Localized Meta-Learning: A PAC-Bayes Analysis for Meta-Learning Beyond Global Prior

Anonymous authors
Paper under double-blind review

ABSTRACT

Meta-learning methods learn the meta-knowledge among various training tasks and aim to promote the learning of new tasks under the task similarity assumption. However, such meta-knowledge is often represented as a fixed distribution, which is too restrictive to capture various specific task information. In this work, we present a localized meta-learning framework based on the PAC-Bayes theory. In particular, we propose an LCC-based prior predictor that allows the meta learner to adaptively generate local meta-knowledge for specific tasks. We further develop a practical algorithm with deep neural network based on the bound. Empirical results on real-world datasets demonstrate the efficacy of the proposed method.

1 INTRODUCTION

Recent years have seen a resurgence of interest in the field of meta-learning, or learning-to-learn (Thrun & Pratt, 2012), especially for empowering deep neural networks the capability of fast adapting to unseen tasks just as humans (Finn et al., 2017; Ravi & Larochelle, 2017). More concretely, the neural networks are trained from a sequence of datasets, associated with different learning tasks sampled from a meta-distribution (also called task environment (Baxter, 2000; Maurer, 2005)). The principal aim of meta learner is to extract transferable meta-knowledge from observed tasks and facilitate the learning of new tasks sampled from the same meta-distribution. The performance is measured by the generalization ability from a finite set of observed tasks, which is evaluated by learning related unseen tasks. For this reason, there has been considerable interest in theoretical bounds on the generalization in terms of the meta-learning algorithm (Denevi et al., 2018b/a). One typical line of work (Pentina & Lampert, 2014; Amit & Meir, 2018) use PAC-Bayes bound to analyze the generalization behavior of the meta learner and quantify the relation between the expected loss on new tasks and the average loss on the observed tasks. In this setup, we formulate meta-learning as hierarchical Bayes. Accordingly, meta-knowledge is instantiated as a global distribution over all possible priors, which we call hyperprior and is chosen before observing training tasks. Each prior is a distribution over a family of classifiers w.r.t. a particular task. To learn versatile meta-knowledge across tasks, the meta learner observes a sequence of training tasks and adjusts its hyperprior into a hyperposterior distribution over the set of priors. To solve a new task, the base learner produces a posterior distribution over a family of classifiers based on the associated sample set and the prior generated by the hyperposterior.

However, such meta-knowledge is shared across tasks. The global hyperposterior is rather generic, typically not well-tailored to various specific tasks. Consequently, it leads to sub-optimal performance for any individual prediction task. As a motivational example, suppose we have two different tasks: distinguishing motorcycle versus bicycle and distinguishing motorcycle versus car. Intuitively, each task uses distinct discriminative patterns and thus the desired meta-knowledge is required to extract these patterns simultaneously. This could be a challenging problem to represent it with a global hyperposterior since the most significant patterns in the first task could be irrelevant or even detrimental to the second task.

Hence, we are motivated to pursue a meta-learning framework to effectively define the hyperposterior. The inspiration comes from the PAC-Bayes literature on data distribution dependent priors (Catoni, 2007; Parrado-Hernández et al., 2012; Dziugaite & Roy, 2018). The choice of posterior in each task is constrained by the need to minimize the relative entropy between prior and posterior since this divergence forms part of the bound and is typically large in standard PAC-Bayes
approaches [Lever et al. 2013]. Thus, choosing an appropriate prior for each task which is close to the related posterior could yield improved generalization bounds.

Inspired by this, we propose a **Localized Meta-Learning** (LML) framework. Instead of formulating meta-knowledge as a global hyperposterior, we learn a conditional hyperposterior given task data distribution that allows a meta learner to adaptively generate an appropriate prior for a new task. However, the task data distribution is unknown, and our only perception for it is via the associated sample set. Nevertheless, if the conditional hyperposterior is relatively stable to perturbations of the sample set, then the generated prior could still reflect the underlying task data distribution, resulting in a generalization bound that still holds with smaller probability. Following this intuition, the dependence of a conditional hyperposterior on the task data distribution is parameterized by a prior predictor using Local Coordinate Coding (LCC) [Yu et al., 2009]. In particular, if the classifier in each task is specialized to a parametric model, including deep neural network, the proposed LCC-based prior predictor predicts the model parameters using the sample set by exploiting the local information on the latent manifold. LCC-based prior predictor is invariant under permutations of its inputs and could be further used for unseen tasks.

The main contributions of this work include: (i) We present a localized meta-learning framework which provides a means to tighten the original PAC-Bayes meta-learning bound [Pentina & Lampert, 2014; Amit & Meir, 2018] by minimizing the task-complexity term by choosing data-dependent prior; (ii) We propose an LCC-based prior predictor, an implementation of conditional hyperposterior, to generate local meta-knowledge for specific task; (iii) We derive a practical localized meta-learning algorithm for deep neural networks by minimizing the bound; (iv) Experimental results demonstrate improved performance over meta-learning method in this field.

2 Preliminaries

2.1 Local Coordinate Coding

We first review some definitions of Local Coordinate Coding (LCC) [Yu et al., 2009] based on which we develop the proposed LCC-based prior predictor.

**Definition 1. (Lipschitz Smoothness [Yu et al., 2009].)** A function $f(x)$ on $\mathbb{R}^d$ is a $(\alpha, \beta)$-Lipschitz smooth w.r.t. a norm $\| \|$ if $\| f(x) - f(x') \| \leq \alpha \| x - x' \|$ and $\| f(x') - f(x) - \nabla f(x)'(x' - x) \| \leq \beta \| x - x' \|^2$.

**Definition 2. (Coordinate Coding [Yu et al., 2009].)** A coordinate coding is a pair $(\gamma, C)$, where $C \subset \mathbb{R}^d$ is a set of anchor points, and $\gamma$ is a map of $x \in \mathbb{R}^d$ to $[\gamma_u(x)]_{u \in C} \in \mathbb{R}^{|C|}$ such that $\sum_{u \in C} \gamma_u(x) = 1$. It induces the following physical approximation of $x$ in $\mathbb{R}^d$: $\gamma(x) = \sum_{u \in C} \gamma_u(x) u$.

**Definition 3. (Latent Manifold [Yu et al., 2009].)** A subset $\mathcal{M} \subset \mathbb{R}^d$ is called a smooth manifold with an intrinsic dimension $d := d_\mathcal{M}$ if there exists a constant $c_\mathcal{M}$ such that given any $x \in \mathcal{M}$, there exists $d$ bases $u_1(x), \ldots, u_d(x) \in \mathbb{R}^d$ so that $\forall x' \in \mathcal{M}$:

$$\inf_{\gamma \in \mathbb{R}^{|C|}} \| x' - x - \sum_{j=1}^d \gamma_j u_j(x) \|_2 \leq c_\mathcal{M} \| x' - x \|_2^2,$$

where $\gamma = [\gamma_1, \ldots, \gamma_d]^T$ are the local codings w.r.t. the bases.

**Definition 4. (Covering Number [Yu et al., 2009].)** The covering number $\mathcal{N}(\epsilon, \mathcal{M})$ is the smallest cardinality of an $\epsilon$-cover $C \subset \mathcal{M}$. That is,

$$\sup_{x \in \mathcal{M}} \inf_{v \in C} \| x - v \| \leq \epsilon.$$

Definition 2 and 3 imply that any point in $\mathbb{R}^d$ can be expressed as a linear combination of a set of anchor points. Later, we will use them to develop the prior predictor.

2.2 PAC-Bayes Meta-Learning

In order to present the advances proposed in this paper, we next recall some definitions in PAC-Bayes theory for single-task learning and meta-learning [Catoni, 2007; Baxter, 2000; Pentina & Lampert]...
In the context of classification, we assume all tasks share the same input space $\mathcal{X}$, output space $\mathcal{Y}$, space of classifiers (hypotheses) $\mathcal{H} \subset \{h : \mathcal{X} \rightarrow \mathcal{Y}\}$ and loss function $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow [0, 1]$. The meta learner observes $n$ tasks in the form of sample sets $S_1, \ldots, S_n$. The number of samples in task $i$ is denoted by $m_i$. Each observed task $i$ consists of a set of i.i.d. samples $S_i = \{(x_j, y_j)\}_{j=1}^{m_i}$, which is drawn from a data distribution $S_i \sim D_i^{m_i}$. Following the meta-learning setup in (Baxter, 2000), we assume that each data distribution $D_i$ is generated i.i.d. from the same meta distribution $\tau$. Let $h(x)$ be the prediction of $x$, the goal of each task is to find a classifier $h$ that minimizes the expected loss $E_{x \sim D}(\ell(h(x), y))$. Since the underlying ‘true’ data distribution $D_i$ is unknown, the base learner receives a finite set of samples $S_i$ and produces an “optimal” classifier $h = A_b(S_i)$ with a deterministic learning algorithm $A_b(\cdot)$ that will be used to predict the labels of unseen inputs.

PAC-Bayes theory studies the properties of randomized classifier, called Gibbs classifier. Let $Q$ be a posterior distribution over $\mathcal{H}$, to make a prediction, the Gibbs classifier samples a classifier $h \in \mathcal{H}$ according to $Q$ and then predicts a label with the chosen $h$. The expected error under data distribution $D$ and empirical error on the sample set $S$ are then given by averaging over distribution $Q$, namely

$$er(Q) = E_{h \sim Q} E_{(x, y) \sim D} \ell(h(x), y)$$

and

$$\hat{er}(Q) = E_{h \sim Q} \frac{1}{m} \sum_{j=1}^{m} \ell(h(x_j), y_j),$$

respectively. Then, we can get the following PAC-Bayes generalization bound of Catoni (2007) in a simplified form suggested by (Germain et al., 2009).

**Theorem 1. (Catoni’s bound)** Let $P$ be some prior distribution over $\mathcal{H}$. Then for any $\delta \in (0, 1]$, and any real number $c > 0$, the following inequality holds uniformly for all posteriors distribution $Q$ with probability at least $1 - \delta$,

$$er(Q) \leq \frac{c}{1 - e^{-c}} \left[ \hat{er}(Q) + \frac{KL(Q\|P) + \log \frac{1}{\delta}}{mc} \right].$$

(1)

The PAC-Bayes bound holds uniformly for all $Q$, it also holds for the data dependent $Q$. By choosing the posterior $Q$ that minimizes the PAC-Bayes bound, we obtain an learning algorithm with generalization guarantees. Note that the value $c$ allows to control the trade-off between the empirical error and the complexity term.

