, Y$ of $V$

**Output:** Either

- A rational map
  
  $\text{ID}(G, X, Y) : P(V) \mapsto P(Y \mid \text{do}(X))$, or

- A certificate of non-existence of such a map

**ID assumes…**

- *Exact* knowledge of observed distribution $P$
- *Exact* knowledge of the model $G$ (no “missing” edges)
Stability of the identification map

\[ G = (V, E, U, D) \text{ is a semi-Markovian graph} \]
\[ \text{ID}(G, X, Y) : P(V) \mapsto P(Y \mid \text{do}(X)) \]

**Statistical stability**

How sensitive is \( \text{ID}(G, X, Y) \) to small perturbations in the input \( P \)?
G = (V, E, U, D) is a semi-Markovian graph

ID(G, X, Y) : P(V) ⟼ P(Y | do(X))

**Statistical stability**

How sensitive is ID(G, X, Y) to small perturbations in the input P?

**Model Stability**

How sensitive is ID(G, X, Y) to extra assumptions (missing edges) in G?
Perturbations in the input: Condition number

\[ G = (V, E, U, D) \] is a semi-Markovian graph

\[ \text{ID}(G, X, Y) : P(V) \mapsto P(Y \mid \text{do}(X)) \]

Suppose instead of \( P \), we get \( \tilde{P} \) as input to \( \text{ID}(G, X, Y) \), such that

\[
(1 - \epsilon) \leq \frac{\tilde{P}(\cdot)}{P(\cdot)} \leq (1 + \epsilon) \quad \equiv \quad \text{Rel} \ P \leq \epsilon, \text{ in } \| \cdot \|_\infty \text{ norm}
\]

**Condition number**

\[
\kappa_{\text{ID}(G, X, Y)} = \sup \frac{\text{Rel} \ P(Y \mid \text{do}(X))}{\text{Rel} \ P}
\]

How large is the relative error in the output compared to that in the input?
Perturbations in the input: Condition number

\[ G = (V, E, U, D) \text{ is a semi-Markovian graph} \]

\[ \text{ID}(G, X, Y): P(V) \mapsto P(Y \mid \text{do}(X)) \]

Suppose instead of \( P \), we get \( \tilde{P} \) as input to \( \text{ID}(G, X, Y) \), such that

\[
(1 - \epsilon) \leq \frac{\tilde{P}(\cdot)}{P(\cdot)} \leq (1 + \epsilon) \quad \equiv \quad \text{Rel } P \leq \epsilon, \text{ in } \| \cdot \|_\infty \text{ norm}
\]

**Condition number**

\[
\kappa_{\text{ID}(G, X, Y)} = \lim_{\epsilon \downarrow 0} \sup_{\text{Rel } P \leq \epsilon} \frac{\text{Rel } \tilde{P}(Y \mid \text{do}(X))}{\text{Rel } P}
\]

How large is the relative error in the output compared to that in the input?

E.g., \( \kappa \) for computing conditional probabilities from \( P \) is at most 2.
Perturbations in the input: Condition number

\[ G = (V, E, U, D) \] is a semi-Markovian graph
\[ \text{ID}(G, X, Y) : P(V) \mapsto P(Y \mid \text{do}(X)) \]

Suppose instead of \( P \), we get \( \tilde{P} \) as input to \( \text{ID}(G, X, Y) \), such that

\[ (1 - \epsilon) \leq \frac{\tilde{P}(\cdot)}{P(\cdot)} \leq (1 + \epsilon) \equiv \text{Rel } P \leq \epsilon, \text{ in } \| \cdot \|_{\infty} \text{ norm} \]

Sources of perturbations

- Standard model for floating-point round off in numerical analysis

Statistical sampling errors: usually additive (even worse)
Intentionally introduced errors: e.g. by some differential privacy mechanisms
Perturbations in the input: Condition number

\[ G = (V, E, U, D) \] is a semi-Markovian graph

\[ \text{ID}(G, X, Y) : P(V) \mapsto P(Y \mid \text{do}(X)) \]

