Workshop on Artificial Intelligence, Causality and Personalized Medicine

→8. and 9. September Hybrid Event

Leibniz AILab

Causality-inspired ML: what can causality do for ML?

Sara Magliacane
University of Amsterdam
MIT-IBM Watson AI Lab

What can ideas from causality do for ML?

- **Real-world ML** needs to deal with:
  - **Biased data** (fairness, selection bias, generalization)
  - **Heterogeneous** data, small samples, missing/corrupted data, **not iid**
  - **Actionable insights** (decisions cannot be made on correlations)

What can ideas from causality do for ML?

- **Real-world ML** needs to deal with:
  - **Biased data** (fairness, selection bias, generalization)
  - **Heterogeneous** data, small samples, missing/corrupted data, **not iid**
  - **Actionable insights** (decisions cannot be made on correlations)
- **Causal inference** can help with some of these questions:
  - Systematic **data fusion** and **reuse** with biased data, heterogenous, not iid data
  - A systematic way to **extract actionable insights**

What can ideas from causality do for ML?

- **Real-world ML** needs to deal with:
  - **Biased data** (fairness, selection bias, generalization)
  - **Heterogeneous** data, small samples, missing/corrupted data, **not iid**
  - **Actionable insights** (decisions cannot be made on correlations)
- **Causal inference** can help with some of these questions:
  - Systematic **data fusion** and **reuse** with biased data
  - A systematic way to **extract actionable insights**
- **"Full" causality** can be **not necessary** or **too expensive** ->

E.g. too many experiments to fully identify the graph

What can ideas from causality do for ML?

- **Real-world ML** needs to deal with:
  - **Biased data** (fairness, selection bias, generalization)
  - **Heterogeneous** data, small samples, missing/corrupted data, **not iid**
  - **Actionable insights** (decisions cannot be made on correlations)
- **Causal inference** can help with some of these questions:
  - Systematic **data fusion** and **reuse** with biased data, heterogenous, not iid data
  - A systematic way to **extract actionable insights**
- **“Full” causality** can be **not necessary** or **too expensive** -> *Causality-Inspired*

What can ideas from causality do for ML?

- **Real-world ML** needs to deal with:
  - **Biased data** (fairness, selection bias, generalization)
  - **Heterogeneous** data, small samples, missing/corrupted data, **not iid**
  - **Actionable insights** (decisions cannot be made on correlations)
- **Causal inference** can help with some of these questi
  - Systematic **data fusion** and **reus** iid data
  - A systematic way to **extract actionable**

In this talk: example in domain adaptation, but lots of related work

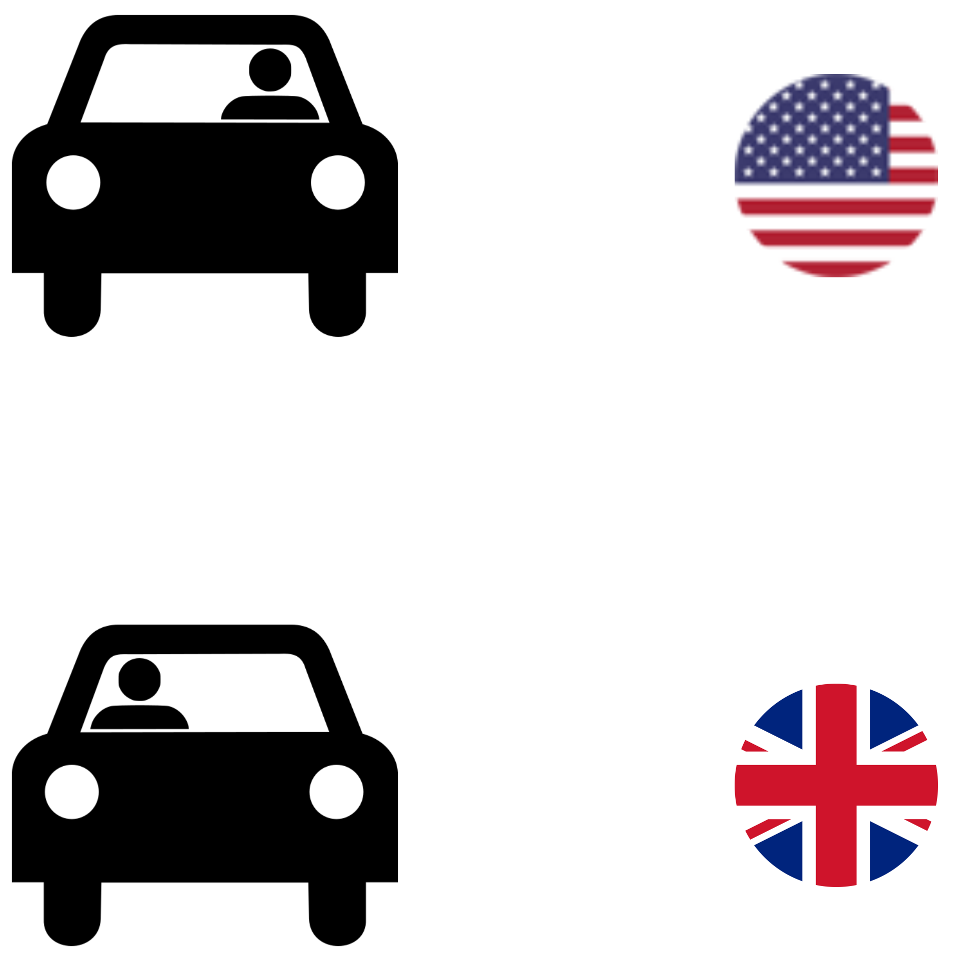
- **"Full" causality** can be **not necessary** or **too expensive** -> *Causality-Inspired*

Transfer learning and causal inference

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

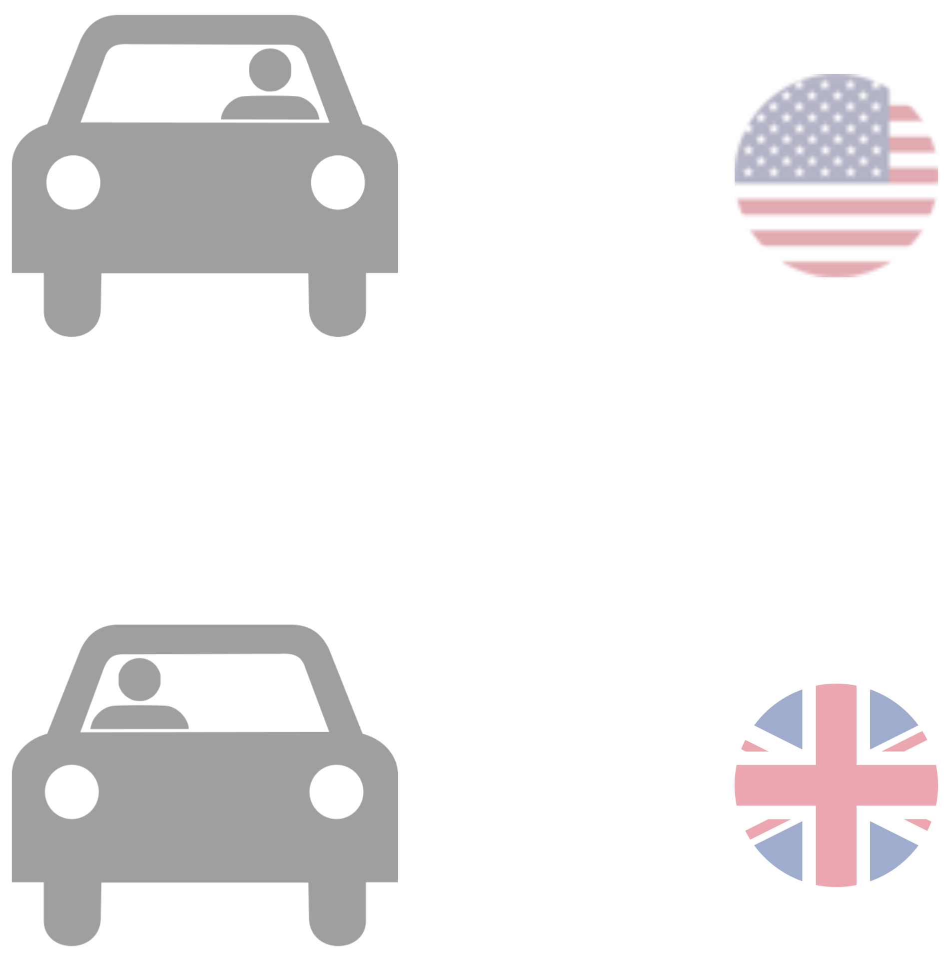
Transfer learning and causal inference

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



Transfer learning and causal inference

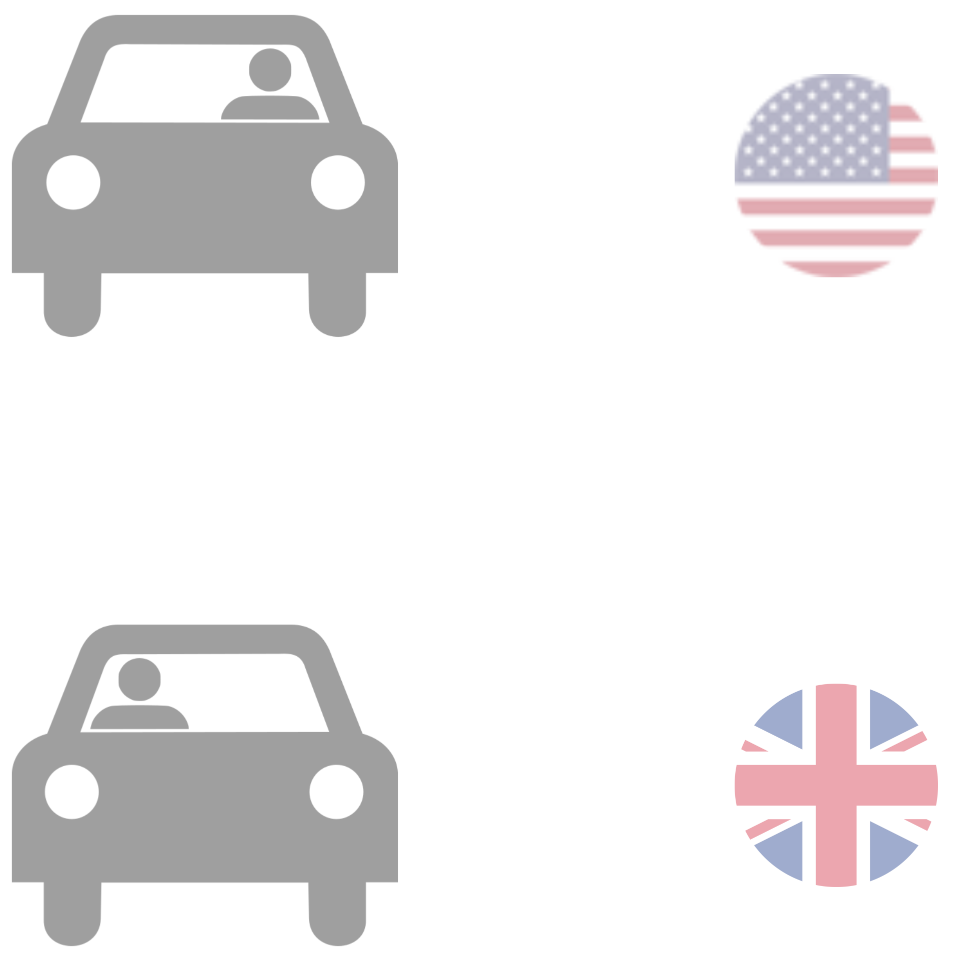
- Transfer learning:
  - How can I predict what happens when the distribution changes?



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

Transfer learning and causal inference

- Transfer learning:
  - How can I predict what ha
    when the distribution chan



Very general - can model also changes in distribution that are not from “real” interventions

pens when
ter an

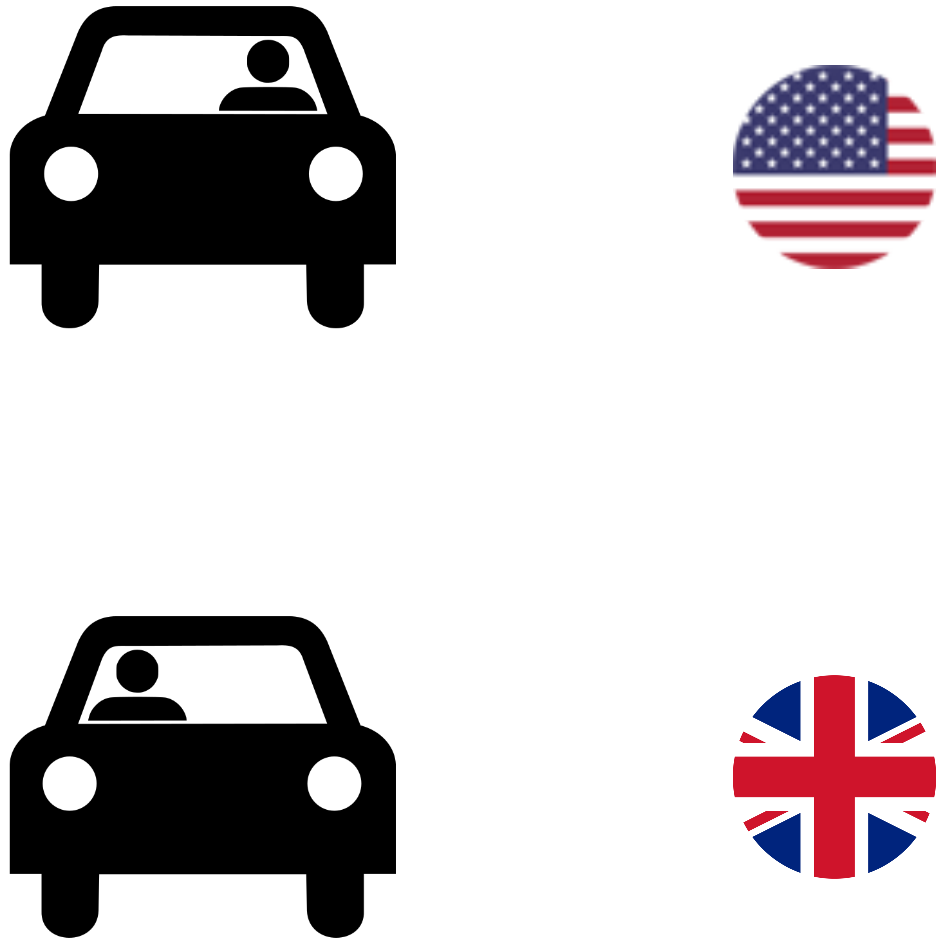
ntervention: do-calculus
[Pe l, 2009]

- X is independent of its parents

- **Soft intervention on X:**
  - Change of P(X| parents)

Transfer learning and causal inference

- Transfer learning:
  - How can I predict what ha... when the distribution chan...



...pens when ... ter an

...tervention: do-calculus [Pe...l, 2009]

- X is independent of its parents

Very general - can model also changes in distribution that are not from “real” interventions

- **Soft intervention on X:**
  - Change of P(X| parents)

Not a new idea!

On Causal and Anticausal Learning

ICML 2012

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

Abstract

We consider the problem of function estimation in the case where an underlying causal model can be inferred. This has implications for popular scenarios such as covariate shift, concept drift, transfer learning and semi-supervised learning. We argue that causal knowledge may facilitate some approaches for a given problem, and rule out others. In particular, we formulate a hypothesis for when semi-supervised learning can help, and corroborate it with empirical results.

for causal inference in the machine learning community.

An example illustrating the difference between the statistical and the causal point of view is the correlation between the frequency of storks and the human birth rate (Matthews, 2000). We may be able to train a good predictor of the birth rate which uses the frequency of storks (along with other features) as an input. However, if politicians asked us whether one could boost the birth rate by increasing the number of storks, we would have to tell them that this kind of *intervention* is not covered by the standard i.i.d. assumption of statistical learning. In practice, however, interventions can be relevant, distributions may shift over time, and we might want to combine data recorded under different

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

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]
[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

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

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

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

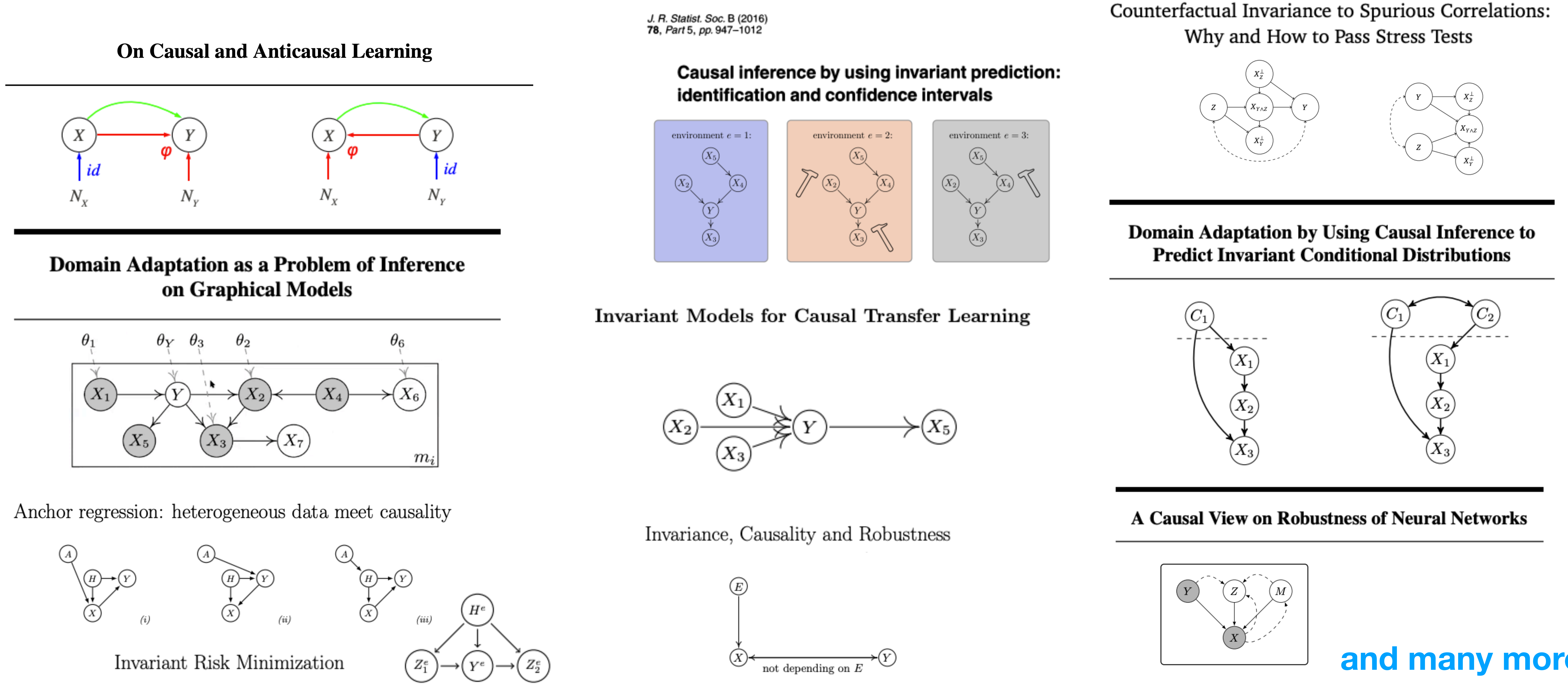
Cheng Zhang * Microsoft Research Cheng.Zhang@microsoft.com
Kun Zhang Carnegie Mellon University kunz1@cmu.edu
Yingzhen Li * Microsoft Research Yingzhen.Li@microsoft.com

