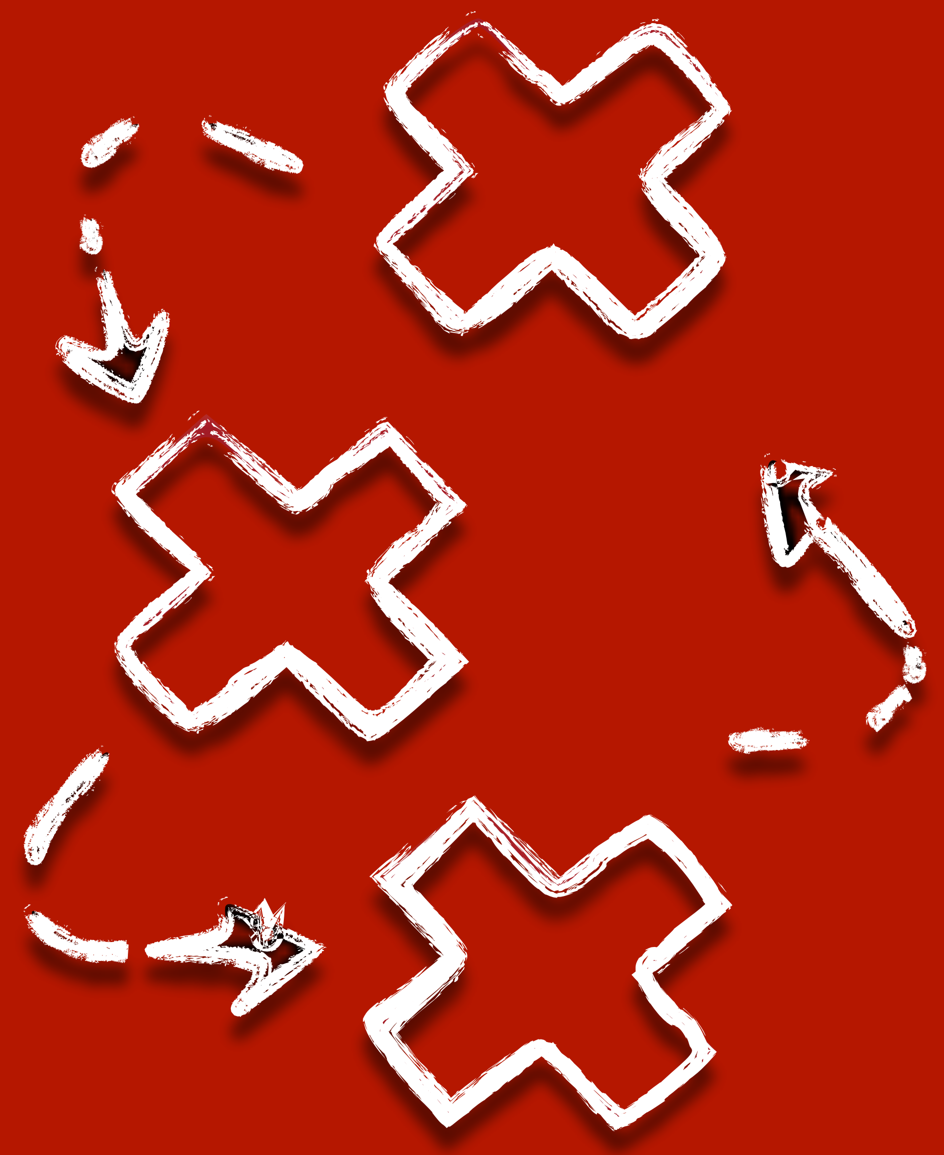


Deep learning 2: Causality & DL

2.3: Causality-inspired ML

Lecturer: Sara Magliacane

UvA - Spring 2022

Causal Hierarchy [Pearl 2009, 2018]

Most ML

Causality

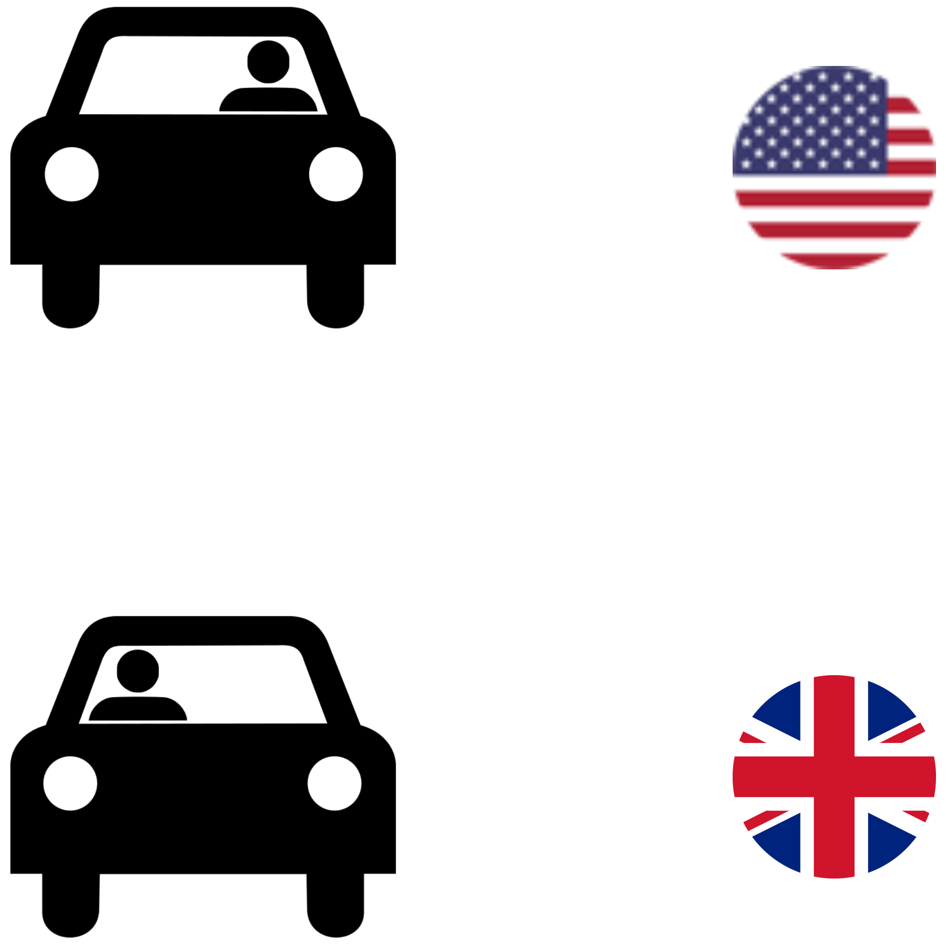
Level (Symbol)	Typical Activity	Typical Questions	Examples
1. Association $P(y\|x)$	Seeing	What is? How would seeing X change my belief in Y?	What does a symptom tell me about a disease? What does a survey tell us about the election results?
2. Intervention $P(y\|do(x), z)$	Doing Intervening	What if? What if I do X?	What if I take aspirin, will my headache be cured? What if we ban cigarettes?
3. Counterfactuals $P(y_x\|x', y')$	Imagining, Retrospection	Why? Was it X that caused Y? What if I had acted differently?	Was it the aspirin that stopped my headache? Would Kennedy be alive had Oswald not shot him? What if I had not been smoking the past 2 years?

CAUSALITY-INSPIRED ML
(not necessarily trying to reconstruct causal relations)

→ TRANSFER LEARNING / DISTRIBUTION SHIFTS
RL

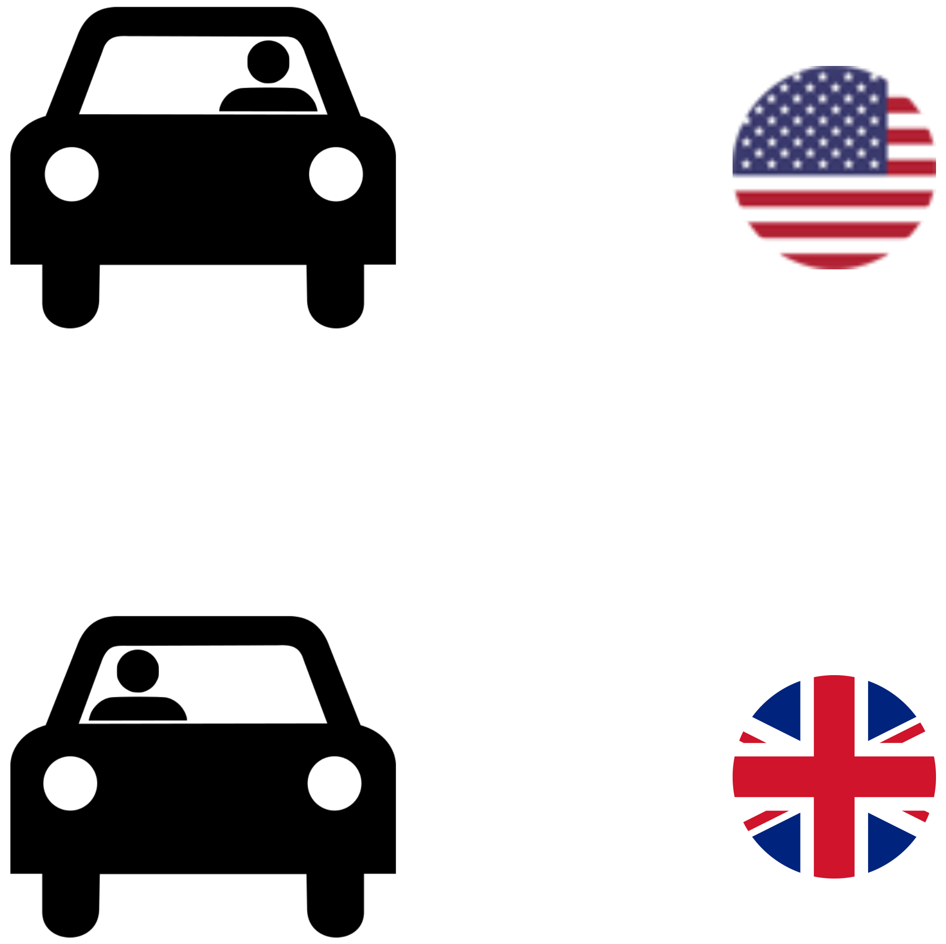
Causality vs Transfer learning

- Transfer learning:
  - How can I predict what happens when the distribution changes?



Causality vs Transfer learning

- Transfer learning:
  - How can I predict what happens when the distribution changes?



- Causal inference:
  - How can I predict what happens when the distribution changes after an intervention?
    - Perfect intervention do(X):
      - do-calculus [Pearl, 2009]
    - **Soft intervention on X** ≈ change of distribution of P(X| parents)

Causality allows us to reason systematically about distribution shifts

On Causal and Anticausal Learning

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang FIRST.LAST@TUE.MPG.DE
Max Planck Institute for Intelligent Systems, Spemannstrasse, 72076 Tübingen, Germany
Joris Mooij J.MOOIJ@CS.RU.NL
Institute for Computing and Information Sciences, Radboud University, Nijmegen, The Netherlands

Domain Adaptation as a Problem of Inference on Graphical Models

Kun Zhang[1*], **Mingming Gong**[2*], **Petar Stojanov**[3]
Biwei Huang[1], **Qingsong Liu**[4], **Clark Glymour**[1]
[1] Department of philosophy, Carnegie Mellon University
[2] School of Mathematics and Statistics, University of Melbourne
[3] Computer Science Department, Carnegie Mellon University, [4] Unisound AI Lab
kunz1@cmu.edu, mingming.gong@unimelb.edu.au, liuqingsong@unisound.com
{pstojano, biweih, cg09}@andrew.cmu.edu

Anchor regression: heterogeneous data meet causality

Dominik Rothenhäusler, Nicolai Meinshausen, Peter Bühlmann and Jonas Peters

Invariant Risk Minimization

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, David Lopez-Paz

J. R. Statist. Soc. B (2016)
78, *Part* 5, *pp.* 947–1012

Causal inference by using invariant prediction: identification and confidence intervals

Jonas Peters
Max Planck Institute for Intelligent Systems, Tübingen, Germany, and Eidgenössische Technische Hochschule Zürich, Switzerland
and Peter Bühlmann and Nicolai Meinshausen
Eidgenössische Technische Hochschule Zürich, Switzerland

Invariant Models for Causal Transfer Learning

Mateo Rojas-Carulla MR597@CAM.AC.UK
Max Planck Institute for Intelligent Systems
Tübingen, Germany
Department of Engineering
Univ. of Cambridge, United Kingdom
Bernhard Schölkopf BS@TUEBINGEN.MPG.DE
Max Planck Institute for Intelligent Systems
Tübingen, Germany
Richard Turner RET26@CAM.AC.UK
Department of Engineering
Univ. of Cambridge, United Kingdom
Jonas Peters* JONAS.PETERS@MATH.KU.DK
Department of Mathematical Sciences
Univ. of Copenhagen, Denmark

Invariance, Causality and Robustness

2018 Neyman Lecture *

Peter Bühlmann †
Seminar for Statistics, ETH Zürich

Counterfactual Invariance to Spurious Correlations: Why and How to Pass Stress Tests

Victor Veitch[1,2], Alexander D'Amour[1], Steve Yadlowsky[1], and Jacob Eisenstein[1]
[1]*Google Research*
[2]*University of Chicago*

Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions

Sara Magliacane
IBM Research*
sara.magliacane@gmail.com
Thijs van Ommen
University of Amsterdam
thijsvanommen@gmail.com
Tom Claassen
Radboud University Nijmegen
tomc@cs.ru.nl
Stephan Bongers
University of Amsterdam
srbongers@gmail.com
Philip Versteeg
University of Amsterdam
p.j.j.p.versteeg@uva.nl
Joris M. Mooij
University of Amsterdam
j.m.mooij@uva.nl

A Causal View on Robustness of Neural Networks

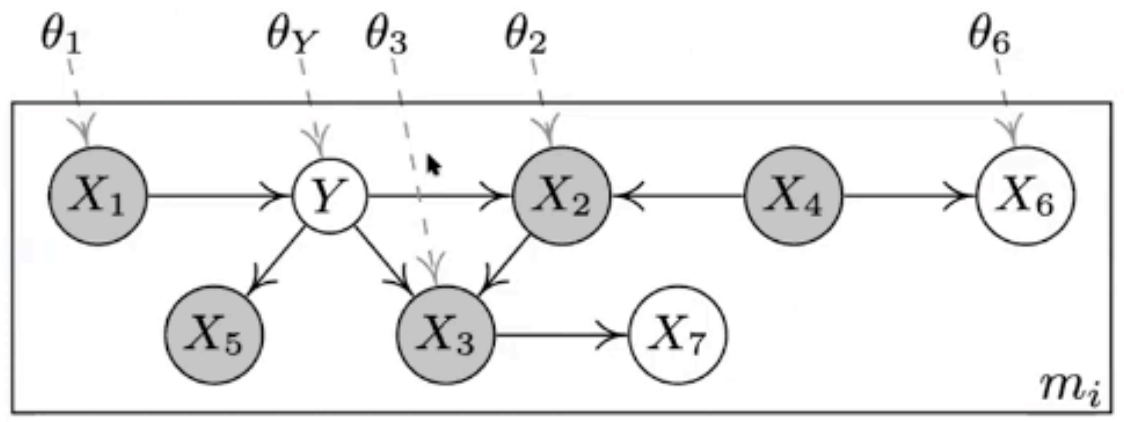
Cheng Zhang *
Microsoft Research
Cheng.Zhang@microsoft.com
Kun Zhang
Carnegie Mellon University
kunz1@cmu.edu
Yingzhen Li *
Microsoft Research
Yingzhen.Li@microsoft.com

and many many more….

