Unlabeled Data Can Provably Enhance In-Context Learning of Transformers

Renpu Liu
University of Virginia
Charlottesville, VA 22903
renpu@virginia.edu

Jing Yang
University of Virginia
Charlottesville, VA 22903
yangjing@virginia.edu

Abstract

Large language models (LLMs) exhibit impressive in-context learning (ICL) capabilities, yet the quality of their predictions is fundamentally limited by the few costly *labeled* demonstrations that can fit into a prompt. Meanwhile, there exist vast and continuously growing amounts of *unlabeled* data that may be closely related to the ICL task. How to utilize such unlabeled data to provably enhance the performance of ICL thus becomes an emerging fundamental question. In this work, we propose a novel *augmented ICL* framework, in which the prompt includes a small set of labeled examples alongside a block of unlabeled inputs. We focus on the multi-class linear classification setting and demonstrate that, with chain-of-thought (CoT) prompting, a multi-layer transformer can effectively emulate an expectation–maximization (EM) algorithm. This enables the transformer to implicitly extract useful information from both labeled and unlabeled data, leading to provable improvements in ICL accuracy. Moreover, we show that such a transformer can be trained via teacher forcing, with its parameters converging to the desired solution at a linear rate. Experiments demonstrate that the augmented ICL framework consistently outperforms conventional few-shot ICL, providing empirical support for our theoretical findings. *To the best of our knowledge, this is the first theoretical study on the impact of unlabeled data on the ICL performance of transformers.*

1 Introduction

Since the introduction (Vaswani et al., 2017), transformers have become foundational models in diverse fields such as natural language processing (Radford, 2018; Devlin et al., 2019), computer vision (Dosovitskiy, 2020), and reinforcement learning (Chen et al., 2021). A key driver of their impact is the remarkable capability for In-Context Learning (ICL) (Brown et al., 2020). Without requiring parameter updates, transformers performing ICL can adapt to new tasks based solely on contextual examples provided within the prompt. This enables state-of-the-art few-shot performance across a multitude of applications, including reasoning and language understanding (Chowdhery et al., 2023), dialog generation (Thoppilan et al., 2022), and linear regression (Garg et al., 2022; Fu et al., 2023), etc.

Despite the power of ICL, its reliance on labeled examples presents a significant bottleneck for large language models (LLMs). Acquiring high-quality labeled data is oin general expensive and time-consuming (Zhou et al., 2023; Chung et al., 2024; Sun et al., 2023; Wang et al., 2023). For example, creating the instruction-tuning and RLHF datasets for models like GPT-3.5 and GPT-4 involved thousands of expert annotator hours, yet constituted less than 0.1% of the tokens encountered during pre-training (Ouyang et al., 2022; Achiam et al., 2023).

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Some existing approaches attempt to mitigate labeled data scarcity in ICL. For instance, Wan et al. (2023); Chen et al. (2025a) use an LLM to automatically generate pseudo-demonstrations at inference time by pairing unlabeled queries with the model's own predictions as pseudo labels. However, model-generated pseudo-labels inevitably inherit the biases and error patterns of the teacher model, resulting in noisy demonstrations that may limit potential performance gains.

In this work, instead of synthesizing examples with pseudo-labels, we explore a different approach by directly utilizing abundant and continuously growing (Raffel et al., 2020; Touvron et al., 2023) unlabeled data during ICL. The fundamental question we aim to answer is:

Can we provably enhance the ICL performance of transformers by effectively leveraging plentiful unlabeled data alongside limited labeled examples?

We answer this question affirmatively from a new *augmented in-context learning* perspective. This paradigm involves prompting a transformer with a mixture of a few labeled examples and numerous unlabeled examples, aiming to infer the missing labels within a single forward pass. By reasoning over unlabeled examples directly in the prompt, it bypasses the need for potentially costly, time-consuming, and bias-introducing labeling or pseudo-label generation steps in conventional ICL. In this work, we focus on augmented ICL for *multi-class linear classification*. Our main contributions are as follows.

- **Expressiveness with CoT Prompting.** First, we show that through Chain-of-Thought (CoT) prompting, a multi-layer transformer can leverage both labeled and unlabeled data to effectively solve the multi-class linear classification problem during ICL. Essentially, the transformer is able to obtain an initial estimation of the mean vectors of classes using the *labeled* data, and then iteratively refine the estimates by clustering the *unlabeled* data in an Expectation–Maximization (EM) fashion. We explicitly characterize the design of the transformer and theoretically prove that the class mean estimation will converge to the ground truth as the CoT steps increase. For a prompt consisting of N labeled and M unlabeled samples, our results indicate that the excess risk of our approach scales in $\mathcal{O}\left(1/\sqrt{N\,\mathrm{poly}(M)}\right)$, strictly improving the excess risk lower bound of $\mathcal{O}\left(1/\sqrt{N}\right)$ for any classifier that utilizes N labeled samples only. Our results indicate that the augmented ICL can effectively utilize the information from the unlabeled data, enabling steady performance improvement as unlabeled data increases.
- **Training Convergence under Teacher Forcing.** Second, we prove that, with proper initialization, when applying gradient descent on the population loss defined through teacher forcing, the tunable parameters of the transformer converge to the desired solution linearly. Thus, the trained transformer can mimic the EM algorithm through CoT prompting during inference, theoretically validating the transformer's expressiveness for augmented ICL. Our proof involves a novel decomposition of the gradient of the CoT training loss into two analytically tractable terms. For each of them, we leverage the inherent isotropy of the involved quantities to simplify the analysis, which enables us to derive a tight upper bound on the critical inner-product term and obtain the linear convergence rate.
- **Empirical Results.** Finally, we evaluate the performance of augmented ICL in transformers trained via teacher forcing. Our experimental results show that the augmented ICL approach significantly outperforms conventional ICL in both class mean estimation and label prediction, with the advantage becoming more pronounced as the number of unlabeled data samples increases. Moreover, augmented ICL surpasses the Bayes-optimal classifier that relies solely on labeled data. These empirical observations are consistent with our theoretical findings.