Invariant Risk Minimization

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

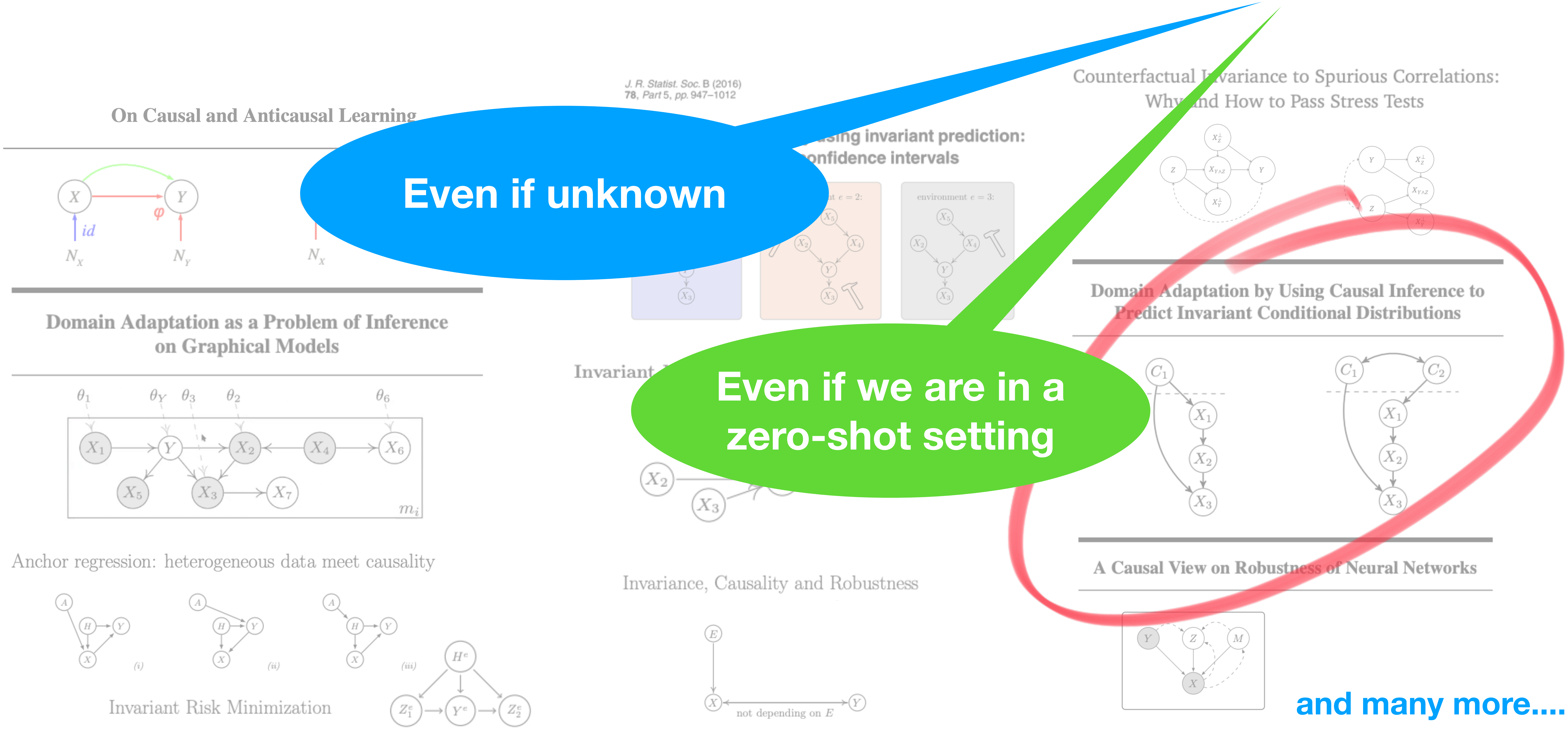
and many many more....

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



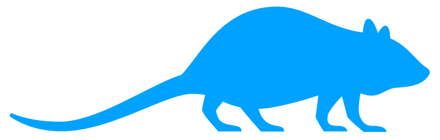
and many more....

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



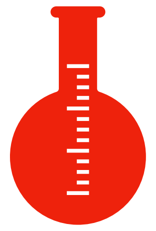
and many more....

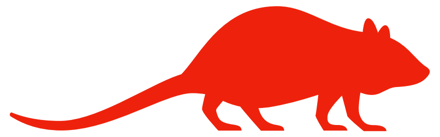
Unsupervised multi-source domain adaptation



	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0

Source domain





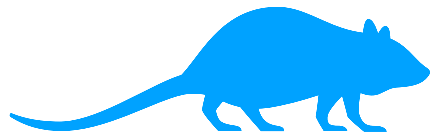
	X1	X2	Y
Gene A	3.1	2	?
Gene A	3.2	3	?
Gene A	4	2	?
Gene A	3.2	3	?

Target domain

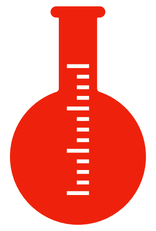
No labels in target

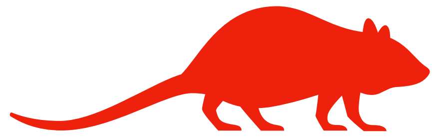
- Estimate $\hat{f}$ in $Y = \hat{f}(\mathbf{X})$ from source domains and by exploiting the knowledge of th **change** from the **unlabelled data in target**

Domain adaptation from a graphical perspective [Zhang et al. 2013]



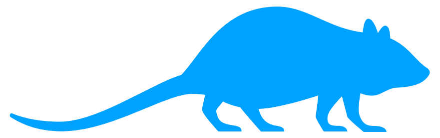
	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0



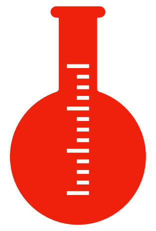


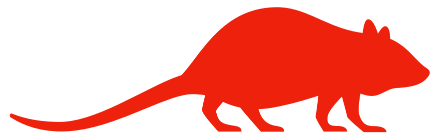
	X1	X2	Y
Gene A	3.1	2	?
Gene A	3.2	3	?
Gene A	4	2	?
Gene A	3.2	3	?

Domain adaptation from a graphical perspective [Zhang et al. 2013]



D	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0

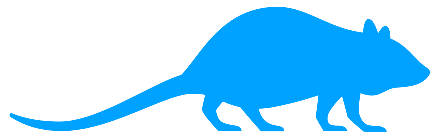


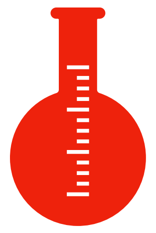


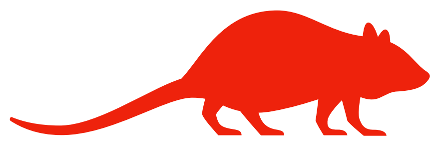
D	X1	X2	Y
Gene A	3.1	2	?
Gene A	3.2	3	?
Gene A	4	2	?
Gene A	3.2	3	?

- Add a variable D to represent the **domain**

Domain adaptation from a graphical perspective [Zhang et al. 2013]



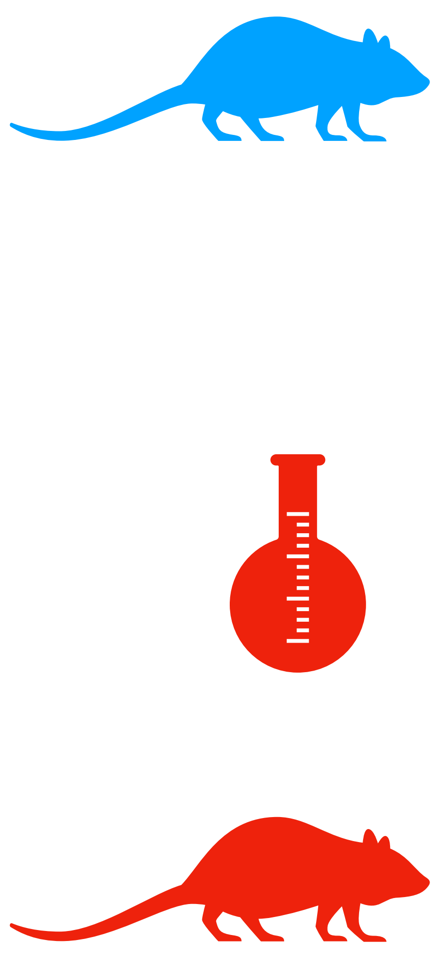




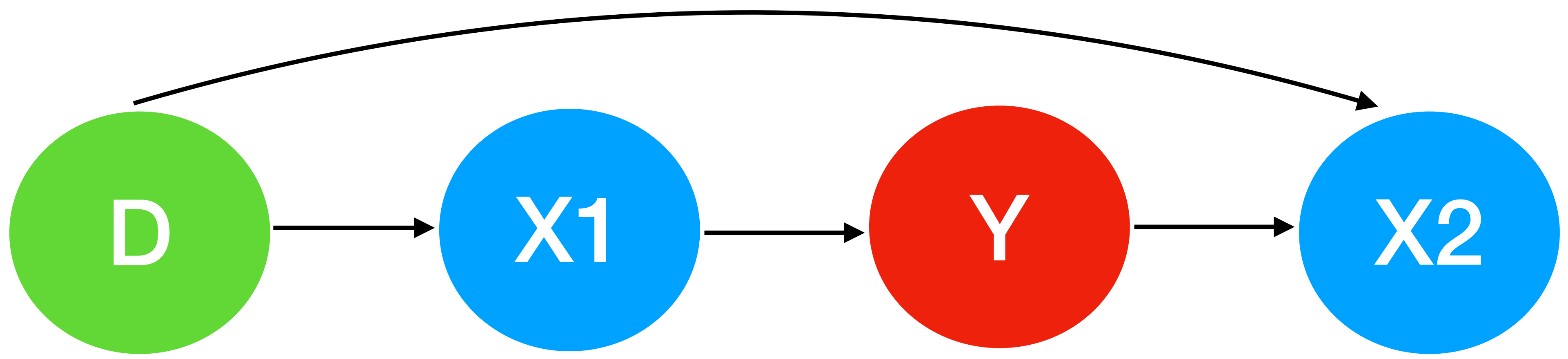
D	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0
Gene A	3.1	2	?
Gene A	3.2	3	?
Gene A	4	2	?
Gene A	3.2	3	?

- Add a variable D to represent the **domain**
- Consider the data as coming from a single distribution P(**X**,Y, D)

Domain adaptation from a graphical perspective [Zhang et al. 2013]



D	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0
Gene A	3.1	2	?
Gene A	3.2	3	?
Gene A	4	2	?
Gene A	3.2	3	?

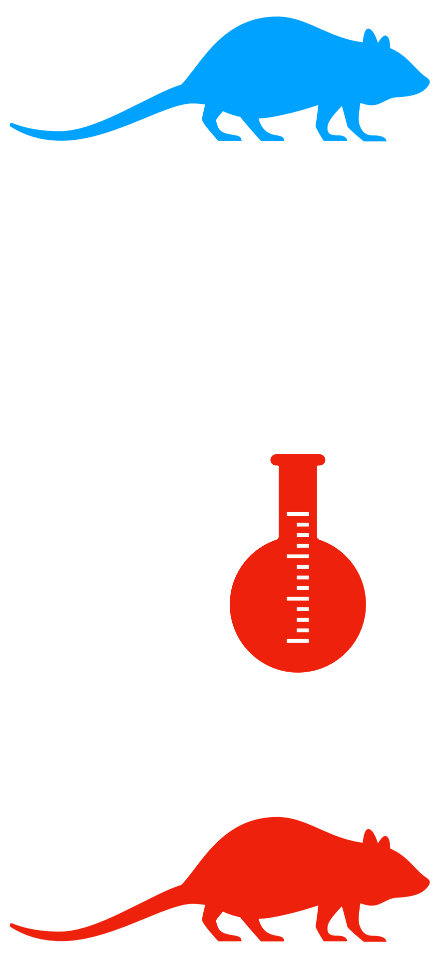


- We can represent P(**X**,Y, D) with a **(possibly unknown)** causal graph

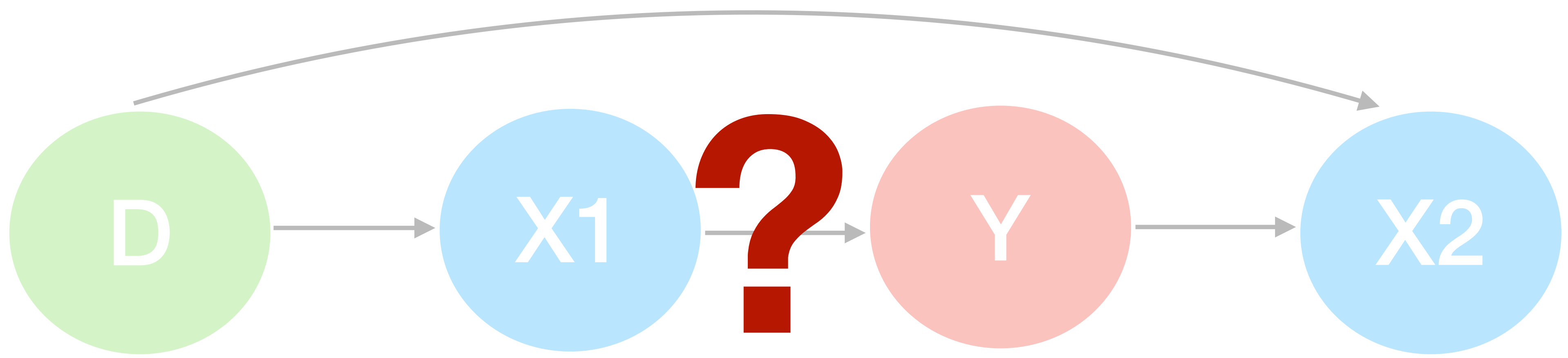
- Add a variable D to represent the **domain**

- Consider the data as coming from a single distribution P(**X**,Y, D)

Domain adaptation from a graphical perspective [Zhang et al. 2013]



D	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0
Gene A	3.1	2	?
Gene A	3.2	3	?
Gene A	4	2	?
Gene A	3.2	3	?



- We can represent P(**X**,Y, D) with a (possibly unknown) causal graph

We can still use d-separations/ conditional independences to reason about invariances

- Add a variable D to represent the **domain**
- Consider the data as coming from a single distribution P(**X**,Y, D)

Structural causal model - domain/environment variable

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$$D = 0$$

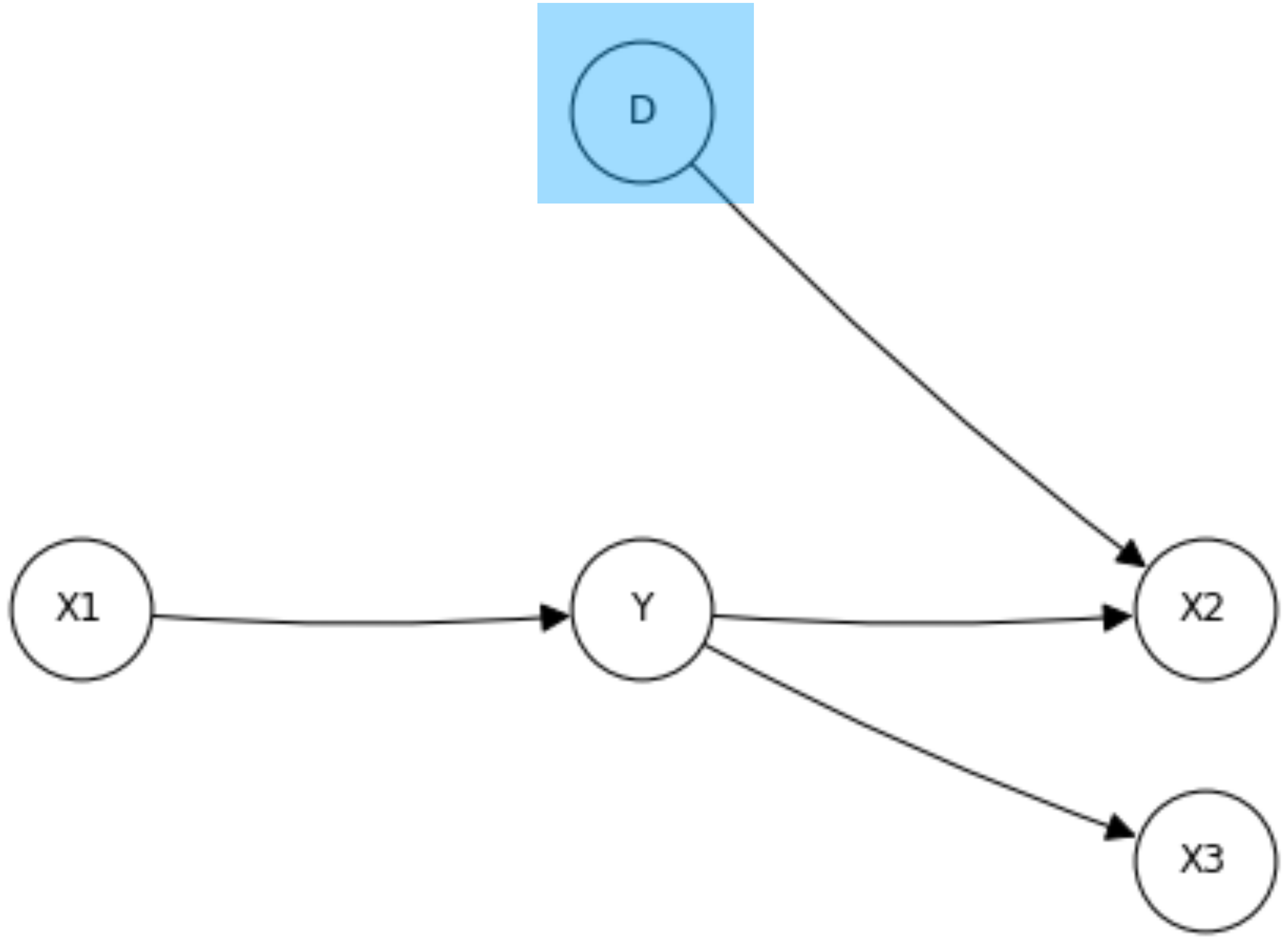
$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$$D = 1$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_Y \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

$$D = 2$$

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = \begin{cases} -2Y + \epsilon_2 & \text{if } D = 0 \\ 1 & \text{if } D = 1 \\ 10Y + \epsilon_Y & \text{if } D = 2 \end{cases} \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$



