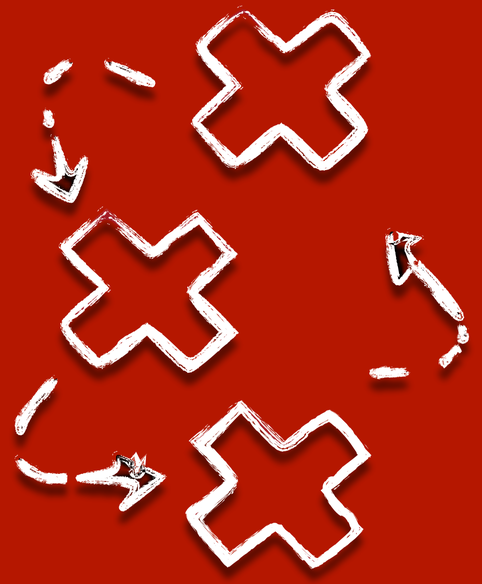


Causal Data Science

Lecture 12.2: Causality-inspired ML

Lecturer: Sara Magliacane

UvA - Spring 2024

Causality + machine learning (non-exhaustive list)

1. **Machine learning (ML) helps causality**
   - Causal discovery - learning causal graphs from data
   - Causal effect estimation - matching, weighting, double ML
   - (Causal) representation learning

2. **Causality (in the most general definition) helps machine learning**
   - Robustness, Transfer learning
   - Reinforcement Learning
   - Bias mitigation, fairness

https://arxiv.org/pdf/1705.08821.pdf, https://arxiv.org/pdf/1802.05664.pdf, https://arxiv.org/pdf/1605.03661.pdf, https://crl.causalai.net/, https://www.youtube.com/watch?v=Obuu3w809CI&ab_channel=ConnorJerzak and many many others

Causality + machine learning (non-exhaustive list)

1. **Machine learning (ML) helps causality**
   - **Causal discovery - learning causal graphs from data**
   - **Causal effect estimation - matching, weighting, double ML**
   - **(Causal) representation learning**

2. **Causality (in the most general definition) helps machine learning**
   - **Robustness, Transfer learning**
   - **Reinforcement Learning**
   - **Bias mitigation, fairness**

https://arxiv.org/pdf/1705.08821.pdf, https://arxiv.org/pdf/1802.05664.pdf, https://arxiv.org/pdf/1605.03661.pdf, https://crl.causalai.net/, https://www.youtube.com/watch?v=Obuu3w809CI&ab_channel=ConnorJerzak and many many others

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?

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?
3. Counterfactuals $P(y_x\|x', y')$	Imagining, Retrospection		

E.g. need many experiments or strong assumptions to identify the causal graph or the causal variables

“Full” causality can be **not necessary** or **too expensive** ->

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?

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

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** $\approx$ change of distribution of P(X| parents)

Causality vs Transfer learning

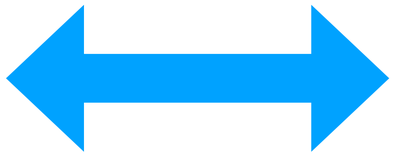
- Transfer learning:
  - How can I predict what ha ens when when the distribution chang an

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


t intervention do(X):

- do-calculus [Pearl, 2009]





- **Soft intervention on X** ≈ change of distribution of P(X| parents)

Unsupervised domain adaptation - toy example

	X1	X2	Y
Wildtype	0,1	2	0
Wildtype	0,2	3	0
Wildtype	1,1	2	1
Wildtype	0,1	3	0

Source domain

No labels in target

	X1	X2	Y
Gene A	3,1	2	?
Gene A	3,2	3	?
Gene A	4	2	?
Gene A	3,2	3	?

Target domain

- **Task:** Learn a model Y = $\hat{f}$(X1, X2) on the source domain so that it can reliably estimate Y in target domain

Domain adaptation from the graphical perspective

C	X1	X2	Y
0	0,1	2	0
0	0,2	3	0
0	1,1	2	1
0	0,1	3	0

Source domain

C	X1	X2	Y
1	3,1	2	?
1	3,2	3	?
1	4	2	?
1	3,2	3	?

Target domain

1. We add a context variable C to distinguish the domains

Domain adaptation from the graphical perspective

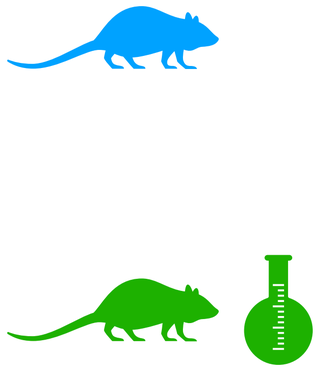
C	X1	X2	Y
0	0,1	2	0
0	0,2	3	0
0	1,1	2	1
0	0,1	3	0
1	3,1	2	?
1	3,2	3	?
1	4	2	?
1	3,2	3	?

Source domain

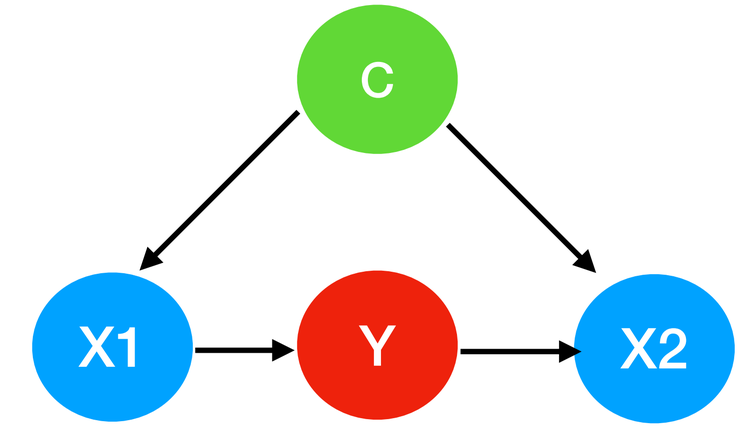
Target domain

1. We add a context variable C to distinguish the domains
2. We consider the data as coming from a single distribution P(X1, X2,Y, C)

Domain adaptation from the graphical perspective



C	X1	X2	Y
0	0,1	2	0
0	0,2	3	0
0	1,1	2	1
0	0,1	3	0
1	3,1	2	?
1	3,2	3	?
1	4	2	?
1	3,2	3	?

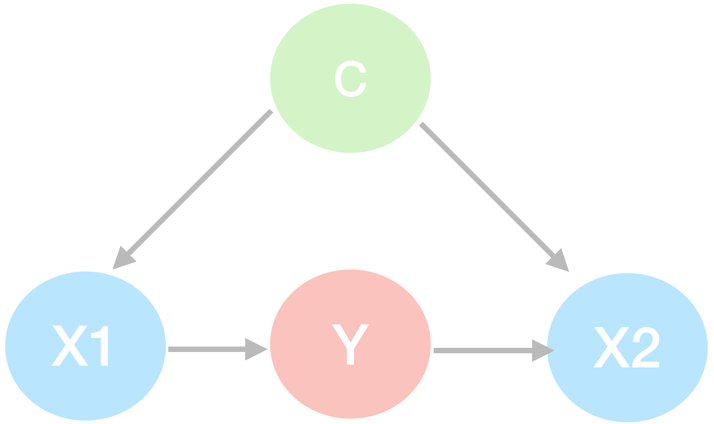


We can represent P(X1, X2,Y, C) with an **(unknown)** causal graph

1. We add a context variable C to distinguish the domains
2. We consider the data as coming from a single distribution P(X1, X2,Y, C)

Separating features = safe for (adversarial) domain adaptation

- **Separating features:** sets of features that d-separate Y from the context

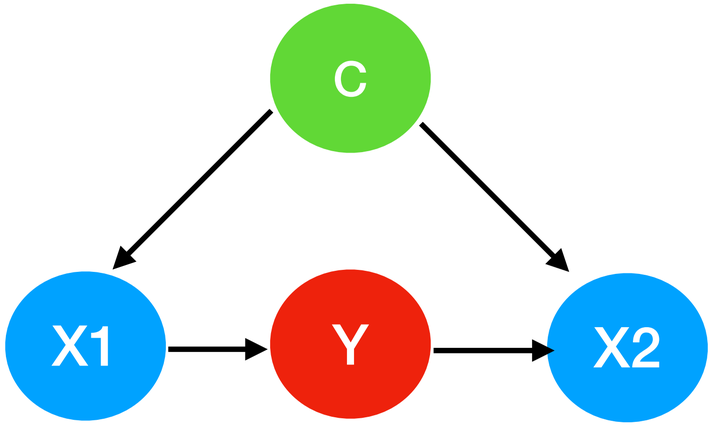


Step 1: Known causal graph

Separating features = safe for (adversarial) domain adaptation

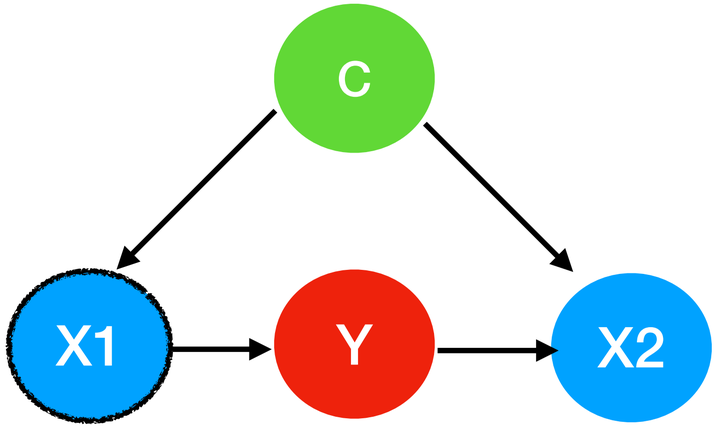
Aka stable features, invariant features etc.

