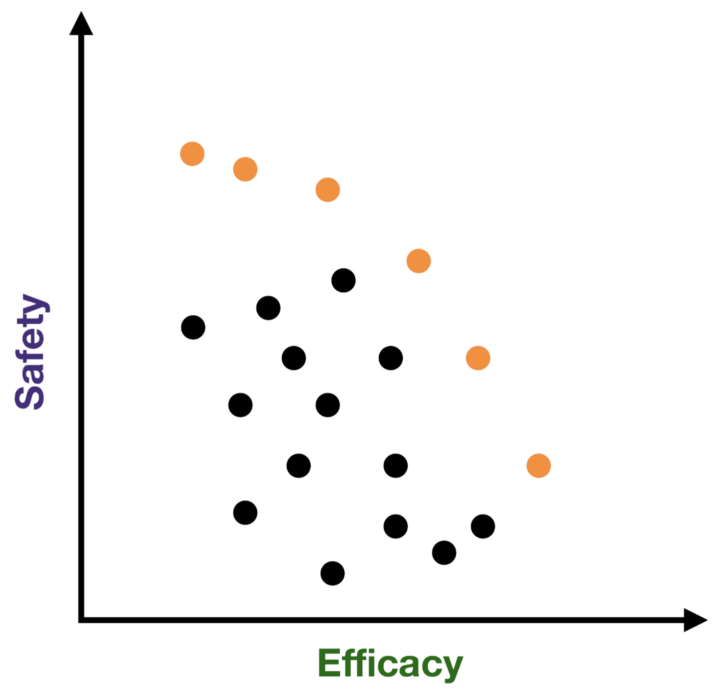
Pareto Set Identification With Posterior Sampling

Cyrille Kone[1], **Marc Jourdan**[2], **Emilie Kaufmann**[1]

[1] Univ. Lille, Inria, CNRS, Centrale Lille, UMR 9198-CRIStAL, F-59000 Lille, France
[2] EPFL, Lausanne, Switzerland

MOTIVATION

Imagine you're testing some vaccines. Some are highly effective. Others are safer. Some are cheap to produce.



⇒ Which ones offer the best trade-off?

- Traditional single-metric approaches fail when trade-offs are unclear
- **Pareto Set:** all solutions that are *not worse in all objectives* compared to another
- Objectives are often **correlated**

PARETO SET IDENTIFICATION

Setting

- Each arm $i \in \{1, \dots, K\}$ is associated with a distribution $\nu_i = N(\mu_i, \boldsymbol{\Sigma})$, where $\mu_i \in \mathbb{R}^d$ and $\boldsymbol{\Sigma} \in \mathbb{R}^{d\times d}$ is a shared covariance matrix (possibly correlated)
- The agent interacts with the arms in a **sequential manner**, selecting one arm A_t at each time step and observing a sample $X_t \sim \nu_{A_t}$

Our goal: Identify the **Pareto-optimal set** of arms:

$$S^\star(\boldsymbol{\mu}) := \{i \in [K] : \nexists j \neq i \text{ such that } \mu_j \succ \mu_i\}$$

where $\mu_j \succ \mu_i$ means μ_j dominates μ_i in all objectives.

With δ-PAC guarantees: Return $S^\star$ with probability at least $1-\delta$, using as few samples as possible.

Applications: clinical trials, large-scale recommender systems, software and hardware design, hyper-parameter optimization etc.

Contributions :

1. Efficient and asymptotically optimal policy for PSI with correlations
2. Supports structured, correlated, transductive bandit settings
3. Opens path for generalizing posterior-sampling-based pure exploration.

