Partial Transportability for Domain Generalization

Kasra Jalaldoust

Alexis Bellot

Elias Bareinboim

Columbia University
Computer Science

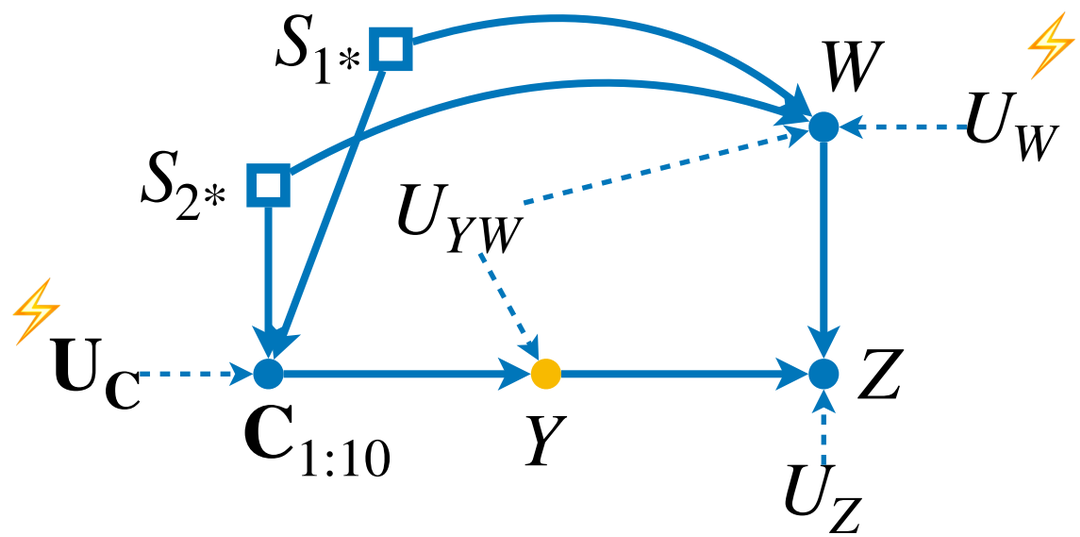
Outline

- Non-transportable queries.
    - Bounding by canonical SCMs
    - Bounding by neural parametrization.
- In search for best worst-case classifier;
    - Causal Robust Optimization Algorithm

An example

Source SCMs. $\mathscr{M}^1, \mathscr{M}^2$

Target SCMs. $\mathscr{M}^*$

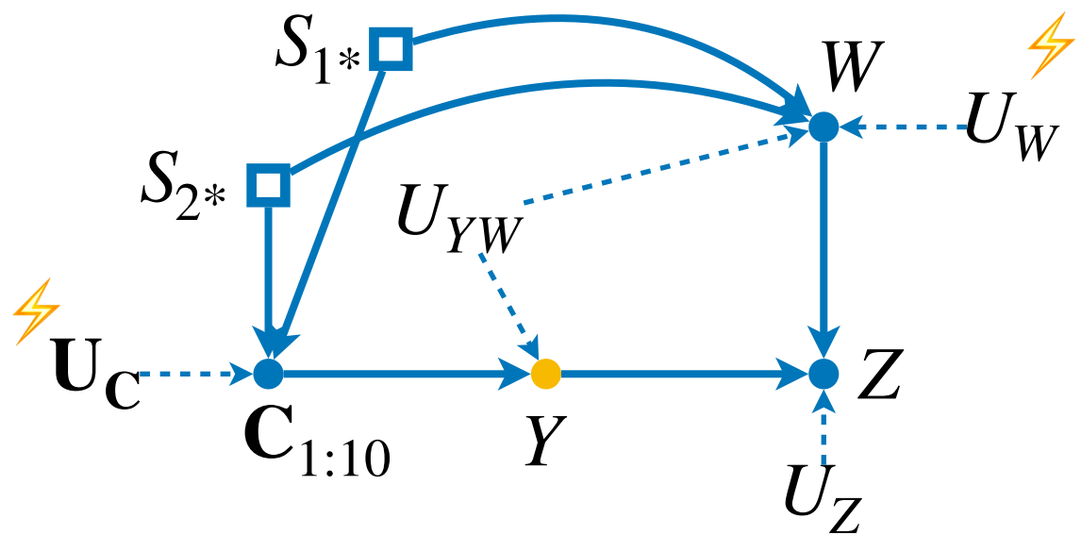


$$P^{1,2,*}(\mathbf{U}): \begin{cases} U_{C_1}, U_{C_2}, \ldots, U_{C_{10}} \sim \begin{cases} \text{Bern}(0.1) & \text{in } \mathscr{M}^1 \\ \text{Bern}(0.5) & \text{in } \mathscr{M}^2 \\ \text{Bern}(0.7) & \text{in } \mathscr{M}^* \end{cases} \\ U_W \sim \begin{cases} \text{Bern}(0.01) & \text{in } \mathscr{M}^1 \\ \text{Bern}(0.02) & \text{in } \mathscr{M}^2 \\ \text{Bern}(0.5) & \text{in } \mathscr{M}^* \end{cases} \\ U_{YW} \sim \text{Bern}(0.2) \\ U_Z \sim \text{Bern}(0.9) \end{cases}$$

$$\mathscr{F}^{1,2,*}: \begin{cases} W \leftarrow U_{YW} \oplus U_W \\ C_j \leftarrow U_{C_j} \\ Y \leftarrow U_{YW} \oplus \bigoplus_{j=1}^{10} C_j \\ Z \leftarrow Y \cdot U_Z + W \cdot (1 - U_Z) \end{cases}$$

Applying TR results

$$\mathscr{F}^{1,2,*}: \begin{cases} W \leftarrow U_{YW} \oplus U_W \\ C_j \leftarrow U_{C_j} \\ Y \leftarrow U_{YW} \oplus \bigoplus_{j=1}^{10} C_j \\ Z \leftarrow Y \cdot U_Z + W \cdot (1 - U_Z) \end{cases}$$



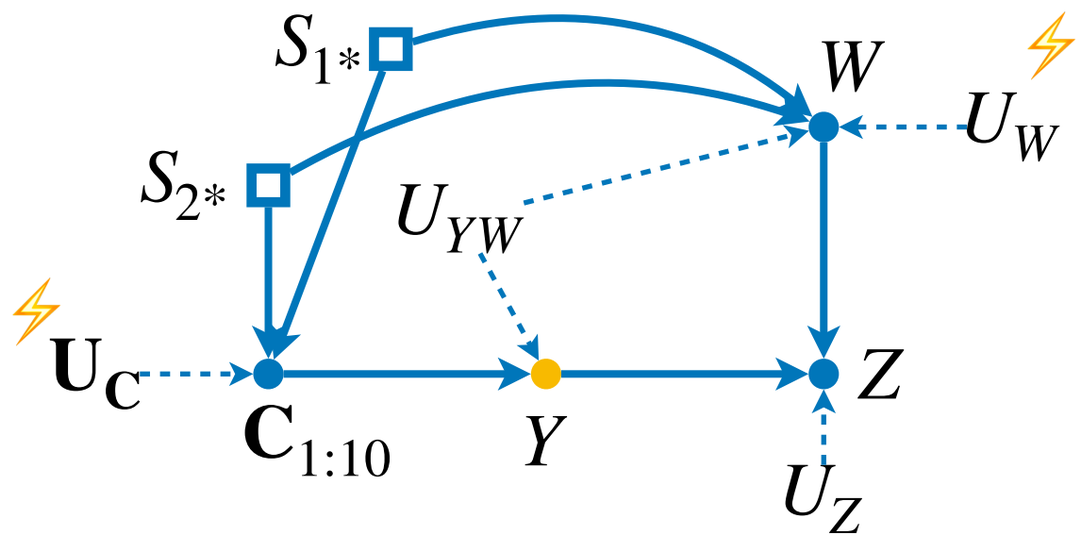
Maximal TR representation (feature): $P^*(y \mid \mathbf{c}) = P^{1,2}(y \mid \mathbf{c})$

Remember $U_{YW} \sim \text{Bern}(0.2)$ ⟶ $P^{1,2,*}(Y = \bigoplus_{j=1}^{10} C_j) \geq 0.8$

⟶ For $h_1(\mathbf{c}) = \bigoplus_{j=1}^{10} C_j$, we would have $\mathscr{R}_{P^{1,2,*}}(h_1) \leq 0.2$.

A common pitfall

$$\mathscr{F}^{1,2,*}: \begin{cases} W \leftarrow U_{YW} \oplus U_W \\ C_j \leftarrow U_{C_j} \\ Y \leftarrow U_{YW} \oplus \bigoplus_{j=1}^{10} C_j \\ Z \leftarrow Y \cdot U_Z + W \cdot (1 - U_Z) \end{cases}$$



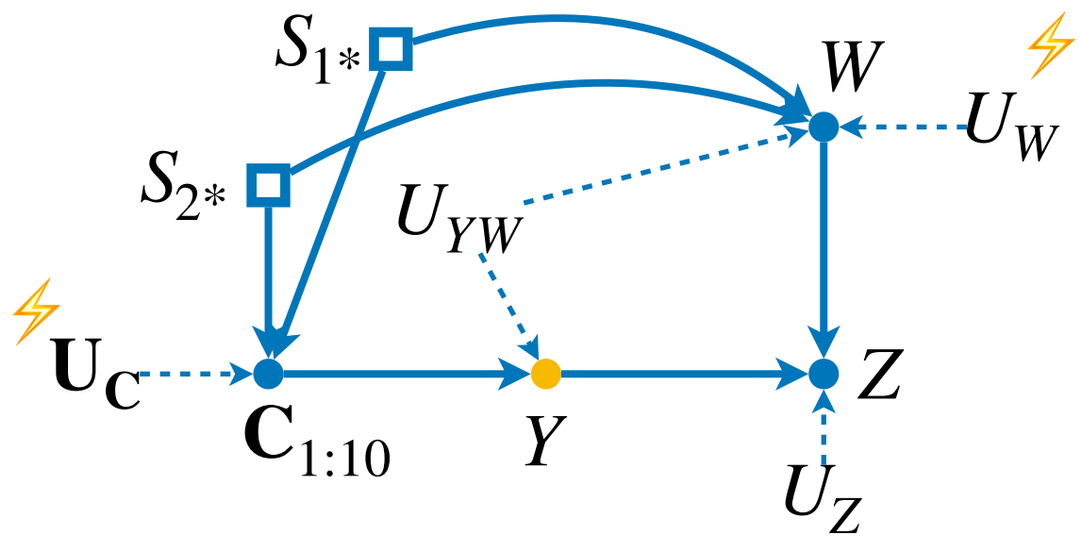
Only if we could access U_{YW}, we could predict Y even better!

Fact 2. $P^{1,2,*}(W \neq U_{YW}) \leq P^{1,2,*}(U_W = 1) = \begin{cases} 0.01 & \text{in } \mathscr{M}^1 \\ 0.02 & \text{in } \mathscr{M}^2 \\ 0.5 & \text{in } \mathscr{M}^* \end{cases}$

$\longrightarrow$ For $h_2(\mathbf{c}, w) = W \oplus \bigoplus_{j=1}^{10} C_j$, we would have: $\begin{cases} \mathscr{R}_{P^{1,2}}(h_1) \leq 0.02 \\ \mathscr{R}_{P^*}(h_1) \approx 0.5 \end{cases}$

Surprising observation

$$\mathscr{F}^{1,2,*}: \begin{cases} W \leftarrow U_{YW} \oplus U_W \\ C_j \leftarrow U_{C_j} \\ Y \leftarrow U_{YW} \oplus \bigoplus_{j=1}^{10} C_j \\ Z \leftarrow Y \cdot U_Z + W \cdot (1 - U_Z) \end{cases}$$

