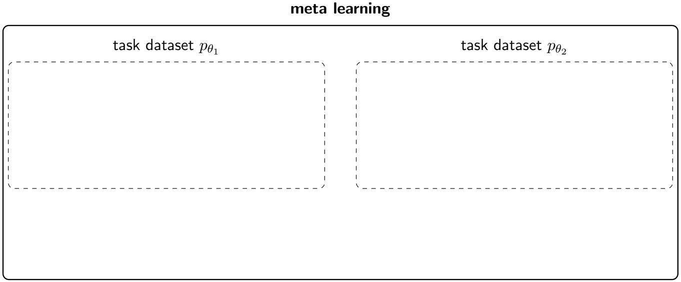
PAC Prediction Sets for Meta-Learning

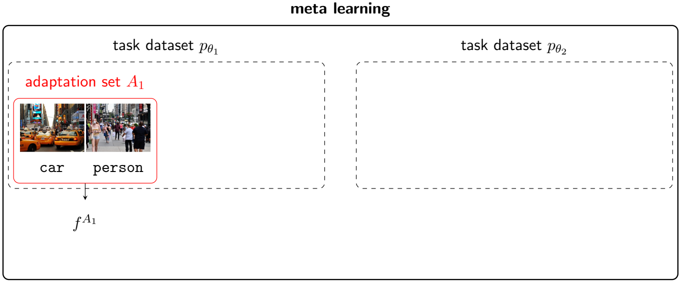
Sangdon Park, Edgar Dobriban, Insup Lee, and Osbert Bastani