- **Separating features:** sets of features that d-separate Y from the context



Separating features = safe for (adversarial) domain adaptation

- **Separating features:** sets of features that d-separate Y from the context

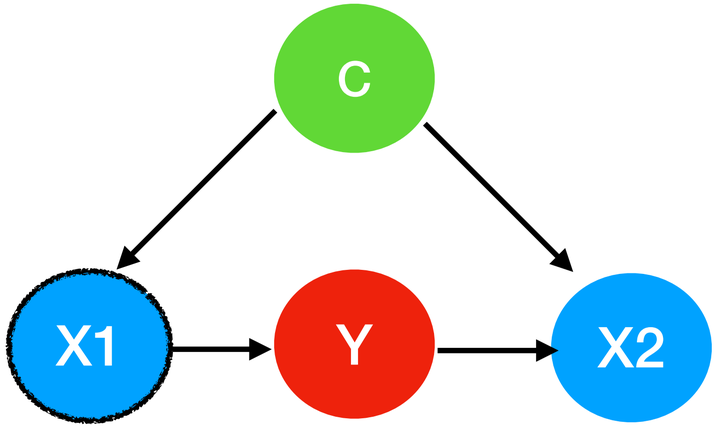


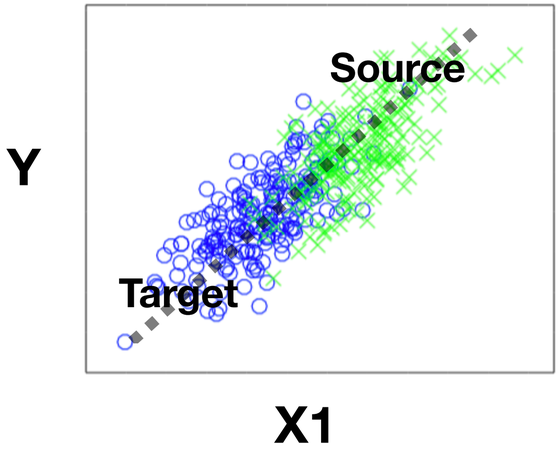
$$Y \perp\!\!\!\perp_d C \mid X_1 \iff Y \perp\!\!\!\perp C \mid X_1$$

(under Markov and faithfulness assumptions)

Separating features = safe for (adversarial) domain adaptation

- **Separating features:** sets of features that d-separate Y from the context





$$Y \perp\!\!\!\perp_d C \,|\, X_1 \iff Y \perp\!\!\!\perp C \,|\, X_1$$

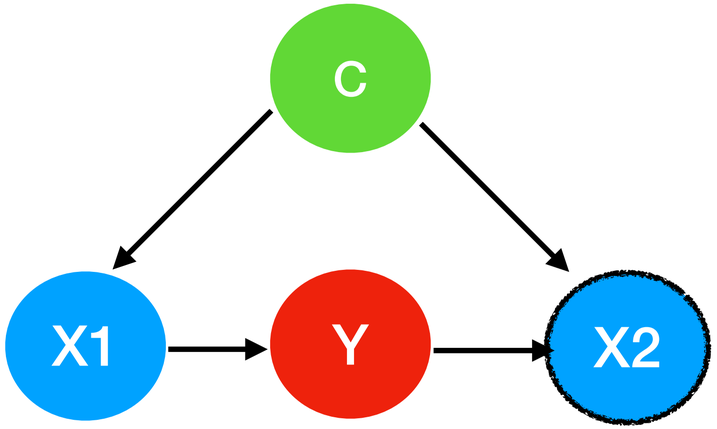
(under Markov and faithfulness assumptions)

$$Y \perp\!\!\!\perp C \,|\, X_1 \equiv$$

$$P(Y \,|\, X_1, C = 0) = P(Y \,|\, X_1, C = 1)$$

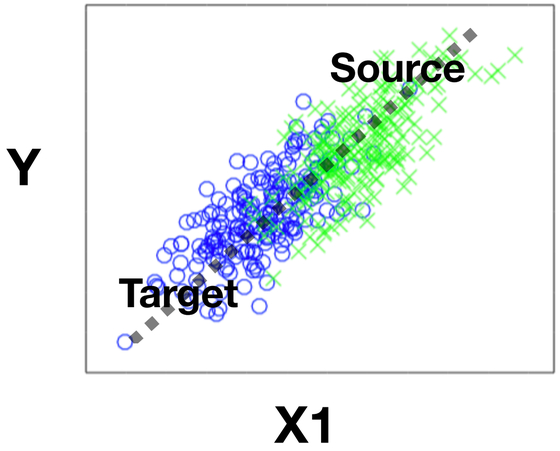
Separating features = safe for (adversarial) domain adaptation

- **Separating features:** sets of features that d-separate Y from the context



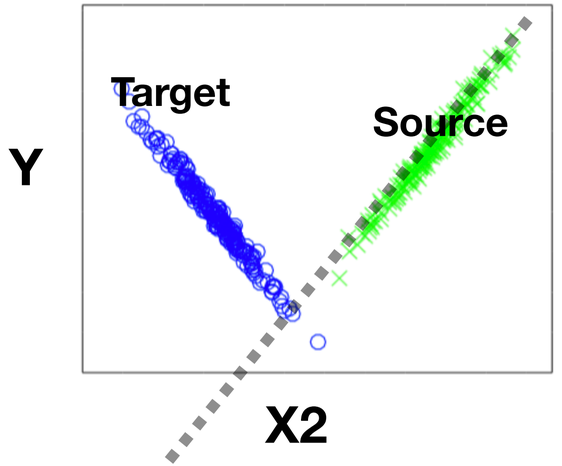
$$Y \not\perp\!\!\!\perp_d C \mid X_2 \iff Y \not\perp\!\!\!\perp C \mid X_2$$

(under Markov and faithfulness assumptions)



$$Y \perp\!\!\!\perp C \mid X_1 \equiv$$

$$P(Y \mid X_1, C = 0) = P(Y \mid X_1, C = 1)$$

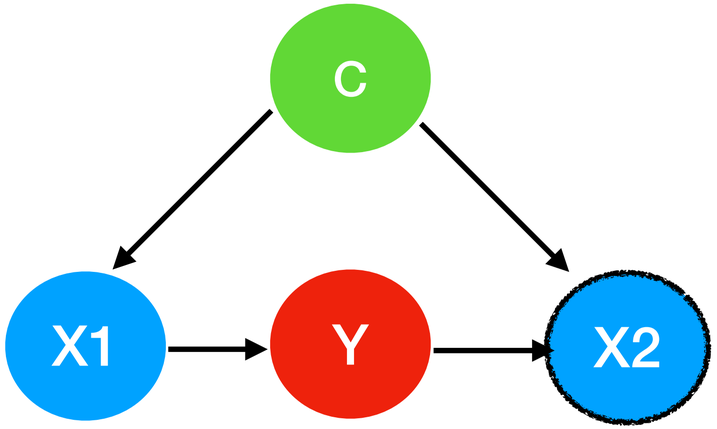


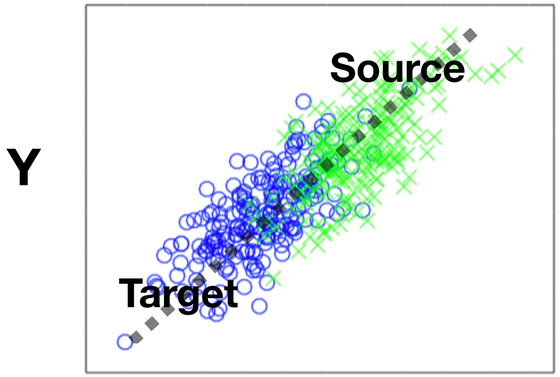
$$Y \not\perp\!\!\!\perp C \mid X_2 \equiv$$

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

Separating features = safe for (adversarial) domain adaptation

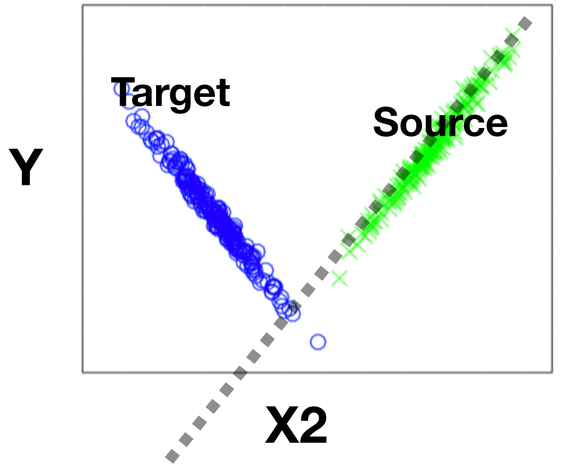
- **Separating features:** sets of features that d-separate Y from the context





