Conservative causal discovery by use of supervised machine learning

Anne Helby Petersen

Joint work with Joseph Ramsey, Claus Ekstrøm & Peter Spirtes

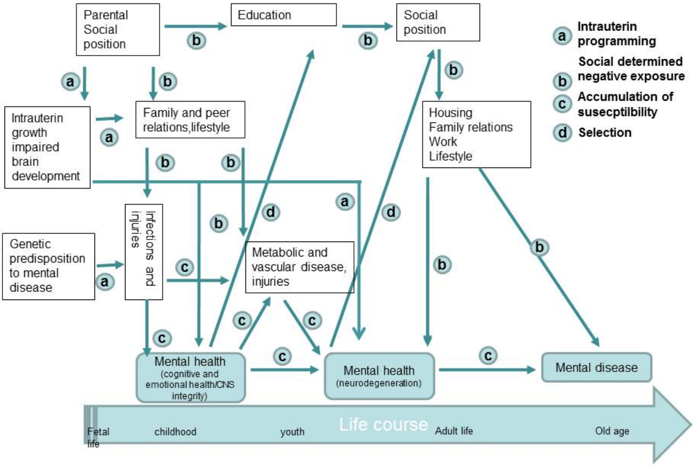
Motivation

RQ: What factors influence development of alcohol abuse?

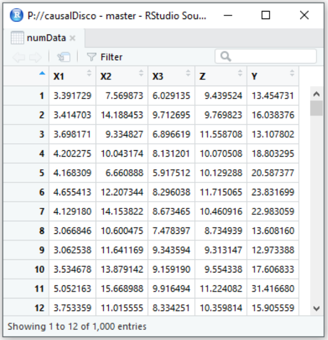
Motivation

RQ: What factors influence development of alcohol abuse?

Fig 1. Life-course model for mental health with an indication of the mechanisms linking life expoures and mental disease



A statistician's dream



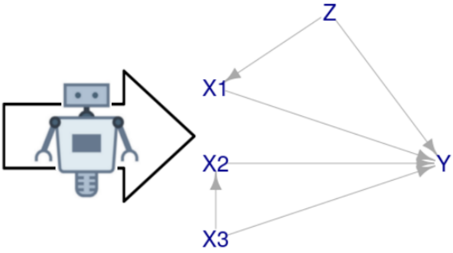
P://causalDisco - master - RStudio Sou...

numData

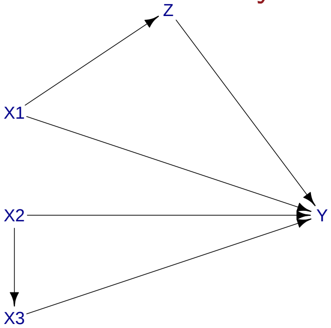
Filter

	X1	X2	X3	Z	Y
1	3.391729	7.569873	6.029135	9.439524	13.454731
2	3.414703	14.188453	9.712695	9.769823	16.038376
3	3.698171	9.334827	6.896619	11.558708	13.107802
4	4.202275	10.043174	8.131201	10.070508	18.803295
5	4.168309	6.660888	5.917512	10.129288	20.587377
6	4.655413	12.207344	8.296038	11.715065	23.831699
7	4.129180	14.153822	8.673465	10.460916	22.983059
8	3.066846	10.600475	7.478397	8.734939	13.608160
9	3.062538	11.641169	9.343594	9.313147	12.973388
10	3.534678	13.879142	9.159190	9.554338	17.606833
11	5.052163	15.668988	9.916494	11.224082	31.416680
12	3.753359	11.015555	8.334251	10.359814	15.905559

Showing 1 to 12 of 1,000 entries

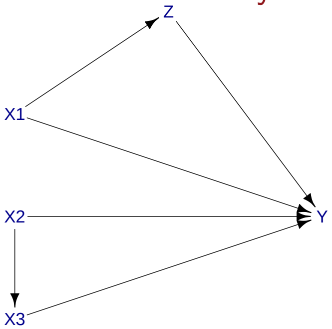


Causal models 101: Directed acyclic graph (DAG)



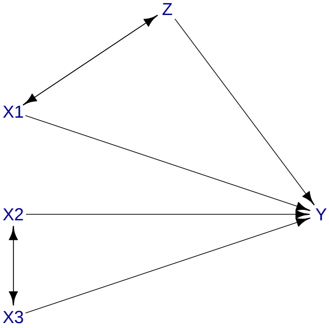
- **DAG interpretation:** Arrow from X to Y means that X is a cause of Y.
- **Markov property:** Often, DAG structure $\Rightarrow$ conditional independencies in distribution.
- **Faithfulness assumption:** We also assume that conditional independencies in distribution $\Rightarrow$ DAG structure.

