Set-Valued Policy Learning

Laura Fuentes-Vicente[1], Mathieu Even[1], Gaëlle Dormion[2], Antoine Chambaz[3], Uri Shalit[4], Julie Josse[1]

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

Journées de Statistique de la SFdS
June 3 2026, Clermont-Ferrand

Medical Motivation

Classical policy learning: Support decision-making by tailoring treatment decisions to individual patient characteristics

Clinical example (IVF): Estimate the optimal hormone dose to maximize the number of oocytes produced

Medical Motivation

Classical policy learning: Support decision-making by tailoring treatment decisions to individual patient characteristics

Clinical example (IVF): Estimate the optimal hormone dose to maximize the number of oocytes produced

Clinical utility depends heavily on integration into real-world practice (i.e. human-AI interactions) [2, 6, 7]

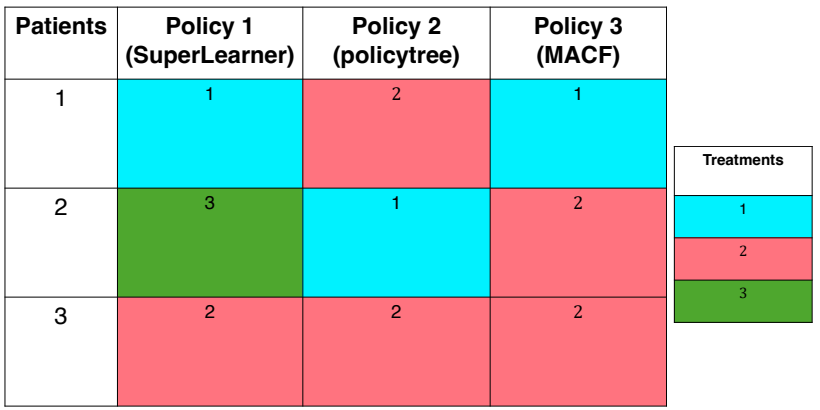
Medical Motivation

Classical policy learning: Support decision-making by tailoring treatment decisions to individual patient characteristics

Clinical example (IVF): Estimate the optimal hormone dose to maximize the number of oocytes produced

Clinical utility depends heavily on integration into real-world practice (i.e. human-AI interactions) [2, 6, 7]

- × Conflicting recommendations
- × Rigid recommendations:
  - × Conflict with practitioner preferences
  - × Ignore contextual factors (e.g. cost, adverse events)



Statistical modeling

- Set of n i.i.d. observations $\mathcal{D}_1, \ldots, \mathcal{D}_n \sim P$
- Generic data structure $\mathcal{D} = (X, A, Y)$:
  - $X \in \mathcal{X}$: vector of covariates
  - $A \in \mathcal{A} = \{1, \ldots, K\}$: categorical treatment assignment
  - $Y \in \mathcal{Y}$: outcome
- Relevant nuisances: $\mu(X, a) = E[Y|X, A = a]$

Policies and their values

A policy $\pi \in \Pi \subset \mathcal{A}^{\mathcal{X}}$ maps any $x \in \mathcal{X}$ to a treatment assignment $a \in \mathcal{A}$

The **value** of a policy $\pi \in \Pi$ under P is defined as

$$\mathcal{V}(\pi) = E[\mu(X, \pi(X))]$$

Policies and their values

A policy $\pi \in \Pi \subset \mathcal{A}^{\mathcal{X}}$ maps any $x \in \mathcal{X}$ to a treatment assignment $a \in \mathcal{A}$

The **value** of a policy $\pi \in \Pi$ under P is defined as

$$\mathcal{V}(\pi) = E[\mu(X, \pi(X))]$$

Learning a value-optimal policy π^*, maximizing $\pi \mapsto \mathcal{V}(\pi)$ can be performed via two primary paradigms:

- **direct** maximization of a consistent estimator of $\pi \mapsto \mathcal{V}(\pi)$ [8, 4]
- **indirect** maximization of an estimator of $(x, a) \mapsto \mu(x, a)$

Policies and their values

A policy $\pi \in \Pi \subset \mathcal{A}^{\mathcal{X}}$ maps any $x \in \mathcal{X}$ to a treatment assignment $a \in \mathcal{A}$

The **value** of a policy $\pi \in \Pi$ under P is defined as

$$\mathcal{V}(\pi) = E[\mu(X, \pi(X))]$$

Learning a value-optimal policy π^*, maximizing $\pi \mapsto \mathcal{V}(\pi)$ can be performed via two primary paradigms:

- **direct** maximization of a consistent estimator of $\pi \mapsto \mathcal{V}(\pi)$ [8, 4]
- **indirect** maximization of an estimator of $(x, a) \mapsto \mu(x, a)$

Define the X-specific set of optimal treatment regimes as

$$\Pi^*(X) = \{\pi^*(X) : \pi^* \in \operatorname*{argmax}_{\pi \in \Pi} \mathcal{V}(\pi)\}$$

Conformal prediction (1/2)

Conformal prediction (CP) [10, 12, 11] is a model-agnostic and distribution-free framework for uncertainty quantification, providing prediction sets with guaranteed marginal coverage properties

- Set of $(n+1)$ i.i.d. samples $\mathcal{O}_1, \ldots, \mathcal{O}_{n+1}$
- Each sample $\mathcal{O} = (X, T)$ comprises
  - $X \in \mathcal{X}$: vector of covariates
  - $T \in \mathcal{T} = \{1, \ldots, K\}$ a label, with $K > 1$
- **Goal**: Predict unseen label T_{n+1} at X_{n+1} with high probability
  For fixed $\alpha > 0$, build $C^\alpha : \mathcal{X} \to \mathcal{P}(\mathcal{T})$ such that

$$P(T_{n+1} \in C^\alpha(X_{n+1})) \geq 1 - \alpha$$

Conformal prediction (2/2)

Inductive CP (ICP)

0. Partition $\{\mathcal{O}_i\}_{i\in[n]}$ into two disjoint train and calibration subsets
1. Fit a nonconformity score function $s : \mathcal{X} \times \mathcal{T} \to \mathbb{R}$ (train)
2. Evaluate nonconformity scores $\{S_i = s(X_i, T_i)\}_{i\in\mathcal{I}_{\text{cal}}}$ (calibration)
3. For fixed $\alpha > 0$, let

$$q_{1-\alpha} = \text{Quantile}\left(\frac{\lceil (1-\alpha)(1+n_{\text{cal}})\rceil}{n_{\text{cal}}}; \{S_i\}_{i\in\mathcal{I}_{\text{cal}}}\right)$$