$Y \perp\!\!\!\perp C \mid X_1 \equiv$

$P(Y \mid X_1, C = 0) = P(Y \mid X_1, C = 1)$



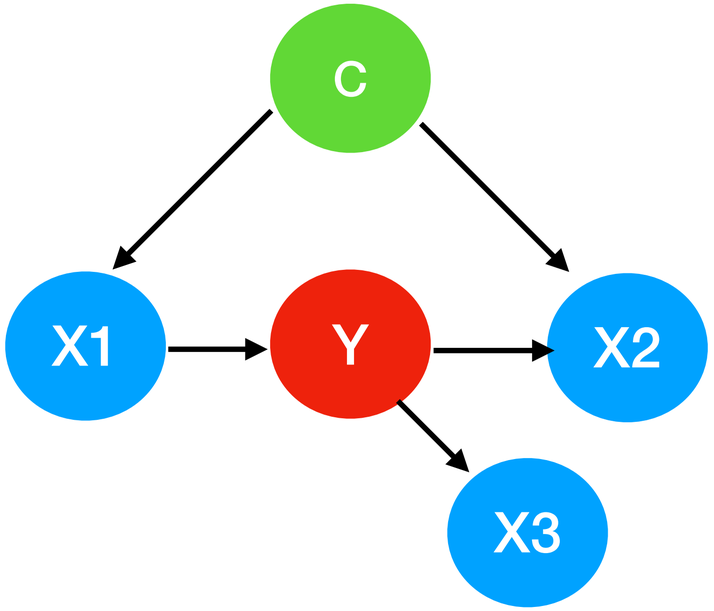
$Y \not\perp\!\!\!\perp C \mid X_2 \equiv$

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

{X1} is a separating feature, {X2} and {X1, X2} are not -> **arbitrarily large error**

Separating features = safe for (adversarial) domain adaptation

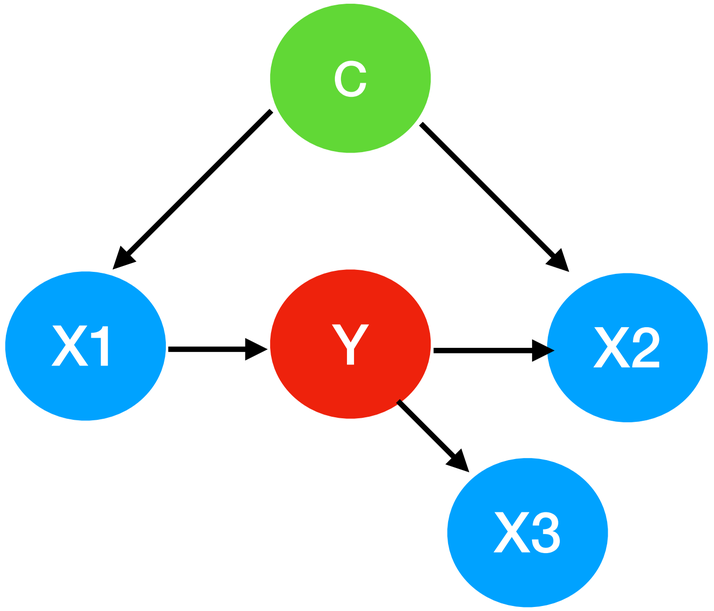
- **Separating features:** sets of features that d-separate Y from the context



$$Y \perp_d C \mid \{X_1, X_3\} \; ?$$

Separating features = safe for (adversarial) domain adaptation

- **Separating features:** sets of features that d-separate Y from the context

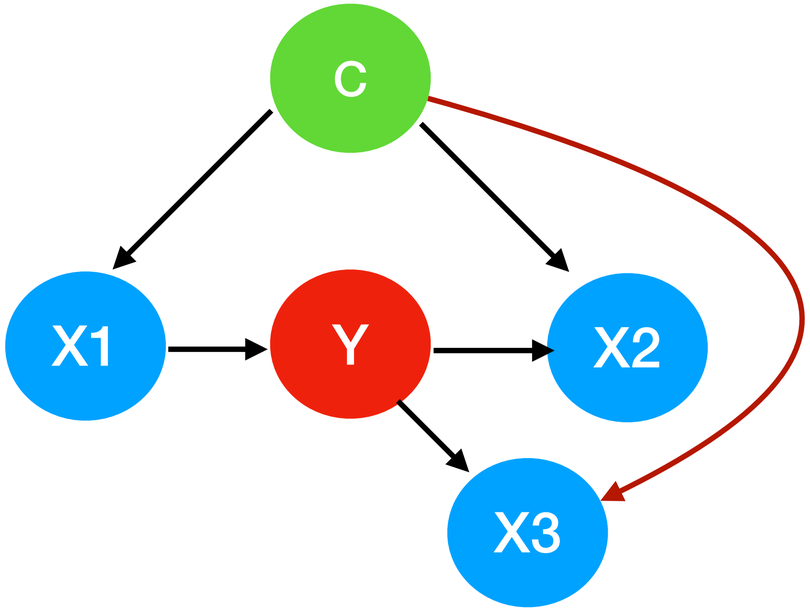


$$Y \perp_d C \mid \{X_1, X_3\} \; ?$$

$Y \perp_d C \mid \{X_1, X_3\}$ (separating features are not necessarily parents of Y)

Separating features = safe for (adversarial) domain adaptation

- **Separating features:** sets of features that d-separate Y from the context

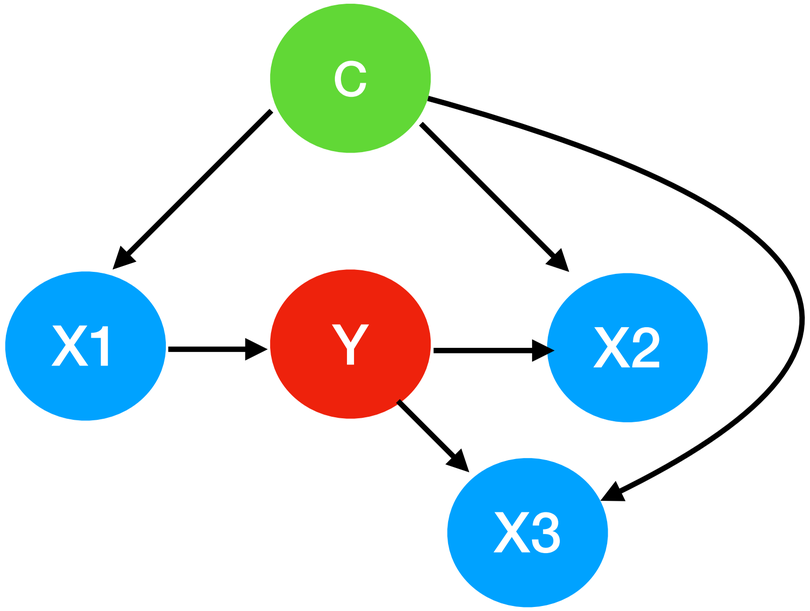


Intervention on every variable except Y = domain generalisation

$$Y \perp_d C \mid \{X_1, X_3\} \; ?$$

Separating features = safe for (adversarial) domain adaptation

- **Separating features:** sets of features that d-separate Y from the context

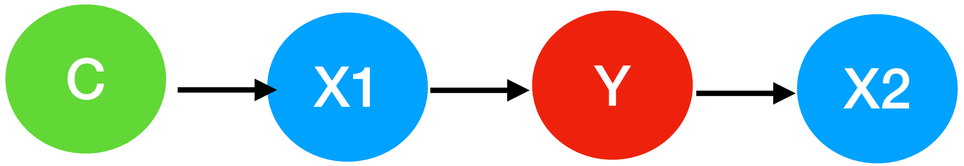


Intervention on every variable except Y = domain generalisation

$$Y \perp_d C \mid \{X_1, X_3\} \; ?$$

Under every possible distribution shift (except directly on the label Y), the separating set of feature are the **causal parents**

Misconceptions 1: an invariant feature need not be causal

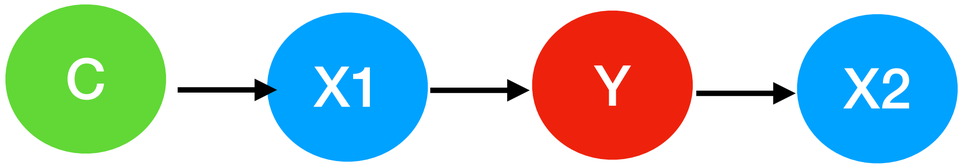


$$Y \perp\!\!\!\perp C \mid X_1$$

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

- $P(Y \mid X_1, X_2)$ is invariant $\Longrightarrow$ invariant features are not necessarily parents of Y

Misconceptions 1: an invariant feature need not be causal



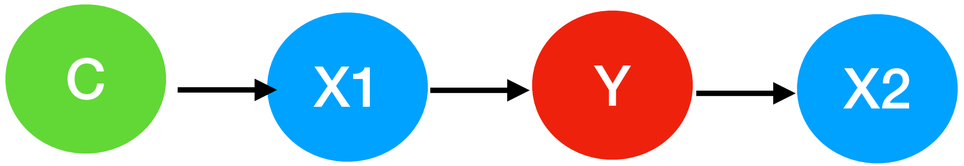
$$Y \perp\!\!\!\perp C \mid X_1$$

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

- $P(Y \mid X_1, X_2)$ is invariant $\Longrightarrow$ 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