NeurIPS 2022

Meta-Learning: Learning a Predictor for Fast Adaptation

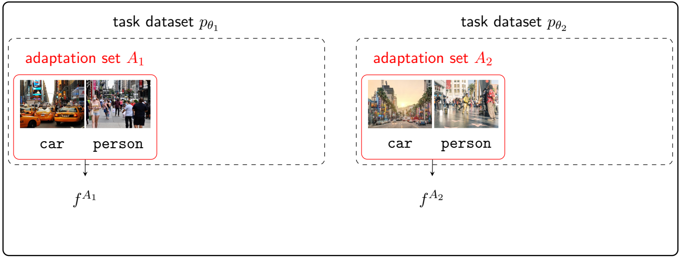


Meta-Learning: Learning a Predictor for Fast Adaptation

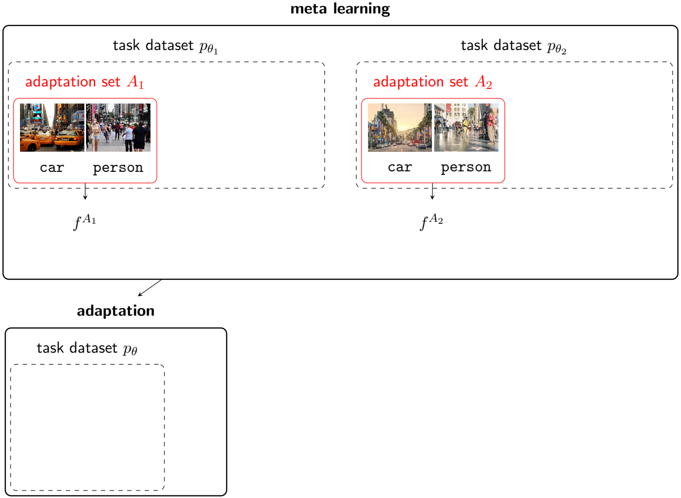


Meta-Learning: Learning a Predictor for Fast Adaptation

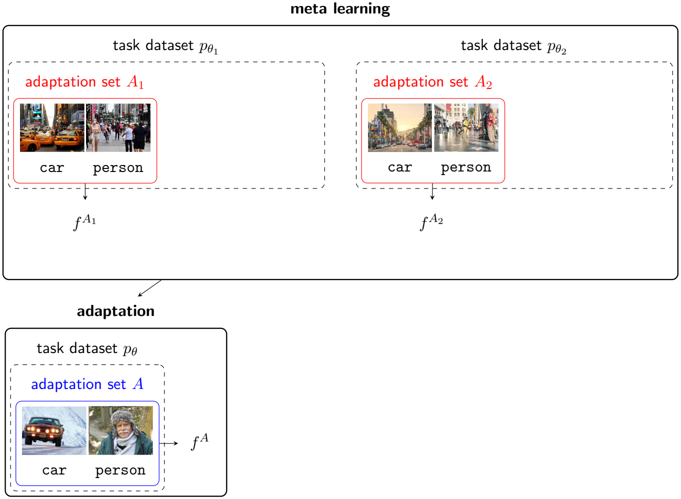
meta learning



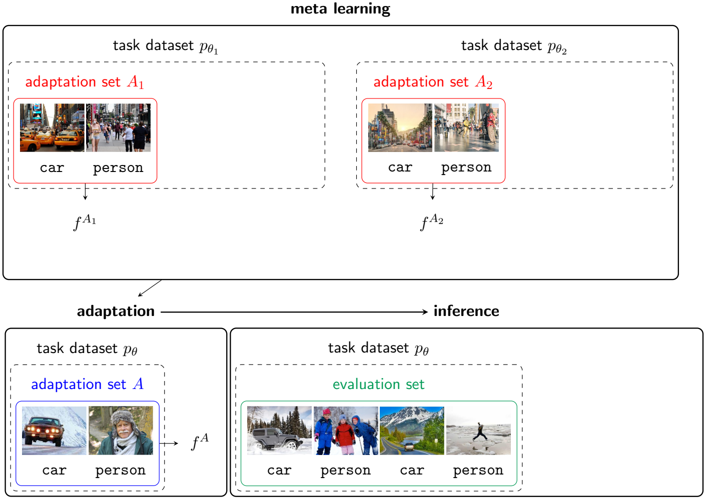
Meta-Learning: Learning a Predictor for Fast Adaptation



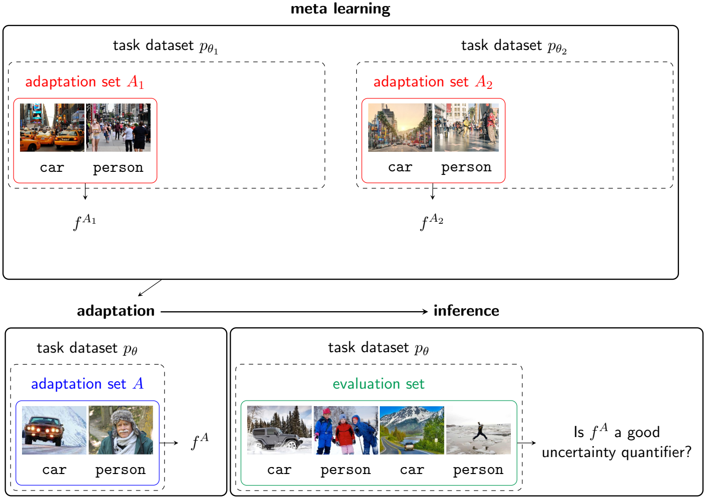
Meta-Learning: Learning a Predictor for Fast Adaptation



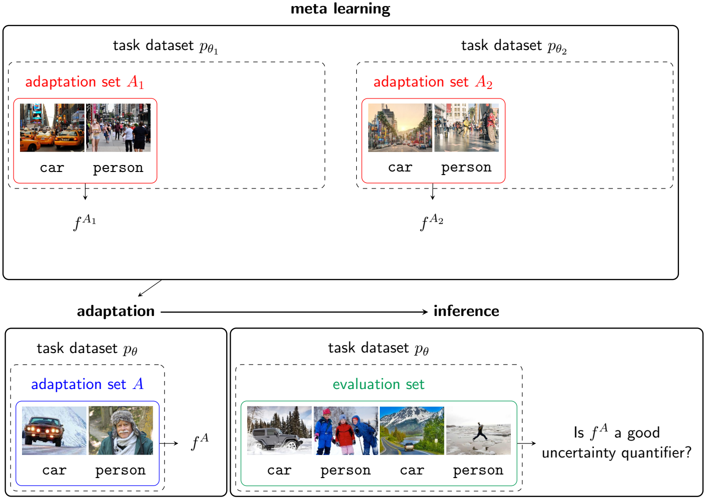
Meta-Learning: Learning a Predictor for Fast Adaptation



Meta-Learning: Learning a Predictor for Fast Adaptation



Meta-Learning: Learning a Predictor for Fast Adaptation



No guarantee on quantified uncertainty by the meta-learned and adapted model.

