Multi-level Decision-Making with Causal Bandits

Fabio Massimo Zennaro

University of Bergen

Paper

This work is based on the paper:

Causally Abstracted Multi-armed Bandits

Fabio Massimo Zennaro[1] **Nicholas Bishop**[2] **Joel Dyer**[2] **Yorgos Felekis**[3] **Anisoara Calinescu**[2]
Michael Wooldridge[2] **Theodoros Damoulas**[3]

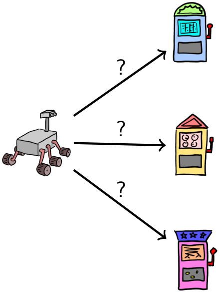
[1]University of Bergen
[2]University of Oxford
[3]University of Warwick

The presentation is based on the presentation given at UAI by Nick.

1. Multi-Armed Bandits

Multi-armed bandits (MABs) - Idea

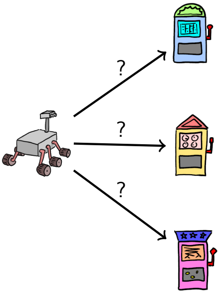
MABs represent simple standard **decision-making problems**:



(Art by Troels A. Bojesen)

Multi-armed bandits (MABs) - Idea

MABs represent simple standard **decision-making problems**:

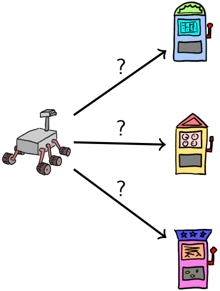


(Art by Troels A. Bojesen)

How to choose between options of unknown value?

Multi-armed bandits (MABs) - Idea

MABs represent simple standard **decision-making problems**:



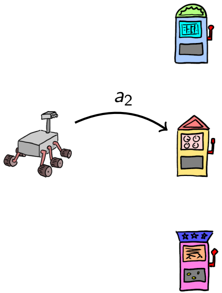
(Art by Troels A. Bojesen)

How to choose between options of unknown value?

✓ It models many real problems: *drug assessment*, *ads placement*, *policy making*...

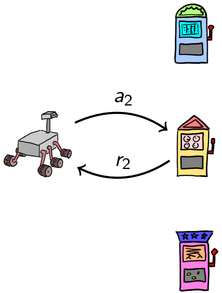
Multi-armed bandits (MABs) - Lifecycle

Multi-armed bandits (MABs) - Lifecycle



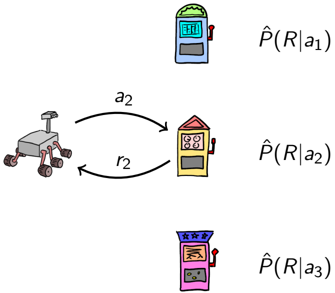
1. The agent chooses an *action/arm/lever* a_i;

Multi-armed bandits (MABs) - Lifecycle



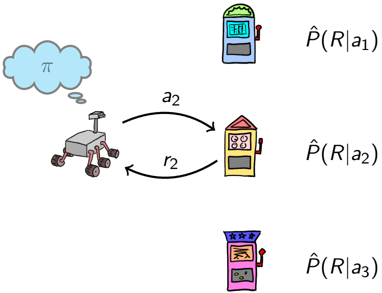
1. The agent chooses an *action/arm/lever* a_i;
2. It receives a *sample* r_i of a *rewards/payouts*;

Multi-armed bandits (MABs) - Lifecycle



1. The agent chooses an *action/arm/lever* a_i;
2. It receives a *sample* r_i of a *rewards/payouts*;
3. It updates an *estimate* $P(R|a_i)$ of the value of action a_i;

Multi-armed bandits (MABs) - Lifecycle



1. The agent chooses an *action/arm/lever* a_i;
2. It receives a *sample* r_i of a *rewards/payouts*;
3. It updates an *estimate* $P(R|a_i)$ of the value of action a_i;
4. It uses inform its own *policy* π

Multi-armed bandits (MABs) - Objective

A MAB agent tries to maximize:

Multi-armed bandits (MABs) - Objective

A MAB agent tries to maximize:

- **Simple Regret**: maximize the reward of the *next action*:

$$\bar{R}(t) = \mu^* - \mathbb{E}_\pi[\mu_{a_t}]$$

Multi-armed bandits (MABs) - Objective

A MAB agent tries to maximize:

- **Simple Regret**: maximize the reward of the *next action*:

$$\bar{R}(t) = \mu^* - \mathbb{E}_\pi[\mu_{a_t}]$$

- **Cumualtive Regret**: maximize the reward in the *horizon T*:

$$R(T) = T\mu^* - \sum_{t=1}^{T} \mathbb{E}_\pi[\mu_{a_t}]$$

Multi-armed bandits (MABs) - Modelling

Maximizing regret requires balancing :

Multi-armed bandits (MABs) - Modelling

Maximizing regret requires balancing :

- **Exploitation**: take the action currently estimated the best;

Multi-armed bandits (MABs) - Modelling

Maximizing regret requires balancing :

- **Exploitation**: take the action currently estimated the best;
- **Exploration**: take action aimed at improving your estimates.

Multi-armed bandits (MABs) - Modelling

Maximizing regret requires balancing :

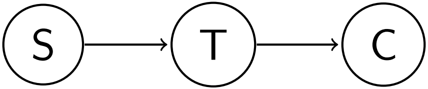
- **Exploitation**: take the action currently estimated the best;
- **Exploration**: take action aimed at improving your estimates.

There exist many algorithms: *UCB*, *epsilon*, *Thompson*...

2. Causal Models

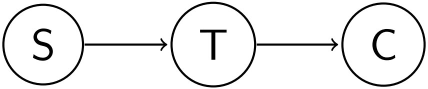
Structural Causal Models (SCMs) - Idea

SCMs represent **causal systems**.



Structural Causal Models (SCMs) - Idea

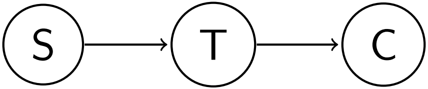
SCMs represent **causal systems**.



SCMs integrates a *graphical model* and *probabilities distributions*.

Structural Causal Models (SCMs) - Idea

SCMs represent **causal systems**.

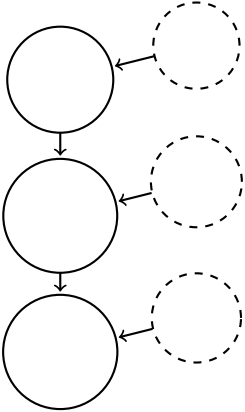


SCMs integrates a *graphical model* and *probabilities distributions*.

✓ It allows us to reason *causally* beyond pure statistic-correlation.