Misconceptions 1: an invariant feature need not be causal



$$Y \perp\!\!\!\perp C \mid X_1$$

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

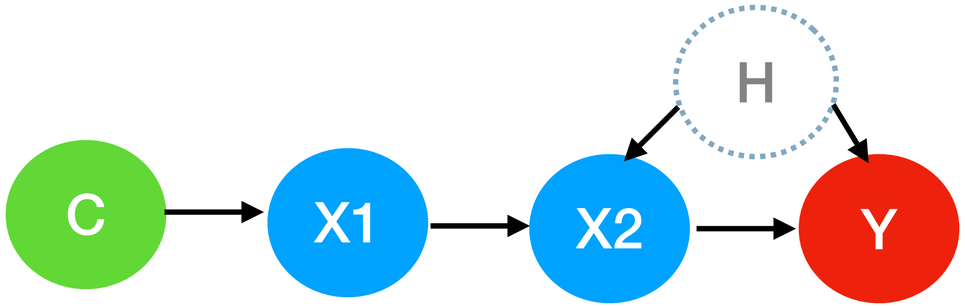
- $P(Y \mid X_1, X_2)$ is invariant $\Longrightarrow$ invariant features are not necessarily parents of Y

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

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

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

Misconceptions 2: observed parents are not enough under latent confounding



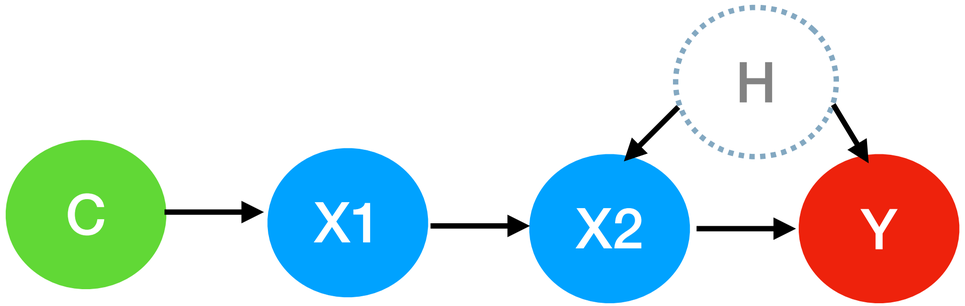
$$Y \perp\!\!\!\perp C \mid X_1$$

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

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

- $P(Y \mid X_1)$ is invariant, $P(Y \mid X_2)$ is not

Misconceptions 2: observed parents are not enough under latent confounding



$$Y \perp\!\!\!\perp C \mid X_1$$

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

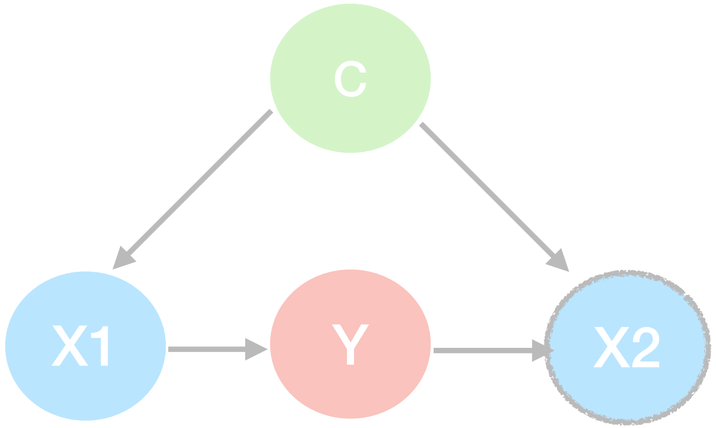
$$Y \perp\!\!\!\perp C \mid X_1, X_2$$

- $P(Y \mid X_1)$ is invariant, $P(Y \mid X_2)$ is not

- **Conclusion:** causality (e.g. using the causal parents, learning the complete causal graph) is **neither necessary or sufficient*** for transfer, what we care about are **conditional independences/d-separations**

Separating features = safe for (adversarial) domain adaptation

- **Separating features:** sets of features that d-separate Y from the context

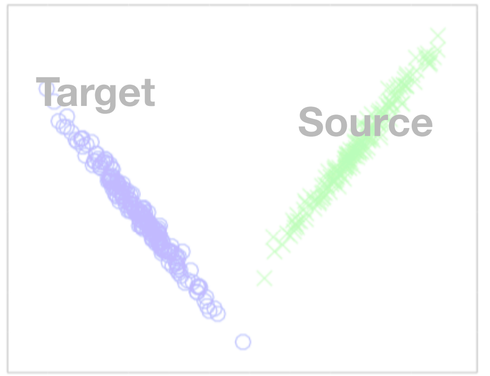


$$Y \not\perp\!\!\!\perp_d C \mid X_2$$

Step 2: **Unknown graph and no labels in target!**

$$Y \perp\!\!\!\perp C \mid X_1 \equiv$$

$$P(Y \mid X_1, C = 0) = P(Y \mid X_1, C = 1)$$



$$Y \not\perp\!\!\!\perp C \mid X_2 \equiv$$

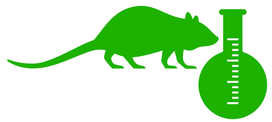
$$P(Y \mid X_2, C = 0) \neq P(Y \mid X_2, C = 1)$$

{X1} is a separating feature, {X2} and {X1, X2} are not -> **arbitrarily large error**

Desiderata for a causal domain adaptation method

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

Unsupervised multi-source domain adaptation - toy example



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

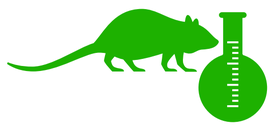
Source domains

Target domain

https://arxiv.org/abs/1707.06422

Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions NeurIPS 2018

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



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	?

Source domains

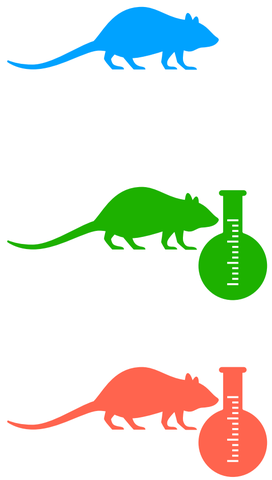
Target domain

- We assume we can model all the domains in with a **single unknown acyclic causal graph** with **multiple context variables** [Mooij et al. 2020]

https://arxiv.org/abs/1707.06422

Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions NeurIPS 2018

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



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	?

Step 1: Known causal graph

- **Find features** $\mathbf{S} \subseteq \mathbf{X} : Y \perp_d C_1 \,|\, \mathbf{S}$

- We assume we can model all the domains in with a **single unknown acyclic causal graph** with **multiple context variables** [Mooij et al. 2020]

https://arxiv.org/abs/1707.06422

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

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

- **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
0	1	0,2	0	0
0	1	0,3	0	1
0	1	0,3	1	0
1	0	3,1	2	?
1	0	3,2	3	?
1	0	4	3	?

What if the causal graph is unknown?

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

$$Y \perp\!\!\!\perp C_1 \mid X_1 ? \qquad Y \perp\!\!\!\perp C_1 \mid X_2 ?$$

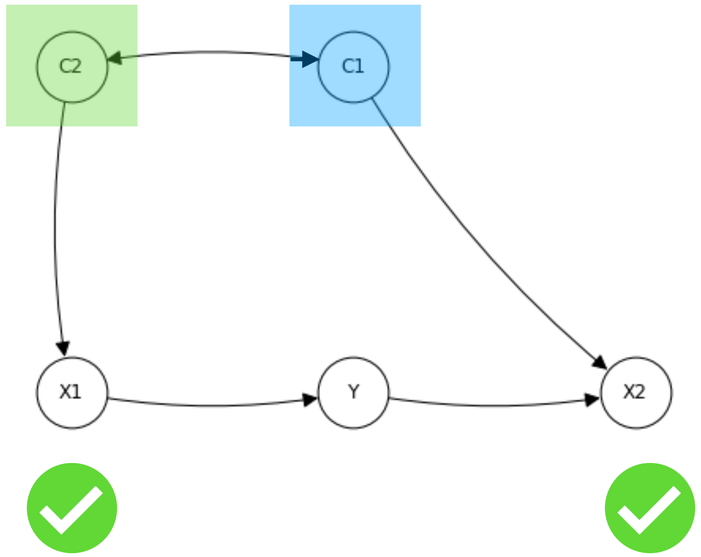
- **Problem:** Y is always missing when C1=1, so we cannot test these