Causal models 101: Directed acyclic graph (DAG)



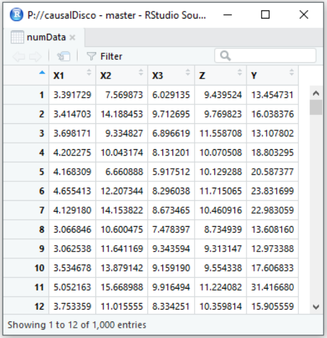
- **DAG interpretation:** Arrow from X to Y means that X is a cause of Y.
- **Markov property:** Often, DAG structure $\Rightarrow$ conditional independencies in distribution.
- **Faithfulness assumption:** We also assume that conditional independencies in distribution $\Rightarrow$ DAG structure.
- **Idea:** Use conditional independencies in data to infer DAG(?)

Completed partially directed acyclic graph (CPDAG)



- **Observational equivalence:** Some DAGs produce the same conditional independencies. Example: $X \to Y$ and $Y \to X$.
- **Equivalence class:** A CPDAG describes the equivalence class of all DAGs that imply the same conditional independencies.
- **CPDAG interpretation:** As DAG, but undirected edges means we do not know orientation.

A statistician's dream made realistic

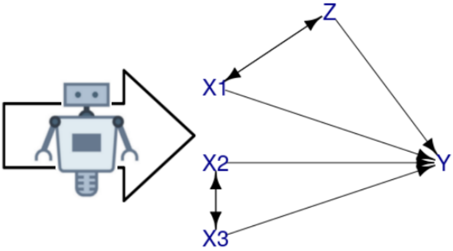

P://causalDisco - master - RStudio Sou...

numData

Filter

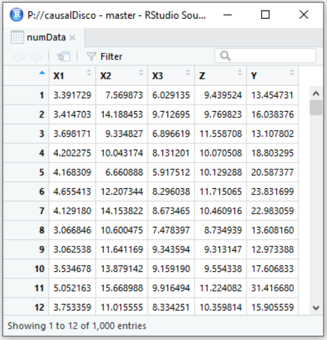
	X1	X2	X3	Z	Y
1	3.391729	7.569873	6.029135	9.439524	13.454731
2	3.414703	14.188453	9.712695	9.769823	16.038376
3	3.698171	9.334827	6.896619	11.558708	13.107802
4	4.202275	10.043174	8.131201	10.070508	18.803295
5	4.168309	6.660888	5.917512	10.129288	20.587377
6	4.655413	12.207344	8.296038	11.715065	23.831699
7	4.129180	14.153822	8.673465	10.460916	22.983059
8	3.066846	10.600475	7.478397	8.734939	13.608160
9	3.062538	11.641169	9.343594	9.313147	12.973388
10	3.534678	13.879142	9.159190	9.554338	17.606833
11	5.052163	15.668988	9.916494	11.224082	31.416680
12	3.753359	11.015555	8.334251	10.359814	15.905559

Showing 1 to 12 of 1,000 entries

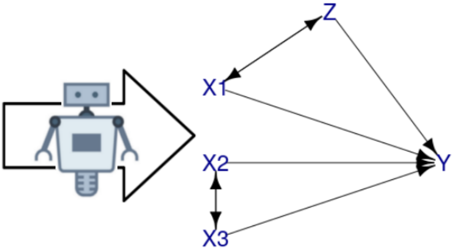


Goal of causal discovery: Estimate CPDAG by analyzing data.

A statistician's dream made realistic



	X1	X2	X3	Z	Y
1	3.391729	7.569873	6.029135	9.439524	13.454731
2	3.414703	14.188453	9.712695	9.769823	16.038376
3	3.698171	9.334827	6.896619	11.558708	13.107802
4	4.202275	10.043174	8.131201	10.070508	18.803295
5	4.168309	6.660888	5.917512	10.129288	20.587377
6	4.655413	12.207344	8.296038	11.715065	23.831699
7	4.129180	14.153822	8.673465	10.460916	22.983059
8	3.066846	10.600475	7.478397	8.734939	13.608160
9	3.062538	11.641169	9.343594	9.313147	12.973388
10	3.534678	13.879142	9.159190	9.554338	17.606833
11	5.052163	15.668988	9.916494	11.224082	31.416680
12	3.753359	11.015555	8.334251	10.359814	15.905559



Goal of causal discovery: Estimate CPDAG by analyzing data.

Goal of today's talk: Do this in a **conservative** manner, and ensure acceptable performance on **small and moderate samples**.

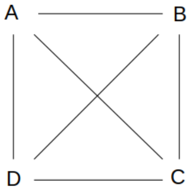
Conservative discovery

Budget statistical error so that we get:

- Rather too many edges than too few.
- Rather too few oriented edges than too many.

But do make some causal claims (no trivial solutions, and as informative as possible).

Conservative discovery

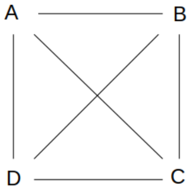


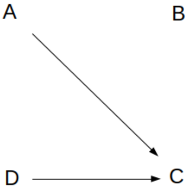
Budget statistical error so that we get:

- Rather too many edges than too few.
- Rather too few oriented edges than too many.