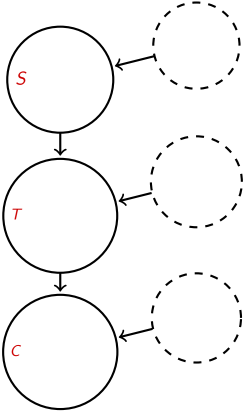
Structural Causal Models (SCMs) - Definition

We express a **SCM** as $\mathcal{M} = \langle \mathcal{X}, \mathcal{U}, \mathcal{F}, \mathcal{P} \rangle$ [6, 7]:



Structural Causal Models (SCMs) - Definition

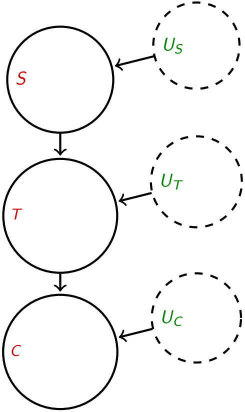
We express a **SCM** as $\mathcal{M} = \langle \mathcal{X}, \mathcal{U}, \mathcal{F}, \mathcal{P} \rangle$ [6, 7]:



- $\mathcal{X}$: set of *endogenous nodes* (S, T, C) representing variables of interest

Structural Causal Models (SCMs) - Definition

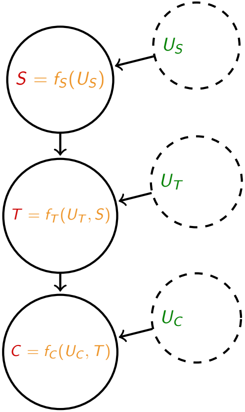
We express a **SCM** as $\mathcal{M} = \langle \mathcal{X}, \mathcal{U}, \mathcal{F}, \mathcal{P} \rangle$ [6, 7]:



- $\mathcal{X}$: set of *endogenous nodes* (S, T, C) representing variables of interest
- $\mathcal{U}$: Set of *exogenous nodes* (U_S, U_T, U_C) representing stochastic factors

Structural Causal Models (SCMs) - Definition

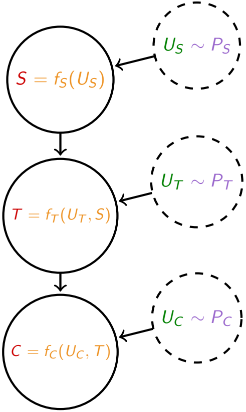
We express a **SCM** as $\mathcal{M} = \langle \mathcal{X}, \mathcal{U}, \mathcal{F}, \mathcal{P} \rangle$ [6, 7]:



- $\mathcal{X}$: set of *endogenous nodes* (S, T, C) representing variables of interest
- $\mathcal{U}$: Set of *exogenous nodes* (U_S, U_T, U_C) representing stochastic factors
- $\mathcal{F}$: Set of *structural functions* (f_S, f_T, f_C) describing the dynamics of each variable

Structural Causal Models (SCMs) - Definition

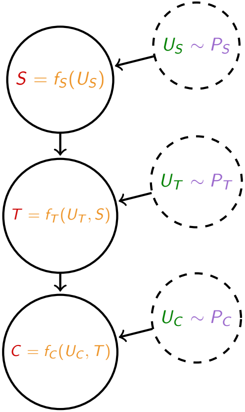
We express a **SCM** as $\mathcal{M} = \langle \mathcal{X}, \mathcal{U}, \mathcal{F}, \mathcal{P} \rangle$ [6, 7]:



- $\mathcal{X}$: set of *endogenous nodes* (S, T, C) representing variables of interest
- $\mathcal{U}$: Set of *exogenous nodes* (U_S, U_T, U_C) representing stochastic factors
- $\mathcal{F}$: Set of *structural functions* (f_S, f_T, f_C) describing the dynamics of each variable
- $\mathcal{P}$: Set of *distributions* (P_S, P_T, P_C) describing the random factors

Structural Causal Models (SCMs) - Definition

We express a **SCM** as $\mathcal{M} = \langle \mathcal{X}, \mathcal{U}, \mathcal{F}, \mathcal{P} \rangle$ [6, 7]:

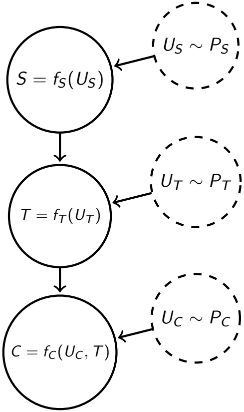


- $\mathcal{X}$: set of *endogenous nodes* (S, T, C) representing variables of interest
- $\mathcal{U}$: Set of *exogenous nodes* (U_S, U_T, U_C) representing stochastic factors
- $\mathcal{F}$: Set of *structural functions* (f_S, f_T, f_C) describing the dynamics of each variable
- $\mathcal{P}$: Set of *distributions* (P_S, P_T, P_C) describing the random factors

Every SCM $\mathcal{M}$ implies a (joint) **distribution** $P_{\mathcal{M}}$: $P_{\mathcal{M}}(S, T, C)$

Structural Causal Models (SCMs) - Interventions

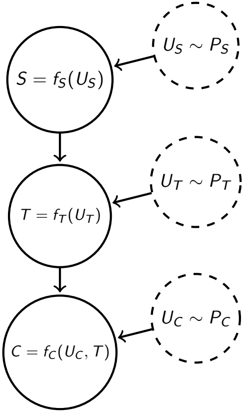
We can perform **interventions** on a causal model [6, 7]:



1.
2.

Structural Causal Models (SCMs) - Interventions

We can perform **interventions** on a causal model [6, 7]:

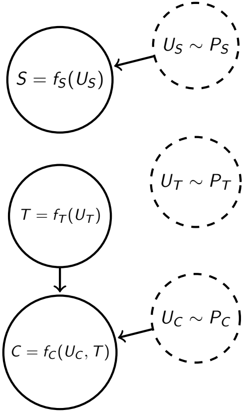


$do(T = 1)$

1.
2.

Structural Causal Models (SCMs) - Interventions

We can perform **interventions** on a causal model [6, 7]:

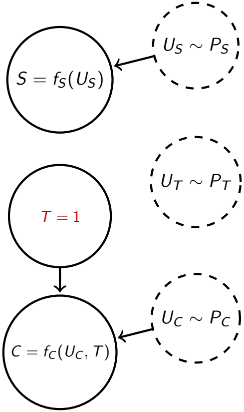


$do(T = 1)$

1. Remove incoming edges in the intervened node
2.

Structural Causal Models (SCMs) - Interventions

We can perform **interventions** on a causal model [6, 7]:



$do(T = 1)$

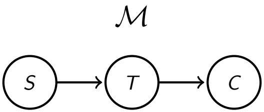
1. Remove incoming edges in the intervened node
2. Set the value of the intervened node

Structural Causal Models (SCMs) - Distributions

An *intervention* ι defines a new **intervened model** $\mathcal{M}_\iota$ with new distributions.

Structural Causal Models (SCMs) - Distributions

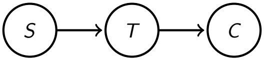
An *intervention* ι defines a new **intervened model** $\mathcal{M}_\iota$ with new distributions.



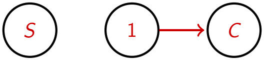
Structural Causal Models (SCMs) - Distributions

An *intervention* ι defines a new **intervened model** $\mathcal{M}_\iota$ with new distributions.