4. The prediction set associated to the test point X_{n+1}

$$C^{\alpha}(X_{n+1}) = \{t \in \mathcal{T} : s(X_{n+1}, t) < q_{1-\alpha}\} \subseteq \mathcal{P}(\mathcal{T})$$

Conformal prediction (2/2)

Inductive CP (ICP)

0. Partition $\{\mathcal{O}_i\}_{i\in[n]}$ into two disjoint train and calibration subsets
1. Fit a nonconformity score function $s : \mathcal{X} \times \mathcal{T} \to \mathbb{R}$ (train)
2. Evaluate nonconformity scores $\{S_i = s(X_i, T_i)\}_{i\in\mathcal{I}_{\text{cal}}}$ (calibration)
3. For fixed $\alpha > 0$, let

$$q_{1-\alpha} = \text{Quantile}\left(\frac{\lceil (1-\alpha)(1+n_{\text{cal}})\rceil}{n_{\text{cal}}}; \{S_i\}_{i\in\mathcal{I}_{\text{cal}}}\right)$$

4. The prediction set associated to the test point X_{n+1}

$$C^{\alpha}(X_{n+1}) = \{t \in \mathcal{T} : s(X_{n+1}, t) < q_{1-\alpha}\} \subseteq \mathcal{P}(\mathcal{T})$$

Noisy label regime: ICP applied to noisy proxies (e.g. user annotations, label estimations). Valid coverage for the unavailable true labels requires additional conditions [5, 9, 3, 1] (details next)

Set-valued policy learning

Definition (Set-valued policy)

A set-valued policy C is an element of $\mathcal{P}(\mathcal{A})^{\mathcal{X}}$ that maps covariates to a set of treatments

Our goal: Build set-valued policies C that contain an optimal treatment with high probability (fix $\alpha > 0$)

$$P(\Pi^*(X_{n+1}) \cap C(X_{n+1}) \neq \emptyset) \geq 1 - \alpha$$

Set-valued policy learning

Definition (Set-valued policy)

A set-valued policy C is an element of $\mathcal{P}(\mathcal{A})^{\mathcal{X}}$ that maps covariates to a set of treatments

Our goal: Build set-valued policies C that contain an optimal treatment with high probability (fix $\alpha > 0$)

$$P(\Pi^*(X_{n+1}) \cap C(X_{n+1}) \neq \emptyset) \geq 1 - \alpha$$

CP for policy learning: True samples $\{\mathcal{O}_i = (X_i, A_i^*)\}_{i \in [n]}$ are inaccessible due to the fundamental challenge of causal inference (i.e. $A_i^* \in \Pi^*(X_i)$ unobserved), hindering direct application of ICP

Set-valued policy learning

Definition (Set-valued policy)

A set-valued policy C is an element of $\mathcal{P}(\mathcal{A})^{\mathcal{X}}$ that maps covariates to a set of treatments

Our goal: Build set-valued policies C that contain an optimal treatment with high probability (fix $\alpha > 0$)

$$P(\Pi^*(X_{n+1}) \cap C(X_{n+1}) \neq \emptyset) \geq 1 - \alpha$$

CP for policy learning: True samples $\{\mathcal{O}_i = (X_i, A_i^*)\}_{i \in [n]}$ are inaccessible due to the fundamental challenge of causal inference (i.e. $A_i^* \in \Pi^*(X_i)$ unobserved), hindering direct application of ICP

We use $\{\mathcal{D}_i = (X_i, A_i, Y_i)\}_{i \in [n]}$ to estimate noisy samples $\{\hat{\mathcal{O}}_i = (X_i, \hat{A}_i^*)\}_{i \in [n]}$, casting the problem into a noisy label regime

Noisy ICP for policy learning

0. Partition $\{\mathcal{D}_i = (X_i, A_i, Y_i)\}_{i \in [n]}$ into $\mathcal{D}_b \cup \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{cal}}$

Noisy ICP for policy learning

0. Partition $\{\mathcal{D}_i = (X_i, A_i, Y_i)\}_{i \in [n]}$ into $\mathcal{D}_b \cup \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{cal}}$
1. **Black-box label generation.** Train policy learning algorithm $\mathcal{B}$ using $\mathcal{D}_b$: $\mathcal{B}(\mathcal{D}_b) : \mathcal{X} \to \mathcal{A}$

Noisy ICP for policy learning

0. Partition $\{\mathcal{D}_i = (X_i, A_i, Y_i)\}_{i \in [n]}$ into $\mathcal{D}_b \cup \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{cal}}$
1. **Black-box label generation.** Train policy learning algorithm $\mathcal{B}$ using $\mathcal{D}_b$: $\mathcal{B}(\mathcal{D}_b) : \mathcal{X} \to \mathcal{A}$
2. **Nonconformity score.** Fit the nonconformity score (i.e. estimate μ) $(x, a) \mapsto s(x, a) = \max_{a' \neq a} \mu(x, a') - \mu(x, a)$ using $\mathcal{D}_{\text{train}}$

Noisy ICP for policy learning

0. Partition $\{\mathcal{D}_i = (X_i, A_i, Y_i)\}_{i\in[n]}$ into $\mathcal{D}_b \cup \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{cal}}$
1. **Black-box label generation.** Train policy learning algorithm $\mathcal{B}$ using $\mathcal{D}_b$: $\mathcal{B}(\mathcal{D}_b) : \mathcal{X} \to \mathcal{A}$
2. **Nonconformity score.** Fit the nonconformity score (i.e. estimate μ) $(x, a) \mapsto s(x, a) = \max_{a' \neq a} \mu(x, a') - \mu(x, a)$ using $\mathcal{D}_{\text{train}}$
3. **Noisy calibration.**
   (i) Generate the noisy samples $\hat{A}_i^* = \mathcal{B}(\mathcal{D}_b)(X_i)$, for all $i \in \mathcal{I}_{\text{cal}}$
   (ii) Evaluate nonconformity scores $\{\hat{S}_i = s(X_i, \hat{A}_i^*)\}_{i\in\mathcal{I}_{\text{cal}}}$