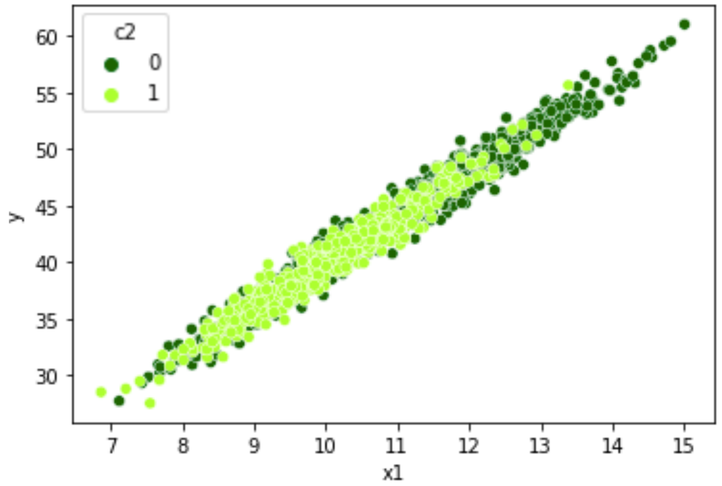
C1	C2	X1	X2	Y
0	0	0,1	1	0
0	0	0,2	1	0
0	0	1,1	2	1
0	1	0,2	0	0
0	1	0,3	0	1
0	1	0,3	1	0

Idea: Invariant features in source domains are also separating in the target domain

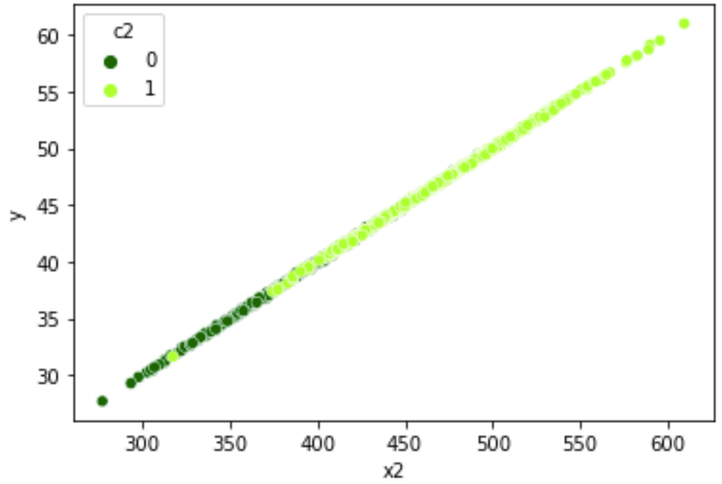
$$Y \perp\!\!\!\perp C_2 \mid \{X_1, C_1 = 0\} \implies Y \perp\!\!\!\perp C_1 \mid X_1$$

Separating features in sources are also separating in target - counterexample



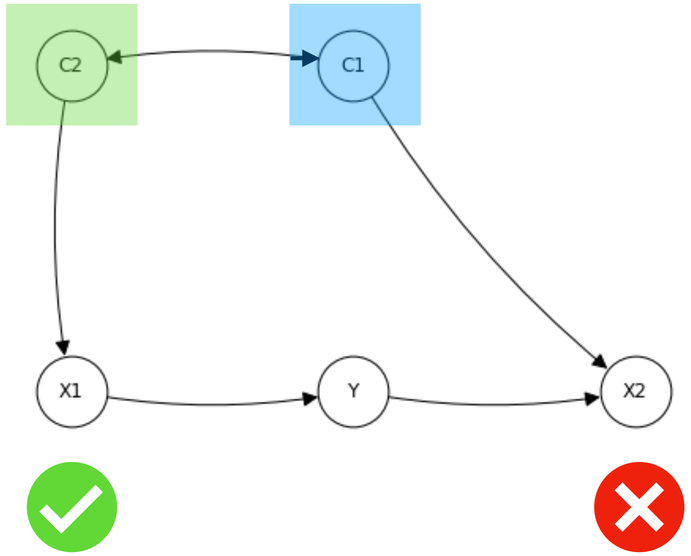


$$Y \perp\!\!\!\perp C_2 \mid \{X_1, C_1 = 0\}$$

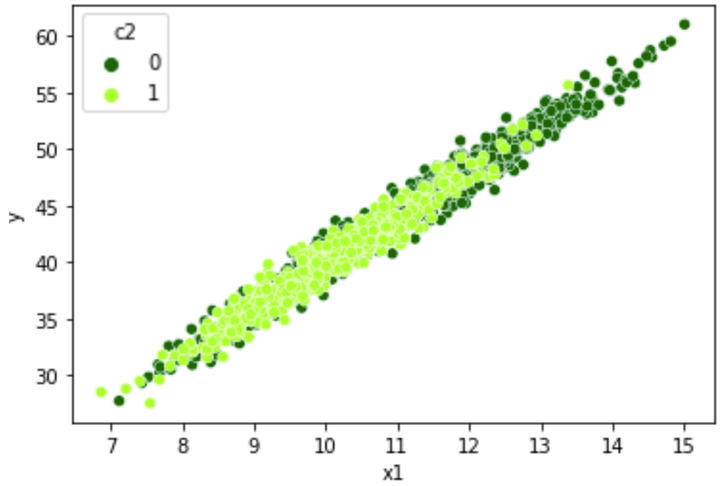


$$Y \perp\!\!\!\perp C_2 \mid \{X_2, C_1 = 0\}$$

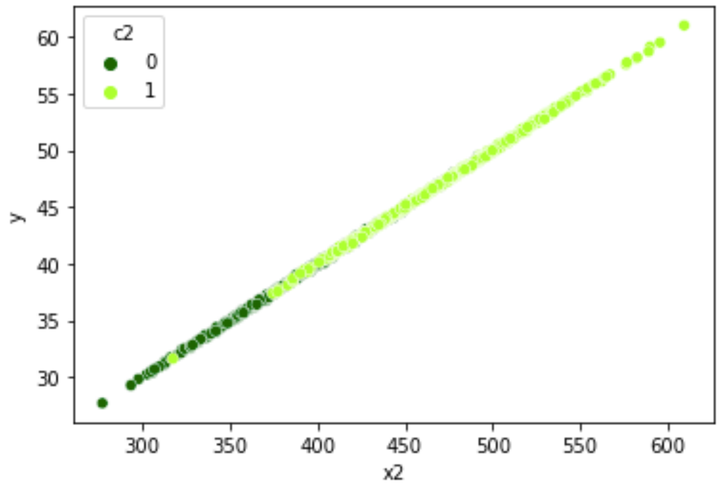
Separating features in sources are also separating in target - counterexample

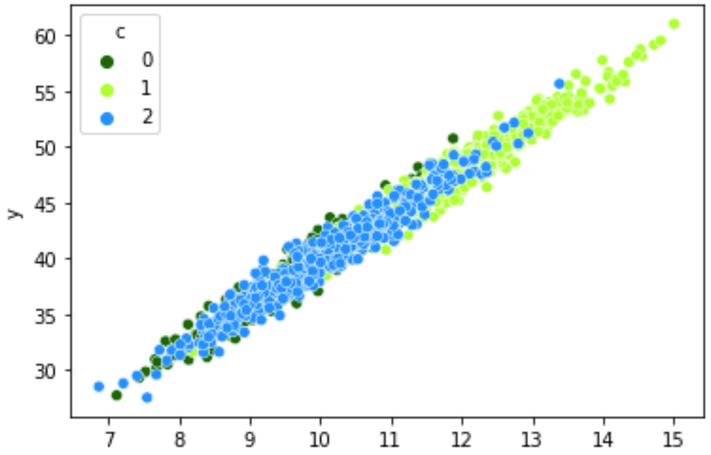


$Y \perp\!\!\!\perp C_2 \mid \{X_1, C_1 = 0\}$

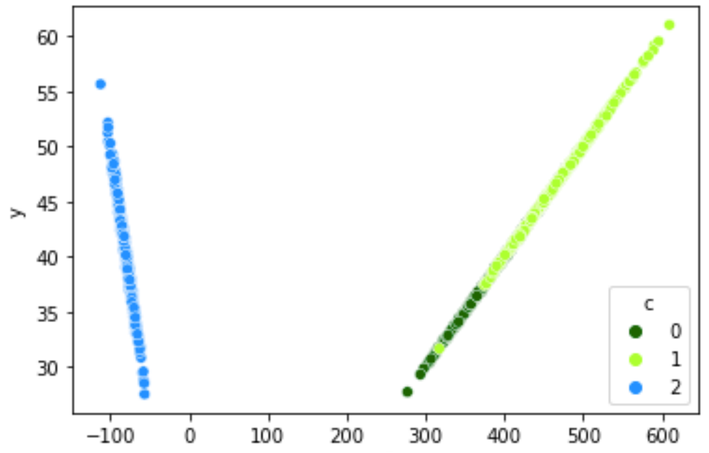


$Y \perp\!\!\!\perp C_2 \mid \{X_2, C_1 = 0\}$





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



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

What if the causal graph is unknown?

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

$$Y \perp\!\!\!\perp C_1 \mid X_1 ? \qquad Y \perp\!\!\!\perp C_1 \mid X_2 ?$$

- **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
0	1	0,2	0	0
0	1	0,3	0	1
0	1	0,3	1	0

Idea: Invariant features in source domains are also separating in the target domain

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

This is a strong assumption

What if the causal graph is unknown?