$\mathcal{M}$



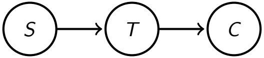
$\mathcal{M}_\iota$



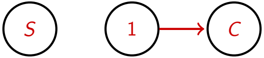
Structural Causal Models (SCMs) - Distributions

An *intervention* ι defines a new **intervened model** $\mathcal{M}_\iota$ with new distributions.

$\mathcal{M}$



$\mathcal{M}_\iota$

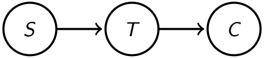


$P_{\mathcal{M}}$

Structural Causal Models (SCMs) - Distributions

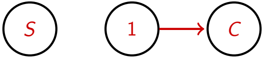
An *intervention* ι defines a new **intervened model** $\mathcal{M}_\iota$ with new distributions.

$\mathcal{M}$



$P_{\mathcal{M}}$

$\mathcal{M}_\iota$

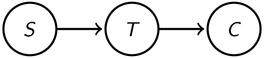


$P_{\mathcal{M}_\iota}$

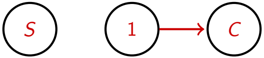
Structural Causal Models (SCMs) - Distributions

An *intervention* ι defines a new **intervened model** $\mathcal{M}_\iota$ with new distributions.

$\mathcal{M}$



$\mathcal{M}_\iota$



$P_\mathcal{M}$

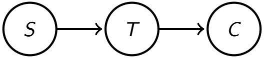
$P_{\mathcal{M}_\iota}$

$P_\mathcal{M}(C|S)$

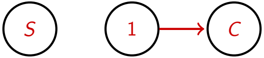
Structural Causal Models (SCMs) - Distributions

An *intervention* ι defines a new **intervened model** $\mathcal{M}_\iota$ with new distributions.

$\mathcal{M}$



$\mathcal{M}_\iota$



$P_{\mathcal{M}}$

$P_{\mathcal{M}_\iota}$

$P_{\mathcal{M}}(C|S)$

$P_{\mathcal{M}}(C|S, do(T = 1)) = P_{\mathcal{M}_\iota}(C|S)$

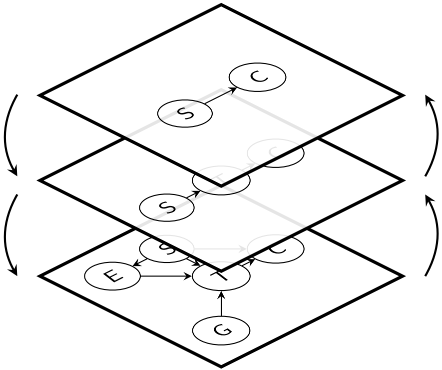
3. Causal Abstraction

Causal Abstractions (CAs) - Idea

Systems may be represented at different **levels of abstraction** (LoA) [3].

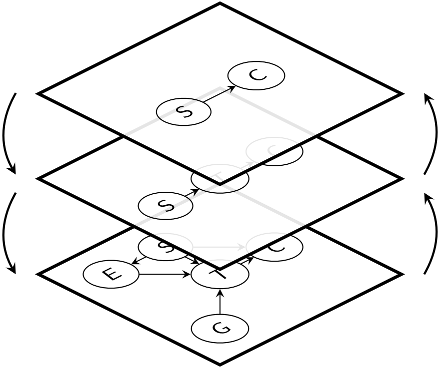
Causal Abstractions (CAs) - Idea

Systems may be represented at different **levels of abstraction** (LoA) [3].



Causal Abstractions (CAs) - Idea

Systems may be represented at different **levels of abstraction** (LoA) [3].



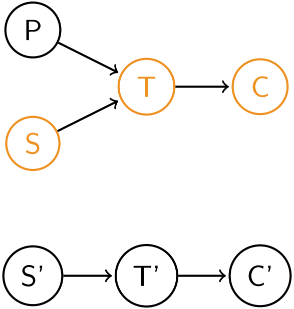
✓ With CA we want to work *simultaneously* at *multiple levels*, integrating data and saving computation.

Causal Abstractions (CAs) - Definition

An α-**abstraction** $\langle R, a, \alpha_i \rangle$ [9, 8] is defined as:

Causal Abstractions (CAs) - Definition

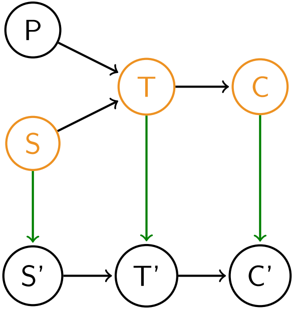
An **α-abstraction** $\langle R, a, \alpha_i \rangle$ [9, 8] is defined as:



- *R*: a set of *relevant variables*;

Causal Abstractions (CAs) - Definition

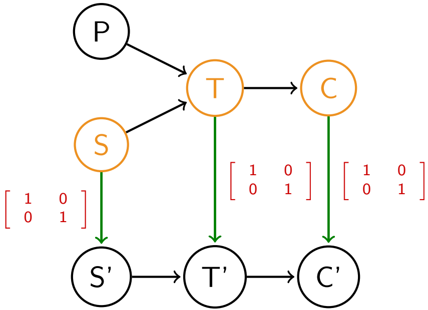
An **α-abstraction** $\langle R, a, \alpha_i \rangle$ [9, 8] is defined as:



- *R*: a set of *relevant variables*;
- *a*: a surjective function between *variables*;

Causal Abstractions (CAs) - Definition

An α-**abstraction** $\langle R, a, \alpha_i \rangle$ [9, 8] is defined as:



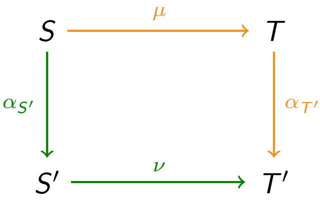
- *R*: a set of *relevant variables*;
- *a*: a surjective function between *variables*;
- α_i: a collection of surjective functions between *outcomes*.

Causal Abstractions (CAs) - Consistency

We want an abstraction to guarantee *interventional consistency*.

Causal Abstractions (CAs) - Consistency

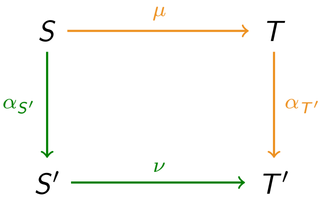
We want an abstraction to guarantee *interventional consistency*.



.

Causal Abstractions (CAs) - Consistency

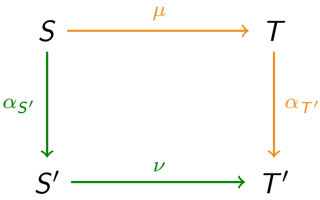
We want an abstraction to guarantee *interventional consistency*.



- Ideally, mechanisms and abstractions *commute*.

Causal Abstractions (CAs) - Consistency

We want an abstraction to guarantee *interventional consistency*.