Quantify Provable Uncertainty via PAC Prediction Sets

A **PAC prediction set** (Wilks, 1941; Vovk, 2013; Park et al., 2020) is a **prediction set** that comes with the **probably approximately correct (PAC) guarantee** (Valiant, 1984).

Quantify Provable Uncertainty via PAC Prediction Sets

A **PAC prediction set** (Wilks, 1941; Vovk, 2013; Park et al., 2020) is a **prediction set** that comes with the **probably approximately correct (PAC) guarantee** (Valiant, 1984).

A **prediction set** is a set-valued predictor

$$F_\tau(x) = \{y \in \mathcal{Y} \mid f(x, y) \geq \tau\},$$

where a conformity score function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ and a parameter $\tau \in \mathbb{R}_{\geq 0}$ are given.

Quantify Provable Uncertainty via PAC Prediction Sets

A **PAC prediction set** (Wilks, 1941; Vovk, 2013; Park et al., 2020) is a **prediction set** that comes with the **probably approximately correct (PAC) guarantee** (Valiant, 1984).

A parameter $\tau \in \mathbb{R}_{\geq 0}$ is ε-**correct** if

$$\mathbb{P}_\theta\Big\{Y \in F_\tau(X)\Big\} \geq 1 - \varepsilon,$$

where the probability is taken over $(X, Y) \sim p_\theta$.

We denote a set of all ε-correct taus by $T_\varepsilon(\theta)$.

Quantify Provable Uncertainty via PAC Prediction Sets

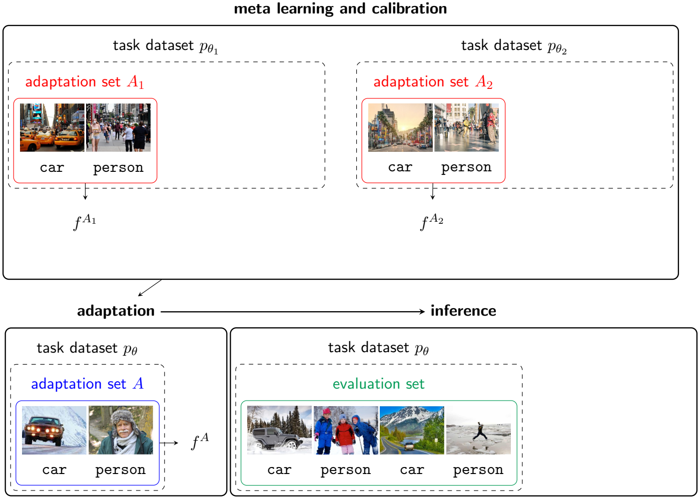
A **PAC prediction set** (Wilks, 1941; Vovk, 2013; Park et al., 2020) is a **prediction set** that comes with the **probably approximately correct (PAC) guarantee** (Valiant, 1984).

An estimator $\hat{\gamma}_{\varepsilon,\delta} : (\mathcal{X} \times \mathcal{Y})^* \to \mathbb{R}_{\geq 0}$ is (ε, δ)-**probably approximately correct (PAC)** if

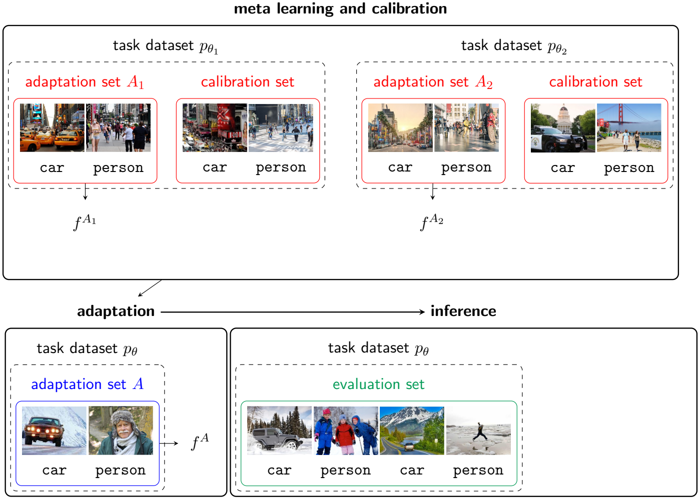
$$\mathbb{P}\left\{ \hat{\gamma}_{\varepsilon,\delta}(S) \in T_\varepsilon(\theta) \right\} \geq 1 - \delta,$$

where the probability is taken over a calibration set $S \in (\mathcal{X} \times \mathcal{Y})^*$.