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

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

- **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
0	1	0,2	0	0
0	1	0,3	0	1
0	1	0,3	1	0
1	0	3,1	2	?
1	0	3,2	3	?
1	0	4	3	?

$$X_1 \not\perp\!\!\!\perp X_2$$
$$X_1 \not\perp\!\!\!\perp C_1$$
$$X_1 \not\perp\!\!\!\perp X_2 | C_1$$
$$X_1 \perp\!\!\!\perp X_2 | Y, C_1 = 0$$
$$\dots$$

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

Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions NeurIPS 2018

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

- We search for **separating features** that d-separate Y from C_1 (target)
- We assume **no extra dependences involving Y** in target domain C1=1

Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions NeurIPS 2018

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

- We search for **separating features** that d-separate Y from C_1 (target)
- 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

https://arxiv.org/abs/1707.06422

Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions NeurIPS 2018

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

- We search for **separating features** that d-separate Y from C_1 (target)
- We assume **no extra dependences involving Y** in target domain C1=1

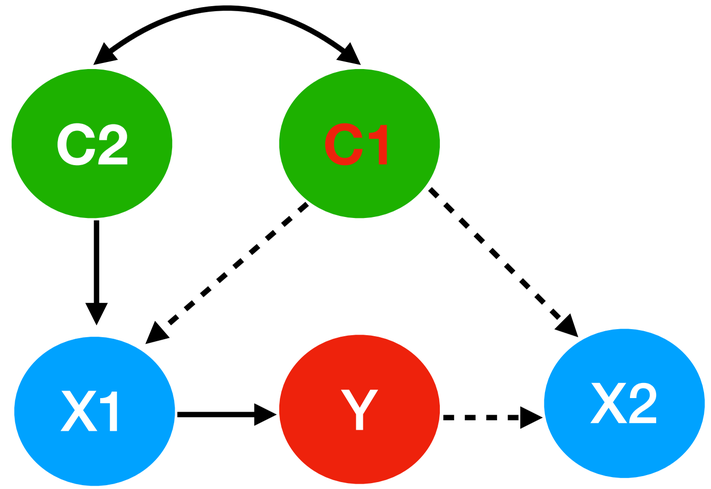
C1	C2	X1	X2	Y
0	0	0,1	1	0
0	0	0,2	1	0
0	0	1,1	2	1
0	1	3,1	2	1
0	1	3,2	3	1
0	1	4	3	1
1	0	0,2	0	?
1	0	0,3	0	?
1	0	0,3	1	?

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

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

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

Perform allowed CI tests

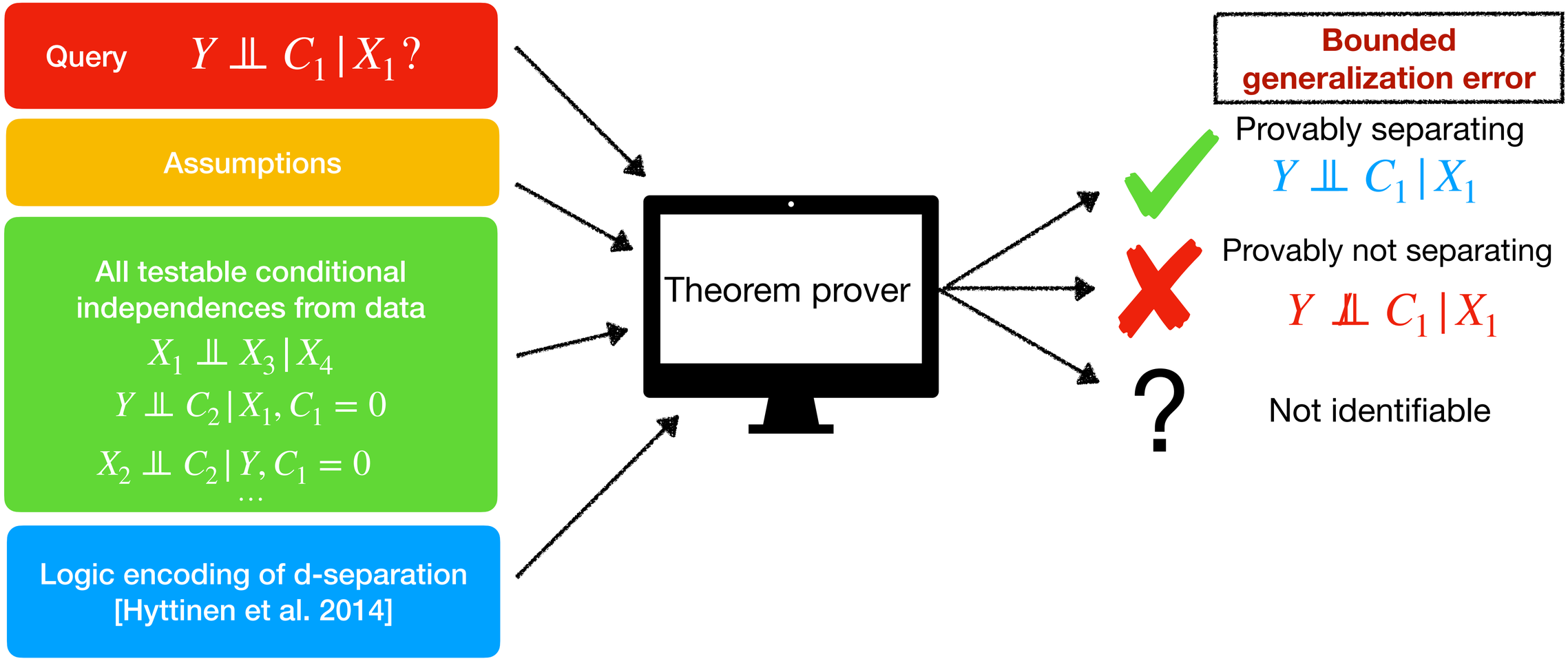


All possible compatible graphs

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

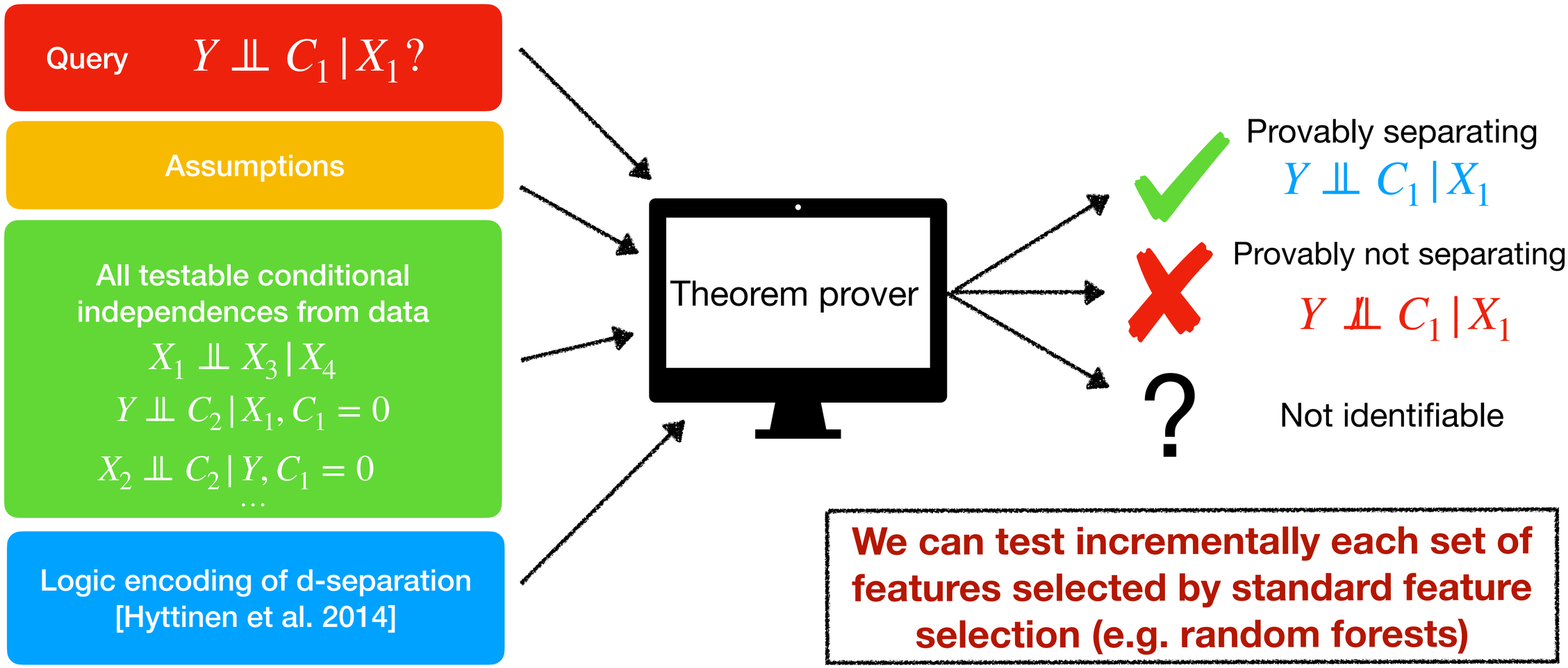
https://arxiv.org/abs/1707.06422

Inferring separating sets of features



https://arxiv.org/abs/1707.06422

Inferring separating sets of features



We can test incrementally each set of features selected by standard feature selection (e.g. random forests)

https://arxiv.org/abs/1707.06422

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

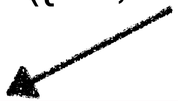
Source domains data

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

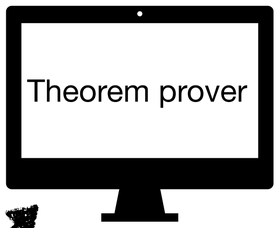
$X_1 \perp\!\!\!\perp X_3 \mid X_4$

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

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

…

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

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

Not identifiable

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, X2, 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 $Y \perp\!\!\!\perp C_1 \mid S$