The goal of the meta learner is to extract meta-knowledge contained in the observed tasks that will be used as prior knowledge for learning new tasks. The prior knowledge $P$ is in the form of a distribution over classifiers $\mathcal{H}$. In each task, the base learner produces a posterior $Q = A_b(S, P)$ over $\mathcal{H}$ based on a sample set $S$ and a prior $P$. All tasks are learned through the same learning procedure. The meta learner treats the prior $P$ itself as a random variable and assumes the meta-knowledge is in the form of a distribution over all possible priors. Let hyperprior $P$ be an initial distribution over priors, meta learner uses the observed tasks to adjust its original hyperprior $P$ into hyperposterior $Q$ from the learning process. The quality of the hyperposterior $Q$ is measured by the expected task error of learning new tasks using priors generated from it, which is formulated as:

$$er(Q) = E_{P \sim Q} E_{(D, m) \sim \tau, S \sim D} er(Q = A_b(S, P)).$$

(2)

Accordingly, the empirical counterpart of the above quantity is given by:

$$\hat{er}(Q) = E_{P \sim Q} \frac{1}{n} \sum_{i=1}^{n} \hat{er}(Q = A_b(S_i, P)).$$

(3)

## 3 PAC-BAYES META-LEARNING BOUND WITH GAUSSIAN RANDOMIZATION

Based on the above definition, (Pentina & Lampert, 2014) and (Amit & Meir, 2018) present meta-learning PAC-Bayes generalization bounds w.r.t. hyperposterior $Q$ by using McAllester’s single-task bound (McAllester, 1999). Here we present a new meta-learning generalization bound with Gaussian randomization by using Catoni’s bound in Eq. (1). In particular, the classifier $h$ is parameterized as $h_w$, with $w \in \mathbb{R}^{d_w}$. The prior and posterior is a distribution over the set of all possible parameters $w$. We choose both the prior $P$ and posterior $Q$ to be spherical Gaussians, i.e. $P = \mathcal{N}(w^P, \sigma_w^2 I_{d_w})$ and $Q = \mathcal{N}(w^Q, \sigma_w^2 I_{d_w})$. The mean $w^P$ is a random variable distributed first according to the hyperprior $P$, which we formulate as $\mathcal{N}(0, \sigma_w^2 I_{d_w})$, and later according to hyperposterior $Q$, which we model as $\mathcal{N}(w^Q, \sigma_w^2 I_{d_w})$. When encountering a new task $i$, we first sample the mean of prior
\[ w^p \] from the hyperposterior \( N(w^Q, \sigma^2 w I_{d_w}) \), and then use it as a basis to learn the mean of posterior \( w^Q \), as shown in Figure 1(left). Then, we could derive the following PAC-Bayes meta-learning bound.

**Theorem 2.** Consider the Meta-Learning (ML) framework, given the hyperprior \( \mathcal{P} = N(0, \sigma^2_I D_w) \), then for any hyperposterior \( Q \), any \( c_1, c_2 > 0 \) and any \( \delta \in (0, 1] \) with probability \( \geq 1 - \delta \) we have,

\[
er(Q) \leq c'_1 c'_2 \sqrt{\text{er}(Q)} + \sum_{i=1}^{n} \frac{c_1 c'_2}{2c_2 n m_i \sigma^2 w} \| \mathbb{E} w^Q_i - w^Q \|^2 + \sum_{i=1}^{n} \frac{c_1 c'_2}{2c_2 n m_i \sigma^2 w} \| \mathbb{E} w^Q_i - w^Q \|^2 
+ \sum_{i=1}^{n} \frac{c_1 c'_2}{2c_2 n m_i \sigma^2 w} \left( \frac{1}{2} + \log \frac{2n}{\delta} \right) \frac{c_1}{c_1 n \sigma^2 w} \log \frac{2}{\delta}, \tag{4}
\]

where \( c'_1 = \frac{c_1}{1 - \epsilon - c_1} \) and \( c'_2 = \frac{c_2}{1 - \epsilon - c_2} \).

**Proof.** See Appendix [B.3] for the proof.

Notice that the expected task generalization error is bounded by the empirical multi-task error plus two complexity terms. The first term demonstrates the environment-complexity which converges to zero if an infinite number of tasks are observed from the task environment \( (n \to \infty) \), while the second is the task-complexity of the observed tasks which converges to zero when the sufficient samples in each task is observed \( (m_i \to \infty) \). Besides, the derived bound converges at the rate of \( O\left(\frac{1}{m}\right) \) instead of \( O(\frac{1}{\sqrt{m}}) \) in \([\text{Pentina & Lampert, 2014}] \) \([\text{Amit & Meir, 2018}] \), due to the use of Catoni’s bound.

### 4 PAC-BAYES LOCALIZED META-LEARNING

#### 4.1 Overall Framework

Our motivation stems from a core challenge in PAC-Bayes meta-learning bound in \([11] \), wherein the complexity term \( \sum_{i=1}^{n} \frac{c_1 c'_2}{2c_2 n m_i \sigma^2 w} \| \mathbb{E} w^Q_i - w^Q \|^2 \) is typically vital to the bound and so finding the tightest possible bound generally depends on minimizing this term. It is obvious that the optimal \( w^Q \) is \( \sum_{i=1}^{n} \frac{c_1 c'_2}{2c_2 n m_i \sigma^2 w} \mathbb{E} w^Q_i \). However, if the learned posteriors for each task are mutually exclusive, i.e., one learned posterior has a negative effect on another task, this term could be inevitably large.

\( w^Q \) is the mean of hyperposterior \( Q \) and this term naturally indicates the divergence between the mean of prior \( w^p \) sampled from the hyperposterior \( Q \) and the mean of posterior \( w^Q \) in each task.

Therefore, we propose to adaptively choose the mean of prior \( w^p_i \) according to task \( i \). It is obvious that the complexity term vanishes if we set \( w^p_i = w^Q_i \), but the prior \( P_i \) in each task has to be chosen independently of the sample set \( S_i \). Fortunately, the PAC-Bayes theorem allows us to choose prior upon the data distribution \( D_i \). Therefore, we propose a prior predictor \( \Phi : D^m \to w^P \) which receives task data distribution \( D^m \) and outputs the mean of prior \( w^P \). In this way, the generated priors could focus locally on those regions of model parameters that are of particular interest in solving specific tasks.

Particularly, the prior predictor is parameterized as \( \Phi_v \) with \( v \in \mathbb{R}^{d_v} \). We abuse notation \( \mathcal{P} \) and \( Q \) and assume \( v \) as a random variable distributed first according to the hyperprior \( \mathcal{P} \), which we reformulate as \( N(0, \sigma^2 v I_{d_v}) \), and later according to hyperposterior \( Q \), which we reformulate as \( N(v^Q, \sigma^2 v I_{d_v}) \). Given a new task \( i \), we first sample \( v \) from hyperposterior \( N(v^Q, \sigma^2 v I_{d_v}) \) and estimate the mean of prior \( w^p_i \) by leveraging prior predictor \( w^p_i = \Phi_v(D^m) \). Then, the base learner utilizes the sample set \( S_i \) and the prior \( P_i = N(w^p_i, \sigma^2 w I_{d_w}) \) to produce a mean posterior \( w^Q_i = A_b(S_i, P_i) \), as illustrated in Figure 1(right).

#### 4.2 LCC-BASED PRIOR PREDICTOR

To make \( w^p \) close to \( w^Q \) in each task, the prior predictor is required to (i) uncover the tight relationship between the sample set and model parameter. Intuitively, features and parameters yield similar local and global structures in their respective spaces in the classification problem. Features in the
same category tend to be spatially clustered together while maintaining the separation between different categories. Take linear classifiers as an example, let $w_k$ be the parameters w.r.t. category $k$, the separability between classes is implemented as $x \cdot w_k$, which also explicitly encourages intra-class compactness. A reasonable choice of $w_k$ is to maximize the inner product distance with the input features in the same category and minimize the distance with the input features of the non-belonging categories. Besides, the prior predictor should be (ii) category-agnostic since it will be used continuously as new tasks and hence new categories become available. Lastly, it should be (iii) invariant under permutations of its inputs.

To satisfy the above conditions, we follow the idea of nearest class mean classifier [Mensink et al., 2013], which represents class parameter by averaging its feature embeddings. This idea has been explored in transductive few-shot learning problem (Bertinetto et al. [2016] Yang et al. [2018], Snell et al. [2017]) learns a metric space across tasks such that when represented in this embedding, prototype (centroid) of each class can be used for label prediction in the new task. [Qiao et al. [2018]] directly predicts the classifier weights using the activations by exploiting the close relationship between the parameters and the activations in a neural network associated with the same category. In summary, the classification problem of each task is transformed as a generic metric learning problem which is shared across tasks. Once this mapping has been learned on observed tasks, due to the structure-preserving property, it could be easily generalized to new tasks. Formally, let each task be a $K$-class classification problem. Then the parameter of the classifier in task $i$ is represented as $w_i = [w_i[1], \ldots, w_i[k], \ldots, w_i[K]]$. The prior predictor for class $k$ could be defined as:

$$w^P_i[k] = \Phi_v(D^m_{ik}) = \mathbb{E}_{S_{ik} \sim D^m_{ik}} \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \phi_v(x_j), \quad (5)$$

where $\phi_v(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^{d_w}$ is the feature embedding function, $m_{ik}$ is the number of samples belonging to category $k$, $S_{ik}$ and $D_{ik}$ are the sample set and data distribution for category $k$ in task $i$. We call this function the expected prior predictor. Since data distribution $D_{ik}$ is considered unknown and our only insight as to $D_{ik}$ is through the sample set $S_{ik}$, we approximate the expected prior predictor by its empirical counterpart, based on $m_{ik}$ observed samples in the category $k$:

$$\bar{w}^P_i[k] = \bar{\Phi}_v(S_{ik}) = \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \phi_v(x_j), \quad (6)$$

which we call the empirical prior predictor. Although we can implement the embedding function $\phi_v(\cdot)$ with a multilayer perceptron (MLP), both input $x$ and model parameter $w$ are high-dimensional, making the empirical prior predictor $\bar{\Phi}_v(\cdot)$ difficult to learn. According to Definition (3), any points on the latent manifold can be approximated by a linear combination of a set of anchor points. Inspired by this, if the anchor points are sufficiently localized, the empirical prior predictor $\bar{\Phi}_v(S)$ can also be approximated by a linear function w.r.t. a set of codings. Accordingly, we propose an LCC-based prior predictor, which is defined as:

$$\bar{w}^P_i[k] = \bar{\Phi}_v(S_{ik}) = \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \sum_{u \in C} \gamma_u(x_j) \phi_v(u), \quad (7)$$
where \( \Phi_v(u) \in \mathbb{R}^{d_w} \) is the feature embedding of base \( u \in \mathbb{R}^d \). As such, the parameters of LCC-based prior predictor w.r.t. category \( k \) can be represented as \( v_k = [\Phi_v(u_1), \Phi_v(u_2), \ldots, \Phi_v(u_{C'})] \). Lemma 1 illustrates the approximation error between expected prior predictor and LCC-based prior predictor.