- Ideally, mechanisms and abstractions *commute*.
- Otherwise, we compute an abstraction error as the *worst-case discrepancy* over all possible interventions:

$$E_{\boldsymbol{\alpha}}(S', T') = \max_{\iota} D(\alpha_{T'} \cdot \mu, \nu \cdot \alpha_{S'})$$

Causal Abstractions (CAs) - Abstraction Error

An abstraction implies multiple *causal mechanism diagrams*:

Causal Abstractions (CAs) - Abstraction Error

An abstraction implies multiple *causal mechanism diagrams*:

$$
\begin{array}{ccc}
S & \xrightarrow{\mu} & T \\
\downarrow{\scriptstyle \alpha_{S'}} & & \downarrow{\scriptstyle \alpha_{T'}} \\
S' & \xrightarrow{\nu} & T'
\end{array}
$$

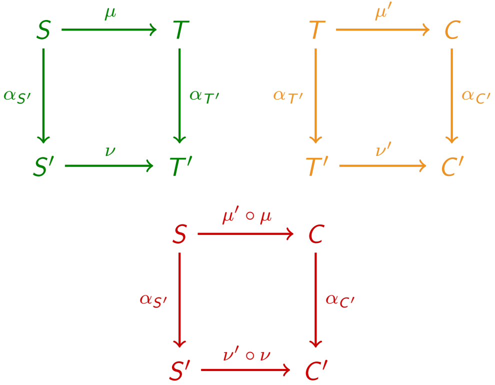
Causal Abstractions (CAs) - Abstraction Error

An abstraction implies multiple *causal mechanism diagrams*:

$$
\begin{array}{ccc}
S & \xrightarrow{\mu} & T \\
{\scriptstyle \alpha_{S'}}\downarrow & & \downarrow{\scriptstyle \alpha_{T'}} \\
S' & \xrightarrow{\nu} & T'
\end{array}
\qquad\qquad
\begin{array}{ccc}
T & \xrightarrow{\mu'} & C \\
{\scriptstyle \alpha_{T'}}\downarrow & & \downarrow{\scriptstyle \alpha_{C'}} \\
T' & \xrightarrow{\nu'} & C'
\end{array}
$$

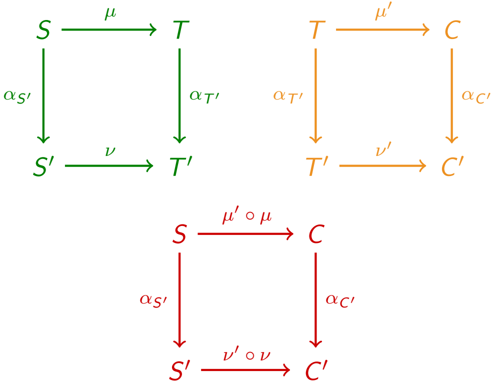
Causal Abstractions (CAs) - Abstraction Error

An abstraction implies multiple *causal mechanism diagrams*:



Causal Abstractions (CAs) - Abstraction Error

An abstraction implies multiple *causal mechanism diagrams*:

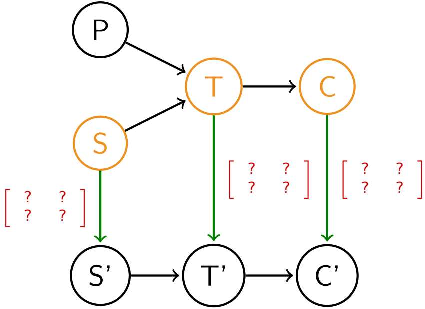


A **(global) abstraction error** [9] $e(\alpha)$ is the maximum abstraction error over all diagrams.

$$e(\boldsymbol{\alpha}) = \sup_{\mathbf{X}', \mathbf{Y}' \subseteq \mathcal{X}'} E_{\boldsymbol{\alpha}}(\mathbf{X}', \mathbf{Y}')$$

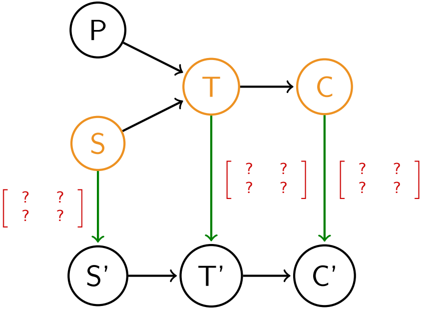
Causal Abstractions (CAs) - Learning

We may want to *learn* an abstraction α or part of it from data:



Causal Abstractions (CAs) - Learning

We may want to *learn* an abstraction $\boldsymbol{\alpha}$ or part of it from data:



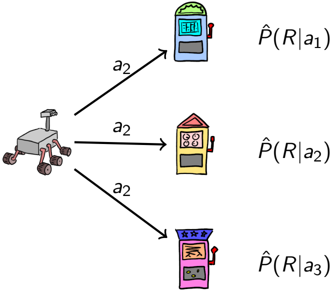
A number of methods [10, 2, 4] express the learning problem as a **minimization of abstraction error**:

$$\min_{\boldsymbol{\alpha}} \; e(\boldsymbol{\alpha})$$

4. Causal Bandits

Multi-armed bandits (MABs)

In standard MABs, all the outcomes are **independent**.

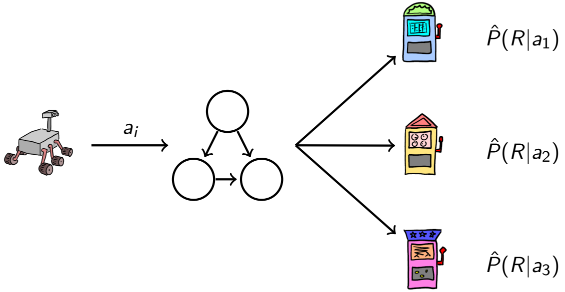


Causal multi-armed bandits (CMABs) - Idea

In a CMAB a causal model **mediates** the outcomes.

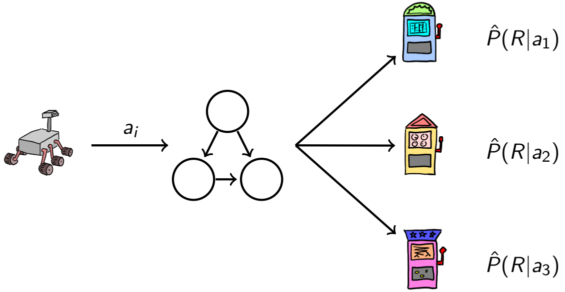
Causal multi-armed bandits (CMABs) - Idea

In a CMAB a causal model **mediates** the outcomes.



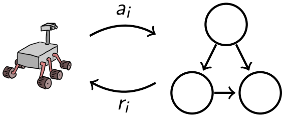
Causal multi-armed bandits (CMABs) - Idea

In a CMAB a causal model **mediates** the outcomes.

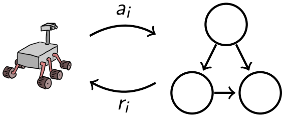


✓ A CMAB represent a *more realistic* setting where we can *relate* and *reason* about actions.