Suppose instead of \( P \), we get \( \tilde{P} \) as input to \( \text{ID}(G, X, Y) \), such that

\[
(1 - \epsilon) \leq \frac{\tilde{P}(\cdot)}{P(\cdot)} \leq (1 + \epsilon) \quad \equiv \quad \text{Rel} \, P \leq \epsilon, \text{ in } \| \cdot \|_\infty \text{ norm}
\]

Sources of perturbations

- Standard model for floating-point round off in numerical analysis
- **Statistical sampling errors**: usually additive (even worse)
Perturbations in the input: Condition number

\[ G = (V, E, U, D) \] is a semi-Markovian graph

\[ \text{ID}(G, X, Y) : P(V) \mapsto P(Y | \text{do}(X)) \]

Suppose instead of \( P \), we get \( \tilde{P} \) as input to \( \text{ID}(G, X, Y) \), such that

\[
(1 - \epsilon) \leq \frac{\tilde{P}(\cdot)}{P(\cdot)} \leq (1 + \epsilon) \equiv \text{Rel} P \leq \epsilon, \text{ in } \| \cdot \|_\infty \text{ norm}
\]

Sources of perturbations

- Standard model for floating-point round off in numerical analysis
- **Statistical sampling errors**: usually additive (even worse)
- **Intentionally introduced errors**: e.g. by some differential privacy mechanisms
Perturbations in the input: Inaccurate models

\[ G = (V, E, U, D) \] is a semi-Markovian graph

\[ \text{ID}(G, X, Y) : P(V) \leftrightarrow P(Y | \text{do}(X)) \]

Suppose instead of \( P \), we get \( \tilde{P} \) as input to \( \text{ID}(G, X, Y) \), such that

\[
(1 - \epsilon) \leq \frac{\tilde{P}(\cdot)}{P(\cdot)} \leq (1 + \epsilon) \quad \equiv \quad \text{Rel} \, P \leq \epsilon, \text{ in } || \cdot ||_\infty \text{ norm}
\]

Ignoring “weak” edges

The same framework of perturbations to \( P \) can handle “model stability” as well!

[see paper for details]
Results: Condition of causal identification

Theorem: There exist highly ill-conditioned examples!

There exists an infinite sequence of semi-Markovian graphs $G_n$ with $n$ observed vertices and disjoint subsets $S_n$ and $T_n$ of the observed vertices such that

$$\kappa_{ID}(G_n,T_n,S_n) = \exp\left( \Omega \left( n^{0.49} \right) \right)$$

- This is a property of the ID map itself, not of an algorithm computing it!
Results: Condition of causal identification

Theorem: There exist highly ill-conditioned examples!

There exists an infinite sequence of semi-Markovian graphs $G_n$ with $n$ observed vertices and disjoint subsets $S_n$ and $T_n$ of the observed vertices such that

$$\kappa_{\text{ID}}(G_n, T_n, S_n) = \exp\left(\Omega\left(n^{0.49}\right)\right)$$

- This is a property of the ID map itself, not of an algorithm computing it!

On these examples, any algorithm computing ID may lose $\Omega\left(n^{0.49}\right)$ bits of precision
Theorem: An important class of well-conditioned examples

Let $G$ be a semi-Markovian graph and let $X$ be an observed node in $G$ such that it is not possible to reach a child of $X$ from $X$ using only the hidden edges. Then, for any subset $S$ of $V$ not containing $X$.

$$\kappa_{\text{ID}}(G,X,S) = O(|V|).$$

- Identifiability under the above condition was proved by Tian and Pearl [2002].
Primitives of identifiability
Easy cases: no directed edges

\[ P(Y \mid \text{do}(X = x)) = \sum_x P(Y, X = x) = P(Y) \]