But do make some causal claims (no trivial solutions, and as informative as possible).

Conservative discovery



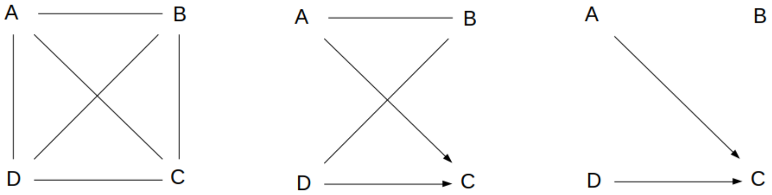


Budget statistical error so that we get:

- Rather too many edges than too few.
- Rather too few oriented edges than too many.

But do make some causal claims (no trivial solutions, and as informative as possible).

Conservative discovery



Budget statistical error so that we get:

- Rather too many edges than too few.
- Rather too few oriented edges than too many.

But do make some causal claims (no trivial solutions, and as informative as possible).

Small/moderate sample performance of existing methods

- Most existing causal discovery algorithms use **sequential testing** or **greedy search strategies**.
- Under appropriate assumptions, these algorithms are **correct and complete as** $n \to \infty$.
- In practice, we see **poor small/moderate sample performance**, most likely due to error propagation

Small/moderate sample performance of existing methods

- Most existing causal discovery algorithms use **sequential testing** or **greedy search strategies**.
- Under appropriate assumptions, these algorithms are **correct and complete as** $n \to \infty$.
- In practice, we see **poor small/moderate sample performance**, most likely due to error propagation

Example: PC algorithm

1. Start with fully connected undirected graph
2. Repeat: For each pair of variables (A, B), look for separating sets S among neighbors of A or B s.t. $A \perp\!\!\!\perp B \mid S$. If such an S exists: Remove edge between A and B.
3. Apply orientation rules making use of unshielded colliders and acyclicity assumption

Proposed solution: Supervised learning discovery (SLdisco)

1. Simulate training data with known data generating mechanisms
2. Train machine learning model on training data (simulated data) + labels (true CPDAGs)
3. Use resulting classification function as a one-step causal discovery procedure on real data

Proposed solution: Supervised learning discovery (SLdisco)

1. Simulate training data with known data generating mechanisms
2. Train machine learning model on training data (simulated data) + labels (true CPDAGs)
3. Use resulting classification function as a one-step causal discovery procedure on real data

Motivation:

Proposed solution: Supervised learning discovery (SLdisco)

1. Simulate training data with known data generating mechanisms
2. Train machine learning model on training data (simulated data) + labels (true CPDAGs)
3. Use resulting classification function as a one-step causal discovery procedure on real data

Motivation:

Sample size: Learn full graph structure jointly $\Rightarrow$ errors do not propagate

Proposed solution: Supervised learning discovery (SLdisco)

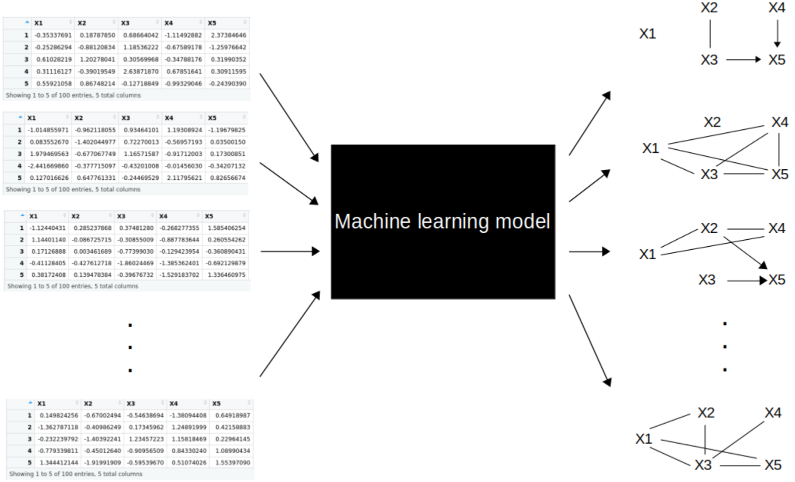
1. Simulate training data with known data generating mechanisms
2. Train machine learning model on training data (simulated data) + labels (true CPDAGs)
3. Use resulting classification function as a one-step causal discovery procedure on real data

Motivation:

Sample size: Learn full graph structure jointly ⇒ errors do not propagate

Error tradeoff: No built-in bias towards sparse/dense graphs + outputs probabilities ⇒ can be calibrated to prefered error tradeoff