Noisy ICP for policy learning

0. Partition $\{\mathcal{D}_i = (X_i, A_i, Y_i)\}_{i\in[n]}$ into $\mathcal{D}_b \cup \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{cal}}$
1. **Black-box label generation.** Train policy learning algorithm $\mathcal{B}$ using $\mathcal{D}_b$: $\mathcal{B}(\mathcal{D}_b) : \mathcal{X} \to \mathcal{A}$
2. **Nonconformity score.** Fit the nonconformity score (i.e. estimate μ) $(x, a) \mapsto s(x, a) = \max_{a' \neq a} \mu(x, a') - \mu(x, a)$ using $\mathcal{D}_{\text{train}}$
3. **Noisy calibration.**
   (i) Generate the noisy samples $\hat{A}_i^* = \mathcal{B}(\mathcal{D}_b)(X_i)$, for all $i \in \mathcal{I}_{\text{cal}}$
   (ii) Evaluate nonconformity scores $\{\hat{S}_i = s(X_i, \hat{A}_i^*)\}_{i\in\mathcal{I}_{\text{cal}}}$
4. **Set-valued policy.** Let $\hat{q}_{1-\alpha} = \text{Quantile}\left(\frac{\lceil(1-\alpha)(1+n_{\text{cal}})\rceil}{n_{\text{cal}}}, \{\hat{S}_i\}\right)$,

$$\text{define} \quad x \mapsto \hat{C}^{\alpha}(x) = \{a \in \mathcal{A} : s(x, a) < \hat{q}_{1-\alpha}\}$$

Noisy ICP for policy learning

0. Partition $\{\mathcal{D}_i = (X_i, A_i, Y_i)\}_{i\in[n]}$ into $\mathcal{D}_b \cup \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{cal}}$
1. **Black-box label generation.** Train policy learning algorithm $\mathcal{B}$ using $\mathcal{D}_b$: $\mathcal{B}(\mathcal{D}_b) : \mathcal{X} \to \mathcal{A}$
2. **Nonconformity score.** Fit the nonconformity score (i.e. estimate μ) $(x, a) \mapsto s(x, a) = \max_{a' \neq a} \mu(x, a') - \mu(x, a)$ using $\mathcal{D}_{\text{train}}$
3. **Noisy calibration.**
   (i) Generate the noisy samples $\hat{A}_i^* = \mathcal{B}(\mathcal{D}_b)(X_i)$, for all $i \in \mathcal{I}_{\text{cal}}$
   (ii) Evaluate nonconformity scores $\{\hat{S}_i = s(X_i, \hat{A}_i^*)\}_{i\in\mathcal{I}_{\text{cal}}}$
4. **Set-valued policy.** Let $\hat{q}_{1-\alpha} = \text{Quantile}\left(\frac{\lceil(1-\alpha)(1+n_{\text{cal}})\rceil}{n_{\text{cal}}}, \{\hat{S}_i\}\right)$,

$$\text{define} \quad x \mapsto \hat{C}^\alpha(x) = \{a \in \mathcal{A} : s(x, a) < \hat{q}_{1-\alpha}\}$$

"$P(\hat{A}_{n+1}^* \in \hat{C}^\alpha(X_{n+1})) \geq 1 - \alpha$" $\neq$ "$P(A_{n+1}^* \in \hat{C}^\alpha(X_{n+1})) \geq 1 - \alpha$"

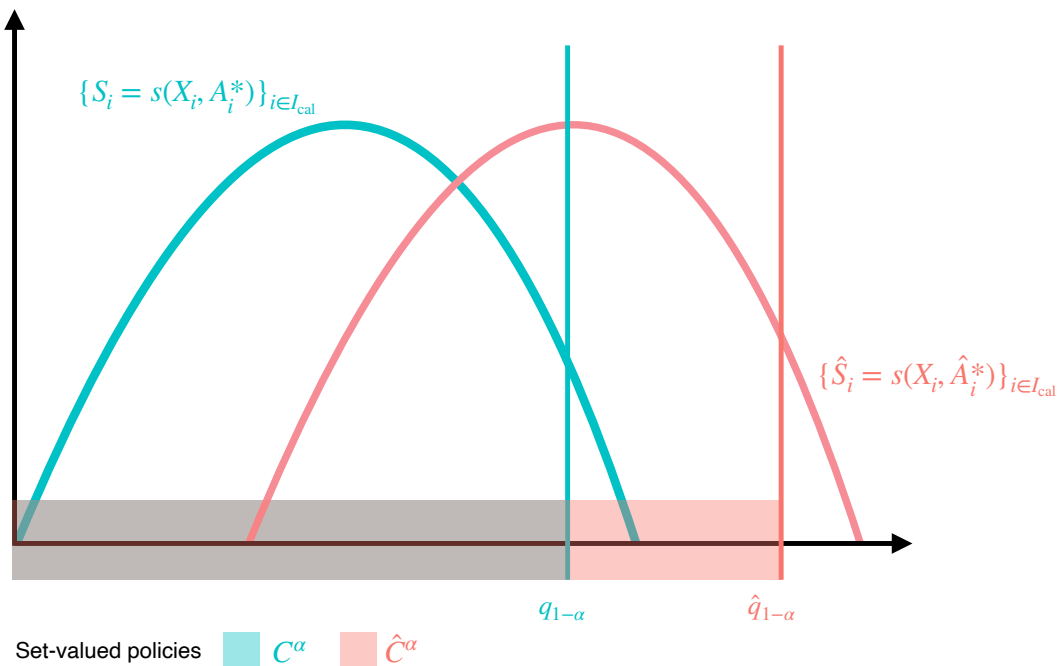
Valid coverage under label noise

A sufficient condition to guarantee

$$\forall \alpha > 0, \quad P(A^*_{n+1} \in \hat{C}^\alpha(X_{n+1})) \geq 1 - \alpha, \tag{1}$$

is first order stochastic dominance: $\hat{S}_i \succeq_{(1)} S_i \implies C^\alpha \subseteq \hat{C}^\alpha$

i.e. noisy labels $\hat{A}^*_i$ are "less reliable" than the true optimal treatments A^*_i