STATISTICAL COMPLEXITY

Sample complexity lower bound: for any δ-PAC policy for PSI, $\mathbb{E}_{\boldsymbol{\nu}}[\tau] \geqslant (\Gamma^\star_{\boldsymbol{\nu}})^{-1}\log(1/(2.4\delta))$, where

$$\Gamma^\star_{\boldsymbol{\nu}} := \sup_{w\in\triangle} \inf_{\boldsymbol{\lambda}\in Alt(S^\star(\boldsymbol{\mu}))} \sum_{k=1}^K \frac{w_k}{2}\|\mu_k - \lambda_k\|^2_{\boldsymbol{\Sigma}^{-1}}$$

- $\triangle$ is the probability simplex
- $Alt(S^\star(\boldsymbol{\mu})) := \{\boldsymbol{\lambda} \in \mathbb{R}^{K\times d} \mid S^\star(\boldsymbol{\lambda}) \neq S^\star(\boldsymbol{\mu})\}$ is the set of alternative (incorrect) answers
- $w^\star(\boldsymbol{\mu})$: set maximizers of $\Gamma^\star_{\boldsymbol{\nu}}$ are *optimal pulls allocations*

⇒ an algorithm is *asymptotically optimal* if its sample complexity matches that lower bound, namely if $\limsup_{\delta\to 0} \mathbb{E}_{\boldsymbol{\nu}}[\tau]/\log(1/\delta) \leqslant \Gamma^\star(\boldsymbol{\theta})^{-1}$

Computational challenge

- Finding $w^\star(\boldsymbol{\mu})$ requires oracle $\mathfrak{O}_{\boldsymbol{\mu}} : w \in \triangle \mapsto \operatorname*{argmin}_{\boldsymbol{\lambda}\in Alt(S^\star(\boldsymbol{\mu}))} \sum_{k=1}^K \frac{w_k}{2}\|\mu_k - \lambda_k\|^2_{\boldsymbol{\Sigma}^{-1}}$
- $Alt(S^\star(\boldsymbol{\theta}))$ is not convex but only countably convex
- For $\boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I}_d$, $\mathfrak{O}_{\boldsymbol{\mu}}$ can be computed by solving $\mathcal{O}(K|S^\star(\boldsymbol{\mu})|^d)$ separably convex problems (cf Crepon et al [2024])

⚠ **No efficient oracle when $\boldsymbol{\Sigma}$ is non-diagonal**

OUR ALGORITHM

Main Idea: Interpret the lower bound as the *saddle-point* of a two-player game

$$\Gamma^\star_{\boldsymbol{\nu}} = \sup_{w\in\triangle}\inf_{\boldsymbol{\lambda}\in Alt(S^\star(\boldsymbol{\mu}))} \sum_k \frac{w_k}{2}\|\mu_k - \lambda_k\|^2_{\boldsymbol{\Sigma}^{-1}}$$

- Use **no-regret learners** for both players:
  - $\mathfrak{A}_\triangle$ for the `sup` player use e.g., AdaHedge
  - $\mathfrak{A}_{Alt}$ for the `inf` use **Continuous Exponential Weights (CEW)** on the Alt

💡 **CEW with squared loss ≈ Gaussian Posterior Sampling**

- α-inflated Gaussian posterior distribution :

$$\boldsymbol{\Pi}^\alpha_t = \otimes_{k=1}^{k=K} N(\hat{\mu}_{t,k}, \alpha\boldsymbol{\Sigma}/N_{t,k}) \text{ and } S_t = S^\star(\hat{\boldsymbol{\mu}}_t)$$

Algorithm 1: `PSI-PS Sampling Rule`

Initialize: pull each arm once, let $w_{exp} = \mathbb{1}/K$, $\alpha \in (0, 1/2)$
Input: learner $\mathfrak{A}_\triangle$ *and stopping rule* `PS_Stopping`
Set $t \leftarrow K+1$
while *not* `PS_Stopping` $(t-1)$ **do**
  Sample $\boldsymbol{\lambda}^\eta_t \sim \mathrm{Trunc}(\boldsymbol{\Pi}^{(1/\eta_t)}_t, Alt(S_t))$
  Get w_t from $\mathfrak{A}_\triangle(\mathcal{H}_{t-1})$
  Pull $A_t \sim (1-\gamma_t)w_t + \gamma_t w_{\text{exp}}$, $\gamma_t = t^{-\alpha}$
  Update $\mathfrak{A}_\triangle$ with bonuses $U_t : k \mapsto \left\|\hat{\mu}_{t,k} - \lambda^\eta_{t,k}\right\|^2_{\boldsymbol{\Sigma}^{-1}}$
  Increment t
return S_t

- When to stop collecting new samples?