Domain adaptation example

$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = -2Y + \epsilon_2 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

obs

Source domains

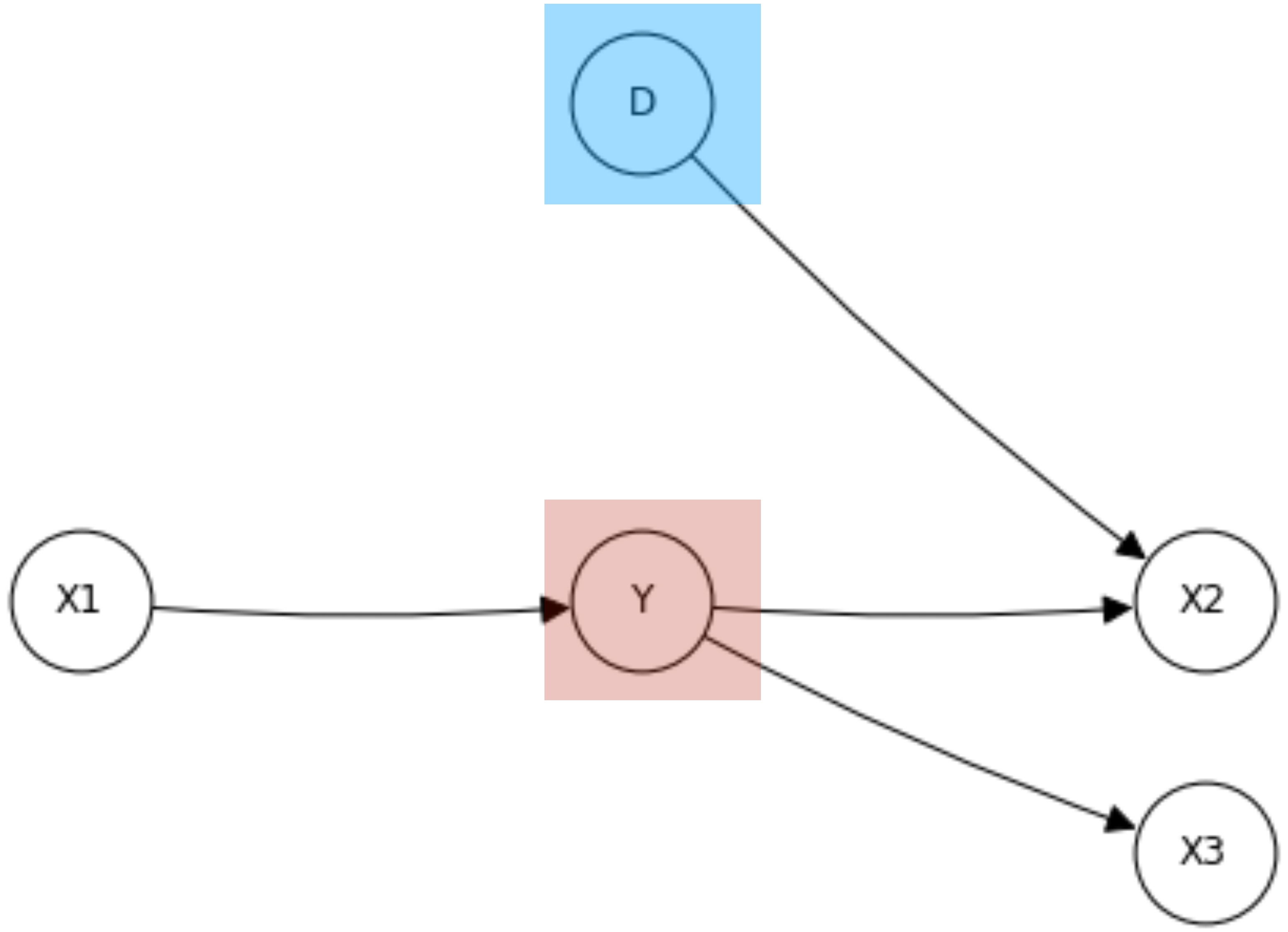
$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 1 \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

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

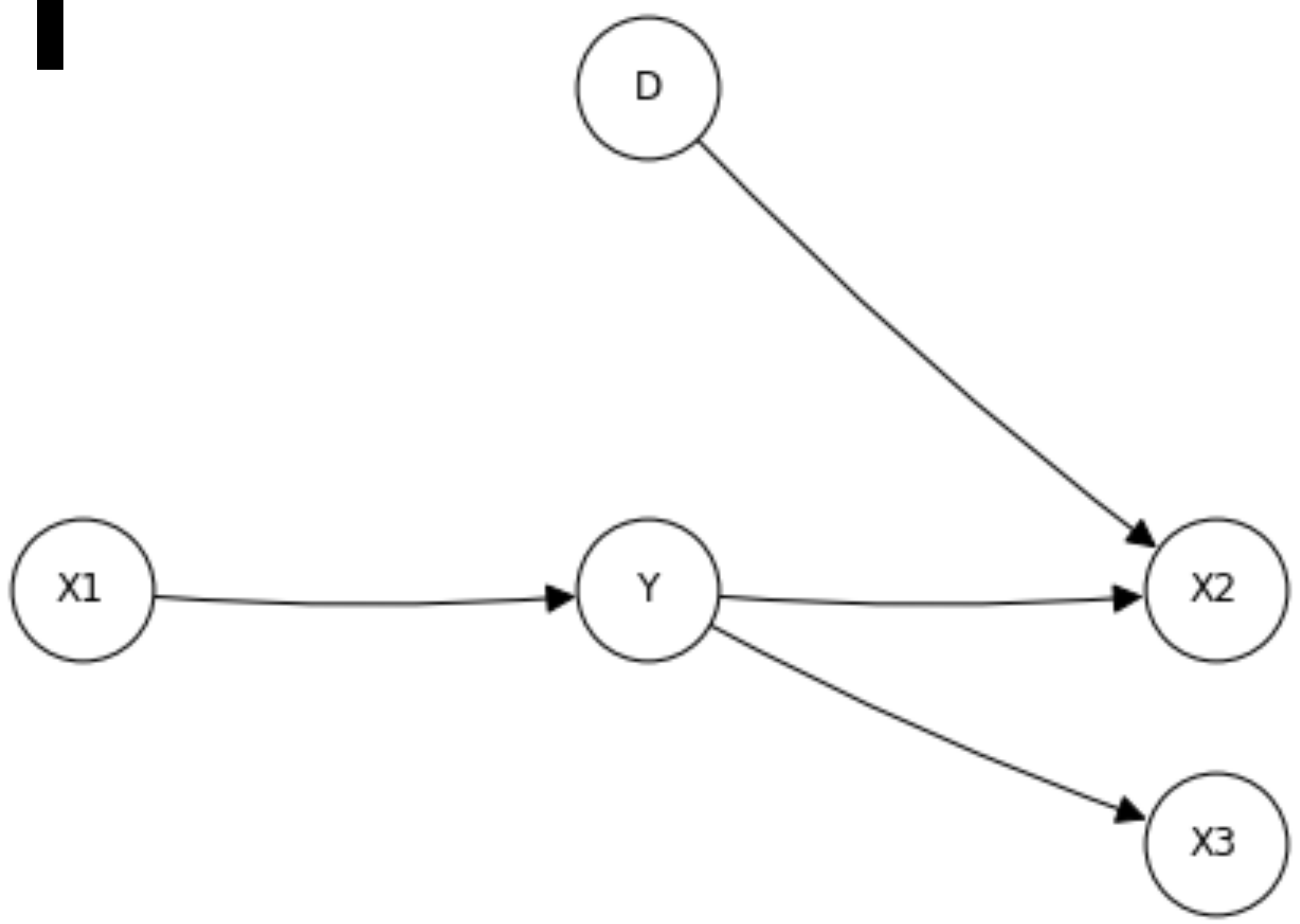
$$\begin{cases} \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_Y \sim \mathcal{N}(0,1) \\ X_1 = 10 + \epsilon_1 \\ Y = 3X_1 + \epsilon_Y \\ X_2 = 10Y + \epsilon_Y \\ X_3 = 2Y + 0.1\epsilon_3 \end{cases}$$