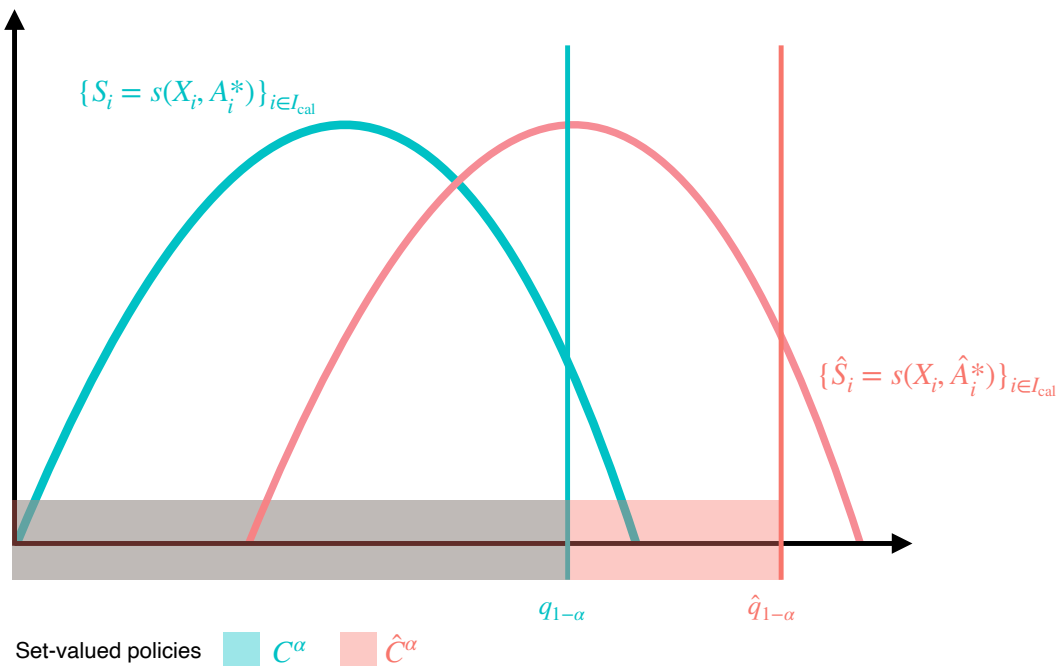
Valid coverage under label noise

A sufficient condition to guarantee

$$\forall \alpha > 0, \quad P(A^*_{n+1} \in \hat{C}^\alpha(X_{n+1})) \geq 1 - \alpha, \tag{1}$$

is first order stochastic dominance: $\hat{S}_i \succeq_{(1)} S_i \implies C^\alpha \subseteq \hat{C}^\alpha$

i.e. noisy labels $\hat{A}^*_i$ are "less reliable" than the true optimal treatments A^*_i



Estimation errors in s can lead to lower scores for suboptimal treatments.
By favoring similarity to incorrect labels, this behavior compromises (1)

Randomness injection: introducing exploration

To encourage $\hat{S}_i \succeq_{(1)} S_i$ we perturb $\hat{A}_i^* = B(\mathcal{D}_b)(X_i)$, for all $i \in \mathcal{I}_{\text{cal}}$
(i.e. reduce impact of estimation errors on s)

Randomness injection: introducing exploration

To encourage $\hat{S}_i \succeq_{(1)} S_i$ we perturb $\hat{A}_i^* = B(\mathcal{D}_b)(X_i)$, for all $i \in \mathcal{I}_{\text{cal}}$ (i.e. reduce impact of estimation errors on s)

Suppose that the random label $A_i^{\text{rd}} \sim \mathcal{U}([K])$ satisfies

Assumption

$S_i^{\text{rd}} = s(X_i, A_i^{\text{rd}}) \succeq_{(1)} S_i = s(X_i, A_i^*)$ *and* $S_i^{\text{rd}} \succeq_{(1)} \hat{S}_i^*$

Fix $r \in [0, 1]$, introduce $R_i \sim \text{Ber}(r)$, and define the perturbed label as

$$\hat{A}_{r,i}^* = R_i \cdot A_i^{\text{rd}} + (1 - R_i) \cdot \hat{A}_i^*$$

Randomness injection: introducing exploration

To encourage $\hat{S}_i \succeq_{(1)} S_i$ we perturb $\hat{A}_i^* = B(\mathcal{D}_b)(X_i)$, for all $i \in \mathcal{I}_{\text{cal}}$
(i.e. reduce impact of estimation errors on s)

Suppose that the random label $A_i^{\text{rd}} \sim \mathcal{U}([K])$ satisfies

Assumption

$S_i^{\text{rd}} = s(X_i, A_i^{\text{rd}}) \succeq_{(1)} S_i = s(X_i, A_i^*)$ *and* $S_i^{\text{rd}} \succeq_{(1)} \hat{S}_i^*$

Fix $r \in [0, 1]$, introduce $R_i \sim \text{Ber}(r)$, and define the perturbed label as

$$\hat{A}_{r,i}^* = R_i \cdot A_i^{\text{rd}} + (1 - R_i) \cdot \hat{A}_i^*$$

The set-policy $\hat{C}^\alpha$ built using $\hat{A}_{r,i}^*$ satisfies marginal coverage iff

$$r \geq \bar{r} = \frac{E[P(s(X_i, \hat{A}_i^*) \leq \hat{q}_{1-\alpha}) - P(s(X_i, A_i^*) \leq \hat{q}_{1-\alpha})]}{E[P(s(X_i, \hat{A}_i^*) \leq \hat{q}_{1-\alpha}) - P(s(X_i, A_i^{\text{rd}}) \leq \hat{q}_{1-\alpha})]}$$

(expectations w.r.t. $\hat{q}_{1-\alpha}$, and $\bar{r}$ unknown)

Simulation design

- Gaussian covariates $X \in \mathbb{R}^4$ with two uninformative dimensions
- $K = 5$ treatment levels
- Covariate space is divided in two:
  optimal treatments (i.e. larger outcomes) are (i) {1,2} (ii) {3,4}

