IDA algorithm and further problems
General overview

- Introduction
- DAGs and conditional independence
- DAGs and causal effects
- Learning DAGs from observational data
- IDA algorithm
- Further problems
Estimating causal effects when equivalence class is given

- Due to the equivalence class, the problem of estimating causal effects is under-determined. Hence, effects may be unidentifiable.
Due to the equivalence class, the problem of estimating causal effects is under-determined. Hence, effects may be unidentifiable.

Existing work: Prediction algorithm *(Spirtes et al, 2000)*
- Allows for hidden variables
- Estimates only identifiable causal effects
- Goes through all possible orderings of the variables; computationally infeasible for large problems
Due to the equivalence class, the problem of estimating causal effects is under-determined. Hence, effects may be unidentifiable.

Existing work: Prediction algorithm (Spirtes et al, 2000)
- Allows for hidden variables
- Estimates only identifiable causal effects
- Goes through all possible orderings of the variables; computationally infeasible for large problems

IDA algorithm (MM, Kalisch & Bühlmann, 2009)
- Intervention-calculus when the DAG is Absent
- Does not allow for hidden variables
- Estimates multi-set of possible causal effects
- Local method that scales well to large graphs
The true causal effect is in $\Theta$. We can obtain bounds for the size of the causal effect.
Bounds based on $\Theta^L$ are identical to bounds based on $\Theta$. Proof uses graph theoretic properties of the CPDAG.
Example: what is the causal effect of $X_1$ on $Y$?
Example: what is the causal effect of $X_1$ on $Y$?

DAGs

$X_2 \rightarrow X_3$

$X_1 \rightarrow X_4 \rightarrow Y$

CPDAG

$X_2 \rightarrow X_3$

$X_1 \rightarrow X_4 \rightarrow Y$

$X_2 \leftarrow X_3$

$X_1 \rightarrow X_4 \rightarrow Y$

$X_2 \leftarrow X_3$

$X_1 \leftarrow X_4 \rightarrow Y$
Example: what is the causal effect of $X_1$ on $Y$?

$\beta_{1|S}$ is the coefficient of $X_1$ in the regression of $Y$ on $X_1$ and $S$.
Example: what is the causal effect of $X_1$ on $Y$?

**CPDAG**

\[
\text{DAGs}
\]

\[
\begin{align*}
X_2 &\rightarrow X_3 \\
X_1 &\rightarrow X_4 \rightarrow Y \\
X_2 &\rightarrow X_3 \\
X_1 &\rightarrow X_4 \rightarrow Y \\
X_2 &\leftarrow X_3 \\
X_1 &\rightarrow X_4 \rightarrow Y \\
X_2 &\rightarrow X_3 \\
X_1 &\leftarrow X_4 \rightarrow Y
\end{align*}
\]

**causal effects**

\[
\begin{align*}
\beta_{1|\emptyset} \\
\beta_{1|X_2} \\
\Theta = \{\beta_{1|\emptyset}, \beta_{1|X_2}, \beta_{1|X_4}\}
\end{align*}
\]

$\beta_{1|S}$ is the coefficient of $X_1$ in the regression of $Y$ on $X_1$ and $S$.
Example: what is the causal effect of $X_1$ on $Y$?

$\Theta = \{\beta_{1|\emptyset}, \beta_{1|X_2}, \beta_{1|X_2, X_1}, \beta_{1|X_4}\}$

$\beta_{1|S}$ is the coefficient of $X_1$ in the regression of $Y$ on $X_1$ and $S$
Example: what is the causal effect of $X_1$ on $Y$?
Example: what is the causal effect of $X_1$ on $Y$?

parents

$\emptyset$

CPDAG

$X_2 \rightarrow X_3$

$X_1 \rightarrow X_4 \rightarrow Y$

$X_2, X_4$
Example: what is the causal effect of \( X_1 \) on \( Y \)?

