Fast-Rate PAC-Bayesian Generalization Bounds for Meta-Learning

Jiechao Guan[1,3], **Zhiwu Lu**[2,3,*]

[1]School of Information, Renmin University of China, Beijing, China
[2]Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China
[3]Beijing Key Laboratory of Big Data Management and Analysis Methods, Beijing, China

*Corresponding Author {2014200990,luzhiwu}@ruc.edu.cn

July 12, 2022

Presentation Outline

PAC-Bayesian Bounds for Single-Task Learning

Table 1: Notations of PAC-Bayesian single-task learning. The loss $\ell : \mathcal{H} \times \mathcal{Z} \to [0,1]$. $\widehat{er}(Q,S) = \mathbf{E}_{h\sim Q} \frac{1}{m} \sum_{i=1}^{m} \ell(h, z_i)$, $er(Q,D) = \mathbf{E}_{h\sim Q}\mathbf{E}_{z\sim D}\ell(h,z)$. KL-divergences $\mathcal{K}(Q,P) = \mathbf{E}_{h\sim Q} \ln \frac{\mathrm{d}Q}{\mathrm{d}P}$, $\mathrm{kl}(p,q) = p \ln \frac{p}{q} + (1-p) \ln \frac{1-p}{1-q}$.

Unknown Task	$D \in \mathcal{M}_1(\mathcal{Z})$	Hypothesis Space	$\mathcal{H}$
Prior	$P \in \mathcal{M}_1(\mathcal{H})$	Posterior	$Q \in \mathcal{M}_1(\mathcal{H})$
Empirical Error	$\widehat{er}(Q,S)$	Expected Error	$er(Q,D)$

Theorem 1.1

[5] [Corollary 2.1-2.2] Let ℓ be $\{0,1\}$-valued loss. $\forall$ fixed prior P, $\delta, \lambda > 0$, with probability $\geq 1-\delta$ over i.i.d. S, for any posterior $Q \in \mathcal{M}_1(\mathcal{H})$:

$$\mathrm{kl}(\widehat{er}(Q,S), er(Q,D)) \leq \frac{\mathcal{K}(Q,P) + \ln\left(2\sqrt{m}/\delta\right)}{m},$$

$$er(Q,D) \leq \frac{\lambda}{m(1-\mathrm{e}^{-\lambda/m})}\widehat{er}(Q,S) + \frac{\mathcal{K}(Q,P) + \ln(1/\delta)}{m(1-\mathrm{e}^{-\lambda/m})}.$$

PAC-Bayesian Framework for Meta-Learning

Table 2: Notations of PAC-Bayesian meta-learning. The training sample $\mathbf{S} = \{S_i\}_{i=1}^n$, where S_i is the dataset in the i-th training task and is formed by sampling m times from distribution D_i, where $D_i \sim \tau$. $Q(S, P)$ is the output posterior by running algorithm with sample S and prior P as input.

Sample	$S \in \mathcal{Z}^m$
Training Set	$\mathbf{S} = \{S_i\}_{i=1}^n \in (\mathcal{Z}^m)^n$
Task Environment	$\tau \in \mathcal{M}_1(\mathcal{M}_1(\mathcal{Z}))$
Hyper-Prior	$\mathcal{P} \in \mathcal{M}_1(\mathcal{M}_1(\mathcal{H}))$
Hyper-Posterior	$\mathcal{Q} \in \mathcal{M}_1(\mathcal{M}_1(\mathcal{H}))$
Empirical Multi-Task Error	$\widehat{er}(\mathcal{Q}) = \mathbf{E}_{P\sim\mathcal{Q}} 1/n \sum_{i=1}^n \widehat{er}(Q(S_i, P), S_i)$
Expected Multi-Task Error	$\widetilde{er}(\mathcal{Q}) = \mathbf{E}_{P\sim\mathcal{Q}} 1/n \sum_{i=1}^n er(Q(S_i, P), D_i)$
Transfer Error	$er(\mathcal{Q}) = \mathbf{E}_{P\sim\mathcal{Q}} \mathbf{E}_{D\sim\tau} \mathbf{E}_{S\sim D^m} er(Q(S, P), D)$

The goal of PAC-Bayesian meta-learning theory is thus to give a generalization bound on the transfer error $er(\mathcal{Q})$ based on $\widehat{er}(\mathcal{Q})$.

The Limitation of Existing PAC-Bayesian Bounds for Meta-Learning

To give PAC-Bayesian bounds for meta-learning, we need to choose convex function $\mathcal{D}(p, q)$ and then bound the moment generating function (MGF) of $\mathcal{D}(p, q)$ (i.e., $\mathbf{E}\exp\{\mathcal{D}(er(\mathcal{Q}), \widetilde{er}(\mathcal{Q}))\}$ and $\mathbf{E}\exp\{\mathcal{D}(\widetilde{er}(\mathcal{Q}), \widehat{er}(\mathcal{Q}))\}$).