Not identifiable

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, X2, 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

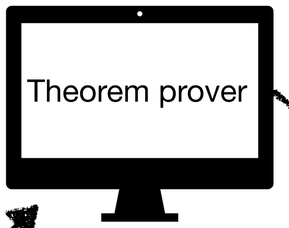
$X_1 \perp\!\!\!\perp X_3 \mid X_4$

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

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

…

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



Provably separating

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

Learn $\hat{f}$(S) on source domains

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

Iterate until empty

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

Provably separating

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

Learn $\hat{f}$(S) on source domains

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

Bounded generalisation error

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 s…

$Y \perp\!\!\!\perp C_$

Not identifiable

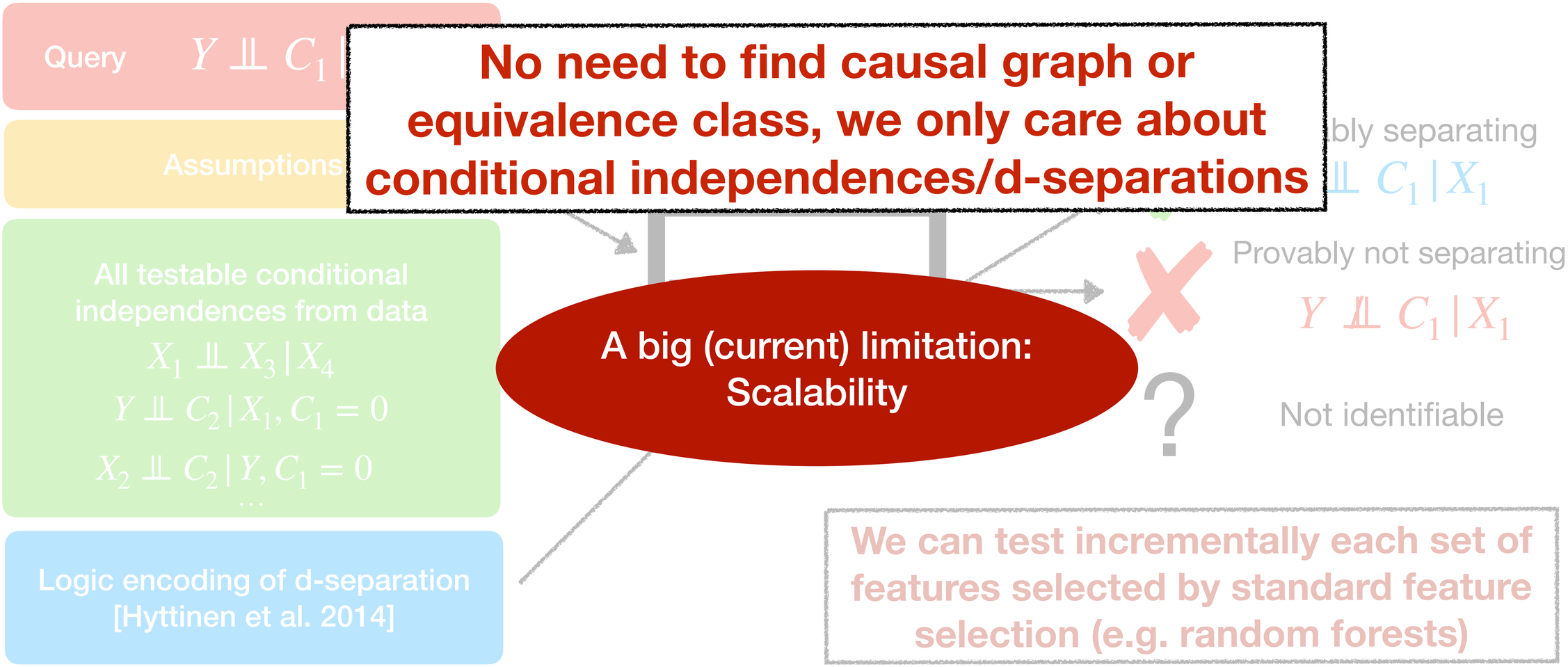
Provably separating

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

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

Learn $\hat{f}$(S) on source domains

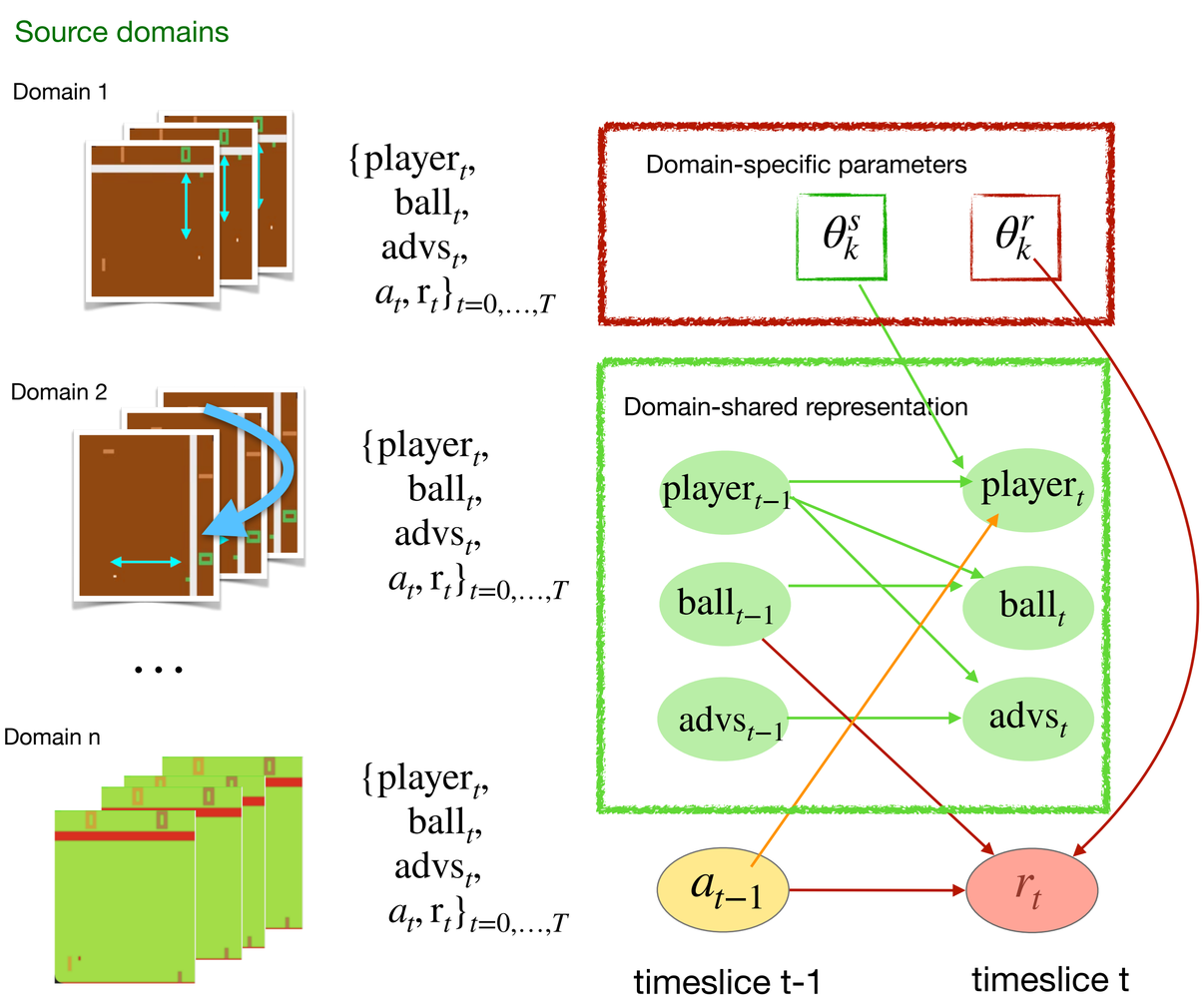
Inferring separating sets of features



https://arxiv.org/abs/1707.06422

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

Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, Kun Zhang ICLR 2022

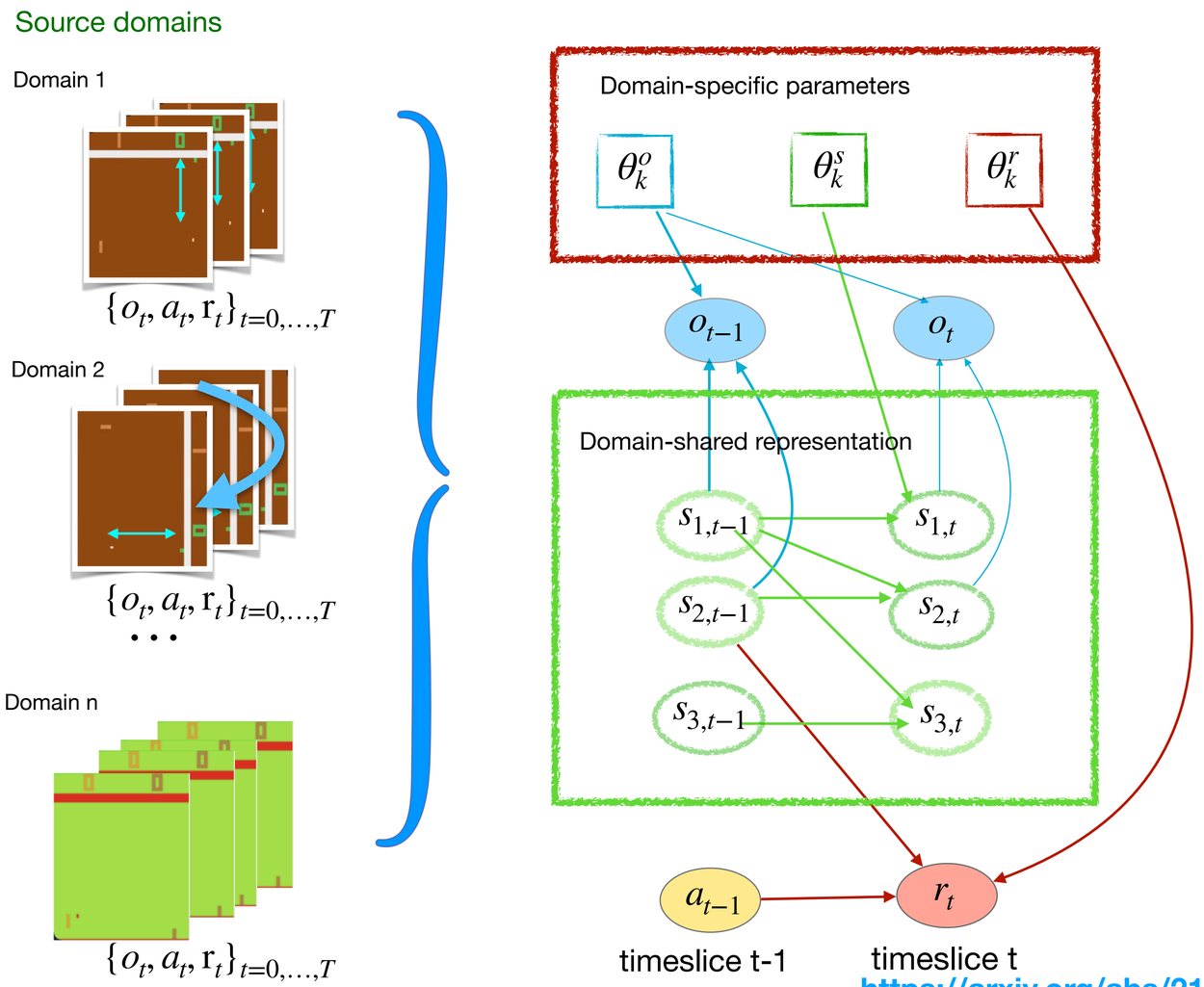


When we learn from symbolic inputs, the causal graph can be identified, but we don't have guarantees on what the latent change factors are

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 ICLR 2022



When we learn from images, we cannot identify the causal variables, so what we learn is not necessarily causal… but it is still useful

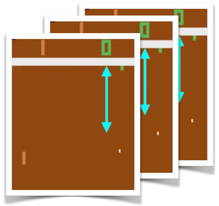
https://arxiv.org/abs/2107.02729

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

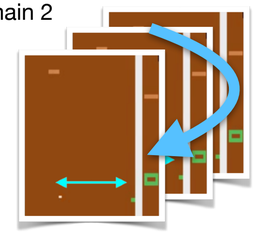
Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, Kun Zhang ICLR 2022

Source domains

Domain 1



Domain 2