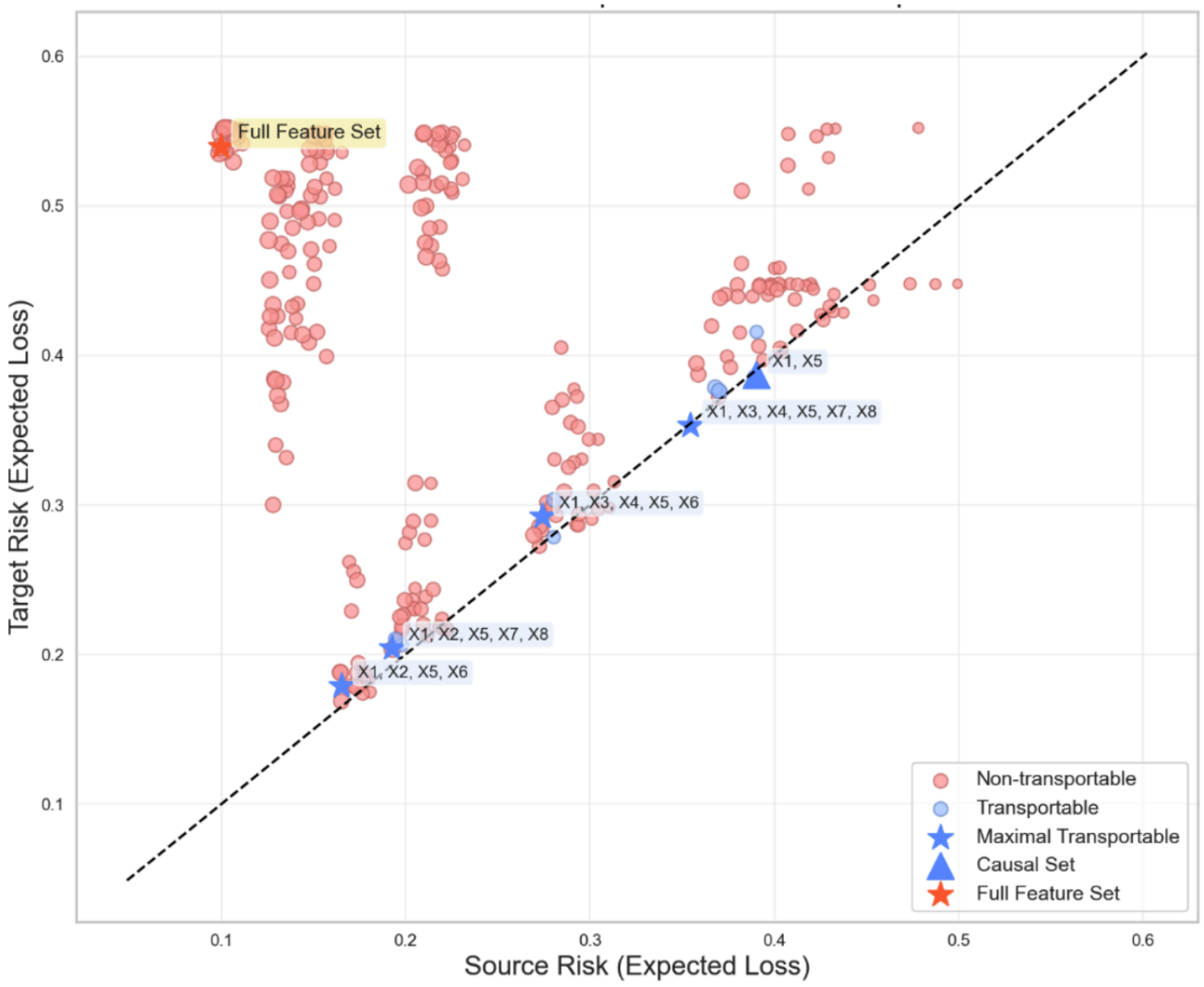


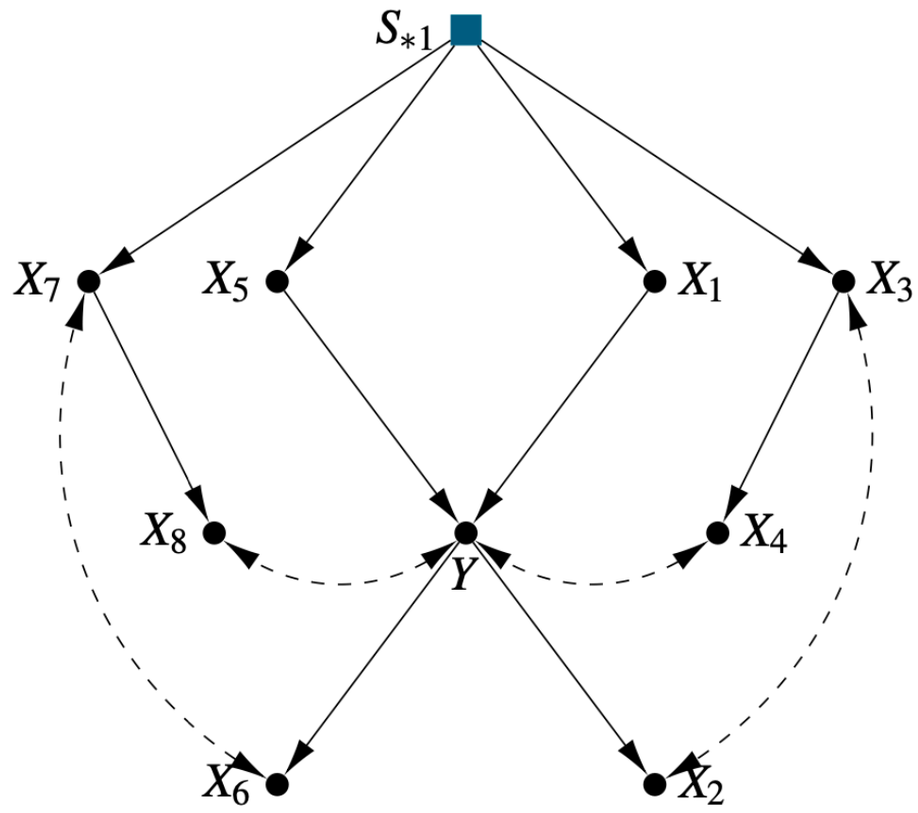
Only if we could access U_{YW}, we could predict Y even better!

Fact 3. $P^{1,2,*}(Z \neq Y) \leq P^{1,2,*}(U_Z) = 0.1$ in $\mathscr{M}^1, \mathscr{M}^2, \mathscr{M}*$

⟶ For $h_3(z) = z$, we would have $\mathscr{R}_{P^{1,2,*}}(h_1) \leq 0.1$

TR → Low Target Risk?



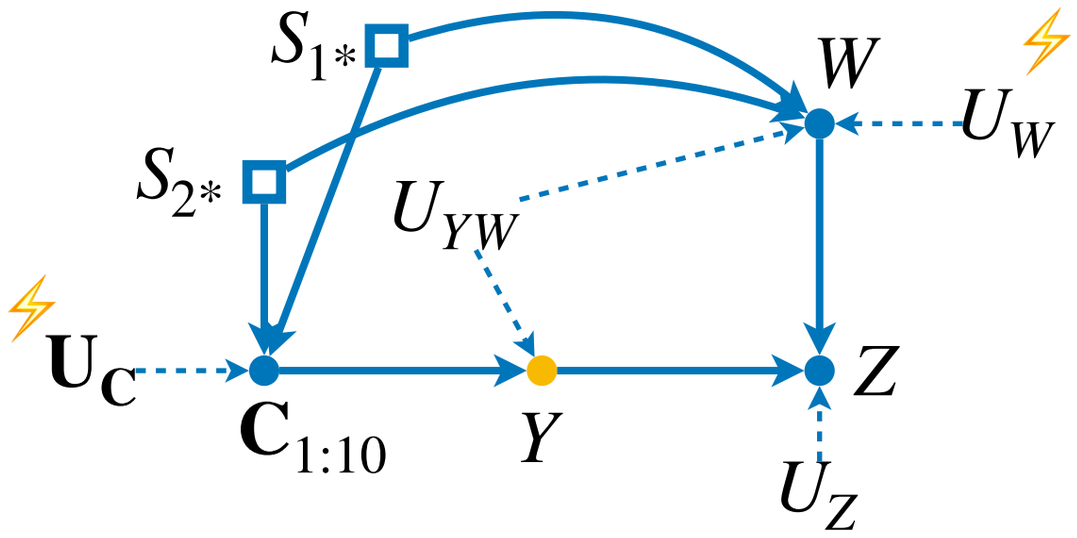


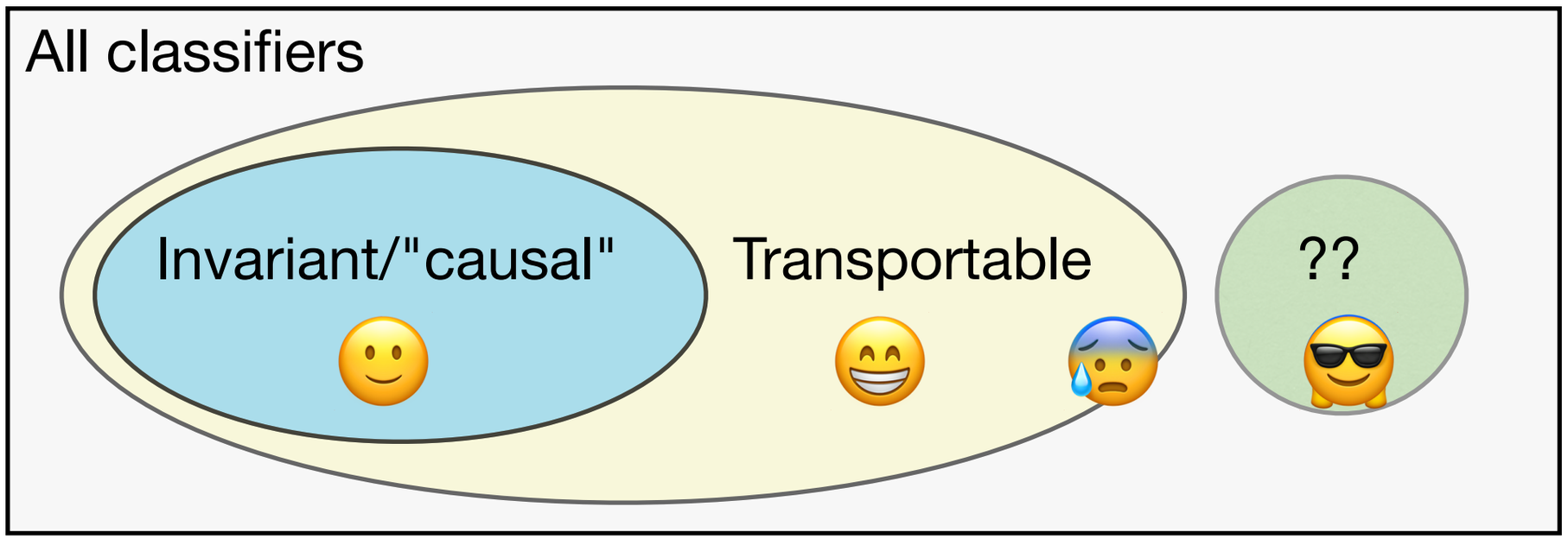
Previously, we saw that transportable classifiers have good source and target risk. On the other hand, non-transportable ones are off-diagonal, meaning that they mostly perform poorly.

The new classifier is not TR, but is very well-performing in the source and target domains.

Comparing classifiers

	$\mathscr{R}_{P1}$	$\mathscr{R}_{P2}$	$\mathscr{R}_{P*}$
$h_1(\mathbf{c})$	20 %	20 %	20 %
$h_2(\mathbf{c}, w)$	1 %	2 %	50 %
$h_3(z)$	3 %	5 %	4 %





🤔

Is there a final solution?

What is the core challenge here?

Partial Transportability

Definition 11.4.1 -- Partial transportability. Consider a tuple of source and target SCMs

$$\mathbb{M} = \langle \mathscr{M}^1, \mathscr{M}^2, \ldots, \mathscr{M}^K, \mathscr{M}^* \rangle$$

that induces the selection diagram $\mathscr{G}^{\Delta}$ over the variables $\mathbf{V}$ and entails the source distributions

$$\mathbb{P} = \langle P^1(\mathbf{v}), P^2(\mathbf{v}), \ldots, P^K(\mathbf{v}) \rangle$$

That we have access to, and the unseen target distribution $P^*(\mathbf{v})$. A functional $\psi : \text{supp}_{\mathbf{V}} \to \mathbb{R}$ is partially transportable from $\mathbb{P}$ given $\mathscr{G}^{\Delta}$ if,

$$\mathbb{E}_{P^{\mathscr{M}_0^*}}[\psi(\mathbf{V}] \leq q_{\max}, \forall \text{ SCMs } \mathbb{M}_0 \text{ that entail } \mathbb{P} \text{ and induce } \mathscr{G}^{\Delta},$$

Where $q_{\max} \in \mathbb{R}$ is a constant that can be computed from $\mathbb{P}, \mathscr{G}^{\Delta}$.

For a fixed classifier h, let $\psi(\mathbf{x}, y) = \mathscr{L}(y, h(x))$. Then the above is an upper-bound for the worst-case target risk!