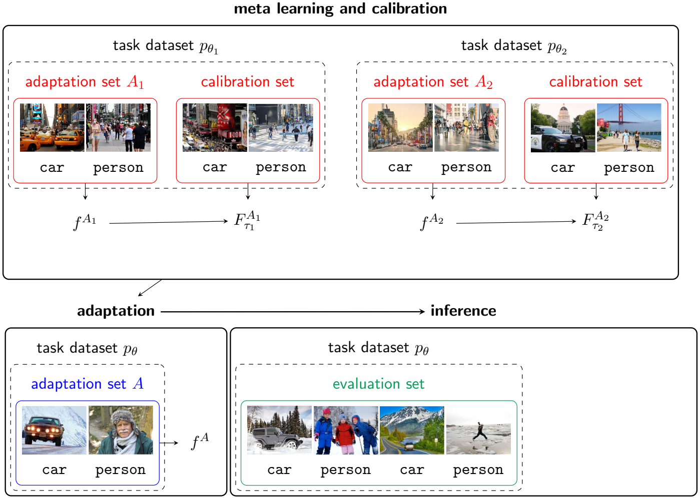
Meta-Calibration: Learning PAC Prediction Sets for Fast Adaptation



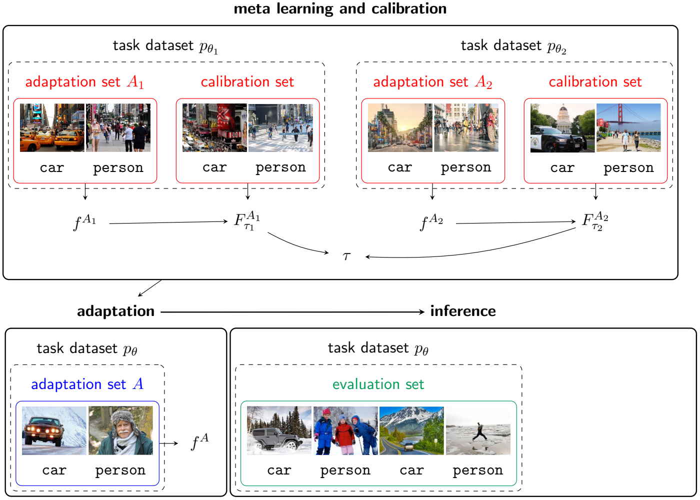
Meta-Calibration: Learning PAC Prediction Sets for Fast Adaptation



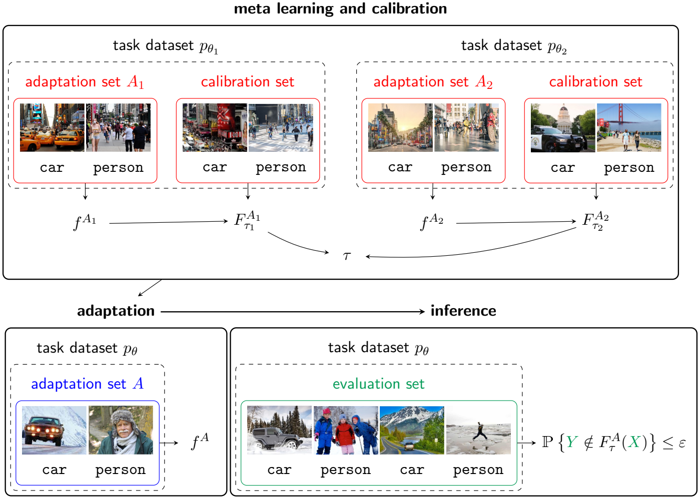
Meta-Calibration: Learning PAC Prediction Sets for Fast Adaptation



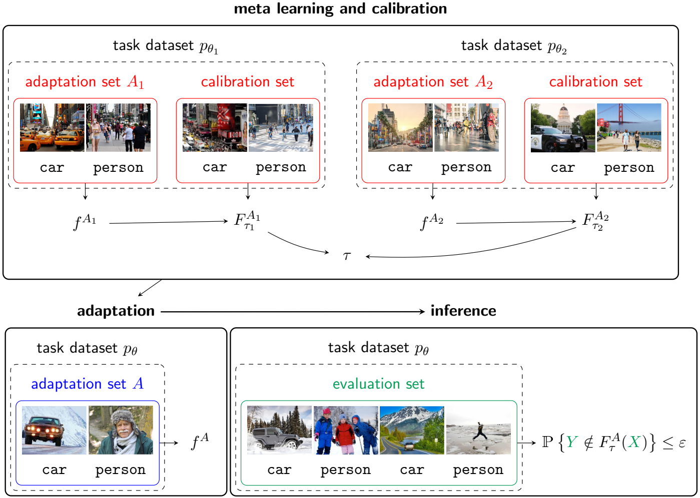
Meta-Calibration: Learning PAC Prediction Sets for Fast Adaptation