In order to derive a PAC-Bayes generalization bound for localized meta-learning, we first bound the approximation error between expected prior predictor and LCC-based prior predictor.

**Lemma 1.** (Empirical Prior Predictor Approximation) Given the definition of \( \hat{w}_i^p[k] \) and \( \hat{w}_i^p[k] \) in Eq. (6) and Eq. (7), let \( (\gamma, C) \) be an arbitrary coordinate coding on \( \mathbb{R}^d \) and \( \phi \) be an \((\alpha, \beta)\)-Lipschitz smooth function. We have for all \( x \in \mathbb{R}^d \)

\[
||\hat{w}_i^p[k] - \hat{w}_i^p[k]|| \leq \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \left( \alpha ||x_j - \bar{x}_j|| + \beta \sum_{u \in C} \|\bar{x}_j - u\|^2 \right) = O_{\alpha, \beta}(\gamma, C),
\]

where \( \bar{x}_j = \sum_{u \in C} \gamma_u(x_j) u \). Then given any \( \epsilon > 0 \), there exists a coding \((\gamma, C)\) such that

\[
|C| \leq (1 + d_M)N(\epsilon, \mathcal{M}),
\]

\[
O_{\alpha, \beta}(\gamma, C) \leq [\alpha c_M + (1 + 5\sqrt{d_M})\beta]\epsilon^2.
\]

**Proof.** See appendix B.1 for the proof.

The first inequality of Lemma 1 demonstrates that a good LCC-based prior predictor should make \( x \) close to its physical approximation \( \bar{x} \) and should be localized. The second and third inequality show that if a set of anchor points \( C \) has cardinality \( O(d_M N(\epsilon, \mathcal{M})) \), empirical prior predictor can be linearly approximated using LCC up to accuracy \( O(\sqrt{d_M}\epsilon^2) \). The complexity of the LCC coding scheme depends only on the number of anchor points \( |C| \) instead of the input dimension. In fact, a small \( |C| \) is usually sufficient to achieve good approximation.

**Optimization of LCC.** We minimize the first inequality in (8) to obtain a set of anchor points. As with [Yu et al. 2009], we simplify the localization error term by assuming \( \bar{x} = x \), and then we optimize the following objective function:

\[
\arg\min_{\gamma, C} \sum_{i=1}^n \sum_{x_j \in S_i} \alpha ||x_j - \bar{x}_j||^2 + \beta \sum_{u \in C} ||x_j - u||^2 \quad s.t. \quad \sum_{u \in C} \gamma_u(x) = 1, \forall x,
\]

where \( \bar{x} = \sum_{u \in C} \gamma_u(x) u \). In practice, we update \( C \) and \( \gamma \) by alternately optimizing a LASSO problem and a least-square regression problem, respectively.

### 4.3 PAC-Bayes Localized Meta-Learning Bound with Gaussian Randomization

In order to derive a PAC-Bayes generalization bound for localized meta-learning, we first bound the approximation error between expected prior predictor and LCC-based prior predictor.

**Lemma 2.** Given the definition of \( w^p \) and \( \hat{w}^p \) in Eq. (5) and (7), let \( X \) be a compact set with radius \( R \), i.e., \( \forall x, x' \in X, ||x - x'|| \leq R \). For any \( \delta \in (0, 1] \) with probability \( \geq 1 - \delta \), we have

\[
||w^p - \hat{w}^p||^2 \leq \sum_{k=1}^K \left( \frac{\alpha R}{\sqrt{m_{ik}}} \left( 1 + \frac{1}{2} \log \left( \frac{1}{\delta} \right) \right) + O_{\alpha, \beta}(\gamma, C) \right)^2.
\]
Lemma 2 shows that the approximation error between expected prior predictor and LCC-based prior predictor depends on (i) the concentration of prior predictor and (ii) the quality of LCC coding scheme. The first term implies the number of samples for each category should be larger for better approximation. This is consistent with the results of estimating the center of mass (Cristianini & Shawe-Taylor, 2004). Based on Lemma 2 we have the following PAC-Bayes LML bound.

**Theorem 3.** Consider the Localized Meta-Learning (LML) framework, give the hyperprior \( P = N(0, \sigma^2_P I_{d_a}) \), then for any hyperposterior \( Q \), any \( c_1, c_2 > 0 \) and any \( \delta \in (0, 1] \) with probability \( \geq 1 - \delta \) we have,

\[
er(Q) \leq c_1' c_2' e^2r(Q) + \left( \sum_{i=1}^{n} \frac{c'_1 c'_2}{2 c_{2nm_i} \sigma^2_V} + \frac{c'_1}{2c_1 n \sigma^2_V} \right) \|\nu\|^2 + \sum_{i=1}^{n} \frac{c'_1 c'_2}{c_{2nm_i} \sigma^2_W} \|\mathbb{E}_v Q - \Phi_v \circ (S_i)\|^2 + \frac{\alpha R}{\sqrt{m_{ik}}} \left( 1 + \sqrt{\frac{1}{2} \log\left( \frac{4n}{\delta} \right)} \right) + O_{\alpha, \beta}(\gamma, C) + d_w K \left( \frac{\sigma_v}{\sigma_w} \right)^2 \sum_{i=1}^{n} \frac{c'_1 c'_2}{c_{2nm_i} \sigma^2_W} \log \frac{4n}{\delta} + \frac{c'_1}{2c_1 n \sigma^2_V} \log \frac{2}{\delta},
\]

(12)

where \( c'_1 = \frac{c_1}{1-e^{-2}} \) and \( c'_2 = \frac{c_2}{1-e^{-2}} \). To get a better understanding, we further simplify the notation and obtain that

\[
er(Q) \leq c_1' c_2' e^2r(Q) + \left( \sum_{i=1}^{n} \frac{c'_1 c'_2}{2 c_{2nm_i} \sigma^2_V} + \frac{c'_1}{2c_1 n \sigma^2_V} \right) \|\nu\|^2 + \sum_{i=1}^{n} \frac{c'_1 c'_2}{c_{2nm_i} \sigma^2_W} \|\mathbb{E}_v Q - \Phi_v \circ (S_i)\|^2 + \text{const}(\alpha, \beta, R, \delta, n, m_i).
\]

(13)

**Proof.** See appendix B.3 for the proof.

Similarly with the PAC-Bayes meta-learning bound in Theorem 2 and the bounds in (Pentina & Lampert, 2014; Amit & Meir, 2018), the expected task error \( er(Q) \) is bounded by the empirical task error \( e^2r(Q) \) plus the task-complexity and environment-complexity terms. The main innovation here is to exploit the potential to choose the mean of prior \( w^P \) based on task data \( S \). Intuitively, if the selection of the LCC-based prior predictor is appropriate, it will narrow the divergence between the mean of prior \( w^P \) sampled from the hyperposterior \( Q \) and the mean of posterior \( w^Q \) in each task. Therefore, the bound can be tighter than the ones in the meta-learning framework. Our empirical study in Section 5 illustrates that the algorithms derived from this bound can achieve better performance than the methods derived from standard PAC-Bayes meta-learning bounds.

When one is choosing the LCC-based prior predictor \( \Phi_v(\cdot) \), the number of anchor points \( |C| \), there is a balance between accuracy and simplicity. As we increase \( |C| \), it will essentially increase the expressive power of \( \Phi_v(\cdot) \) and reduce the complexity term \( \|\mathbb{E}_v Q - \Phi_v \circ (S_i)\|^2 \). However, at the same time, it will increase the complexity term \( \|\nu\|^2 \) and make the bound loose. If we set \( |C| \) to 1, it is degraded to the regular meta-learning framework.

### 4.4 Localized Meta-Learning Algorithm

Since the bound in (27) holds uniformly w.r.t. \( Q \), the guarantees of Theorem 3 also hold for the resulting learned hyperposterior \( Q = N(\nu^Q, \sigma^2_P I_{d_a}) \), so the mean of prior \( w^P \) sampled from the learned hyperposterior work well for future tasks. The PAC-Bayes localized meta-learning bound in (27) can be compactly written as

\[
\sum_{i=1}^{n} \mathbb{E}\err_i(Q_i = A_i(S_i, P)) + \alpha_1 \|\nu\|^2 + \sum_{i=1}^{n} \frac{\alpha_2}{m_i} \|\mathbb{E}_v Q - \Phi_v \circ (S_i)\|^2,
\]

(14)

where \( \alpha_1, \alpha_2 > 0 \) are hyperparameters. For task \( i \), the learning algorithm \( A_i(\cdot) \) can be formulated as \( w^*_i = \arg\min_{w^Q \in \mathbb{V}} \mathbb{E}\err_i(Q_i = A_i(S_i, P)) \). Following Amit & Meir (2018), we jointly opti-
mize the parameters of LCC-based prior predictor \( v \) and the parameters of classifiers in each task \( w_1, w_2, \ldots, w_n \), which is formulated as

\[
    \arg \min_{v, w_1, \ldots, w_n} \sum_{i=1}^{n} \mathbb{E}_{r_i} [v_i] + \alpha_1 \|v\|^2 + \sum_{i=1}^{n} \frac{\alpha_2}{m_i} \|\mathbb{E}_{w_i} - \Phi_{v, \phi}(S_i)\|^2. \tag{15}
\]

We can optimize \( v \) and \( w \) via mini-batch SGD. The details of algorithms for meta-training are given in Algorithms[] The expectation over Gaussian distribution and its gradient can be efficiently

**Algorithm 1 Localized Meta-Learning (LML) algorithm**

**Input:** Data sets of observed tasks: \( S_1, \ldots, S_n \).

**Output:** Learned prior predictor \( \Phi \) parameterized by \( v \).

Initialize \( v \in \mathbb{R}^{d_v} \) and \( w_i \in \mathbb{R}^{d_w} \) for \( i = 1, \ldots, n \).

Construct LCC scheme \((\gamma, C)\) from the whole training data by optimizing Eq. \((10)\).

while not converged do

for each task \( i \in \{1, \ldots, n\} \) do

Sample a random mini-batch from the data \( S'_i \subset S_i \).