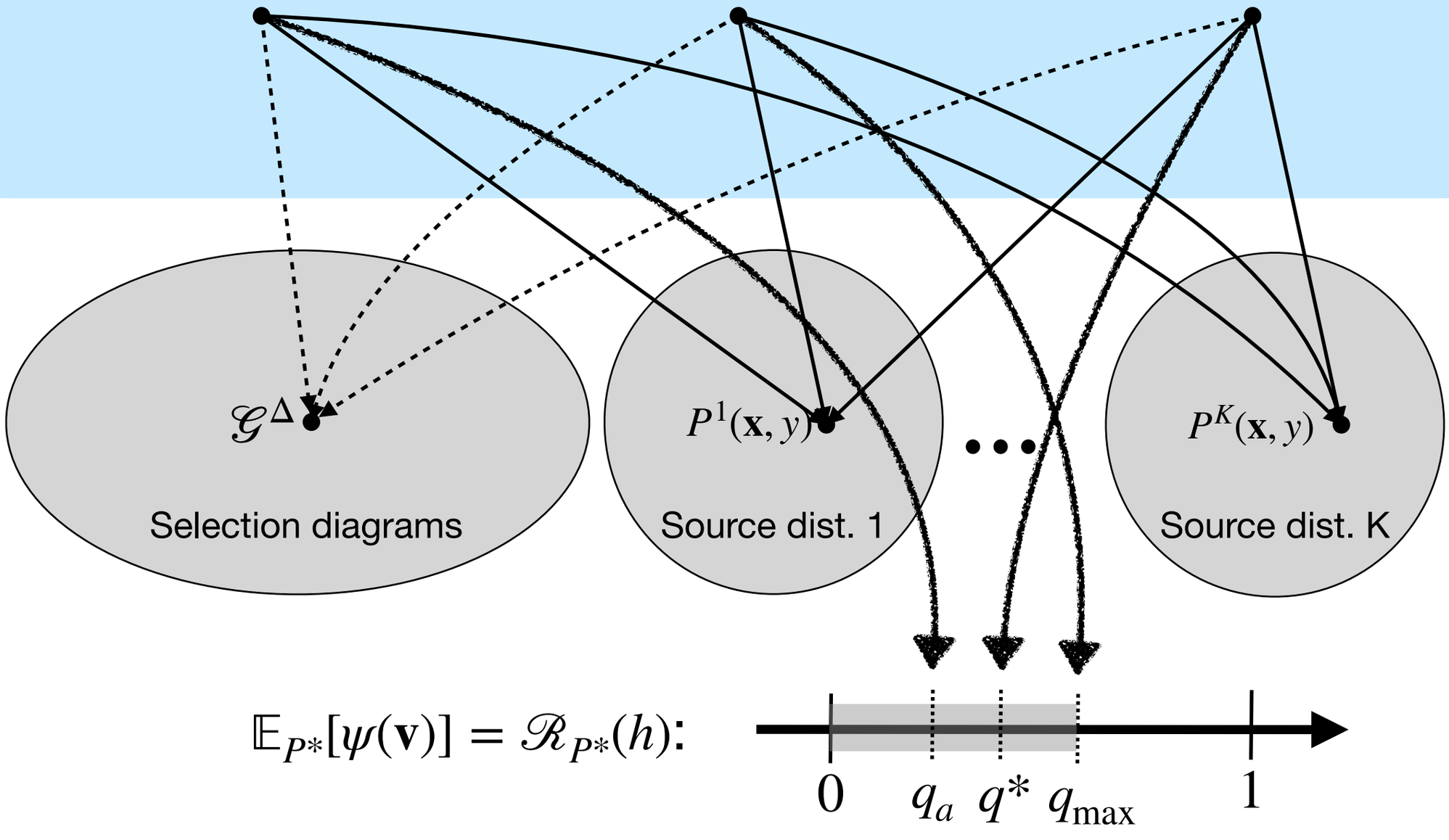
$$\mathscr{R}_{P^*}(h) \leq \max_{\mathbb{M}_0} \mathscr{R}_{P^{\mathscr{M}_0^*}}(h)$$

Partial-TR Schema

SCM tuples

(True SCM tuple)

$\mathbb{M}_a = \langle \mathscr{M}_a^1, \mathscr{M}_a^2, \ldots, \mathscr{M}_a^K, \mathscr{M}_a^* \rangle$ $\mathbb{M}_b = \langle \mathscr{M}_b^1, \mathscr{M}_b^2, \ldots, \mathscr{M}_b^K, \mathscr{M}_b^* \rangle$ $\mathbb{M} = \langle \mathscr{M}^1, \mathscr{M}^2, \ldots, \mathscr{M}^K, \mathscr{M}^* \rangle$



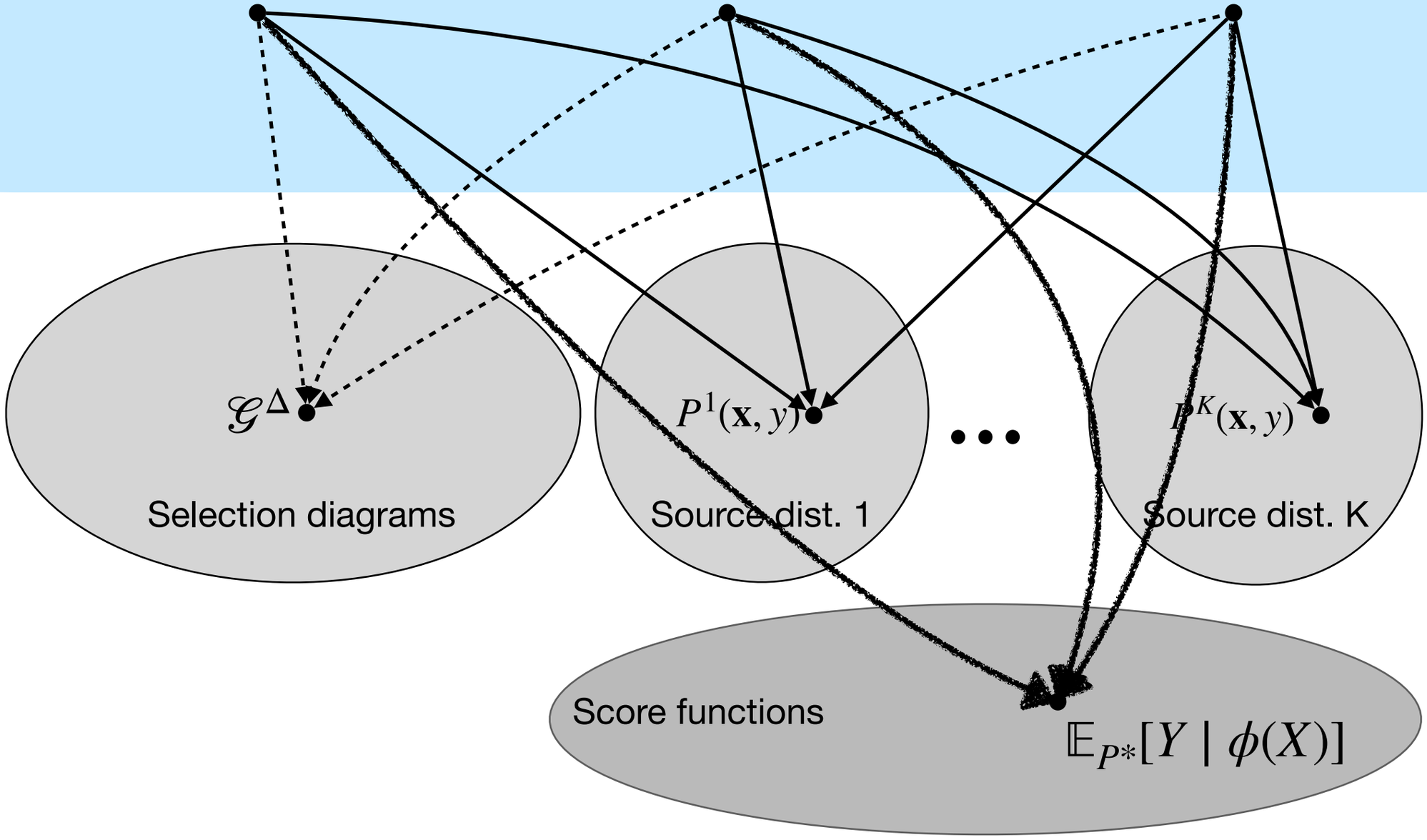
$\mathbb{E}_{P^*}[\psi(\mathbf{v})] = \mathscr{R}_{P^*}(h)$:

Recall: Repr. TR Schema

SCM tuples

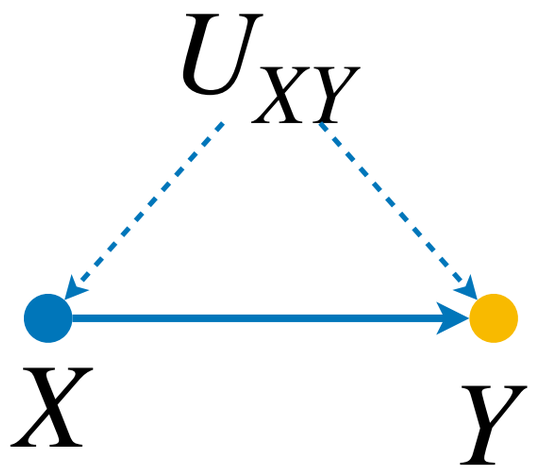
(True SCM tuple)

$\mathbb{M}_a = \langle \mathscr{M}_a^1, \mathscr{M}_a^2, \ldots, \mathscr{M}_a^K, \mathscr{M}_a^* \rangle$ $\mathbb{M}_b = \langle \mathscr{M}_b^1, \mathscr{M}_b^2, \ldots, \mathscr{M}_b^K, \mathscr{M}_b^* \rangle$ $\mathbb{M} = \langle \mathscr{M}^1, \mathscr{M}^2, \ldots, \mathscr{M}^K, \mathscr{M}^* \rangle$



Optimization-based Transportability

Example--the bow model.



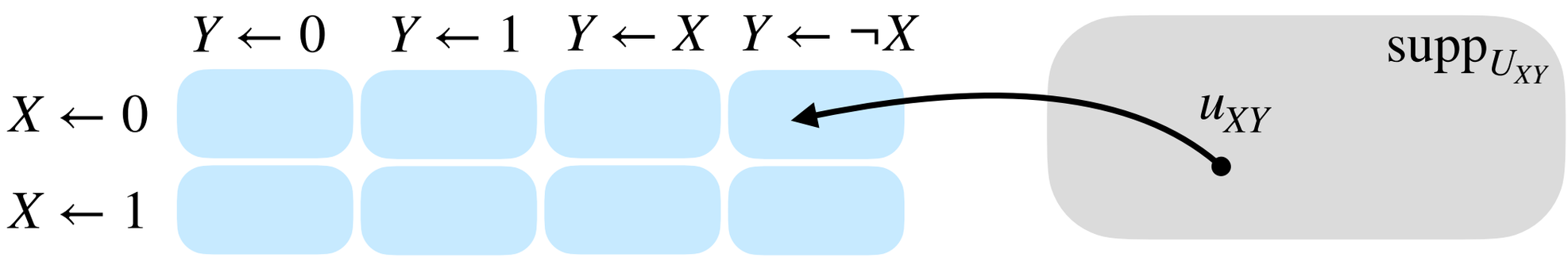
An SCM over binary X, Y.

For a fixed value $U_{XY} = u_{XY}$:

1. the variable X takes value from the set $\{0,1\}$

2. the variable Y takes value **based on** X through functions $\{0, X, \neg X, 1\}$

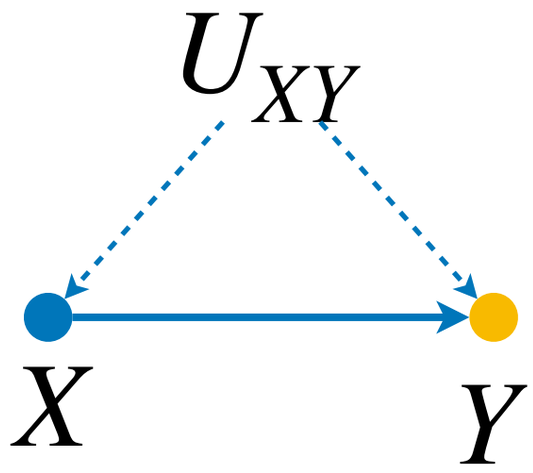
→ Every value of U_{XY} belongs to one of $2 \times 4 = 8$ categories:



Canonical Partitioning

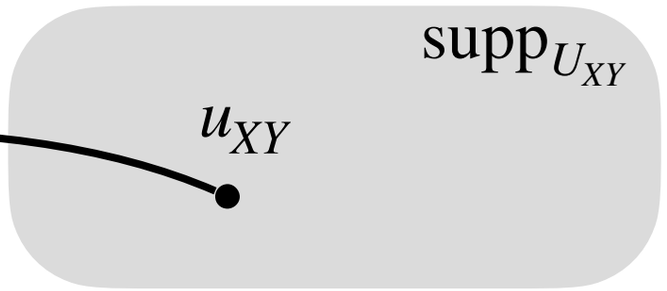
Example--the bow model.

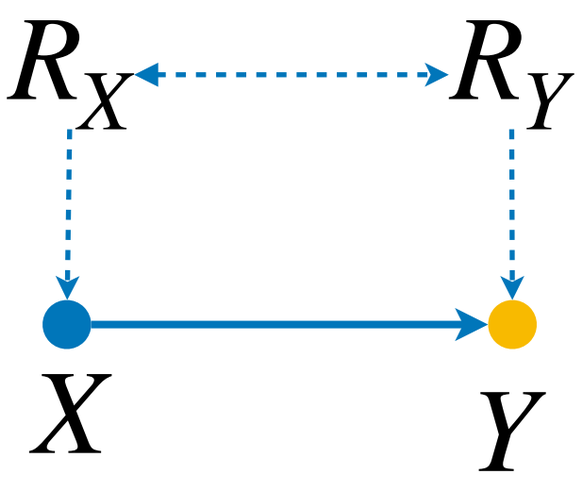
An SCM over binary X, Y.



$$(R_Y = 1)(R_Y = 2)\,(R_Y = 3)(R_Y = 4)$$

		$Y \leftarrow 0$	$Y \leftarrow 1$	$Y \leftarrow X$	$Y \leftarrow \neg X$
$(R_X = 1)$	$X \leftarrow 0$	(1,1)	(1,2)	(1,3)	(1,4)
$(R_X = 2)$	$X \leftarrow 1$	(2,1)	(2,2)	(2,3)	(2,4)





$$P(R_X, R_Y) \quad \longleftarrow\!\!\!\!\!\!\longrightarrow \quad P(U_{XY})$$

Support is known

Distribution can be parameterized by $\theta \in \Delta^7$

Support is unknown

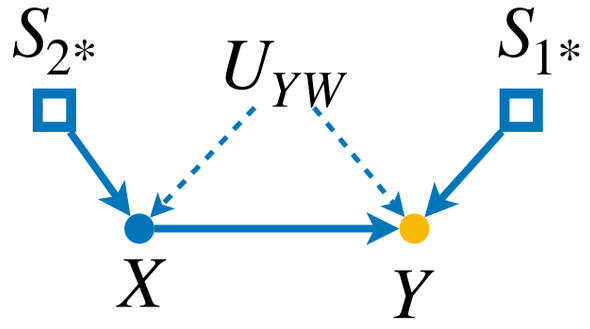
No parametrization is possible

Canonical SCMs

Proposition 11.4.2 -- (bow canonical parametrization). For every SCM $\mathscr{M}$ over binary X, Y that induces the bow graph, there exists a canonical SCM $\mathscr{N}$ specified by $P(R_X, R_Y)$ (that can be parameterized by a point in the simplex Δ^7) such that $\mathscr{M}$ and $\mathscr{N}$ agree on all L1,L2,L3 queries.