Meta-Calibration: Learning PAC Prediction Sets for Fast Adaptation



Meta-Calibration: Learning PAC Prediction Sets for Fast Adaptation



Three sources of randomness in the (1) meta calibration, (2) adaptation, and (3) evaluation.

Problem: PAC Prediction Sets for Meta-Learning

We control the three sources of randomness via three parameters δ, α, and ε.

Problem: PAC Prediction Sets for Meta-Learning

We control the three sources of randomness via three parameters δ, α, and ε.

An estimator $\hat{\tau} : \boldsymbol{\mathcal{S}} \times \boldsymbol{\mathcal{A}} \to \mathbb{R}_{\geq 0}$ is $(\varepsilon, \alpha, \delta)$-**meta PAC (mPAC)** if

$$\hat{\tau}(\boldsymbol{S}, \boldsymbol{A}) \in T_{\varepsilon}(\theta, A)$$

where the first probability is taken over $(\boldsymbol{S}, \boldsymbol{A})$ and the second probability is taken over (θ, A).

Problem: PAC Prediction Sets for Meta-Learning

We control the three sources of randomness via three parameters δ, α, and ε.

An estimator $\hat{\tau} : \boldsymbol{\mathcal{S}} \times \boldsymbol{\mathcal{A}} \to \mathbb{R}_{\geq 0}$ is $(\varepsilon, \alpha, \delta)$-**meta PAC (mPAC)** if

$$\mathbb{P}\left\{\hat{\tau}(\boldsymbol{S}, \boldsymbol{A}) \in T_{\varepsilon}(\theta, A)\right\} \geq 1 - \alpha$$

where the first probability is taken over $(\boldsymbol{S}, \boldsymbol{A})$ and the second probability is taken over (θ, A).

Problem: PAC Prediction Sets for Meta-Learning

We control the three sources of randomness via three parameters δ, α, and ε.

An estimator $\hat{\tau} : \boldsymbol{\mathcal{S}} \times \boldsymbol{\mathcal{A}} \to \mathbb{R}_{\geq 0}$ is $(\varepsilon, \alpha, \delta)$-**meta PAC (mPAC)** if

$$\mathbb{P}\left\{\mathbb{P}\left\{\hat{\tau}(\boldsymbol{S}, \boldsymbol{A}) \in T_{\varepsilon}(\theta, A)\right\} \geq 1 - \alpha\right\} \geq 1 - \delta,$$

where the first probability is taken over $(\boldsymbol{S}, \boldsymbol{A})$ and the second probability is taken over (θ, A).

Algorithm (Simplified)

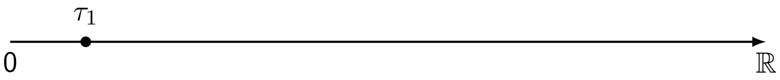
The proposed algorithm **Meta-PS** internally uses any PAC prediction set algorithm $\hat{\gamma}_{\varepsilon,\delta}$ (*e.g.*, Park et al. (2020)).

$$\textbf{Meta-PS}(\mathbf{S}, \mathbf{A}, \varepsilon, \alpha, \delta)$$

Algorithm (Simplified)

The proposed algorithm **Meta-PS** internally uses any PAC prediction set algorithm $\hat{\gamma}_{\varepsilon,\delta}$ (*e.g.*, Park et al. (2020)).