Almost all existing works [8, 9, 10] set $\mathcal{D}(p, q) = p - q$, apply Hoeffding's lemma to bound the MGF of $\mathcal{D}(p, q)$, and finally obtain a PAC-Bayesian meta learning bound of $O(1/t + t/K)(\forall t > 0)$, which suffers a slow convergence rate of $O(1/\sqrt{K})$ $(K > 0)$, where K is the number of observations.

Motivation of Our Fast-Rate PAC-Bayesian Bounds

In contrast, we set $\mathcal{D}(p,q)$ as $\mathrm{kl}(q,p)$ or ${}^1\Phi_{\frac{\lambda}{K}}(p)-q$, $(\lambda>0)$, as what we do to obtain the PAC-Bayesian kl-bound and Catoni-bound in single-task learning. However, since $\widetilde{er}(\mathcal{Q})$ and $\widehat{er}(\mathcal{Q})$ are the summations of independent $[0,1]$-valued random variables (not i.i.d. $\{0,1\}$-valued ones as in Theorem 1.1), we can not directly apply the results in Theorem 1.1 to bound the MGF of $\mathcal{D}(p,q)$. To overcome this challenge, we use the following lemma to bound the expectation of the function of the sum of independent $[0,1]$-valued random variables (rvs) with the expectation of the function of the sum of i.i.d. $\{0,1\}$-valued ones. Such result is originated from [2].

Lemma 2.1

Let $\{\xi_k\}_{k=1}^K$ be a sequence of independent random variables with $P(0\le\xi_k\le 1)=1$, and $\{\eta_k\}_{k=1}^K$ be a sequence of i.i.d. Bernoulli random variables with $\mathbf{E}\eta_k=K^{-1}(\sum_{k=1}^K\mathbf{E}\xi_k)$. Then for any convex function g,

$$\mathbf{E}g\Big(\frac{1}{K}\sum_{k=1}^K\xi_k\Big)\le\mathbf{E}g\Big(\frac{1}{K}\sum_{k=1}^K\eta_k\Big).$$

Extending PAC-Bayesian kl-Bound and Catoni-Bound to the Independent Setting

Theorem 2.2

Let $\mathcal{F}$ be a set of rvs f. Let $\mathcal{S} = \{\xi_k\}_{k=1}^K$ be a sequence of random variables with each component ξ_k ($k \in [K]$) drawn independently according to the measure μ_k over the set A_k. Let $R(f) = \frac{1}{K}\sum_{k=1}^K \mathbf{E}_{\xi_k} g_k(f, \xi_k)$, $r(f) = \frac{1}{K}\sum_{k=1}^K g_k(f, \xi_k)$, where $g_k : \mathcal{F} \times A_k \to [0,1]$ is a bounded function. Denote $\mathbf{E}_{f\sim\rho}(R(f)), \mathbf{E}_{f\sim\rho}(r(f))$ by $\rho(R), \rho(r)$ respectively. Then $\forall \delta > 0, \lambda > 0$, $\forall$ fixed $\pi \in \mathcal{M}_1(\mathcal{F})$, with probability $\geq 1-\delta$ over $\mathcal{S}$, the following holds for any measure ρ over $\mathcal{F}$:

$$\mathrm{kl}(\rho(r), \rho(R)) \leq \frac{\mathcal{K}(\rho, \pi) + \ln\left(2\sqrt{K}/\delta\right)}{K},$$

$$\rho(R) \leq \frac{\lambda\rho(r)}{K(1-\mathrm{e}^{-\lambda/K})} + \frac{\mathcal{K}(\rho, \pi) + \ln(1/\delta)}{K(1-\mathrm{e}^{-\lambda/K})}.$$

Proof Sketch. Note that $\mathcal{D}(\rho(R), \rho(r)) \leq \frac{1}{\lambda}\left[\mathcal{K}(\rho, \pi) + \ln \mathbf{E}_{f\sim\pi}\mathbf{E}_{\mathcal{S}}\mathrm{e}^{\lambda\mathcal{D}(R(f), r(f))}/\delta\right]$ holds with high probability for any convex function $\mathcal{D}(\cdot,\cdot)$. With Lemma 2.1 we can bound $\mathbf{E}_{\mathcal{S}}\mathrm{e}^{\lambda\mathcal{D}(R(f), r(f))}$ with the MGF of convex function of the sum of i.i.d. Bernoulli rvs. Setting $\mathcal{D}(p, q)$ as $\mathrm{kl}(q, p)$ or $\Phi_{\lambda/K}(p) - q$, and using Theorem 1.1 finish the proof.