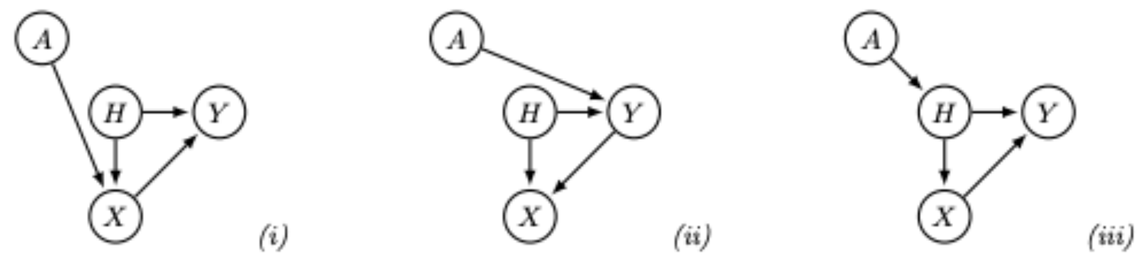
Causality allows us to reason systematically about distribution shifts, e.g. through graphs

On Causal and Anticausal Learning

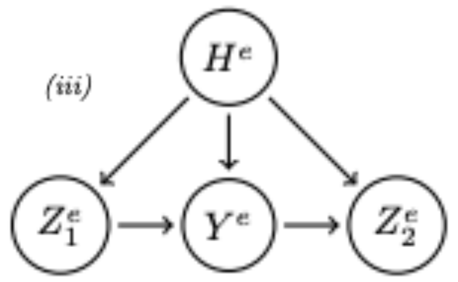
Domain Adaptation as a Problem of Inference on Graphical Models



Anchor regression: heterogeneous data meet causality

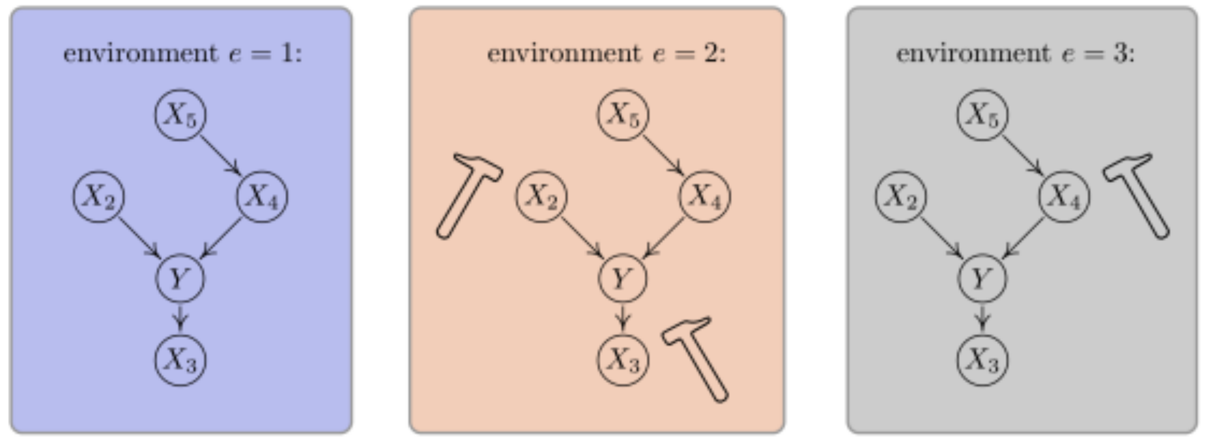


Invariant Risk Minimization

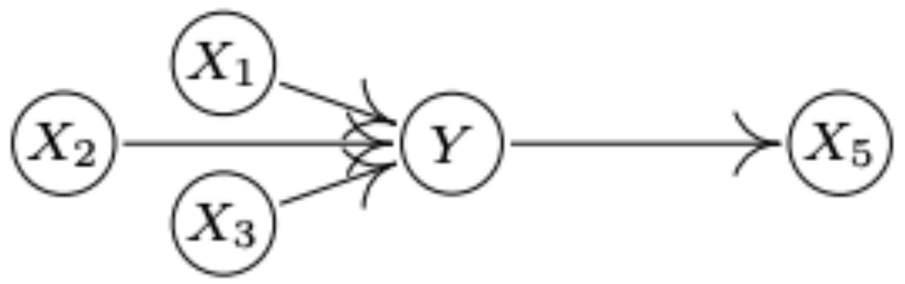


J. R. Statist. Soc. B (2016)
78, Part 5, pp. 947–1012

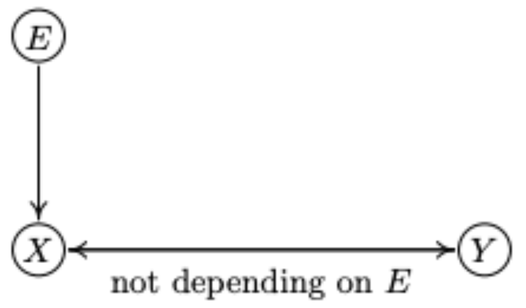
Causal inference by using invariant prediction: identification and confidence intervals



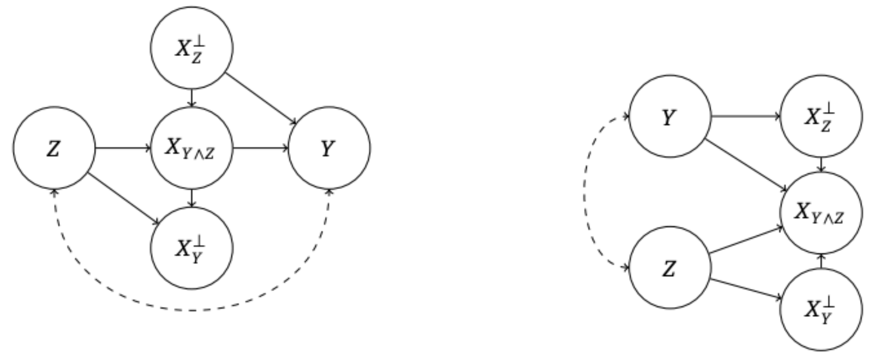
Invariant Models for Causal Transfer Learning



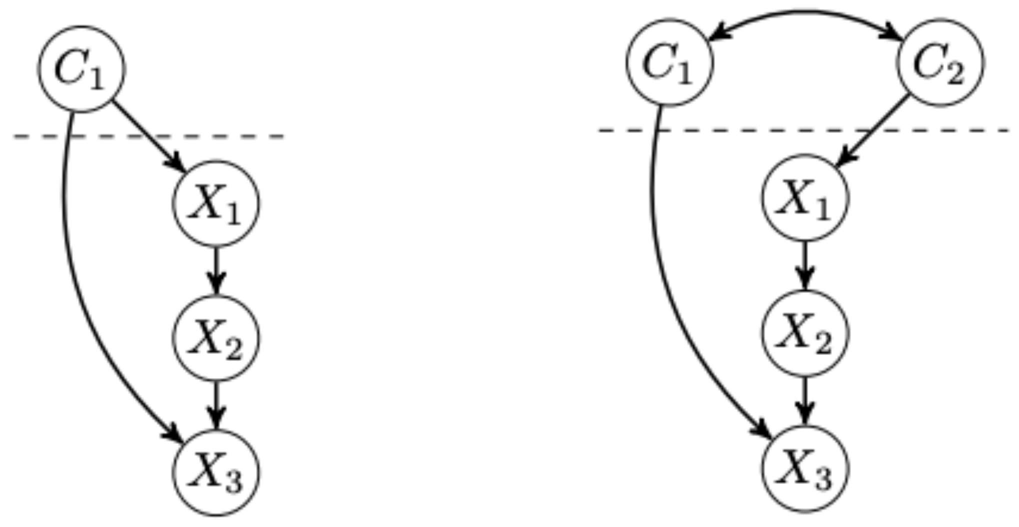
Invariance, Causality and Robustness



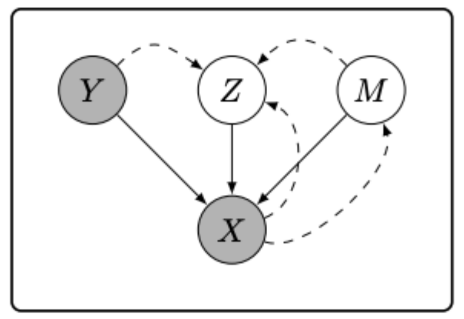
Counterfactual Invariance to Spurious Correlations: Why and How to Pass Stress Tests



Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions



A Causal View on Robustness of Neural Networks



and many more....

Causality allows us to reason systematically about distribution shifts, e.g. through graphs

A description of domain adaptation tasks:

- Supervised multi-source domain adaptation

C1	C2	X1	X2	X3	Y	X4
1	0	1200	1000	1500	-0.1	9
1	0	1201	800	1500	?	8
1	0	1195	200	1499	?	7
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

- Estimate $\hat{f}$ in Y = $\hat{f}$(X1, X2, X3, X4) from source domains and few labels in target domain

A description of domain adaptation tasks:

- **Unsupervised** multi-source domain adaptation

No labels in target

C1	C2	X1	X2	X3	Y	X4
1	0	1200	1000	1500	?	9
1	0	1201	800	1500	?	8
1	0	1195	200	1499	?	7
1	0	….	….	….	….	….
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	….	….	….	….	….
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	….

Target domain

Source domains

- Estimate $\hat{f}$ in Y = $\hat{f}$(X1, X2, X3, X4) from source domains and by exploiting the knowledge of the **change** from the **unlabelled data in target**

E.g. edges from C1 to X4

A description of domain adaptation tasks:

- **Domain generalisation:** required to work under **any intervention**

C1	C2	X1	X2	X3	Y	X4
1	0	?	?	?	?	?
1	0	?	?	?	?	?
1	0	?	?	?	?	?
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

No data in target

Target domain

Source domains

- Estimate $\hat{f}$ in $Y = \hat{f}(X1, X2, X3, X4)$ from source domains, no idea about what happens in the target

A description of domain adaptation tasks:

C1	C2	X1	X2	X3	Y	X4
1	0	1200	1000	1500	-0.1	9
1	0	1201	800	1500	?	8
1	0	1195	200	1499	?	7
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

C1	C2	X1	X2	X3	Y	X4
1	0	1200	1000	1500	?	9
1	0	1201	800	1500	?	8
1	0	1195	200	1499	?	7
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

C1	C2	X1	X2	X3	Y	X4
1	0	?	?	?	?	?
1	0	?	?	?	?	?
1	0	?	?	?	?	?
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

- We interpret the change in the target domain as a **(soft) intervention**
- **We assume Y cannot be intervened upon directly -** P(Y) can still change

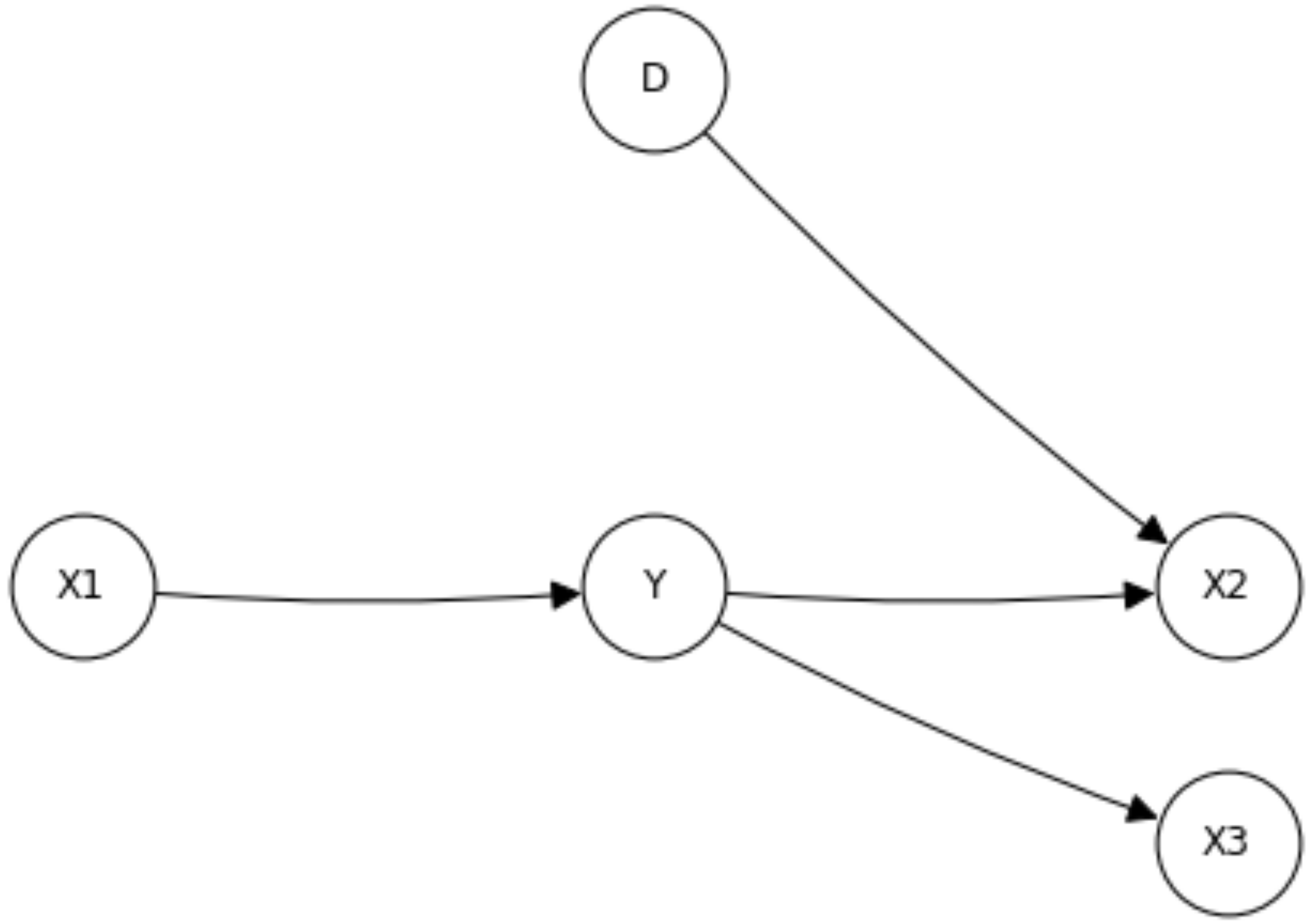
Structural causal model - domain/environment variable

```python
def linearSCM(n_samples, domain_number=0):
    epsilon_x1 = randn(n_samples)
    epsilon_y = randn(n_samples)
    epsilon_x2 = randn(n_samples)
    epsilon_x3 = randn(n_samples)

    x1 = epsilon_x1 + 10
    y = 3 * x1 + epsilon_y
    if domain_number==0:
        x2 = - 2 * y + epsilon_x2
    elif domain_number==1:
        x2 = 1
    else:
        x2 = 10 * y + epsilon_x2
    x3 = 2 * y + 0.1*epsilon_x3
    df = pd.DataFrame({"d": domain_number, "x1": x1, "y": y, "x2": x2, "x3": x3})
    return df
```