...

Estimate graph over estimated s_k, θ_k

Identify s_t^{min}, θ_t^{min} from the estimated graph

Learn optimal policy $\pi^*(s_k^{min}, \theta_k^{min})$ on source domains

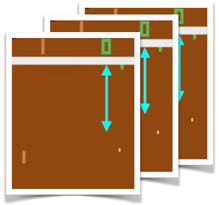
https://arxiv.org/abs/2107.02729

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

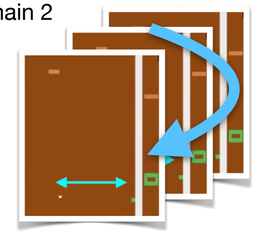
Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, Kun Zhang ICLR 2022

Source domains

Domain 1



Domain 2



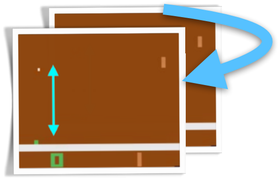
...

Estimate graph over estimated s_k, θ_k

Identify s_t^{min}, θ_t^{min} from the estimated graph

Learn optimal policy $\pi^*(s_k^{min}, \theta_k^{min})$ on source domains

Target domain



$\{o_t, a_t, r_t\}_{t=0,\ldots,T}$

Use model to estimate $s_{target}^{min}, \theta_{target}^{min}$ with few samples

Apply policy $\pi^*(s_{target}^{min}, \theta_{target}^{min})$

Simplifying assumption: no new edges in target domain

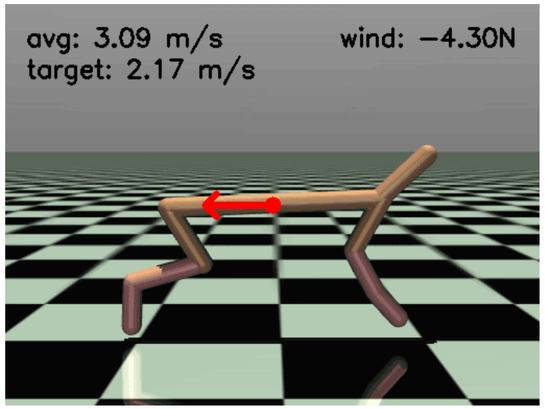
https://arxiv.org/abs/2107.02729

FansRL: Factored Adaptation for Non-Stationary Reinforcement Learning

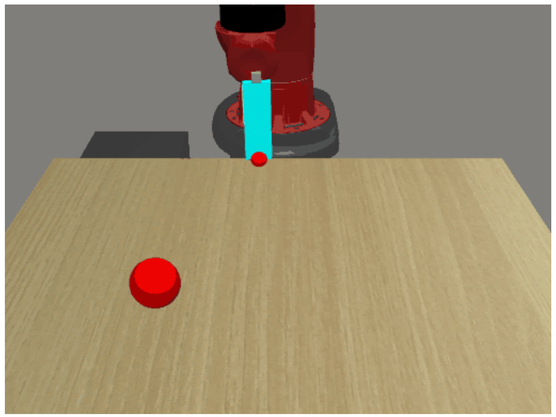
Fan Feng, Biwei Huang, Kun Zhang, Sara Magliacane

NeurIPS 2022

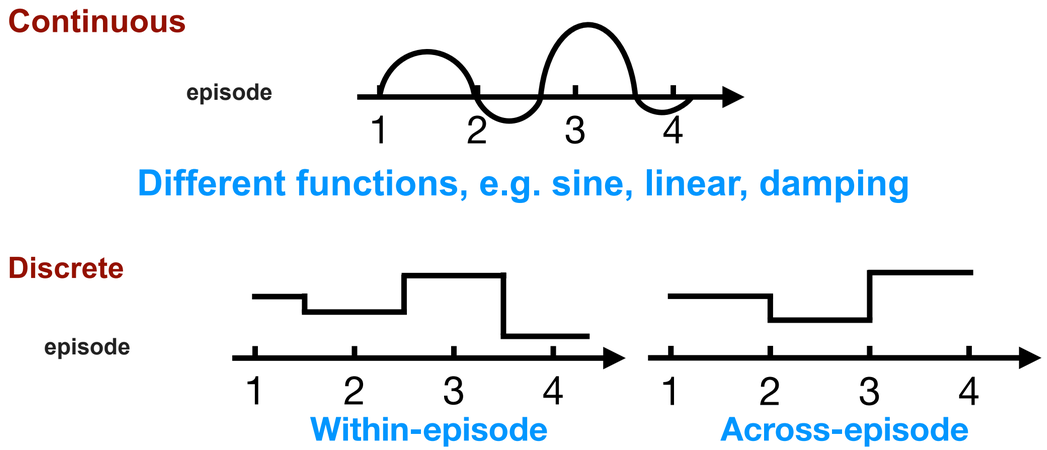
- **Task:** RL agent has to learn a policy that is robust to different types of non-stationarity, including **multiple simultaneous changes of different types**



Non-stationary environments
(wind changes)



Non-stationary rewards
(target changes)



https://arxiv.org/abs/2203.16582

Conclusions

- Graphical models and d-separation [Pearl 1988] are a principled way to reason about **invariances and distribution shift**
  - Not a new observation, known since [Schoelkopf et al 2012]
  - Even with **unknown causal graphs, Missing data/zero-shot settings**
- Often we **do not need to reconstruct the causal graph**, we only need to infer missing conditional independences
- These ideas seem empirically useful even if we **cannot guarantee** that we are **learning the true causal variables or the true causal graph**