Fast-Rate PAC-Bayesian kl-Bound for Meta-Learning

Apply the kl-bound in Theorem 2.2 to bound $\mathrm{kl}(er(\mathcal{Q}), \widetilde{er}(\mathcal{Q}))$ and $\mathrm{kl}(\widetilde{er}(\mathcal{Q}), \widehat{er}(\mathcal{Q}))$ respectively, and use the union bound, we have

Theorem 3.1

For any predefined hyper-prior $\mathcal{P}$, with probability at least $1-\delta$ over the draw of the training sample $\{S_i\}_{i=1}^n$, the following holds for any hyper-posterior $\mathcal{Q}$:

$$er(\mathcal{Q}) \leq \widehat{er}(\mathcal{Q}) + \sqrt{\frac{\mathcal{K}(\mathcal{Q},\mathcal{P}) + \ln\frac{2\sqrt{n}}{\delta}}{2n}} + \sqrt{\frac{2\Delta\widehat{er}(\mathcal{Q})}{mn}} + \frac{2\Delta}{mn},$$

where $\Delta = \mathcal{K}(\mathcal{Q},\mathcal{P}) + \mathbf{E}_{P\sim\mathcal{Q}}\sum_{i=1}^n \mathcal{K}(Q_i, P) + \ln\frac{2\sqrt{mn}}{\delta}$.

Fast-Rate PAC-Bayesian Catoni-Bound for Meta-Learning

Apply the Catoni-bound in Theorem 2.2 to bound $er(\mathcal{Q}) - \widetilde{er}(\mathcal{Q})$ and $\widetilde{er}(\mathcal{Q}) - \widehat{er}(\mathcal{Q})$ respectively, and use the union bound, we have

Theorem 3.2

For any predefined hyper-prior $\mathcal{P}$, any $\delta \in (0,1)$, any $C_1, C_2 > 1$, with probability at least $1-\delta$ over the draw of the training sample $\{S_i\}_{i=1}^n$, the following holds for any hyper-posterior $\mathcal{Q}$:

$$er(\mathcal{Q}) \leq \frac{C_1 C_2 \ln C_1 \ln C_2}{(C_1-1)(C_2-1)} \widehat{er}(\mathcal{Q}) + \frac{C_1\big(\mathcal{K}(\mathcal{Q},\mathcal{P}) + \ln(2/\delta)\big)}{n(C_1-1)}$$
$$+ \frac{C_1 C_2 \ln C_1 \big(\mathcal{K}(\mathcal{Q},\mathcal{P}) + \mathbf{E}_{P\sim\mathcal{Q}} \sum_{i=1}^n \mathcal{K}(Q_i, P) + \ln(2/\delta)\big)}{(C_1-1)(C_2-1)nm}.$$

Comparisons between Different PAC-Bayesian Bounds for Meta-Learning

Table 3: Different PAC-Bayesian meta-learning bounds on $er(\mathcal{Q})$. **Bound = Empirical Error + Environment-level Complexity + Task-level Complexity**. n is the number of training tasks. m is the sample size per task. n is the number of training tasks. m is the sample size per task. $\mathcal{P}, \mathcal{Q}$ are the hyper-prior and hyper-posterior respectively. In our Catoni-bound, the constant $C > 1$.

Classical Bounds	Empirical Error	Environment-Level Complexity	Task-Level Complexity
[8, p. ICML2014]	$\widehat{er}(\mathcal{Q})$	$O\left(\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})}{\sqrt{n}}\right)$	$O\left(\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})+\sum_{i=1}^{n}\mathbf{E}_{P\sim\mathcal{Q}}\mathcal{K}(Q_i,P)}{n\sqrt{m}}+\frac{1}{\sqrt{m}}\right)$
[1, p. ICML2018]	$\widehat{er}(\mathcal{Q})$	$O\left(\sqrt{\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})+\ln n}{n}}\right)$	$O\left(\frac{1}{n}\sum_{i=1}^{n}\sqrt{\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})+\mathbf{E}_{P\sim\mathcal{Q}}\mathcal{K}(Q_i,P)+\ln(2nm)}{m}}\right)$
[10, p. ICML2021]	$\widehat{er}(\mathcal{Q})$	$O\left(\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})}{\sqrt{n}}\right)$	$O\left(\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})+\sum_{i=1}^{n}\mathbf{E}_{P\sim\mathcal{Q}}\mathcal{K}(Q_i,P)}{n\sqrt{m}}+\frac{1}{\sqrt{n}}\right)$
kl-bound (ours)	$\widehat{er}(\mathcal{Q})$	$O\left(\sqrt{\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})+\ln\sqrt{n}}{n}}\right)$	$O\left(\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})+\mathbf{E}_{P\sim\mathcal{Q}}\sum_{i=1}^{n}\mathcal{K}(Q_i,P)+\ln\sqrt{nm}}{mn}\right)$
Catoni-bound (ours)	$C\widehat{er}(\mathcal{Q})$	$O\left(\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})}{n}\right)$	$O\left(\frac{\mathcal{K}(\mathcal{Q},\mathcal{P})+\mathbf{E}_{P\sim\mathcal{Q}}\sum_{i=1}^{n}\mathcal{K}(Q_i,P)}{mn}\right)$

Closed-Form of Hyper-Posterior when Minimizing Catoni-Bound (I)

We first give a corollary of Theorem 3.2 by choosing the Gibbs optimal posterior for each training task.