$\mathrm{do}(X_2 = 1)$

$\mathrm{do}(X_2 = f_2'(Y, \epsilon_{X_2}))$



Structural causal model - domain/environment variable

x1	y	x2	x3
8.973763	26.130494	-51.648475	52.330948
10.428340	31.894998	-64.373356	63.802704
8.911484	25.166962	-52.313502	50.279162
9.841798	29.783299	-60.419296	59.539914
8.969118	27.660573	-55.075839	55.327185

x1	y	x2	x3
9.941015	28.696601	1	57.475345
8.762380	25.715927	1	51.275390
9.636201	28.407387	1	56.884332
10.875069	31.370200	1	62.686789
10.023968	31.253540	1	62.388444

Source domains

$\mathrm{do}(X_2 = 1)$

x1	y	x2	x3
9.671277	26.556214	265.034283	53.338139
9.613139	27.120226	270.746784	54.340341
10.718335	29.589532	295.318526	59.291053
9.002388	26.629254	264.942583	53.340389
9.289340	29.030355	289.747562	58.098312

$\mathrm{do}(X_2 = f_2'(Y, \epsilon_{X_2}))$

Target domain

d	x1	y	x2	x3
0	8.973763	26.130494	-51.648475	52.330948
0	10.428340	31.894998	-64.373356	63.802704
0	8.911484	25.166962	-52.313502	50.279162
0	9.841798	29.783299	-60.419296	59.539914
0	8.969118	27.660573	-55.075839	55.327185

d	x1	y	x2	x3
1	9.941015	28.696601	1	57.475345
1	8.762380	25.715927	1	51.275390
1	9.636201	28.407387	1	56.884332
1	10.875069	31.370200	1	62.686789
1	10.023968	31.253540	1	62.388444

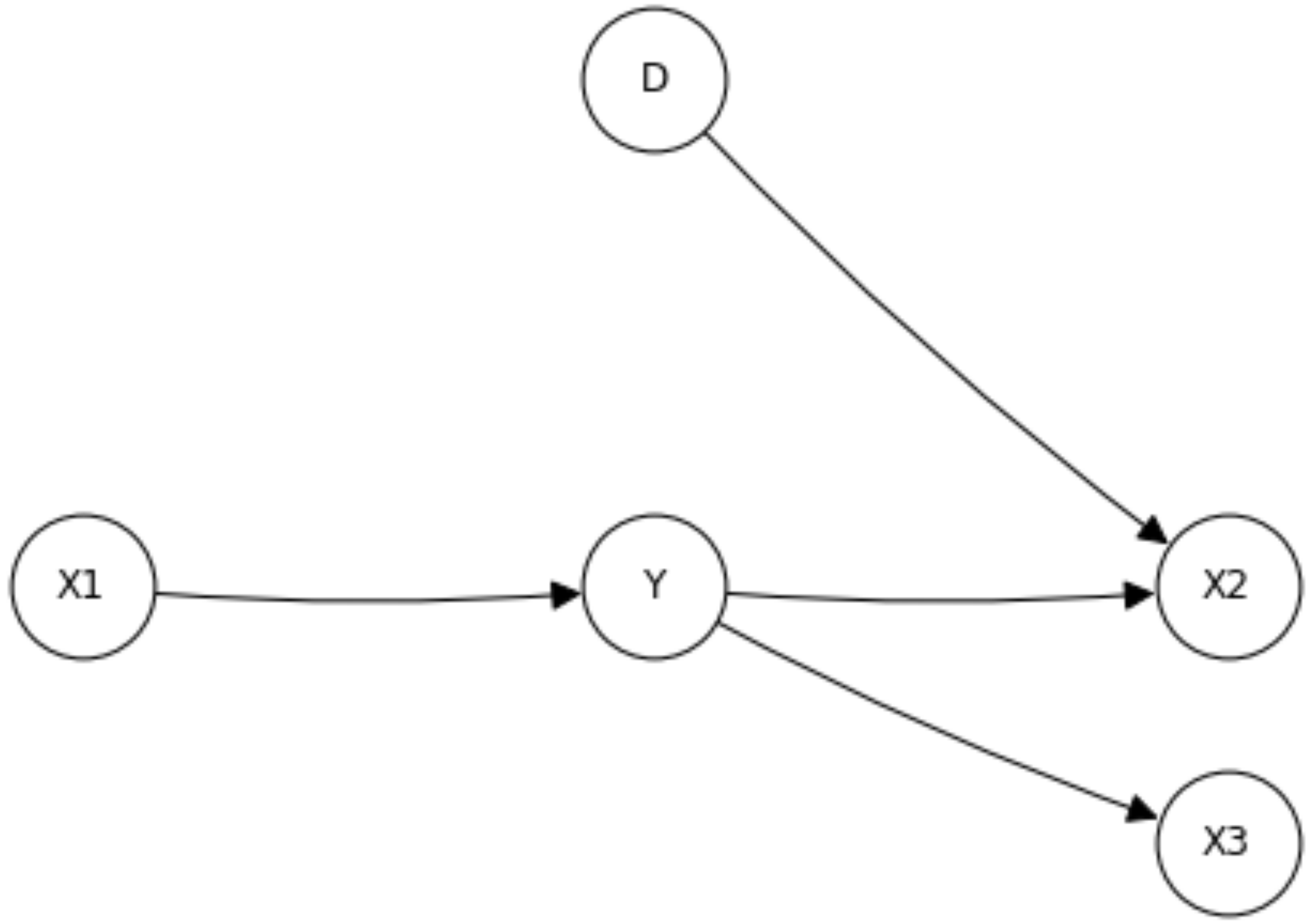
d	x1	y	x2	x3
2	9.671277	26.556214	265.034283	53.338139
2	9.613139	27.120226	270.746784	54.340341
2	10.718335	29.589532	295.318526	59.291053
2	9.002388	26.629254	264.942583	53.340389
2	9.289340	29.030355	289.747562	58.098312

Structural causal model - domain/environment variable

d	x1	y	x2	x3
0	8.973763	26.130494	-51.648475	52.330948
0	10.428340	31.894998	-64.373356	63.802704
0	8.911484	25.166962	-52.313502	50.279162
0	9.841798	29.783299	-60.419296	59.539914
0	8.969118	27.660573	-55.075839	55.327185

d	x1	y	x2	x3
1	9.941015	28.696601	1	57.475345
1	8.762380	25.715927	1	51.275390
1	9.636201	28.407387	1	56.884332
1	10.875069	31.370200	1	62.686789
1	10.023968	31.253540	1	62.388444

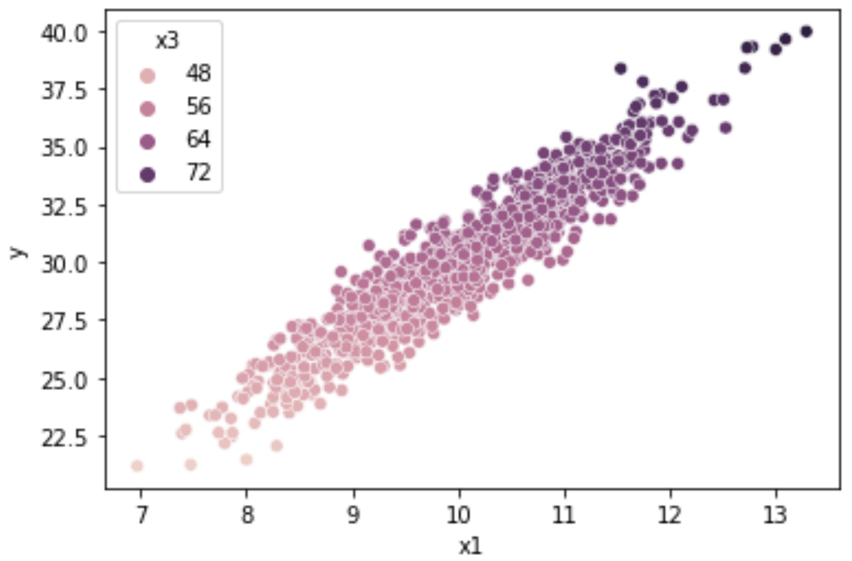
d	x1	y	x2	x3
2	9.671277	26.556214	265.034283	53.338139
2	9.613139	27.120226	270.746784	54.340341
2	10.718335	29.589532	295.318526	59.291053
2	9.002388	26.629254	264.942583	53.340389
2	9.289340	29.030355	289.747562	58.098312



$p(Y|X_1)$ is invariant

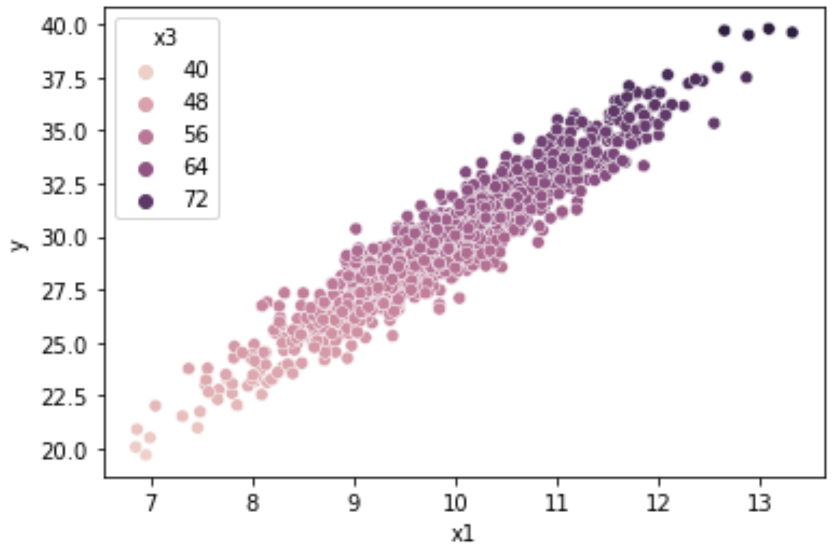
```
sns.scatterplot(data = df_0, x="x1", y="y", hue="x3")
```

<AxesSubplot:xlabel='x1', ylabel='y'>



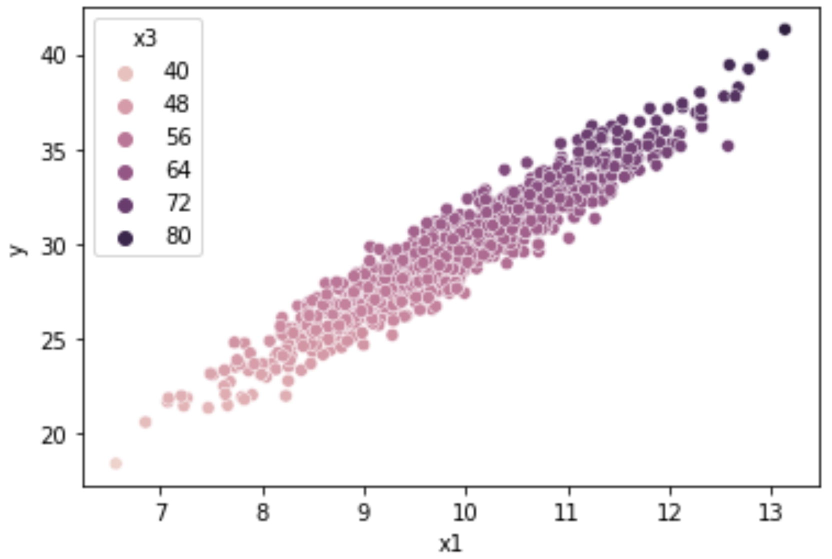

```
sns.scatterplot(data = df_1, x="x1", y="y", hue="x3")
```

<AxesSubplot:xlabel='x1', ylabel='y'>




```
sns.scatterplot(data = df_2, x="x1", y="y", hue="x3")
```