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

Target domain



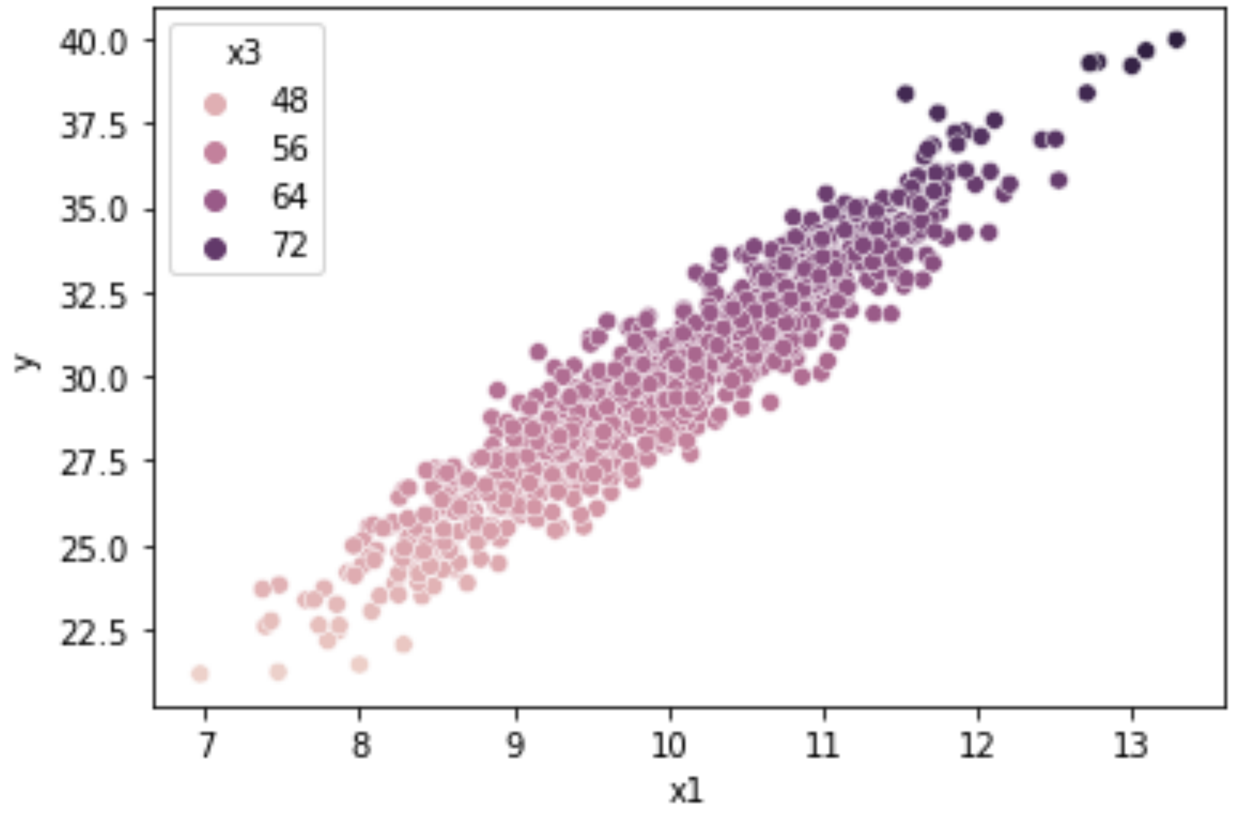
Domain adaptation example - X1



$p(Y|X_1)$ is invariant

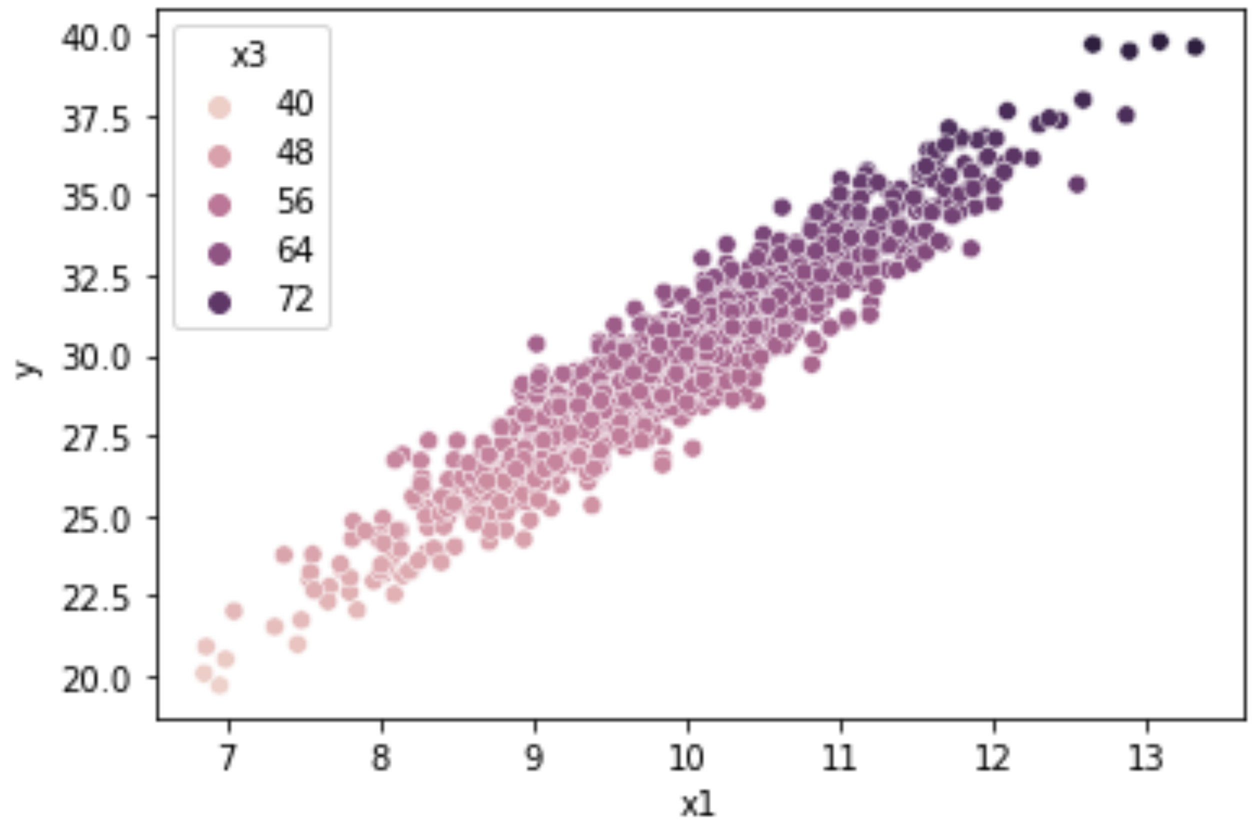
$D = 0$

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



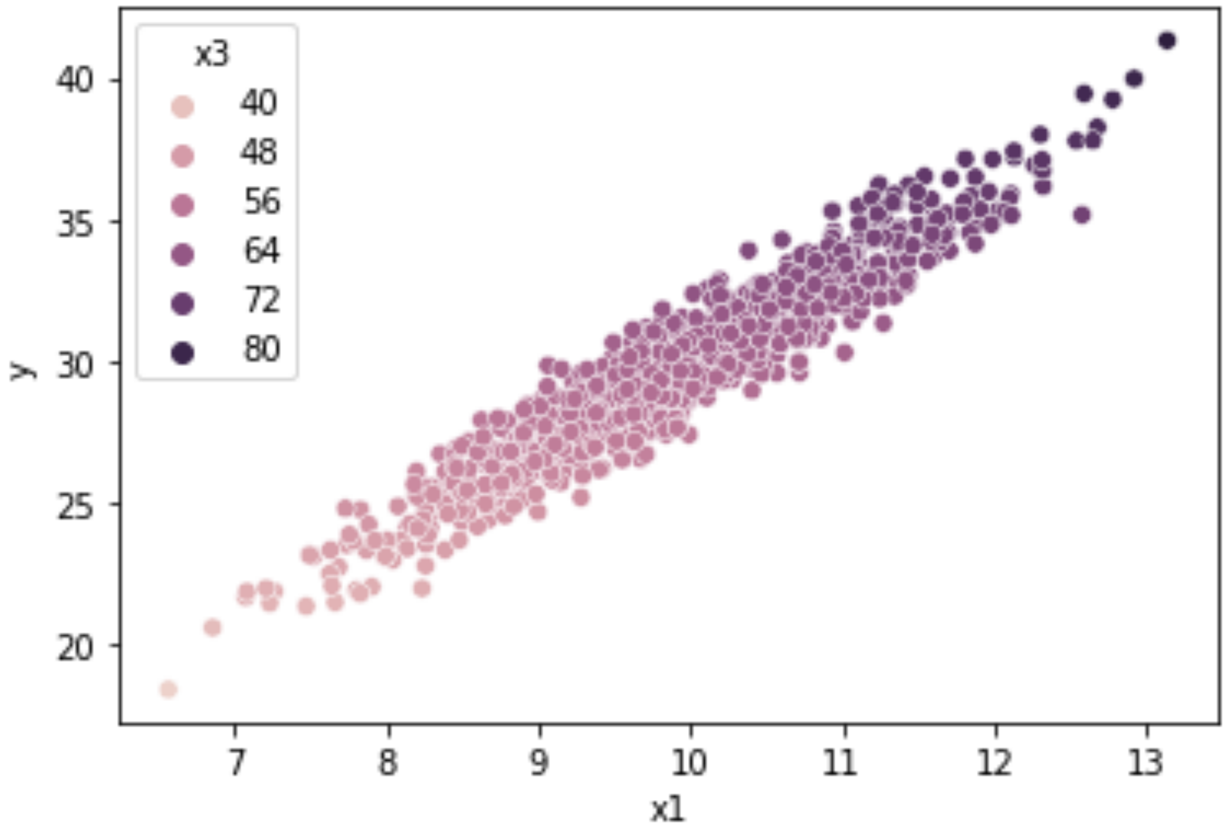
$D = 1$

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

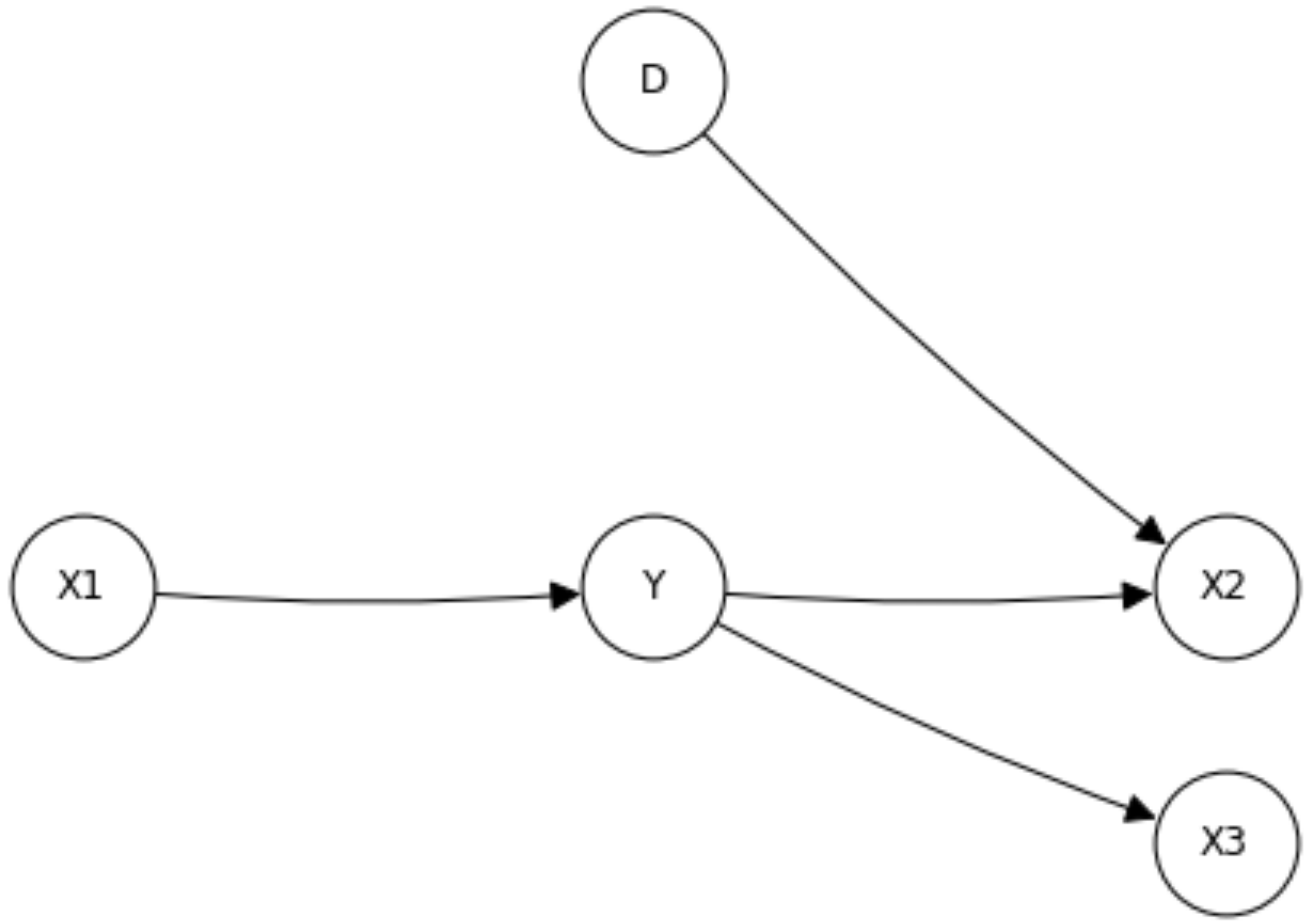


$D = 2$

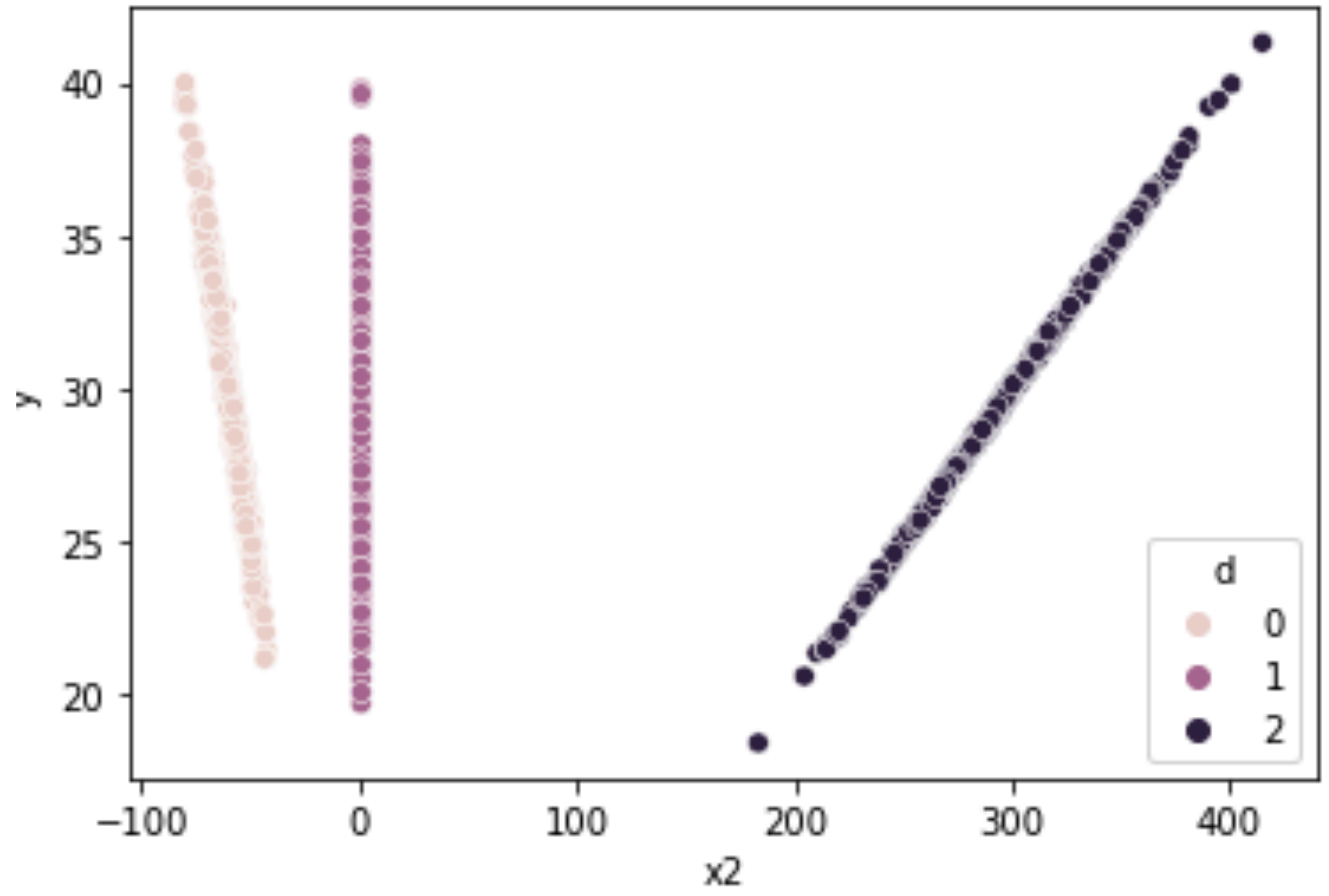
d	x1	y	x2	x3
2	9.671277	26.556214	265.034283	53.338139
2	9.613139	27.120226	270.746784	54.340341
2	10.718335	29.589532	295.318526	59.291053
2	9.002388	26.629254	264.942583	53.340389
2	9.289340	29.030355	289.747562	58.098312