2 Related Works

ICL with Transformers. Brown et al. (2020) first shows that GPT-3, a transformer-based LLM, can perform new tasks from input-output pairs without parameter updates, suggesting its ICL ability. This intriguing phenomenon of transformers has attracted much attention, leading to various interpretations and hypotheses about its underlying mechanism. Research on ICL often demonstrates how transformers can emulate learning algorithms. For instance, several studies have designed transformers that execute gradient descent for linear and non-linear regression tasks (Akyürek et al., 2023; Von Oswald et al., 2023a) . Recent works demonstrate that transformers can implement more advanced optimization algorithms other than vanilla gradient descent on various ICL tasks (Bai et al.,

2024; Von Oswald et al., 2023b; Zhang et al., 2024a; Ahn et al., 2024; Liu et al., 2025). Another line of research adopts a statistical perspective: ICL can be viewed as an implicit form of Bayesian updating based on the examples provided in the prompt, with the diversity of pretraining data shaping the prior (Xie et al., 2022; Raventós et al., 2023; Garg et al., 2022).

Several studies (Gupta et al., 2024; Agarwal et al., 2024) investigate "unsupervised ICL", in which the prompt consists solely of unlabeled inputs. Another line of work leverages LLMs to generate pseudo-labels for unlabeled data, which are then used as demonstrations during ICL (Chen et al., 2023; Wan et al., 2023; Yang et al., 2023; Chen et al., 2025a). Our work leverages both labeled and unlabeled examples within the prompt to enhance ICL performance in a *semi-supervised learning* manner, which stands in sharp contrast to the aforementioned studies.

Notably, a recent concurrent work (Li et al., 2025) also investigates the impact of the semi-supervised data model on the ICL performance of transformers. Specifically, Li et al. (2025) focus on a linear transformer without nonlinear activations in a binary classification setting, and characterize the asymptotic ICL performance as the number of unlabeled samples approaches infinity. In contrast, we study a more realistic architecture that incorporates the softmax attention mechanism and establish a *non-asymptotic* convergence guarantee in the general *multi-class* setting.

Training Dynamics of Transformers. A number of recent works aim to provide theoretical characterizations of the training dynamics of transformers. Ahn et al. (2024); Mahankali et al. (2023); Zhang et al. (2024a); Huang et al. (2023) investigate the training dynamics of transformers with a single attention layer and a single head for in-context linear regression tasks. Cui et al. (2024) prove that transformers with multi-head attention layers outperform those with single-head attention. Cheng et al. (2024) show that local optimal solutions in transformers can perform gradient descent in-context for non-linear functions. Kim and Suzuki (2024) study the non-convex mean-field dynamics of transformers, and Nichani et al. (2024) characterize the convergence rate for the training loss in learning a causal graph. Additionally, Chen et al. (2024) investigate the gradient flow in training multi-head single-layer transformers for multi-task linear regression. Chen and Li (2025) propose a supervised training algorithm for multi-head transformers. The training dynamics of transformers for binary classification (Tarzanagh et al., 2023b,a; Vasudeva et al., 2024; Li et al., 2023; Deora et al., 2023; Li et al., 2024a), multi-class classification (Shen et al., 2025) and next-token prediction (Tian et al., 2023, 2024; Li et al., 2024b; Huang et al., 2024) have also been studied recently.

Transformers with CoT. In language modeling tasks, transformers have been proven to be powerful across various downstream tasks. However, transformers struggle to solve mathematical or scientific problems with a single generation, particularly when multiple reasoning steps are required. CoT prompting is introduced to enable transformers to generate intermediate results autoregressively before reaching the final answer, and has been shown to boost performance on arithmetic, commonsense, and scientific tasks (Wei et al., 2022; Kojima et al., 2022)

Recently, the training dynamics of transformers with CoT have been studied in Huang et al. (2025a) for weight prediction in linear regression, in Li et al. (2024a) for in-context supervised learning, in Kim and Suzuki (2025); Wen et al. (2025) for the parity problems, and in Huang et al. (2025b) for the even pairs problem. None of these studies, however, address whether the multi-step reasoning capacity through CoT can be utilized to extract information from *unlabeled* inputs.

3 Preliminaries

Notations. For matrix $\mathbf{X}$, we use $[\mathbf{X}]_{p:q,r:s}$ to denote the submatrix that contains rows p to q and columns r to s, and we use $[\mathbf{X}]_{:,i}$ and $[\mathbf{X}]_{j,:}$ to denote the i-th column and j-th row of $\mathbf{X}$, respectively. For convenience, we occasionally denote the i-th column $\mathbf{X}$ by $[\mathbf{X}]_i$ when no ambiguity arises. $[\mathbf{X}]_{:,-C:-1}$ means the last C columns of matrix $\mathbf{X}$. We use $\|\mathbf{X}\|_F$ to denote its Frobenius norm. For vector $\mathbf{x}$, we use $\|\mathbf{x}\|_1$, $\|\mathbf{x}\|$ and $\|\mathbf{x}\|_\infty$ to denote its ℓ_1, ℓ_2 and ℓ_∞ norms, respectively. We denote by $\mathbb{1}_d$ and $\mathbf{0}_d$ the d-dimensional all-1 and all-0 column vectors, respectively. $\mathbb{1}_{a\times b}$ and $\mathbf{0}_{a\times b}$ denote the all-1 and all-0 matrices of size $a \times b$, respectively. We denote the indicator function as $\mathbf{1}_{\{A\}}$, which equals 1 if event A is true.