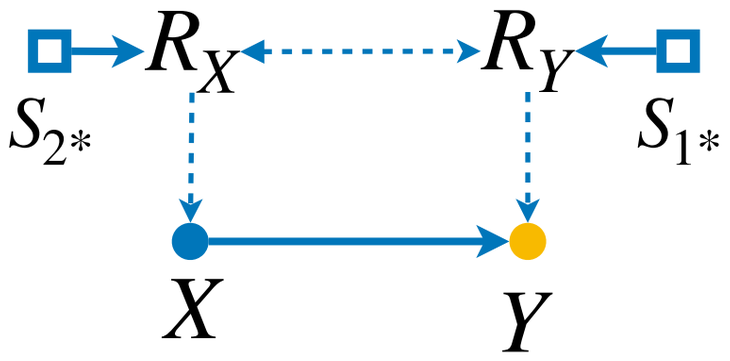
This means that we can not parameterize the space of SCMs!

Example. Consider SCMs $\mathscr{M}^1, \mathscr{M}^2, \mathscr{M}*$.



Consider a classifier $h(x) = \neg x$ (predicts $y = \neg x$).

Question. What is the worst-case risk $\mathscr{R}_{P*}(h)$?

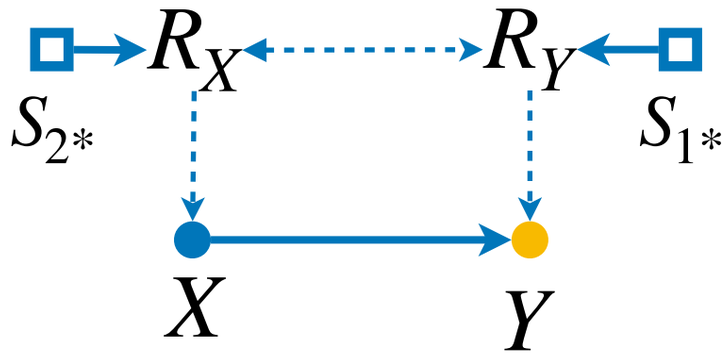


Idea. Use canonical parametrization for all SCMs!

Canonical SCMs for partial-TR

Consider a classifier $h(x) = \neg x$ (predicts $y = \neg x$).

Question. What is the worst-case risk $\mathscr{R}_{P^*}(h)$?



Idea. Parametrize canonical SCMs $\mathscr{N}^1, \mathscr{N}^2, \mathscr{N}^*$, and then

1. Impose distributional constraints $P^{\mathscr{N}^i}(x, y) = P^i(x, y)$ for $i \in \{1,2\}$

2. Impose domain discrepancies: $\begin{cases} \Delta_{1*} = \{Y\} \implies P^{\mathscr{N}^1}(R_X) = P^{\mathscr{N}^*}(R_X) \\ \Delta_{2*} = \{Y\} \implies P^{\mathscr{N}^2}(R_Y) = P^{\mathscr{N}^*}(R_Y) \end{cases}$

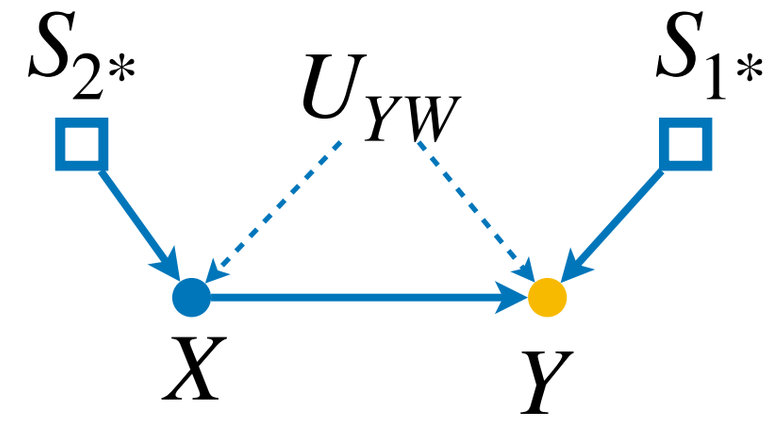
3. Solve $q_{\max} \leftarrow \max_{\mathscr{N}^1, \mathscr{N}^2, \mathscr{N}^*} P^{\mathscr{N}^*}(h(X) \neq Y)$

4. Claim $R_{P^*}(h) \leq q_{\max}$ Tight? 🤔

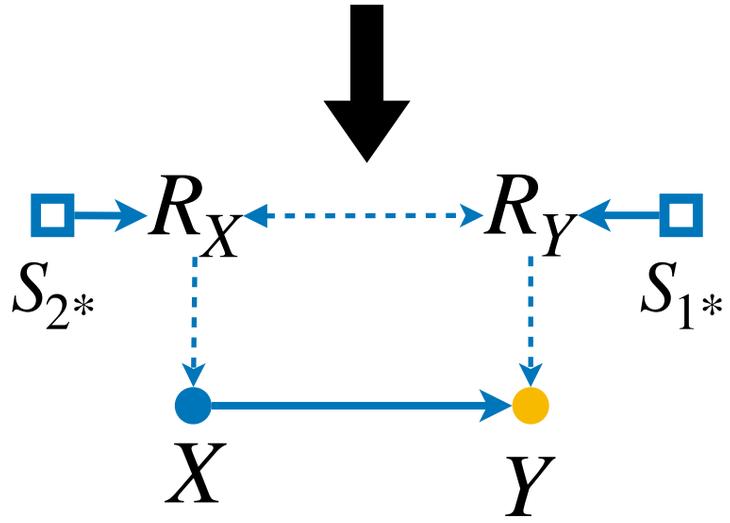
Canonical SCMs for partial-TR

Consider a classifier $h(x) = \neg x$ (predicts $y = \neg x$).

Question. What is the worst-case risk $\mathscr{R}_{P^*}(h)$?



Optimization problem. In this case, it is a linear program, with linear constraints and linear objective.



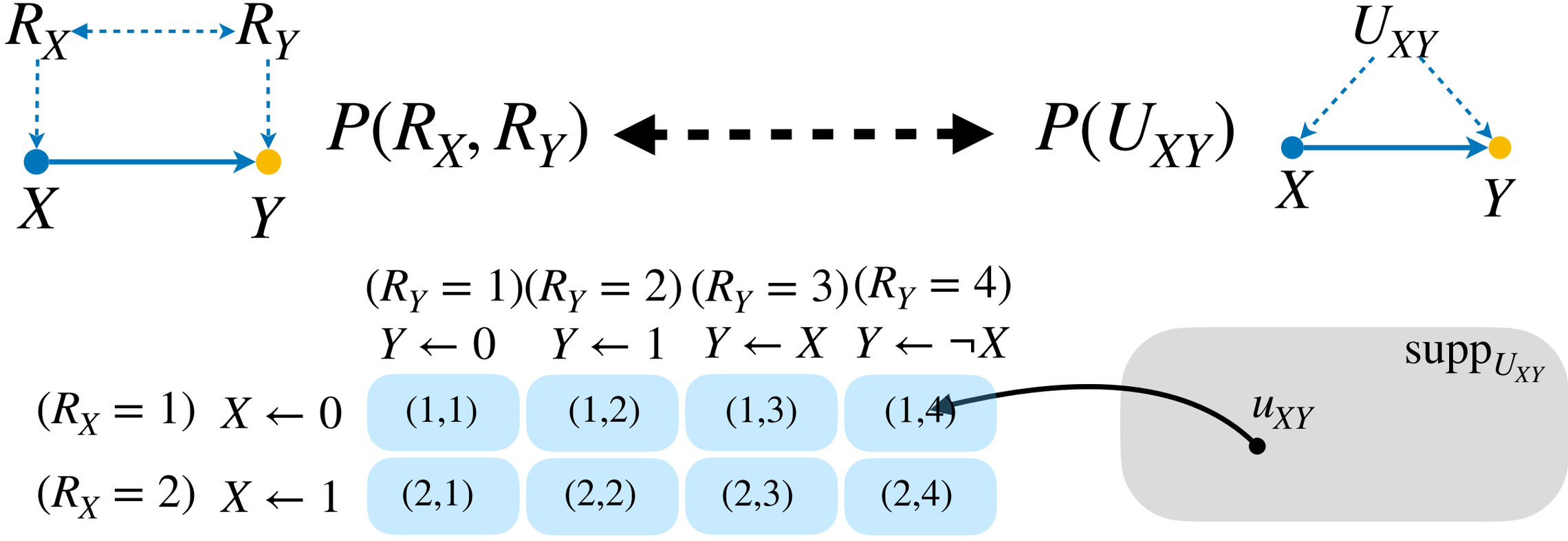
$$\max_{\mathcal{N}^1,\mathcal{N}^2,\mathcal{N}^*} P^{\mathcal{N}^*}(Y \neq \neg X)$$

$$\text{s.t. } P^{\mathcal{N}^1}(r_Y) = P^{\mathcal{N}^*}(r_Y), \quad P^{\mathcal{N}^2}(r_X) = P^{\mathcal{N}^*}(r_X) \qquad (Y \notin \Delta_1, \text{ and } X \notin \Delta_2)$$

$$P^{\mathcal{N}^1}(x,y) = P^1(x,y), \quad P^{\mathcal{N}^2}(x,y) = P^2(x,y) \qquad \text{(matching source dists)}$$

Canonical SCMs for partial-TR

Definition 5 (Canonical SCM)**.** A canonical SCM is an SCM $\mathcal{N} = \langle \boldsymbol{U}, \boldsymbol{V}, \mathcal{F}, P(\boldsymbol{U}) \rangle$ defined as follows. The set of endogenous variables $\boldsymbol{V}$ is discrete. The set of exogenous variables $\boldsymbol{U} = \{R_V : V \in \boldsymbol{V}\}$, where $\text{supp}_{R_V} = \{1, \ldots, m_V\}$ and $m_V = |\{h_V : \text{supp}_{pa_V} \rightarrow \text{supp}_V\}|$. For each $V \in \boldsymbol{V}$, $f_V \in \mathcal{F}$ is defined as $f_V(pa_V, r_V) = h_V^{(r_V)}(pa_V)$.



Canonical SCMs for partial-TR

Theorem 1 (Partial-TR with canonical models)**.** *Consider the tuple of SCMs* $\mathbb{M}$ *that induces the selection diagram* $\mathcal{G}^{\boldsymbol{\Delta}}$ *over the variables* $\boldsymbol{V}$*, and entails the source distributions* $\mathbb{P}$*, and the target distribution* P^**. Let* $\psi : \Omega_{\boldsymbol{V}} \to \mathbb{R}$ *be a functional of interest. Consider the following optimization scheme:*

$$\max_{\mathcal{N}^1,\mathcal{N}^2,\ldots,\mathcal{N}^*} \mathbb{E}_{P^{\mathcal{N}^*}}[\psi(\boldsymbol{V})] \; s.t. \; P^{\mathcal{N}^i}(\boldsymbol{v}) = P^i(\boldsymbol{v}) \qquad \forall i \in \{1,2,\ldots,K,*\} \tag{6}$$

$$P^{\mathcal{N}^i}(r_V) = P^{\mathcal{N}^j}(r_V), \quad \forall i,j \in \{1,2,\ldots,K,*\} \quad \forall V \notin \Delta_{i,j}$$

where each $\mathcal{N}^i$ *is a canonical model characterized by a joint distribution over* $\{R_V\}_{V \in \boldsymbol{V}}$*. The value of the above optimization is a tight upper-bound for the quantity* $\mathbb{E}_{P^*}[\psi(\boldsymbol{V})]$ *among all tuples of SCMs that induce* $\mathcal{G}^{\boldsymbol{\Delta}}$ *and entail* $\mathbb{P}$*.* □

Takeaway. Every functional of the variables, e.g., the risk of a classifier, can be upper-bounded using the canonical parameterization by imposing distributional and domain-discrepancy assumptions.