<AxesSubplot:xlabel='x1', ylabel='y'>

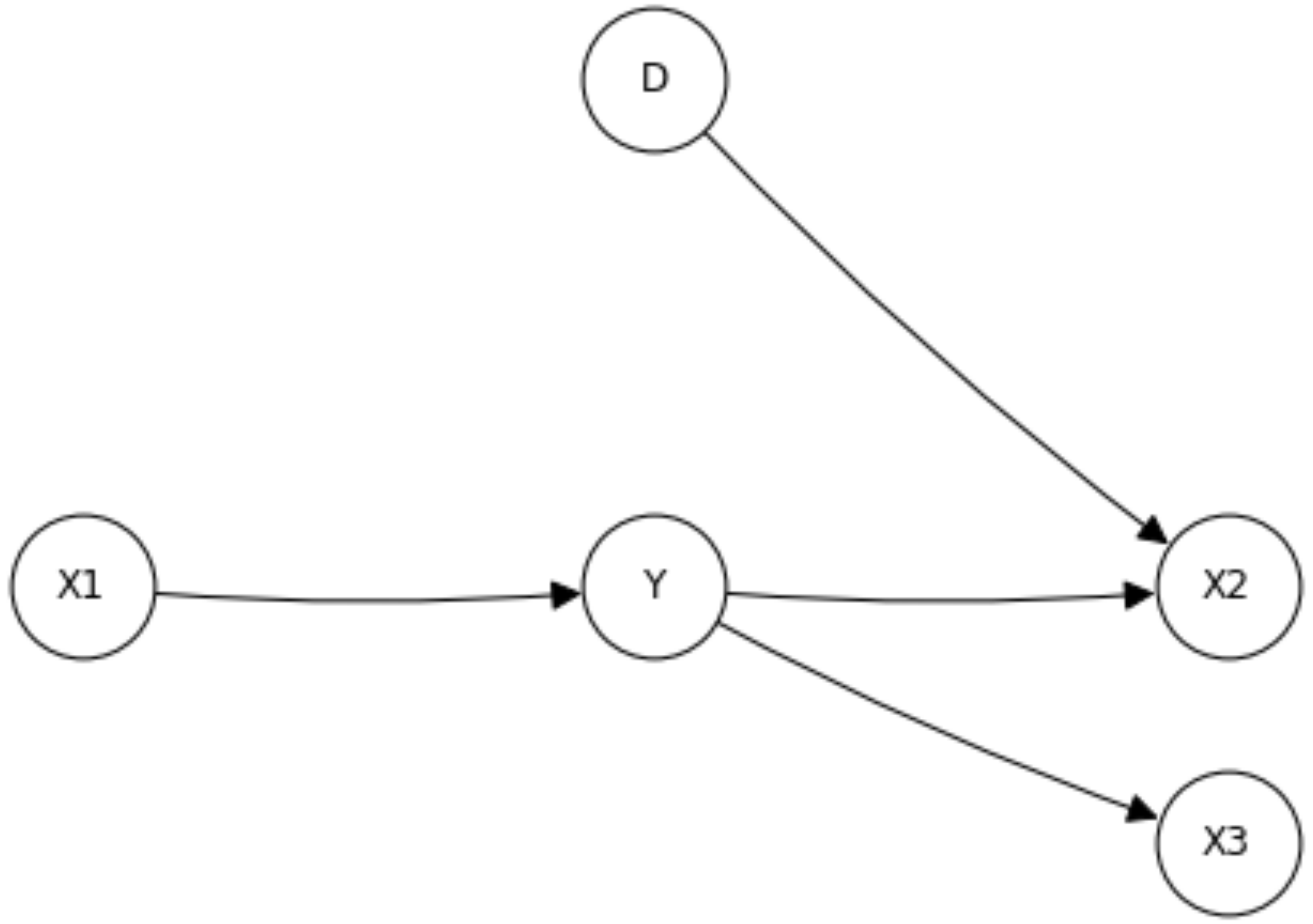


Structural causal model - domain/environment variable

d	x1	y	x2	x3
0	8.973763	26.130494	-51.648475	52.330948
0	10.428340	31.894998	-64.373356	63.802704
0	8.911484	25.166962	-52.313502	50.279162
0	9.841798	29.783299	-60.419296	59.539914
0	8.969118	27.660573	-55.075839	55.327185

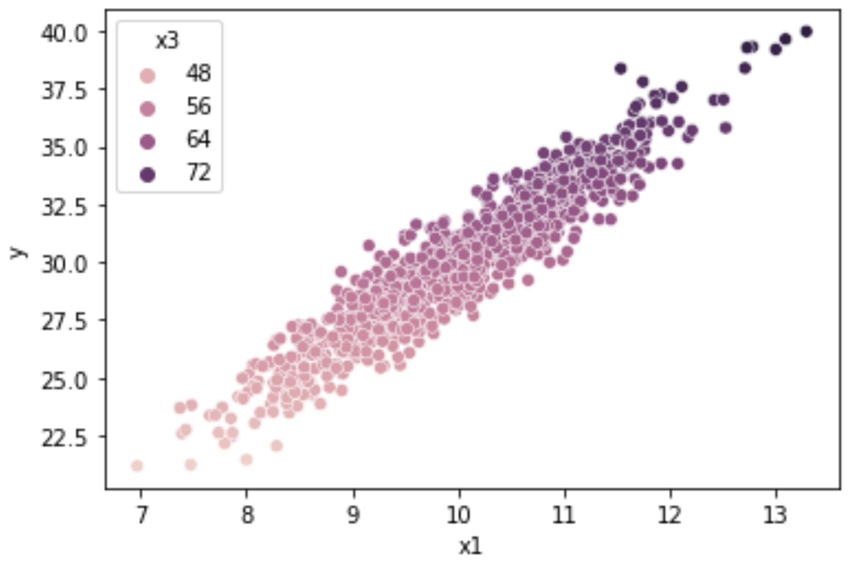
d	x1	y	x2	x3
1	9.941015	28.696601	1	57.475345
1	8.762380	25.715927	1	51.275390
1	9.636201	28.407387	1	56.884332
1	10.875069	31.370200	1	62.686789
1	10.023968	31.253540	1	62.388444

d	x1	y	x2	x3
2	9.671277	26.556214	265.034283	53.338139
2	9.613139	27.120226	270.746784	54.340341
2	10.718335	29.589532	295.318526	59.291053
2	9.002388	26.629254	264.942583	53.340389
2	9.289340	29.030355	289.747562	58.098312



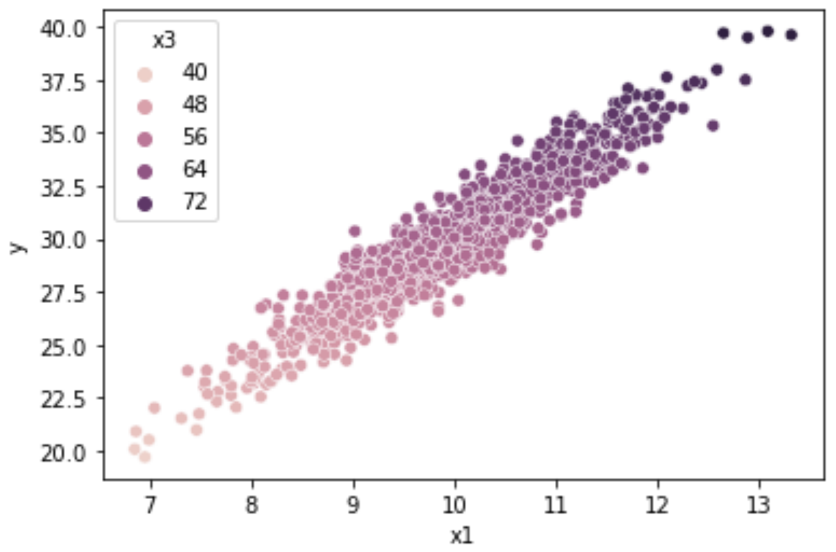

```
sns.scatterplot(data = df_0, x="x1", y="y", hue="x3")
```

<AxesSubplot:xlabel='x1', ylabel='y'>




```
sns.scatterplot(data = df_1, x="x1", y="y", hue="x3")
```

<AxesSubplot:xlabel='x1', ylabel='y'>




```
Y_0 = df_0["y"].values.reshape(-1, 1)
Y_2 = df_2["y"].values.reshape(-1, 1)
X1_0 = df_0["x1"].values.reshape(-1, 1)
X1_2 = df_2["x1"].values.reshape(-1, 1)
model = LinearRegression().fit(X1_0, Y_0)
est_Y_2 = model.predict(X1_2)
print("Mean squared error predicting Y in environment 2 based on model learnt in environment 0 from X1", mean_squared_error(Y_2,est_Y_2))
```

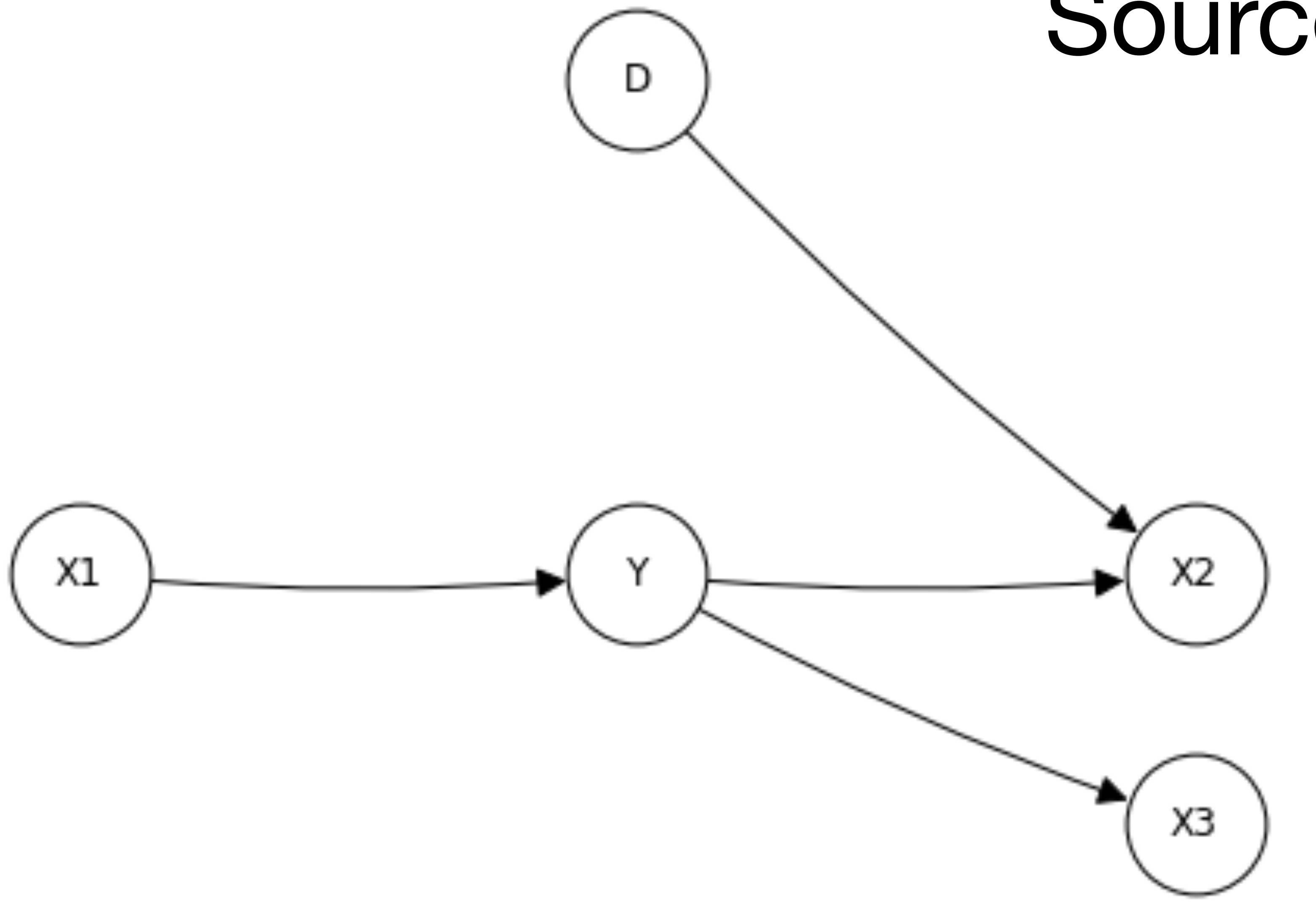
```
Mean squared error predicting Y in environment 2 based on model learnt in environment 0 from X1 0.9336539410357941
```

Structural causal model - domain/environment variable

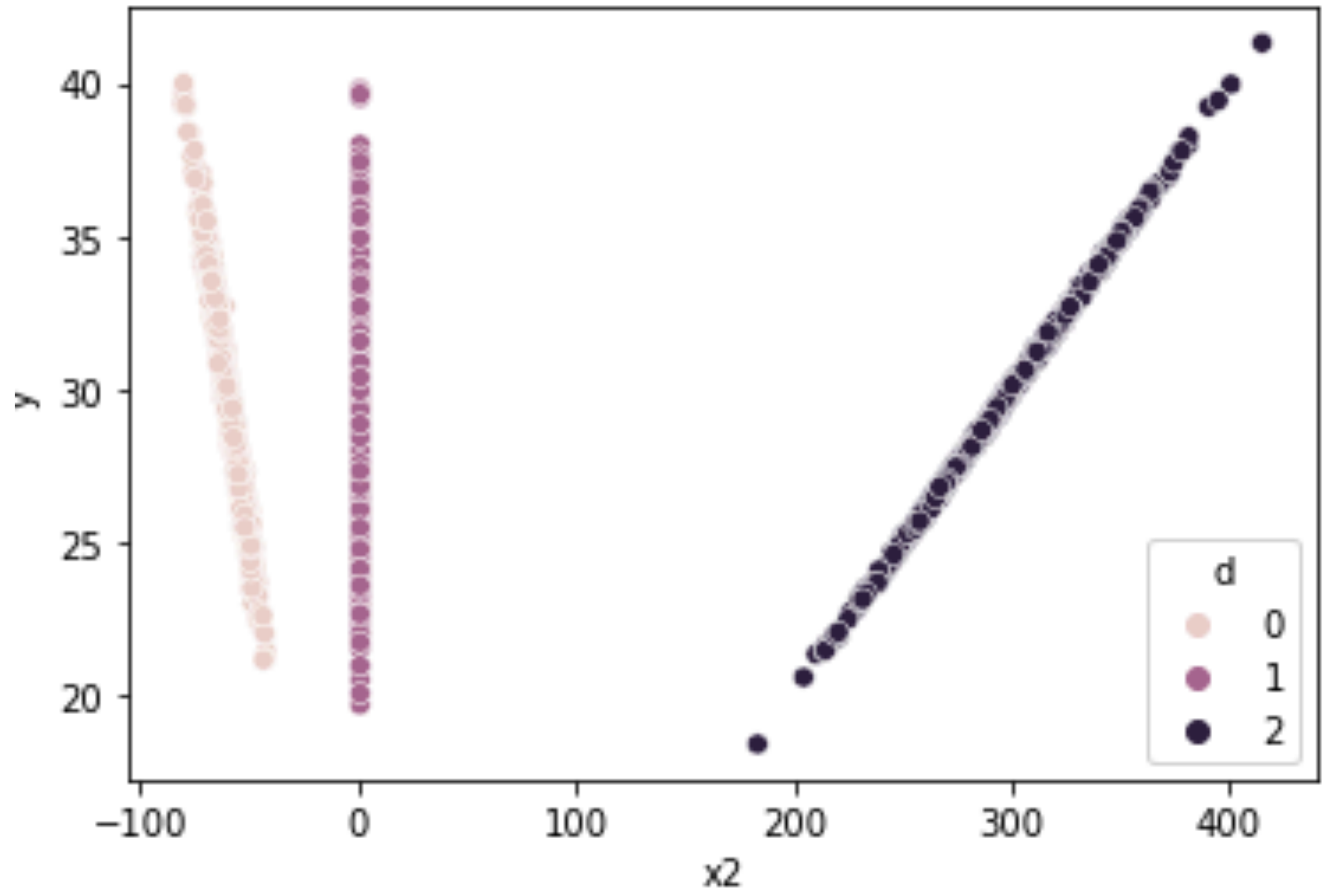
d	x1	y	x2	x3
0	8.973763	26.130494	-51.648475	52.330948
0	10.428340	31.894998	-64.373356	63.802704
0	8.911484	25.166962	-52.313502	50.279162
0	9.841798	29.783299	-60.419296	59.539914
0	8.969118	27.660573	-55.075839	55.327185

d	x1	y	x2	x3
1	9.941015	28.696601	1	57.475345
1	8.762380	25.715927	1	51.275390
1	9.636201	28.407387	1	56.884332
1	10.875069	31.370200	1	62.686789
1	10.023968	31.253540	1	62.388444

d	x1	y	x2	x3
2	9.671277	26.556214	265.034283	53.338139
2	9.613139	27.120226	270.746784	54.340341
2	10.718335	29.589532	295.318526	59.291053
2	9.002388	26.629254	264.942583	53.340389
2	9.289340	29.030355	289.747562	58.098312