Key idea : Stop when the posterior concentrates

Algorithm 2: `PS_Stopping`(t)

Input: $\mathcal{H}_{t-1}$, risk $\delta \in (0,1)$ budget and inflation $M(t,\delta)$ and $c(t,\delta)$
for $m = 1$ *to* $M(t,\delta)$ **do**
  Sample $\boldsymbol{\lambda}^m_t \sim \boldsymbol{\Pi}^{c(t,\delta)}_t$
  if $S^\star(\boldsymbol{\lambda}^m_t) \neq S^\star(\hat{\boldsymbol{\mu}}_t)$ **then**
    break and **return** `false`
return `true`

- **Stopping time** : $\tau_{\text{PS}} := \inf\{t \geqslant K : \texttt{PS\_Stopping}(t) \text{ is } true\}$

Lemma

For $c(t,\delta) \approx 1 + \dfrac{dK\log\log(t)}{\log(1/\delta)}$ and $M(t,\delta) \approx r_{\boldsymbol{\Sigma},\delta}(|S_t|)\dfrac{\log(t/\delta)}{\delta}$ with $r_{\boldsymbol{\Sigma},\delta}(n) = \mathcal{O}(\mathrm{poly}_n(\sqrt{(\log(1/\delta)}), |||\boldsymbol{\Sigma}^{-1}|||))$ the PS stopping is δ-PAC for any $\delta \in (0,1)$

Theorem : PS-PSI Optimality

- Sample complexity:

$$\lim_{\delta\to 0}\frac{\mathbb{E}_{\boldsymbol{\nu}}[\tau_{\text{PS}}]}{\log(1/\delta)} = 1/\Gamma^\star_{\boldsymbol{\nu}}$$

- Posterior contraction : with probability 1

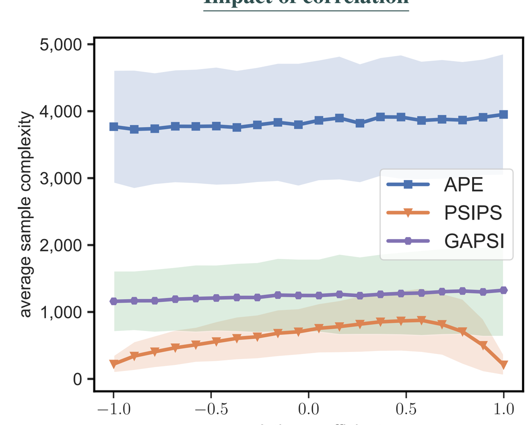
$$\lim_{t\to+\infty} -t^{-1}\log\boldsymbol{\Pi}_t(Alt(S^\star(\boldsymbol{\mu}))) = \Gamma^\star_{\boldsymbol{\nu}}$$

Key advantages :

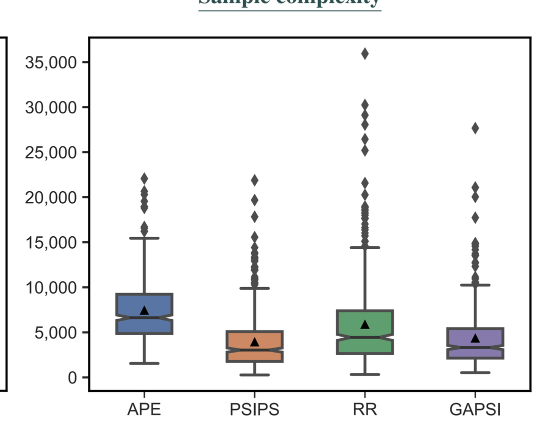
- Asymptotically optimal algorithm for PSI w/o correlation
- Exploration and sampling relies on posterior sampling and online learning
- Sample efficient and computationally efficient

EXPERIMENTAL RESULTS

Impact of correlation



Sample complexity



Runtime