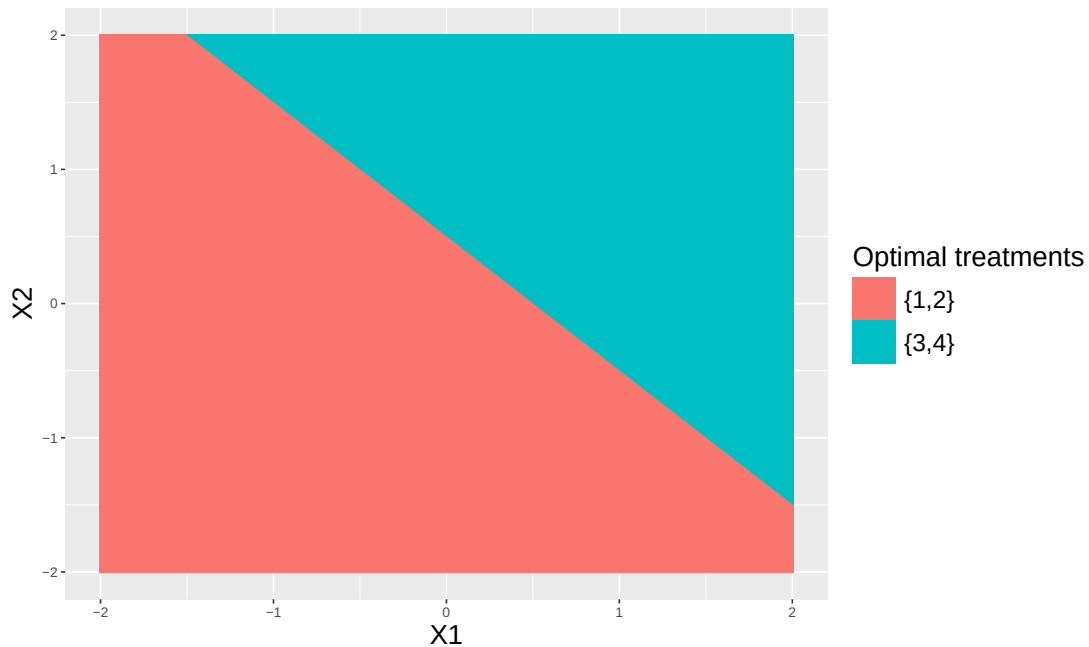


Figure: Distribution of optimal treatment assignments by feature values

Randomness injection enforces marginal coverage

	OCP	r=0	r=0.2	r=0.5
Coverage $\{1, 2\}$	0.89	0.83	0.91	0.96
Coverage $\{3, 4\}$	0.90	0.87	0.91	0.95
Overall coverage	0.90	**0.85**	**0.91**	0.96
$E[\lvert C^{\alpha}(X_i)\rvert]$	2.97	**2.57**	**3.21**	3.96
SPV	6.99	7.17	6.84	5.96

Figure: Comparison of conditional coverage (on $\Pi^{\star}(X_i)$), overall coverage, mean cardinality and set-policy value across different set-valued policy learning methods ($\alpha = 0.1$ and $n = 6{,}000$). *OCP*: Oracular CP, *SPV*: Set-Policy Value

Application to IVF data: dose de-escalation?

Dataset consists of $n = 18{,}538$ recorded ovarian stimulation cycles

- Baseline characteristics ($X \in \mathbb{R}^{16}$)
- Ordinal gonadotropin dosages ($\mathcal{A} = \{1, ..., 6\}$)
- Outcomes: follicular yield (Y) to be maximized and estradiol levels (ξ), higher levels increase the risk of ovarian hyper stimulation syndrome

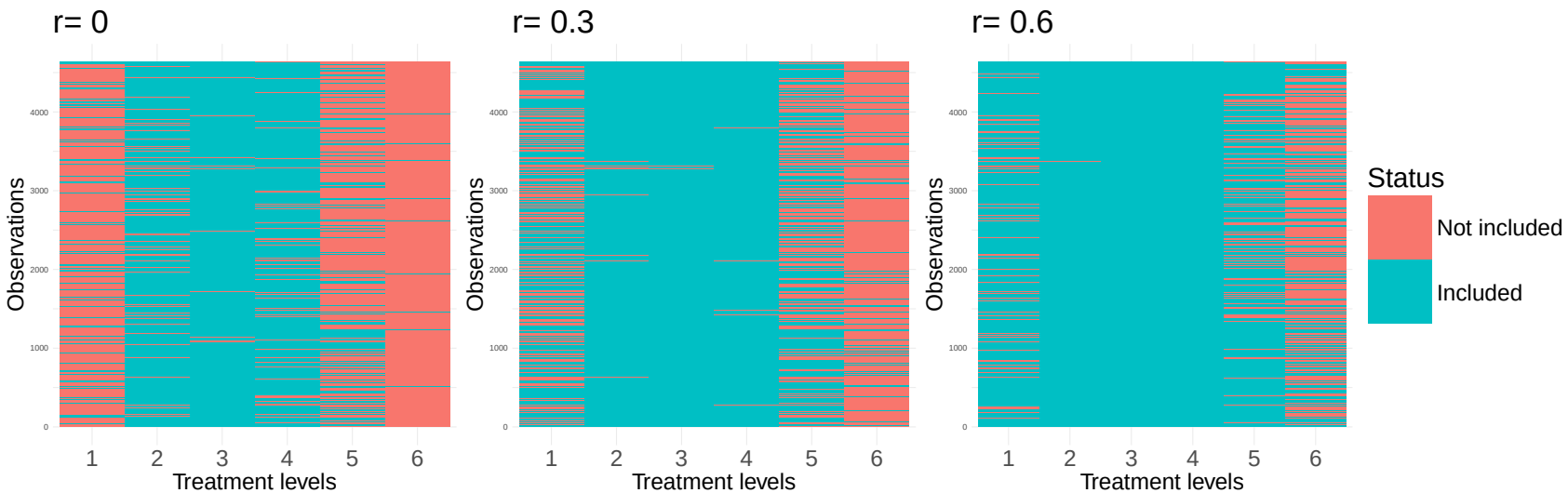


Figure: Rows: individual observations. Columns: treatment levels included in conformal set-valued policies, for $\alpha = 0.1$.

Application to IVF data: dose de-escalation?

