Meta-Learning Hypothesis Spaces for Sequential Decision-making

Parnian Kassraie, Jonas Rothfuss, Andreas Krause



ETH zürich
Learning &
Adaptive Systems

Confidence Sets for Sequential Decision-making

Interact with an environment via

actions $\quad \sigma^2$ sub-Gaussian

noisy observ. $y_t = f(\boldsymbol{x}_t) + \varepsilon_t$

transition dynamics, rewards ...

unknown $\quad f : \mathcal{X} \to \mathbb{R} \qquad f \in \mathcal{H}_{k^*} \qquad \|f\|_{k^*} \leq B$

$\mathcal{X} \subset \mathbb{R}^{d_0}$, compact

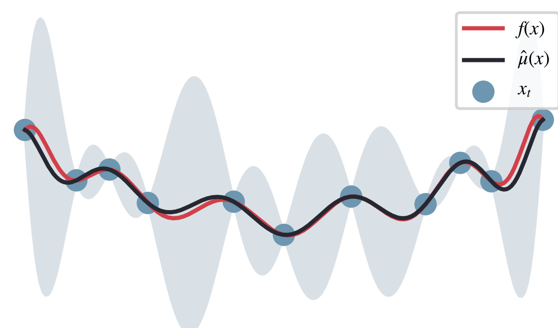
Receive a reward

Repeat

explore

exploit

Confidence sets are great for guiding explorations!



width $\longleftrightarrow$ current uncertainty

center $\longleftrightarrow$ current knowledge

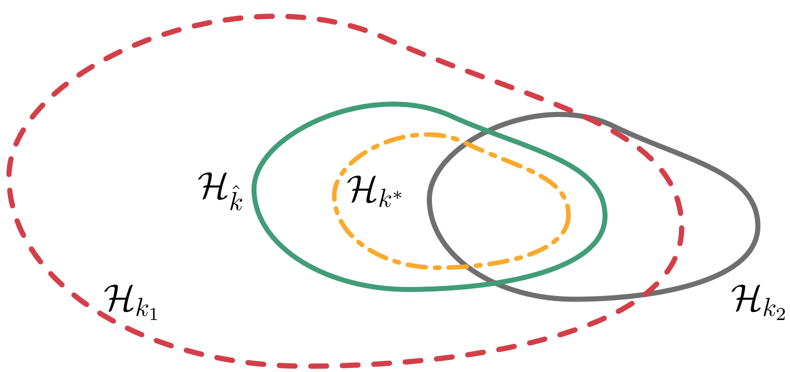
Hypothesis Spaces and Confidence Sets

We commit to confidence sets of the form,

$$\mathcal{C}_{t-1}(k;\boldsymbol{x}) = [\mu_{t-1}(k;\boldsymbol{x}) \pm \nu_t \sigma_{t-1}(k;\boldsymbol{x})]$$

The sets $\mathcal{C}_{t-1}(k;\boldsymbol{x})$ are any-time valid if $\mathbb{P}\left(\forall \boldsymbol{x} \in \mathcal{X},\, \forall t \geq 1 :\; f(\boldsymbol{x}) \in \mathcal{C}_{t-1}(k;\boldsymbol{x})\right) \geq 1-\delta$

Scenarios:

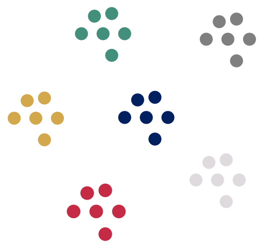


$\mathcal{C}_{t-1}(k_2;\boldsymbol{x})$	Invalid
$\mathcal{C}_{t-1}(k_1;\boldsymbol{x})$	Valid but too wide
$\mathcal{C}_{t-1}(\hat{k};\boldsymbol{x})$	Valid and tight
$\mathcal{C}_{t-1}(k^*;\boldsymbol{x})$	True sets (Valid)

How can we find a good $\mathcal{H}_{\hat{k}}$?