3.1 Transformer Architecture

In this work, we consider the encoder-based transformer architecture (Vaswani et al., 2017), where each transformer layer consists of an attention layer followed by a multi-layer perception (MLP) layer.

Definition 3.1 (Attention layer)**.** *Denote an M-head attention layer parameterized by $\{(\mathbf{V}_m, \mathbf{Q}_m, \mathbf{K}_m)_{m\in[M]}\}$ as $\mathrm{attn}_{\{(\mathbf{V}_m,\mathbf{Q}_m,\mathbf{K}_m)\}}(\cdot)$, where $\mathbf{V}_m, \mathbf{Q}_m, \mathbf{K}_m \in \mathbb{R}^{D\times D}$, $\forall m \in [M]$. Then, given an input sequence $\mathbf{H} \in \mathbb{R}^{D\times(N+1)}$, the output sequence of the attention layer is*

$$\mathrm{attn}_{\{(\mathbf{V}_m,\mathbf{Q}_m,\mathbf{K}_m)\}}(\mathbf{H}) = \mathbf{H} + \sum_{m=1}^{M} (\mathbf{V}_m\mathbf{H}) \times \sigma\big((\mathbf{K}_m\mathbf{H})^\top(\mathbf{Q}_m\mathbf{H})\big),$$

where σ is a non-linear activation function.

Definition 3.2 (MLP layer)**.** *Given $\mathbf{W}_1 \in \mathbb{R}^{D'\times D}$, $\mathbf{W}_2 \in \mathbb{R}^{D\times D'}$ and a bias vector $\mathbf{b} \in \mathbb{R}^{D'}$, an MLP layer following the decoder attention layer, denoted as $\mathrm{MLP}_{\{\mathbf{W}_1,\mathbf{W}_2,\mathbf{b}\}}$, maps each token in the input sequence (i.e, each column $\mathbf{h}_i$ in $\mathbf{H} \in \mathbb{R}^{D\times N}$) to another token as*

$$\mathrm{MLP}_{\{\mathbf{W}_1,\mathbf{W}_2,\mathbf{b}\}}(\mathbf{h}_i) = \mathbf{h}_i + \mathbf{W}_2\sigma(\mathbf{W}_1\mathbf{h}_i + \mathbf{b}),$$

where σ is a non-linear activation function.

3.2 Augmented In-context Learning

Conventional In-Context Learning (ICL). For an ICL task, a trained transformer is given an ICL instance $\mathcal{I} = (\mathcal{D}, \mathbf{x}_{N+1})$, where $\mathcal{D} = \{(\mathbf{x}_j, y_j)\}_{j\in[N]}$ and $\mathbf{x}_{N+1}$ is a query. Here, $\mathbf{x}_j \in \mathbb{R}^d$ is an in-context example, and y_j is the corresponding label for $\mathbf{x}_j$. For each instance, $\{(\mathbf{x}_j, y_j)\}_{j=1}^{N+1}$ are generated independently accordingly to an underlying distribution. The objective of ICL is to predict y_{N+1} without any parameter updating of the transformer.

Augmented ICL. In this work, we consider a new unlabeled data augmented ICL framework. Specifically, each ICL instance now comprises a set of labeled examples, $\mathcal{D}_{\text{label}} := \{(\mathbf{x}_j, y_j)\}_{j=1}^{N}$, and a set of unlabeled examples, $\mathcal{D}_{\text{unlabel}} := \{\mathbf{x}_j\}_{j=N+1}^{N+M}$, i.e., $\mathcal{I} = \mathcal{D}_{\text{label}} \cup \mathcal{D}_{\text{unlabel}}$. Similar to conventional ICL, all $(\mathbf{x}_j, y_j)$ pairs follow the same distribution. The objective of augmented ICL is then to predict labels for all the M unlabeled samples in $\mathcal{D}_{\text{unlabel}}$.

We note that the augmented ICL generalizes the conventional ICL framework, and reduces to it when $M = 1$. While the conventional ICL can be utilized to solve the prediction for those M unlabeled samples individually *in parallel*, by augmenting them in the same ICL instance, it provides an opportunity for the transformer to extract common statistical information in those unlabeled data, which can be utilized to improve the joint prediction accuracy.

Augmented ICL for Multi-class Linear Classification. We consider augmented ICL for a multi-class linear classification problem. We assume there exist C classes, and the label space $\mathcal{Y}$ consists of one-hot vectors $\{\mathbf{e}_1, \ldots, \mathbf{e}_C\}$, where each $\mathbf{e}_i \in \mathbb{R}^C$ is the i-th unit vector. For each ICL instance $\mathcal{I}_{\mathbf{M}}$, the samples are randomly generated according to

$$\mathbf{M} \sim P_{\mathbf{M}}, \quad \mathbf{y}_j \sim \mathrm{Uniform}(\mathcal{Y}), \quad \boldsymbol{\epsilon}_j \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}), \quad \mathbf{x}_j = \mathbf{M}\mathbf{y}_j + \boldsymbol{\epsilon}_j, \quad j \in [M+N], \tag{3.1}$$

where $\mathbf{M} \in \mathbb{R}^{d\times C}$ and $P_{\mathbf{M}}$ is a prior distribution over $\mathbb{R}^{d\times C}$. Denote the columns of $\mathbf{M}$ as $\{\boldsymbol{\mu}_i\}_{i=1}^{C}$. Then, each $\mathbf{x}_j$ essentially follows a C-component mixture Gaussian distribution parametrized by mean vectors $\{\boldsymbol{\mu}_i\}_{i=1}^{C}$ and shared covariance matrix $\boldsymbol{\Sigma}$. In this work, we assume $\boldsymbol{\Sigma}$ is isotropic. We adopt this assumption for theoretical tractability, as it is crucial for deriving the closed-form update rules for the transformer. This approach is a standard and widely adopted practice in related literature to facilitate theoretical analysis (He et al., 2025; Zhang et al., 2024b; Chen et al., 2025b).