\[
\begin{array}{c|c}
\text{parents} & \text{effect} \\
\hline
\emptyset & \beta_1 | \emptyset \\
X_2 & \beta_1 | X_2 \\
X_4 & \beta_1 | X_4 \\
X_2, X_4 & \beta_1 | X_2, X_4 \\
\end{array}
\]
Example: what is the causal effect of $X_1$ on $Y$?

\[
\begin{array}{ccc}
\text{parents} & \text{effect} & \text{nr of DAGs} \\
\emptyset & \beta_1|\emptyset & 1 \\
X_2 & \beta_1|X_2 & 2 \\
X_4 & \beta_1|X_4 & 1 \\
X_2, X_4 & \beta_1|X_2, X_4 & 0 \\
\end{array}
\]
Example: what is the causal effect of $X_1$ on $Y$?

<table>
<thead>
<tr>
<th>parents</th>
<th>effect</th>
<th>nr of DAGs</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\emptyset$</td>
<td>$\beta_1</td>
<td>\emptyset$</td>
</tr>
<tr>
<td>$X_2$</td>
<td>$\beta_1</td>
<td>X_2$</td>
</tr>
<tr>
<td>$X_4$</td>
<td>$\beta_1</td>
<td>X_4$</td>
</tr>
</tbody>
</table>

$\Theta = \{\beta_1|\emptyset, \beta_1|X_2, \beta_1|X_2, \beta_1|X_4\}$
Example: what is the causal effect of $X_1$ on $Y$?

<table>
<thead>
<tr>
<th>parents</th>
<th>effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\emptyset$</td>
<td>$\beta_1</td>
</tr>
<tr>
<td>$X_2$</td>
<td>$\beta_1</td>
</tr>
<tr>
<td>$X_4$</td>
<td>$\beta_1</td>
</tr>
<tr>
<td>$X_2, X_4$</td>
<td>$\beta_1</td>
</tr>
</tbody>
</table>

CPDAG

$X_2 \rightarrow X_3$

$X_1 \rightarrow X_4 \rightarrow Y$
Example: what is the causal effect of $X_1$ on $Y$?

<table>
<thead>
<tr>
<th>parents</th>
<th>effect</th>
<th>locally valid</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\emptyset$</td>
<td>$\beta_1</td>
<td>\emptyset$</td>
</tr>
<tr>
<td>$X_2$</td>
<td>$\beta_1</td>
<td>X_2$</td>
</tr>
<tr>
<td>$X_4$</td>
<td>$\beta_1</td>
<td>X_4$</td>
</tr>
<tr>
<td>$X_2, X_4$</td>
<td>$\beta_1</td>
<td>X_2,X_4$</td>
</tr>
</tbody>
</table>

locally valid = no additional v-structure with $X_1$ as collider
Example: what is the causal effect of $X_1$ on $Y$?

<table>
<thead>
<tr>
<th>parents</th>
<th>effect</th>
<th>locally valid</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\emptyset$</td>
<td>$\beta_1</td>
<td>\emptyset$</td>
</tr>
<tr>
<td>$X_2$</td>
<td>$\beta_1</td>
<td>X_2$</td>
</tr>
<tr>
<td>$X_4$</td>
<td>$\beta_1</td>
<td>X_4$</td>
</tr>
<tr>
<td>$X_2$, $X_4$</td>
<td>$\beta_1</td>
<td>X_2,X_4$</td>
</tr>
</tbody>
</table>

$\Theta^L = \{\beta_1|\emptyset, \beta_1|X_2, \beta_1|X_4\}$

locally valid = no additional v-structure with $X_1$ as collider
- Theorem: $\Theta^\text{set} \equiv \Theta^L$
  - Distinct elements of $\Theta$ and $\Theta^L$ are the same
  - Multiplicities may differ
Theorem: $\Theta \stackrel{\text{set}}{=} \Theta^L$