Canonical SCMs for partial-TR schema

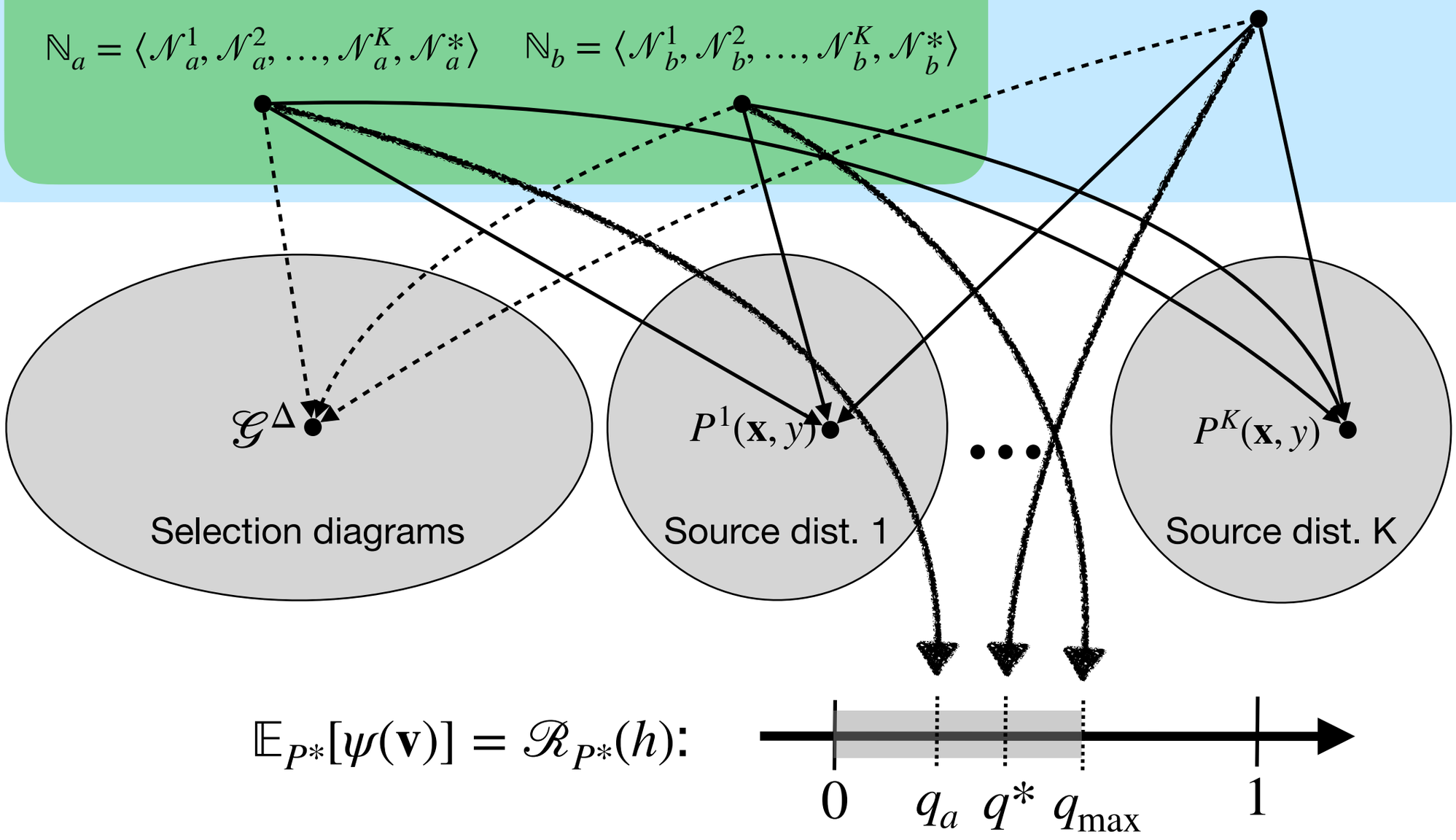
SCM tuples

(True SCM tuple)

$\mathbb{M} = \langle \mathscr{M}^1, \mathscr{M}^2, \ldots, \mathscr{M}^K, \mathscr{M}^* \rangle$

Canonical SCM tuples

$\mathbb{N}_a = \langle \mathscr{N}_a^1, \mathscr{N}_a^2, \ldots, \mathscr{N}_a^K, \mathscr{N}_a^* \rangle$ $\mathbb{N}_b = \langle \mathscr{N}_b^1, \mathscr{N}_b^2, \ldots, \mathscr{N}_b^K, \mathscr{N}_b^* \rangle$



$\mathbb{E}_{P^*}[\psi(\mathbf{v})] = \mathscr{R}_{P^*}(h)$:

Summary

- Previously, we considered $P^*(y \mid \phi)$ as a query that corresponds to the generalization capacity of a rep/classifier.
- This way, our search was limited to TR cases, e.g., finding maximal transportable representations through UIRM program.
- However, we realized that the query $P^*(Y \neq h(\mathbf{X}))$ captures generalizability more accurately.
- Often it is not transportable, but we can still bound this quantity.

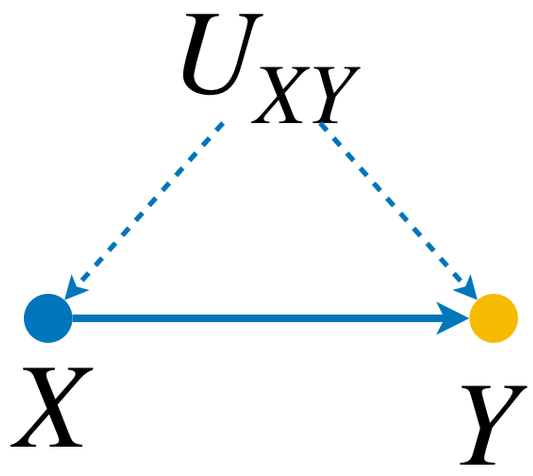
Can we utilize gradient-based methods for this task?

Neural Parametrization

Example--the bow model.

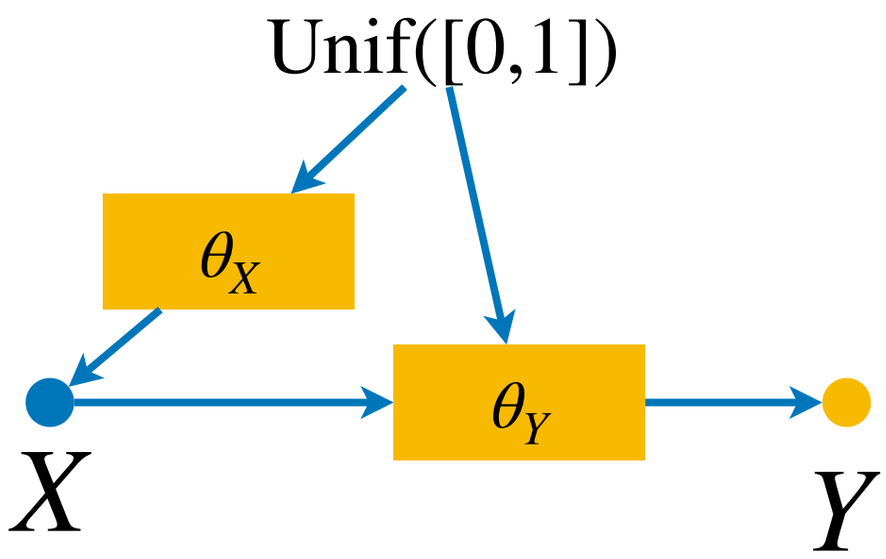
An SCM over binary X, Y.

Recall the Neural Causal Model corresponding to the bow causal diagram.



feed-forward neural networks parametrized by θ_X, θ_Y encode the structural constraints.

We can compute $P(x, y; \theta_X, \theta_Y)$ through sampling, and using L1 data $D \sim P(x, y)$, we can impose distribution constraints by maximizing Likelihood.

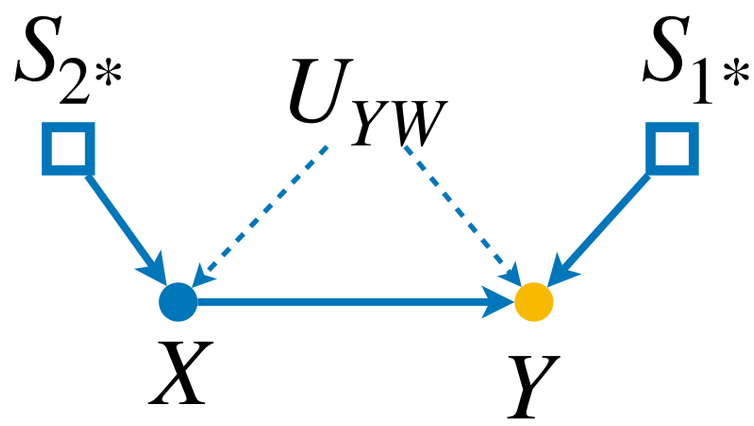


Domain discrepancies in NCMs

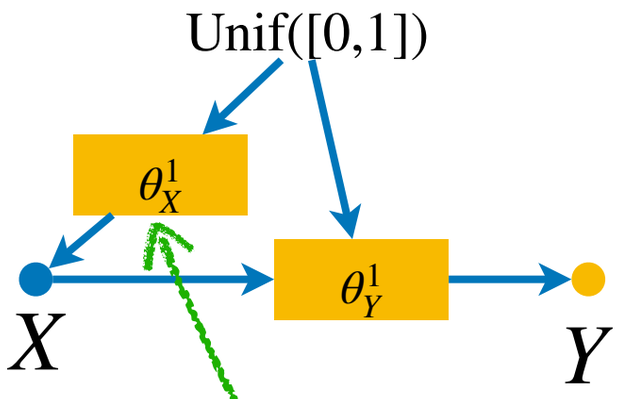
Example--the bow model.

Source SCMs $\mathscr{M}^1, \mathscr{M}^2$ and target SCM $\mathscr{M}^*$

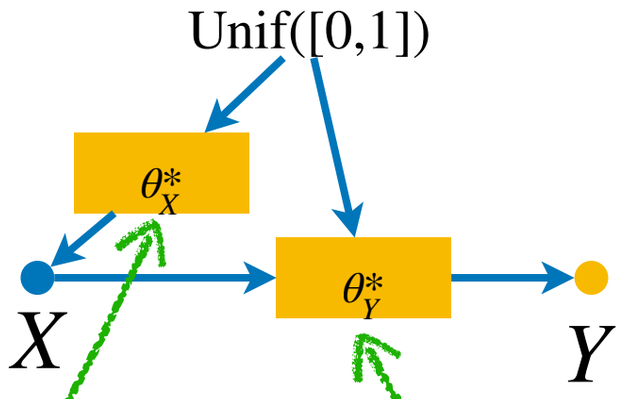
Data D^1, D^2



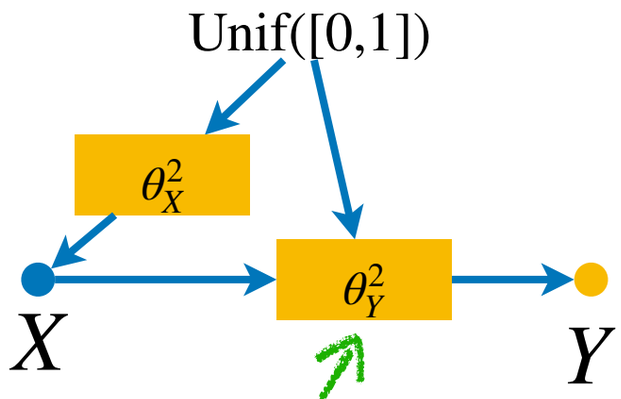
NCM $\theta^1 = \langle \theta^1_X, \theta^1_Y \rangle$



NCM $\theta^* = \langle \theta^*_X, \theta^*_Y \rangle$

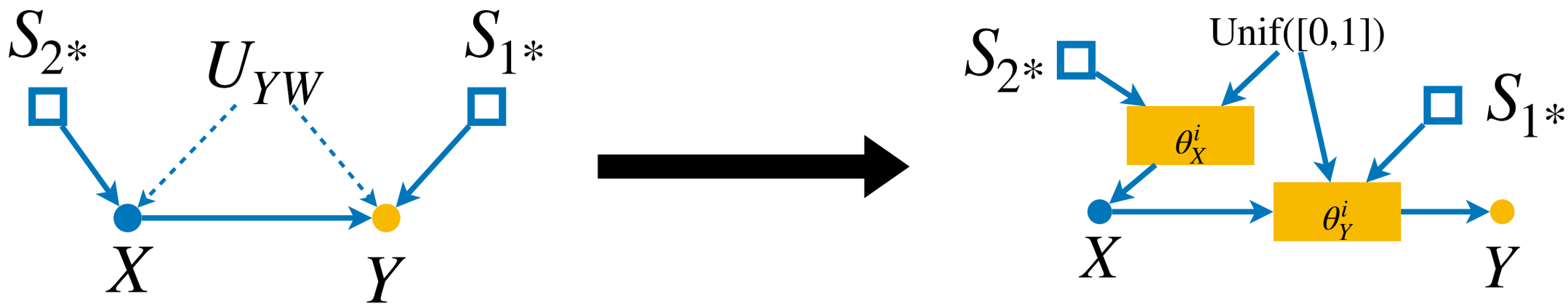


NCM $\theta^2 = \langle \theta^2_X, \theta^2_Y \rangle$



Distributional constraints

Example--the bow model. Source SCMs $\mathscr{M}^1$, $\mathscr{M}^2$ and target SCM $\mathscr{M}^*$