Corollary 3.3

$\forall i \in [n]$, any prior $P \in \mathcal{M}_1(\mathcal{H})$, any training data $\{S_i\}_{i=1}^n$, let Q_i^* be the Gibbs optimal posterior such that $\frac{\mathrm{d}Q_i^*}{\mathrm{d}P} = \exp\{-m\widehat{er}(h, S_i)\}/Z(S_i, P)$, where $Z(S_i, P) = \int_{\mathcal{H}} \mathrm{e}^{-m\widehat{er}(h,S_i)}\mathrm{d}P(h)$ is a normalization constant. Then $\forall \delta > 0, C_1 > 1$, with probability at least $1-\delta$ over the draw of training datasets $\{S_i\}_{i=1}^n$, the following holds for any hyper-posterior $\mathcal{Q}$:

$$er(\mathcal{Q}) \leq \frac{\mathrm{e}C_1 \ln C_1}{(C_1-1)(\mathrm{e}-1)} \mathbf{E}_{P\sim\mathcal{Q}} \frac{-1}{nm} \sum_{i=1}^{n} [\ln Z(S_i, P)]$$
$$+ \frac{C_1\left(\mathcal{K}(\mathcal{Q},\mathcal{P}) + \ln\frac{2}{\delta}\right)}{n(C_1-1)} + \frac{\mathrm{e}C_1 \ln C_1\left(\mathcal{K}(\mathcal{Q},\mathcal{P}) + \ln\frac{2}{\delta}\right)}{nm(C_1-1)(\mathrm{e}-1)}.$$

Closed-Form of Hyper-Posterior when Minimizing Catoni-Bound (II)

Next we can obtain the explicit form of Gibbs optimal hyper-posterior by minimizing the RHS of the inequality in Corollary 3.3.

Corollary 3.4

(Gibbs Optimal Hyper-posterior) For any hyper-prior $\mathcal{P}$ and any training datasets $\{S_i\}_{i=1}^n$, the hyper-posterior $\mathcal{Q}$ that minimizes the PAC-Bayesian meta-learning bound in Corollary 3.3 has the following explicit form:

$$\frac{\mathrm{d}\mathcal{Q}^*}{\mathrm{d}\mathcal{P}}(P) = \exp\{\frac{\beta}{nm}\sum_{i=1}^{n}\ln Z(S_i, P)\}/Z(\mathbf{S}, \mathcal{P}),$$

where $\beta = \frac{\mathrm{e}C_1 \ln C_1}{(C_1-1)(\mathrm{e}-1)\alpha}$, $\alpha = \frac{\mathrm{e}C_1 \ln C_1}{nm(C_1-1)(\mathrm{e}-1)} + \frac{C_1}{n(C_1-1)}$, $Z(\mathbf{S}, \mathcal{P}) = \int_{\mathcal{M}_1(\mathcal{H})} \exp\{\frac{\beta}{nm}\sum_{i=1}^{n}\ln Z(S_i, P)\}\mathrm{d}\mathcal{P}(P)$ *is a normalization constant.*

Fractional Cover of Dependent Data

We introduce two concepts to analyze the meta-learning setting with dependent samples.

Definition 3.5

(Dependence Graph) Let $\mathcal{S} = \{\xi_1, \ldots, \xi_K\}$ be a set of K random variables. The dependence graph $\Gamma(\mathcal{S}) = (V, E)$ of $\mathcal{S}$ is such that: (1) the set of vertices V of $\Gamma(\mathcal{S})$ is $V = [K]\}$. (2) $(i, j) \notin \mathrm{E}$ (i.e., there is no edge between i and j) $\Leftrightarrow$ ξ_i and ξ_j are independent.

Definition 3.6

(Fractional Covers [6]) Let $\Gamma = (V, E)$ be an undirected graph with $V = [K]$. (1) $C \subseteq V$ is independent if the vertices in C are independent (i.e., no two vertices in C are connected). (2) $\mathbf{C} = \{C_j\}_{j=1}^{J}$, with $C_j \subseteq V$, is a proper cover of V if each C_j is independent and $\bigcup_{j=1}^{J} C_j = V$. (3) $\mathbf{C} = \{(C_j, w_j)\}_{j=1}^{J}$, with $C_j \subseteq V$ and $w_j \in [0, 1]$, is a proper exact fractional cover of V if C_j is independent and $\forall i \in V$, $\sum_{j=1}^{J} w_j \mathbf{1}_{i \in C_j} = 1$; $\mathbf{w}(\mathbf{C}) = \sum_{j=1}^{J} w_j$ is defined as the chromatic weight of $\mathbf{C}$. (4) The fractional chromatic number $\chi^*(\Gamma)$ is the minimum chromatic weight over all proper exact fractional covers of the dependence graph $\Gamma = (V, E)$.