Approximate \( \mathbb{E} r_i (w_i) \) using \( S'_i \).

end for

Compute the objective in \((15)\), i.e. \( J \leftarrow \sum_{i=1}^{n} \mathbb{E} r_i (w_i) + \alpha_1 \|v\|^2 + \sum_{i=1}^{n} \frac{\alpha_2}{m_i} \|\mathbb{E} w_i - \Phi_{v, \phi}(S_i)\|^2 \).

Evaluate the gradient of \( J \) w.r.t. \( \{v, w_1, \ldots, w_n\} \) using backpropagation.

Take an optimization step.

end while

estimated by using the re-parameterization trick \cite{KingmaW14, RezendePS14}. For example, to sample \( w \) from the posterior \( Q = \mathcal{N}(w^N, \sigma_w^2 I_{d_w}) \), we first draw \( \xi \sim \mathcal{N}(0, I_{d_w}) \) and then apply the deterministic function \( w^N + \xi \odot \sigma \), where \( \odot \) is an element-wise multiplication.

5 EXPERIMENTS

![Figure 3: The average test accuracy of learning a new task for different number of training tasks \(|C| = 64\).](image_a)

(a) With pre-trained feature extractor

![Figure 3: The average test accuracy of learning a new task for different number of training tasks \(|C| = 64\).](image_b)

(b) Without pre-trained feature extractor

Figure 4: (a) The impact of the number of bases \(|C|\) in LCC. (b) The divergence value (normalized) between the mean generated prior \( w^P \) and the mean of learned posterior \( w^Q \).

5.1 DATASETS AND SETUP

We use CIFAR-100 and Caltech-256 in our experiments. CIFAR-100 \cite{Krizhevsky09} contains 60,000 images from 100 fine-grained categories and 20 coarse-level categories. As in \cite{ZhouSLC17}. 

8
since the bound w.r.t. ML-A in Theorem 2 converges at the rate of $O(\sqrt{m})$ if large numbers of tasks are observed. ML-A consistently outperforms ML-PL and ML-AM methods. This is consistent with the generalization bounds that the complexity term converges to the performance continually increases as we increase the number of training tasks for all the training tasks in different settings (with or without a pre-trained feature extractor). It is obvious in Figure 3, we demonstrate the average test error of learning a new task based on the number of approaches adopt the same network architecture and pre-trained feature extractor. MAML (Finn et al., 2017) and MatchingNet (Vinyals et al., 2016). To ensure a fair comparison, all methods, we use their main theorems about the PAC-Bayes generalization bound to derive the objective for the algorithm. We also compare with two typical meta-learning few-shot learning methods: MAML (Finn et al., 2017) and MatchingNet (Vinyals et al., 2016). To ensure a fair comparison, all approaches adopt the same network architecture and pre-trained feature extractor.

5.2 RESULTS

In Figure 3, we demonstrate the average test error of learning a new task based on the number of training tasks in different settings (with or without a pre-trained feature extractor). It is obvious that the performance continually increases as we increase the number of training tasks for all the methods. This is consistent with the generalization bounds that the complexity term converges to zero if large numbers of tasks are observed. ML-A consistently outperforms ML-PL and ML-AM since the bound w.r.t. ML-A in Theorem 2 converges at the rate of $O(\frac{1}{\sqrt{m}})$ while the bounds w.r.t. ML-PL and ML-AM converge at the rate of $O(\frac{1}{\sqrt{m}})$. This demonstrates the importance of using tight generalization bound. Our proposed LML significantly outperforms the baselines, which validates the effectiveness of the proposed LCC-based prior predictor. It is a more suitable representation for meta-knowledge than the traditional global hyperposterior in ML-A, ML-AM, and ML-PL.

Moreover, we can find that all PAC-Bayes baselines outperform MAML and MatchingNet. Note that MAML and MatchingNet adopt the episodic training paradigm to solve the few-shot learning problem. The meta-training process requires millions of tasks and each task contains limited samples, which is not the case in our experiment. Scarce tasks in meta-training leads to severely meta-overfitting. In our method, the learned prior serves both as an initialization of base model and as a regularizer which restricts the solution space while allowing variation based on specific task data. It yields a model with smaller error than its unbiased counterpart when applied to a similar task.

Finally, we observe that if the pre-trained feature extractor is provided, all of these methods do better than meta-training with random initialization. This is because pre-trained feature extractor can be regarded as a data-dependent hyperprior. It is closer to the hyperposterior than the randomly initialized hyperprior. Therefore, it reduces the environment complexity term and improves the generalization performance.

In Figure 4(b), we show the divergence between the mean of generated prior $\mathbf{w}^P$ from meta model and the mean of learned posterior $\mathbf{w}^Q$ for LML and ML-A. This further validates the effectiveness of the LCC-based prior predictor which could narrow the divergence term and thus tight the bound. In Figure 4(a), we vary the number of bases $|C|$ in LCC scheme from 4 to 256, the optimal value is around 64 in both datasets. This indicates that LML is sensitive to the number of bases $|C|$, which further affects the quality of LCC-based prior predictor and the performance of LML.

6 RELATED WORK

Meta-Learning. Meta-learning literature commonly considers the empirical task error by directly optimizing a loss of meta learner across tasks in the training data. Recently, this has been success-
fully applied in a variety of models for few-shot learning (Ravi & Larochelle 2017; Snell et al. 2017; Finn et al. 2017; Vinyals et al. 2016). Although Vuorio et al. (2018); Rusu et al. (2019); Zintgraf et al. (2019); Wang et al. (2019) consider task adaptation when using meta-knowledge for specific tasks, all of them are not based on generalization error bounds, which is the focus of our work. Meta-learning in the online setting has regained attention recently (Denevi et al. 2018a,b, 2019; Balcan et al. 2019), in which online-to-batch conversion results could imply generalization bounds. Galanti et al. (2016) analyzes transfer learning in neural networks with PAC-Bayes tools. Most related to our work are (Pentina & Lampert 2014; Amit & Meir 2018) which provide a PAC-Bayes generalization bound for meta-learning framework. In contrast, neither work considers localized meta-knowledge for specific tasks.

Localized PAC-Bayes Learning. There has been a prosperous line of research for learning priors to improve the PAC-Bayes bounds (Catoni 2007; Guedj 2019). (Parrado-Hernández et al. 2012) showed that priors can be learned by splitting the available training data into two parts, one for learning the prior, one for learning the posterior. (Lever et al. 2013) derived an expression for the overall best prior, i.e. the distribution resulting in the smallest possible bound value and bounded the KL divergence by a term independent of data distribution. Recently, (Rivasplata et al. 2018) bounded the KL divergence by investigating the stability of the hypothesis. (Dziugaite & Roy 2018) optimized the prior term in a differentially private way. In summary, these methods construct some quantities that reflect the underlying data distribution, rather than the sample set, and then choose the prior $\mathcal{P}$ based on these quantities. These works, however, are only applicable for single-task problem and could not transfer knowledge across tasks in meta-learning setting.

7 CONCLUSION

This work contributes a novel localized meta-learning framework from a theoretical perspective. We propose a generalization bound based on PAC-Bayes theory with Gaussian randomization. Instead of formulating meta-knowledge as a global distribution, we propose an LCC-based prior predictor to output local meta-knowledge by using task information. We further develop a practical algorithm with deep neural network based on the bound. An interesting topic for future work would be to explore other principles to construct the prior predictor and apply the localized meta-learning framework to a more realistic scenario that tasks are sampled non-i.i.d. from an environment. Another challenging problem is to extend our techniques to derive localized meta-learning algorithms for regression and reinforcement learning problem.

REFERENCES


in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018, pp. 457–466, 2018a.


This supplementary document contains the technical proofs of theoretical results and details of experiments. It is structured as follows: Appendix A presents notations for prior predictor. Appendix B gives the proofs of the main results. Appendix B.1 and B.2 show the approximation error between LCC-based prior predictor and empirical prior predictor, expected prior predictor, respectively. They are used in the proof of Theorem 3. Next, in Appendix B.3 and B.4 we show the PAC-Bayes generalization bound of localized meta-learning in Theorem 3 and also provides the PAC-Bayes generalization bound of regular meta-learning in Theorem 2. Finally, details of experiments are presented in Appendix C.

A NOTATIONS

Let \( \phi_v : \mathbb{R}^d \rightarrow \mathbb{R}^{d_w} \) be the feature embedding function. \( m_{ik} \) denotes the number of samples belonging to category \( k \). \( S_{ik} \) and \( D_{ik} \) are the sample set and data distribution for category \( k \) in task \( i \), respectively. Then, the expected prior predictor w.r.t. class \( k \) in task \( i \) is defined as:

\[
\mathbf{w}^P_i[k] = \Phi_v(D_{ik}) = \frac{1}{S_{ik}} \sum_{x_j \in S_{ik}} \phi_v(x_j).
\]

The empirical prior predictor w.r.t. class \( k \) in task \( i \) is defined as:

\[
\hat{\mathbf{w}}^P_i[k] = \hat{\Phi}_v(S_{ik}) = \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \phi_v(x_j).
\]

The LCC-based prior predictor w.r.t. class \( k \) in task \( i \) is defined as:

\[
\bar{\mathbf{w}}^P_i[k] = \bar{\Phi}_v(S_{ik}) = \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \sum_{u \in C} \gamma_u(x_j) \phi_v(u).
\]

B THEORETICAL RESULTS

B.1 PROOF OF LEMMA 1

This lemma bounds the error between the empirical prior predictor \( \hat{\mathbf{w}}^P_i[k] \) and the LCC-based prior predictor \( \bar{\mathbf{w}}^P_i[k] \).