SLdisco: Supervised learning discovery



	X1	X2	X3	X4	X5
1	-0.35337691	0.18787850	0.68664042	-1.11492882	2.37384646
2	-0.25286294	-0.88120834	1.18536222	-0.67589178	-1.25976642
3	0.61028219	1.20278041	0.30569968	-0.34788176	0.31990352
4	0.31116127	-0.39019549	2.63871870	0.67851641	0.30911595
5	0.55921058	0.86748214	-0.12718849	-0.99329046	-0.24390390

Showing 1 to 5 of 100 entries, 5 total columns

	X1	X2	X3	X4	X5
1	-1.014855971	-0.962118055	0.93464101	1.19308924	-1.19679825
2	0.083552670	-1.402044977	0.72270013	-0.56957193	0.03500150
3	1.979469563	-0.677067749	1.16571587	-0.91712003	0.17300851
4	-2.441669860	-0.377715097	-0.43201008	-0.01456030	-0.34207132
5	0.127016626	0.647761331	-0.24469529	2.11795621	0.82656674

Showing 1 to 5 of 100 entries, 5 total columns

	X1	X2	X3	X4	X5
1	-1.12440431	0.285237868	0.37481280	-0.268277355	1.585406254
2	1.14401140	-0.086725715	-0.30855009	-0.887783644	0.260554262
3	0.17126888	0.003461689	-0.77399030	-0.129423954	-0.360890431
4	-0.41128405	-0.427612718	-1.86024469	-1.385362401	-0.692129879
5	0.38172408	0.139478384	-0.39676732	-1.529183702	1.336460975

Showing 1 to 5 of 100 entries, 5 total columns

	X1	X2	X3	X4	X5
1	0.149824256	-0.67002494	-0.54638694	-1.38094408	0.64918987
2	-1.362787118	-0.40986249	0.17345962	1.24891999	0.42158883
3	-0.232239792	-1.40392241	1.23457223	1.15818469	0.22964145
4	-0.779339811	-0.45012640	-0.90956509	0.84330240	1.08990434
5	1.344412144	-1.91991909	-0.59539670	0.51074026	1.55397090

Showing 1 to 5 of 100 entries, 5 total columns

SLdisco: Supervised learning discovery

	X1	X2	X3	X4	X5
1	-0.35337691	0.18787850	0.68664042	-1.11492882	2.37384646
2	-0.25286294	-0.88120834	1.18536222	-0.67589178	-1.25976642
3	0.61028219	1.20278041	0.30569968	-0.34788176	0.31990352
4	0.31116127	-0.39019549	2.63871870	0.67851641	0.30911595
5	0.55921058	0.86748214	-0.12718849	-0.99329046	-0.24390390

Showing 1 to 5 of 100 entries, 5 total columns

	X1	X2	X3	X4	X5
1	-1.014855971	-0.962118055	0.93464101	1.19308924	-1.19679825
2	0.083552670	-1.402044977	0.72270013	-0.56957193	0.03500150
3	1.979469563	-0.677067749	1.16571587	-0.91712003	0.17300851
4	-2.441669860	-0.377715097	-0.43201008	-0.01456030	-0.34207132
5	0.127016626	0.647761331	-0.24469529	2.11795621	0.82656674

Showing 1 to 5 of 100 entries, 5 total columns

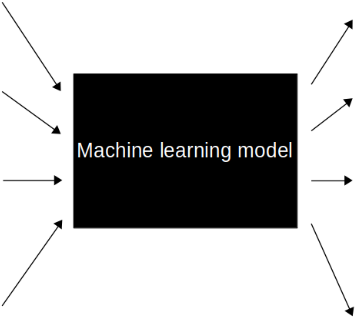
	X1	X2	X3	X4	X5
1	-1.12440431	0.285237868	0.37481280	-0.268277355	1.585406254
2	1.14401140	-0.086725715	-0.30855009	-0.887783644	0.260554262
3	0.17126888	0.003461689	-0.77399030	-0.129423954	-0.360890431
4	-0.41128405	-0.427612718	-1.86024469	-1.385362401	-0.692129879
5	0.38172408	0.139478384	-0.39676732	-1.529183702	1.336460975

Showing 1 to 5 of 100 entries, 5 total columns

⋮

	X1	X2	X3	X4	X5
1	0.149824256	-0.67002494	-0.54638694	-1.38094408	0.64918987
2	-1.362787118	-0.40986249	0.17345962	1.24891999	0.42158883
3	-0.232239792	-1.40392241	1.23457223	1.15818469	0.22964145
4	-0.779339811	-0.45012640	-0.90956509	0.84330240	1.08990434
5	1.344412144	-1.91991909	-0.59539670	0.51074026	1.55397090

Showing 1 to 5 of 100 entries, 5 total columns



	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	0	0	0	0
[2,]	0	0	1	0	0
[3,]	0	1	0	0	0
[4,]	0	0	0	0	0
[5,]	0	0	1	1	0

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	0	1	1	1
[2,]	0	0	0	0	0
[3,]	1	0	0	1	1
[4,]	1	0	1	0	1
[5,]	1	0	1	1	0