- Distinct elements of $\Theta$ and $\Theta^L$ are the same
- Multiplicities may differ

Example: $\Theta = \{1.5, 1.5, -1\}$, $\Theta^L = \{1.5, -1\}$
Theorem: $\Theta \seteq \Theta^L$

- Distinct elements of $\Theta$ and $\Theta^L$ are the same
- Multiplicities may differ

Example: $\Theta = \{1.5, 1.5, -1\}$, $\Theta^L = \{1.5, -1\}$

Corollary: Minimum absolute values of $\Theta^L$ and $\Theta$ are identical
Theorem: $\Theta \text{ set} \equiv \Theta^L$

- Distinct elements of $\Theta$ and $\Theta^L$ are the same
- Multiplicities may differ

Example: $\Theta = \{1.5, 1.5, -1\}$, $\Theta^L = \{1.5, -1\}$

Corollary: Minimum absolute values of $\Theta^L$ and $\Theta$ are identical

Sketch of proof:

- **Trivial**: $\Theta \subseteq \Theta^L$
  - Any DAG in the equivalence class is locally valid

- **Non-trivial**: $\Theta^L \subseteq \Theta$
  - Whenever a graph is locally valid, its remaining edges can be directed to form a valid DAG in the equivalence class
Bounds based on $\Theta^L$ are identical to bounds based on $\Theta$. Proof uses graph theoretic properties of the CPDAG.
The estimates are consistent in sparse high-dimensional settings
Summary of IDA

- IDA estimates **bounds on causal effects from observational data**, assuming the data come from an **unknown DAG**:
  - computationally feasible for large sparse systems due to PC algorithm and local method
• IDA estimates **bounds on causal effects from observational data**, assuming the data come from an **unknown DAG**:
  - computationally feasible for large sparse systems due to PC algorithm and local method
  - **software: R-package** `pcalg` (Kalisch et al, 2012)
• IDA estimates bounds on causal effects from observational data, assuming the data come from an unknown DAG:
  • computationally feasible for large sparse systems due to PC algorithm and local method
  • software: R-package pcalg (Kalisch et al, 2012)
  • consistency in sparse high-dimensional settings (MM, Kalisch & Bühlmann, 2009)
Summary of IDA

- IDA estimates **bounds on causal effects from observational data**, assuming the data come from an **unknown DAG**:
  - computationally feasible for large sparse systems due to PC algorithm and local method
  - software: R-package `pcalg` (Kalisch et al, 2012)
  - consistency in sparse high-dimensional settings (MM, Kalisch & Bühlmann, 2009)
  - validations on real data:
    - **Gene regulatory network of yeast** (Hughes et al, 2000; MM et al, 2010)
    - **DREAM4 In Silico Network Challenge** (Marbach et al, 2009; MM et al, 2010)
    - **Genes affecting flowering time in Arabidopsis thaliana** (Stekhoven et al, 2012)
General overview

- Introduction
- DAGs and conditional independence
- DAGs and causal effects
- Learning DAGs from observational data
- IDA algorithm
- Further problems: instability issues
Yeast results (re-scaled $y$-axis)

- **True positives**
  - IDA (dashed red line)
  - Random (dotted green line)

- **False positives**
  - Range from 0 to 4000

- **True positives range**
  - 0 to 2400

- **Y-axis scale**
  - Re-scaled to better visualize the data points.
Results are highly sensitive to the variable ordering
Yeast: estimated skeleton for 25 random orderings

Number of permutations:
- 5
- 10
- 15
- 20
- 25

Number of edges:
- 5000
- 10000
- 15000

Diagram showing the estimated skeleton for 25 random orderings of yeast.
Three steps:

- Determine the skeleton
- Determine the v-structures ($\rightarrow \leftarrow$)
- Orient as many of the remaining edges as possible
PC algorithm

Three steps:
- Determine the skeleton
- Determine the v-structures (→←)
- Orient as many of the remaining edges as possible