Lemma 1 Given the definition of \( \hat{\mathbf{w}}^P_i[k] \) and \( \bar{\mathbf{w}}^P_i[k] \) in Eq. (6) and Eq. (7), let \((\gamma, C)\) be an arbitrary coordinate coding on \( \mathbb{R}^d \) and \( \phi \) be an \((\alpha, \beta)\)-Lipschitz smooth function and . We have for all \( x \in \mathbb{R}^d \):

\[
\| \hat{\mathbf{w}}^P_i[k] - \bar{\mathbf{w}}^P_i[k] \| \leq \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \left( \alpha \| x_j - \bar{x}_j \| + \beta \sum_{u \in C} \| x_j - u \|^2 \right) = O_{\alpha,\beta}(\gamma, C),
\]

where \( \bar{x}_j = \sum_{u \in C} \gamma_u(x_j) u \). Then given any \( \epsilon > 0 \), there exists a coding \((\gamma, C)\) such that

\[
|C| \leq (1 + d_M) N(\epsilon, M),
\]

\[
O_{\alpha,\beta}(\gamma, C) \leq [\alpha c_M + (1 + 5\sqrt{d_M}) \beta] \epsilon^2.
\]
Proof. Let \( \bar{x}_j = \sum_{u \in C} \gamma_u(x_j)u \). We have
\[
\|\hat{\Phi}_v(S_{ik}) - \hat{\Phi}_v(S_{ik})\|_2 = \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \|\Phi_v(x_j) - \sum_{u \in C} \gamma_u(x_j)\Phi_v(u)\|_2 \\
\leq \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \left( \|\Phi_v(x_j) - \Phi_v(\bar{x}_j)\|_2 + \|\sum_{u \in C} \gamma_u(x_j)\Phi_v(u) - \Phi_v(\sum_{u \in C} \gamma_u(x_j)u) - \nabla \Phi_v(\bar{x}_j)(u - \bar{x}_j)\|_2 \right) \\
\leq \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \left( \|\Phi_v(x_j) - \Phi_v(\bar{x}_j)\|_2 + \|\gamma_u(x_j)\| \|\Phi_v(u) - \Phi_v(\sum_{u \in C} \gamma_u(x_j)u) - \nabla \Phi_v(\bar{x}_j)(u - \bar{x}_j)\|_2 \right) \\
\leq \frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \left( \alpha \|x_j - \bar{x}_j\|_2 + \beta \|\sum_{u \in C} \gamma_u(x_j)\| \|\bar{x}_j - u\|_2^2 \right) = O_{\alpha, \beta}(\gamma, C)
\]

In the above derivation, the first inequality holds by the triangle inequality. The second equality holds since \( \sum_{u \in C} \gamma_u(x_j) = 1 \) for all \( x_j \). The last inequality uses the assumption of \((\alpha, \beta)\)-Lipschitz smoothness of \( \Phi_v(\cdot) \).

According to the Manifold Coding Theorem in [Yu et al., 2009], if the data points \( x \) lie on a compact smooth manifold \( M \). Then given any \( \epsilon > 0 \), there exists anchor points \( C \subset M \) and coding \( \gamma \) such that
\[
|C| \leq (1 + d_M)N(\epsilon, M), \\
\frac{1}{m_{ik}} \sum_{x_j \in S_{ik}} \left( \alpha \|x_j - \bar{x}_j\|_2 + \beta \|\sum_{u \in C} \gamma_u(x_j)\| \|\bar{x}_j - u\|_2^2 \right) \leq [\alpha c_M + (1 + 5\sqrt{d_M})\beta^2] \epsilon^2. \tag{18}
\]
This implies the desired bound. \( \square \)

The first inequality of this lemma demonstrates that the quality of LCC approximation is bounded by two terms: the first term \( \|x_j - \bar{x}_j\|_2 \) indicates \( x \) should be close to its physical approximation \( \bar{x} \), the second term \( \|\bar{x}_j - u\| \) implies that the coding should be localized. The second and third inequality show that the approximation error of local coordinate coding depends on the intrinsic dimension of the manifold instead of the dimension of input. If a set of anchor points \( C \) has cardinality \( O(d_MN(\epsilon, M)) \), empirical prior predictor can be linearly approximated using LCC up to accuracy \( O(\sqrt{d_M}\epsilon^2) \).

B.2 PROOF OF LEMMA 2

In order to prove Lemma 2, we first introduce a relevant theorem.

Theorem 4. (Vector-valued extension of McDiarmid's inequality (Rivasplata et al., 2018)) Let \( X_1, \ldots, X_m \in \mathcal{X} \) be independent random variables, and \( f : \mathcal{X}^m \rightarrow \mathbb{R}^m \) be a vector-valued mapping function. If, for all \( i \in \{1, \ldots, m\} \), and for all \( x_1, \ldots, x_m, x'_i \in \mathcal{X} \), the function \( f \) satisfies
\[
\sup_{x_i, x'_i} \|f(x_{1:i-1}, x_i, x_{i+1:m}) - f(x_{1:i-1}, x'_i, x_{i+1:m})\| \leq c_i \tag{19}
\]
Then \( \mathbb{E}[\|f(X_{1:m}) - \mathbb{E}[f(X_{1:m})]\|] \leq \sqrt{\sum_{i=1}^m c_i^2} \). For any \( \delta \in (0, 1) \) with probability \( \geq 1 - \delta \) we have
\[
\|f(X_{1:m}) - \mathbb{E}[f(X_{1:m})]\| \leq \sqrt{\sum_{i=1}^m c_i^2} + \sqrt{\frac{\sum_{i=1}^m c_i^2}{2}} \log\left(\frac{1}{\delta}\right). \tag{20}
\]
The above theorem indicates that bounded differences in norm implies the concentration of \( f(X_{1:m}) \) around its mean in norm, i.e., \( \|f(X_{1:m}) - \mathbb{E}[f(X_{1:m})]\| \) is small with high probability.
Then, we bound the error between expected prior predictor \( w_P^P \) and the empirical prior predictor \( \hat{w}_P^P \).

**Lemma 3.** Given the definition of \( w_P^P[k] \) and \( \hat{w}_P^P[k] \) in (6) and (7), let \( \mathcal{X} \) be a compact set with radius \( R \), i.e., \( \forall x, x' \in \mathcal{X}, \| x - x' \| \leq R \). For any \( \delta \in (0, 1) \) with probability \( \geq 1 - \delta \), we have

\[
\| w_P^P[k] - \hat{w}_P^P[k] \| \leq \frac{\alpha R}{\sqrt{m_k}} (1 + \sqrt{\frac{1}{2} \log \left( \frac{1}{\delta} \right)}). \tag{21}
\]

**Proof.** According to the definition of \( \hat{\Phi}_\nu(\cdot) \) in (6), for all points \( x_1, \ldots, x_{j-1}, x_j, \ldots, x_{m_k+1} \) in the sample set \( S_{ik} \), we have

\[
\frac{1}{m_k} \sup_{x_j, x'_j} | \Phi_\nu(x_j) - \Phi_\nu(x'_j) | \leq \frac{1}{m_k} \sup_{x_j, x'_j} \| x_j - x'_j \| \leq \frac{\alpha R}{m_k}, \tag{22}
\]

where \( R \) denotes the domain of \( x \), say \( R = \sup_x \| x \| \). The first inequality follows from the Lipschitz smoothness condition of \( \Phi_\nu(\cdot) \) and the second inequality follows by the definition of domain \( \mathcal{X} \). Utilizing Theorem 4 for any \( \delta \in (0, 1) \) with probability \( \geq 1 - \delta \) we have

\[
\| w_P^P[k] - \hat{w}_P^P[k] \| = \| \hat{\Phi}_\nu(S_{ik}) - E[\Phi_\nu(S_{ik})] \| \leq \frac{\alpha R}{\sqrt{m_k}} (1 + \sqrt{\frac{1}{2} \log \left( \frac{1}{\delta} \right)}). \tag{23}
\]

This implies the bound. \( \square \)

Lemma 3 shows that the bounded difference of \( \Phi_\nu(\cdot) \) implies its concentration, which can be further used to bound the differences between empirical prior predictor \( w_P^P[k] \) and expected prior predictor \( w_P^P[k] \). Now, we bound the error between expected prior predictor \( w_P^P \) and the LCC-based prior predictor \( \hat{w}_P^P \).

**Lemma 2** Given the definition of \( w_P^P \) and \( \hat{w}_P^P \) in (5), and (6), let \( \mathcal{X} \) be a compact set with radius \( R \), i.e., \( \forall x, x' \in \mathcal{X}, \| x - x' \| \leq R \). For any \( \delta \in (0, 1) \) with probability \( \geq 1 - \delta \), we have

\[
\| w_P^P - \hat{w}_P^P \|^2 \leq \sum_{k=1}^K \left( \frac{\alpha R}{\sqrt{m_k}} (1 + \sqrt{\frac{1}{2} \log \left( \frac{1}{\delta} \right)) + O_{\alpha, \beta}(\gamma, C) \right)^2. \tag{24}
\]

**Proof** According to the definition of \( w_P^P, \hat{w}_P^P \) and \( \hat{w}_P^P \), we have

\[
\| w_P^P - \hat{w}_P^P \|^2 = \sum_{k=1}^K \| \hat{w}_P^P[k] - \hat{w}_P^P[k] \|^2
\]

\[
= \sum_{k=1}^K \| E[\hat{\Phi}_\nu(S_{ik})] - \Phi_\nu(S_{ik}) + \hat{\Phi}_\nu(S_{ik}) - \Phi_\nu(S_{ik}) \|^2
\]

\[
= \sum_{k=1}^K \left( \| E[\hat{\Phi}_\nu(S_{ik})] - \Phi_\nu(S_{ik}) \|^2 + \| \Phi_\nu(S_{ik}) - \hat{\Phi}_\nu(S_{ik}) \|^2 + 2 \| E[\hat{\Phi}_\nu(S_{ik})] - \Phi_\nu(S_{ik}) \|^2 \right)
\]

\[
\leq \sum_{k=1}^K \left( \| E[\hat{\Phi}_\nu(S_{ik})] - \Phi_\nu(S_{ik}) \|^2 + \| \Phi_\nu(S_{ik}) - \Phi_\nu(S_{ik}) \|^2 + 2 \| E[\hat{\Phi}_\nu(S_{ik})] - \Phi_\nu(S_{ik}) \|^2 \right) \tag{25}
\]

Substitute Lemma 3 and Lemma 4 into the above inequality, we can derive

\[
\mathbb{P}_{S_{ik} \sim D} \left\{ \| w_P^P - \hat{w}_P^P \|^2 \leq \sum_{k=1}^K \left( \frac{\alpha R}{\sqrt{m_k}} (1 + \sqrt{\frac{1}{2} \log \left( \frac{1}{\delta} \right)) + O_{\alpha, \beta}(\gamma, C) \right)^2 \right\} \geq 1 - \delta. \tag{26}
\]
This gives the assertion.

Lemma[2] shows that the approximation error between expected prior predictor and LCC-based prior predictor depends on the number of samples in each category and the quality of the LCC coding scheme.