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	1	0	1	0
[2,]	1	0	0	1	0
[3,]	0	0	0	0	0
[4,]	1	1	0	0	0
[5,]	0	1	1	0	0

⋮

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	1	1	0	1
[2,]	1	0	1	0	0
[3,]	1	1	0	1	1
[4,]	0	0	1	0	0
[5,]	1	0	1	0	0

SLdisco: Supervised learning discovery

	X1	X2	X3	X4	X5
1	-0.35337691	0.18787850	0.68664042	-1.11492882	2.37384646
2	-0.25286294	-0.88120834	1.18536222	-0.67589178	-1.25976642
3	0.61028219	1.20278041	0.30569968	-0.34788176	0.31990352
4	0.31116127	-0.39019549	2.63871870	0.67851641	0.30911595
5	0.55921058	0.86748214	-0.12718849	-0.99329046	-0.24390390

Showing 1 to 5 of 100 entries, 5 total columns

	X1	X2	X3	X4	X5
1	-1.014855971	-0.962118055	0.93464101	1.19308924	-1.19679825
2	0.083552670	-1.402044977	0.72270013	-0.56957193	0.03500150
3	1.979469563	-0.677067749	1.16571587	-0.91712003	0.17300851
4	-2.441669860	-0.377715097	-0.43201008	-0.01456030	-0.34207132
5	0.127016626	0.647761331	-0.24469529	2.11795621	0.82656674

Showing 1 to 5 of 100 entries, 5 total columns

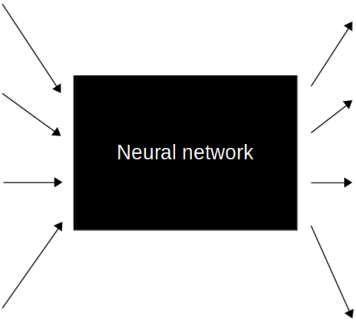
	X1	X2	X3	X4	X5
1	-1.12440431	0.285237868	0.37481280	-0.268277355	1.585406254
2	1.14401140	-0.086725715	-0.30855009	-0.887783644	0.260554262
3	0.17126888	0.003461689	-0.77399030	-0.129423954	-0.360890431
4	-0.41128405	-0.427612718	-1.86024469	-1.385362401	-0.692129879
5	0.38172408	0.139478384	-0.39676732	-1.529183702	1.336460975

Showing 1 to 5 of 100 entries, 5 total columns

.
.
.

	X1	X2	X3	X4	X5
1	0.149824256	-0.67002494	-0.54638694	-1.38094408	0.64918987
2	-1.362787118	-0.40986249	0.17345962	1.24891999	0.42158883
3	-0.232239792	-1.40392241	1.23457223	1.15818469	0.22964145
4	-0.779339811	-0.45012640	-0.90956509	0.84330240	1.08990434
5	1.344412144	-1.91991909	-0.59539670	0.51074026	1.55397090

Showing 1 to 5 of 100 entries, 5 total columns



	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	0	0	0	0
[2,]	0	0	1	0	0
[3,]	0	1	0	0	0
[4,]	0	0	0	0	0
[5,]	0	0	1	1	0

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	0	1	1	1
[2,]	0	0	0	0	0
[3,]	1	0	0	1	1
[4,]	1	0	1	0	1
[5,]	1	0	1	1	0

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	1	0	1	0
[2,]	1	0	0	1	0
[3,]	0	0	0	0	0
[4,]	1	1	0	0	0
[5,]	0	1	1	0	0

.
.
.

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	1	1	0	1
[2,]	1	0	1	0	0
[3,]	1	1	0	1	1
[4,]	0	0	1	0	0
[5,]	1	0	1	0	0

SLdisco: Supervised learning discovery

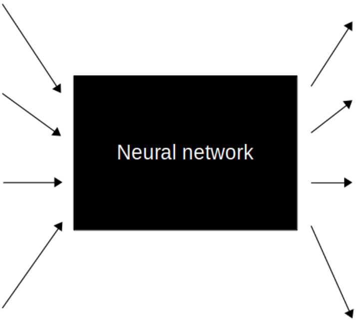
	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	1.0000	0.0216	-0.0240	0.0179	0.0020
[2,]	0.0216	1.0000	-0.2318	-0.0100	0.0738
[3,]	-0.0240	-0.2318	1.0000	-0.0089	-0.2486
[4,]	0.0179	-0.0100	-0.0089	1.0000	0.5613
[5,]	0.0020	0.0738	-0.2486	0.5613	1.0000

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	1.0000	0.0319	0.2786	0.9212	-0.8851
[2,]	0.0319	1.0000	0.0024	0.0335	-0.0330
[3,]	0.2786	0.0024	1.0000	0.0216	0.1064
[4,]	0.9212	0.0335	0.0216	1.0000	-0.9547
[5,]	-0.8851	-0.0330	0.1064	-0.9547	1.0000

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	1.0000	0.3571	-0.0257	0.5526	0.1212
[2,]	0.3571	1.0000	0.0088	-0.0198	0.3726
[3,]	-0.0257	0.0088	1.0000	-0.0314	0.4080
[4,]	0.5526	-0.0198	-0.0314	1.0000	-0.0182
[5,]	0.1212	0.3726	0.4080	-0.0182	1.0000

.
.
.