3.3 Chain-of-Thought Prompting for Augmented ICL

The core challenge in augmented ICL is leveraging both unlabeled data and labeled examples to infer task structure from a single instance. Unlike standard few-shot ICL, which often uses direct pattern matching, the augmented ICL requires more complex inference to effectively utilize the larger unlabeled set, making a simple one-step prediction insufficient.

Chain-of-Thought (CoT) reasoning offers a promising way to enhance a transformer's ICL capabilities. This is crucial for augmented ICL, as it enables the transformer to effectively utilize unlabeled data through iterative latent parameter estimation and refinement.

To implement augmented in-context learning via CoT prompting, we first encode a task instance $\mathcal{I}$ into an embedding matrix $\mathbf{H}$ by concatenating three column blocks: the labeled example block, the unlabeled example block, and the inference prompt block as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{x}_1 & \cdots & \mathbf{x}_N & \mathbf{x}_{N+1} & \cdots & \mathbf{x}_{N+M} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{y}_1 & \cdots & \mathbf{y}_N & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{p}_1 & \cdots & \mathbf{p}_N & \mathbf{p}_{N+1} & \cdots & \mathbf{p}_{N+M} & \mathbf{q}_1 & \cdots & \mathbf{q}_C \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{X}_\ell & \mathbf{X}_u & \mathbf{0} \\ \mathbf{Y}_\ell & \mathbf{0} & \mathbf{0} \\ \mathbf{P}_\ell & \mathbf{P}_u & \mathbf{Q}^{(0)} \end{bmatrix}, \quad (3.2)$$

where $\mathbf{p}_j \in \mathbb{R}^{d_p}$ is an auxiliary embedding that stores the (predicted) classification probability vector for the j-th sample, as well as a binary indicator to distinguish the labeled and unlabeled data. $\mathbf{q}_i \in \mathbb{R}^{d_p}$ serves as the initial CoT token for class i, which contains the one-hot vector $\mathbf{e}_i$ to indicate the corresponding class, and an all-zero vector representing the transformer's initial estimate for the mean vector $\boldsymbol{\mu}_i$.

Denote a trained transformer with parameter $\boldsymbol{\Theta}$ as $\mathrm{TF}_{\boldsymbol{\Theta}}$. With CoT, we will use the transformer to generate T intermediate steps before it outputs the prediction. Specifically, let $\widehat{\mathbf{H}}^{(t-1)}$ be the input sequence at the t-th step of CoT, where $\widehat{\mathbf{H}}^{(0)} = \mathbf{H}$, and $\mathrm{TF}_{\boldsymbol{\Theta}}(\widehat{\mathbf{H}}^{(t-1)})$ as the corresponding output of the transformer. Then, we will take out the last C columns of $\mathrm{TF}_{\boldsymbol{\Theta}}(\widehat{\mathbf{H}}^{(t-1)})$, and append it to the end of $\widehat{\mathbf{H}}^{(t-1)}$ to form the input for the next CoT step. Specifically, we have

$$\widehat{\mathbf{H}}^{(t)} = \left[\widehat{\mathbf{H}}^{(t-1)}, [\mathrm{TF}_{\boldsymbol{\Theta}}(\widehat{\mathbf{H}}^{(t-1)})]_{:,-C:-1}\right] = \begin{bmatrix} \mathbf{X}_\ell & \mathbf{X}_u & \mathbf{0} & \star & \cdots & \star \\ \mathbf{Y}_\ell & \mathbf{0} & \mathbf{0} & \star & \cdots & \star \\ \mathbf{P}_\ell & \mathbf{P}_u & \mathbf{Q}^{(0)} & \mathbf{Q}^{(1)} & \cdots & \mathbf{Q}^{(t)} \end{bmatrix}, \quad (3.3)$$

where

$$\mathbf{Q}^{(t)} = \begin{bmatrix} \mathbf{e}_1 & \cdots & \mathbf{e}_C \\ \widehat{\boldsymbol{\mu}}_1^{(t)} & \cdots & \widehat{\boldsymbol{\mu}}_C^{(t)} \\ \star & \cdots & \star \end{bmatrix}. \quad (3.4)$$

Here $\star$ is a placeholder for dummy tokens, $\mathbf{e}_i$ is the i-th unit vector, and $\widehat{\boldsymbol{\mu}}_i^{(t)}$ is the estimated mean vector for class i at the t-th CoT step.

After T iterations, we read out $\widehat{\boldsymbol{\mu}}_1^{(T)} \cdots \widehat{\boldsymbol{\mu}}_C^{(T)}$ from $\mathbf{Q}^{(T)}$ as the final estimation of the class mean vectors. Then, the label of each unlabeled data can be estimated through a maximum likelihood estimation, i.e.,

$$\widehat{\mathbf{y}}_j = \left\{ \mathbf{e}_i : i = \arg\min_{i \in [C]} \left\| \mathbf{x}_j - \widehat{\boldsymbol{\mu}}_i^{(T)} \right\| \right\}, \quad j \in [N+1 : N+M]. \quad (3.5)$$

4 Expressiveness with CoT Prompting for Augmented ICL

In this section, we show that a multi-layer transformer *can* implement an Expectation-Maximization (EM)-style algorithm to extract useful statistical information from the unlabeled data, which will be combined with information extracted from the labeled data to jointly estimate the class means and improve the augmented ICL performance. Specifically, we have the following result.