In general, if \( X \) is not an ancestor of \( Y \), it can be marginalized.
Easy cases: no hidden edges

Identification

\[ P(YZ \mid do(X = x)) = P(YZ \mid X) \]
Easy cases: no hidden edges (slightly more complicated)

A generalization of this is the crucial tool in the identification algorithms described earlier.

$$P(YZ \mid \text{do}(X = x)) = P(Z)P(Y \mid Z, X)$$
Easy cases: no hidden edges (slightly more complicated)

\[ P(YZ \mid \text{do}(X = x)) = P(Z)P(Y \mid Z, X) \]

- A generalization of this is the crucial tool in the identification algorithms described earlier
- ...and also, in connivance with the innocuous marginalization described above, the main source of ill-conditioning!
\( \{Y, Z\} \) in the above graph is a \textbf{C-component}: a \textit{maximal} connected component among observed nodes induced by the hidden edges.
C–components

\{Y, Z\} in the above graph is a C–component: a maximal connected component among observed nodes induced by the hidden edges.

C–components are identifiable

If \( S \subseteq V \) is a C–component in \( G = (V, E, U, D) \) then

\[
P(S \mid \text{do}(V - S)) = \prod_{A \in S} P(A \mid V_{\pi(<A)}),
\]

where \( \pi \) is a topological order on \( V \) according to \( E \).
C-components and general identifiability

The hardest case

The “hardest” case for identifiability is $P(S|\text{do}(X))$, where

- $X$ is an ancestor set for $S$ in $G$, and
- $S$ is a C-component in $G - X$
The hardest case

The "hardest" case for identifiability is $P(S|\text{do}(X))$, where

- $X$ is an ancestor set for $S$ in $G$, and
- $S$ is a C-component in $G - X$

\textbf{Case 1} \hspace{1cm} S \cup X \text{ is a C-component in } G: \text{ ID}(G, S, X) \text{ does not exist}
The hardest case

The “hardest” case for identifiability is $P(S|\text{do}(X))$, where
- $X$ is an ancestor set for $S$ in $G$, and
- $S$ is a C-component in $G - X$

**Case 1**  $S \cup X$ is a C-component in $G$: $\text{ID}(G, S, X)$ does not exist

**Case 2**  $S$ is a C-component in $G$: Use C-component identifiability
C-components and general identifiability

The hardest case

The “hardest” case for identifiability is $P(S \mid \text{do}(X))$, where

- $X$ is an ancestor set for $S$ in $G$, and
- $S$ is a C-component in $G - X$

Case 1 $S \cup X$ is a C-component in $G$: $\text{ID}(G, S, X)$ does not exist

Case 2 $S$ is a C-component in $G$: Use C-component identifiability

Case 3 $S \cup X'$ is a C-component in $G$, for some $X' \subsetneq X$:

Recursion

Call $\text{ID}(S \cup X', X', S)$, but with $P$ replaced by

$$P'(S \cup X') := \prod_{A \in S \cup X'} P(A \mid V_{\pi(<A)})$$

where $\pi$ is a topological order on $V$ according to $E$

Recursion will fail immediately unless some $X'$ is no more an ancestor of $S$!
The ill-conditioned examples
A warm-up calculation: \( \kappa \) is at least \( \Omega(n) \)

\[
\begin{align*}
P(\cdot) \mapsto P(S \mid \text{do}(Y)) &= P(\cdot) \mapsto \prod_{i=1}^{n} P(S_i \mid S_{<i} Y_{<i})
\end{align*}
\]
A warm-up calculation: \( \kappa \) is at least \( \Omega(n) \)

\[
P(\cdot) \mapsto P(S \mid \text{do}(Y)) = P(\cdot) \mapsto \prod_{i=1}^{n} P(S_i \mid S_{<i} Y_{<i})
\]

- When \( P \) is uniform, the output of the map is the uniform distribution
- However, one can construct a \( \tilde{P} \) that is \( \epsilon \)-close to \( P \) and such that each conditional probability above has a positive \( \Omega(\epsilon) \) relative error,
  - for a total relative error of \( \Omega(n\epsilon) \).