Meta-PS$(\mathbf{S}, \mathbf{A}, \varepsilon, \alpha, \delta)$

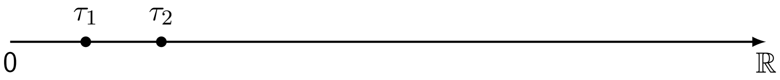


$$\tau_1 = \hat{\gamma}_{\varepsilon,\alpha/2}((S_1, A_1))$$

Algorithm (Simplified)

The proposed algorithm **Meta-PS** internally uses any PAC prediction set algorithm $\hat{\gamma}_{\varepsilon,\delta}$ (*e.g.*, Park et al. (2020)).

$$\textbf{Meta-PS}(\mathbf{S}, \mathbf{A}, \varepsilon, \alpha, \delta)$$

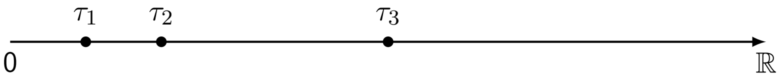


$$\tau_2 = \hat{\gamma}_{\varepsilon,\alpha/2}((S_2, A_2))$$

Algorithm (Simplified)

The proposed algorithm **Meta-PS** internally uses any PAC prediction set algorithm $\hat{\gamma}_{\varepsilon,\delta}$ (*e.g.*, Park et al. (2020)).

Meta-PS$(\mathbf{S}, \mathbf{A}, \varepsilon, \alpha, \delta)$

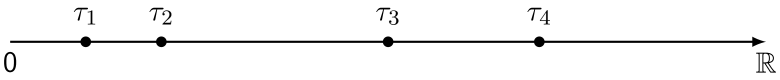


$$\tau_3 = \hat{\gamma}_{\varepsilon,\alpha/2}((S_3, A_3))$$

Algorithm (Simplified)

The proposed algorithm **Meta-PS** internally uses any PAC prediction set algorithm $\hat{\gamma}_{\varepsilon,\delta}$ (*e.g.*, Park et al. (2020)).

Meta-PS$(\mathbf{S}, \mathbf{A}, \varepsilon, \alpha, \delta)$

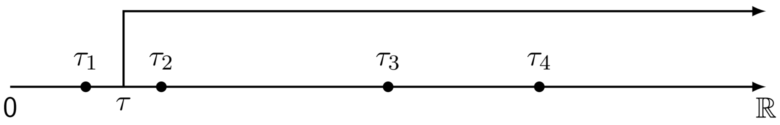


$$\tau_4 = \hat{\gamma}_{\varepsilon,\alpha/2}((S_4, A_4))$$

Algorithm (Simplified)

The proposed algorithm **Meta-PS** internally uses any PAC prediction set algorithm $\hat{\gamma}_{\varepsilon,\delta}$ (*e.g.*, Park et al. (2020)).

Meta-PS$(\mathbf{S}, \mathbf{A}, \varepsilon, \alpha, \delta)$

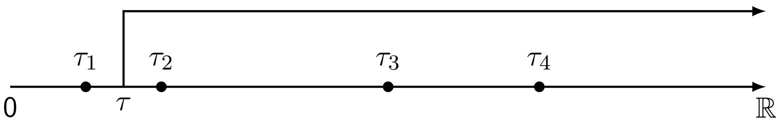


$$\tau = \hat{\gamma}_{\alpha/2,\delta}((\tau_1, 1), (\tau_2, 1), (\tau_3, 1), (\tau_4, 1))$$

Algorithm (Simplified)

The proposed algorithm **Meta-PS** internally uses any PAC prediction set algorithm $\hat{\gamma}_{\varepsilon,\delta}$ (*e.g.*, Park et al. (2020)).

Meta-PS$(\mathbf{S}, \mathbf{A}, \varepsilon, \alpha, \delta)$



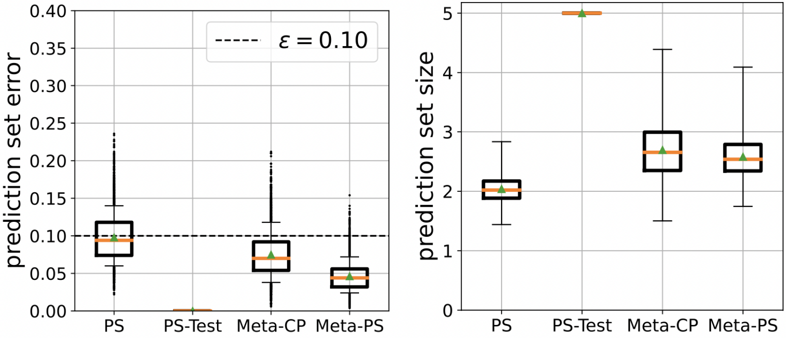
$$\tau = \hat{\gamma}_{\alpha/2,\delta}((\tau_1, 1), (\tau_2, 1), (\tau_3, 1), (\tau_4, 1))$$

Theorem (Meta PAC Guarantee)

The estimator $\hat{\tau}_{\varepsilon,\alpha,\delta}$ implemented by **Meta-PS** *is $(\varepsilon, \alpha, \delta)$-mPAC.*

Results: Mini-ImageNet

We empirically demonstrated the correctness guarantee of our algorithm **Meta-PS**.