Theorem 4.1. *There exists a 4-layer transformer, such that its output sequence at the $(t+1)$-th CoT step satisfies*

$$\widehat{\boldsymbol{\mu}}_i^{(t+1)} = \widehat{\boldsymbol{\mu}}_i^{(t)} - \frac{\eta^{(t)}}{M} \sum_{j=N+1}^{N+M} p_{ij}^{(t)} \left(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j \right) + \mathbf{1}_{\{t=0\}} \cdot \frac{C}{N} \sum_{j=1}^{N} (\mathbf{e}_i^\top \mathbf{y}_j) \mathbf{x}_j, \quad (4.1)$$

for any $i \in [C]$, where $\eta^{(t)} = \alpha/(T'+t)$ for some positive constants α and T', $p_{ij}^{(t)}$ is the normalized weight

$$p_{ij}^{(t)} = \frac{\sum_{\tau=0}^{t} \exp\left(-\frac{1}{2} \|\widehat{\boldsymbol{\mu}}_i^{(\tau)} - \mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2 + \beta\tau \right)}{\sum_{\tau=0}^{t} \sum_{c=1}^{C} \exp\left(-\frac{1}{2} \|\widehat{\boldsymbol{\mu}}_c^{(\tau)} - \mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2 + \beta\tau \right)}, \quad (4.2)$$

and β is a positive constant.

We outline the construction of each layer of the transformer below, and defer the detailed derivation and specific parameter implementation to Appendix C.

The four-layer architecture is designed to mirror an EM iteration for Gaussian mixture model clustering (Zhao et al., 2020; Sula and Zheng, 2022) within the transformer's forward pass. The EM algorithm operates iteratively. First, in the **E-step**, it utilizes the current class mean estimates embedded in the input sequence to compute the estimated class membership for each *unlabeled* data point. Subsequently, the **M-step** updates the class mean estimates by performing a maximum likelihood estimation of the unlabeled data, and then combining them with the estimates obtained from the *labeled* data. Through this iterative process, the algorithm can achieve an accurate estimate of the underlying class means, enabling accurate classification.

The first layer. The first transformer layer includes a *softmax-activated* attention layer followed by an MLP layer. We construct its parameters so that it outputs the class membership estimate for the each unlabeled sample as in the form of Equation (4.2), where the mean estimates $\{\widehat{\boldsymbol{\mu}}_1^{(\tau)}, \cdots, \widehat{\boldsymbol{\mu}}_C^{(\tau)}\}_{\tau=1}^t$ are embedded in the reasoning blocks $\mathbf{Q}^{(1)} \cdots \mathbf{Q}^{(t)}$ in the input sequence, and the parameter β is embedded in the first layer as well. This probability represents how likely sample j is estimated to be in class i. Since the temperature parameter $\beta\tau$ is proportional to the step index τ, estimates from earlier CoT steps carry less importance. In the limit of $\beta \to \infty$, the weight vector depends only on the latest CoT step, i.e.,

$$p_{ij}^{(t)} = \frac{\exp\Big(-\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2\Big)}{\sum_{c=1}^C \exp\Big(-\frac{1}{2}\|\boldsymbol{\mu}_c^{(t)} - \mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2\Big)}. \tag{4.3}$$

The second and third layers. The second and third transformer layers consist of a *linear* attention layer followed by an MLP layer. These layers are designated for the M-step (Maximization step) of the EM algorithm. In this step, the class mean estimates $\{\widehat{\boldsymbol{\mu}}_i\}_{i=1}^C$ are updated by maximizing the overall log-likelihood of the unlabeled data with the estimated class membership probabilities $p_{ji}^{(t)}$. It aims to solve

$$P_1: \quad \{\boldsymbol{\mu}_c^{(t+1)}\}_c = \arg\max_{\{\mu_c\}_c} \sum_{j=N+1}^{N+M} \sum_{i=1}^{C} p_{ji}^{(t)} \log \mathcal{N}\big(\mathbf{x}_j;\, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}\big).$$

The implementation for these two layers is equivalent to tasking one step of gradient descent over P_1, i.e.,

$$\widehat{\boldsymbol{\mu}}_i^{(t+1)} = \widehat{\boldsymbol{\mu}}_i^{(t)} - \frac{\eta^{(t)}}{M} \sum_{j=N+1}^{N+M} p_{ij}^{(t)} \big(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j\big). \tag{4.4}$$

The fourth layer. Finally, the last transformer layer includes a *ReLU-activated* attention layer followed by an MLP layer. This layer calculates the initial class mean estimates for the *labeled* dataset and is only activated at the first CoT step. It implements the following updating rule:

$$\widehat{\boldsymbol{\mu}}_i^{(t+1)} = \mathbf{1}_{\{t=0\}} \cdot \frac{C}{N} \sum_{j=1}^{N} (\mathbf{e}_i^\top \mathbf{y}_j)\mathbf{x}_j, \tag{4.5}$$

which initialize $\widehat{\boldsymbol{\mu}}_i^1$ to be the average of $\mathbf{x}_j$'s for the labeled data samples in class i. This initialization will be refined iteratively through the CoT steps by leveraging the information from the unlabeled data.

We note that the parameters of the last three layers are data-independent and can be explicitly constructed beforehand, and only the parameters of the first layer depend on the distribution of the data, which can be obtained through CoT training, as elaborated in Section 5.

Next, we will show that the transformer specified in Theorem 4.1 will recover $\{\mu_i\}_{i=1}^C$ accurately with high probability, and explicitly characterize the benefit of unlabeled data in this augmented ICL.