Fast-Rate kl-Bound for Non-Identically Non-Independently Distributed Data

Then we can obtain a chromatic PAC-Bayesian kl-bound with fast convergence rate $O(1/K)$ for dependent random variables.

Theorem 3.7

In the same setting of Theorem 2.2 with the only difference that $\mathcal{S} = \{\xi_k\}_{k=1}^K$ *is a sequence of dependent random variables. Let* $\chi^*(\mathcal{S})$ *denote the fractional chromatic number of the dependence graph of* $\mathcal{S}$*. Then with probability with at least* $1-\delta$ *over the draw of* $\mathcal{S}$*, the following holds for any measure* ρ *over* $\mathcal{F}$*:*

$$\mathrm{kl}(\rho(r), \rho(R)) \leq \frac{\chi^*(\mathcal{S})}{K}[\mathcal{K}(\rho, \pi) + \ln(\frac{2}{\delta}\sqrt{\frac{K}{\chi^*(\mathcal{S})}})].$$

Fast-Rate kl-Bound for Meta-Learning with Dependent Tasks

Use the above theorem to bound $\mathrm{kl}(er(\mathcal{Q}), \widetilde{er}(\mathcal{Q}))$ and $\mathrm{kl}(\widetilde{er}(\mathcal{Q}), \widehat{er}(\mathcal{Q}))$, we obtain PAC-Bayesian bound for meta-learning with dependent samples.

Theorem 3.8

For any given hyper-prior $\mathcal{P}$, with probability at least $1-\delta$ over the draw of the training sample $\{S_i\}_{i=1}^n$, the following holds for any hyper-posterior $\mathcal{Q}$:

$$er(\mathcal{Q}) \leq \widehat{er}(\mathcal{Q}) + \sqrt{\frac{\Delta_1}{2n}} + \sqrt{\frac{2\Delta_2 \widehat{er}(\mathcal{Q})}{mn}} + \frac{2\Delta_2}{mn},$$

where $\Delta_1 = \chi^*(\mathbf{D})[\mathcal{K}(\mathcal{Q},\mathcal{P}) + \ln(\frac{2}{\delta}\sqrt{\frac{n}{\chi^*(\mathbf{D})}})]$, $\Delta_2 = \chi^*(\mathbf{S})\big[\mathcal{K}(\mathcal{Q},\mathcal{P}) + \mathbf{E}_{P\sim\mathcal{Q}}\sum_{i=1}^n \mathcal{K}(Q_i,P) + \ln\frac{2\sqrt{mn}}{\delta\sqrt{\chi^*(\mathbf{S})}}\big]$, $\chi^*(\mathbf{D}), \chi^*(\mathbf{S})$ *denote the fractional chromatic numbers of the dependence graphs of* $\mathbf{D} = \{D_i\}_{i=1}^n$, $\mathbf{S} = \{S_i\}_{i=1}^n$.

Two PAC-Bayesian Bound-Minimization Meta Classification Algorithms

We set isotropic Gaussian for hyper-prior and hyper-posterior: $\mathcal{P} = \mathcal{N}(0, \kappa_{\mathcal{P}}^2 I_{d\times d})$, $\mathcal{Q}_\theta = \mathcal{N}(\theta, \kappa_{\mathcal{Q}}^2 I_{d\times d})$. Then the KL-divergence is

$$\mathcal{K}(\mathcal{Q}_\theta, \mathcal{P}) = \frac{||\theta||_2^2 + \kappa_{\mathcal{Q}}^2}{2\kappa_{\mathcal{P}}^2} + \ln\frac{\kappa_{\mathcal{P}}}{\kappa_{\mathcal{Q}}} - \frac{1}{2}.$$

We set factorized Gaussian distributions for prior/posterior:

$$P_\theta(w) = \prod_{k=1}^{d} \mathcal{N}(w_k; \mu_{P,k}, \sigma_{P,k}^2),$$

$$Q_{\phi_i}(w) = \prod_{k=1}^{d} \mathcal{N}(w_k; u_{i,k}, \sigma_{i,k}^2),$$

Then $\mathcal{K}(Q_{\phi_i}, P_\theta) = \frac{1}{2}\sum_{k=1}^{d} \ln\frac{\sigma_{P,k}^2}{\sigma_{i,k}^2} + \frac{(\sigma_{i,k}^2 + (\mu_{i,k} - \mu_{P,k})^2)}{\sigma_{P,k}^2}$. To approximate the expectation $P \sim \mathcal{Q}$, we use Monte-Carlo method. The pseudo code is listed in the right column.