Domain adaptation example - X2



Source domains Target domain

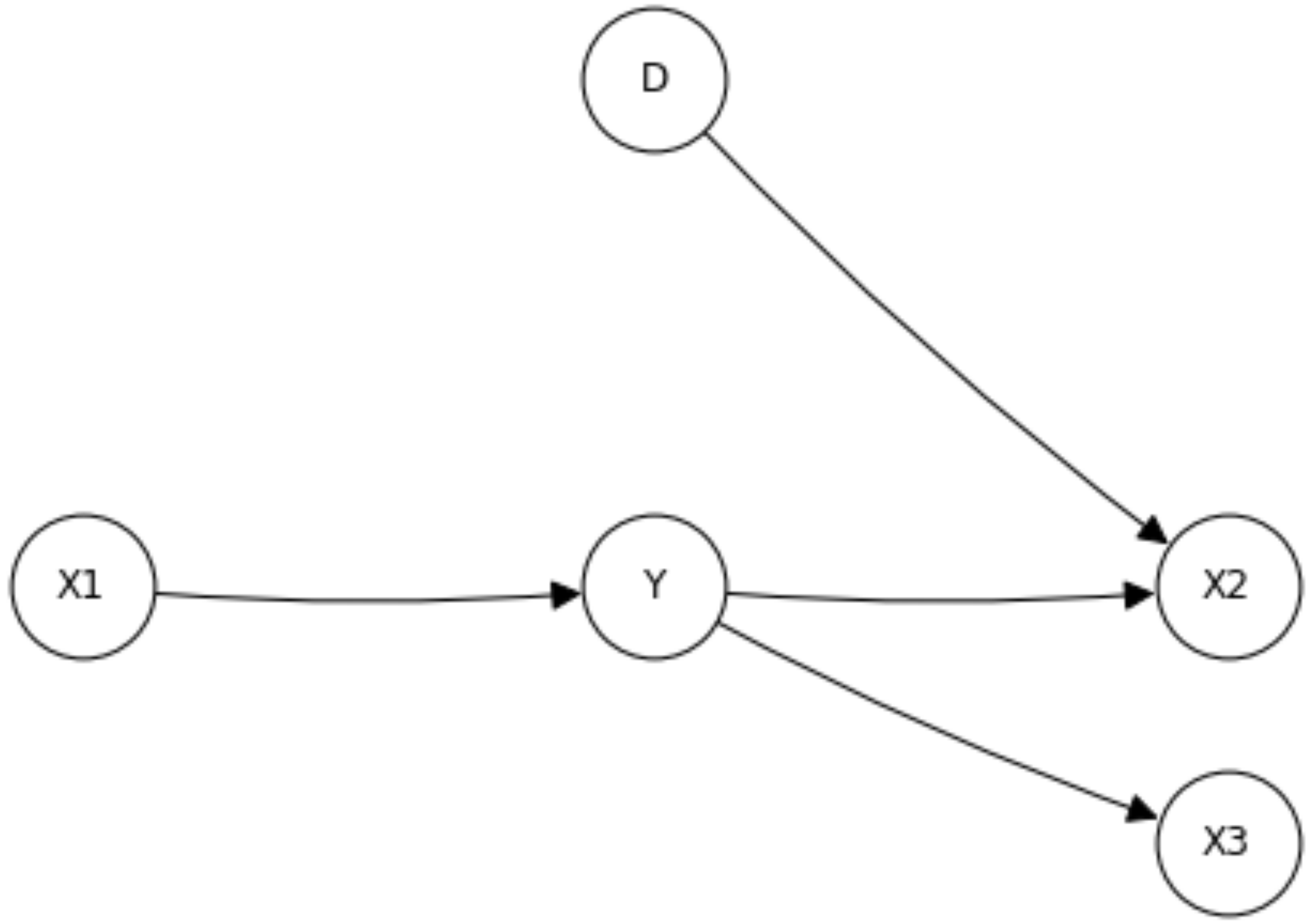


$P(Y \mid X_2)$ is not invariant

```
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 - X1



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

$p(Y|X_1)$ is invariant

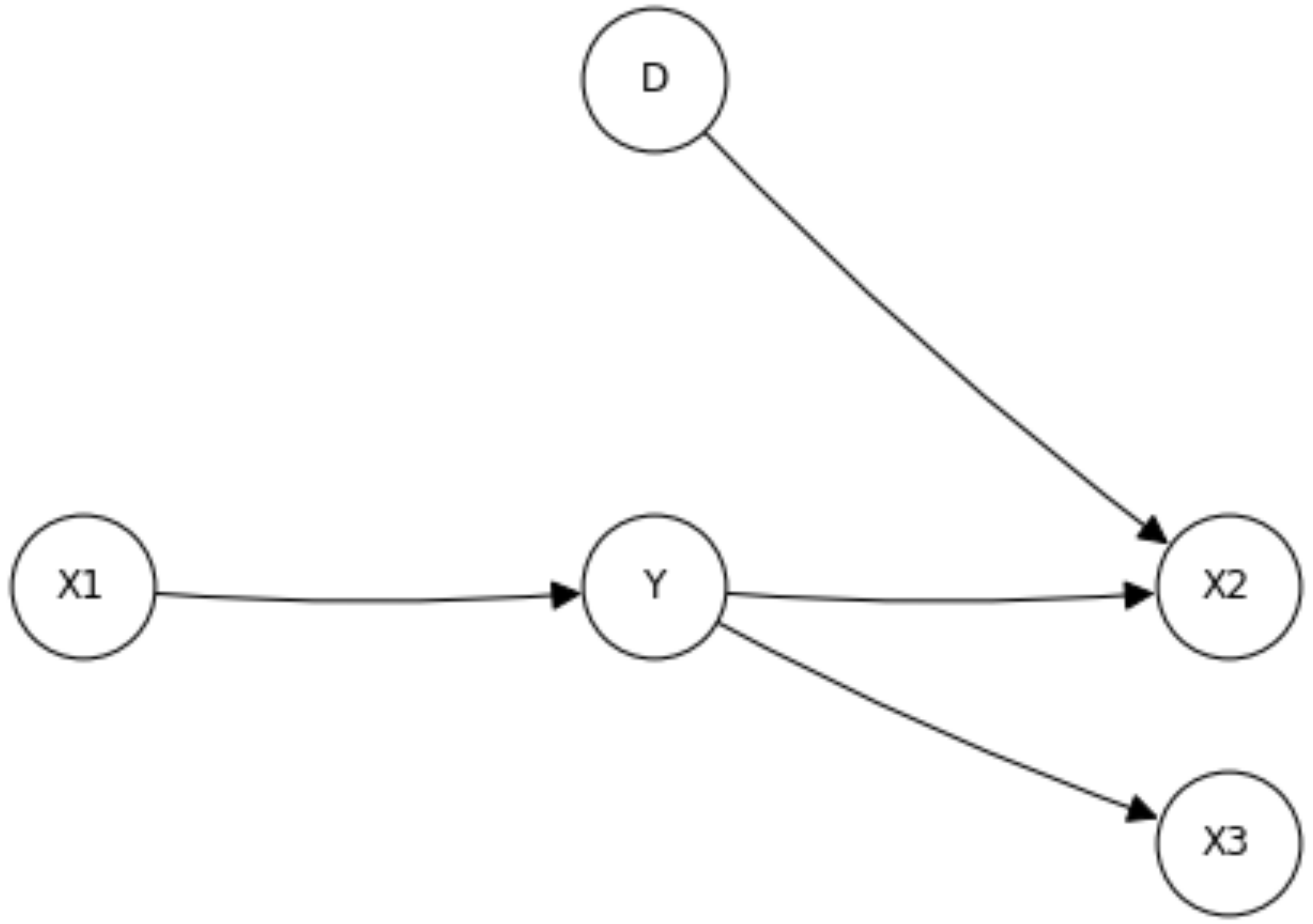
$$P(Y|X_1, D=0) = P(Y|X_1, D=1) = P(Y|X_1, D=2)$$
$$= P(Y|X_1)$$

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

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

d-separation [Pearl 1988]

Separating features intuition - X2



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

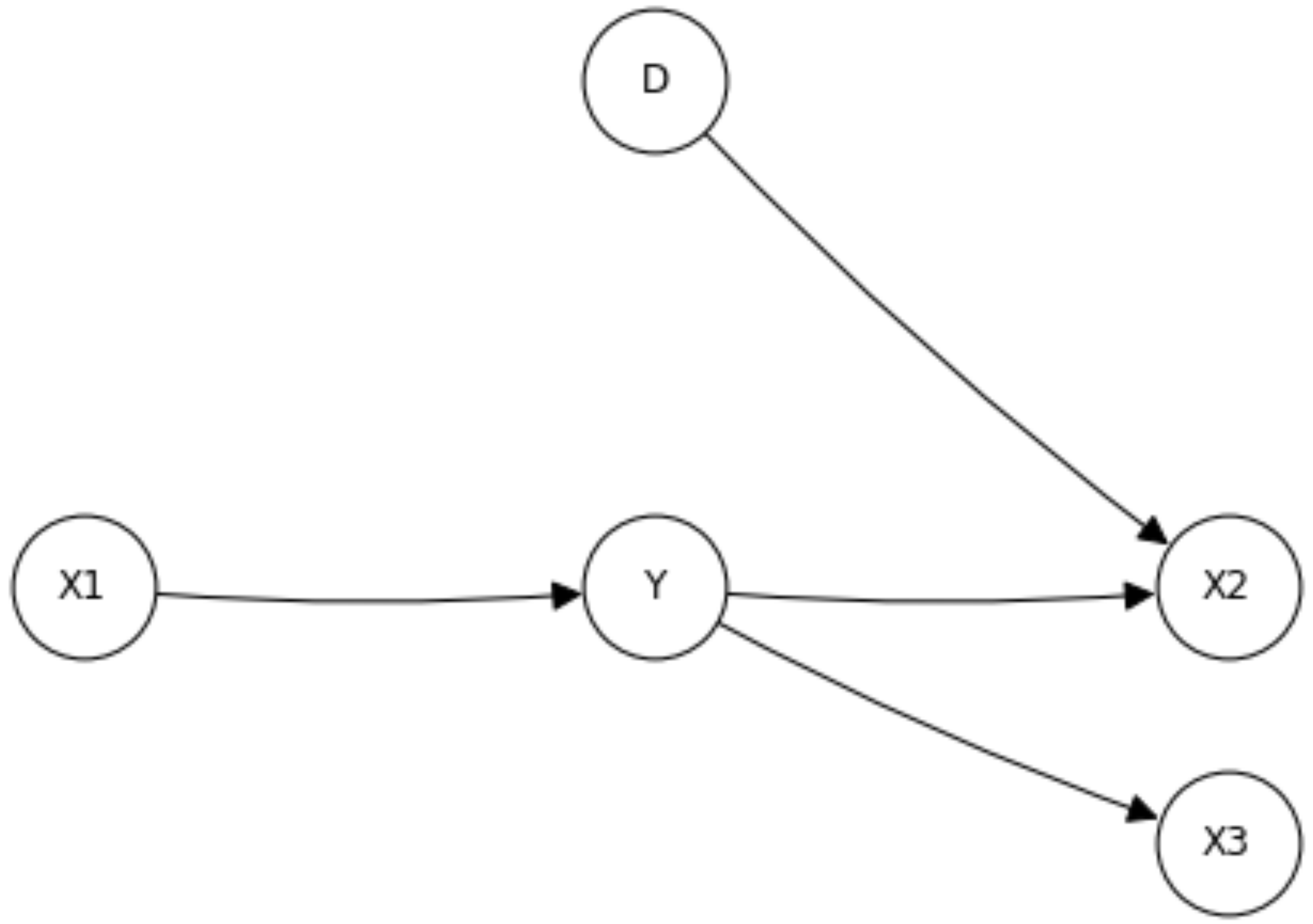
$$P(Y \mid X_2, D=0) \neq P(Y \mid X_2, D=1) \neq P(Y \mid X_2, D=2)$$

↳ 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 - X2



$P(Y \mid X_2)$ is <u>not</u> 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 $Y \not\perp D \mid X_2$

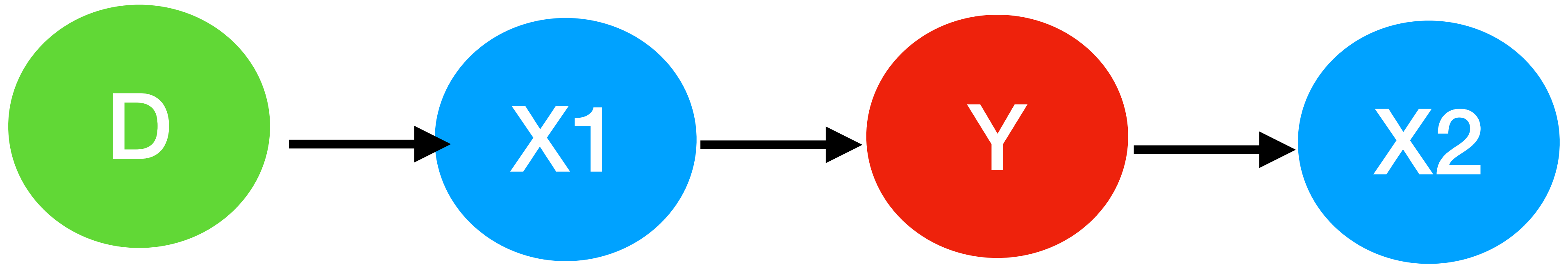
$Y \not\perp_d D \mid X_2$

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

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

Separating features [Magliacane et al 2018]

Common misconceptions: 1. An invariant feature need not be causal

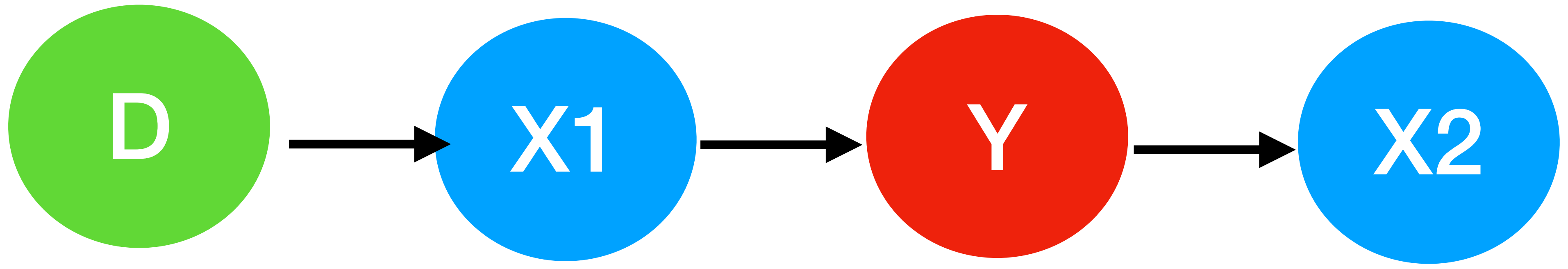


$$Y \perp\!\!\!\perp D \mid X_1$$

$$Y \perp\!\!\!\perp D \mid X_1, X_2$$

- Y|X1,X2 is invariant $\Longrightarrow$ invariant features are not necessarily parents of Y

Common misconceptions: 1. An invariant feature need not be causal



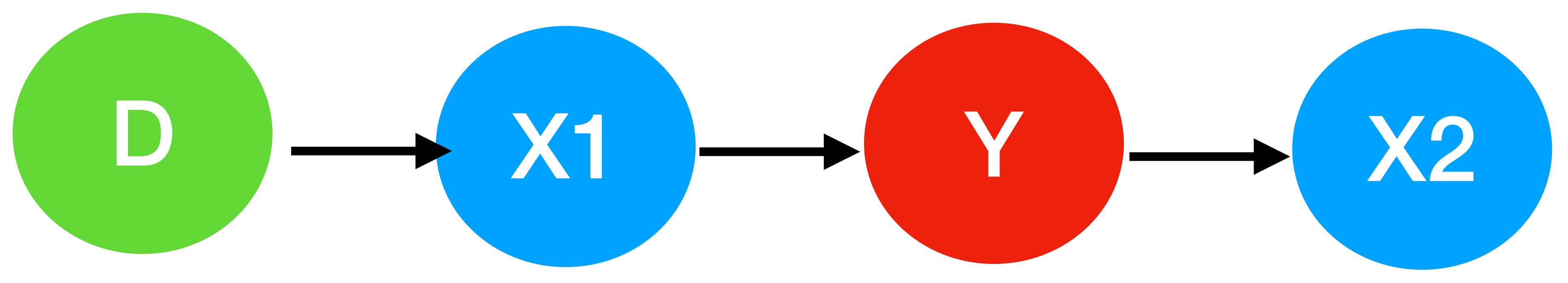
$$Y \perp\!\!\!\perp D \mid X_1$$

$$Y \perp\!\!\!\perp D \mid X_1, X_2$$

- Y|X1,X2 is invariant $\implies$ invariant features are not necessarily parents of Y

Invariant feature across "many different datasets" is not enough in general to find causal parents, need more assumptions

Common misconceptions: 1. An invariant feature need not be causal



$$Y \perp\!\!\!\perp D \mid X_1$$

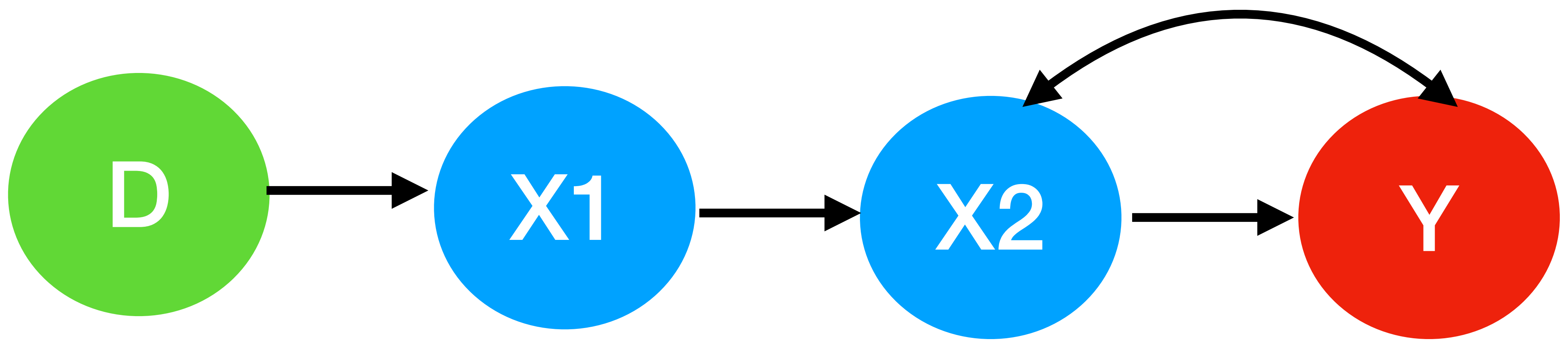
$$Y \perp\!\!\!\perp D \mid X_1, X_2$$

- Y|X1,X2 is invariant $\Longrightarrow$ invariant features are not necessarily parents of Y

- Invariant Causal Prediction [Peters et al. 2016] under causal sufficiency:

$$\mathbf{S}^* = \bigcap_{Y \perp\!\!\!\perp D \mid \mathbf{S}} \mathbf{S} \subseteq Pa(Y) \qquad \{X_1, X_2\} \cap \{X_1\} = \{X_1\}$$

Common misconception 2: Parents are not enough under latent confounding



$Y \perp\!\!\!\perp D \mid X_1$

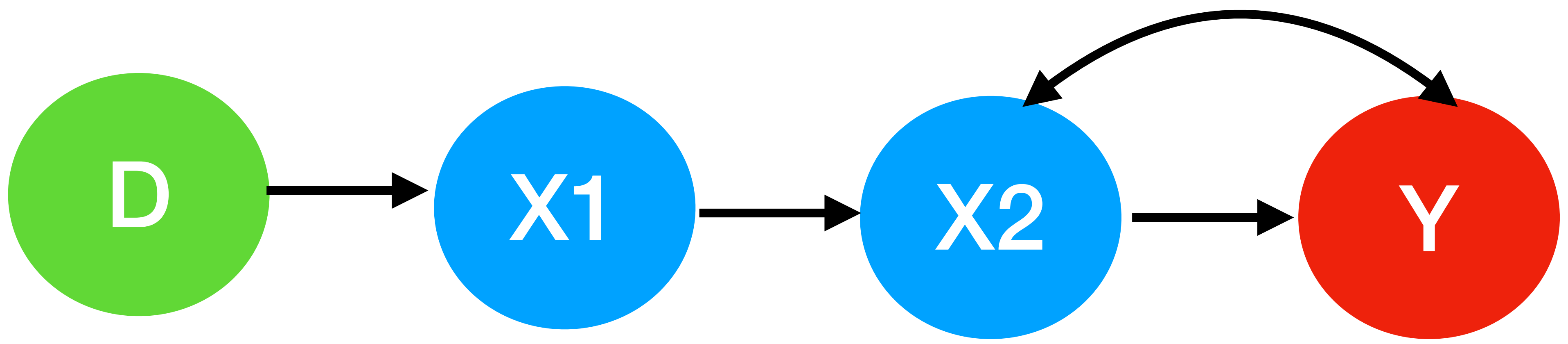
$Y \not\!\perp\!\!\!\perp D \mid X_2$

$Y \not\!\perp\!\!\!\perp D \mid X_1, X_2$

- Y|X1 is invariant, Y|X2 is not

Even if we knew all the parents, under latent confounding this wouldn't necessarily help transfer

Common misconception 2: Parents are not enough under latent confounding



$Y \perp\!\!\!\perp D \mid X_1$

$Y \not\!\perp\!\!\!\perp D \mid X_2$

$Y \not\!\perp\!\!\!\perp D \mid X_1, X_2$

- Y|X1 is invariant, Y|X2 is not

Even if we knew all the parents, under latent confounding this wouldn't necessarily help transfer

- **Conclusion:** causality (e.g. using the causal parents, learning the complete causal graph) is **neither necessary or sufficient*** for transfer

Desiderata for a causality inspired domain adaptation method

- **X,** Y and changes can be represented by an **unknown** causal graph
- Allow for **latent confounders**
- Avoid **parametric assumptions,** allow for heterogeneous effects across domains

Desiderata for a causality inspired domain adaptation method

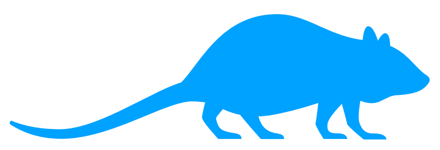
- **X,** Y and changes can be represented by an **unknown** causal graph
- Allow for **latent confounders**
- Avoid **parametric assumptions,** allow for heterogeneous effects across domains
- Instead of modeling **changes between each domain**, distinguish the change between the **mixture of sources and the target**

Desiderata for a causality inspired domain adaptation method

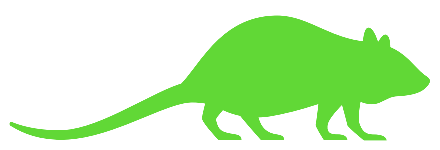
- **X,** Y and changes can be represented by an **unknown** causal graph
- Allow for **latent confounders**
- Avoid **parametric assumptions,** allow for heterogeneous effects across domains
- Instead of modeling **changes between each domain**, distinguish the change between the **mixture of sources and the target**
- Avoid common assumption that **if Y| T(X) is invariant across multiple source domains,** then **Y|T(X) is invariant** also in the **target domain**
  - This also (implicitly) assumed by methods based on the idea that invariance => causality

Causal domain adaptation problem
[Magliacane et al. 2018]

- Unsupervised **multi-source** domain adaptation
- 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



	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0



	X1	X2	Y
Gene A	3.1	2	**1**
Gene A	3.2	3	**1**
Gene A	4	1	**1**
Gene A	3.2	3	**0**

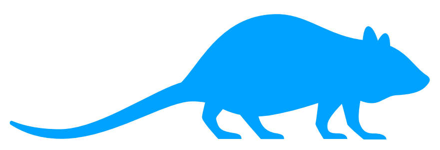