### B.3 PROOF OF THEOREM[2]

**Theorem 3** Let $Q$ be the posterior of base learner $Q = \mathcal{N}(w^Q, \sigma_w^2 I_{d_w})$ and $P$ be the prior $\mathcal{N}(\Phi_f(S), \sigma_f^2 I_{d_f})$. The mean of prior is produced by the LCC-based prior predictor $\Phi_f(S)$ in Eq. (7) and its parameter $v$ is sampled from the hyperposterior of meta learner $Q = \mathcal{N}(v^Q, \sigma_v^2 I_{d_v})$. Give the hyperprior $P = \mathcal{N}(0, \sigma_f^2 I_{d_f})$, then for any hyperposterior $Q$, any $c_1, c_2 > 0$ and any $\delta \in (0, 1]$ with probability $\geq 1 - \delta$ we have,

\[
er(Q) \leq c_1' c_2' \bar{\epsilon}(Q) + \left( \sum_{i=1}^{n} \frac{c_1'^2 c_2'^2}{2c_2 n m_i \sigma_v^2} + \frac{c_1'^2}{2c_1 n \sigma_v^2} \right) ||v^Q||^2 + \sum_{i=1}^{n} \frac{c_1'^2 c_2'^2}{2c_2 n m_i \sigma_v^2} \|\mathbb{E} w_i^Q - \Phi_f^Q(S_i)\|^2
\]

\[
+ \sum_{i=1}^{n} \frac{c_1'^2 c_2'^2}{2c_2 n m_i \sigma_v^2} \log \frac{4n}{\delta} + \frac{c_1'^2}{2c_1 n \sigma_v^2} \log \frac{2}{\delta},
\]

where $c_1' = \frac{c_1}{1 - e^{-c}}$ and $c_2' = \frac{c_2}{1 - e^{-c}}$. We can simplify the notation and obtain that

\[
er(Q) \leq c_1' c_2' \bar{\epsilon}(Q) + \left( \sum_{i=1}^{n} \frac{c_1'^2 c_2'^2}{2c_2 n m_i \sigma_v^2} + \frac{c_1'^2}{2c_1 n \sigma_v^2} \right) ||v^Q||^2 + \sum_{i=1}^{n} \frac{c_1'^2 c_2'^2}{2c_2 n m_i \sigma_v^2} \|\mathbb{E} w_i^Q - \Phi_f^Q(S_i)\|^2
\]

\[
+ \text{const}(\alpha, \beta, R, \delta, n, m_i).
\]

**Proof** Our proof contains two steps. First, we bound the error within observed tasks due to observing a limited number of samples. Then we bound the error on the task environment level due to observing a finite number of tasks. Both of the two steps utilize Catoni’s classical PAC-Bayes bound (Catoni 2007) to measure the error. We give here a general statement of the Catoni’s classical PAC-Bayes bound.

**Theorem 5. (Classical PAC-Bayes bound, general notations)** Let $\mathcal{X}$ be a sample space and $\mathcal{X}$ be some distribution over $\mathcal{X}$, and let $\mathcal{F}$ be a hypotheses space of functions over $\mathcal{X}$. Define a loss function $g(f, X) : \mathcal{F} \times \mathcal{X} \to [0, 1]$, and let $\mathcal{X}_i \triangleq \{X_1, \ldots, X_G\}$ be a sequence of $K$ independent random variables distributed according to $\mathcal{X}$. Let $\pi$ be some prior distribution over $\mathcal{F}$ (which must not depend on the samples $X_1, \ldots, X_K$). For any $\delta \in (0, 1]$, the following bounds holds uniformly for all posterior distribution $\rho$ over $\mathcal{F}$ (even sample dependent),

\[
\mathbb{P}_{X_i \sim X, \pi \sim \rho} \left\{ \mathbb{E}_{X \sim \mathcal{X}, \pi \sim \rho} g(f, X) \leq \frac{c}{1 - e^{-c}} \left[ \frac{1}{G} \sum_{g=1}^{G} \mathbb{E}_{f \sim \pi} g(f, X_k) + \frac{KL(\rho || \pi) + \log \frac{2}{\delta}}{K \times c} \right], \forall \rho \right\}
\]

\[
\geq 1 - \delta.
\]

**First step** We utilize Theorem[5] to bound the generalization error in each of the observed tasks. Let $i \in 1, \ldots, n$ be the index of task. For task $i$, we substitute the following definition into the Catoni’s PAC-Bayes Bound. Specifically, $X_g \triangleq (x_{ij}, y_{ij}), K \triangleq m_i$ denotes the samples and $\mathcal{X} \triangleq D_i$ denotes the data distribution. We instantiate the hypotheses with a hierarchical model $f \triangleq (v, w)$, where $v \in \mathbb{R}^{d_v}$ and $w \in \mathbb{R}^{d_w}$ are the parameters of meta learner (prior predictor) $\Phi_f(\cdot)$ and base learner $h(\cdot)$ respectively. The loss function only considers the base learner, which is defined as $g(f, X) \triangleq \ell(h_w(x), y)$. The prior over model parameter is represented as $\pi \triangleq (P, P) \triangleq (\mathcal{N}(0, \sigma_v^2 I_{d_v}), \mathcal{N}(w^P, \sigma_w^2 I_{d_w}))$, a Gaussian distribution (hyperprior of meta
learner) centered at 0 and a Gaussian distribution (prior of base learner) centered at \( w^P \), respectively. We set the posterior to \( \rho \triangleq (Q, Q) \triangleq (N(\nu, \sigma^2 I_d), N(w^Q, \sigma^2 w I_dw)) \), a Gaussian distribution (hyperposterior of meta learner) centered at \( \nu^Q \) and a Gaussian distribution (posterior of base learner) centered at \( w^Q \). According to Theorem 5, the generalization bound holds for any posterior distribution including the one generated in our localized meta-learning framework. Specifically, we first sample \( v \) from hyperposterior \( N(\nu^Q, \sigma^2_v I_dv) \) and estimate \( w^P \) by leveraging expected prior predictor \( \bar{\Phi}_v(D) \). The base learner algorithm \( A_b(S, P) \) utilizes the sample set \( S \) and the prior \( P = N(w^P, \sigma^2_w I_dw) \) to produce a posterior \( Q = A_b(S, P) = N(w^Q, \sigma^2_w I_dw) \). Then we sample base learner parameter \( w \) from posterior \( N(w^Q, \sigma^2_w I_dw) \) and compute the incurred loss \( \ell(h_w(x), y) \). On the whole, meta-learning algorithm \( A_m(S_1, \ldots, S_n, P) \) observes a series of tasks \( S_1, \ldots, S_n \) and adjusts its hyperprior \( P = N(\nu^P, \sigma^2_v I_dv) \) into hyperposterior \( Q = A_m(S_1, \ldots, S_n, P) = N(\nu^Q, \sigma^2_v I_dv) \).

The KL divergence term between prior \( \nu \) and posterior \( \rho \) is computed as follows:

\[
KL(\rho || \nu) = E \log \frac{\rho(f)}{\nu(f)} = E \log \frac{N(\nu^Q, \sigma^2_v I_dv)N(w^Q, \sigma^2_w I_dw)}{N(0, \sigma^2_v I_dv)N(w^P, \sigma^2_w I_dw)}
\]

\[
= E \log \frac{N(\nu^Q, \sigma^2_v I_dv)}{N(0, \sigma^2_v I_dv)} + E \log \frac{N(w^Q, \sigma^2_w I_dw)}{N(w^P, \sigma^2_w I_dw)}
\]

\[
= \frac{1}{2\sigma^2_v} \|w^Q\|^2 + \frac{1}{2\sigma^2_v} \|w^Q - w^P\|^2. \tag{30}
\]

In our localized meta-learning framework, in order to make \( KL(Q || P) \) small, the center of prior distribution \( w^P \) is generated by the expected prior predictor \( \bar{\Phi}_v(D) \). However, the data distribution \( D \) is considered unknown and our only insight as to \( D_{ik} \) is through the sample set \( S_{ik} \). In this work, we approximate the expected prior predictor \( \bar{\Phi}_v(D) \) with the LCC-based prior predictor \( \bar{\Phi}_v(S) \). Denote the term \( E \frac{1}{\sigma^2_w} \|w^Q - w^P\|^2 \) by \( E \frac{1}{\sigma^2_w} \|w^Q - w^P\|^2 \) for convenience, we have

\[
E \frac{1}{\sigma^2_w} \|w^Q - w^P\|^2 = E \frac{1}{\nu} \|w^Q - \bar{w}^P + \bar{w}^P - w^P\|^2
\]

\[
= E \frac{1}{\nu} \left[ \|w^Q - \bar{w}^P\|^2 + \|\bar{w}^P - w^P\|^2 + 2(w^Q - \bar{w}^P)^T (\bar{w}^P - w^P) \right]
\]

\[
\leq E \frac{1}{\nu} \left[ \|w^Q - \bar{w}^P\|^2 + \|\bar{w}^P - w^P\|^2 + 2\|w^Q - \bar{w}^P\| \|\bar{w}^P - w^P\| \right]
\]

\[
\leq \frac{1}{\sigma^2_w} E \|w^Q - \bar{\Phi}_v(S)\|^2 + \frac{1}{\sigma^2_{\nu}} \|\bar{w}^P - w^P\|^2. \tag{31}
\]

Since \( \bar{w}^P = \bar{\Phi}_v(S_i) = [\bar{\Phi}_v(S_{i1}), \ldots, \bar{\Phi}_v(S_{ik}), \ldots, \bar{\Phi}_v(S_{iK})] \), we have

\[
E \|w^Q - \bar{\Phi}_v(S_i)\|^2 = \sum_{k=1}^{K} E \|w^Q[k] - \bar{\Phi}_v(S_{ik})\|^2
\]

\[
= \sum_{k=1}^{K} \left( E \|w^Q[k]\|^2 - 2 \langle E \|\bar{w}^Q[k]\|^2, \bar{\Phi}_v(S_{ik}) \rangle + \|\bar{\Phi}_v(S_{ik})\|^2 + ||\bar{\Phi}_v(S_{ik})||^2 \right)
\]

\[
= \sum_{k=1}^{K} \left( \|w^Q[k] - \bar{\Phi}_v(S_{ik})\|^2 + \frac{d_{\nu}}{|C|} \sigma^2_{\nu} \right)
\]

\[
= \|w^Q - \bar{\Phi}_v(S_i)\|^2 + d_{\nu} K \sigma^2_{\nu}. \tag{32}
\]
where \(\mathbb{V}[\|\hat{\Phi}_v(S_{ik})\|]\) denotes the variance of \(\|\Phi_v(S_{ik})\|\). The last equality uses the fact that \(d_v = |C|d_w\). Combining Lemma 2 for any \(\delta' \in (0, 1)\) with probability \(\geq 1 - \delta'\) we have

\[
\mathbb{E} \frac{1}{\sqrt{2\sigma_w^2}} \|w_i^Q - w_i^P\|^2 \leq \frac{1}{\sigma_w^2} \mathbb{E} \left[ \|w_i^Q - \Phi_v \circ (S_i)\|^2 + d_w K\left(\frac{\sigma_v}{\sigma_w}\right)^2 + \frac{1}{\sigma_w^2} \sum_{k=1}^{K} \left( \frac{\alpha R}{\sqrt{m_{ik}}} \left( 1 + \sqrt{\frac{1}{2} \log\left(\frac{1}{\delta}\right)} + O_{\alpha,\beta}(\gamma, C) \right) \right)^2 \right]
\]