Order-dependence:
- All three steps are order-dependent: the output depends on the ordering of the variables
- This order-dependence was known. But we were surprised to find that it is so very problematic high-dimensional settings.
PC algorithm: the skeleton

- Idea:
  - No edge between $X_i$ and $X_j$
  
  $\iff \ X_i \perp \!\!\!\!\!\!\perp X_j \mid S$ for some subset $S$ of the remaining variables
  
  $\iff \ X_i \perp \!\!\!\!\!\!\perp X_j \mid S'$ for some subset $S'$ of $\text{adj}(X_i)$ or $\text{adj}(X_j)$
PC algorithm: the skeleton

- **Idea:**
  - No edge between $X_i$ and $X_j$ \iff $X_i \indep X_j \mid S$ for some subset $S$ of the remaining variables \iff $X_i \indep X_j \mid S'$ for some subset $S'$ of $\text{adj}(X_i)$ or $\text{adj}(X_j)$

- **Implementation:**
  - Start with complete graph
  - Consider marginal independence of all pairs of variables; remove edges
  - Consider conditional independence given subsets of size $k$ of the adjacency sets, for $k = 1, 2, \ldots$
PC algorithm: the skeleton

- **Idea:**
  - No edge between $X_i$ and $X_j$
    \[ \iff \]
    \[ X_i \perp \perp X_j | S \text{ for some subset } S \text{ of the remaining variables} \]
    \[ \iff \]
    \[ X_i \perp \perp X_j | S' \text{ for some subset } S' \text{ of } \text{adj}(X_i) \text{ or } \text{adj}(X_j) \]

- **Implementation:**
  - Start with complete graph
  - Consider marginal independence of all pairs of variables; remove edges
  - Consider conditional independence given subsets of size $k$ of the adjacency sets, for $k = 1, 2, \ldots$

- **Order-dependence in sample version:** the order in which variables are tested determines which edges are removed first. This affects which tests are considered later on.
PC algorithm: the skeleton

- **Idea:**
  - No edge between $X_i$ and $X_j$ \iff \begin{align*}
  X_i \perp \!\!\!\!\perp X_j|S & \text{ for some subset } S \text{ of the remaining variables} \\
  X_i \perp \!\!\!\!\perp X_j|S' & \text{ for some subset } S' \text{ of } \text{adj}(X_i) \text{ or } \text{adj}(X_j)
  \end{align*}

- **Implementation:**
  - Start with complete graph
  - Consider marginal independence of all pairs of variables; remove edges
  - Consider conditional independence given subsets of size $k$ of the adjacency sets, for $k = 1, 2, \ldots$

- **Order-dependence in sample version:** the order in which variables are tested determines which edges are removed first. This affects which tests are considered later on.

- **PC-stable:** fix adjacency sets within each level of $k$
Order-independent PC algorithm

Three steps:

- Determine the skeleton
  - PC-stable

- Determine the v-structures (→←)
  - Use Conservative PC (CPC) or Majority rule PC (MPC)

- Orient as many of the remaining edges as possible
  - Use lists and allow bi-directed edges (L)
Order-independent PC algorithm

Three steps:
- Determine the skeleton
  - PC-stable
- Determine the v-structures (→←)
  - Use Conservative PC (CPC) or Majority rule PC (MPC)
- Orient as many of the remaining edges as possible
  - Use lists and allow bi-directed edges (L)

Properties of the modified algorithms:
- Oracle versions are identical to original PC
- Sample versions are consistent in high-dimensional settings under the same conditions as PC
- LCPC-stable and LMPC-stable are fully order-independent in the sample version
Related constraint-based causal structure learning algorithms

- Systems with hidden variables:
  - FCI (Spirtes et al, 2000)
  - RFCI (Colombo et al, 2012)
  - FCI+ (Claassen et al, 2013)

- Cyclic systems:
  - CCD (Richardson, 1996)
Related constraint-based causal structure learning algorithms