Algorithm 1 Catoni-bound-minimizing meta-learning algorithm (meta-training phase)

Input: Datasets : $S_1, \ldots, S_n$.
Output: Parameters θ of hyper-posterior $\mathcal{Q}_\theta$.
Initialize:
$\theta = (\mu_P, \rho_P) \in \mathbb{R}^{2d}$, $\phi_i = (\mu_i, \rho_i) \in \mathbb{R}^{2d}, i = 1, \ldots, n$.
while *not converged* **do**
  for $i \in \{1, ..n\}$ **do**
    Sample a mini-batch S_i' from datasets S_i.
    Calculate $\mathbf{E}_{P_\theta\sim\mathcal{Q}_\theta}\widehat{er}(Q_i, S_i)$ with the mini-batch S_i' by averaging Monte-Carlo draws.
    Calculate $\mathcal{K}(\mathcal{Q}_\theta, \mathcal{P})$.
    Calculate $\mathbf{E}_{P_\theta\sim\mathcal{Q}_\theta}\mathcal{K}(Q_{\phi_i}, P_\theta)$ by averaging Monte-Carlo draws.
  end for
  Calculate the meta-training Catoni-bound with $\mathbf{E}_{P_\theta\sim\mathcal{Q}_\theta}\widehat{er}(Q_i, S_i)$, $\mathcal{K}(\mathcal{Q}_\theta, \mathcal{P})$ and $\mathbf{E}_{P_\theta\sim\mathcal{Q}_\theta}\mathcal{K}(Q_{\phi_i}, P_\theta)$, $i = 1, \ldots, n$.
  Calculate the gradient of Catoni-bound w.r.t $\{\theta, \phi_1, \ldots, \phi_n\}$ using backpropagation.
  Take an optimization step.
end while
Return θ

Performance on Classification Datasets

Table 4: Comparisons of different PAC-Bayesian meta-learning methods. The average test bounds and test errors are reported over 20 test tasks (the $\pm$ shows the 95% confidence interval) in three different pixel-shuffled environments.

Method	100 Pixels Swaps		200 Pixels Swaps		300 Pixels Swaps	
	Test Bound	Test Error (%)	Test Bound	Test Error (%)	Test Bound	Test Error (%)
VB	N/A	1.606 ± 0.001	N/A	1.962 ± 0.001	N/A	2.649 ± 0.130
MAML	N/A	1.876 ± 0.001	N/A	2.241 ± 0.002	N/A	2.788 ± 0.102
[11, JMLR2002]	0.133 ± 0.034	1.629 ± 0.000	0.285 ± 0.049	1.972 ± 0.001	0.408 ± 0.062	2.523 ± 0.001
[8, p. ICML2014]	0.190 ± 0.022	1.939 ± 0.001	0.240 ± 0.030	2.631 ± 0.002	0.334 ± 0.036	3.767 ± 0.003
[1, p. ICML2018]	0.126 ± 0.012	1.587 ± 0.001	0.197 ± 0.019	1.948 ± 0.001	0.270 ± 0.018	2.630 ± 0.001
[10, p. ICML2021]	0.174 ± 0.023	1.921 ± 0.001	0.224 ± 0.030	2.634 ± 0.001	0.318 ± 0.036	3.754 ± 0.003
kl-bound (ours)	0.119 ± 0.024	1.746 ± 0.001	0.189 ± 0.027	2.594 ± 0.001	0.359 ± 0.042	2.993 ± 0.002
Catoni-bound (ours)	$\mathbf{0.093 \pm 0.027}$	$\mathbf{1.545 \pm 0.001}$	$\mathbf{0.128 \pm 0.025}$	$\mathbf{1.889 \pm 0.001}$	$\mathbf{0.210 \pm 0.035}$	$\mathbf{2.433 \pm 0.001}$