Causal multi-armed bandits (CMABs) - Terminology

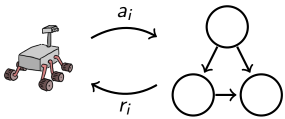


Causal multi-armed bandits (CMABs) - Terminology



1. *Actions* a_i are *interventions*;

Causal multi-armed bandits (CMABs) - Terminology



1. *Actions* a_i are *interventions*;
2. *Reward* r_i are *causal effects*.

CMAB algortihms take advantage of causal structure [5, 1].

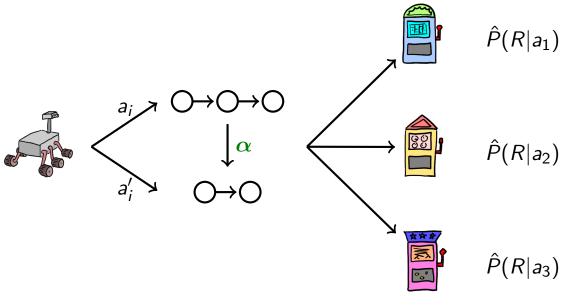
5. Causally Abstracted Bandits

Causally abstracted multi-armed bandits (CAMABs) - Idea

In a CAMAB, an agent has **multiple causal models**.

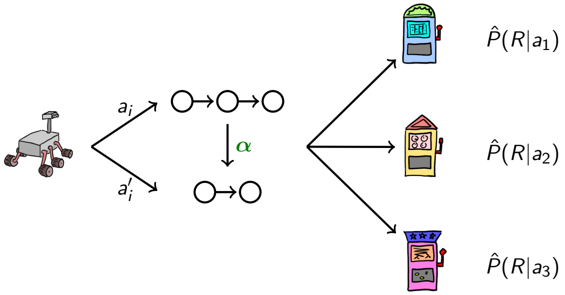
Causally abstracted multi-armed bandits (CAMABs) - Idea

In a CAMAB, an agent has **multiple causal models**.



Causally abstracted multi-armed bandits (CAMABs) - Idea

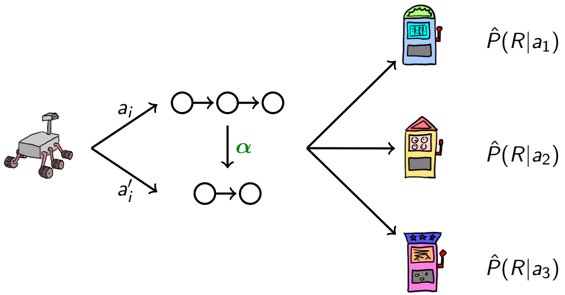
In a CAMAB, an agent has **multiple causal models**.



✓ A CAMAB capture a setting where *multiple actors* tackle the same problem at different levels of abstraction.

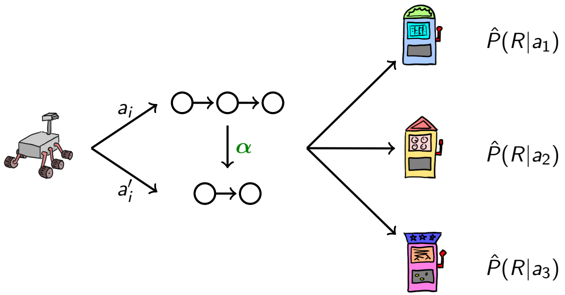
Causally abstracted multi-armed bandits (CAMABs) - Problem

How do we take advantage of α?



Causally abstracted multi-armed bandits (CAMABs) - Problem

How do we take advantage of α?

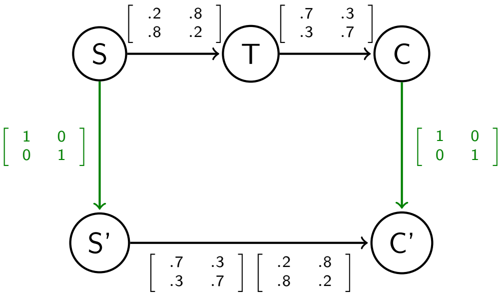


We will consider some approaches inspired by *reinforcement learning*.

6. Some CAMAB Results

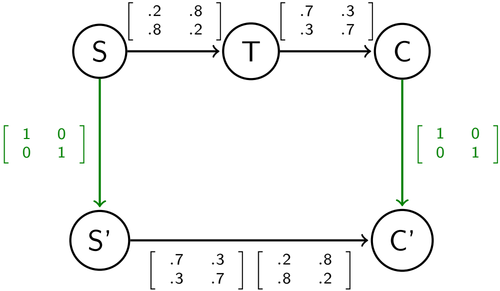
CAMAB - Transporting Optimal Action

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:



CAMAB - Transporting Optimal Action

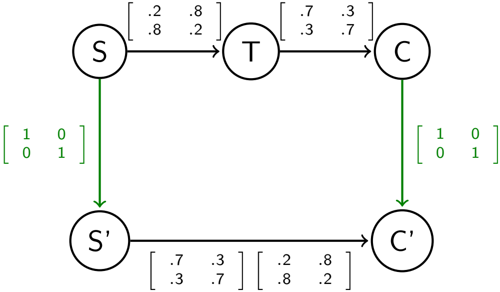
Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:



Let us assume:

CAMAB - Transporting Optimal Action

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:

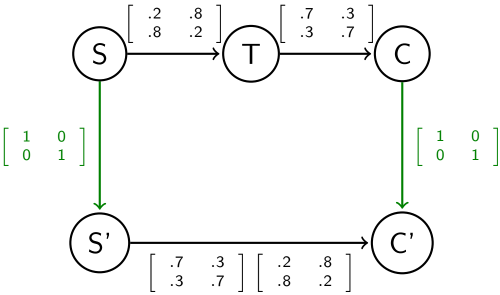


Let us assume:

- An abstraction α with zero error;

CAMAB - Transporting Optimal Action

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:

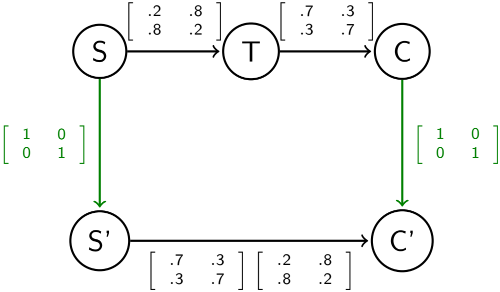


Let us assume:

- An abstraction α with zero error;
- An *optimal action* a^* in $\mathcal{M}$.

CAMAB - Transporting Optimal Action

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:



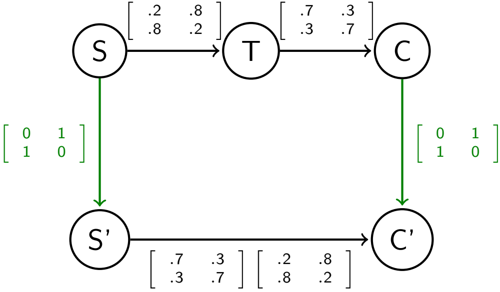
Let us assume:

- An abstraction α with zero error;
- An *optimal action* a^* in $\mathcal{M}$.

Does it hold that: $a'^* = \alpha(a^*)$?

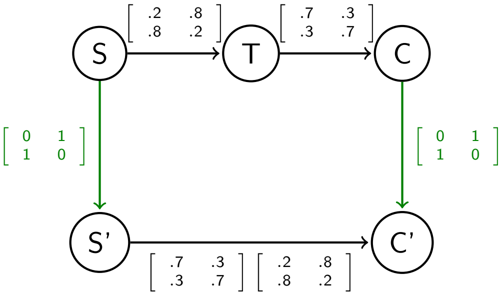
CAMAB - Transporting Optimal Action

It does **NOT**:



CAMAB - Transporting Optimal Action

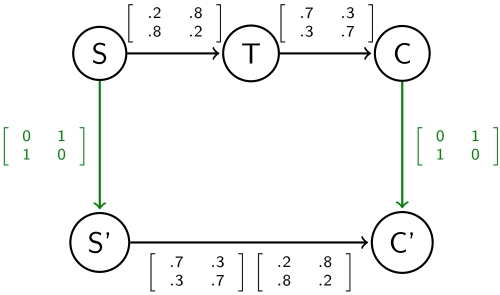
It does **NOT**:



Optimality may not be preserved:

CAMAB - Transporting Optimal Action

It does **NOT**:

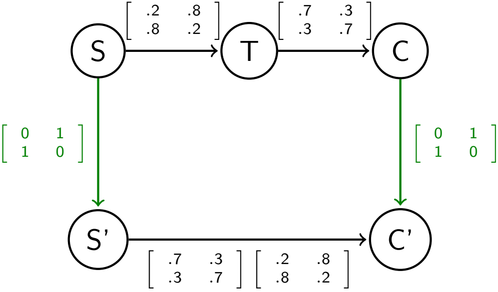


Optimality may not be preserved:

- If actions and outcomes are *consistently* flipped.

CAMAB - Transporting Optimal Action

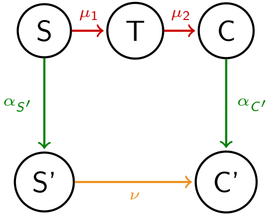
It does **NOT**:



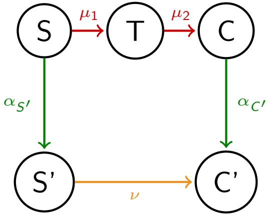
Optimality may not be preserved:

- If actions and outcomes are *consistently* flipped.
- (If the domains of the outcomes are different).

CAMAB - Reward Discrepancy



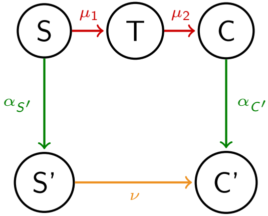
CAMAB - Reward Discrepancy



If we want to study CAMABs *abstraction error* is not enough:

$$e(\boldsymbol{\alpha}) = \sup_{\mathbf{X}', \mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\alpha_{C'} \cdot \mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

CAMAB - Reward Discrepancy



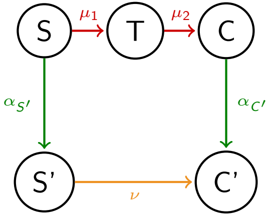
If we want to study CAMABs *abstraction error* is not enough:

$$e(\boldsymbol{\alpha}) = \sup_{\mathbf{X}', \mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\alpha_{C'} \cdot \mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

We want to consider also **reward discrepancy**:

$$s(\boldsymbol{\alpha}) = \sup_{\mathbf{X}', \mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

CAMAB - Reward Discrepancy



If we want to study CAMABs *abstraction error* is not enough:

$$e(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\alpha_{C'} \cdot \mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

We want to consider also **reward discrepancy**:

$$s(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

(Assuming same dimension of the domains of C and C')

CAMAB - Triangular Inequality

Abstract error:

$$e(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\alpha_{C'} \cdot \mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

CAMAB - Triangular Inequality

Abstract error:

$$e(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\alpha_{C'} \cdot \mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

Reward discrepancy:

$$s(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

CAMAB - Triangular Inequality

Abstract error:

$$e(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\alpha_{C'} \cdot \mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

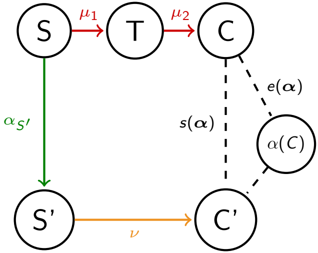
Reward discrepancy:

$$s(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

This immediately gives us a **triangular inequality**:

$$|\mu_{a'} - \mu_{\boldsymbol{\alpha}(a)}| \leq e(\boldsymbol{\alpha}) + s(\boldsymbol{\alpha})$$

CAMAB - Triangular Inequality



Abstraction error:

$$e(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\alpha_{C'} \cdot \mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

Reward discrepancy:

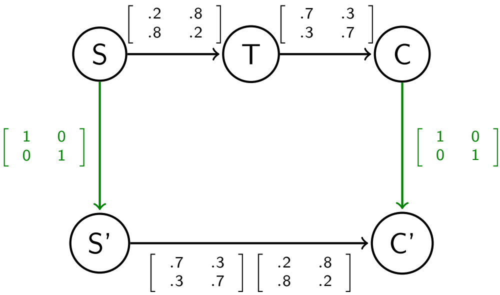
$$s(\boldsymbol{\alpha}) = \sup_{\mathbf{X}',\mathbf{Y}' \subseteq \mathcal{X}'} \max_{\iota} D(\mu_2 \cdot \mu_1, \nu \cdot \alpha_{S'})$$

This immediately gives us a **triangular inequality**:

$$|\mu_{a'} - \mu_{\boldsymbol{\alpha}(a)}| \leq e(\boldsymbol{\alpha}) + s(\boldsymbol{\alpha})$$

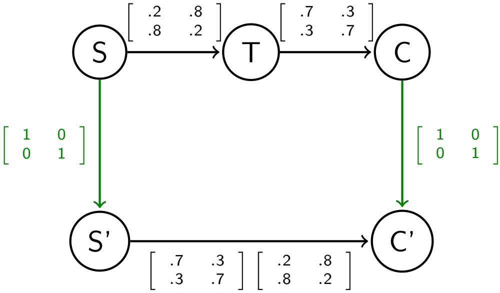
CAMAB - Transporting Actions

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:



CAMAB - Transporting Actions

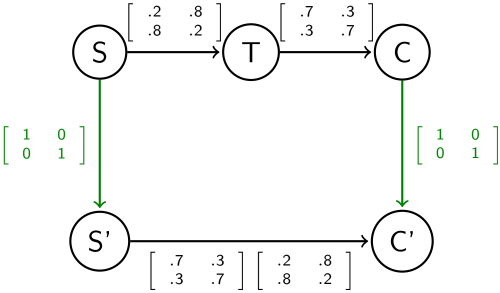
Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:



Let us assume:

CAMAB - Transporting Actions

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:

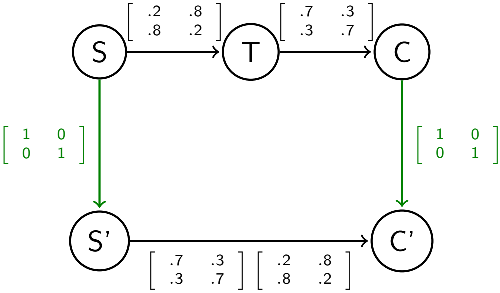


Let us assume:

- We have the collection of all the action $a^{(t)}$ taken on $\mathcal{M}$.

CAMAB - Transporting Actions

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:

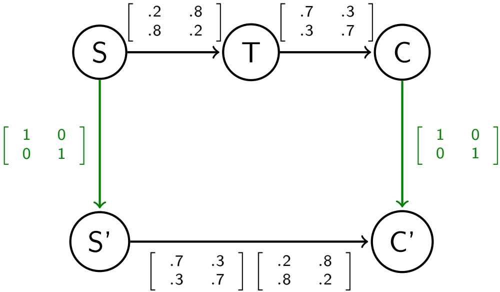


Let us assume:

- We have the collection of all the action $a^{(t)}$ taken on $\mathcal{M}$.
- We have *optimality preservation*.

CAMAB - Transporting Actions

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:



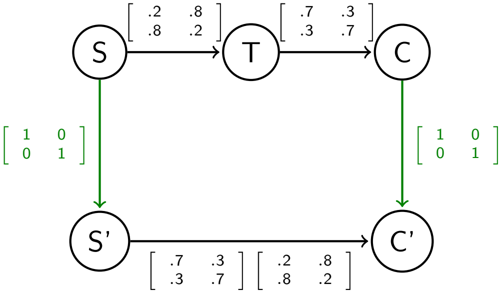
Let us assume:

- We have the collection of all the action $a^{(t)}$ taken on $\mathcal{M}$.
- We have *optimality preservation*.

Can I earn anything by *imitation*, that is playing: $a'^{(t)} = \alpha(a^{(t)})$?

CAMAB - Transporting Actions

Let us consider a *CAMAB* made up by two CMABs $\mathcal{M}$ and $\mathcal{M}'$:



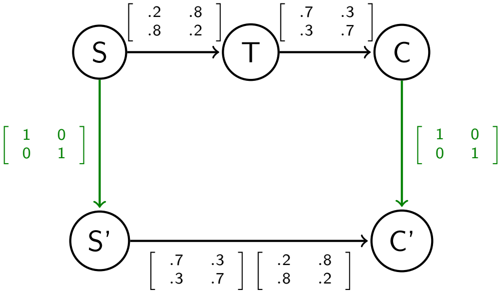
Let us assume:

- We have the collection of all the action $a^{(t)}$ taken on $\mathcal{M}$.
- We have *optimality preservation*.

Can I earn anything by *imitation*, that is playing: $a'^{(t)} = \alpha(a^{(t)})$?
If so, when?

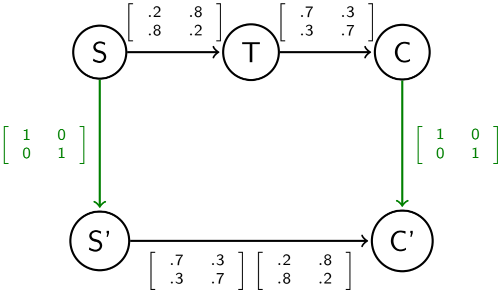
CAMAB - Transporting Actions

Let us refine our assumptions further:



CAMAB - Transporting Actions

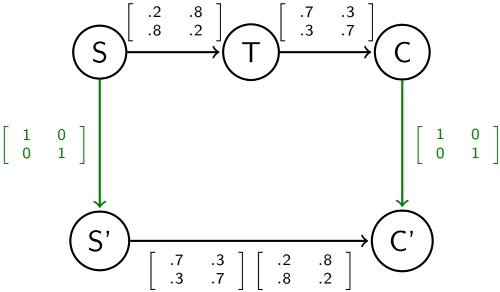
Let us refine our assumptions further:



Let us assume:

CAMAB - Transporting Actions

Let us refine our assumptions further:

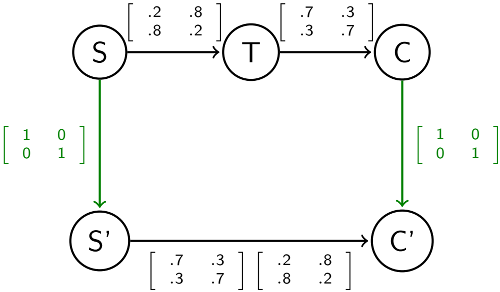


Let us assume:

- We have run the *UCB* algorithm on $\mathcal{M}$ for T steps.

CAMAB - Transporting Actions

Let us refine our assumptions further:



Let us assume:

- We have run the *UCB* algorithm on $\mathcal{M}$ for T steps.

When is it that the *imitation* algorithm on $\mathcal{M}'$ performs better than *UCB* on $\mathcal{M}'$?

CAMAB - Transporting Actions

The *imitation* protocol has a lower regret bound than UCB if:

$$\underbrace{3 \sum_{a' \in \mathcal{A}'} \Delta(a') \left[1 - \mathcal{K}(a')\right]}_{\text{fixed cost with possible oversampling arms}} \quad + \underbrace{16 \log T \sum_{a' \in \mathcal{A}'} \left[\frac{\Delta(a')}{\Delta(a')^2} - \sum_{a \in \mathcal{A} | \alpha(a) = a'} \frac{\Delta(a')}{\Delta(a)^2} \right]}_{\text{variable cost driven by the base model}} \geq 0$$

CAMAB - Transporting Actions

The *imitation* protocol has a lower regret bound than UCB if:

$$\underbrace{3\sum_{a'\in\mathcal{A}'}\Delta(a')\left[1-\mathcal{K}(a')\right]}_{\text{fixed cost with possible oversampling arms}} \quad +16\log T\underbrace{\sum_{a'\in\mathcal{A}'}\left[\frac{\Delta(a')}{\Delta(a')^2}-\sum_{a\in\mathcal{A}|\alpha(a)=a'}\frac{\Delta(a')}{\Delta(a)^2}\right]}_{\text{variable cost driven by the base model}}\geq 0$$

- $\mathcal{K}(a')$ gives us the *number of base actions* a mapping to a'.
- $\Delta(a)$ is the *optimality gap* for action a.