(33)

Then, according to Theorem 5, we obtain that for any \(\delta \leq \frac{\delta_i}{2} \geq 0\)

\[
\mathbb{P}_{S_i \sim D_i^m} \left\{ \mathbb{E}_{(x,y) \sim D_i, \mathcal{V} \sim N(\mathcal{V}^Q, \sigma^2_{\mathcal{V}} I_{d_w})} \mathbb{E}_{w \sim N(w^Q, \sigma^2_w I_{d_w})} \ell(h_w(x), y) \right\} \leq \frac{c_2}{1 - e^{-c_2}} \cdot \frac{1}{m_i} \sum_{j=1}^{m_i} \mathbb{E}_{v \sim N(V^Q, \sigma^2_{\mathcal{V}} I_{d_w})} \mathbb{E}_{w \sim N(w^Q, \sigma^2_w I_{d_w})} \ell(h_w(x_j), y_j) + \frac{1}{(1 - e^{-c_2})m_i} \left( \frac{1}{2\sigma_v^2} \|v\|^2 + \frac{1}{\sigma_v^2} \mathbb{E} \left[ \|w_i^Q - \Phi_v \circ (S_i)\|^2 + \log \frac{2}{\delta_i} + d_w K\left(\frac{\sigma_v}{\sigma_w}\right)^2 \right], \forall Q \right\} \geq 1 - \delta_i,
\]

(34)

for all observed tasks \(i = 1, \ldots, n\). Define \(\delta' = \frac{\delta_i}{2} \geq 0\) and combine inequality (33), we obtain

\[
\mathbb{P}_{S_i \sim D_i^m} \left\{ \mathbb{E}_{(x,y) \sim D_i, \mathcal{V} \sim N(\mathcal{V}^Q, \sigma^2_{\mathcal{V}} I_{d_w})} \mathbb{E}_{w \sim N(w^Q, \sigma^2_w I_{d_w})} \ell(h_w(x), y) \right\} \leq \frac{c_2}{1 - e^{-c_2}} \cdot \frac{1}{m_i} \sum_{j=1}^{m_i} \mathbb{E}_{v \sim N(V^Q, \sigma^2_{\mathcal{V}} I_{d_w})} \mathbb{E}_{w \sim N(w^Q, \sigma^2_w I_{d_w})} \ell(h_w(x_j), y_j)
\]

\[
+ \frac{1}{(1 - e^{-c_2})m_i} \left( \frac{1}{2\sigma_v^2} \|v\|^2 + \frac{1}{\sigma_v^2} \mathbb{E} \left[ \|w_i^Q - \Phi_v \circ (S_i)\|^2 + \log \frac{2}{\delta_i} + d_w K\left(\frac{\sigma_v}{\sigma_w}\right)^2 \right], \forall Q \right\} \geq 1 - \delta_i,
\]

(35)

Using the notations in Section 4, the above bound can be simplified as

\[
\mathbb{P}_{S_i \sim D_i^m} \left\{ \mathbb{E}_{v \sim N(V^Q, \sigma^2_{\mathcal{V}} I_{d_w})} \mathbb{E}_{w \sim N(w^Q, \sigma^2_w I_{d_w})} \mathbb{E}_{(x,y) \sim D_i, \mathcal{V} \sim N(\mathcal{V}^Q, \sigma^2_{\mathcal{V}} I_{d_w})} \mathbb{E}_{w \sim N(w^Q, \sigma^2_w I_{d_w})} \ell(h_w(x), y) \right\} \leq \frac{c_2}{1 - e^{-c_2}} \cdot \frac{1}{m_i} \sum_{j=1}^{m_i} \mathbb{E}_{v \sim N(V^Q, \sigma^2_{\mathcal{V}} I_{d_w})} \mathbb{E}_{w \sim N(w^Q, \sigma^2_w I_{d_w})} \ell(h_w(x_j), y_j)
\]

\[
+ \frac{1}{(1 - e^{-c_2})m_i} \left( \frac{1}{2\sigma_v^2} \|v\|^2 + \frac{1}{\sigma_v^2} \mathbb{E} \left[ \|w_i^Q - \Phi_v \circ (S_i)\|^2 + \log \frac{2}{\delta_i} + d_w K\left(\frac{\sigma_v}{\sigma_w}\right)^2 \right], \forall Q \right\} \geq 1 - \delta_i.
\]

(36)

**Second step** Next we bound the error due to observing a limited number of tasks from the environment. We use Theorem 5 with the following substitutions. The samples are \((D_i, m_i, S_i)\), \(i = 1, \ldots, n\), where \((D_i, m_i)\) are sampled from the same meta distribution \(\tau\) and \(S_i \sim D_i^m\). The hypothesis is parameterized as \(\Phi_v(D)\) with meta learner parameter \(v\). The loss function is \(g(f, X) \triangleq \mathbb{E}_{(x,y) \sim D_w} \mathbb{E}_{N(w^Q, \sigma^2_w I_{d_w})} \ell(h_w(x), y), \) where \(w^Q = A_0(S_i, P_i)\). Let \(\pi \triangleq N(0, \sigma_{\mathcal{V}}^2 I_{d_w})\) be
the prior over meta learner parameter, the following holds for any $\delta_0 > 0$,

$$
\begin{align*}
\mathbb{P}(D_{m_i}^{i}) \sim \tau, S_i \sim D_{m_i}, i = 1, ..., n \left\{ \mathbb{E}_{(D, m) \sim \tau} \mathbb{E}_{S \sim D_m} \mathbb{E}_{\nu \sim \mathcal{N}(\nu^Q, \sigma^2_{w} I_{d_w})} \mathbb{E}_{w^P = \Phi_{\nu}(D), P_i = \mathcal{N}(w^P, \sigma^2_{w} I_{d_w})} \ell(h_w(x), y) \right\} \\
\leq c_1 \gamma \cdot \frac{1}{1 - e^{-c_1}} \cdot \frac{1}{n} \sum_{i=1}^{n} \nu \sim \mathcal{N}(\nu^Q, \sigma^2_{w} I_{d_w}) \mathbb{E}_{w^P = \Phi_{\nu}(D), P_i = \mathcal{N}(w^P, \sigma^2_{w} I_{d_w})} \ell(h_w(x), y) \\
+ \frac{1}{(1 - e^{-c_1})n} \left( \frac{1}{2\sigma^2_{w}} \|\nu^Q\|^2 + \log \frac{1}{\delta_0} \right), \forall Q \right\} \geq 1 - \delta_0.
\end{align*}
$$

(37)

Using the term in Section 4, the above bound can be simplified as

$$
\begin{align*}
\mathbb{P}(D_{m_i}^{i}) \sim \tau, S_i \sim D_{m_i}, i = 1, ..., n \left\{ \mathbb{E} r(Q) \right\} \\
\leq c_1 \gamma \cdot \frac{1}{1 - e^{-c_1}} \cdot \frac{1}{n} \sum_{i=1}^{n} \nu \sim \mathcal{N}(\nu^Q, \sigma^2_{w} I_{d_w}) \mathbb{E}_{w^P = \Phi_{\nu}(D), P_i = \mathcal{N}(w^P, \sigma^2_{w} I_{d_w})} \mathbb{E} r(A_b(S_i, P_i)) \\
+ \frac{1}{(1 - e^{-c_1})n} \left( \frac{1}{2\sigma^2_{w}} \|\nu^Q\|^2 + \log \frac{1}{\delta_0} \right), \forall Q \right\} \geq 1 - \delta_0.
\end{align*}
$$

(38)

Finally, by employing the union bound, we could bound the probability of the intersection of the events in (36) and (38) For any $\delta > 0$, set $\delta_0 \leq \frac{\delta}{2}$ and $\delta_i \leq \frac{\delta}{2n}$ for $i = 1, ..., n$, we have

$$
\begin{align*}
\mathbb{P}(D_{m_i}^{i}) \sim \tau, S_i \sim D_{m_i}, i = 1, ..., n \left\{ \mathbb{E} r(Q) \right\} \\
\leq c_1 \gamma \cdot \frac{1}{1 - e^{-c_1}} \cdot \frac{1}{n} \sum_{i=1}^{n} \nu \sim \mathcal{N}(\nu^Q, \sigma^2_{w} I_{d_w}) \mathbb{E}_{w^P = \Phi_{\nu}(D), P_i = \mathcal{N}(w^P, \sigma^2_{w} I_{d_w})} \mathbb{E} r(A_b(S_i, P_i)) \\
+ \frac{1}{(1 - e^{-c_1})n} \left( \frac{1}{2\sigma^2_{w}} \|\nu^Q\|^2 + \log \frac{2}{\delta} \right), \forall Q \right\} \geq 1 - \delta.
\end{align*}
$$

(39)

We can further simplify the notation and obtain that

$$
\begin{align*}
\mathbb{P}(D_{m_i}^{i}) \sim \tau, S_i \sim D_{m_i}, i = 1, ..., n \left\{ \mathbb{E} r(Q) \right\} \leq c_1' c_2 \mathbb{E} r(Q) \\
+ \left( \sum_{i=1}^{n} \frac{c_4'}{2c_2'} \right) \|\nu^Q\|^2 + \left( \sum_{i=1}^{n} \frac{c_1'}{2c_2'} \right) \|\nu^Q\|^2 + \frac{1}{(1 - e^{-c_1})n} \left( \frac{1}{2\sigma^2_{w}} \|\nu^Q\|^2 + \log \frac{2}{\delta} \right), \forall Q \right\} \geq 1 - \delta.
\end{align*}
$$

(40)

where $c_1' = \frac{c_1}{1 - e^{-c_1}}$ and $c_2' = \frac{c_2}{1 - e^{-c_1}}$. This completes the proof.