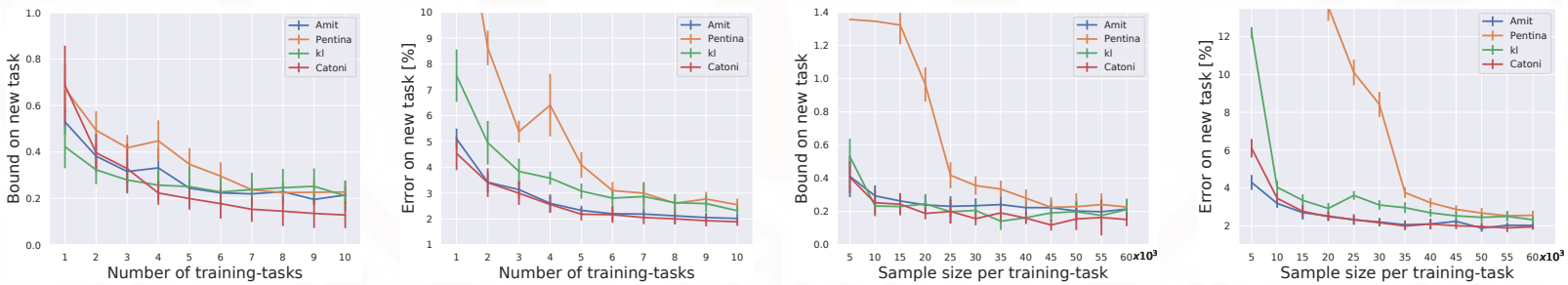

Figure 1: Comparisons between our bounds and others. Both test bounds and test errors are averaged over 20 classification tasks. (1)-(2): Results across a range of number n of training tasks. (3)-(4): Results across a range of sample size m per task.

Gibbs Optimal Hyper-Posterior (GOHP) Meta Regression Algorithms

We use the inference method SVGD [7] to approximate $\mathcal{Q}^*$ as a set of particles $\hat{\mathcal{Q}} = \{P_{\phi_1}, \ldots, P_{\phi_K}\}$, where P_ϕ represents a prior with parameter ϕ. Initially, we sample K particles ϕ_k from $\mathcal{P}$. Then based on the explicit form of $\mathcal{Q}^*$ in Corollary 3.4, we compute the gradient of $\mathcal{Q}^*$ w.r.t. ϕ_k:

$$\nabla_{\phi_k} \ln \mathcal{Q}^*(\phi_k) = \nabla_{\phi_k} \ln \mathcal{P}(\phi_k) + \frac{\beta}{nm} \sum_{i=1}^{n} \nabla_{\phi_k} \ln Z(S_i, P_{\phi_k})$$

where the marginal log-likelihood (MLL) $\ln Z(S_i, P_{\phi_k})$ is approximated by Monte Carlo estimates. Then we update the particles with the SVGD update rule:

$$\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} + \eta\, \mathbf{K}\, \nabla_{\boldsymbol{\phi}} \ln \tilde{\mathcal{Q}}^* + \nabla_{\boldsymbol{\phi}} \mathbf{K},$$

where $\boldsymbol{\phi} = [\phi_1, \ldots, \phi_K]^\top$ is the stacked particles matrix, $\nabla_{\boldsymbol{\phi}} \ln \tilde{\mathcal{Q}}^* = [\nabla_{\phi_1} \ln \mathcal{Q}^*(\phi_1), \ldots, \nabla_{\phi_K} \ln \mathcal{Q}^*(\phi_K)]^\top$ the stacked matrix of gradients, $\mathbf{K} = [k(\phi_k, \phi_{k'})]_{k,k'}$ the kernel matrix induced by the kernel function $k(\cdot, \cdot)$ and η the step size for updates. The Pseudo code for meta-training can be found in Algorithm 2.

Algorithm 2 GOHP with SVGD approximation of $\mathcal{Q}^*$ (meta-training phase)

Input: Hyper-prior $\mathcal{P}$, datasets $S_1, \ldots, S_n$.
Hyper-parameter: SVGD kernel function $k(\cdot, \cdot)$, step size η, scaler factor β.
Output: Set of priors $\{P_{\phi_1}, \ldots, P_{\phi_K}\}$.
Initialize: $\boldsymbol{\phi} := [\phi_1, \ldots, \phi_K]$, with $\phi_k \sim \mathcal{P}$.
while not converged **do**
  for $k = 1, \ldots, K$ **do**
    for $i = 1, \ldots, n$ **do**
      $\ln Z_{i,k} \leftarrow \texttt{MLL\_Estimator}(S_i, P_{\phi_k})$
    end for
    $\nabla_{\phi_k} \ln \tilde{\mathcal{Q}}^* \leftarrow \nabla_{\phi_k} \ln \mathcal{P} + \frac{\beta}{nm} \sum_{i=1}^{n} \nabla_{\phi_k} \ln Z_{i,k}$
  end for
  $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} + \eta\, \mathbf{K}\, \nabla_{\boldsymbol{\phi}} \ln \tilde{\mathcal{Q}}^* + \nabla_{\boldsymbol{\phi}} \mathbf{K}$ // SVGD update
end while
Return $\{P_{\phi_1}, \ldots, P_{\phi_K}\}$

Performance on Regression Datasets

Our GOHP algorithm can achieve comparable results with the latest PACOH.

Table 5: Comparison of meta-learning algorithms in terms of test RMSE in 5 regression environments. Reported are mean and standard deviation across 5 seeds. Our GOHP-NN achieves competitive averaged error over 5 environments.