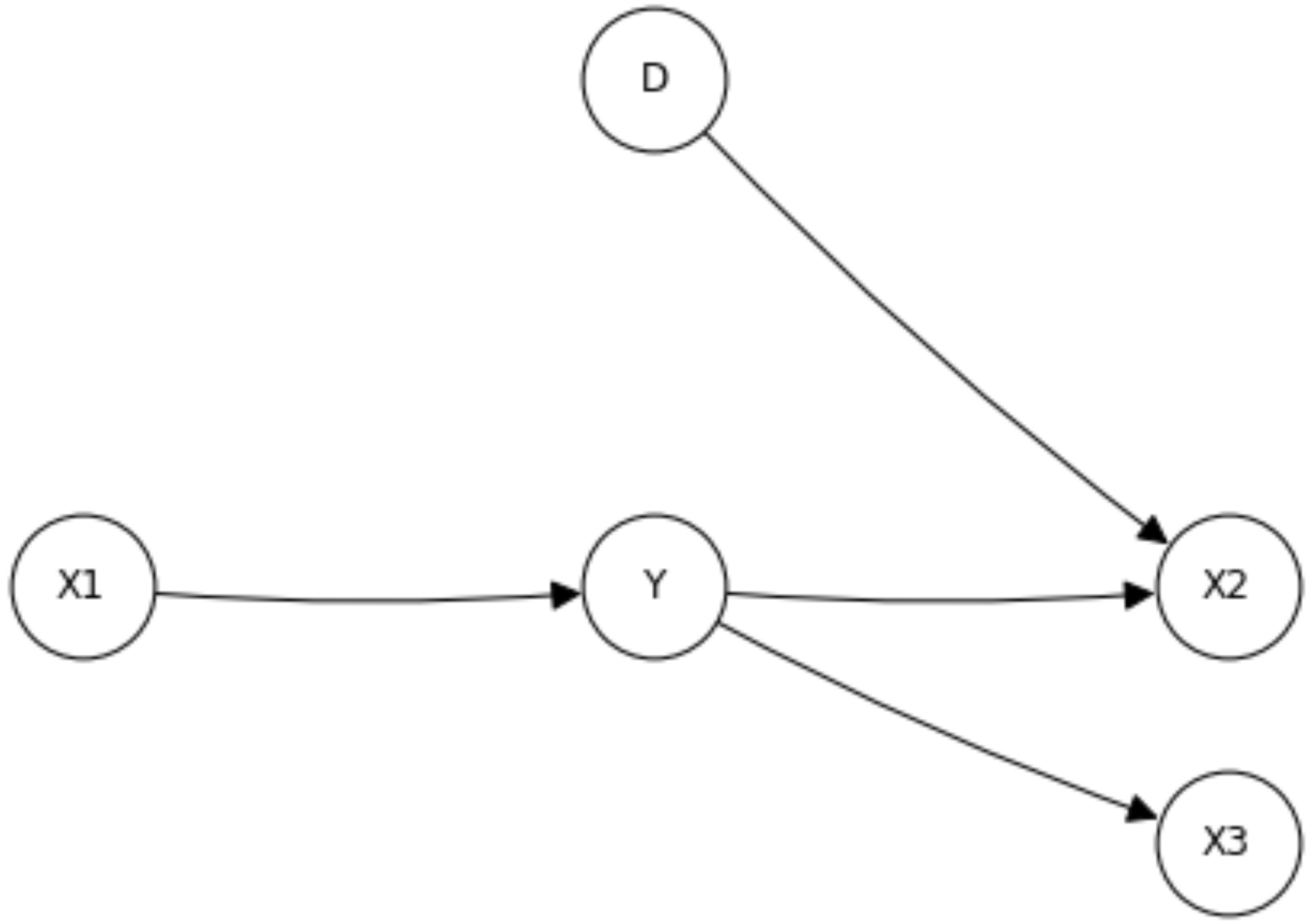
Source domains   Target domain




```
sns.scatterplot(data = df, x="x2", y="y", hue="d")
X2_0 = df_0["x2"].values.reshape(-1, 1)
X2_2 = df_2["x2"].values.reshape(-1, 1)
model = LinearRegression().fit(X2_0, Y_0)
est_Y_2 = model.predict(X2_2)
print("Mean squared error predicting Y in environment 2 based on model learnt in environment 0 from X2", mean_squared_error(Y_2,est_Y_2))
```

```
Mean squared error predicting Y in environment 2 based on model learnt in environment 0 from X2 30518.374428658524
```

Separating features intuition



$P(X_1, Y, X_2, X_3, D)$

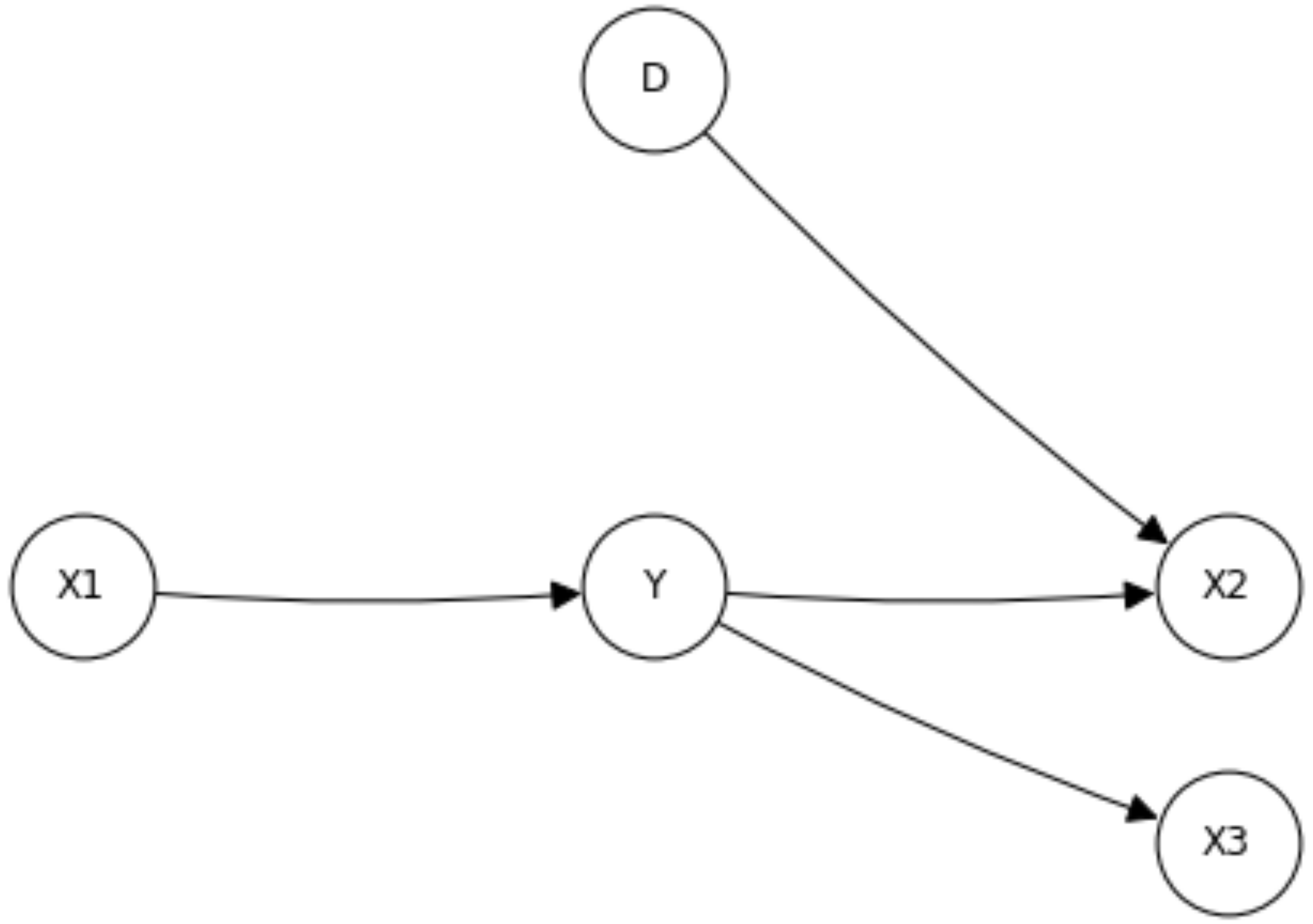
$P(Y|X_1)$ is invariant

$$P(Y|X_1, D=0) = P(Y|X_1, D=1) = P(Y|X_1, D=2) = P(Y|X_1)$$

↳ this is true if $Y \perp\!\!\!\perp D \mid X_1$

$Y \perp_d D \mid X_1$ in true graph

Separating features intuition



$P(Y \mid X_2)$ is not invariant

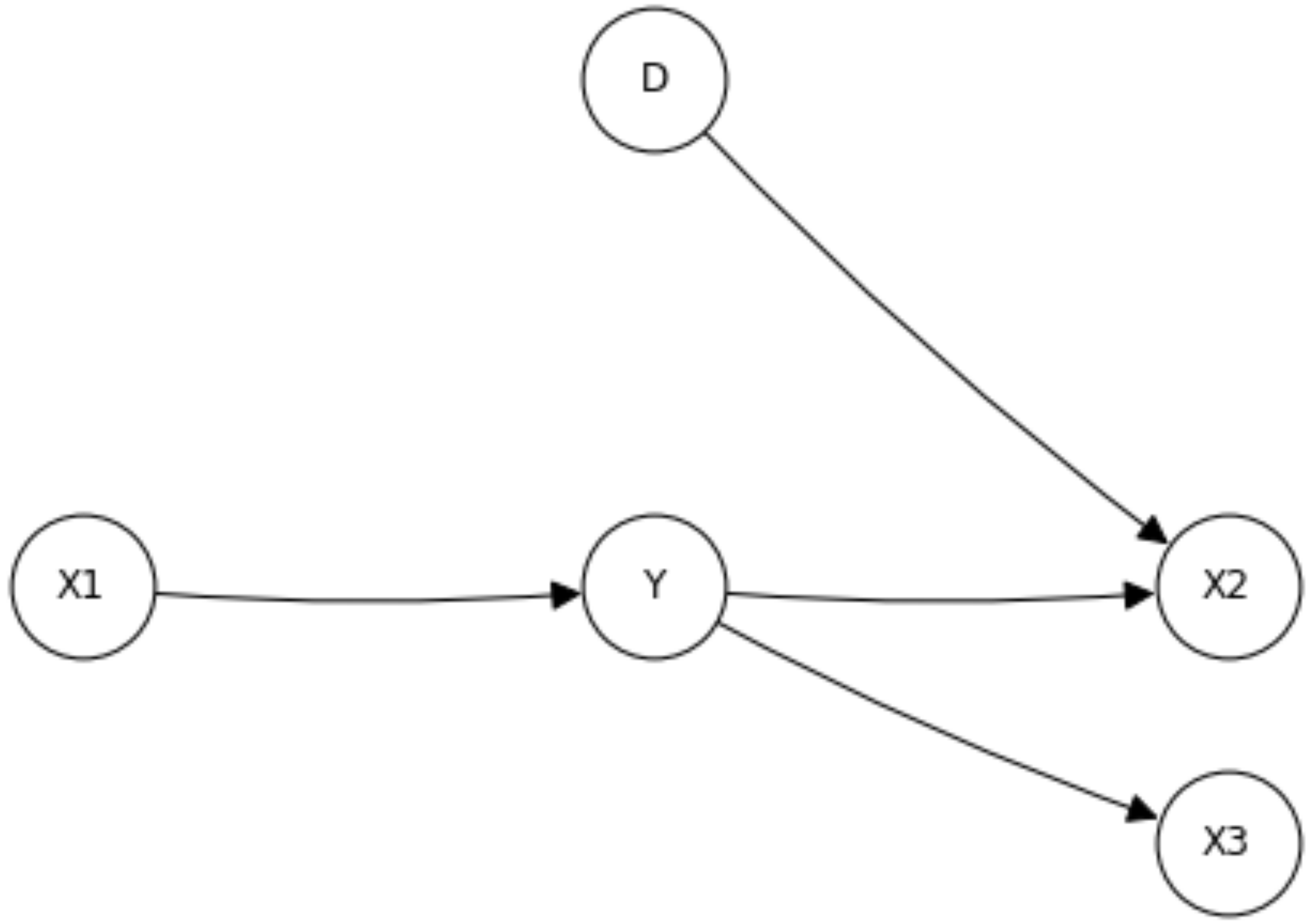
$P(Y \mid X_2, D=0) \neq P(Y \mid X_2, D=1) \neq P(Y \mid X_2, D=2)$