- Systems with hidden variables:
  - FCI (Spirtes et al, 2000)
  - RFCI (Colombo et al, 2012)
  - FCI+ (Claassen et al, 2013)

- Cyclic systems:
  - CCD (Richardson, 1996)

Order-dependence:
- All these algorithms suffer from order-dependence issues, as they run PC as their first step
- One can create order-independent versions as we did for PC
Systems with hidden variables:
- FCI (Spirtes et al, 2000)
- RFCI (Colombo et al, 2012)
- FCI+ (Claassen et al, 2013)

Cyclic systems:
- CCD (Richardson, 1996)

Order-dependence:
- All these algorithms suffer from order-dependence issues, as they run PC as their first step
- One can create order-independent versions as we did for PC

Parallelizability:
- Order-independent versions are also easier to parallelize
Simulations

- We compared the modifications of PC, FCI and RFCI in low-dimensional and high-dimensional settings.
We compared the modifications of PC, FCI and RFCI in low-dimensional and high-dimensional settings.

Low-dimensional ($p = 50, n = 1000, E(N) \in \{2, 4\}$):

- All algorithms perform roughly the same
- Conservative / majority rule versions improve the orientations
We compared the modifications of PC, FCI and RFCI in low-dimensional and high-dimensional settings.

Low-dimensional \((p = 50, n = 1000, E(N) \in \{2, 4\})\):
- All algorithms perform roughly the same
- Conservative / majority rule versions improve the orientations

High-dimensional \((p = 1000, n = 50, E(N) = 2)\):
- Stabilizing the first step of PC largely improves the skeleton
- Conservative / majority rule versions improve the orientations
- Modifications with lists make little difference
Simulation results: high-dimensional skeleton

PC

RFCI

Nm. edges

Nm. errors

TDR

$\alpha$

$\alpha$
Simulation results: high-dimensional CPDAG and PAG

PC

RFCl

[Graph showing SHD edge marks for different algorithms and settings]
Back to the yeast data: PC-stable (skeleton)

24 edges

5
10
15
20
25

5000 10000 15000

permutations

edges
Yeast: IDA with (M/C)PC-stable

![Graph showing true positives vs. false positives for different methods: IDA PC-stable perm, IDA M/CPC-stable, IDA, Random.]
Stekhoven et al. (2012) combined IDA with subsampling (SS)
Incorporating subsampling

- Stekhoven et al. (2012) combined IDA with subsampling (SS)

- Within each subsample, one can also permute the variables (SSP)
- Any version of IDA with SSP is approximately order-independent
Yeast: Modifications with subsampling

![Graph showing true positives vs. false positives for different methods.]

- PC-stable + SS
- PC-stable + SSP
- PC + SSP
- MPC-stable + SS(P)
- PC + SS
- PC-stable
- PC
- RG
- MPC-stable

False positives vs. True positives graph with various lines representing different methods and conditions.
Summary instability issues

- Constraint-based causal structure learning methods can be severely order-dependent in high-dimensional systems
- We developed simple order-independent modifications
- Simulations:
  - All algorithms performed similarly in low-dimensional settings
  - The order-independent versions performed better in high-dimensional settings
- Yeast data:
  - PC-stable+SSP seems the best choice
  - MPC-stable and CPC-stable perform badly due to very small number of directed edges
- Adding sub-sampling (preferably permuting the variable ordering) helps
General overview

- Introduction
- DAGs and conditional independence
- DAGs and causal effects
- Learning DAGs from observational data
- IDA algorithm
- Further problems: hidden variables
Example: what can go wrong when there are hidden variables?

- DAG with two hidden variables $L_1$ and $L_2$:
Example: what can go wrong when there are hidden variables?