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	1.0000	-0.8751	0.5641	0.2042	0.3638
[2,]	-0.8751	1.0000	-0.7438	-0.2758	-0.1741
[3,]	0.5641	-0.7438	1.0000	0.3764	-0.1916
[4,]	0.2042	-0.2758	0.3764	1.0000	-0.0620
[5,]	0.3638	-0.1741	-0.1916	-0.0620	1.0000



	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	0	0	0	0
[2,]	0	0	1	0	0
[3,]	0	1	0	0	0
[4,]	0	0	0	0	0
[5,]	0	0	1	1	0

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	0	1	1	1
[2,]	0	0	0	0	0
[3,]	1	0	0	1	1
[4,]	1	0	1	0	1
[5,]	1	0	1	1	0

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	1	0	1	0
[2,]	1	0	0	1	0
[3,]	0	0	0	0	0
[4,]	1	1	0	0	0
[5,]	0	1	1	0	0

.
.
.

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0	1	1	0	1
[2,]	1	0	1	0	0
[3,]	1	1	0	1	1
[4,]	0	0	1	0	0
[5,]	1	0	1	0	0

Data simulation

Procedure:

1. Construct DAG with randomly drawn density (0-80% missing edges compared to fully connected)
2. Simulate linear Gaussian data according to the DAG with randomly drawn residual variances and regression coefficients, compute correlation matrix
   $\rightarrow$ features
3. Construct CPDAG adjacency matrix corresponding to the DAG
   $\rightarrow$ labels

Orders of variables (columns/rows in matrices) are randomly permuted before training.

Training and testing

We consider all combinations of the following settings:

No. nodes: $p \in \{5, 10, 20\}$

Sample size per correlation matrix:
$n \in \{50, 100, 500, 1000, 5000, 10000, 50000\}$

Threshold: $\tau \in \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}$

We use a very simple convolutional neural network architecture.

All networks are trained on $b_{\text{train}} = 1,000,000$ observations, and evaluated on $b_{\text{test}} = 5000$ observations.

We compare with GES with BIC-type scores and PC with varying significance level $\alpha \in \{10^{-8}, 10^{-4}, 10^{-3}, 0.01, 0.5, 0.1, 0.2, 0.5, 0.8\}$.

Training and testing

We consider all combinations of the following settings:

No. nodes: $p \in \{5, 10, 20\}$

Sample size per correlation matrix:
$n \in \{50, 100, 500, 1000, 5000, 10000, 50000\}$

Threshold: $\tau \in \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}$

We use a very simple convolutional neural network architecture.

All networks are trained on $b_{\text{train}} = 1,000,000$ observations, and evaluated on $b_{\text{test}} = 5000$ observations.

We compare with GES with BIC-type scores and PC with varying significance level $\alpha \in \{10^{-8}, 10^{-4}, 10^{-3}, 0.01, 0.5, 0.1, 0.2, 0.5, 0.8\}$.

Training and testing

We consider all combinations of the following settings:

No. nodes: $p \in \{5, 10, 20\}$

Sample size per correlation matrix:
$n \in \{50, 100, 500, 1000, 5000, 10000, 50000\}$

Threshold: $\tau \in \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}$

We use a very simple convolutional neural network architecture.

All networks are trained on $b_{\text{train}} = 1,000,000$ observations, and evaluated on $b_{\text{test}} = 5000$ observations.

We compare with GES with BIC-type scores and PC with varying significance level $\alpha \in \{10^{-8}, 10^{-4}, 10^{-3}, 0.01, 0.5, 0.1, 0.2, 0.5, 0.8\}$.

Training and testing

We consider all combinations of the following settings:

No. nodes: $p \in \{5, 10, 20\}$

Sample size per correlation matrix:
$n \in \{50, 100, 500, 1000, 5000, 10000, 50000\}$

Threshold: $\tau \in \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}$

We use a very simple convolutional neural network architecture.

All networks are trained on $b_{\text{train}} = 1,000,000$ observations, and evaluated on $b_{\text{test}} = 5000$ observations.

We compare with GES with BIC-type scores and PC with varying significance level $\alpha \in \{10^{-8}, 10^{-4}, 10^{-3}, 0.01, 0.5, 0.1, 0.2, 0.5, 0.8\}$.