↳ this means $Y \not\perp D \mid X_2$

$Y \not\perp_d D \mid X_2$

$P(X_1, Y, X_2, X_3, D)$

Separating features intuition



$P(Y|X_2)$ is ~~not~~ invariant

$P(Y|X_2, D=0) \neq P(Y|X_2, D=1) \neq P(Y|X_2, D=2)$

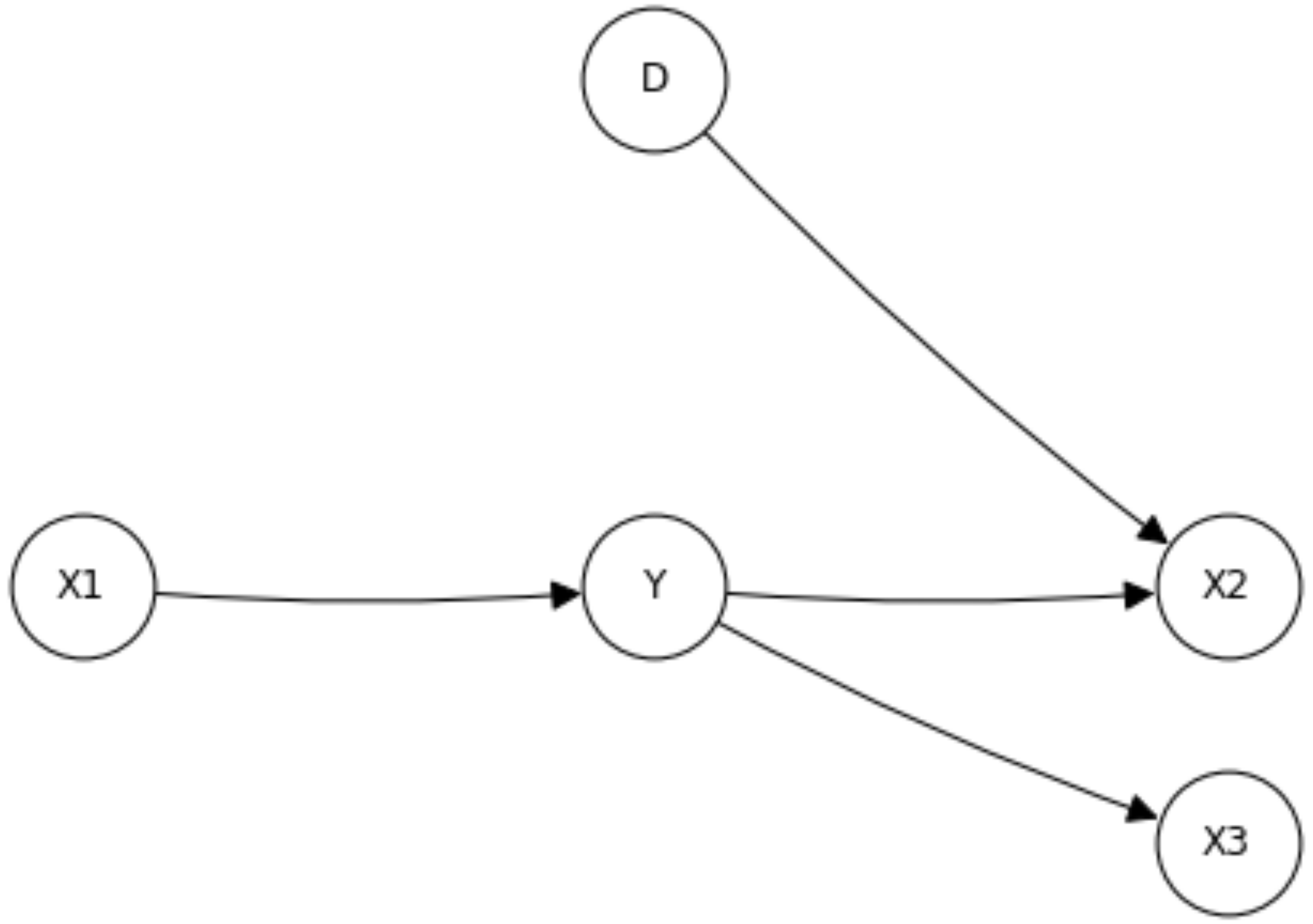
↳ this means $Y \not\perp D \mid X_2$

$Y \not\perp_d D \mid X_2$

$P(X_1, Y, X_2, X_3, D)$

Look for features $S \subseteq X$ $\quad Y \perp_d D \mid S$

Separating features intuition

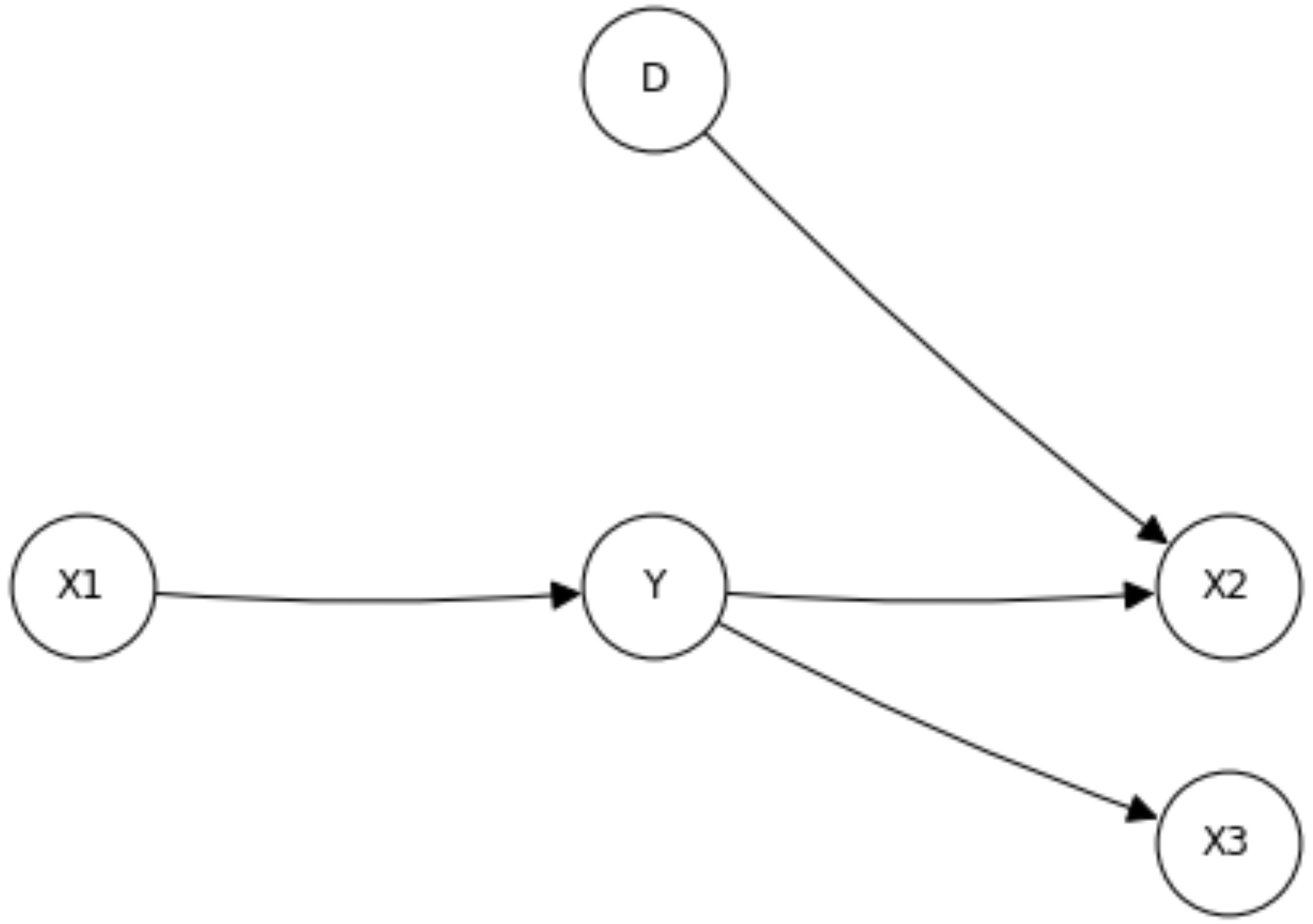


What about X_3?

$Y \perp_d D \mid X_3$?

$P(X_1, Y, X_2, X_3, D)$

Separating features intuition

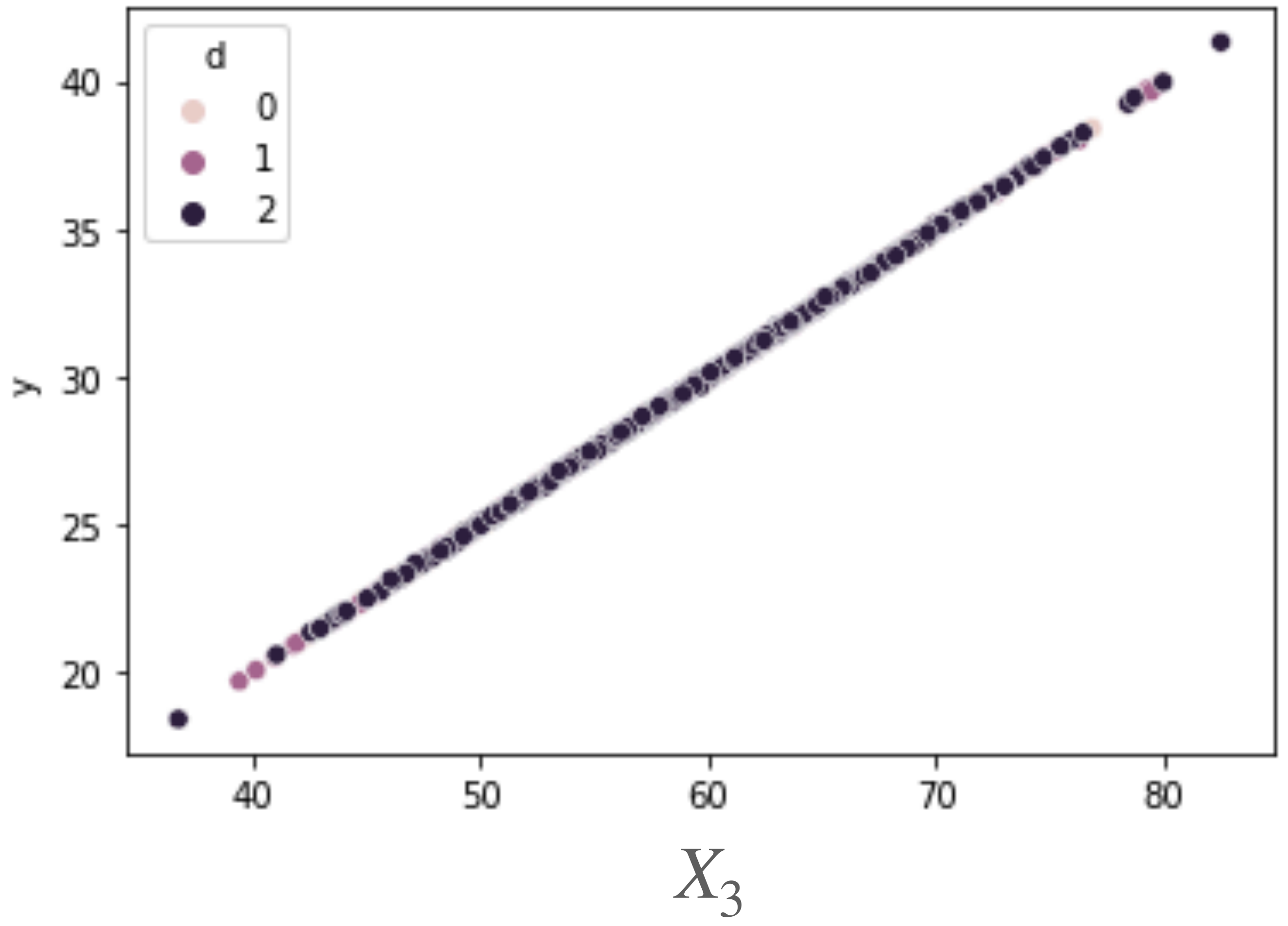


$P(X_1, Y, X_2, X_3, D)$

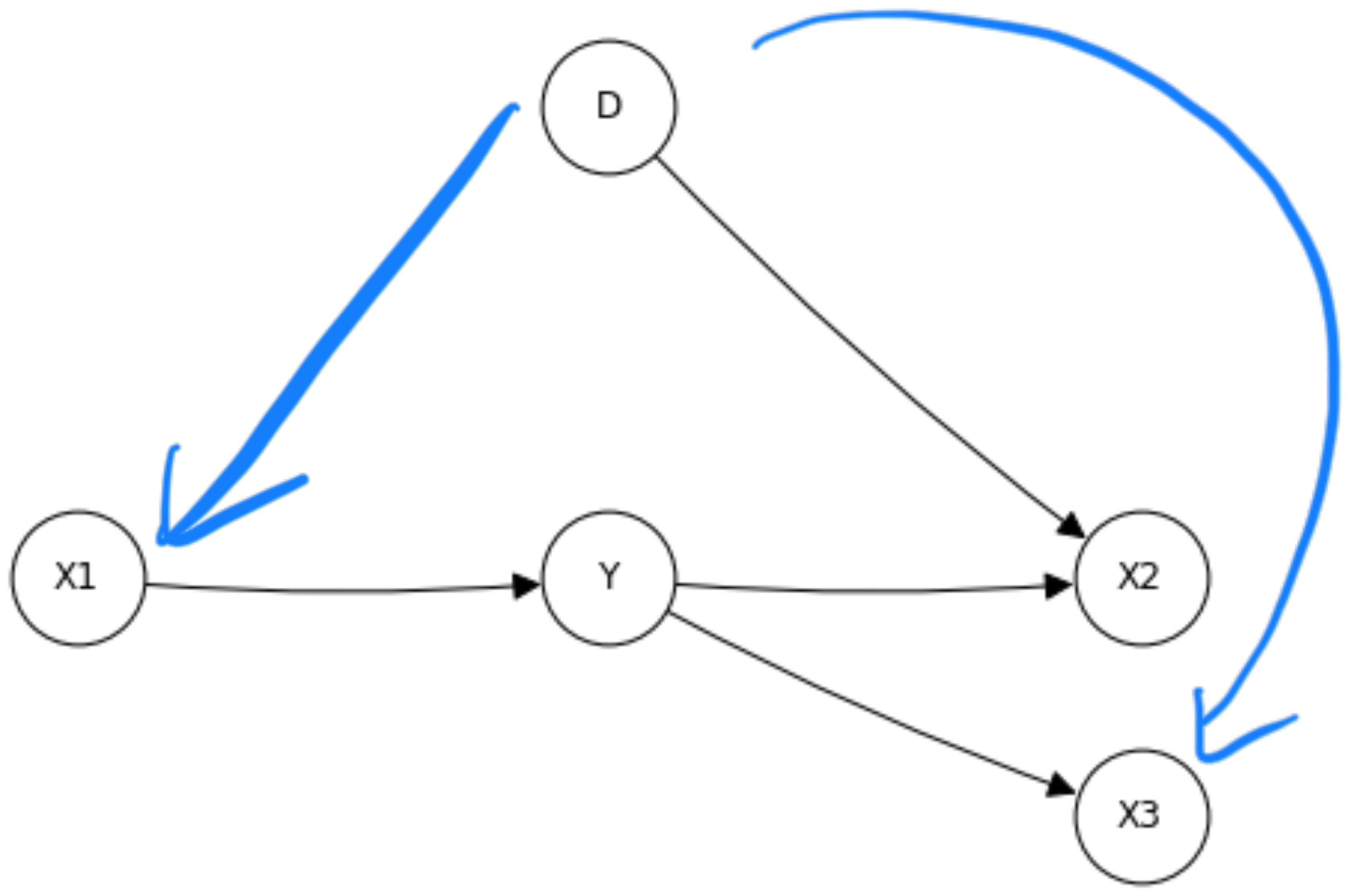
```
sns.scatterplot(data = df, x="x3", y="y", hue="d")

X3_0 = df_0["x3"].values.reshape(-1, 1)
X3_2 = df_2["x3"].values.reshape(-1, 1)
model = LinearRegression().fit(X3_0, Y_0)
est_Y_2 = model.predict(X3_2)
print("Mean squared error predicting Y in environment 2 based on model learnt in environment 0 from X3"
```

```
Mean squared error predicting Y in environment 2 based on model learnt in environment 0 from X3 0.00260
```



Which variables d-separate Y from D now?