- DAG with two hidden variables $L_1$ and $L_2$:
  
  
  \[
  \begin{array}{c}
  \begin{array}{ccc}
  L_1 & \rightarrow & L_2 \\
  \downarrow & & \downarrow \\
  X_1 & \rightarrow & X_2 \\
  \end{array}
  \\
  X_2 & \rightarrow & X_3
  \end{array}
  \]

- The conditional independence relationships among \{X_1, X_2, X_3\} can be uniquely represented by the DAG
  
  \[
  \begin{array}{ccc}
  X_1 & \rightarrow & X_2 \\
  \rightarrow & & \leftarrow \\
  X_2 & \leftarrow & X_3
  \end{array}
  \]
Example: what can go wrong when there are hidden variables?

- DAG with two hidden variables $L_1$ and $L_2$:

\[
\begin{array}{ccc}
L_1 & L_2 \\
X_1 & X_2 & X_3
\end{array}
\]

- The conditional independence relationships among $\{X_1, X_2, X_3\}$ can be uniquely represented by the DAG

\[
X_1 \rightarrow X_2 \leftarrow X_3
\]

- But interpreting this DAG causally would lead us to think that $X_1$ and $X_3$ are causes of $X_2$.

- This is wrong! And this causes the output of IDA to be wrong!
We work with maximal ancestral graphs (MAGs) (Richardson & Spirtes, 2002)

We can consider similar steps as before:
- Learning the equivalence class of MAGs: new algorithms
- Estimating causal effects when the equivalence class is given: generalized backdoor criterion

Note:
- Since we allow hidden variables, the equivalence classes are larger, and the problem is even more underdetermined.
- But we can still learn causal information. In the example, conditional independence relationships among \( \{X_1, X_2, X_3\} \) imply:
  - \( X_2 \) is not a cause of \( X_1 \) nor of \( X_3 \)
  - \( X_1 \) is not a cause of \( X_3 \) and vice versa
  - \( X_1 \) and \( X_3 \) may or may not be causes of \( X_2 \)
Problem:
- In many scientific questions, we are interested in causal relationships
- It is best to investigate such relationships via randomized controlled experiments, but these are not always possible
- Hence, we study causal methods for observational data from complex systems
- Such methods cannot replace randomized controlled experiments. But they can be very useful for exploration:
  - hypothesis generation
  - prioritization of experiments
IDA estimates bounds on causal effects from observational data, assuming the data come from an unknown DAG:

- computationally feasible for large sparse systems
- consistency in sparse high-dimensional settings
- validations in biological systems
Summary

• IDA estimates bounds on causal effects from observational data, assuming the data come from an unknown DAG:
  ● computationally feasible for large sparse systems
  ● consistency in sparse high-dimensional settings
  ● validations in biological systems

• Beware of possible instability for high-dimensional data
  ● use order-independent modifications
Summary

- IDA estimates bounds on causal effects from observational data, assuming the data come from an unknown DAG:
  - computationally feasible for large sparse systems
  - consistency in sparse high-dimensional settings
  - validations in biological systems
- Beware of possible instability for high-dimensional data
  - use order-independent modifications
- In the presence of unmeasured/hidden variables:
  - RFCI and modifications of FCI for fast causal structure learning
  - consistency of (R)FCI in sparse high-dimensional settings
  - generalized backdoor criterion
Summary

• IDA estimates bounds on causal effects from observational data, assuming the data come from an unknown DAG:
  • computationally feasible for large sparse systems
  • consistency in sparse high-dimensional settings
  • validations in biological systems
• Beware of possible instability for high-dimensional data
  • use order-independent modifications
• In the presence of unmeasured/hidden variables:
  • RFCI and modifications of FCI for fast causal structure learning
  • consistency of (R)FCI in sparse high-dimensional settings
  • generalized backdoor criterion
• All software is available in the R-package pcalg
  (Kalisch et al, JSS, 2012)
Thank you for your attention!
• IDA methodology:

• IDA validations:

• Causal structure learning:

• Backdoor criterion:

• Software: