Policy learning under constraint: Maximizing a primary outcome while controlling an adverse event

Laura Fuentes-Vicente[1], Mathieu Even[1], Gaëlle Dormion[2], Julie Josse[1], Antoine Chambaz[3]

1 Inria PreMeDICaL, Inserm, University of Montpellier, France
2 Elixir Health, Paris, France
3 Paris Cité University, CNRS, MAP5, F-75006 Paris, France

April 15 2026, EuroCIM, Oxford

Introduction

Medical motivation

Classical policy learning: Given patient's characteristics, determine the optimal treatment maximizing each patient's outcome

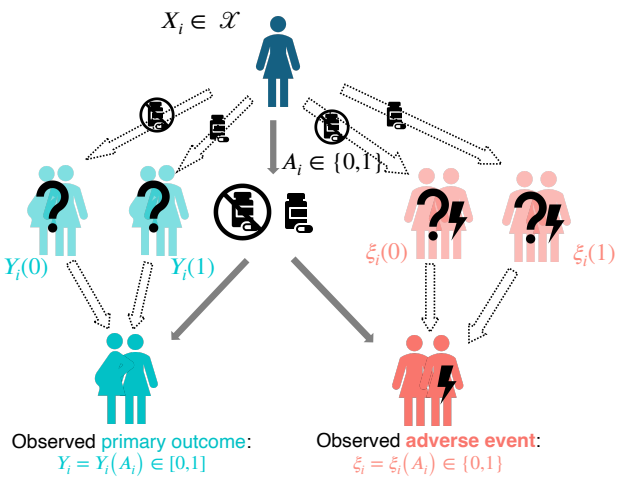
IVF example: Find the optimal hormone dose to maximize the number of oocyte produced

Our goal: Given patient's characteristics, determine the optimal treatment maximizing each patient's outcome while controlling an adverse event

IVF example: Find the optimal hormone dose to maximize the number of oocyte produced while controlling the probability of suffering from ovarian hyperstimulation

Statistical modeling

- Set of n i.i.d. observations $O_1, \ldots, O_n \sim P$
- Generic data structure $O = (X, A, Y, \xi)$:
  - $X \in \mathcal{X}$: vector of covariates
  - $A \in \{0, 1\}$: treatment assignment
  - $Y \in [0, 1]$: primary outcome
  - $\xi \in \{0, 1\}$: adverse event
- Relevant nuisances:
  $\mu_P(a, X) = E_P[Y|A = a, X]$,
  $\nu_P(a, X) = E_P[\xi|A = a, X]$



Policies and their values

A policy $\widetilde{\pi} \in \Pi \subset [0,1]^{\mathcal{X}}$ maps any $x \in \mathcal{X}$ to a treatment assignment probability $\widetilde{\pi}(x)$

The value of a policy $\widetilde{\pi} \in \Pi$ under P is defined as

$$\mathcal{V}_P(\widetilde{\pi}) = E_P[\widetilde{\pi}(X) \cdot \mu_P(1, X) + (1 - \widetilde{\pi}(X)) \cdot \mu_P(0, X)]$$

Define $\Delta\mu_P(\cdot) = \mu_P(1, \cdot) - \mu_P(0, \cdot) : \mathcal{X} \to [-1, 1]$
It holds that

$$\operatorname*{argmax}_{\widetilde{\pi} \in \Pi} \mathcal{V}_P(\widetilde{\pi}) = \operatorname*{argmax}_{\widetilde{\pi} \in \Pi} E_P[\widetilde{\pi}(X) \cdot \Delta\mu_P(X)]$$

Direct and indirect policy learning (1/2)

Learning a value-optimal policy can be performed directly or indirectly, using the fact that

$$\pi_P : x \mapsto \mathbf{1}\{\Delta\mu_P(x) > 0\}$$

maximizes $\widetilde{\pi} \mapsto \mathcal{V}_P(\widetilde{\pi})$

- **Direct**: classification task, aims to predict $\mathbf{1}\{\Delta\mu_P(X) > 0\}$ given X
  e.g. OWL[1], RWL[2], etc.
- **Indirect**: estimation of the heterogeneous treatment effect $\Delta\mu_P$ to create a plug-in estimator of π_P
  e.g. X-learner[3], DR-learner[4], EP-learner [5], etc.

[1] Zhao et al., 2012
[2] Zhou et al., 2017
[3] Künzel et al., 2017
[4] Kennedy, 2023
[5] van der Laan, Carone, and Luedtke, 2024

Direct and indirect policy learning (2/2)

Learning a value-optimal policy can be performed directly or indirectly, using the fact that

$$\pi_P : x \mapsto \mathbf{1}\{\Delta\mu_P(x) > 0\}$$

maximizes $\widetilde{\pi} \mapsto \mathcal{V}_P(\widetilde{\pi})$

EP-learner: introduce $\Psi \subset [-1, 1]^{\mathcal{X}}$, and

$$\psi \mapsto R_P(\psi) = E_P[\psi(X)^2 - 2\psi(X) \cdot \Delta\mu_P(X)] : \tag{1}$$

- $R_P(\cdot)$ is indexed by $\Delta\mu_P$
- $R_P(\cdot)$ targets the "projection" of $\Delta\mu_P$ onto Ψ
- $\tilde{\psi} \in \text{argmin}\{R_P(\psi) : \psi \in \Psi\}$ yields the policy $x \mapsto \mathbf{1}\{\tilde{\psi}(x) > 0\}$

Policy learning under constraint

Fix $\alpha \in [0, 1/2]$, let $\Delta\nu_P(\cdot) = \nu_P(1, \cdot) - \nu_P(0, \cdot)$, and define the constraint

$$\widetilde{\pi} \mapsto S_P(\widetilde{\pi}) = E_P[\widetilde{\pi}(X) \cdot \Delta\nu_P(X)] - \alpha \tag{2}$$

Encourages treatment when it does not significantly raise the probability of an adverse event, with average increase bounded by α

Assumption (Deleterious treatment effect on adverse event)

$\Delta\nu_P \geq 0$

The ideal optimal policy is

$$\text{argmax}\{\mathcal{V}_P(\widetilde{\pi}) : \widetilde{\pi} \in \Pi \text{ s.t. } S_P(\widetilde{\pi}) \leq 0\}$$

Policy learning under constraint

Introduce

$$\Psi = \mathrm{conv}(\{x \mapsto 2\,\mathrm{expit}(\theta^\top x) - 1 : \theta \in \mathbb{R}^d\} \cup \{-1\}) \subset [-1,1]^{\mathcal{X}}, \text{ and}$$

$$[-1,1] \ni u \mapsto \sigma_\beta(u) \propto \log\left(\frac{1+e^{\beta u}}{1+e^{-\beta}}\right) \in [0,1]$$

a smooth and convex approximation of $u \mapsto \mathbf{1}\{u > 0\}$, where $\beta \geq 0$ is a steepness scaling factor

The β-specific optimal policy is the unique solution to

$$\mathrm{argmin}\{R_P(\psi) : \psi \in \Psi \text{ s.t. } S_P(\sigma_\beta \circ \psi) \leq 0\} \tag{3}$$

PLUC: learning a constrained policy (oracular)

From an oracular viewpoint, (3) can be solved via the method of Lagrange multipliers. For $\beta, \lambda \geq 0$, define

$$\psi \mapsto \mathcal{L}_P(\psi, \lambda; \beta) = R_P(\psi) + \lambda S_P(\sigma_\beta \circ \psi), \tag{4}$$

a strongly convex criterion which characterizes a novel non-parametric class of policies

Identification of the best policy:

- Let $\Lambda \times B$ be a set of candidate values for (λ, β)
- Solve (4) for every $(\lambda, \beta) \in \Lambda \times B$ (Frank-Wolfe algorithm [6]), yielding $\widetilde{\pi}_{\lambda,\beta} = \sigma_\beta \circ \psi_{\lambda,\beta}$
- Best policy indexed by any element of

$$\text{argmax}\{\mathcal{V}_P(\widetilde{\pi}_{\lambda,\beta}) : (\lambda, \beta) \in \Lambda \times B \text{ s.t. } S_P(\widetilde{\pi}_{\lambda,\beta}) \leq 0\}$$

[6] Frank, Wolfe, et al., 1956

PLUC: learning a constrained policy (realistic)

Naive approach PLUC

Partition $\{\mathcal{O}_1, \ldots, \mathcal{O}_n\} = \boldsymbol{n}_1 \cup \boldsymbol{n}_2 \cup \boldsymbol{n}_3$ with $|\boldsymbol{n}_1| \approx |\boldsymbol{n}_2| \approx |\boldsymbol{n}_3|$

1- Estimate $\mu_{\boldsymbol{n}_1}$ and $\nu_{\boldsymbol{n}_1}$ using $\boldsymbol{n}_1$, let
$\Delta\mu_{\boldsymbol{n}_1}(\cdot) = \mu_{\boldsymbol{n}_1}(1,\cdot) - \mu_{\boldsymbol{n}_1}(0,\cdot)$ and $\Delta\nu_{\boldsymbol{n}_1}(\cdot) = \nu_{\boldsymbol{n}_1}(1,\cdot) - \nu_{\boldsymbol{n}_1}(0,\cdot)$

2- For every $(\lambda, \beta) \in \Lambda \times B$, using $\boldsymbol{n}_2$, minimize an update of

$$\psi \mapsto \mathcal{L}_{\boldsymbol{n}_1 \cup \boldsymbol{n}_2}(\psi, \lambda; \beta) = \frac{3}{n} \sum_{i \in \boldsymbol{n}_2} [\underbrace{\psi(X_i)^2 - 2\psi(X_i) \cdot \Delta\mu_{\boldsymbol{n}_1}(X_i)}_{R_{\boldsymbol{n}_1}(\psi)} - \lambda \cdot \underbrace{\sigma_\beta \circ \psi(X_i) \cdot \Delta\nu_{\boldsymbol{n}_1}(X_i)}_{S_{\boldsymbol{n}_1}(\sigma_\beta \circ \psi)}] - \alpha$$

simultaneously targeting $\mathcal{L}_P(\psi^0_{\lambda,\beta}, \lambda; \beta), \ldots, \mathcal{L}_P(\psi^k_{\lambda,\beta}, \lambda; \beta)$ [details next]
yielding $\psi_{\lambda,\beta}$ and $\widetilde{\pi}_{\lambda,\beta} = \sigma_\beta \circ \psi_{\lambda,\beta}$

3- Using $\boldsymbol{n}_3$, estimate $\mu_{\boldsymbol{n}_3}$ and $\nu_{\boldsymbol{n}_3}$, build targeted lower-and upper-bound estimators $\underline{\mathcal{V}}^*_{\boldsymbol{n}_3}$ and $\overline{S}^*_{\boldsymbol{n}_3}$, determine

$$\text{argmax}\{\underline{\mathcal{V}}^*_{\boldsymbol{n}_3}(\widetilde{\pi}_{\lambda,\beta}) : (\lambda, \beta) \in \Lambda \times B \text{ s.t. } \overline{S}^*_{\boldsymbol{n}_3}(\widetilde{\pi}_{\lambda,\beta}) \leq 0\}$$

PLUC: learning a constrained policy (realistic)

PLUC: focus on step 2

2- For every $(\lambda, \beta) \in \Lambda \times B$, using $\{O_i : i \in \boldsymbol{n}_2\}$, minimize an update of

$$\psi \mapsto \mathcal{L}^0_{\boldsymbol{n}_1 \cup \boldsymbol{n}_2}(\psi, \lambda; \beta) = \frac{3}{n} \sum_{i \in \boldsymbol{n}_2} [\psi(X_i)^2 - 2\psi(X_i) \cdot \Delta\mu_{\boldsymbol{n}_1}(X_i) - \lambda \cdot \sigma_\beta \circ \psi(X_i) \cdot \Delta\nu_{\boldsymbol{n}_1}(X_i)] - \alpha$$

simultaneously targeting $\mathcal{L}_P(\psi^0_{\lambda,\beta}, \lambda; \beta), \ldots, \mathcal{L}_P(\psi^k_{\lambda,\beta}, \lambda; \beta)$
yielding $\psi_{\lambda,\beta}$ and $\widetilde{\pi}_{\lambda,\beta}$

- We carry out an alternating minimization procedure
- Step k decomposes into:
  - a correction substep: update $\mu^k_{\boldsymbol{n}_1}$ and $\nu^k_{\boldsymbol{n}_1}$ so that $\psi \mapsto \mathcal{L}^k_{\boldsymbol{n}_1 \cup \boldsymbol{n}_2}(\psi, \lambda; \beta)$ targets $\mathcal{L}_P(\psi^0_{\lambda,\beta}, \lambda; \beta), \ldots, \mathcal{L}_P(\psi^k_{\lambda,\beta}, \lambda; \beta)$
  - a minimization substep: minimize $\psi \mapsto \mathcal{L}^k_{\boldsymbol{n}_1 \cup \boldsymbol{n}_2}(\psi, \lambda; \beta)$, yielding $\psi^{k+1}_{\lambda,\beta}$

Simulation design

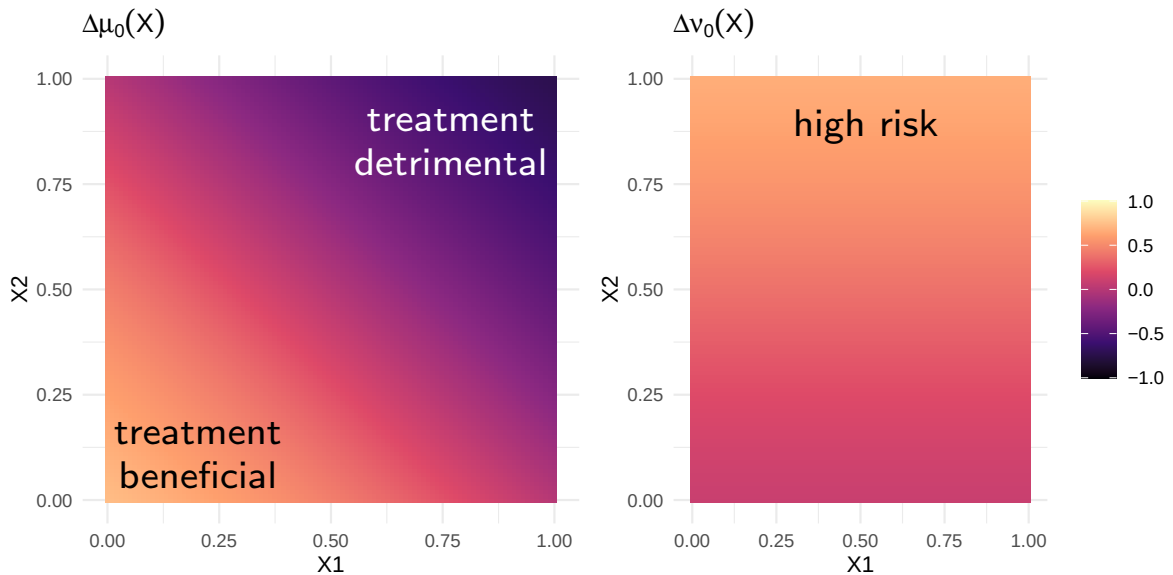
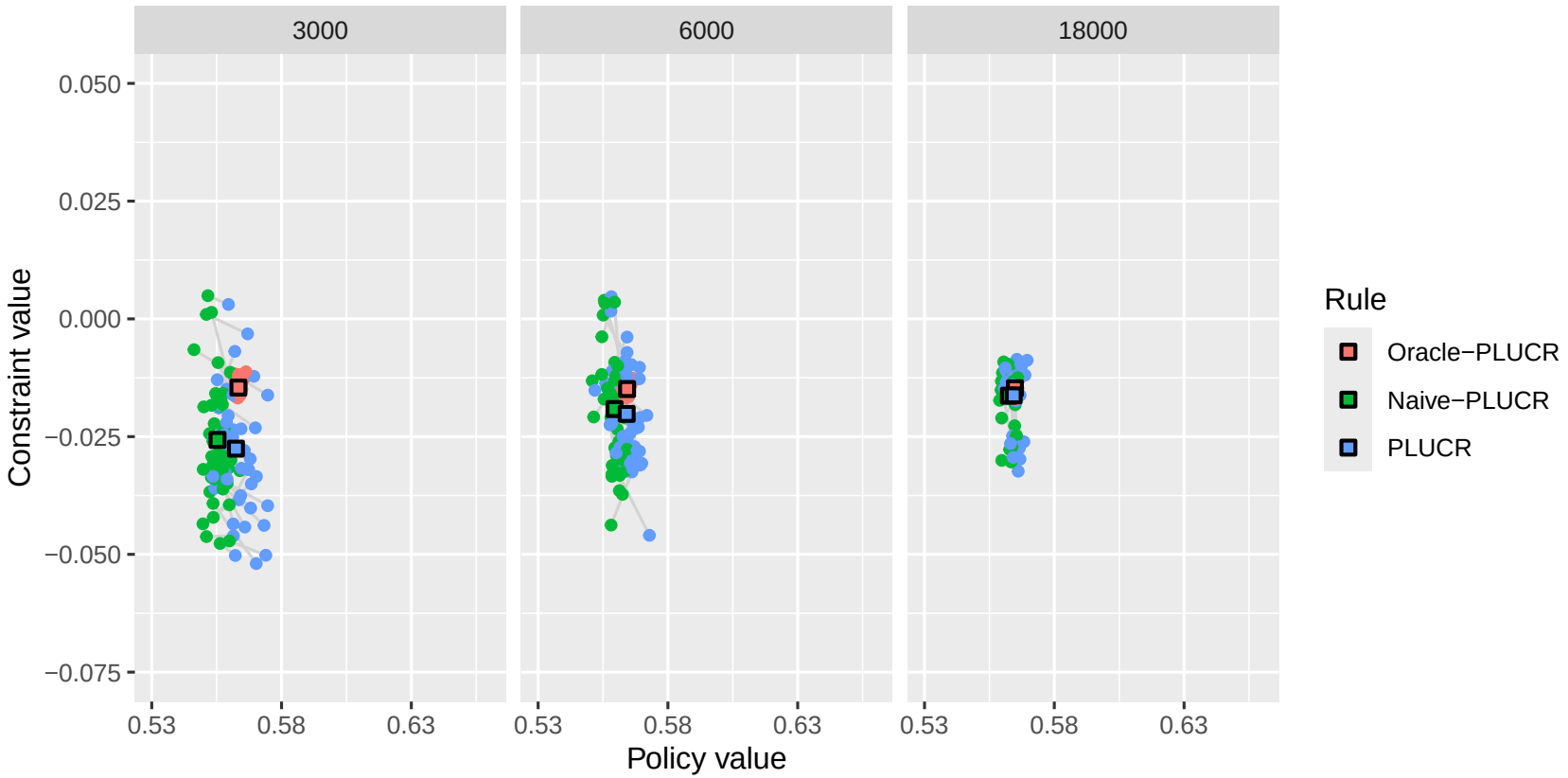


Figure: Treatment effect patterns, with the primary outcome displayed on the left and the adverse event displayed on the right.