	X1	X2	Y
Gene B	0.2	1	**?**
Gene B	0.3	1	**?**
Gene B	0.3	2	**?**
Gene B	0.4	1	**?**

Causal domain adaptation

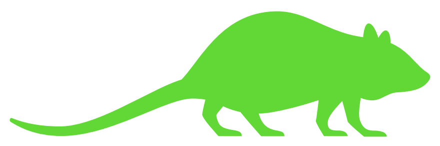
[Magliacane et al. 2018

Multiple context variable
C1, C2 ...

- Unsupervised **multi-source** domain adaptation
- 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



	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0



	X1	X2	Y
Gene A	3.1	2	1
Gene A	3.2	3	1
Gene A	4	1	1
Gene A	3.2	3	0

	X1	X2	Y
Gene B	0.2	1	?
Gene B	0.3	1	?
Gene B	0.3	2	?
Gene B	0.4	1	?

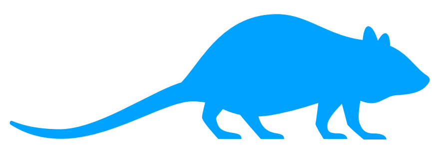
Causal domain adaptation problem

[Magliacane et al. 2018]

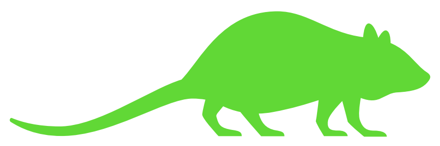
- Unsupervised **multi-source** domain adap

C1 = 1

- 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



	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0



	X1	X2	Y
Gene A	3.1	2	1
Gene A	3.2	3	1
Gene A	4	1	1
Gene A	3.2	3	0

	X1	X2	Y
Gene B	0.2	1	?
Gene B	0.3	1	?
Gene B	0.3	2	?
Gene B	0.4	1	?

Causal domain adaptation problem
[Magliacane et al. 2018]

- Unsupervised **multi-source** domain adaptation
- 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

	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0
Normal	1.1	2	1
Normal	0.1	3	0

	X1	X2	Y
Gene A	3.1	2	1
Gene A	3.2	3	1
Gene A	4	1	1
Gene A	3.2	3	0

	X1	X2	Y
		1	?
			?
		2	?
		1	?

Now the graph is unknown!

Joint Causal Inference [Mooij et al. 2020]

- We represent jointly different distributions as an **unknown** **single causal graph**
- Instead of a single domain variable, we **add several context variables** so we can **disentangle** changes in distribution across the datasets
  - If we know nothing about the changes in the datasets, we use indicator variables

	X1	X2	Y
Normal	0.1	2	0
Normal	0.2	3	0

	X1	X2	Y
Gene A	3.1	2	**1**
Gene A	3.2	3	**1**

	X1	X2	Y
Gene B	0.2	1	?
Gene B	0.3	1	?
Gene B	0.3	2	?
Gene B	0.4	1	?

C1	C2	X1	X2	Y
0	0	0.1	2	0
0	0	0.2	3	0
0	0	1.1	2	1
0	0	0.1	3	0
1	0	3.1	2	**1**
1	0	3.2	3	**1**
1	0	4	1	**1**
1	0	3.2	3	**0**
0	1	0.2	1	?
0	1	0.3	1	?
0	1	0.3	2	?
0	1	0.4	1	?

Joint Causal Inference [Mooij et al. 2020]

- We can learn an equivalence class of the unknown **single causal graph** using **conditional independence tests** on **systematically pooled data**

- We treat context variables as normal variables that we know are uncaused

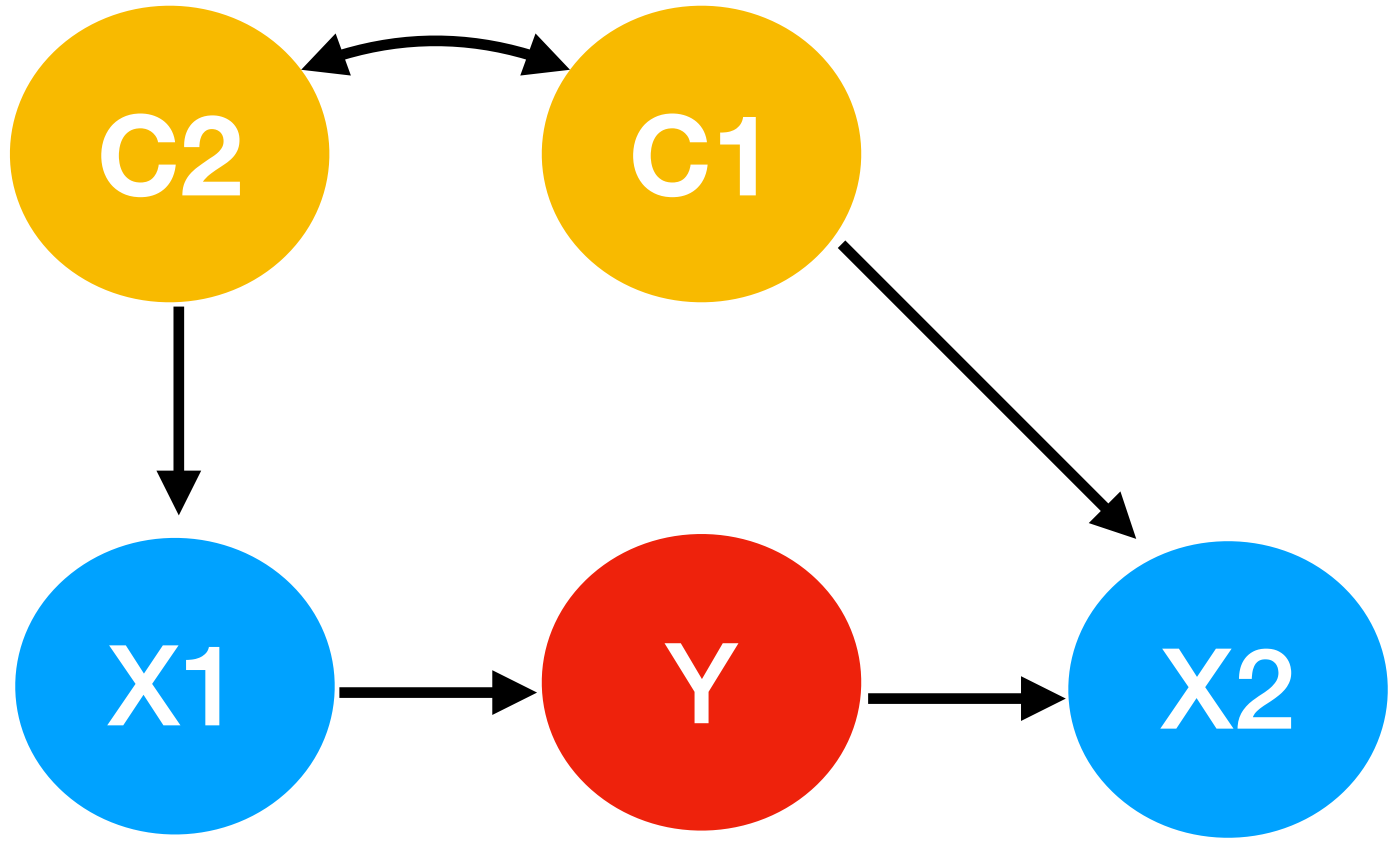
C1	C2	X1	X2	Y
0	0	0.1	1	0
0	0	0.2	1	0
0	0	1.1	2	1
1	0	3.1	2	1
1	0	3.2	3	1
1	0	4	3	1
0	1	0.2	0	0
0	1	0.3	0	1
0	1	0.3	1	0

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

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

$$X_1 \perp\!\!\!\perp X_2 \mid Y, C_2$$

$$\ldots$$



Joint Causal Inference [Mooij et al. 2020]

- We can learn an equivalence c… …ingle causal graph using conditional inde… …pooled data

If there is no missing data - Markov equivalence class

- We treat context variables as … …that we know are uncaused

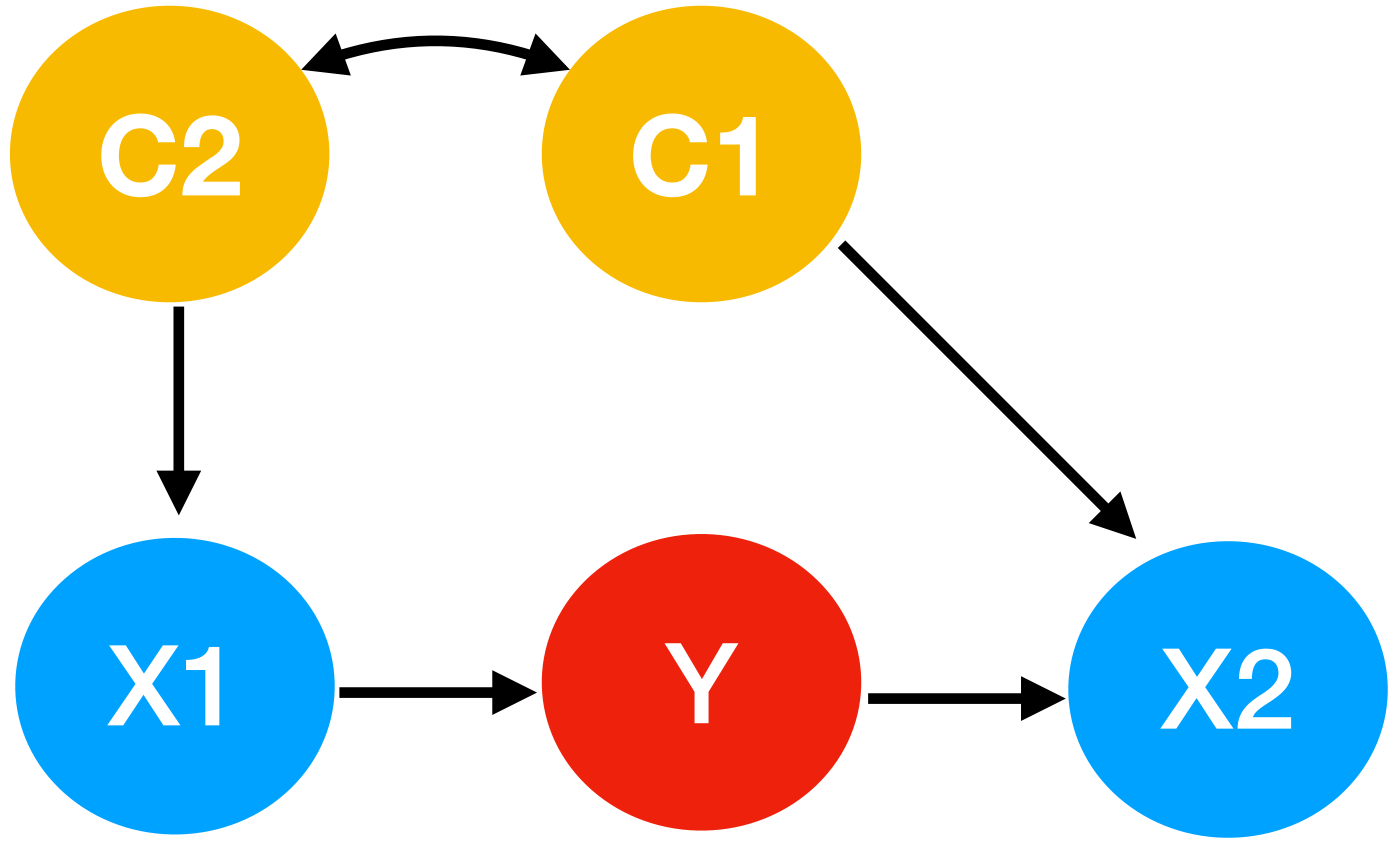
C1	C2	X1	X2	Y
0	0	0.1	1	0
0	0	0.2	1	0
0	0	1.1	2	1
1	0	3.1	2	1
1	0	3.2	3	1
1	0	4	3	1
0	1	0.2	0	0
0	1	0.3	0	1
0	1	0.3	1	0

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

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

$$X_1 \perp\!\!\!\perp X_2 \mid Y, C_2$$

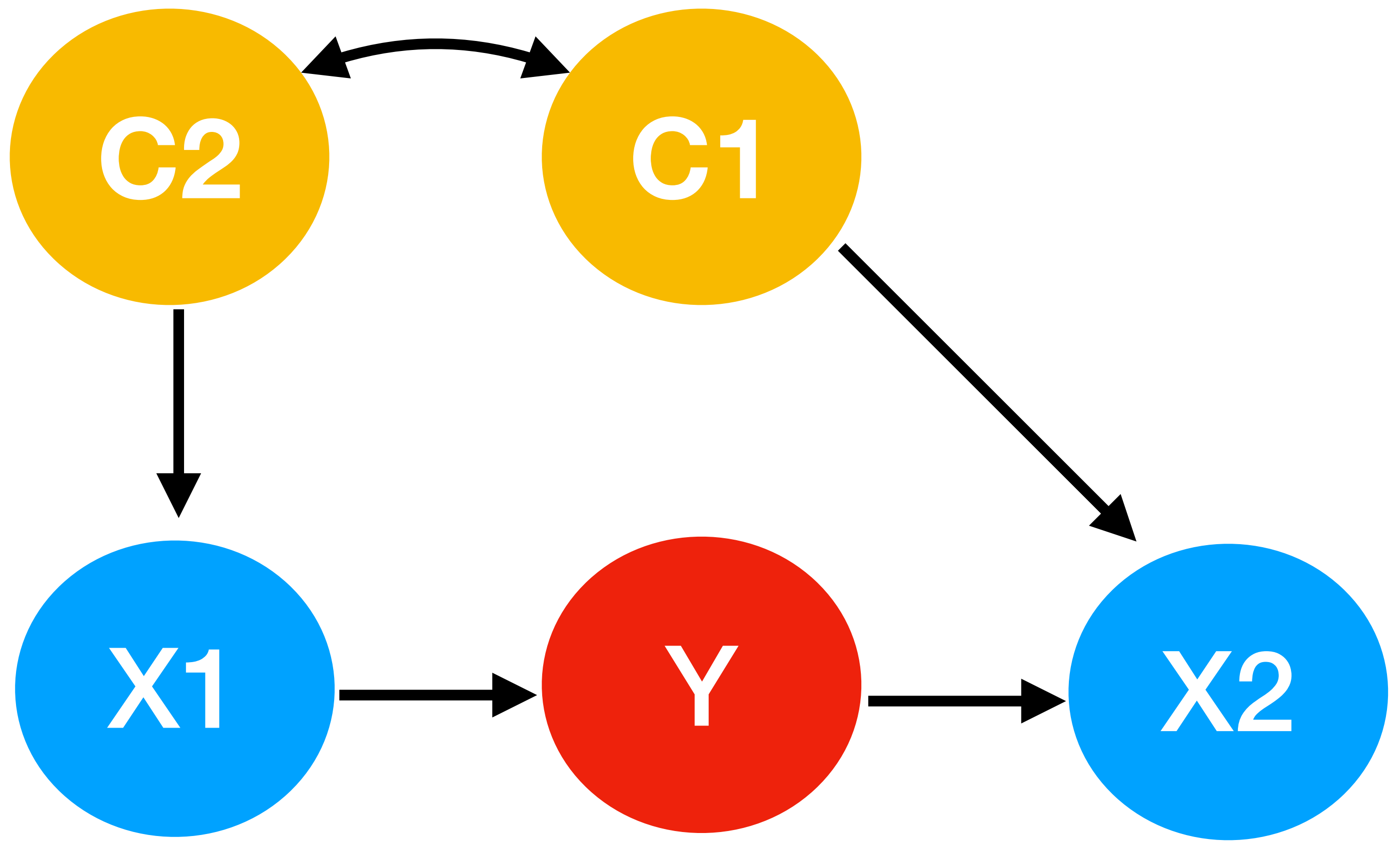
…



Causal domain adaptation: separating features

- **Separating features:** sets of features that d-separate Y from the context variable C1 representing the target domain

Aka stable features, invariant features etc.



- **{X1}** is a separating feature set, {X1, X2} could lead to arbitrary large error

What if the causal graph is unknown?

- **Idea**: we could test the conditional independence in the data

$$Y \perp\!\!\!\perp C_1 \,|\, X_1 ? \qquad Y \perp\!\!\!\perp C_1 \,|\, X_2 ?$$

What if the causal graph is unknown?

- **Idea**: we could test the conditional independence in the data

  ~~$Y \perp\!\!\!\perp C_1 | X_1 ?$~~ ~~$Y \perp\!\!\!\perp C_1 | X_2 ?$~~

- **Problem:** Y is always missing when C1=1, so we cannot test these

C1	C2	X1	X2	Y
0	0	0.1	1	0
0	0	0.2	1	0
0	0	1.1	2	1
1	0	3.1	2	?
1	0	3.2	3	?
1	0	4	3	?
0	1	0.2	0	0
0	1	0.3	0	1
0	1	0.3	1	0

What if the causal graph is unknown?

- **Idea**: we could test the conditional independence in the data

  ~~$Y \perp\!\!\!\perp C_1 | X_1 ?$~~ ~~$Y \perp\!\!\!\perp C_1 | X_2 ?$~~

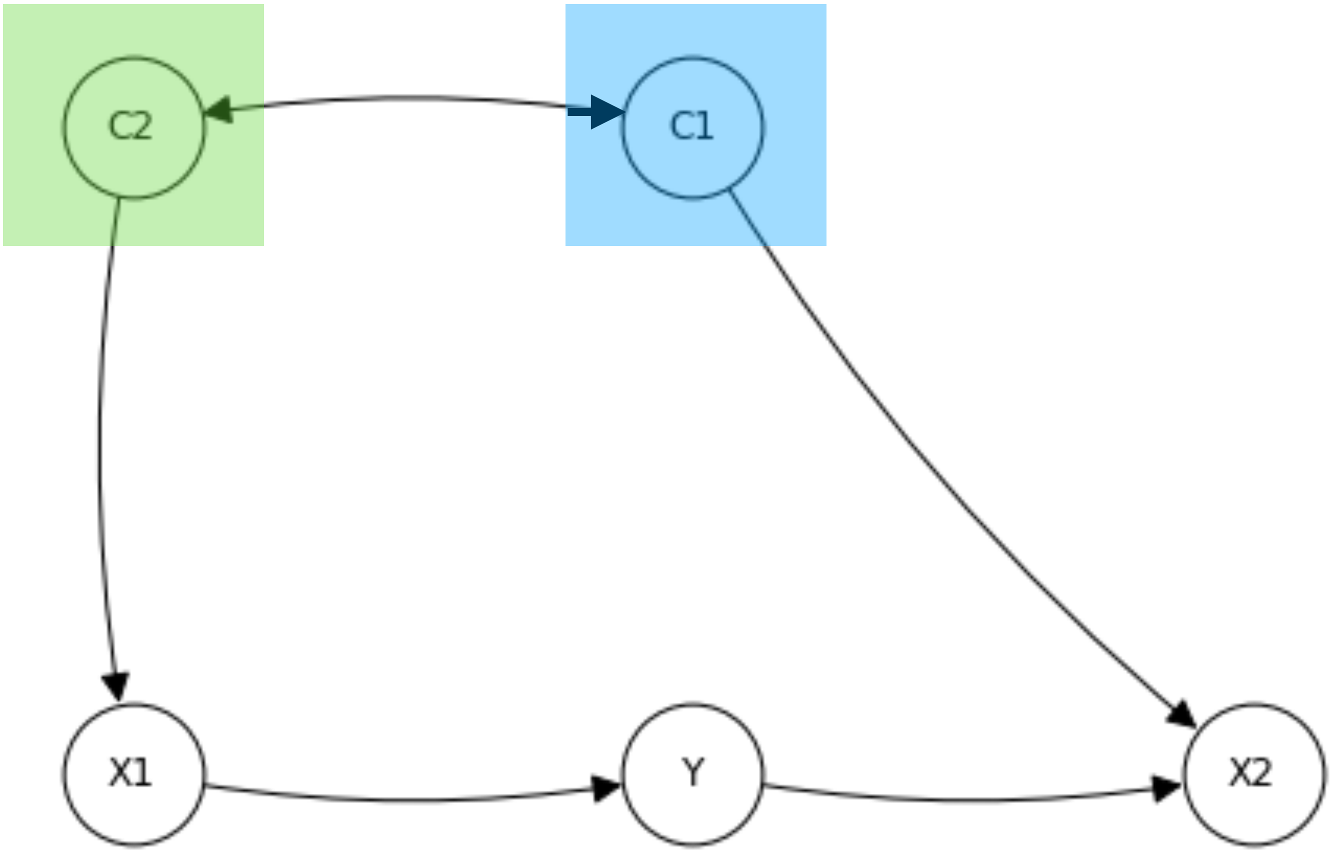
- **Problem:** Y is always missing when C1=1, so we cannot test these

C1	C2	X1	X2	Y
0	0	0.1	1	0
0	0	0.2	1	0
0	0	1.1	2	1
1	0	3.1	2	?
1	0	3.2	3	?
1	0	4	3	?
0	1	0.2	0	0
0	1	0.3	0	1
0	1	0.3	1	0

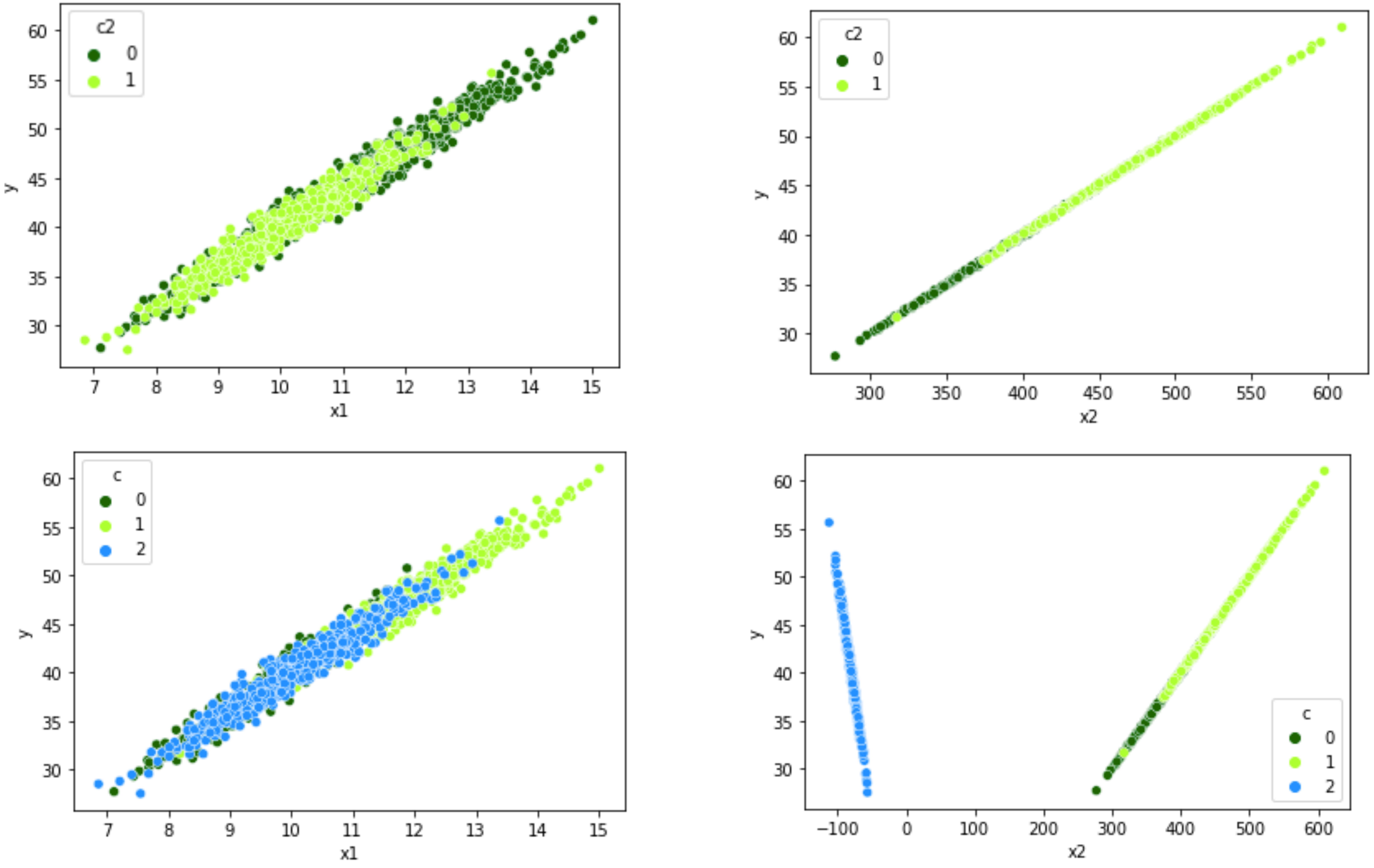
Idea: Separating features in sources are also separating in target

$$Y \perp\!\!\!\perp C_2 | X_1 \implies Y \perp\!\!\!\perp C_1 | X_1$$

Separating features in sources are also separating in target - counterexample