Meta-learning $\mathcal{H}_{k^*}$

When data from similar tasks is available,



$$y_{s,i} = f_s(\boldsymbol{x}_{s,i}) + \varepsilon_{s,i}$$

$$f_s \in \mathcal{H}_{k^*}$$

$$1 \le i \le n \text{ and } 1 \le s \le m$$

And kernel has an additive structure,

$$k^*(\boldsymbol{x}, \boldsymbol{x}') = \sum_{j=1}^{p} \eta_j^* k_j(\boldsymbol{x}, \boldsymbol{x}')$$

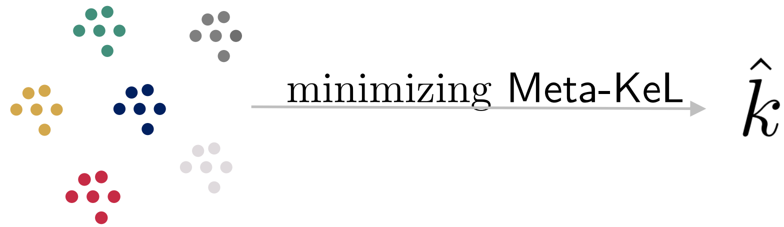
this holds for all Mercer kernels

We propose:

$$\min_{\boldsymbol{\eta}, f_1, \dots, f_m} \frac{1}{m} \sum_{s=1}^{m} \left[\frac{1}{n} \sum_{i=1}^{n} \left(y_{s,i} - f_s(\boldsymbol{x}_{s,i}) \right)^2 \right] + \frac{\lambda}{2} \sum_{s=1}^{m} \| f_s \|_k^2 + \frac{\lambda}{2} \| \boldsymbol{\eta} \|_1$$

$$\text{s.t. } \forall s : f_s \in \mathcal{H}_k,\ k = \sum_{j=1}^{p} \eta_j k_j, 0 \le \boldsymbol{\eta} \qquad \text{(META-KEL)}$$

Properties of the meta-learned kernel

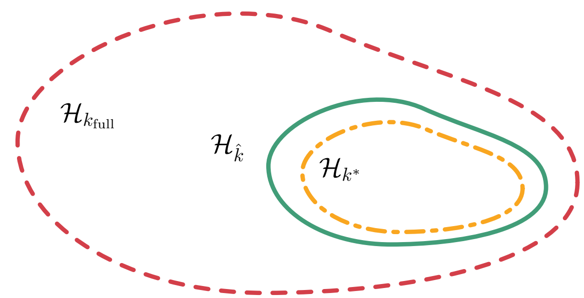


Theorem (Informal)

Under mild regularity assumptions on the meta-data, with probability greater than $1-\delta$,

- $\hat{k}$ *is sparse (in the sense of* $\|\eta\|_1$*)* $\quad k^*(\boldsymbol{x}, \boldsymbol{x}') = \sum_{j=1}^{p} \eta_j^* k_j(\boldsymbol{x}, \boldsymbol{x}')$
- $\mathcal{H}_{k^*} \subseteq \mathcal{H}_{\hat{k}}$
- *For* $f \in \mathcal{H}_{k^*}$*:*

$$\mathbb{P}\left(\forall \boldsymbol{x} \in \mathcal{X}, \forall t \geq 1 :\ f(\boldsymbol{x}) \in \mathcal{C}_{t-1}(\hat{k}; \boldsymbol{x})\right) \geq 1-\delta.$$

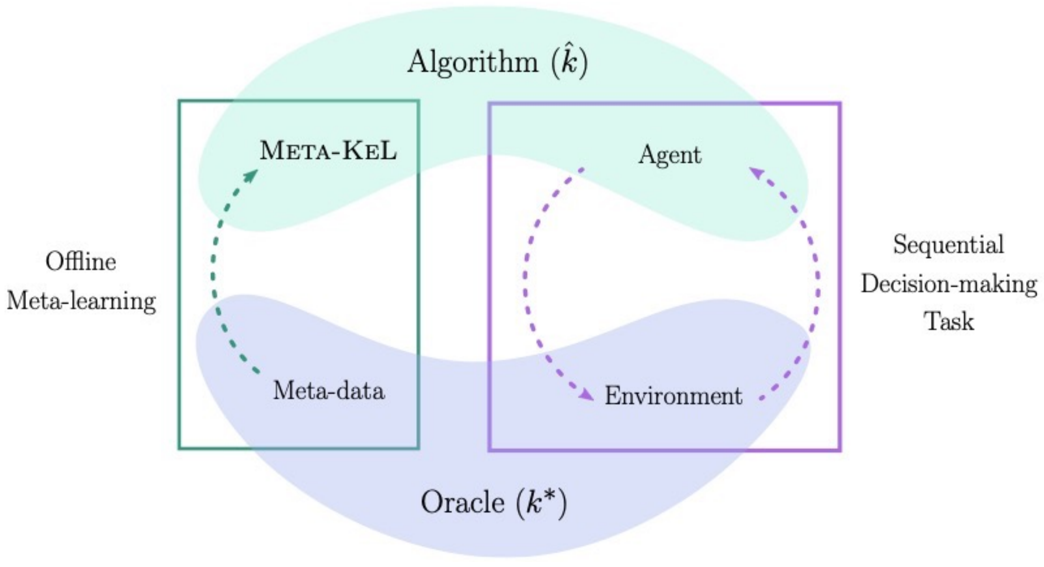


$$k_{\text{full}}(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{p} \sum_{j=1}^{p} k_j(\boldsymbol{x}, \boldsymbol{x}')$$

The meta-learned confidence bounds approach the oracle bounds,
as amounts of offline data grows.