Theorem 4.2 (Class Mean Estimation Error)**.** *Given the transformer described in Theorem 4.1, when $N \geq 36\alpha^2 L^2 \log 1/\epsilon$, $M \geq \max\{(T')^4, \log^2 1/\epsilon\}$, and $t \geq \max\{\sqrt[4]{M}, T'\}$, with probability at least $1-\epsilon$, the output of the transformer after t CoT steps satisfies*

$$\|\widehat{\mathbf{M}}^{(t)} - \mathbf{M}\|_F^2 \leq C\frac{\log(1/\epsilon)}{N\sqrt[4]{M}},$$

where C, α, L, T' are positive constants.

Corollary 4.1 (Label Prediction Error Bound)**.** *Let $\widehat{\mathbf{y}}_j$ be the predicted label for $\mathbf{x}_j$ according to Equation* (3.5)*. Let $\mathcal{R}^*$ be the prediction error under the Bayes-optimal classifier with known class mean vectors $\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_C$. Then, under the same conditions as described in Theorem 4.2, we have*

$$\mathbb{P}[\widehat{\mathbf{y}}_j \neq \mathbf{y} | \boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_C] - \mathcal{R}^* \leq \mathcal{O}\Big(\frac{1}{\sqrt{N\mathrm{poly}(M)}}\Big).$$

Proof sketch. The proof of Theorem 4.2 contains three major steps. In **Step 1**, we utilize the Hoeffding's inequality to ensure that with a sufficient number of labeled data N, the initial class mean estimates $\widehat{\boldsymbol{\mu}}_1^{(1)}, \cdots, \widehat{\boldsymbol{\mu}}_C^{(1)}$ are in a small neighborhood of the ground truth class means $\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_C$. In **Step 2**, we need to bound the gap between the gradient descent updating step for $t > 1$ in Equation (4.4), and one gradient descent step for the expected log-likelihood loss $\mathcal{L}(\{\widehat{\boldsymbol{\mu}}_i^{(t)}\}) = \mathbb{E}_{\mathbf{x}}\left[\log\left(\frac{1}{C}\sum_{i=1}^C \exp\left(-\frac{1}{2}\|\mathbf{x} - \widehat{\boldsymbol{\mu}}_i^{(t)}\|^2\right)\right)\right]$. To ensure that the gap is sufficiently small, we need to design the temperature parameter $\beta\tau$ so that the normalized weight is biased heavily toward the class mean estimation obtained from the current CoT step, and the influence of previous CoT steps is minimized. Then, utilizing Bernstein's inequality, this gap is bounded. In **Step 3**, we utilize Lipschitz continuity of $\mathcal{L}(\{\boldsymbol{\mu}_i\})$, combing the bound on the gradient gap in Step 2, to show that $\|\widehat{\mathbf{M}}^{(t)} - \mathbf{M}\|_F^2 \leq \mathcal{O}(1/\sqrt{N\mathrm{poly}(M)})$ for t large enough if $\widehat{\mathbf{M}}^{(1)}$ is in a small neighborhood of $\mathbf{M}$, which is guaranteed in Step 1. The complete proof can be found in Appendix C. □

Based on the smoothness of the Bayes risk, we have the following corollary as a direct consequence of Theorem 4.2.

Remark 1. *The advantage of utilizing unlabeled data in the augmented ICL becomes evident when comparing Corollary 4.1 with the existing lower bound on the excess risk for classical binary classification. It has been shown that the excess risk for any classifier trained on N labeled data scales in $\Omega(1/\sqrt{N})$ in the worst case of $\mathbf{M}$ (Li et al., 2017), which is in stark contrast to the upper bound $\mathcal{O}\Big(1/\sqrt{N\,\mathrm{poly}(M)}\Big)$ in Corollary 4.1. This result indicates that the designed transformer can effectively utilize the unlabeled data through CoT prompting, and strictly improves the prediction accuracy of any classifier that utilizes the labeled data only.*

5 Training Dynamics with Teacher Forcing

While Section 4 indicates that there exists a transformer that is able to implement an EM-type algorithm to utilize unlabeled data and improve the ICL performance through CoT prompting, in this section, we show that such a transformer can be obtained through teacher forcing training (Kim and Suzuki, 2025; Huang et al., 2025b).

The training objective of teacher forcing is to ensure that the transformer can mimic the trajectory of iterative updating under an EM algorithm during the CoT inference. Mathematically, it requires that the distance between the estimated mean vectors $\{\widehat{\boldsymbol{\mu}}_1^{(t)} \cdots \widehat{\boldsymbol{\mu}}_C^{(t)}\}_{t=1}^T$ during CoT inference and those generated by a reference algorithm f_{ref} is small along the trajectory. Specifically, given $\mathbf{X}_\ell$, $\mathbf{Y}_\ell$ and $\mathbf{X}_u$, we denote the generated reference trajectory as $f_{\text{ref}}(\mathbf{X}_\ell, \mathbf{Y}_\ell, \mathbf{X}_u) = \{\widehat{\boldsymbol{\mu}}_{\text{ref},1}^{(t)} \cdots \widehat{\boldsymbol{\mu}}_{\text{ref},C}^{(t)}\}_{t=1}^T$. Then, we construct the reference embedding sequence at the t-th CoT step as

$$\mathbf{H}_{\text{ref}}^{(t)} = \begin{bmatrix} \mathbf{X}_\ell & \mathbf{X}_u & \mathbf{0} & \star & \cdots & \star \\ \mathbf{Y}_\ell & \mathbf{0} & \mathbf{0} & \star & \cdots & \star \\ \mathbf{P}_\ell & \mathbf{P}_u & \mathbf{Q}^{(0)} & \mathbf{Q}_{\text{ref}}^{(1)} & \cdots & \mathbf{Q}_{\text{ref}}^{(t)} \end{bmatrix}, \quad \mathbf{Q}_{\text{ref}}^{\tau} = \begin{bmatrix} \mathbf{e}_1 & \cdots & \mathbf{e}_C \\ \boldsymbol{\mu}_{\text{ref},1}^{(\tau)} & \cdots & \boldsymbol{\mu}_{\text{ref},C}^{(\tau)} \\ * & \cdots & * \end{bmatrix}, \forall \tau \in [t].$$