No recursion was used here!
The final gadget ($m = 6, k = 4$): $P(S \mid \operatorname{do}(X, Y))$
The final gadget \((m = 6, k = 4)\): \(P(S \mid \text{do}(X, Y))\)

\[
P(X_{[k]}, S_{[m+1]}, Y_{[m],[k]}):= P(X_k = x, X_{[k-1]}) \cdot \prod_{i=1}^{m} P(S_i, Y_{i,[k]} \mid \text{pred}_i) \\
\cdot P(S_{m+1} \mid \text{pred}_{m+1}),
\]
The final gadget \((m = 6, k = 4)\): \(P(S \mid \text{do}(X, Y))\)

\[
\begin{align*}
P'(X_k, S_{m+1}, Y_{m}, \lbrack k-1 \rbrack) & := P(X_k = x, X_{\lbrack k-1 \rbrack}) \cdot \prod_{i=1}^{m} P(S_i, Y_i, \lbrack k-1 \rbrack \mid \text{pred}_i) \\
& \quad \cdot P(S_{m+1} \mid \text{pred}_{m+1}),
\end{align*}
\]

\[
\text{Rel } P = \epsilon \quad \leadsto \quad \text{Rel } P' \sim m \cdot \epsilon
\]
The final gadget ($m = 6, k = 4$): $P(S \mid \text{do}(X, Y))$

$$
\pi(P) \left( X_{[k-1]}, S_{[m+1]}, Y_{[m],[k-1]} \right) := \sum_x P \left( X_k = x, X_{[k-1]} \right) \cdot \prod_{i=1}^m P \left( S_i, Y_{i,[k-1]} \mid \text{pred}_i \right) \\
\cdot P \left( S_{m+1} \mid \text{pred}_{m+1} \right),
$$

Rel $P = \epsilon \Rightarrow$ Rel $P' \sim m \cdot \epsilon \Rightarrow$ Rel $\pi(P) \sim m \cdot \epsilon$
The final gadget \((m = 6, k = 4)\): \(P(S \mid \text{do}(X, Y))\)

\[
\pi(P) (X_{[k-1]}, S_{[m+1]}, Y_{[m],[k-1]}) := \sum_x P(X_k = x, X_{[k-1]}) \cdot \prod_{i=1}^{m} P(S_i, Y_{i,[k-1]} \mid \text{pred}_i) \\
\cdot P(S_{m+1} \mid \text{pred}_{m+1}),
\]

\[
\text{Rel } P = \epsilon \quad \leadsto \quad \text{Rel } P' \sim m \cdot \epsilon \quad \leadsto \quad \text{Rel } \pi(P) ? m \cdot \epsilon
\]

Repeat \(k\) times to get \(\text{Rel ID} \sim m^k \cdot \epsilon\).
Comments

- The marginalization operation can “eat up” the accumulated error if the underlying distribution is uniform

Our proof

With appropriately chosen non-uniform distributions, the marginalization operation propagates errors.
The marginalization operation can “eat up” the accumulated error if the underlying distribution is uniform.

With appropriately chosen non-uniform distributions, the marginalization operation propagates errors.

To get a condition number of $\sim \Omega(\exp(\sqrt{n}))$, choose $m \approx k \approx \sqrt{n}$.