Decision-making with the meta-learned kernel

Plug and Play,



Applications	Bandits
	Bayesian Optimization
	Safe BO
	Model-Based RL

Example: Bayesian Optimization

f is the objective function of a BO problem.

Regret $\quad R_T = \sum_{t=1}^{T}[f(\boldsymbol{x}^*) - f(\boldsymbol{x}_t)]$ $\quad$ Goal $\quad R_T/T \to 0$ as $T \to \infty$

Policy: [GP-UCB, Srinivas et al.]

$$\boldsymbol{x}_t = \arg\max_{\boldsymbol{x}\in\mathcal{X}} \mathcal{C}_{t-1}(\hat{k};\boldsymbol{x})$$

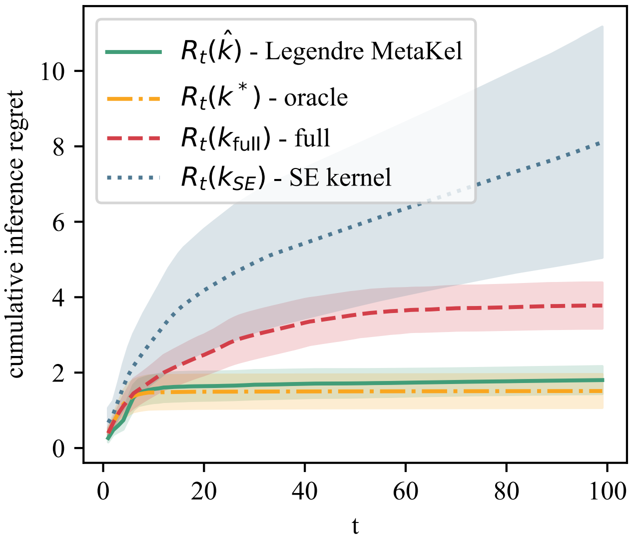
Corollary

Provided that there is enough meta-data,

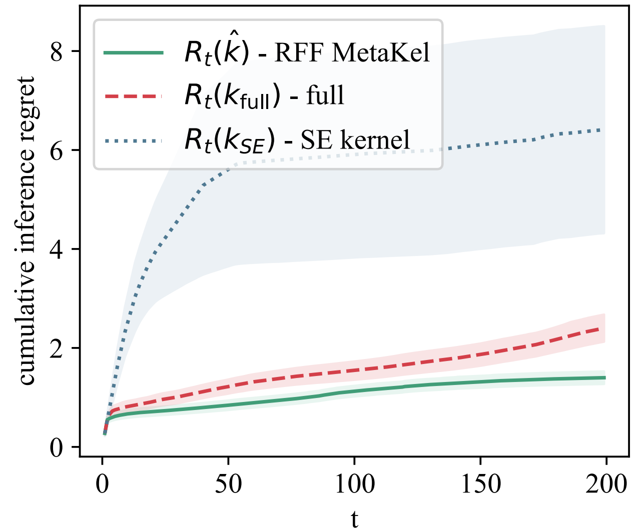
- *The learner achieves sublinear regret, w.h.p.*
- *This guarantee is tight compared to the one for the Oracle learner, and approaches it at a $\mathcal{O}(1/\sqrt{mn})$ rate.*

Experiments: BO with Meta-KeL

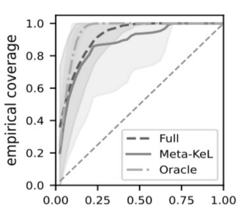
2D synthetic data Legendre features

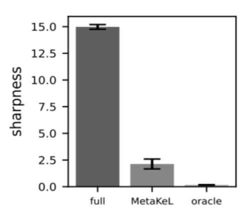


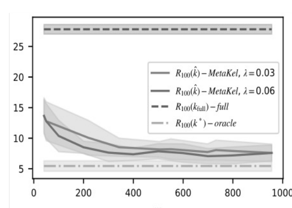
GLMNET data [Friedman et al 2010] + RFF

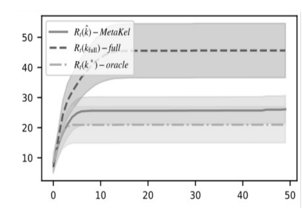


Checkout the paper for more









Thank you.