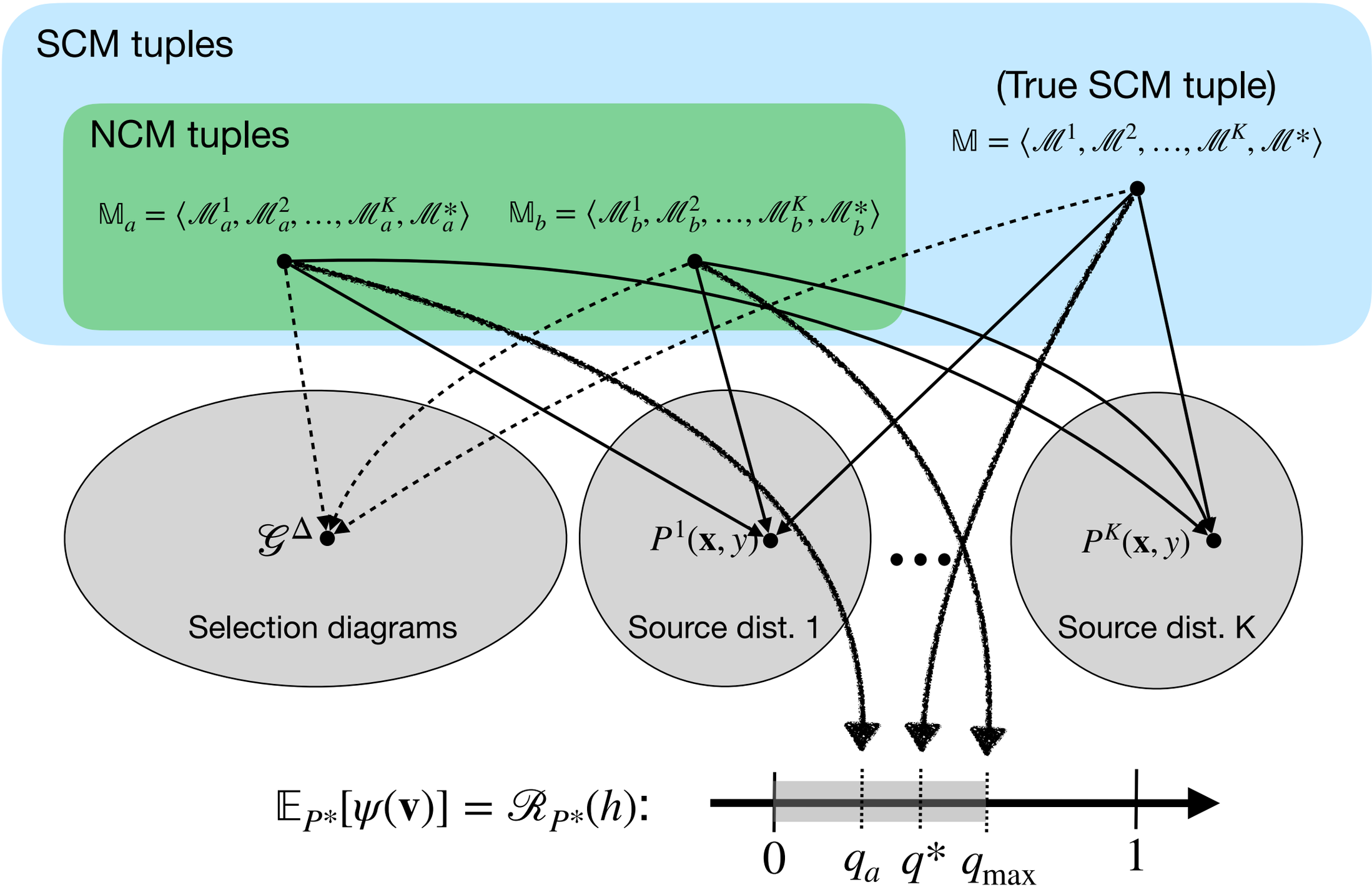


Consider the classifier $h(x) = \neg x$. The optimization program below estimates the upper-bound for target risk of h for $\lambda \to \infty$.

$$\max_{\theta^1,\theta^2,\theta^*} \underbrace{\mathbb{E}[Y \neq h(X); \theta^*]}_{\text{Target risk } \mathscr{R}_{P^*}(h)} + \lambda\Big(\underbrace{\sum_{x,y \in D^1} \log P(x, y; \theta^1)}_{\mathscr{M}_{\theta_1} \overset{L1}{=} \mathscr{M}^1} + \underbrace{\sum_{x,y \in D^2} \log P(x, y; \theta^2)}_{\mathscr{M}_{\theta_2} \overset{L1}{=} \mathscr{M}^2} \Big)$$

Subject to $\theta_X^1 = \theta_X^*, \theta_Y^2 = \theta_Y^*$

Partial-TR with NCMs Schema



Partial-TR with NCMs

Theorem 2 (Partial-TR with NCMs)**.** *Consider a tuple of SCMs* $\mathbb{M}$ *that induces* $\mathcal{G}^{\boldsymbol{\Delta}}$, $\mathbb{P}$ *and* P^* *over the variables* $\boldsymbol{V}$*. Let* $D^i \sim P^i(x, y)$ *denote the samples drawn from the* i*-th source domain. Let* θ^i *denote the parameters of NCM corresponding to* $\mathcal{M}^i \in \mathbb{M}$*. Let* $\mathbb{E}_{P*}[\psi(\boldsymbol{V})]$ *be the target quantity. The solution to the optimization problem,*

$$\hat{\Theta} \in \underset{\Theta:\langle\theta^1,\theta^2,\ldots,\theta^K,\theta*\rangle}{\arg\max} \sum_{\boldsymbol{w}} \psi(\boldsymbol{w}) \cdot \sum_{\boldsymbol{v}\setminus\boldsymbol{w}} P(\boldsymbol{v};\theta^*) \tag{8}$$

$$s.t.\ \theta^i_V = \theta^j_V, \qquad \forall i,j \in \{1,2,\ldots,K,*\} \quad \forall V \notin \Delta_{i,j}$$

$$\theta^i \in \arg\max_{\theta} \sum_{\boldsymbol{v}\in D^i} \log P(\boldsymbol{v};\theta), \quad \forall i \in \{1,2,\ldots,K\}.$$

is a tuple of NCMs that induce $\mathcal{G}^{\boldsymbol{\Delta}}$*, entails* $\mathbb{P}$*. In the large sample limit, the solution yields a tight upper-bound for* $\mathbb{E}_{P*}[\psi(\boldsymbol{V})]$*.* □

Takeaway. Through NCM parametrization we can bound non-transportable queries at [possibly large] computational cost.

Neural-TR algorithm

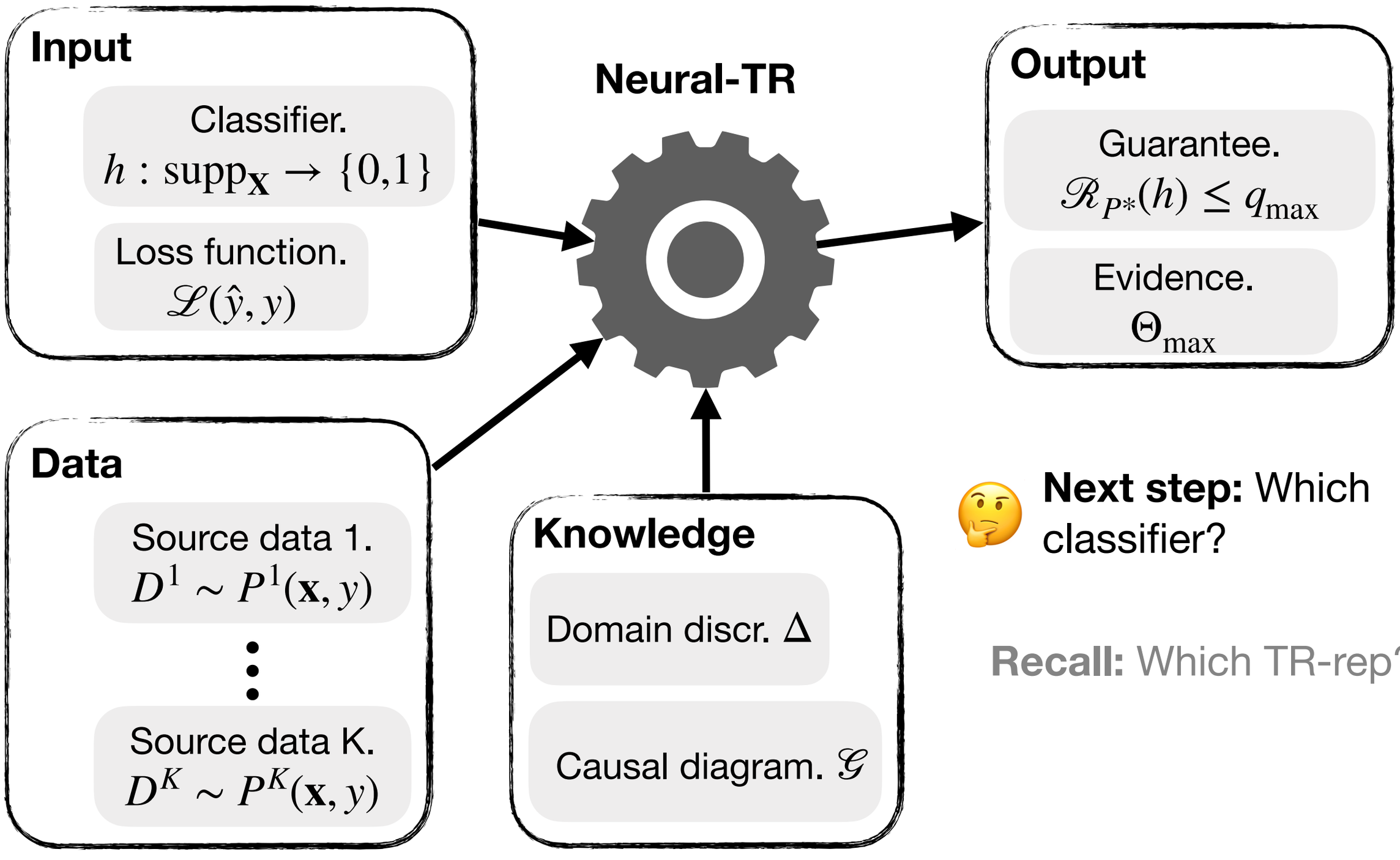
Algorithm 1 Neural-TR

Require: Source data $D^1, D^2, \ldots, D^K$; selection diagram $\mathcal{G}^{\boldsymbol{\Delta}}$; functional $\psi : \Omega_{\boldsymbol{W}} \to [0, 1]$.
Ensure: Upper-bound for $\mathbb{E}_{P^*}[\psi(\boldsymbol{W})]$

$\{\boldsymbol{A}_j\}_{j=1}^m \leftarrow$ c-components of $\boldsymbol{A} := \mathbf{An}_{\mathcal{G}^*}(\boldsymbol{W})$ in causal diagram $\mathcal{G}^*$.
$\Theta, \mathbb{C}_{\text{expert}} \leftarrow \varnothing, \quad \mathcal{L}_{\text{data}} \leftarrow 0$
$P^*(\boldsymbol{w}) := \sum_{\boldsymbol{a} \setminus \boldsymbol{w}} \prod_{j=1}^m P^*(\boldsymbol{a}_j \mid \text{do}(pa_{\boldsymbol{A}_j}))$
for $j = 1$ to m **do**
  if $\exists i \in \{1, 2, \ldots, K\}$ such that $\boldsymbol{A}_j \cap \Delta_{*i} = \varnothing$ **then**
    $\eta^i_{\boldsymbol{A}_j} \leftarrow \arg\max_{\eta_{\boldsymbol{A}_j}} \sum_{\boldsymbol{a}_j, pa_{\boldsymbol{A}_j} \in D^i} \log P(\boldsymbol{a}_j \mid \text{do}(pa_{\boldsymbol{A}_j}); \eta_{\boldsymbol{A}_j})$
    In $P^*(\boldsymbol{w})$, replace $P^*(\boldsymbol{a}_j \mid \text{do}(pa_{\boldsymbol{A}_j}))$ with $P(\boldsymbol{a}_j \mid \text{do}(pa_{\boldsymbol{A}_j}); \eta^i_{\boldsymbol{A}_j})$.
  else
    $\Theta \leftarrow \Theta \cup \{\theta^i_{\boldsymbol{A}_j}\}_{i \in \{1,2,\ldots,K,*\}}$
    In $P^*(\boldsymbol{w})$, replace $P^*(\boldsymbol{a}_j \mid \text{do}(pa_{\boldsymbol{A}_j}))$ with $P(\boldsymbol{a}_j \text{do}(pa_{\boldsymbol{A}_j}); \theta^*_{\boldsymbol{A}_j})$.
    $\mathbb{C}_{\text{expert}} \leftarrow \mathbb{C}_{\text{expert}} \cup \{\{\theta^i_V = \theta^*_V\}_{V \in \boldsymbol{A}_j \setminus \Delta_{*i}}\}_{i=1}^K$
    $\mathcal{L}_{\text{likelihood}} \leftarrow \mathcal{L}_{\text{likelihood}} + \sum_{\boldsymbol{a}_j, pa_{\boldsymbol{A}_j} \in D^i} \log P(\boldsymbol{a}_j, \text{do}(pa_{\boldsymbol{A}_j}); \theta^i_{\boldsymbol{A}_j})$.
  end if
end for
Return $\hat{\Theta} \leftarrow \arg\max_{\Theta} \sum_{\boldsymbol{w}} P^*(\boldsymbol{w}; \Theta) \cdot \psi(\boldsymbol{w}) + \Lambda \cdot \mathcal{L}_{\text{likelihood}}(\Theta)$ subject to $\mathbb{C}_{\text{expert}}$