```
c1 = epsilon_c1
c2 = epsilon_c2
x1 = epsilon_x1 + 10  + 2 * c2
y = 4 * x1 + epsilon_y
x2 = - 2 * y * c1 + epsilon_x2  + 10 * y *(1-c1)
```



What if the causal graph is unknown?

- **Idea:** we could test the conditional independence in the data

  ~~$Y \perp\!\!\!\perp C_1 \mid X_1$?~~ ~~$Y \perp\!\!\!\perp C_2 \mid X_2$?~~

- **Problem:** Y is always missing when C1=1, so we cannot test these

C1	C2	X1	X2	Y
0	0	0.1	1	0
0	0	0.2	1	0
0	0	1.1	2	1
1	0	3.1	2	?
1	0	3.2	3	?
1	0	4	3	?
0	1	0.2	0	0
0	1	0.3	0	1
0	1	0.3	1	0

$$X_1 \not\perp\!\!\!\perp X_2$$
$$X_1 \not\perp\!\!\!\perp C_1$$
$$X_1 \not\perp\!\!\!\perp X_2 \mid C_1$$
$$X_1 \perp\!\!\!\perp X_2 \mid Y, C_1 = 0$$
$$\ldots$$

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

Assumptions [Magliacane et al. 2018]

- We assume that there exists an **acyclic** causal graph that fits all the data (Joint Causal Inference)
- We assume **Y cannot be intervened upon directly**

Assumptions [Magliacane et al. 2018]

- We assume that there exists an **acyclic** causal graph that fits all the data (Joint Causal Inference)
- We assume **Y cannot be intervened upon directly**
- We assume **no extra dependences involving Y** in target domain C1=1

$A, D, \mathbf{B} \subset \mathbf{V} \setminus \{Y, C_1\}$

$$Y \perp\!\!\!\perp A \mid \mathbf{B}, C_1 = 0 \implies Y \perp\!\!\!\perp A \mid \mathbf{B}, C_1 = 1$$

$$A \perp\!\!\!\perp D \mid \mathbf{B}, Y, C_1 = 0 \implies A \perp\!\!\!\perp D \mid \mathbf{B}, Y, C_1 = 1$$

There can be extra independences in the target

A simple example

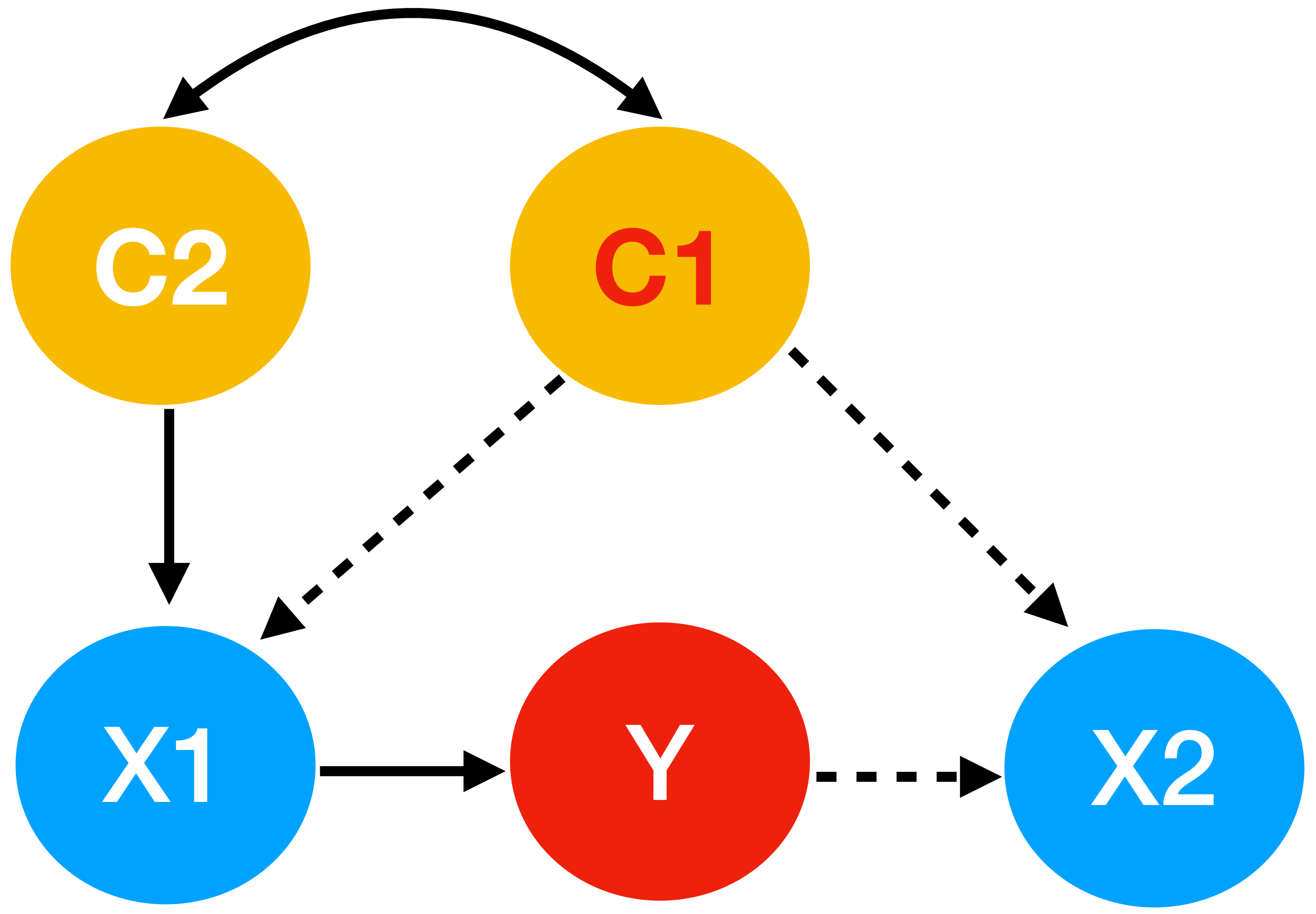
C1	C2	X1	X2	Y
0	0	0.1	1	0
0	0	0.2	1	0
0	0	1.1	2	1
0	1	3.1	2	1
0	1	3.2	3	1
0	1	4	3	1
1	0	0.2	0	?
1	0	0.3	0	?
1	0	0.3	1	?

$$Y \not\perp\!\!\!\perp C_2 \,|\, C_1 = 0$$

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

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

Perform allowed CI tests

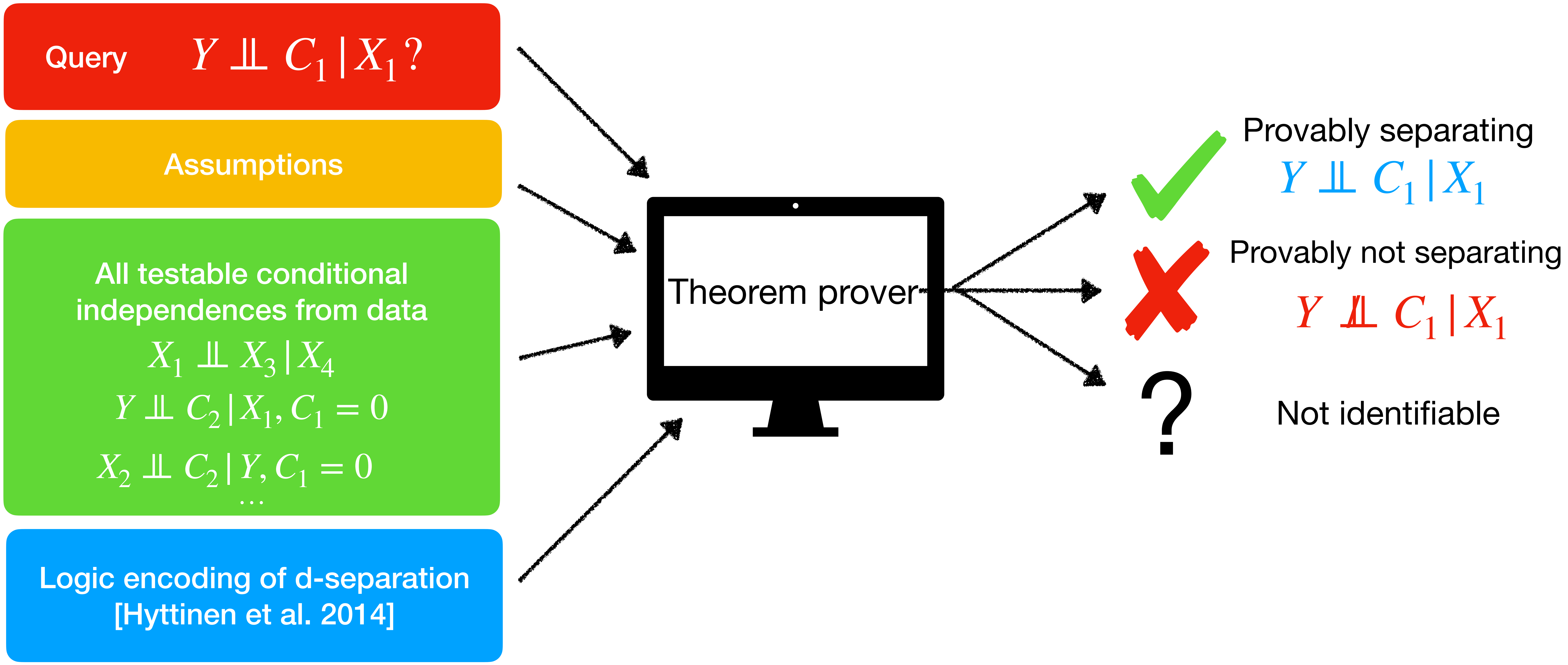


All possible compatible graphs

- We can prove untestable separating test **without reconstructing the graph:**

$Y \perp\!\!\!\perp C_1 \,|\, X_1$ True in all possible compatible graphs

Inferring separating sets without enumerating all possible causal graphs



A simple causal feature selection algorithm

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}, …)

A simple causal feature selection algorithm

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}

A simple causal feature selection algorithm

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

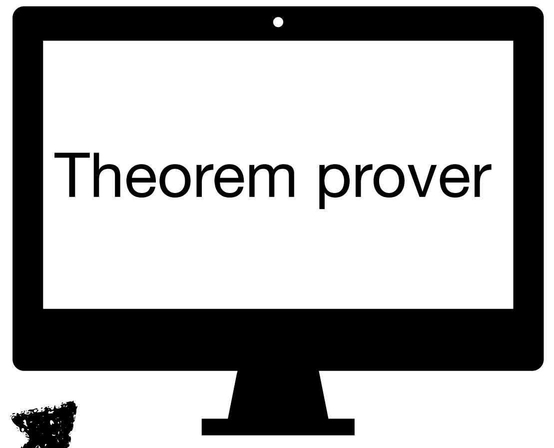
$X_1 \perp\!\!\!\perp X_3 \mid X_4$

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

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

…

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

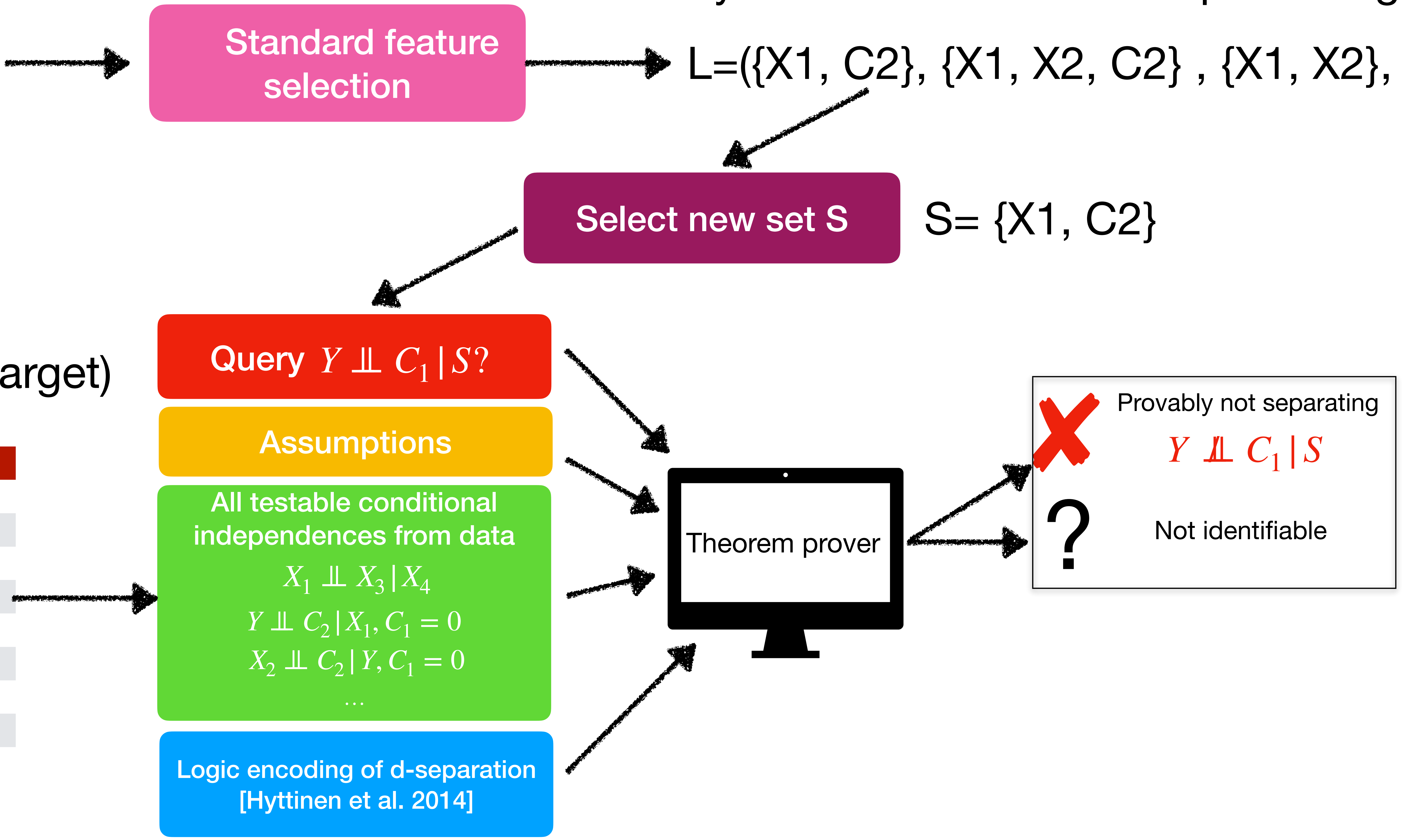


A simple causal feature selection algorithm

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

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



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	?

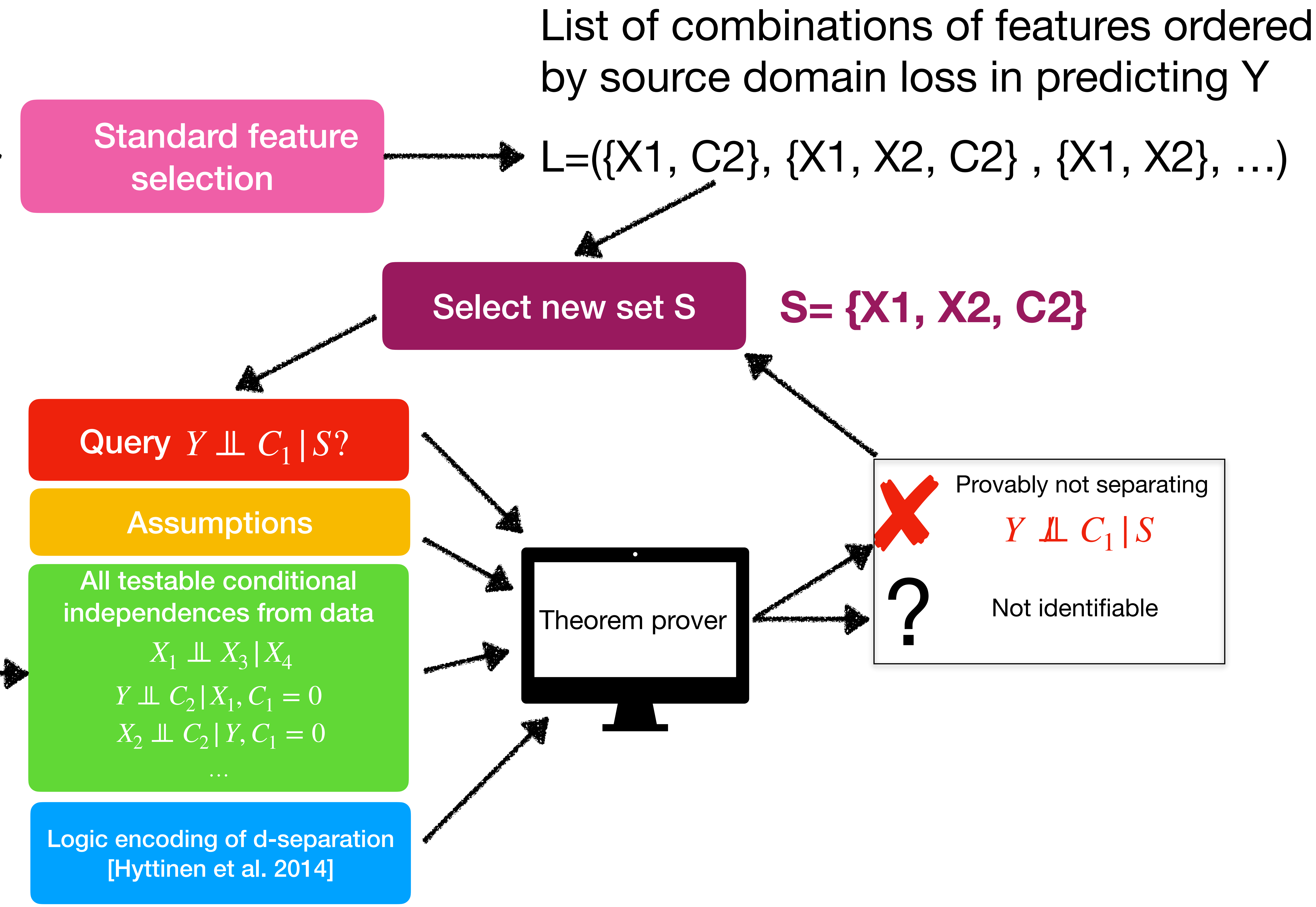
A simple causal feature selection algorithm

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

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	?



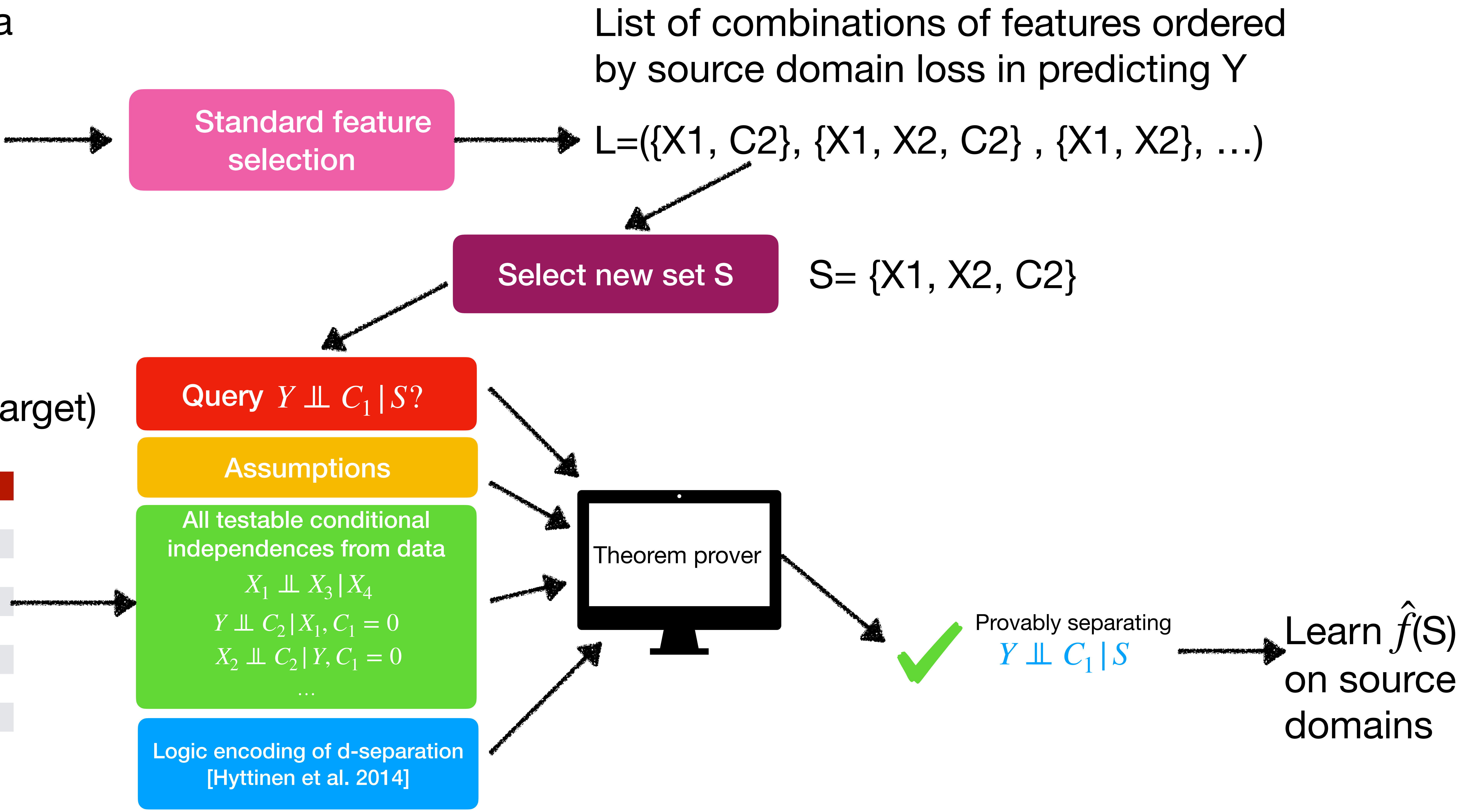
A simple causal feature selection algorithm

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

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	?



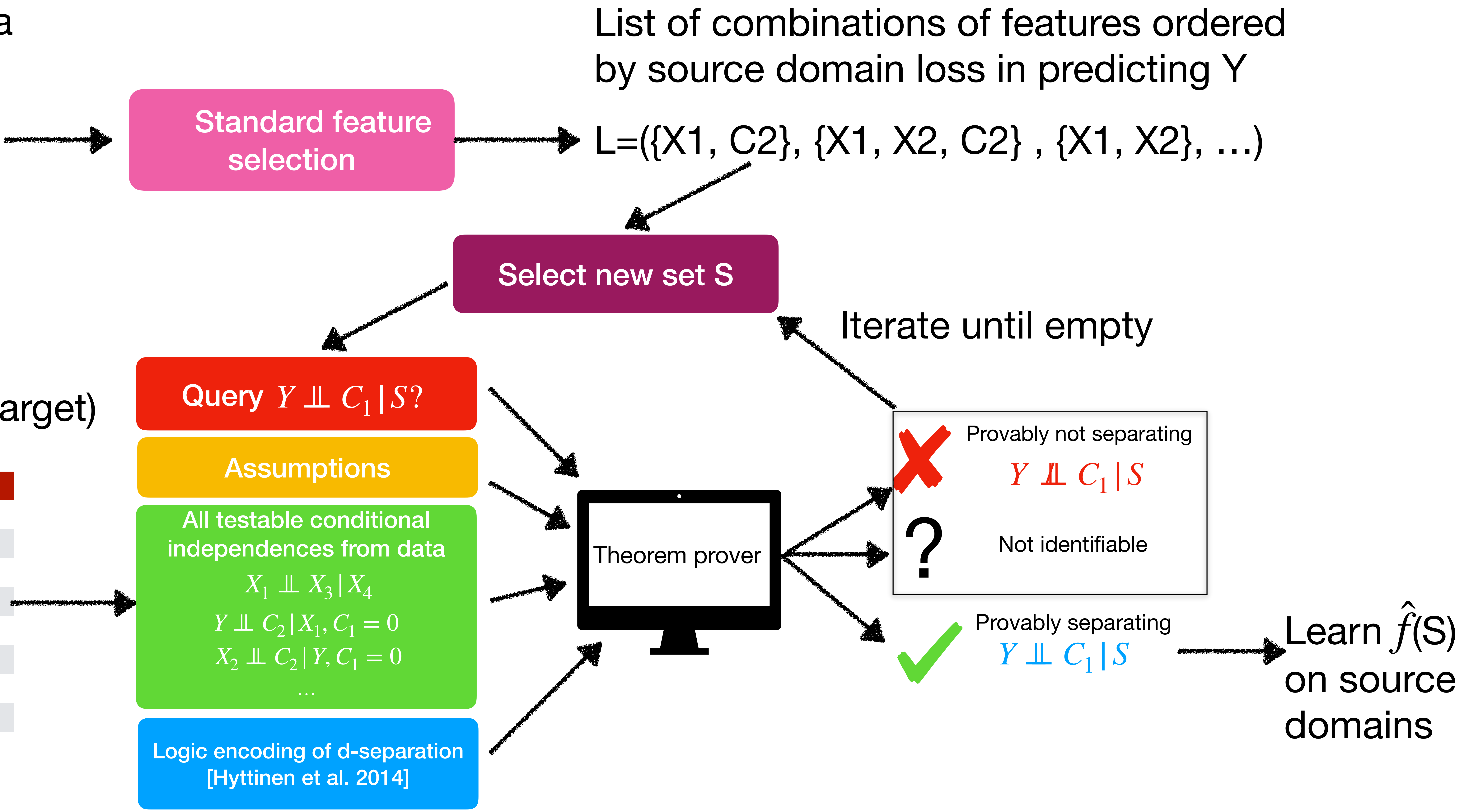
A simple causal feature selection algorithm

Source domains data

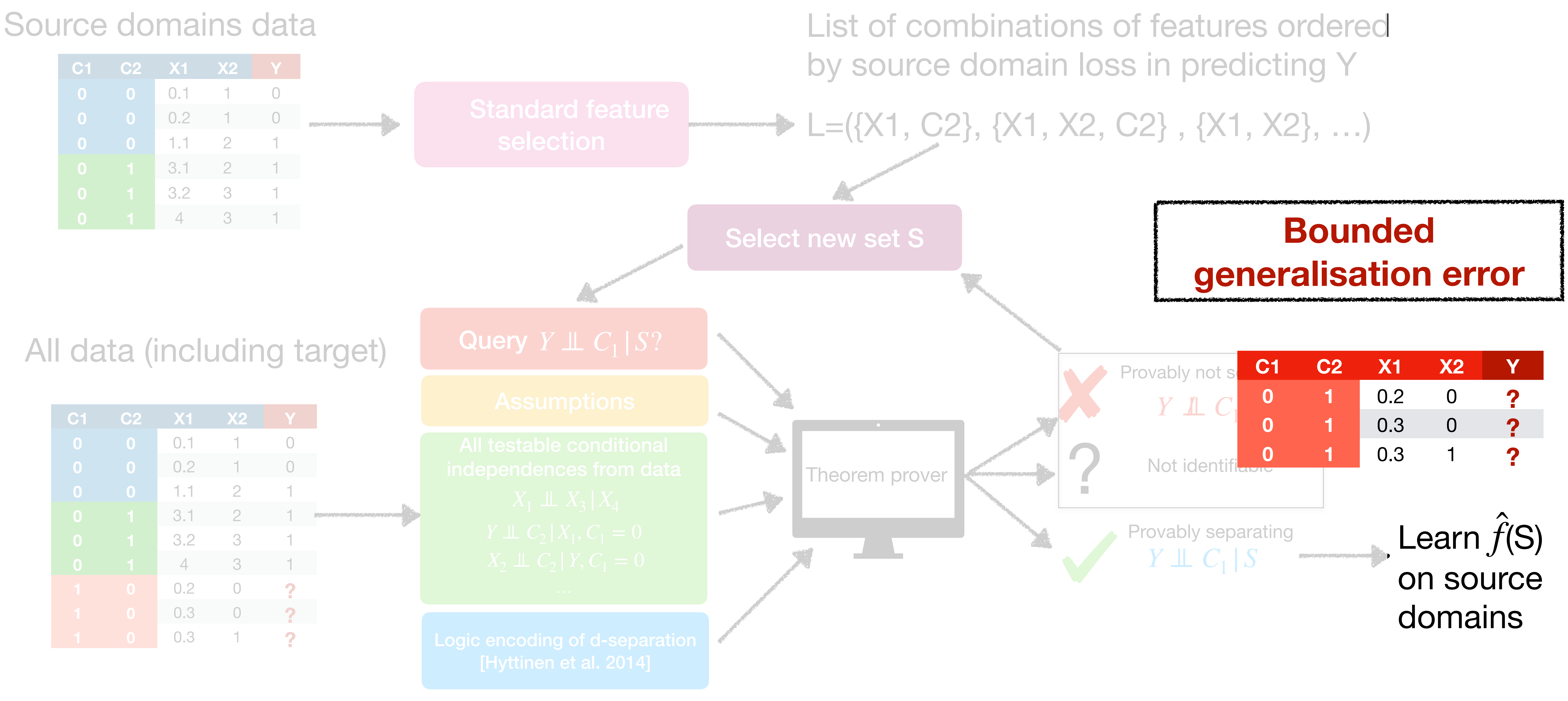
C1	C2	X1	X2	Y
0	0	0.1	1	0
0	0	0.2	1	0
0	0	1.1	2	1
0	1	3.1	2	1
0	1	3.2	3	1
0	1	4	3	1

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	?



A simple causal feature selection algorithm



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

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	?

C1	C2	X1	X2	Y
0	1	0.2	0	?
0	1	0.3	0	?
0	1	0.3	1	?

Desiderata for a causality inspired domain adaptation method

- **X,** Y and changes can be represented by an **unknown** causal graph
- Allow for **latent confounders**
- Avoid **parametric assumptions,** allow for heterogeneous effects across domains