(a) Error (b) Size

Figure: $\varepsilon = 0.1$, $\alpha = 0.1$, $\delta = 10^{-5}$ for **Meta-PS** and $\varepsilon = 0.1$, $\delta = 10^{-5}$ for the other methods.

Results: Mini-ImageNet

We empirically demonstrated the correctness guarantee of our algorithm **Meta-PS**.

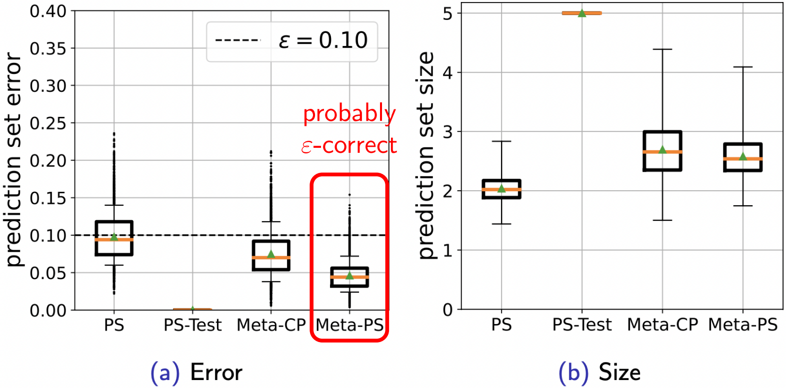
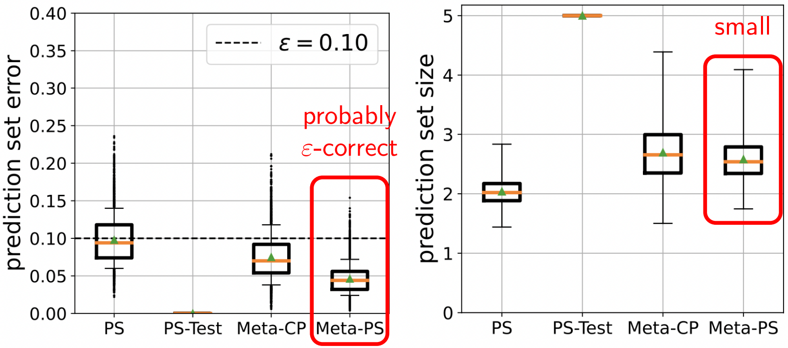


Figure: $\varepsilon = 0.1$, $\alpha = 0.1$, $\delta = 10^{-5}$ for **Meta-PS** and $\varepsilon = 0.1$, $\delta = 10^{-5}$ for the other methods.

Results: Mini-ImageNet

We empirically demonstrated the correctness guarantee of our algorithm **Meta-PS**.



(a) Error (b) Size

Figure: $\varepsilon = 0.1$, $\alpha = 0.1$, $\delta = 10^{-5}$ for **Meta-PS** and $\varepsilon = 0.1$, $\delta = 10^{-5}$ for the other methods.

Conclusion

We proposed a PAC prediction set algorithm for meta learning.

- Controls three sources of uncertainty for our correctness guarantee.
- Evaluated over three application domains: image, language, and medical datasets.
- Code is available: `https://github.com/sangdon/meta-pac-ps`

References

Fisch, A., Schuster, T., Jaakkola, T., and Barzilay, R. (2021). Few-shot conformal prediction with auxiliary tasks.

Park, S., Bastani, O., Matni, N., and Lee, I. (2020). Pac confidence sets for deep neural networks via calibrated prediction. In *International Conference on Learning Representations*.

Valiant, L. G. (1984). A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142.

Vovk, V. (2013). Conditional validity of inductive conformal predictors. *Machine learning*, 92(2-3):349–376.

Wilks, S. S. (1941). Determination of sample sizes for setting tolerance limits. *The Annals of Mathematical Statistics*, 12(1):91–96.