Evaluation metrics

		Actual Class	
		Positive (**P**)	Negative (**N**)
Predicted Class	Positive (**P**)	True Positive (**TP**)	False Positive (**FP**)
	Negative (**N**)	False Negative (**FN**)	True Negative (**TN**)

Adjacency metrics:

Negative predictive value: $\frac{TN}{TN+FN}$ *(conservativeness)*

F1 score: $2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision}+\text{recall}}$ *(informativeness)*

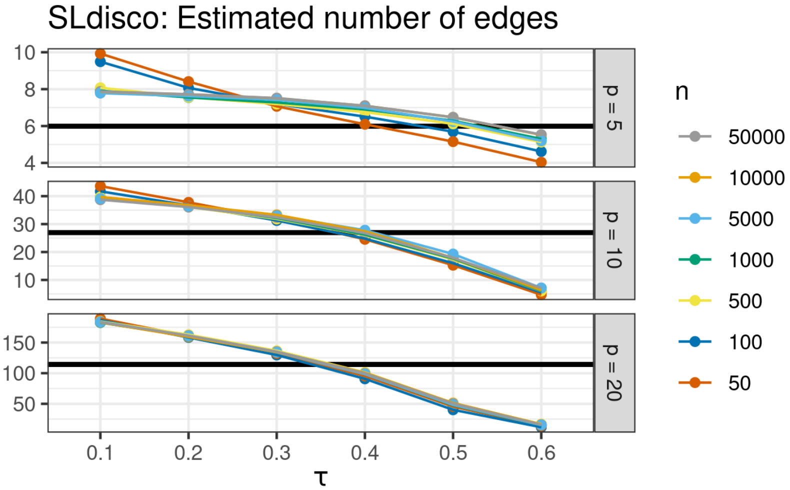
Conditional orientation metrics:

Precision: (= positive predictive value) $\frac{TP}{TP+FP}$ *(conservativeness)*

"G1" score: $2 \cdot \frac{\text{NPV} \cdot \text{specificity}}{\text{NPV}+\text{specificity}}$ *(informativeness)*

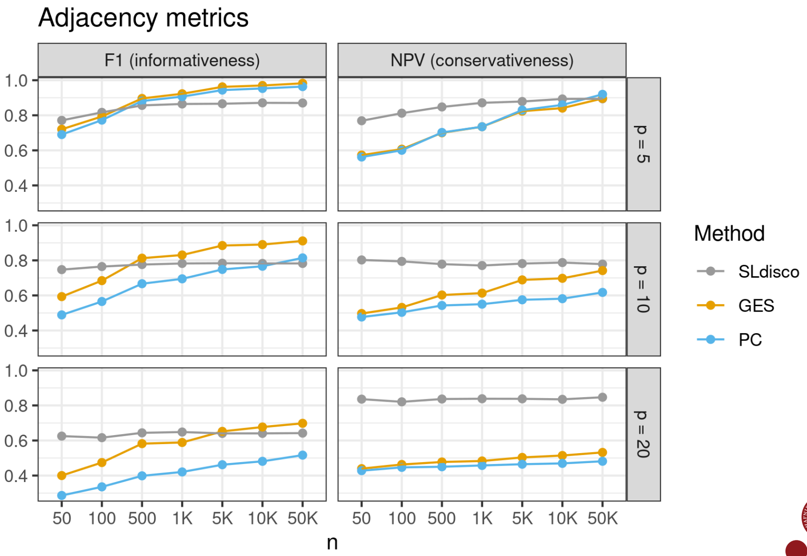
Results: Simulation study

Estimated number of edges

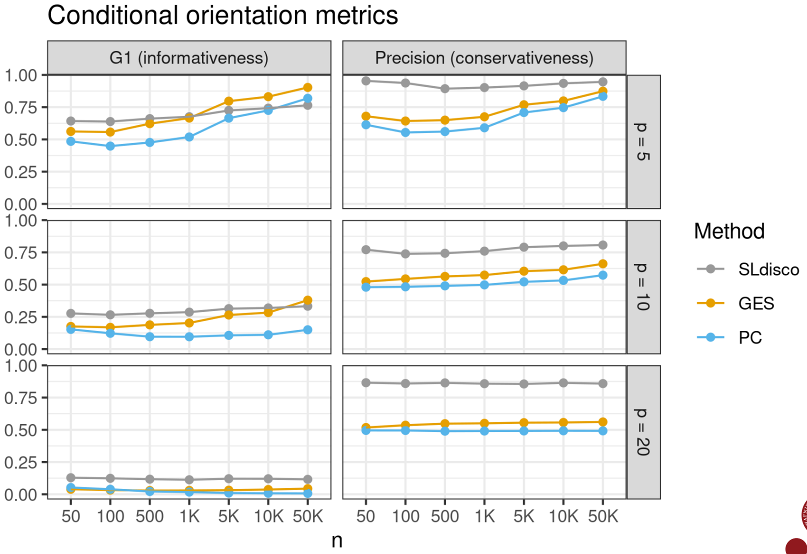


Use $\tau = 0.4$ for $p = 5$, and $\tau = 0.3$ for $p \in \{10, 20\}$.