$P(X_1, Y, X_2, X_3, D)$

Intervention on every variable except Y = domain generalisation

A description of domain adaptation tasks:

- **Domain generalisation:** required to work under **any intervention**

C1	C2	X1	X2	X3	Y	X4
1	0	?	?	?	?	?
1	0	?	?	?	?	?
1	0	?	?	?	?	?
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

No data in target

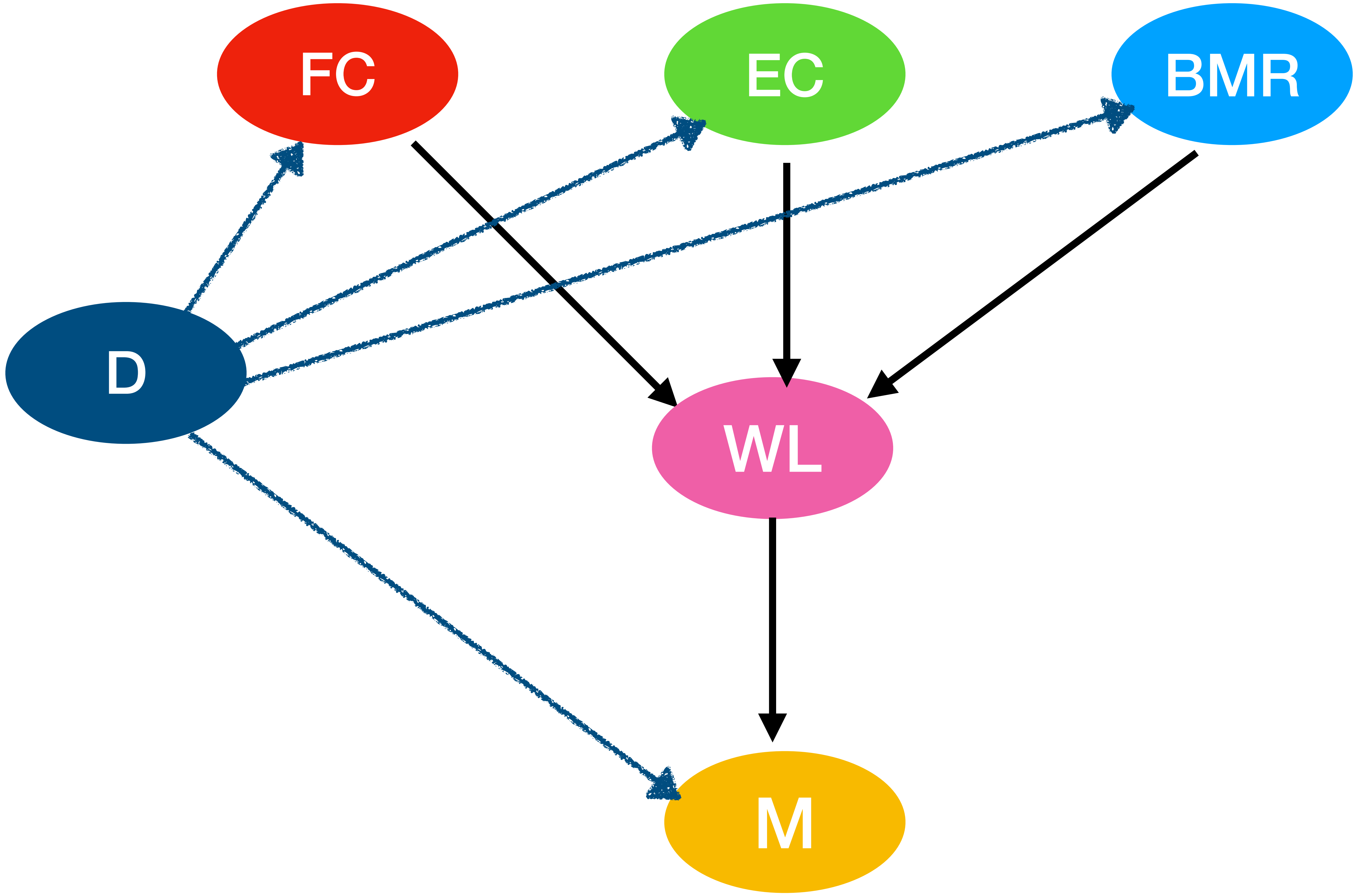
Target domain

Source domains

- Estimate $\hat{f}$ in Y = $\hat{f}$(X1, X2, X3, X4) from source domains, no idea about what happens in the target

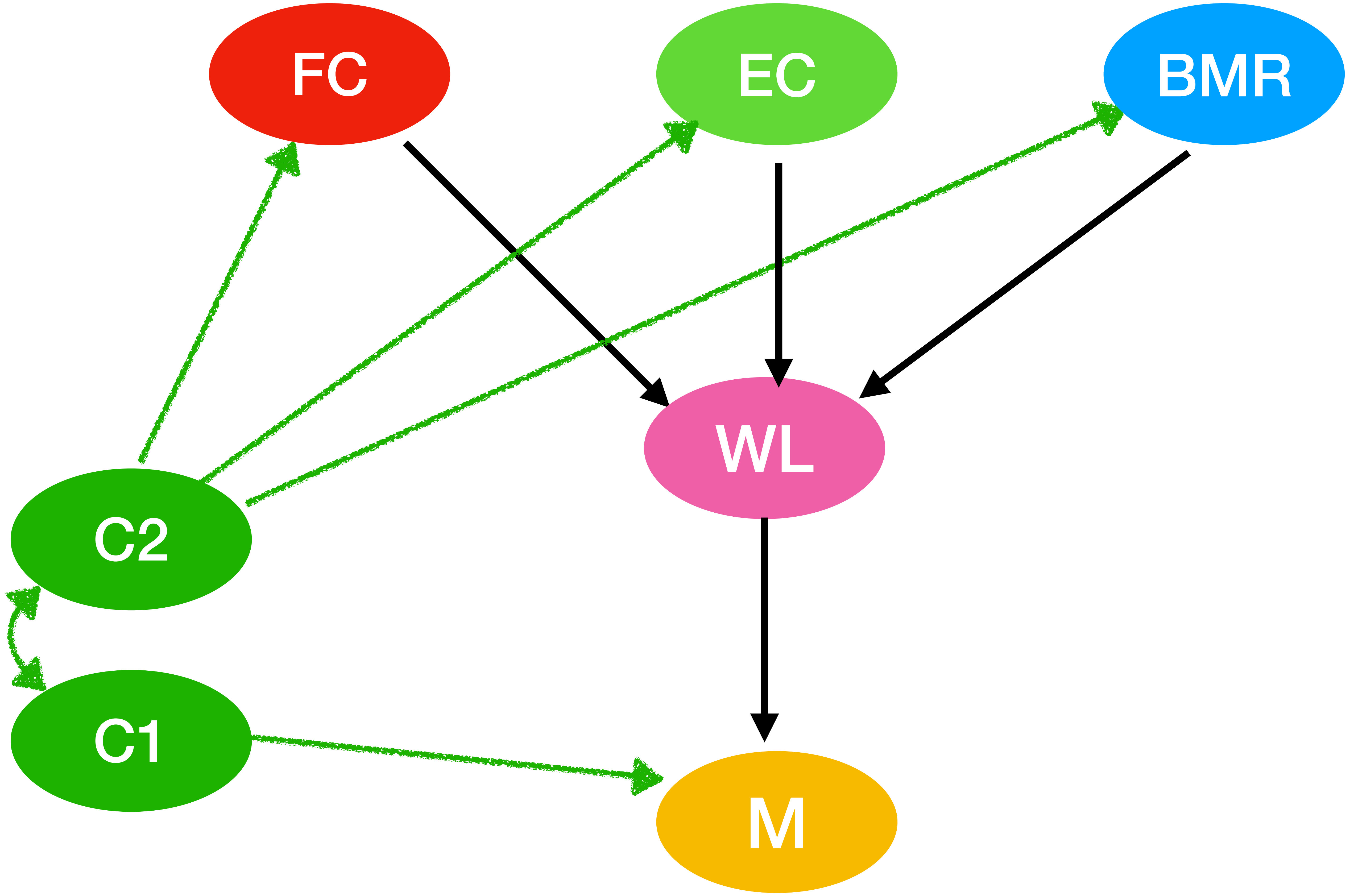
Joint Causal Inference [Mooij et al. 2020]

D	Food Calories	Exercise Calories	BMR	Weight loss	Motivation
0	1200	1000	1500	-0,17	9
0	1201	800	1500	-0,14	8
0	1195	200	1499	-0,07	7
0	....	....	....	....	....
1	2000	600	3000	-0,21	7
1	2190	450	3000	-0,16	8
1	2000	200	2999	-0,16	8
1	....	....	....	....	....
2	1200	1000	1500	-0,17	9
2	1201	800	1500	-0,14	10
2	1195	200	1499	-0,07	10
2	1340	900	1498	-0,14	....



Joint Causal Inference [Mooij et al. 2020]

C1	C2	Food Calories	Exercise Calories	BMR	Weight loss	Motivation
1	0	1200	1000	1500	-0,17	9
1	0	1201	800	1500	-0,14	8
1	0	1195	200	1499	-0,07	7
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....



Now we can learn the graph with standard causal algorithms for observational data - we can add additional knowledge (e.g. context variables don't cause the system variables)

A description of domain adaptation tasks:

- Supervised multi-source domain adaptation

C1	C2	X1	X2	X3	Y	X4
1	0	1200	1000	1500	-0.1	9
1	0	1201	800	1500	?	8
1	0	1195	200	1499	?	7
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

We can try to test for

$Y \perp\!\!\!\perp C_1 \mid S$

- Estimate $\hat{f}$ in Y = $\hat{f}$(X1, X2, X3, X4) from source domains and few labels in target domain

Unsupervised domain adaptation

No labels in target

C1	C2	X1	X2	X3	Y	X4
1	0	1200	1000	1500	?	9
1	0	1201	800	1500	?	8
1	0	1195	200	1499	?	7
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

Target domain

Source domains

- **Problem:** Y is always missing in target, so we cannot test $Y \perp\!\!\!\perp C_1 \mid X_1$ etc.

Unsupervised domain adaptation

No labels in target

C1	C2	X1	X2	X3	Y	X4
1	0	1200	1000	1500	?	9
1	0	1201	800	1500	?	8
1	0	1195	200	1499	?	7
1	0	....	....	....	....	....
0	1	2000	600	3000	-0,21	7
0	1	2190	450	3000	-0,16	8
0	1	2000	200	2999	-0,16	8
0	1	....	....	....	....	....
0	0	1200	1000	1500	-0,17	9
0	0	1201	800	1500	-0,14	10
0	0	1195	200	1499	-0,07	10
0	0	1340	900	1498	-0,14	....

Target domain

Source domains

$X_1 \not\perp\!\!\!\perp X_2$

$X_1 \not\perp\!\!\!\perp C_1$

$X_1 \not\perp\!\!\!\perp X_2 \mid C_1$

$X_1 \perp\!\!\!\perp X_2 \mid Y, C_1 = 0$

- **Idea:** Can we use all other in/dependences?

Inferring separating sets of features

- We can learn an equivalence class of the unknown **single causal graph** using **conditional independence tests** with **Joint Causal Inference**
- We assume **no extra dependences involving Y** in target domain C1=1

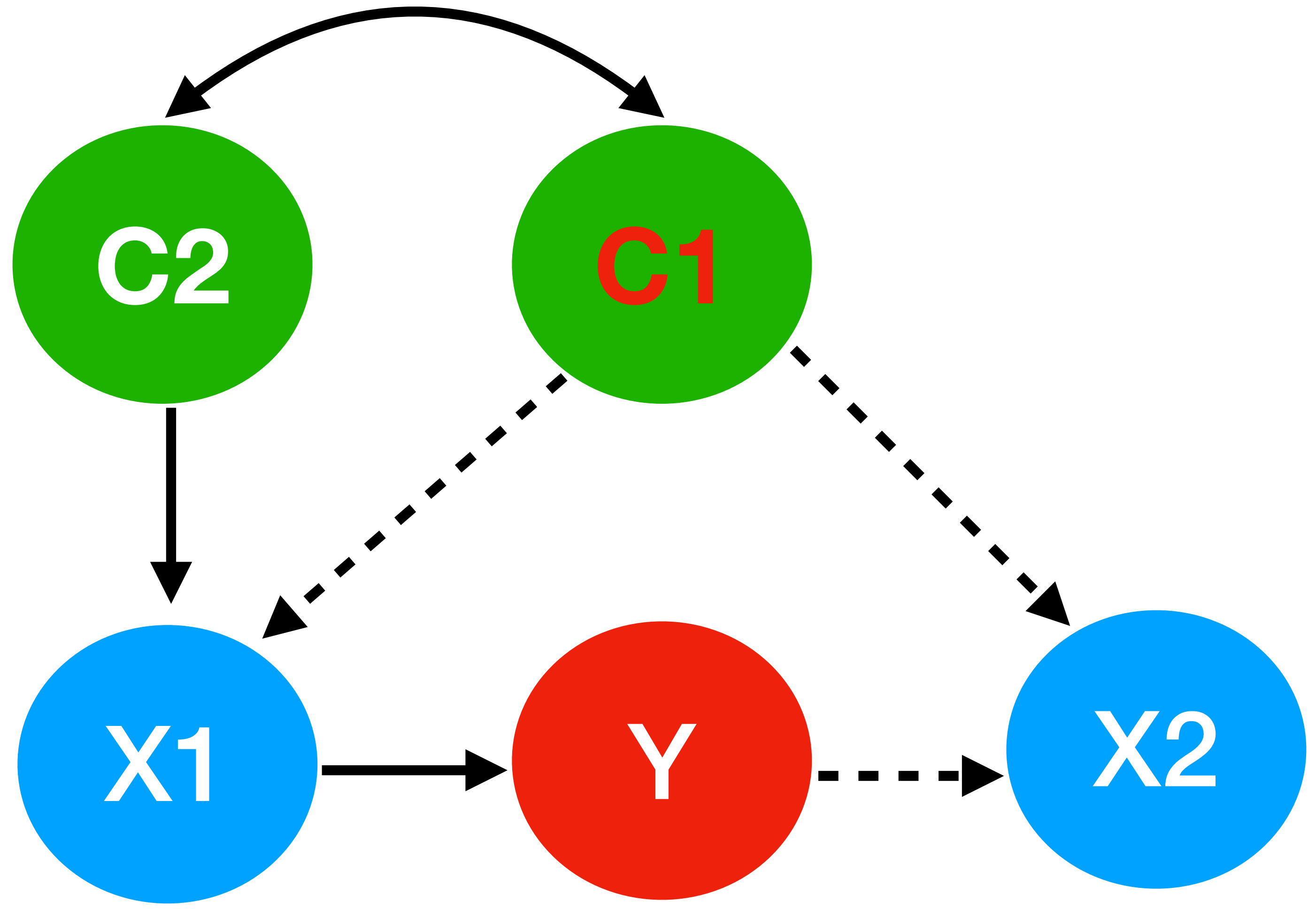
C1	C2	X1	X2	Y
0	0	0,1	1	0
0	0	0,2	1	0
0	0	1,1	2	1
0	1	3,1	2	1
0	1	3,2	3	1
0	1	4	3	1
1	0	0,2	0	?
1	0	0,3	0	?
1	0	0,3	1	?

$$Y \not\perp\!\!\!\perp C_2 \mid C_1 = 0$$

$$Y \perp\!\!\!\perp C_2 \mid X_1, C_1 = 0$$

$$X_2 \perp\!\!\!\perp C_2 \mid Y, C_1 = 0$$

Perform allowed CI tests



All possible compatible graphs

$$Y \perp\!\!\!\perp C_1 \mid X_1 ?$$

Inferring separating sets of features [Magliacane et al 2018]

Query $Y \perp\!\!\!\perp C_1 \mid X_1$?