CAMAB - Transporting Actions

The *imitation* protocol has a lower regret bound than UCB if:

$$\underbrace{3 \sum_{a' \in \mathcal{A}'} \Delta(a') \left[1 - \mathcal{K}(a')\right]}_{\text{fixed cost with possible oversampling arms}} \quad + 16 \log T \underbrace{\sum_{a' \in \mathcal{A}'} \left[\frac{\Delta(a')}{\Delta(a')^2} - \sum_{a \in \mathcal{A} | \alpha(a) = a'} \frac{\Delta(a')}{\Delta(a)^2} \right]}_{\text{variable cost driven by the base model}} \geq 0$$

- $\mathcal{K}(a')$ gives us the *number of base actions* a mapping to a'.
- $\Delta(a)$ is the *optimality gap* for action a.
- Bound derived from results on UCB:

CAMAB - Transporting Actions

The *imitation* protocol has a lower regret bound than UCB if:

$$\underbrace{3 \sum_{a' \in \mathcal{A}'} \Delta(a') \left[1 - \mathcal{K}(a')\right]}_{\text{fixed cost with possible oversampling arms}} \quad + 16 \log T \underbrace{\sum_{a' \in \mathcal{A}'} \left[\frac{\Delta(a')}{\Delta(a')^2} - \sum_{a \in \mathcal{A} | \alpha(a) = a'} \frac{\Delta(a')}{\Delta(a)^2} \right]}_{\text{variable cost driven by the base model}} \geq 0$$

- $\mathcal{K}(a')$ gives us the *number of base actions* a mapping to a'.
- $\Delta(a)$ is the *optimality gap* for action a.
- Bound derived from results on UCB:
  - Fixed cost of sampling all actions;

CAMAB - Transporting Actions

The *imitation* protocol has a lower regret bound than UCB if:

$$\underbrace{3 \sum_{a' \in \mathcal{A}'} \Delta(a') \left[1 - \mathcal{K}(a')\right]}_{\text{fixed cost with possible oversampling arms}} \quad + \underbrace{16 \log T \sum_{a' \in \mathcal{A}'} \left[\frac{\Delta(a')}{\Delta(a')^2} - \sum_{a \in \mathcal{A} | \alpha(a) = a'} \frac{\Delta(a')}{\Delta(a)^2} \right]}_{\text{variable cost driven by the base model}} \geq 0$$

- $\mathcal{K}(a')$ gives us the *number of base actions* a mapping to a'.
- $\Delta(a)$ is the *optimality gap* for action a.
- Bound derived from results on UCB:
  - Fixed cost of sampling all actions;
    - If many actions a are mapped to the same a' you will oversample a'

CAMAB - Transporting Actions

The *imitation* protocol has a lower regret bound than UCB if:

$$\underbrace{3\sum_{a'\in\mathcal{A}'}\Delta(a')\left[1-\mathcal{K}(a')\right]}_{\text{fixed cost with possible oversampling arms}} \quad +16\log T\underbrace{\sum_{a'\in\mathcal{A}'}\left[\frac{\Delta(a')}{\Delta(a')^2}-\sum_{a\in\mathcal{A}|\alpha(a)=a'}\frac{\Delta(a')}{\Delta(a)^2}\right]}_{\text{variable cost driven by the base model}}\geq 0$$

- $\mathcal{K}(a')$ gives us the *number of base actions* a mapping to a'.
- $\Delta(a)$ is the *optimality gap* for action a.
- Bound derived from results on UCB:
  - Fixed cost of sampling all actions;
    - If many actions a are mapped to the same a' you will oversample a'
  - Variables cost to achieve a level of confidence;

CAMAB - Transporting Actions

The *imitation* protocol has a lower regret bound than UCB if:

$$\underbrace{3\sum_{a'\in\mathcal{A}'}\Delta(a')\left[1-\mathcal{K}(a')\right]}_{\text{fixed cost with possible oversampling arms}} \quad +16\log T \underbrace{\sum_{a'\in\mathcal{A}'}\left[\frac{\Delta(a')}{\Delta(a')^2}-\sum_{a\in\mathcal{A}|\alpha(a)=a'}\frac{\Delta(a')}{\Delta(a)^2}\right]}_{\text{variable cost driven by the base model}} \geq 0$$

- $\mathcal{K}(a')$ gives us the *number of base actions* a mapping to a'.
- $\Delta(a)$ is the *optimality gap* for action a.
- Bound derived from results on UCB:
  - Fixed cost of sampling all actions;
    - If many actions a are mapped to the same a' you will oversample a'
  - Variables cost to achieve a level of confidence;
    - If action a has big optimality gap, it will make the corresponding action a' oversampled.

CAMAB - Transporting Actions

The *imitation* protocol has a lower regret bound than UCB if:

$$\underbrace{3 \sum_{a' \in \mathcal{A}'} \Delta(a') \left[1 - \mathcal{K}(a')\right]}_{\text{fixed cost with possible oversampling arms}} \quad + \underbrace{16 \log T \sum_{a' \in \mathcal{A}'} \left[\frac{\Delta(a')}{\Delta(a')^2} - \sum_{a \in \mathcal{A} | \alpha(a) = a'} \frac{\Delta(a')}{\Delta(a)^2} \right]}_{\text{variable cost driven by the base model}} \geq 0$$

- $\mathcal{K}(a')$ gives us the *number of base actions* a mapping to a'.
- $\Delta(a)$ is the *optimality gap* for action a.
- Bound derived from results on UCB:
  - Fixed cost of sampling all actions;
    - If many actions a are mapped to the same a' you will oversample a'
  - Variables cost to achieve a level of confidence;
    - If action a has big optimality gap, it will make the corresponding action a' oversampled.
- Ideally, optimal action a^* and a number of actions with small gap $\Delta(a)$ maps to the optimal a'^*

7. Conclusion

Conclusion

We have seen some ideas and results, but the paper analyzes more closely:

- *Transfer of optimal actions*;
- *Transfer of actions*;
- *Transfer of expected outcomes*.

and also provides *sample application*.

MAB is an established area, but wide space in

- *CMABs*
- *CAMABs*

Thanks!

Thank you for listening!

References I

[1] Elias Bareinboim, Andrew Forney, and Judea Pearl. Bandits with unobserved confounders: A causal approach. *Advances in Neural Information Processing Systems*, 28:1342–1350, 2015.

[2] Yorgos Felekis, Fabio Massimo Zennaro, Nicola Branchini, and Theodoros Damoulas. Causal optimal transport of abstractions. *arXiv preprint arXiv:2312.08107*, 2023.

[3] Luciano Floridi. The method of levels of abstraction. *Minds and machines*, 18:303–329, 2008.

[4] Armin Kekić, Bernhard Schölkopf, and Michel Besserve. Targeted reduction of causal models. *arXiv preprint arXiv:2311.18639*, 2023.

[5] Finnian Lattimore, Tor Lattimore, and Mark D Reid. Causal bandits: Learning good interventions via causal inference. In *Advances in Neural Information Processing Systems*, pages 1181–1189, 2016.

[6] Judea Pearl. *Causality*. Cambridge University Press, 2009.

References II

[7] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: Foundations and learning algorithms*. MIT Press, 2017.

[8] Eigil F Rischel and Sebastian Weichwald. Compositional abstraction error and a category of causal models. *arXiv preprint arXiv:2103.15758*, 2021.

[9] Eigil Fjeldgren Rischel. The category theory of causal models. Master's thesis, University of Copenhagen, 2020.

[10] Fabio Massimo Zennaro, Máté Drávucz, Geanina Apachitei, W. Dhammika Widanage, and Theodoros Damoulas. Jointly learning consistent causal abstractions over multiple interventional distributions. In *2nd Conference on Causal Learning and Reasoning*, 2023.