Details of analyzing this correctly are somewhat involved: please see paper 21.
Conclusion

**Condition number of causality**

- **Highly ill-conditioned examples exist**
  - Very small uncertainties in the model or data can introduce very large errors in causal identification

- **But not all instances are ill-conditioned**
  - A well studied class of examples indeed has small condition number: so numerically stable algorithms can be designed

**Some future directions**
- Algorithmically classify condition numbers for any given model and intervention
  - e.g., for comparing different models of the system being studied
- Find ways to get around an ill-conditioned model by using more data
  - e.g., some measured intervention distributions? error correction?
- Condition number for other causal inference problems, e.g., SEMs

Thank you!
Conclusion

Condition number of causality

Highly ill-conditioned examples exist
Very small uncertainties in the model or data can introduce very large errors in causal identification

But not all instances are ill-conditioned
A well studied class of examples indeed has small condition number: so numerically stable algorithms can be designed

Some future directions
- Algorithmically classify condition numbers for any given model and intervention
  - e.g., for comparing different models of the system being studied
Conclusion

Condition number of causality

Highly ill-conditioned examples exist

- Very small uncertainties in the model or data can introduce very large errors in causal identification

But not all instances are ill-conditioned

- A well studied class of examples indeed has small condition number: so numerically stable algorithms can be designed

Some future directions

- Algorithmically classify condition numbers for any given model and intervention
  - e.g., for comparing different models of the system being studied
- Find ways to get around an ill-conditioned model by using more data
  - e.g. some measured intervention distributions? error correction?
Conclusion

**Condition number of causality**

*Highly ill-conditioned examples exist*

- Very small uncertainties in the model or data can introduce very large errors in causal identification

*But not all instances are ill-conditioned*

- A well studied class of examples indeed has small condition number: so numerically stable algorithms can be designed

**Some future directions**

- Algorithmically classify condition numbers for any given model and intervention
  - e.g., for comparing different models of the system being studied
- Find ways to get around an ill-conditioned model by using more data
  - e.g. some measured intervention distributions? error correction?
- Condition number for other causal inference problems, e.g. SEMs
Conclusion

Condition number of causality

Highly ill-conditioned examples exist

But not all instances are ill-conditioned

Some future directions

- Algorithmically classify condition numbers for any given model and intervention
  - e.g., for comparing different models of the system being studied
- Find ways to get around an ill-conditioned model by using more data
  - e.g. some measured intervention distributions? error correction?
- Condition number for other causal inference problems, e.g. SEMs

Thank you!
Conclusion

Condition number of causality

Highly ill-conditioned examples exist

- Very small uncertainties in the model or data can introduce very large errors in causal identification

But not all instances are ill-conditioned

- A well studied class of examples indeed has small condition number: so numerically stable algorithms can be designed

Some future directions

- Algorithmically classify condition numbers for any given model and intervention
  - e.g., for comparing different models of the system being studied
- Find ways to get around an ill-conditioned model by using more data
  - e.g. some measured intervention distributions? error correction?
- Condition number for other causal inference problems, e.g. SEMs

Thank you!
Conclusion

Condition number of causality

Highly ill-conditioned examples exist
Very small uncertainties in the model or data can introduce very large errors in causal identification

But not all instances are ill-conditioned
A well studied class of examples indeed has small condition number: so numerically stable algorithms can be designed

Some future directions
- Algorithmically classify condition numbers for any given model and intervention
  - e.g., for comparing different models of the system being studied
- Find ways to get around an ill-conditioned model by using more data
  - e.g. some measured intervention distributions? error correction?
- Condition number for other causal inference problems, e.g. SEMs

Thank you!
Condition number and numerical stability

Condition number is a property of the function
Numerical stability is a property of a floating point algorithm

\[ \text{ADD} : (x_1, x_2, \ldots, x_n) \mapsto x_1 + x_2 \ldots x_n \]

Condition number

\[ \kappa = \frac{\sum_{i=1}^{n} |x_i|}{|\sum_{i=1}^{n} x_i|} = 1, \text{ for positive } x_i \]

Numerical stability: Naive linear summation

\[ O(n \cdot \varepsilon \cdot \kappa) \]

Numerical stability: Kahan summation

\[ O(\varepsilon \cdot \kappa), \text{ to first order in } \varepsilon \]

\( \varepsilon \) is the “machine epsilon”
Bibliography I