- Instead of modeling **changes between each domain**, distinguish the change between the **mixture of sources and the target**
- Avoid common assumption that **if Y| T(X) is invariant across multiple source domains,** then **Y|T(X) is invariant** also in the **target domain**
- Only search for invariant features with respect to **current target task**

Thanks to JCI [Mooij et al 2020]

Desiderata for a causality inspired domain adaptation method

- **X,** Y and changes can be represented by an **unknown** causal graph
- Allow for **latent confounders**
- Avoid **parametric assumptions,** allow for heterogeneous effects across domains
- Instead of modeling **changes between each domain**, distinguish the change between the **mixture of sources and the target**
- Avoid common assumption that **if Y| T(X) is invariant across multiple source domains,** then **Y|T(X) is invariant** also in the **target domain**
- Only search for invariant features with respect to **current target task**

No need to find causal graph or equivalence class

Limitations and future work

- **Potentially too conservative:** Separating sets may exist that are not provably separating
    - Extension: can we use **active learning/intervention design** to decide most informative interventions?
- **Scalability:** using (error-correcting) logic-based encoding with all CI tests as input scales to tens of vars (including context variables)
    - Extension: use approximate algorithms, combine with low dim representations
- **Can we apply this to multi-task RL (e.g. in factored MDPs)?**

AdaRL: What, Where, and How to Adapt in Transfer RL

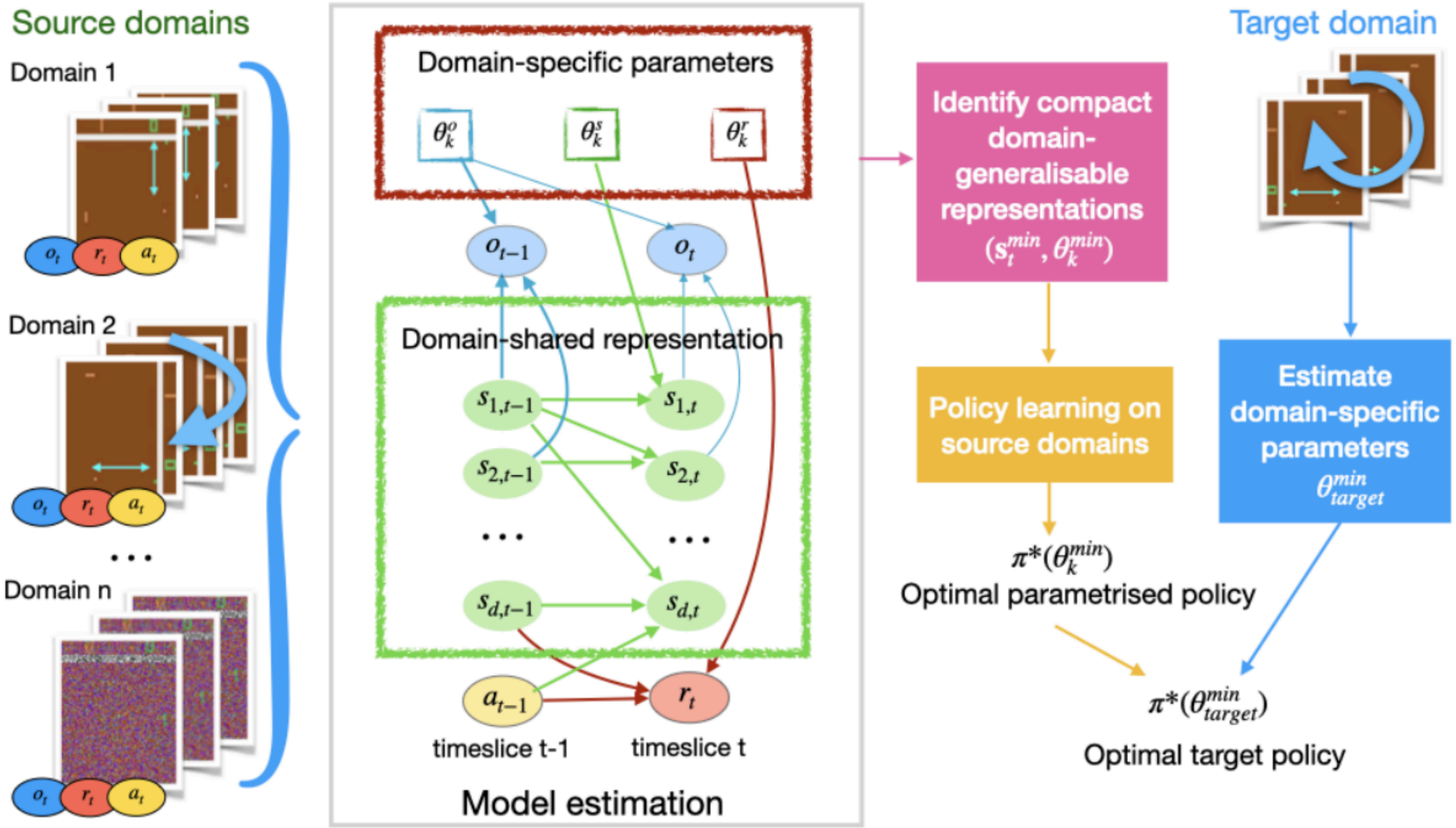
Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, Kun Zhang

- We have n source domains with random trajectories
- Learn a **factored** MDP (symbolic inputs) or POMDP (images) with **latent change factors** that are constant in each domain, but **vary across domains over sources**
  - Identify the minimal dimensions of the state and change factors that are **necessary and sufficient** for policy optimisation
- Learn a policy over all source domains, parametrised in the minimal change factors
- In the **target domain** learn the **value of the change factor** and apply this policy

https://arxiv.org/abs/2107.02729

AdaRL: What, Where, and How to Adapt in Transfer RL

Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, Kun Zhang



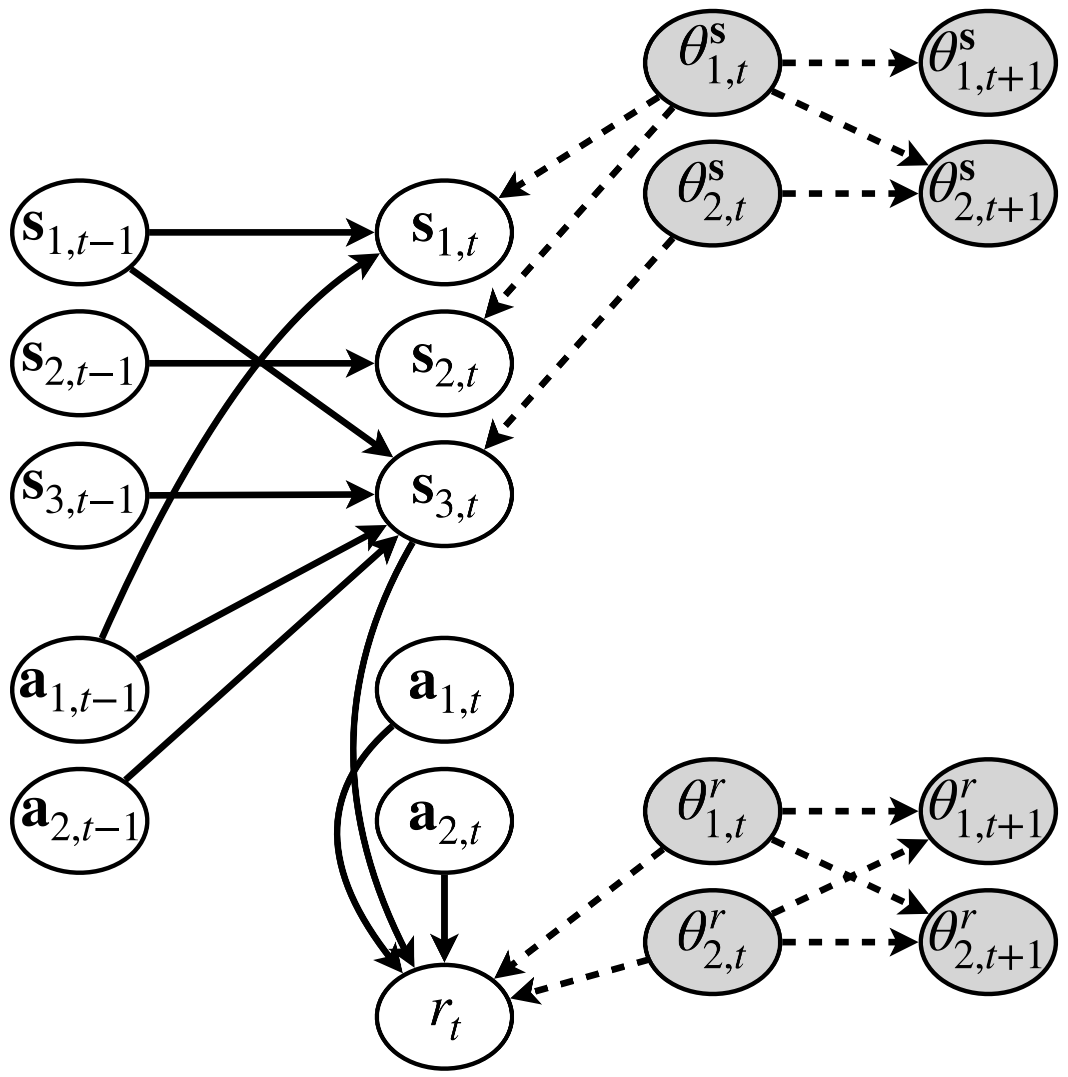
Simplifying assumption: no new edges in target domain

https://arxiv.org/abs/2107.02729

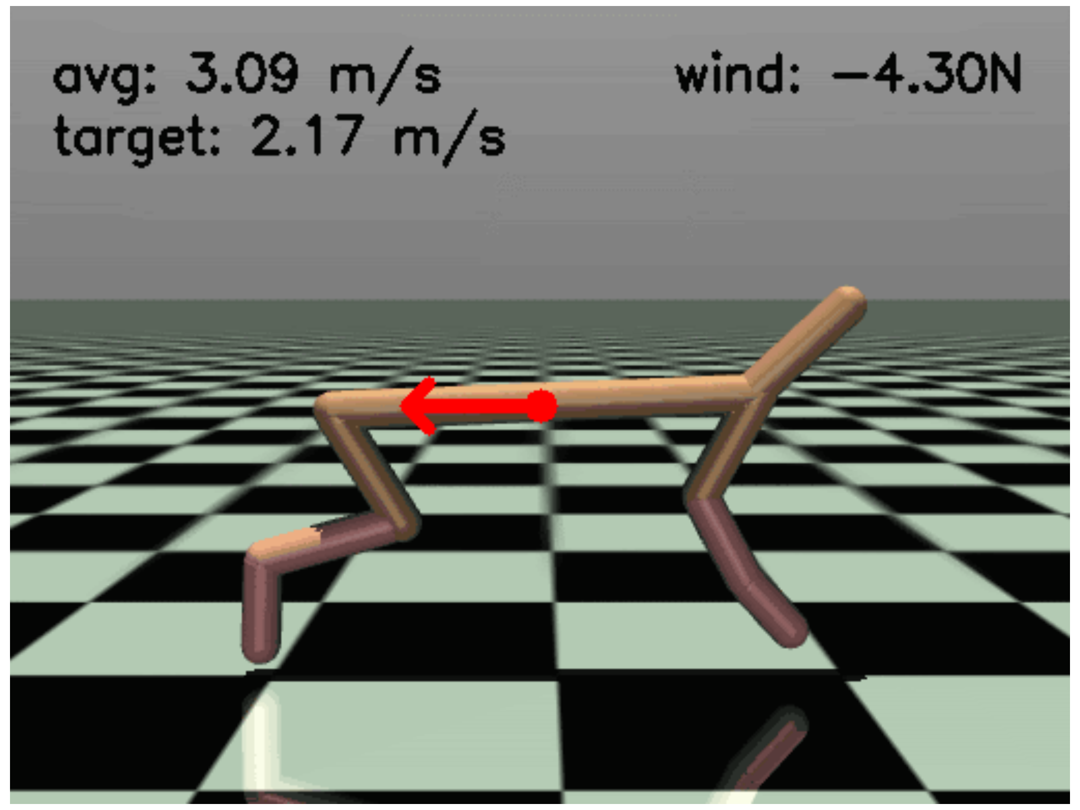
FansRL: Factored Adaptation for Non-Stationary Reinforcement Learning

Fan Feng, Biwei Huang, Kun Zhang, Sara Magliacane

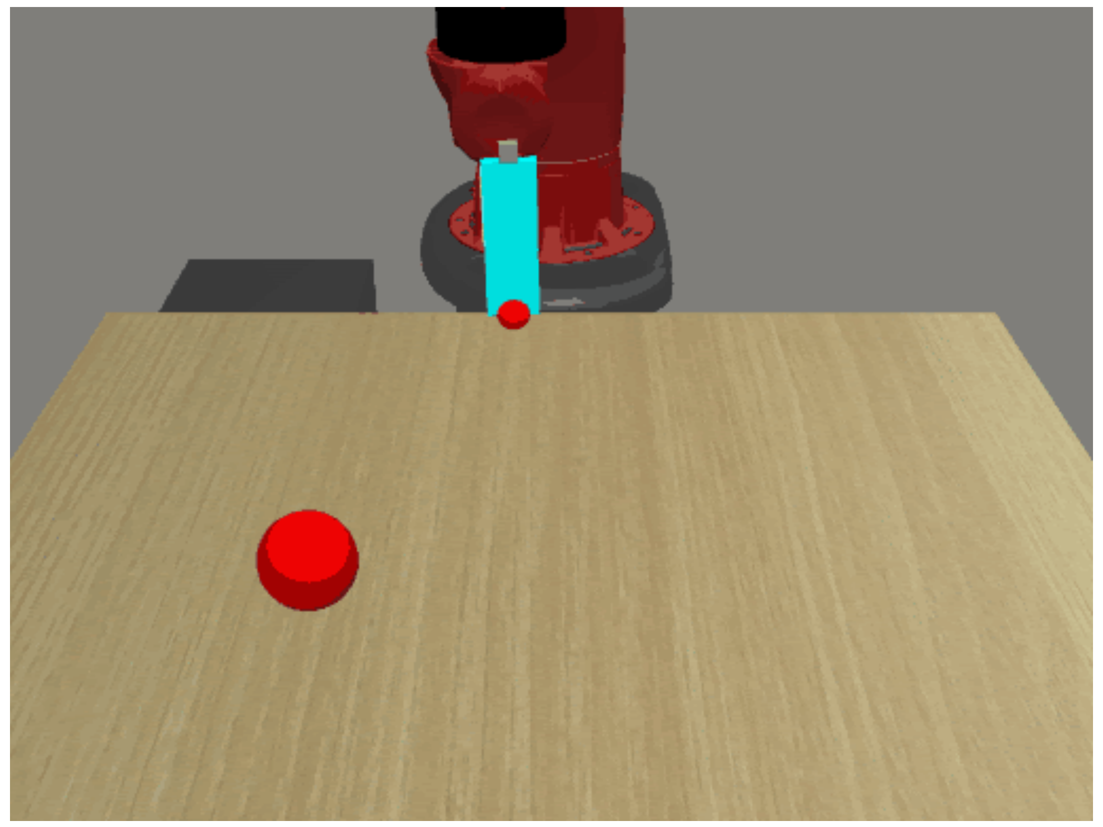
- The **latent change factors** are not constant anymore and they model **non-stationarity**



Change factors follow a Markov process:
- **Discrete/abrupt** changes
- **Continuous/smooth** changes



Non-stationary environments (wind changes)



Non-stationary rewards (target changes)

https://arxiv.org/abs/2203.16582

Conclusions

- D-separation [Pearl 1988] is a principled way to reason about **invariances and distribution shift**
  - Not a new observation, known since [Schoelkopf et al 2012, Zhang et al. 2013]
  - This is true even with:
    - **Unknown causal graph**
    - **Missing data/CI** (so unknown MEC)
- **D-separation logic encodings** [Hyttinen et al 2014] allow us to reason about d-separations even with **missing data,** even without reconstructing MEC

Conclusions

- D-separation [Pearl 1988] is a principled way to reason about **invariances and distribution shift**
  - Not a new observation, known since [Schoelkopf et al 2012, Zhang et al. 2013]
  - This is true even with:
    - **Unknown causal graph**
    - **Missing data/CI** (so unknown MEC)
- **D-separation logic encodings** [Hyttinen et al 2014] allow us to reason about d-separations even with **missing data,** even without reconstructing MEC

Thanks! Questions?

(joint work with Thijs van Ommen, Tom Claassen, Stephan Bongers, Philip Versteeg, Joris Mooij, Biwei Huang, Fan Feng, Chaochao Lu and Kun Zhang)