We note that the reference embedding shares the same structure as the embedding defined in Equation (3.3), except that now the mean estimates are generated by the reference algorithm instead of the transformer itself. We then feed $\mathbf{H}_{\text{ref}}^{(t)}$ to the transformer, and extract the updated mean estimates from its output $\text{TF}_{\boldsymbol{\Theta}}(\mathbf{H}_{\text{ref}}^{(t)})$.

The corresponding CoT training loss can be defined as:

$$\widehat{\mathcal{L}}_{\text{CoT-train}}(\boldsymbol{\Theta}; \mathcal{I}_{\mathbf{M}}) = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{C}\left\|[\text{TF}_{\boldsymbol{\Theta}}(\mathbf{H}_{\text{ref}}^{(t-1)})]_{d+2c+1:2d+2c, M+N+i+C(t-1)} - \boldsymbol{\mu}_{i,\text{ref}}^{(t)}\right\|^2. \quad (5.1)$$

Similar to Ahn et al. (2024); Huang et al. (2025a), in this work, we analyze the training convergence of the population loss defined as:

$$\mathcal{L}_{\text{CoT-train}}(\boldsymbol{\Theta}) = \mathbb{E}_{\mathcal{I}_{\mathbf{M}}}\left[\widehat{\mathcal{L}}_{\text{CoT-train}}(\boldsymbol{\Theta}; \mathcal{I}_{\mathbf{M}})\right], \quad (5.2)$$

where the expectation is taken over the randomness in the generation process of $\mathcal{I}_{\mathbf{M}}$.

Directly analyzing the training dynamics of all layers of the transformer is intractable. On the other hand, as we mentioned in Section 4, the last three layers of the transformer can be constructed explicitly beforehand, as their parameters are data-independent. As a result, in the following, we will freeze these three layers and train the first layer only.

Assumption 1 (Initialization)**.** *We initialize the first layer of the three-layer transformer described in Theorem 4.1 as follows:*

$$\mathbf{Q}^{(0)}\mathbf{K}^{(0)} = \begin{bmatrix} \mathbf{0}_{d\times(d+2C)} & \mathbf{W}^{(0)} & & & \\ & & \mathbf{0}_{(4C+d+2)\times 2C} & & \\ & & & 1 & \mathbf{0}_{1\times 2} & \beta^{(0)} \\ & & & & & 0 \end{bmatrix},$$

$$\mathbf{V}^{(0)} = \text{diag}\big(\mathbf{0}_{(d+2C)\times(d+C)}, \mathbf{I}_C, \mathbf{0}_{(d+C+4)\times(d+2C+4)}\big),$$

where $\mathbf{W}^{(0)}$ is a $d \times d$ matrix whose entries are randomly sampled from a standard Gaussian distribution, $\beta^{(0)}$ is a constant, and all the unspecified entries are equal to zero.

Theorem 5.1 (Training Convergence)**.** *Let $\{\mathbf{Q}^{(k)}, \mathbf{K}^{(k)}, \mathbf{V}^{(k)}\}_{k\geq 0}$ be the parameters of the first attention layer of the transformer after applying k iterations of gradient descent on the population loss defined in Equation* (5.2)*. Then, with the initialization specified in Assumption 1, we have*

$$\|\mathbf{W}^{(k)} - \boldsymbol{\Sigma}^{-1}\|_F^2 \leq c^k \|\mathbf{W}^{(0)} - \boldsymbol{\Sigma}^{-1}\|_F^2$$

for some positive constant c, while the other parameters in $\mathbf{Q}^{(0)}$, $\mathbf{K}^{(0)}$ and $\mathbf{V}^{(0)}$ remain unchanged.

Theorem 5.1 indicates that under teacher forcing training, the parameter matrix $\mathbf{W}^{(k)}$ of the first layer converges to the *precision* matrix $\boldsymbol{\Sigma}^{-1}$, the inverse of the noise covariance matrix linearly. Combining with other parameters in $\mathbf{Q}^{(0)}$, $\mathbf{K}^{(0)}$ and $\mathbf{V}^{(0)}$, we observe that the teacher forcing training recovers the transformer described in Theorem 4.1, theoretically validating the transformer's expressiveness for augmented ICL.

Proof sketch. We use the superscript (k, t) to denote the t-th CoT step in the k-th gradient descent iteration. First, we drop the temperature term $\beta\tau$ in the definition of $p_{ij}^{(k,t)}$ given in Equation (4.2) and approximate it as in Equation (4.3), noting the approximation error can be made arbitrarily small by taking β sufficiently large. Next, we define

$$q_{ij}^{(k,t)} = \frac{\exp\big(-\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_i^{(k,t)} - \mathbf{x}_j\|_{\mathbf{W}^{(k)}}^2\big)}{\sum_{h=1}^{C}\exp\big(-\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_h^{(k,t)} - \mathbf{x}_j\|_{\mathbf{W}^{(k)}}^2\big)},$$

which corresponds to replacing $\boldsymbol{\Sigma}^{-1}$ by $\mathbf{W}^{(k)}$ in the approximation of $p_{ij}^{(k,t)}$.