Adjacency results



Orientation results



Application

Metropolit cohort dataset[1]:

- Longitudinal life course epidemiological dataset
- Follows $n = 2928$ Danish men from their birth in 1953 until 2018
- Retrospective: Condition on being alive at follow-up in 2018
- We use a subset of $p = 10$ variables

[1] Osler, Lund, Kriegbaum, Christensen, & Andersen (2006). Cohort profile: the Metropolit 1953 Danish male birth cohort. International Journal of Epidemiology.

Application

Metropolit cohort dataset[1]:

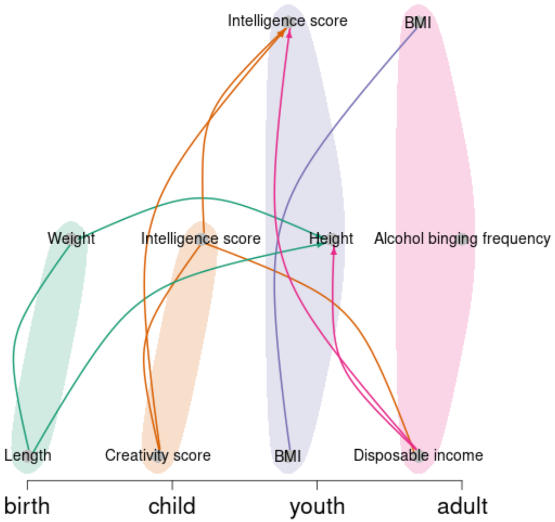
- Longitudinal life course epidemiological dataset
- Follows $n = 2928$ Danish men from their birth in 1953 until 2018
- Retrospective: Condition on being alive at follow-up in 2018
- We use a subset of $p = 10$ variables

We discuss validity on real data in two ways:

1. How plausible is the estimated CPDAG?
2. How stable is it towards random subsampling (smaller n)?

[1] Osler, Lund, Kriegbaum, Christensen, & Andersen (2006). Cohort profile: the Metropolit 1953 Danish male birth cohort. International Journal of Epidemiology.

Metropolit CPDAG: SLdisco (BPCO with $\tau = 0.4$)



Metropolit subsampling stability

"Ground truth": Model estimated using full data ($n = 2928$).

Method	Subsample n	Adj. F1	Adj. NPV	Ori. G1	Ori. prec.
SLdisco	50	0.67	0.88	0.75	1.00
	100	0.67	0.88	0.75	1.00
	500	0.89	0.95	0.55	1.00
	1000	0.95	0.97	0.80	1.00

Metropolit subsampling stability

"Ground truth": Model estimated using full data ($n = 2928$).

Method	Subsample n	Adj. F1	Adj. NPV	Ori. G1	Ori. prec.
SLdisco	50	0.67	0.88	0.75	1.00
	100	0.67	0.88	0.75	1.00
	500	0.89	0.95	0.55	1.00
	1000	0.95	0.97	0.80	1.00
PC	50	0.53	0.78	0.33	1.00
	100	0.53	0.78	0.33	1.00
	500	0.72	0.85	0.20	0.50
	1000	0.75	0.86	0.33	0.71

Metropolit subsampling stability

"Ground truth": Model estimated using full data ($n = 2928$).

Method	Subsample n	Adj. F1	Adj. NPV	Ori. G1	Ori. prec.
SLdisco	50	0.67	0.88	0.75	1.00
	100	0.67	0.88	0.75	1.00
	500	0.89	0.95	0.55	1.00
	1000	0.95	0.97	0.80	1.00
PC	50	0.53	0.78	0.33	1.00
	100	0.53	0.78	0.33	1.00
	500	0.72	0.85	0.20	0.50
	1000	0.75	0.86	0.33	0.71
GES	50	0.56	0.82	0.33	1.00
	100	0.67	0.85	0.29	1.00
	500	0.64	0.86	0.00	1.00
	1000	0.76	0.89	0.00	0.25

Conclusion

SLdisco addresses the two issues:

Error tradeoff: More conservative, only modestly less informative

Sample size: Better small/moderate sample performance

Limitations and next steps

- Looks like we may be **overfitting** for large n
- May be sensitive towards **Gaussianity assumption**
- Some initial **computation time** for training models (but only has to be done once per n-p combination, and fine-tuning of pretrained model could be helpful)
- Assumes **causal sufficiency** (no unobserved confounders)
- Not **permutation equivariant** (variable ordering matters)
- More sophisticated/tailored **machine learning** (NN architecture and training setup) could be interesting
- **Time series** or other specialized data structures could be accommodated easily as 3D/4D/... feature data

Want to know more?

Article: Petersen, Ramsey, Ekstrøm & Spirtes (2022). Causal discovery for observational sciences using supervised machine learning. arXiv:2202.12813.

Code og pretrained models:
https://github.com/annennenne/SLdisco

R package: `causalDisco` - on CRAN

Or reach out at: ahpe@sund.ku.dk