Assumptions

All testable conditional independences from data

$$X_1 \perp\!\!\!\perp X_3 \mid X_4$$
$$Y \perp\!\!\!\perp C_2 \mid X_1, C_1 = 0$$
$$X_2 \perp\!\!\!\perp C_2 \mid Y, C_1 = 0$$
$$\ldots$$

Logic encoding of d-separation [Hyttinen et al. 2014]

Theorem prover

Provably separating $Y \perp\!\!\!\perp C_1 \mid X_1$

Provably not separating $Y \not\perp\!\!\!\perp C_1 \mid X_1$

? Not identifiable

Application to feature selection

Source domains data

C1	C2	X1	X2	Y
0	0	0,1	1	0
0	0	0,2	1	0
0	0	1,1	2	1
0	1	3,1	2	1
0	1	3,2	3	1
0	1	4	3	1

Standard feature selection

List of combinations of features ordered by source domain loss in predicting Y

L=({X1, C2}, {X1, X2, C2} , {X1, X2}, …)

Select new set S

S= {X1, C2}

All data (including target)

C1	C2	X1	X2	Y
0	0	0,1	1	0
0	0	0,2	1	0
0	0	1,1	2	1
0	1	3,1	2	1
0	1	3,2	3	1
0	1	4	3	1
1	0	0,2	0	?
1	0	0,3	0	?
1	0	0,3	1	?

Query $Y \perp\!\!\!\perp C_1 \mid S?$

Assumptions

All testable conditional independences from data

$X_1 \perp\!\!\!\perp X_3 \mid X_4$

$Y \perp\!\!\!\perp C_2 \mid X_1, C_1 = 0$

$X_2 \perp\!\!\!\perp C_2 \mid Y, C_1 = 0$

…

Logic encoding of d-separation [Hyttinen et al. 2014]

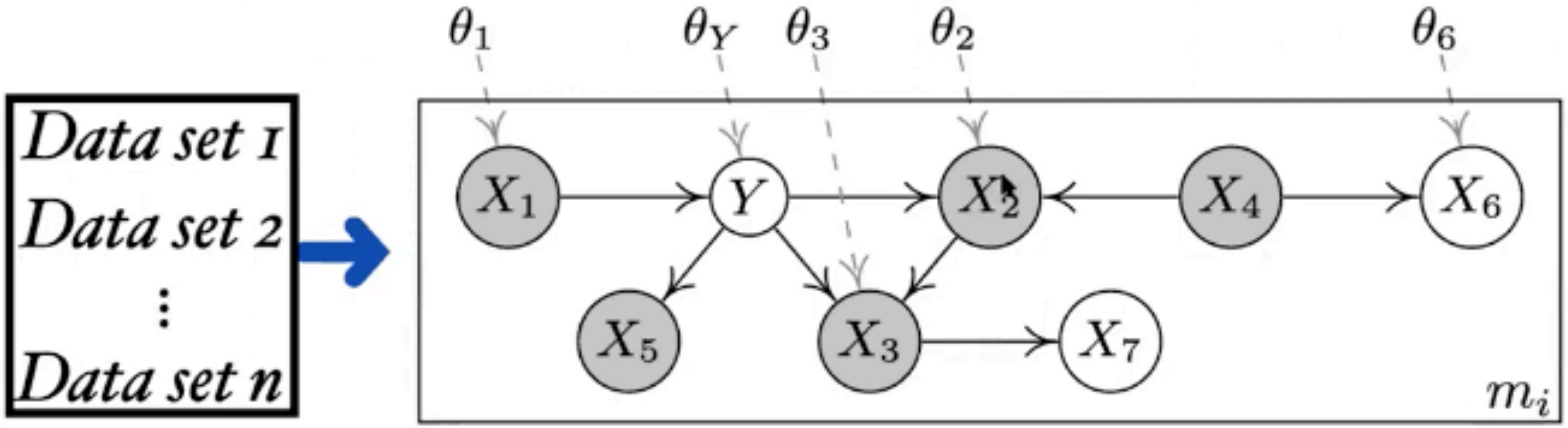
Theorem prover

✗ Provably not separating $Y \not\perp\!\!\!\perp C_1 \mid S$

? Not identifiable

ITERATE UNTIL PROVABLY SEPARATING

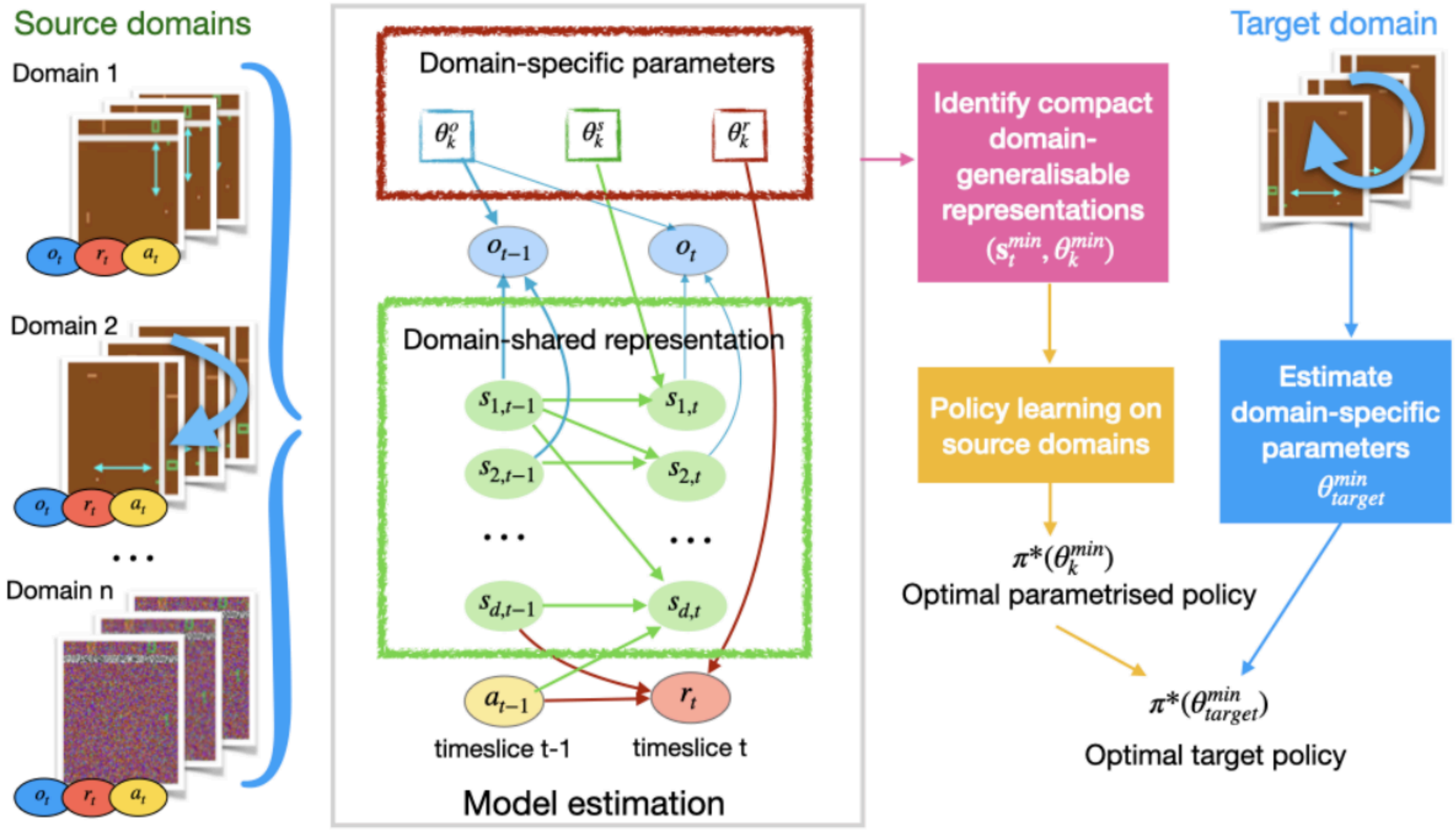
An alternative to JCI: CD-NOD



Simplifying assumption, no new edges in target domain

https://arxiv.org/abs/1903.01672

Application of CD-NOD to fast adaptation (AdaRL)

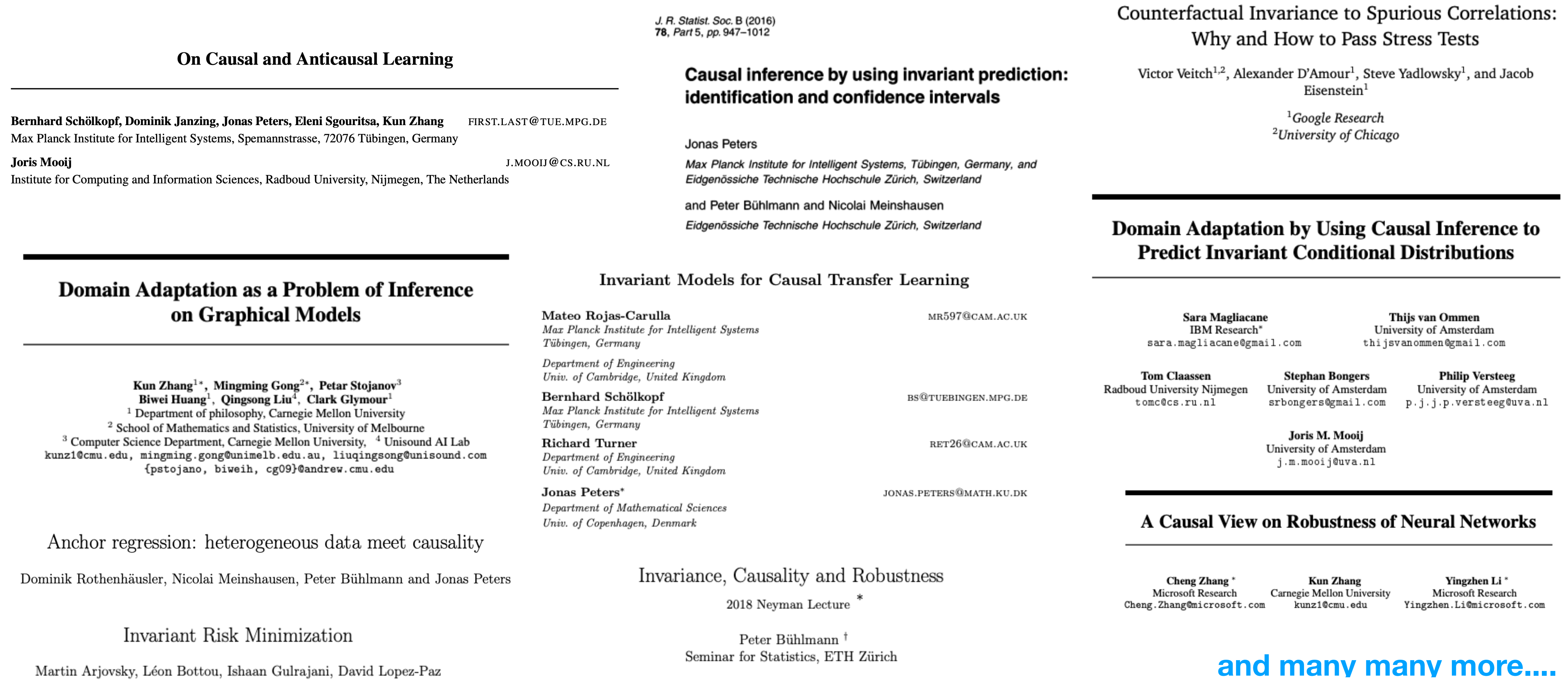


https://arxiv.org/abs/1903.01672

Causality-inspired ML and distribution shifts

- Causal graphs and d-separation [Pearl 2009] are a principled way to reason about **invariances and distribution shift**

- This is true even with:
    - **Unknown causal graph**
    - **Missing data/CI** (so unknown MEC)
        - **D-separation logic encodings** [Hyttinen et al 2014]


On Causal and Anticausal Learning

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang FIRST.LAST@TUE.MPG.DE
Max Planck Institute for Intelligent Systems, Spemannstrasse, 72076 Tübingen, Germany
Joris Mooij J.MOOIJ@CS.RU.NL
Institute for Computing and Information Sciences, Radboud University, Nijmegen, The Netherlands

J. R. Statist. Soc. B (2016)
78, Part 5, pp. 947–1012

Causal inference by using invariant prediction: identification and confidence intervals

Jonas Peters
Max Planck Institute for Intelligent Systems, Tübingen, Germany, and Eidgenössische Technische Hochschule Zürich, Switzerland
and Peter Bühlmann and Nicolai Meinshausen
Eidgenössische Technische Hochschule Zürich, Switzerland

Counterfactual Invariance to Spurious Correlations: Why and How to Pass Stress Tests

Victor Veitch[1,2], Alexander D'Amour[1], Steve Yadlowsky[1], and Jacob Eisenstein[1]
[1]Google Research
[2]University of Chicago

Domain Adaptation as a Problem of Inference on Graphical Models

Kun Zhang[1*], Mingming Gong[2*], Petar Stojanov[3], Biwei Huang[1], Qingsong Liu[4], Clark Glymour[1]

Invariant Models for Causal Transfer Learning

Mateo Rojas-Carulla, Bernhard Schölkopf, Richard Turner, Jonas Peters

Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions

Sara Magliacane, Thijs van Ommen, Tom Claassen, Stephan Bongers, Philip Versteeg, Joris M. Mooij

Anchor regression: heterogeneous data meet causality

Dominik Rothenhäusler, Nicolai Meinshausen, Peter Bühlmann and Jonas Peters

Invariance, Causality and Robustness

2018 Neyman Lecture

Peter Bühlmann
Seminar for Statistics, ETH Zürich

A Causal View on Robustness of Neural Networks

Cheng Zhang, Kun Zhang, Yingzhen Li

Invariant Risk Minimization

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, David Lopez-Paz

and many many more....