Three sample sizes $n \in \{3{,}000; 6{,}000; 18{,}000\}$ with 50 replicates per size
Fixed $\alpha = 0.1$, $\Lambda = \{1, \ldots, 10\}$ and $B = \{0, 0.05, 0.1, 0.25, 0.5\}$

Summary of the results



- PLUC slightly improves Naive PLUC's performance
- Constraint verified

Application to IVF data

- $n = 18,538$ observations
- treatment levels: high dose (A = 1), low dose (A = 0)
- Y: number of retrieved follicles, ξ: estradiol levels exceeding 3000 UI

Policy $\widetilde{\pi}$	$\mathcal{V}_{n_3}(\widetilde{\pi})$	$S_{n_3}(\widetilde{\pi})$	$\widehat{\mathrm{Prob}}(\widetilde{\pi}(X)=1)$
PLUC $(x \mapsto \widetilde{\pi}_{\mathrm{PLUC}}(x))$	**9.857**	**-0.095**	0
PLUC recommendation $(x \mapsto \widetilde{\pi}^{t_P}_{\mathrm{PLUC}}(x))$	**10.079**	**-0.036**	**0.081**
Naive PLUC $(x \mapsto \widetilde{\pi}_{\mathrm{Naive}}(x))$	**10.094**	**-0.050**	0
Naive PLUC recommendation $(x \mapsto \widetilde{\pi}^{t_N}_{\mathrm{Naive}}(x))$	**10.094**	**-0.050**	**0**
Clinician-assigned (A)	10.056	-0.029	0.150
Surrogates (unconstrained / individual-constraint) $x \mapsto \mathbf{1}\{\Delta\mu_{n_1}(x) > 0\}$ $x \mapsto \mathbf{1}\{\Delta\mu_{n_1}(x) > 0,\ \Delta\nu_{n_1}(x) \leq \alpha\}$	10.096	-0.045	0.007

PLUC enforces explicit control of adverse-event risks

Thank you!

- ArXiv preprint

  `https://arxiv.org/pdf/2601.22717`
- PLUC-R package

  `https://github.com/laufuentes/PLUCR`

References I

[1] Marguerite Frank, Philip Wolfe, et al. "An algorithm for quadratic programming." In: *Naval research logistics quarterly* 3.1-2 (1956), pp. 95–110.

[2] Edward H. Kennedy. "Towards optimal doubly robust estimation of heterogeneous causal effects." In: *Electron. J. Stat.* 17.2 (2023), pp. 3008–3049. DOI: 10.1214/23-ejs2157.

[3] Sören R. Künzel et al. "Metalearners for estimating heterogeneous treatment effects using machine learning." In: *Proceedings of the National Academy of Sciences of the United States of America* 116 (2017), pp. 4156–4165.

[4] Lars van der Laan, Marco Carone, and Alex Luedtke. "Combining T-learning and DR-learning: a framework for oracle-efficient estimation of causal contrasts." In: *arXiv preprint arXiv:2402.01972* (2024).

References II

[5] Yingqi Zhao et al. "Estimating individualized treatment rules using outcome weighted learning." In: *J. Amer. Statist. Assoc.* 107.499 (2012), pp. 1106–1118. DOI: 10.1080/01621459.2012.695674.

[6] Xin Zhou et al. "Residual weighted learning for estimating individualized treatment rules." In: *J. Amer. Statist. Assoc.* 112.517 (2017), pp. 169–187. DOI: 10.1080/01621459.2015.1093947.

From policies to deterministic recommendation rules

A policy can be mapped to a deterministic recommendation rule in several ways.

Example: thresholding
For any $\widetilde{\pi} \in \Pi$ and $t \in [0,1]$, define

$$\pi^t : x \mapsto \mathbf{1}\{\widetilde{\pi}(x) \geq t\}$$

The threshold t can be chosen as

$$\underset{t \in [0,1]}{\operatorname{argmax}} \{\underline{\mathcal{V}}^*_{\boldsymbol{n}_3}(\pi^t) \text{ s.t. } \overline{S}^*_{\boldsymbol{n}_3}(\pi^t) \leq 0\}$$