Dataset consists of $n = 18{,}538$ recorded ovarian stimulation cycles

- Baseline characteristics ($X \in \mathbb{R}^{16}$)
- Ordinal gonadotropin dosages ($\mathcal{A} = \{1, ..., 6\}$)
- Outcomes: follicular yield (Y) to be maximized and estradiol levels (ξ), higher levels increase the risk of ovarian hyper stimulation syndrome

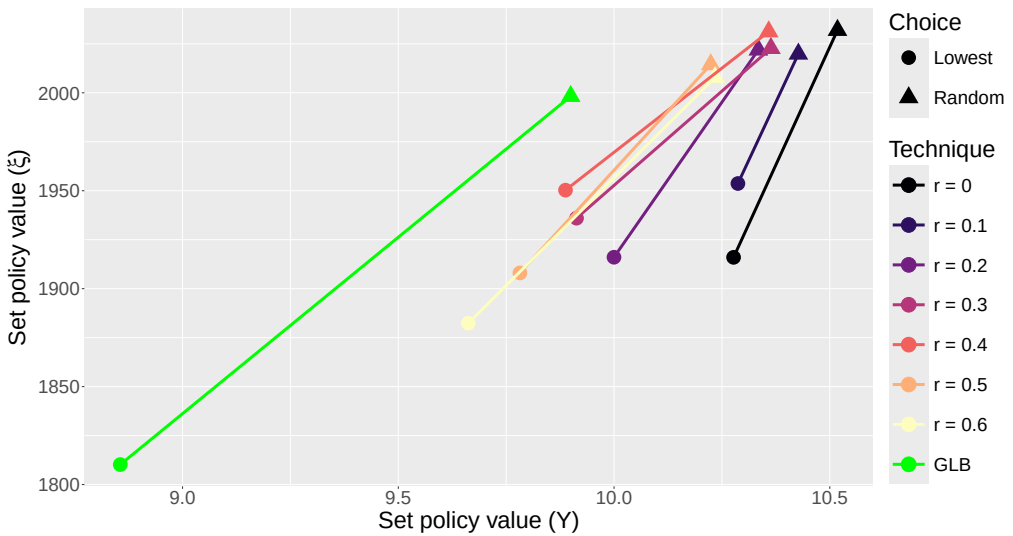


Figure: Set-policy values for Y (x-axis) and ξ (y-axis) across two decision strategies: δ_{lower} (points) and δ_{unif} (triangles), $\alpha = 0.1$.

Thank you!

- ArXiv preprint

  `https://arxiv.org/pdf/2605.19830`
- R package

  `https://github.com/laufuentes/setValuedPolicyLearning.git`

References I

[1] Ahmed M Alaa, Zaid Ahmad, and Mark van der Laan. "Conformal meta-learners for predictive inference of individual treatment effects." In: *Advances in neural information processing systems* 36 (2023), pp. 47682–47703.

[2] Eli Ben-Michael et al. "Does AI help humans make better decisions? A statistical evaluation framework for experimental and observational studies." In: *Proceedings of the National Academy of Sciences* 122.38 (2025), e2505106122.

[3] Maxime Cauchois et al. "Predictive inference with weak supervision." In: *J. Mach. Learn. Res.* 25 (2024), Paper No. [118], 45. ISSN: 1532-4435,1533-7928.

[4] Miroslav Dudík, John Langford, and Lihong Li. "Doubly robust policy evaluation and learning." In: *arXiv preprint arXiv:1103.4601* (2011).

References II

[5] Bat-Sheva Einbinder et al. "Label noise robustness of conformal prediction." In: *J. Mach. Learn. Res.* 25 (2024), Paper No. [328], 66. ISSN: 1532-4435,1533-7928.

[6] Kosuke Imai et al. "Experimental evaluation of algorithm-assisted human decision-making: Application to pretrial public safety assessment." In: *Journal of the Royal Statistical Society Series A: Statistics in Society* 186.2 (2023), pp. 167–189.

[7] Yi Lai, Atreyi Kankanhalli, and Desmond Ong. "Human-AI collaboration in healthcare: A review and research agenda." In: (2021).

[8] Min Qian and Susan A. Murphy. "Performance guarantees for individualized treatment rules." In: *Ann. Statist.* 39.2 (2011), pp. 1180–1210. ISSN: 0090-5364,2168-8966. DOI: 10.1214/10-AOS864. URL: https://doi.org/10.1214/10-AOS864.

References III

[9] Matteo Sesia, Y. X. Rachel Wang, and Xin Tong. "Adaptive conformal classification with noisy labels." In: *J. R. Stat. Soc. Ser. B. Stat. Methodol.* 87.3 (2025), pp. 796–815. ISSN: 1369-7412,1467-9868. DOI: 10.1093/jrsssb/qkae114. URL: https://doi.org/10.1093/jrsssb/qkae114.

[10] Glenn Shafer and Vladimir Vovk. "A tutorial on conformal prediction." In: *J. Mach. Learn. Res.* 9 (2008), pp. 371–421. ISSN: 1532-4435,1533-7928.

[11] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer, New York, 2005, pp. xvi+324. ISBN: 978-0387-00152-4; 0-387-00152-2.

[12] Volodya Vovk, Alexander Gammerman, and Craig Saunders. "Machine-Learning Applications of Algorithmic Randomness." In: *Proceedings of the Sixteenth International Conference on Machine Learning*. ICML '99. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999, pp. 444–453. ISBN: 1558606122.

Set-policy value

Definition (Choice function and set-valued policy value)

Let $\delta : (\mathcal{P}(\mathcal{A}), X) \to A$ a choice function. The δ-specific value of a set-valued policy C is defined as

$$\mathcal{V}(C, \delta) = E[\mu(X, \delta(C(X), X))],$$

the value of the policy $\pi \mapsto \delta(C(x), x)$

A decision-maker strategy to select a treatment from $x \mapsto C(x)$ is given by the choice function δ^*. In their hands, the value of C is $\mathcal{V}(C, \delta_*)$.

To address the unavailability of δ_*, we introduce the *uniform set-policy value*:

$$\bar{\mathcal{V}}_P(C) = E\left[\frac{1}{|C(X)|} \sum_{a \in C(X)} \mu(X, a)\right]$$

based on the choice function $\delta_{\text{unif}}(C(x), x) \sim \mathcal{U}(C(x))$

Greatest lower bound (GLB)

Relies on uncertainty quantification relative to estimators of μ.