Proposition 1. *Neural-TR (Algorithm 1) computes a tuple of NCMs compatible with the source data and graphical assumptions that yields the upper-bound for* $\mathbb{E}_{P^*}[\psi(\boldsymbol{W})]$ *in the large sample limit.* □

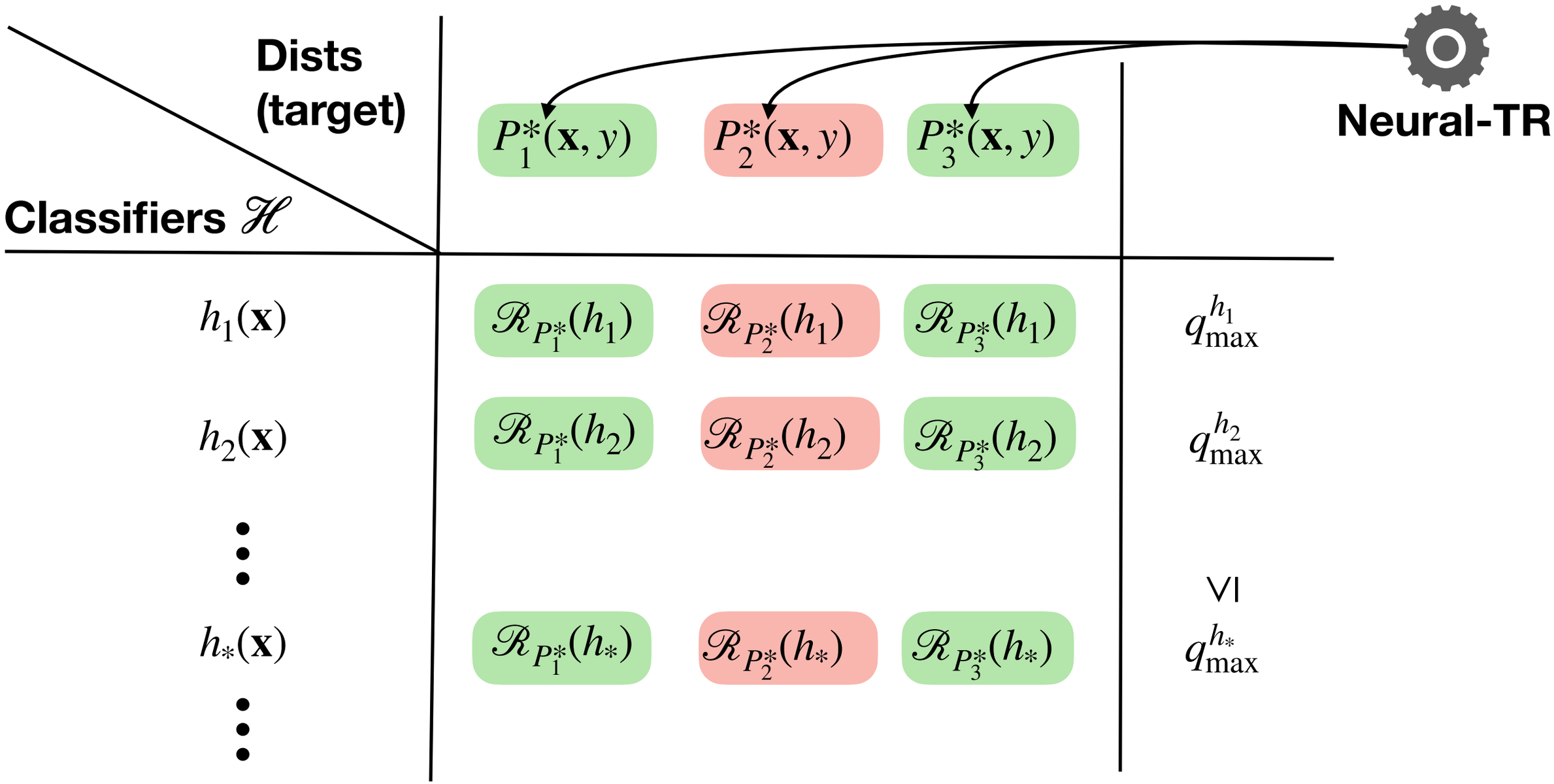
Neural-TR



🤔 **Next step:** Which classifier?

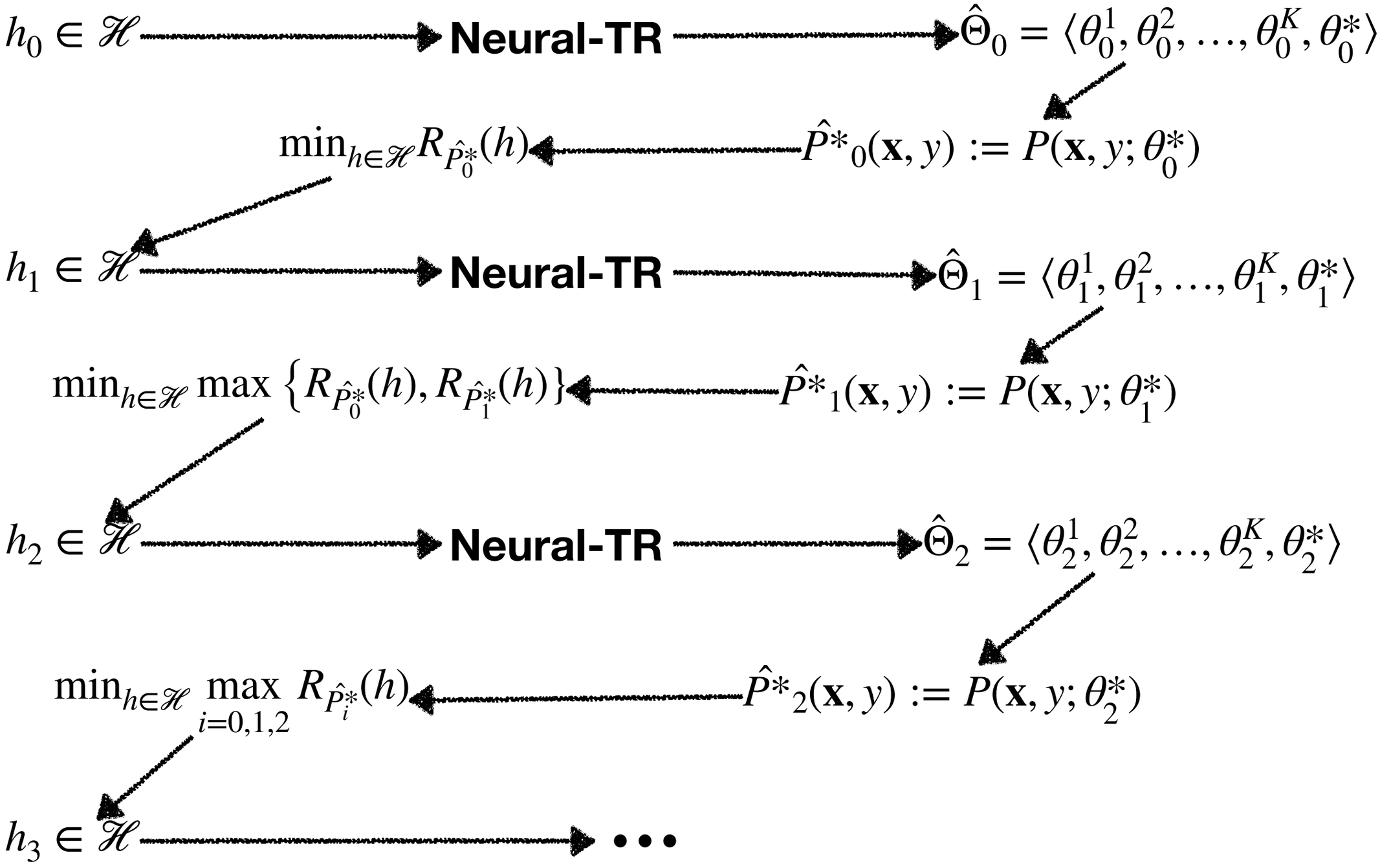
Recall: Which TR-rep?

Searching over classifiers



$$h_* \in \operatorname{argmin}_{h \in \mathscr{H}} \max_{\tilde{P}(\mathbf{x},y) \text{ is valid}} \mathscr{R}_{\tilde{P}}(h)$$

Causal Robust Optimization (CRO)



Causal Robust Optimization (CRO)

Dist. Robust Optimization

$$\min_{h \in \mathcal{H}} \max_{\tilde{P} \in \mathcal{P}} R_{\tilde{P}}(h)$$

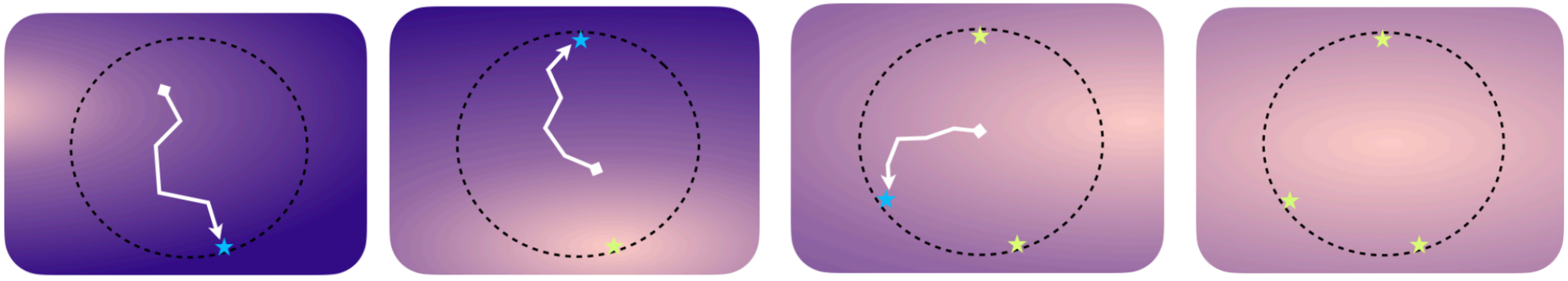
Worst-case risk evaluator

$$\theta^* \leftarrow \text{NeuralTR}(h; \mathbb{P}; \mathcal{G}^{\Delta})$$

Sampler

$$\mathcal{P} \leftarrow \mathcal{P} \cup \{P(\mathbf{x}, y; \theta^*)\}$$

Space of distributions



(a) Iteration 1 (b) Iteration 2 (c) Iteration 3 (d) Equilibrium

Causal Robust Optimization (CRO)

Algorithm 2 CRO (Causal Robust Optimization)

Require: $\mathbb{D} : \langle D^1, D^2, \ldots, D^K \rangle$; $\mathcal{G}^{\mathbf{\Delta}}$; $\delta > 0$
Ensure: $h(\boldsymbol{X})$ with the best worst-case risk.

```
Initialize h randomly and 𝔻* ← ∅
Θ̂ ← Neural-TR(𝔻, 𝒢^Δ, ψ : ℒ(h(x), y))
while R_{P(x,y;θ̂*)}(h) − max_{D∈𝔻*} R_D(h) > δ do
    𝔻* ← 𝔻* ∪ {D* ~ P(x, y; θ*)}
    h ← arg min_h max_{D∈𝔻*} R_D(h)
    Θ̂ ← Neural-TR(𝔻, 𝒢^Δ, ψ : ℒ(h(x), y))
end while
Return h
```

Theorem -- (DG with CRO). Algorithm 2 returns a worst-case optimal classifier from the hypothesis class, that is:

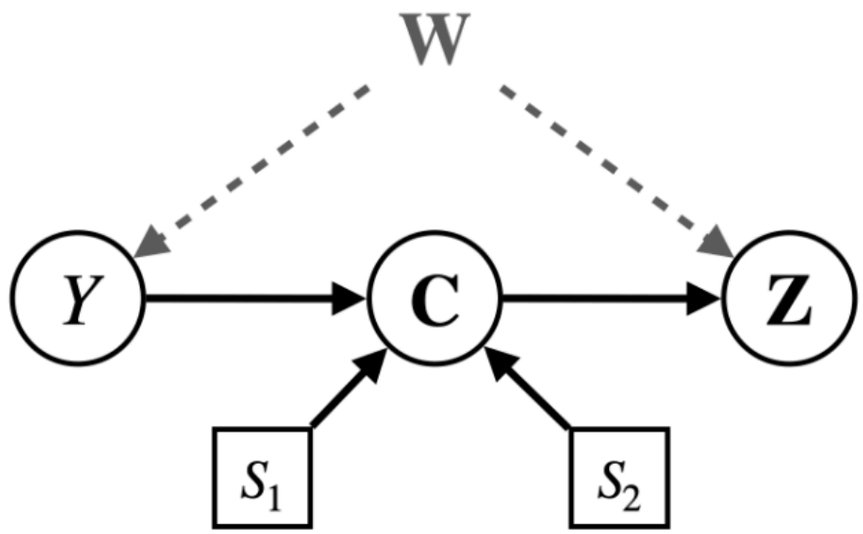
$$\mathrm{CRO}(\mathbb{D}, \mathcal{G}^{\mathbf{\Delta}}) \in \underset{h:\Omega_{\boldsymbol{X}} \to \Omega_Y}{\arg\min} \quad \max_{\textit{tuple of SCMs } \mathbb{M}_0 \textit{ that entails } \mathbb{P} \textit{ \& induces } \mathcal{G}^{\mathbf{\Delta}}} R_{P^{\mathcal{M}_0^*}}(h).$$