\section*{B.4 Proof of Theorem 2}

\textbf{Theorem 2} Let $Q$ be the posterior of base learner $Q = \mathcal{N}(w^Q, \sigma^2_{w} I_{d_w})$ and $P$ be the prior $\mathcal{N}(w^P, \sigma^2_{w} I_{d_w})$. The mean of prior is sampled from the hyperposterior of meta learner $Q = \mathcal{N}(w^Q, \sigma^2_{w} I_{d_w})$. Give the hyperprior $P = \mathcal{N}(0, \sigma^2_{w} I_{d_w})$, then for any hyperposterior $Q$, any
$c_1, c_2 > 0$ and any $\delta \in (0, 1]$ with probability $\geq 1 - \delta$ we have,

$$er(Q) \leq c'_1 c'_2 \varepsilon r(Q) + \sum_{i=1}^{n} \frac{c'_1 c'_2}{2c_i^2 \sigma^2_w} \frac{2}{\sigma^2_w} \|w_i^Q - w_i^P\|^2$$

$$+ \sum_{i=1}^{n} \frac{c'_1 c'_2}{2c_i^2 \sigma^2_w} \left( \frac{1}{2} \log \frac{2n}{\delta} + \frac{c'_1}{c_i \sigma^2_w} \log \frac{2}{\delta} \right), \quad (41)$$

where $c'_1 = \frac{c_1}{1-e^{-c_1}}$ and $c'_2 = \frac{c_2}{1-e^{-c_2}}$.

**Proof** Instead of generating the mean of prior with a prior predictor, the vanilla meta-learning framework directly produces the mean of prior $w^D$ by sampling from hyperposterior $Q = \mathcal{N}(w^Q, \sigma^2_w I_{d_w})$. Then the base learner algorithm $A_b(S, P)$ utilizes the sample set $S$ and the prior $P = \mathcal{N}(w^P, \sigma^2_w I_{d_w})$ to produce a posterior $Q = A_b(S, P) = \mathcal{N}(w^Q, \sigma^2_w I_{d_w})$. Similarly with the two-steps proof in Theorem 3 we first get an intra-task bound by using Theorem 5. For any $\delta_i > 0$, we have

$$\mathbb{P}_{S_i \sim D_i} \left\{ \mathbb{E}_{w^P \sim N(w^Q, \sigma^2_w I_{d_w})} \left[ \frac{1}{2\sigma^2_w} \|w_i^Q - w_i^P\|^2 \right] \right\} \geq 1 - \delta_i, \quad (42)$$

The term $\mathbb{E}_{w^P \sim N(w^Q, \sigma^2_w I_{d_w})} \frac{1}{2\sigma^2_w} \|w_i^Q - w_i^P\|^2$ can be simplified as

$$\mathbb{E}_{w^P \sim N(w^Q, \sigma^2_w I_{d_w})} \frac{1}{2\sigma^2_w} \|w_i^Q - w_i^P\|^2$$

$$= \frac{1}{2\sigma^2_w} \left( \mathbb{E} \|w_i^Q\|^2 - 2 \mathbb{E}_{w^P} \mathbb{E}_{w^P} \|w_i^Q\|^2 + \mathbb{E}_{w^P} \|w_i^Q\|^2 + \mathbb{V}_{w^P} \|w_i^P\|^2 \right)$$

$$= \frac{1}{2\sigma^2_w} \left( \mathbb{E} \|w_i^Q\|^2 - \mathbb{E}_{w^P} \|w_i^Q\|^2 + \sigma^2_w \right), \quad (43)$$

where $\mathbb{V}_{w^P} \|w_i^P\|^2$ denotes the variance of $\|w_i^P\|$. Then we get an inter-task bound. For any $\delta_0 > 0$, we have

$$\mathbb{P}_{(D_i^m, \cdots, S_i, D_i^m)} \left\{ \mathbb{E}_{D_i^m} \mathbb{E}_{S_i} \mathbb{E}_{D_i^m} \mathbb{E}_{w^P \sim N(w^Q, \sigma^2_w I_{d_w})} \mathbb{E}_{w^Q \sim N(w^Q, \sigma^2_w I_{d_w})} \mathbb{E}_{(x,y) \sim D_i} \mathbb{E} \ell(h_w(x), y) \right\}$$

$$\leq \frac{c_1}{1 - e^{-c_1}} \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}_{w^P \sim N(w^Q, \sigma^2_w I_{d_w})} \mathbb{E}_{w^Q \sim N(w^Q, \sigma^2_w I_{d_w})} \mathbb{E}_{(x,y) \sim D_i} \mathbb{E} \ell(h_w(x), y)$$

$$+ \frac{1}{(1 - e^{-c_1})n} \left( \frac{1}{2\sigma^2_w} \|w_i^Q\|^2 + \log \frac{1}{\delta_0} \right), \quad \forall Q \geq 1 - \delta_0. \quad (44)$$
For any $\delta > 0$, set $\delta_0 \triangleq \frac{\delta}{2}$ and $\delta_i \triangleq \frac{\delta}{2^n}$ for $i = 1, \ldots, n$. Using the union bound, we finally get

\[
\mathbb{P}(D_i^{m_i}) \sim \tau, S_i \sim D_i^{m_i}, i = 1, \ldots, n \left\{ err(Q) \leq \frac{c_1 c_2}{(1 - e^{-c_1})(1 - e^{-c_2})} \cdot \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}_{v \sim \mathcal{N}(\mathbb{V}, \sigma_v^2 I_{4w})} \mathbb{E}_{w^p} \mathbb{E}_{\phi_v(D)} \mathbb{E}_{P_i = \mathcal{N}(w^p, \sigma_{w^p}^2 I_{4w})} \hat{e}r(A_b(S_i, P_i)) + \frac{c_1}{1 - e^{-c_1}} \cdot \frac{1}{n} \sum_{i=1}^{n} \frac{1}{(1 - e^{-c_2})} \cdot m_i \left( \frac{1}{2\sigma_w^2} \| w^Q \|^2 + \frac{1}{2\sigma^2 w} \| \mathbf{w} \|_p^2 + \frac{1}{2} + \log \frac{2n}{\delta} \right) + \frac{1}{(1 - e^{-c_1})n} \left( \frac{1}{2\sigma_w^2} \| w^Q \|^2 + \log \frac{2}{\delta} \right), \forall Q \right\} \geq 1 - \delta.
\]

Similarly, we can further simplify the notation and obtain that

\[
\mathbb{P}(D_i^{m_i}) \sim \tau, S_i \sim D_i^{m_i}, i = 1, \ldots, n \left\{ err(Q) \leq c'_1 c'_2 \hat{e}r(Q) + \frac{\sum_{i=1}^{n} c'_1 c'_2}{2c_1 n \sigma_w^2} \left( \| w^Q \|^2 + \sum_{i=1}^{n} \frac{c'_1 c'_2}{2c_2 n m_i \sigma_w^2} \| \mathbf{w} \|_p^2 - \| w^Q \|_p^2 \right) + \text{const}(\delta, n, m_i, \forall Q) \right\} \geq 1 - \delta,
\]

where $c'_1 = \frac{c_1}{1 - e^{-c_1}}$ and $c'_2 = \frac{c_2}{1 - e^{-c_2}}$. This completes the proof.

C DETAILS OF EXPERIMENTS

C.1 DATA PREPARATION

We used the 5-way 50-shot classification setups, where each task instance involves classifying images from 5 different categories sampled randomly from one of the meta-sets. We did not employ any data augmentation or feature averaging during meta-training, or any other data apart from the corresponding training and validation meta-sets.

C.2 NETWORK ARCHITECTURE

**Auto-Encoder for LCC** For CIFAR100, the encoder is 7 layers with 16-32-64-64-128-128-256 channels. Each convolutional layer is followed by a LeakyReLU activation and a batch normalization layer. The 1st, 3rd and 5th layer have stride 1 and kernel size (3, 3). The 2nd, 4th and 6th layer have stride 2 and kernel size (4, 4). The 7th layer has stride 1 and kernel size (4, 4). The decoder is the same as encoder except that the layers are in reverse order. The input is resized to 32 × 32. For Caltech-256, the encoder is 5 layers with 32-64-128-256-256 channels. Each convolutional layer is followed by a LeakyReLU activation and a batch normalization layer. The first 4 layers have stride 2 and kernel size (4, 4). The last layer has stride 1 and kernel size (6, 6). The decoder is the same as encoder except that the layers are in reverse order. The input is resized to 96 × 96.

**Base Model** The network architecture used for the classification task is a small CNN with 4 convolutional layers, each with 32 filters, and a linear output layer, similar to (Finn et al., 2017). Each convolutional layer is followed by a Batch Normalization layer, a LeakyReLU layer, and a max-pooling layer. For CIFAR100, the input is resized to 32 × 32. For Caltech-256, the input is resized to 96 × 96.

C.3 OPTIMIZATION

**Auto-Encoder for LCC** As optimizer we used Adam (Kingma & Ba, 2015) with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The initial learning rate is $1 \times 10^{-4}$. The number of epochs is 100. The batch size is 512.
**LCC Training** We alternatively train the coefficients and bases of LCC with Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. In specifics, for both datasets, we alternatively update the coefficients for 60 times and then update the bases for 60 times. The number of training epochs is 3. The number of bases is 64. The batch size is 256.

**Pre-Training of Feature Extractor** We use a 64-way classification in CIFAR-100 and 150-way classification in Caltech-256 to pre-train the feature embedding only on the meta-training dataset. For both CIFAR100 and Caltech-256, an L2 regularization term of $5e^{-4}$ was used. We used the Adam optimizer. The initial learning rate is $1 \times 10^{-3}$, $\beta_1$ is 0.9 and $\beta_2$ is 0.999. The number of epochs is 50. The batch size is 512.

**Meta-Training** We use the cross-entropy loss as in [Amit & Meir (2018)](ref). Although this is inconsistent with the bounded loss setting in our theoretical framework, we can still have a guarantee on a variation of the loss which is clipped to $[0, 1]$. In practice, the loss is almost always smaller than one. For CIFAR100 and Caltech-256, the number of epochs of meta-training phase is 12; the number of epochs of meta-testing phase is 40. The batch size is 32 for both datasets. As optimizer we used Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. In the setting with a pre-trained base model, the learning rate is $1 \times 10^{-5}$ for convolutional layers and $5 \times 10^{-4}$ for the linear output layer. In the setting without a pre-trained base model, the learning rate is $1 \times 10^{-3}$ for convolutional layers and $5 \times 10^{-3}$ for the linear output layer. The confidence parameter is chosen to be $\delta = 0.1$. The variance hyper-parameter for prior predictor and base model are $\sigma_w = \sigma_v = 0.01$. The hyperparameters $\alpha_1, \alpha_2$ in LML and ML-A are set to 0.01. For MAML [Finn et al. (2017)](ref) and MatchingNet [Vinyals et al. (2016)](ref), both two methods use the Adam optimizer with initial learning rate 0.0001. In the meta-training phase, we randomly split the samples of each class into support set (5 samples) and query set (45 samples). The number of epochs is 100. For MAML, the learning rate of inner update is 0.01.