Fix $\alpha \in [0,1]$ and introduce

- $(x,a) \mapsto \ell_n(x,a;1-\alpha/2)$: lower-bound estimator at level $1-\alpha/2$
- $(x,a) \mapsto u_n(x,a;1-\alpha/2)$: upper-bound estimator at level $1-\alpha/2$

such that for all $(x,a) \in \mathcal{X} \times \mathcal{A}$,

$$P(\mu(x,a) \in [\ell_n(x,a;1-\alpha/2), u_n(x,a;1-\alpha/2)]) \geq 1-\alpha$$

We define the greatest lower-bound treatment policy

$$x \mapsto a_{\text{maxmin}}(x) \in \text{argmax}\{\ell_n(x,a;1-\alpha/2) : a \in \mathcal{A}\}$$

and construct the set-valued policy by including all treatments whose upper-bound exceeds the greatest lower-bound benchmark, yielding

$$x \mapsto C^{\alpha}(x) = \{a \in \mathcal{A} : u_n(x,a;1-\alpha/2) \geq \ell_n(x,a_{\text{maxmin}}(x);1-\alpha/2)\}$$

Additional synthetic results (1/3)

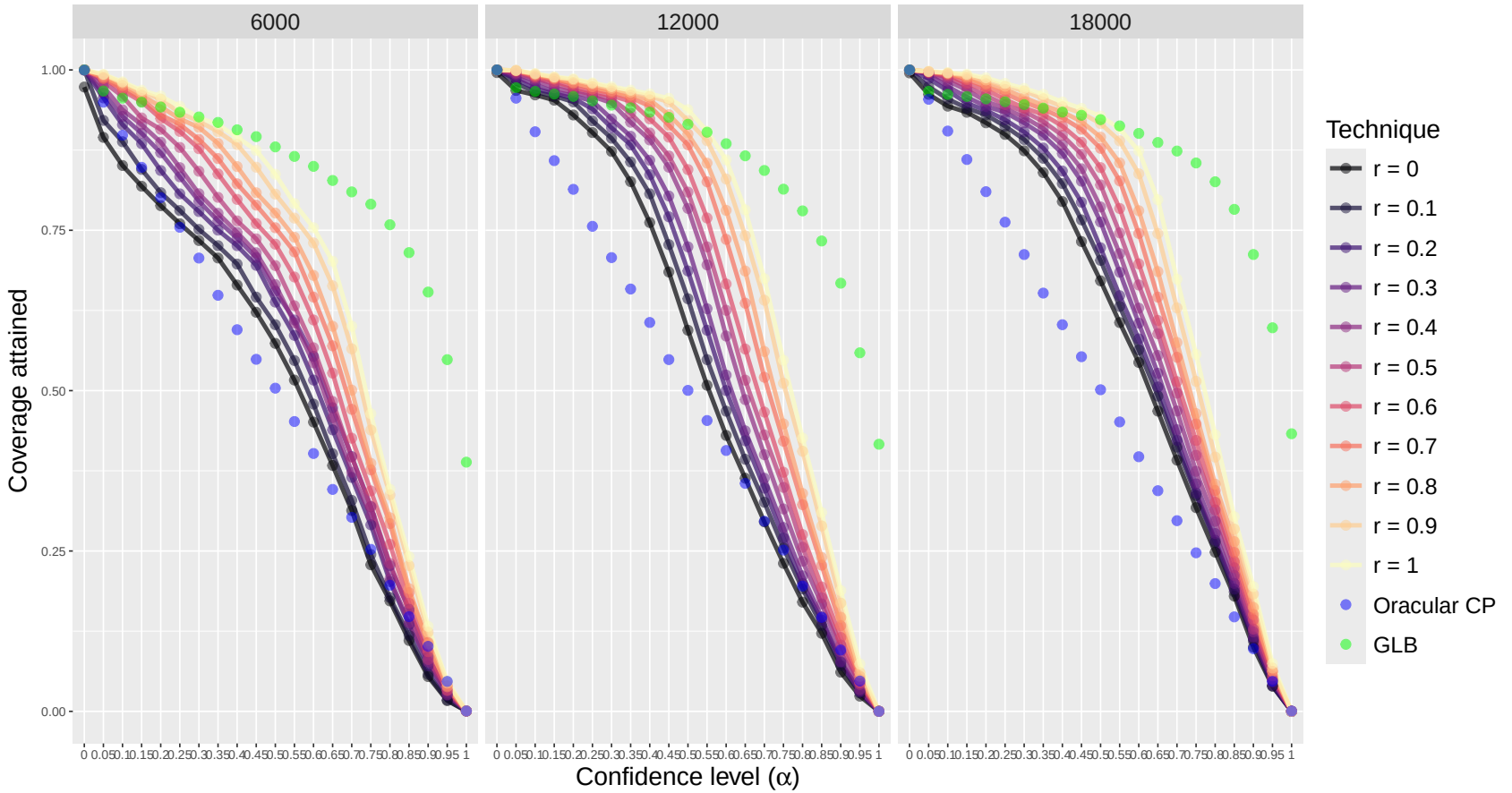


Figure: Marginal coverage for varying levels α. Results compare conformal set-valued policy learning across different randomness levels (r), GLB (green) and Oracular conformal prediction (blue). Columns indicate sample size of training data.

Additional synthetic results (2/3)