Colored-MNIST

Hand-written digits: $W \in \mathbb{R}^{28\times28}$

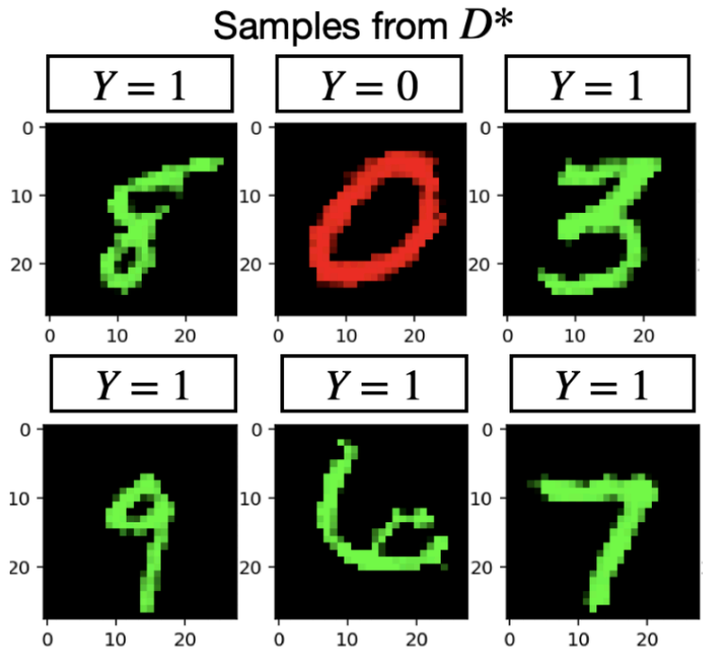
Labels: $Y \in \{0,1,...,9\}$

Color: $C \in \{\text{red}, \text{green}\}$



Colored digit: $Z \in \mathbb{R}^{28\times28\times3}$

Task. We have access to data from two source domains; find a classifier $\hat{y} = h(Z)$ with the smallest worst-case risk.



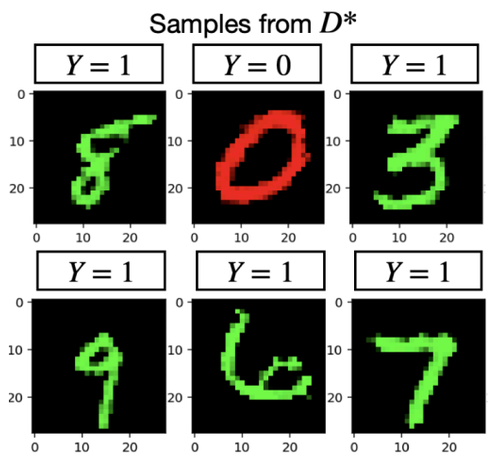
Colored-MNIST

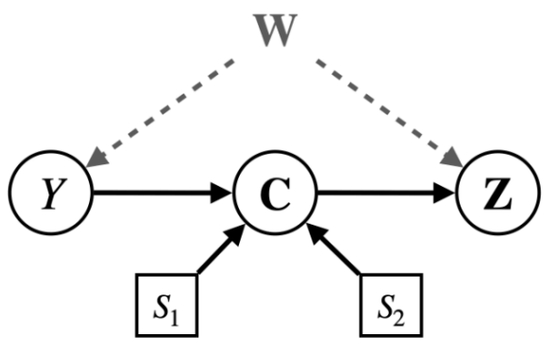
Hand-written digits: $W \in \mathbb{R}^{28\times28}$

Labels: $Y \in \{0,1,...,9\}$

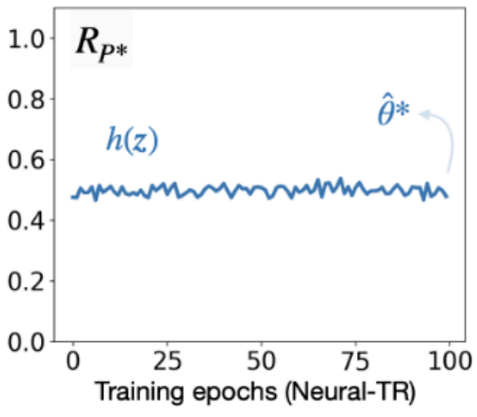
Color: $C \in \{\text{red}, \text{green}\}$

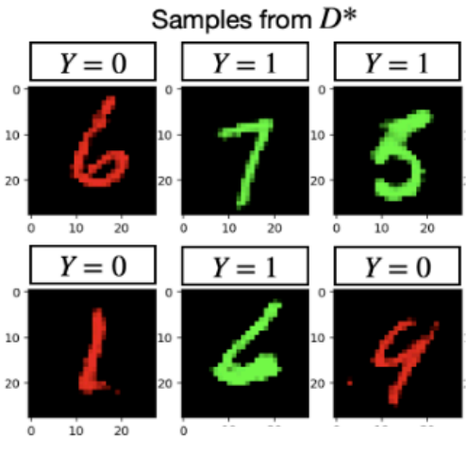
Colored digit: $Z \in \mathbb{R}^{28\times28\times3}$



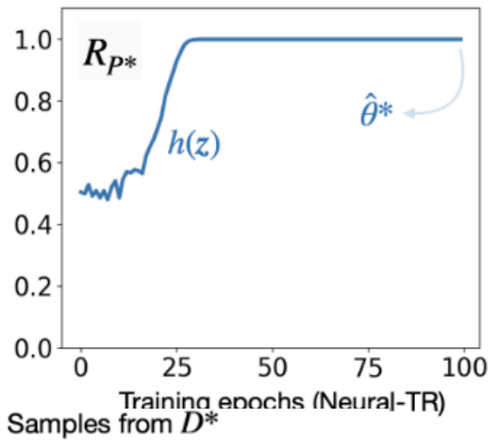


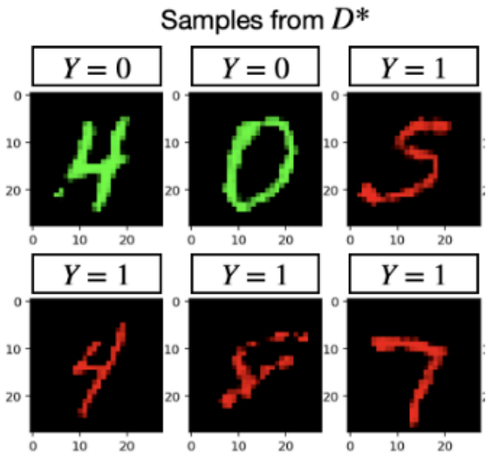
Iteration 1



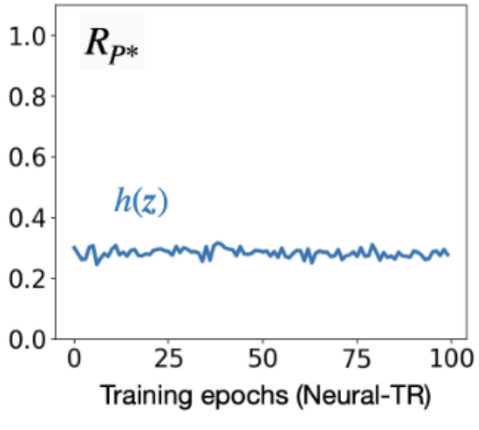


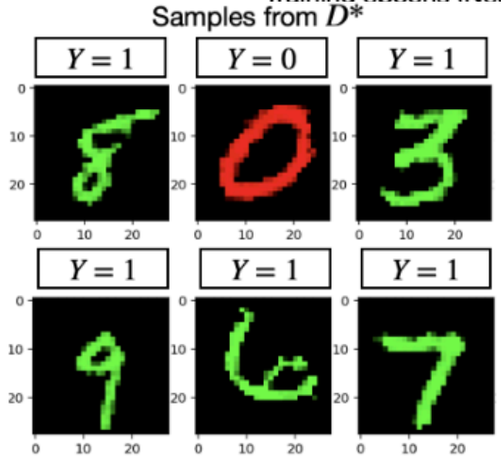
Iteration 2





Iteration 3





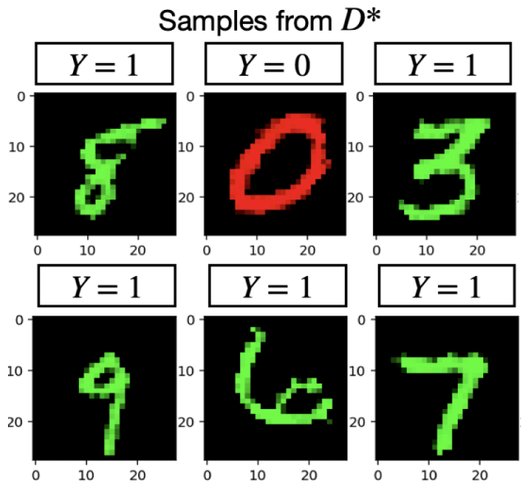
Colored-MNIST

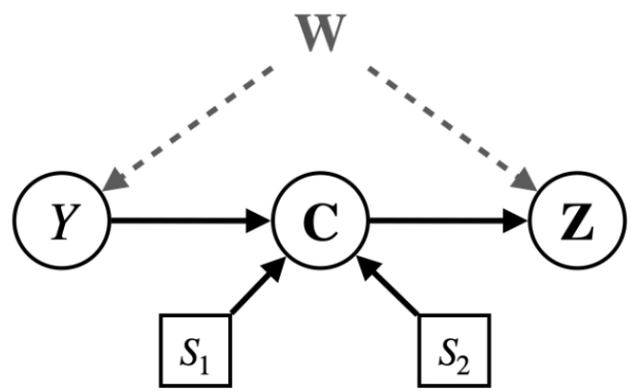
Hand-written digits: $W \in \mathbb{R}^{28\times28}$

Labels: $Y \in \{0,1,...,9\}$

Color: $C \in \{\text{red}, \text{green}\}$

Colored digit: $Z \in \mathbb{R}^{28\times28\times3}$

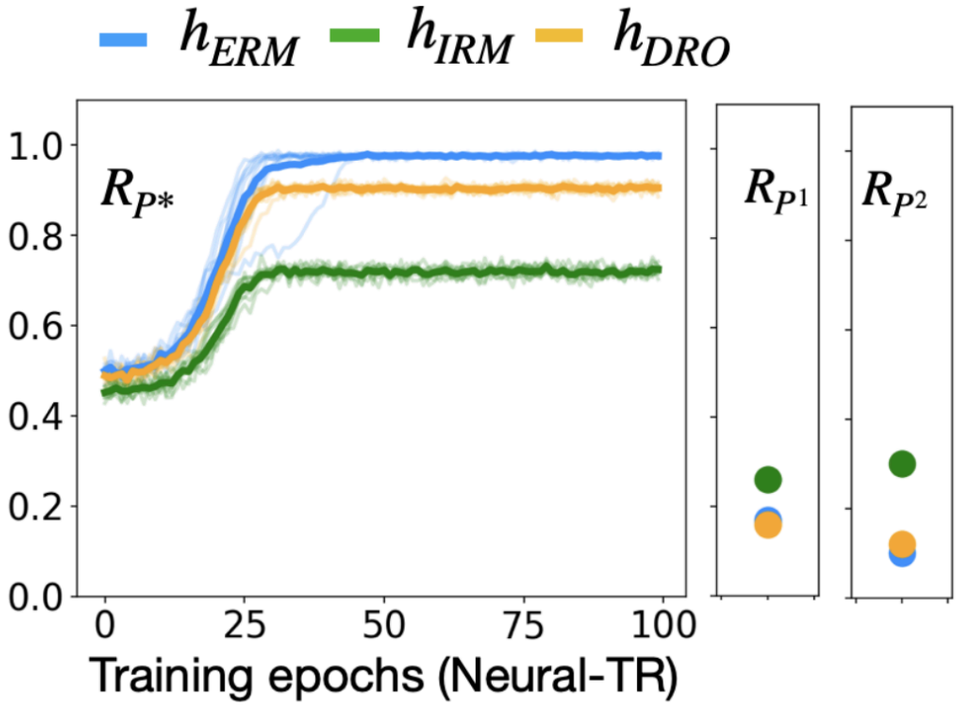




ERM: pool data from both domains and regress; baseline.

IRM: The invariant risk minimization; state-of-the-art.

DRO: Distributionally Robust Optimziation; state-of-the-art.



Conclusions

- Even non-transportable queries can be transported via SCM parametrization, canonical, or neural.
- Canonical parametrization is accurate and sound, though the complexity makes it intractable in practice.
- Neural-TR parametrizes only the components that are not transportable, keeping the parameter space small.
- Causal Robust Optimization (CRO) searches over the space of classifiers, iteratively improving the one at hand using Neural-TR as a subroutine.
- With graphical assumptions, finding the best worst-case classifier within a hypothesis class is now a well-defined and solvable problem using the CRO algorithm, contingent on having large computational resources.