Method	Cauchy	SwissFel	Physionet-GCS	Physionet-HCT	Berkeley-Sensor
Vanilla BNN [7]	0.327 ± 0.008	0.529 ± 0.022	2.664 ± 0.274	3.938 ± 0.869	0.109 ± 0.004
MLL-GP [4]	0.216 ± 0.003	0.974 ± 0.093	1.654 ± 0.094	2.634 ± 0.144	0.058 ± 0.002
MLAP [1]	0.219 ± 0.004	0.486 ± 0.026	2.009 ± 0.248	2.470 ± 0.039	0.050 ± 0.005
MAML [3]	0.219 ± 0.004	0.730 ± 0.057	1.895 ± 0.141	2.413 ± 0.113	0.045 ± 0.003
BMAML [12]	0.225 ± 0.004	0.577 ± 0.044	1.894 ± 0.062	2.500 ± 0.002	0.073 ± 0.014
PACOH-GP [10]	0.209 ± 0.008	0.376 ± 0.024	$\mathbf{1.498 \pm 0.081}$	$\mathbf{2.361 \pm 0.047}$	0.065 ± 0.005
PACOH-NN [10]	$\mathbf{0.195 \pm 0.001}$	0.372 ± 0.002	1.561 ± 0.061	2.405 ± 0.017	$\mathbf{0.043 \pm 0.001}$
GOHP-NN (ours)	0.198 ± 0.016	$\mathbf{0.333 \pm 0.013}$	1.521 ± 0.067	2.422 ± 0.013	$\mathbf{0.043 \pm 0.004}$

Conclusions and Future Works

Our contributions are four-fold:

(1) This work provides a unified demonstration framework of the PAC-Bayesian bounds for single-task learning and meta-learning, extending the tightest PAC-Bayesian kl-bound and Catoni-bound to the meta-learning setting, followed by two bound-minimizing meta-learning classification algorithms.

(2) We show how to obtain the closed-form formula of the Gibbs optimal hyper-posterior by minimizing our Catoni-bound, leading to an efficient meta-learning regression algorithm.

(3) We obtain a fast-rate chromatic PAC-Bayesian kl-bound for the more challenging meta-learning setting where training data show some dependencies.

(4) Experiments further validate the effectiveness of our proposed PAC-Bayesian bounds. In particular, our Catoni-bound obtains the tightest test bounds and the lowest test errors in classification problems, and achieves comparable results with existing methods in regression problems.

References

[1] Ron Amit and Ron Meir. "Meta-Learning by Adjusting Priors Based on Extended PAC-Bayes Theory". In: *International Conference on Machine Learning (ICML)*. 2018, pp. 205–214.

[2] Daniel Berend and Tamir Tassa. "Efficient Bounds on Bell Numbers and on Moments of Sums of Random Variables". In: *Probability and Mathematical Statistics* 30 (2010), pp. 185–205.

[3] Chelsea Finn, Pieter Abbeel, and Sergey Levine. "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks". In: *International Conference on Machine Learning (ICML)*. 2017, pp. 1126–1135.

List of References

[4] Vincent Fortuin and Gunnar Rätsch. "Deep Mean Functions for Meta-Learning in Gaussian Processes". In: *arXiv preprint arXiv:1901.08098* (2019).

[5] Pascal Germain et al. "PAC-Bayesian learning of linear classifiers". In: *International Conference on Machine Learning (ICML)*. 2009, pp. 353–360.

[6] Svante Janson. "Large deviations for sums of partly dependent random variables". In: *Random Structures & Algorithms* 24.3 (2004), pp. 234–248.

[7] Qiang Liu and Dilin Wang. "Stein Variational Gradient Descent: A General Purpose Bayesian Inference Algorithm". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2016, pp. 2370–2378.

List of References

[8] Anastasia Pentina and Christoph H. Lampert. "A PAC-Bayesian bound for Lifelong Learning". In: *International Conference on Machine Learning (ICML)*. 2014, pp. 991–999.

[9] Anastasia Pentina and Christoph H. Lampert. "Lifelong Learning with Non-i.i.d. Tasks". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2015, pp. 1540–1548.

[10] Jonas Rothfuss et al. "PACOH: Bayes-Optimal Meta-Learning with PAC-Guarantees". In: *International Conference on Machine Learning (ICML)*. 2021, pp. 9116–9126.

[11] Matthias W. Seeger. "PAC-Bayesian Generalisation Error Bounds for Gaussian Process Classification". In: *Journal of Machine Learning Research (JMLR)* 3 (2002), pp. 233–269.

List of References

[12] Jaesik Yoon et al. "Bayesian Model-Agnostic Meta-Learning". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2018, pp. 7343–7353.

Thanks!