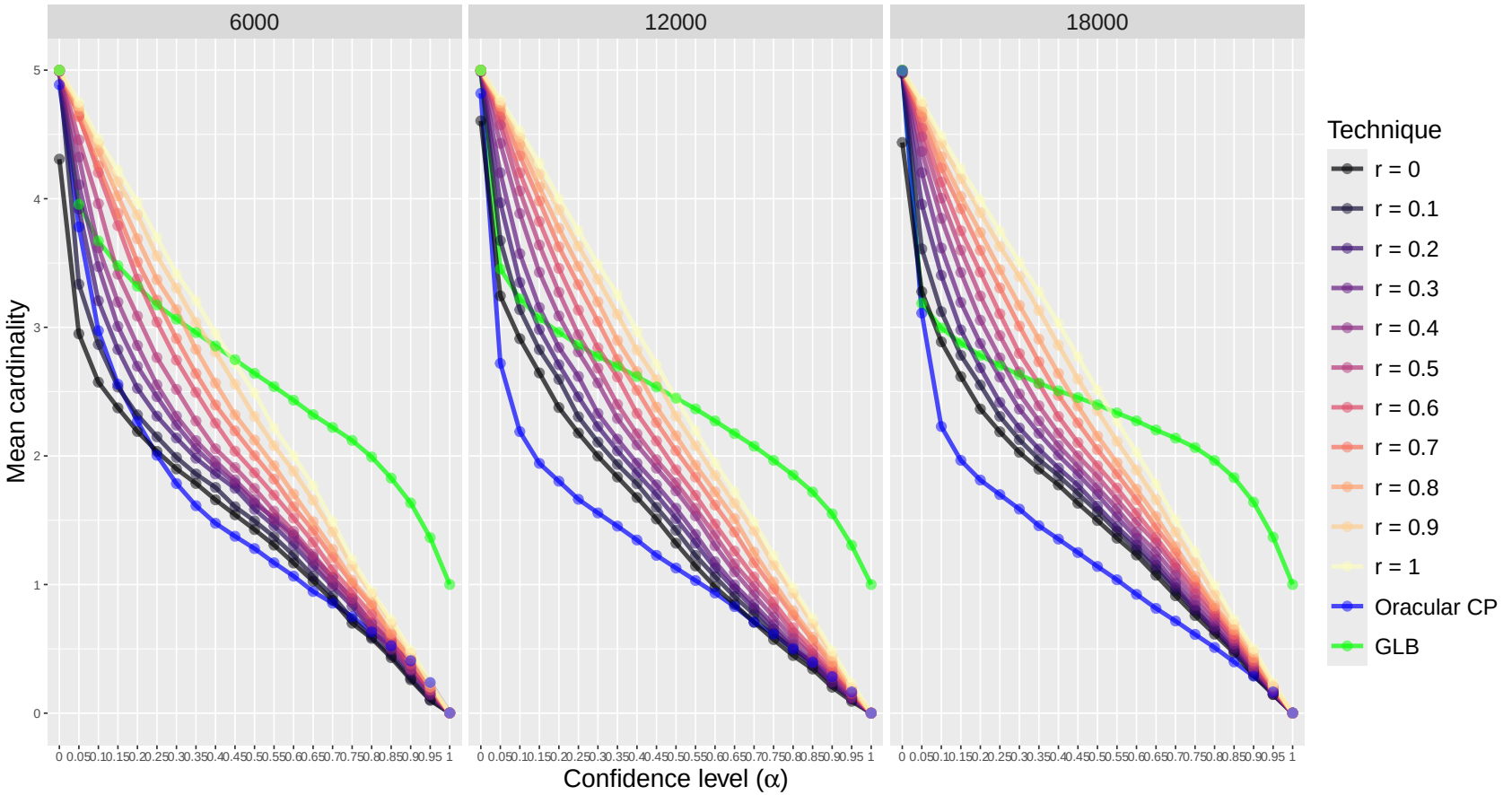


Figure: Mean cardinality for varying levels α. Results compare conformal set-valued policy learning across different randomness levels r, GLB (green) and Oracular conformal prediction (blue). Columns indicate sample size of training data

Additional synthetic results (3/3)

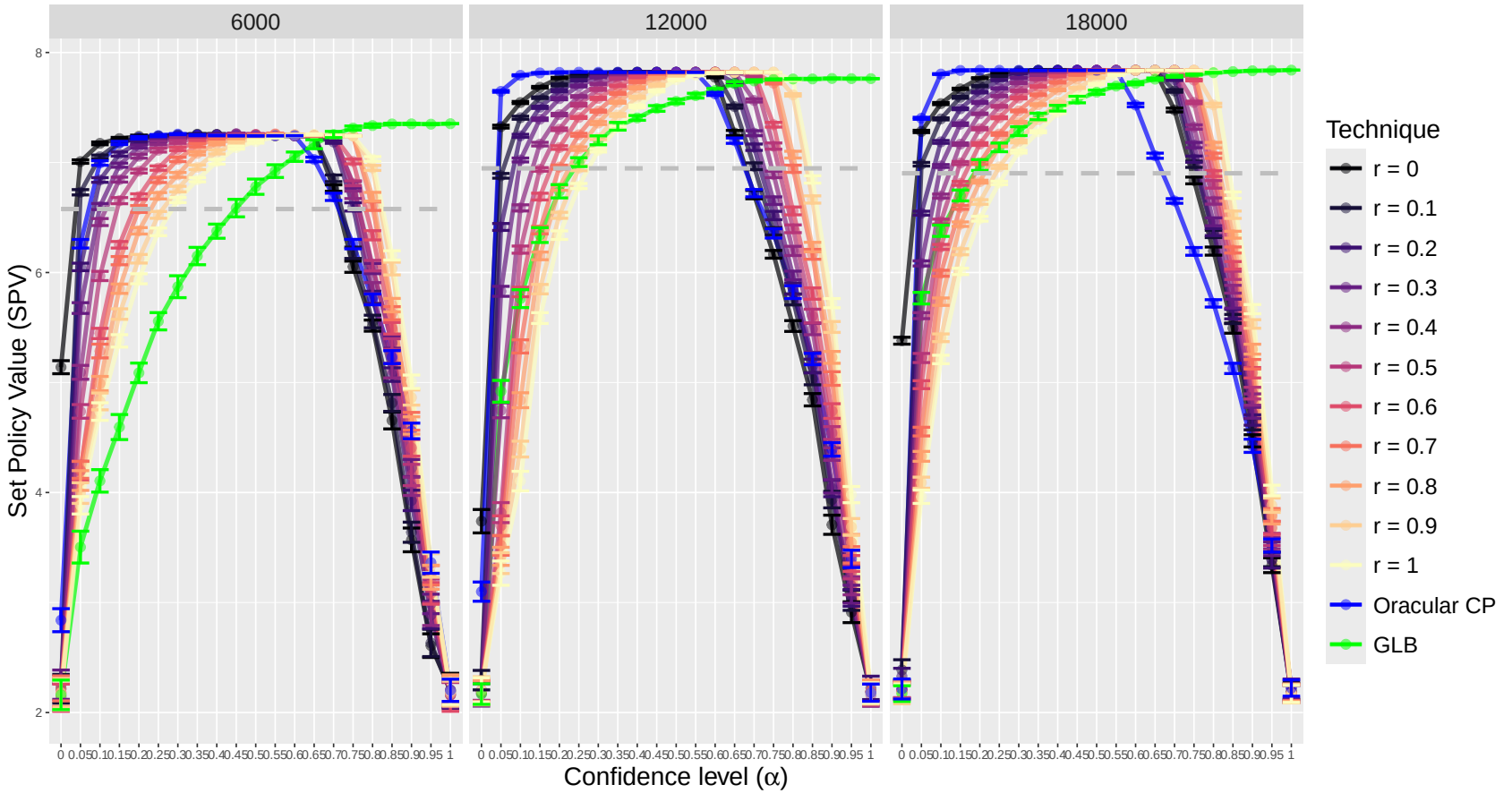


Figure: Uniform set-policy value for varying levels α. Results are shown for conformal set-valued policy learning across different randomness levels r, GLB (green) and Oracular conformal prediction (blue). The gray dashed line represents the policy value achieved by the noisy label generation technique alone.