To prove one-step improvement of gradient descent on the population loss under teacher forcing, we must exhibit a constant $\alpha > 0$ such that $-\big\langle \mathbf{W}^{(k)} - \boldsymbol{\Sigma}^{-1}, \eta^{(k)}\nabla_{\mathbf{W}}\mathcal{L}_{\text{CoT-train}}\big\rangle \leq$

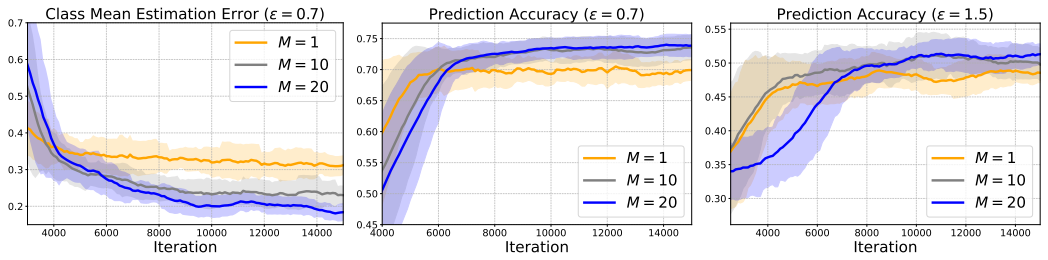

Figure 1: Inference performance of the transformer trained via teacher forcing versus number of gradient descent iterations during training. Number of classes $C = 3$, number of labeled examples $N = 5$, CoT steps $T = 5$. The solid line shows the average results across 5 runs, and the shaded region represents ± 2 standard deviations.

$-\alpha\|\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1}\|_F^2$. Our proof proceeds in three major steps. **Step 1.** Since direct analysis of $\nabla_{\mathbf{W}}\mathcal{L}$ is intractable, we propose a novel decomposition by applying Stein's lemma to break the gradient into two analytically tractable terms: one is the posterior-difference term involving $\mathbb{E}[\mathbf{p}_j^{(k,t)} - \mathbf{q}_j^{(k,t)}]$ and the other is the Jacobian-difference term involving $\mathbb{E}[\nabla\mathbf{p}_j^{(k,t)} - \nabla\mathbf{q}_j^{(k,t)}]$. **Step 2.** We show that an isotropic initialization of $\mathbf{W}^{(k)}$ remains isotropic under gradient descent. The preservation of isotropy enforces alignment between $\mathbf{p}_j$ and $\mathbf{q}_j$ in expectation, i.e., $\mathbb{E}[\mathbf{p}_j^{(k,t)}] = \mathbb{E}[\mathbf{q}_j^{(k,t)}]$. Therefore, the posterior-difference term vanishes. **Step 3.** We analyze $\nabla\mathbf{p}_j^{(k,t)}$ and $\nabla\mathbf{q}_j^{(k,t)}$ based on the their inherent symmetric structure. This analysis shows the Jacobian difference term degenerates to the following symmetric matrix under expectation: $\left(\mathrm{diag}(1/d) - \frac{1}{d^2}\mathbf{1}\mathbf{1}^\top\right)\mathbf{M}^\top\left(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1}\right)$, which enables us to avoid complicated analysis directly on the Jacobian difference term. Combining Steps 2 and 3, we obtain the following upper bound for the inner product term $-\langle\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1}, \nabla\mathcal{L}\rangle \le -\alpha'\|\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1}\|_F^2$, which provides the desired result. The detailed proof can be found in Appendix D. □

6 Experimental Results

Compute resources. All experiments are conducted on an NVIDIA H100 GPU with 80 GB of memory. The experiments require roughly five hours to complete.

Problem setup. In the following experiments, the augmented ICL instances are generated as follows. We set the number of classes $C = 3$ and the data dimension $d = 3$. The class mean vectors $\{\boldsymbol{\mu}_i\}_{i=1}^C$ are randomly sampled from a d-dimensional standard normal distribution. The covariance matrix $\Sigma = \epsilon\mathbf{I}_d$ is shared across classes, where $\mathbf{I}_d$ is the d-dimensional identity matrix. We set $\epsilon \in \{0.7, 1.5\}$. Each instance contains $N = 5$ labeled data points and M unlabeled data points, where $M \in \{1, 10, 20\}$. The $M = 1$ case recovers the conventional ICL setting.

Transformer structure. We construct a transformer with the architecture specified in Theorem 4.1. This model features 4 layers, with each layer composed of an attention module followed by an MLP module. Activation functions for the attention layers were configured as follows: softmax for the first layer, linear for the second and third layers, and ReLU for the fourth layer. We set $d_p = 16$, and the number of CoT steps $T = 5$. During training, in each iteration, we randomly generate 64 augmented ICL instances, and perform one gradient descent (GD) on the average empirical CoT training loss defined in Equation (5.1) over the batch. In total, we perform $15,000$ GD iterations during training.

Results. We evaluate the performance of the trained transformer after every 100 GD iterations. For evaluation, we randomly generated 100 augmented ICL instances, and obtained the corresponding class mean estimates from the trained transformer through CoT prompting. We then utilize these estimated class means to obtain the label prediction results according to Equation (3.5). For each $M \in \{1, 10, 20\}$, we conduct 5 runs. We track the the class mean estimation error and prediction accuracy, and plot the average performance and standard deviation across these 5 runs in Figure 1.

From Figure 1, we observe that augmented ICL outperforms conventional ICL significantly after a sufficient number of training iterations. As M increases, the advantage becomes more prominent: the transformer's class mean estimation error decrease and the classification accuracy increase, as predicted by our theoretical results Theorem 4.2 and Corollary 4.1.

We notice that the advantage of augmented ICL is more significant when ϵ is relatively small. This is because when ϵ is small, the data distribution is less noisy, meaning that the features carry more information relevant to the labels. Therefore, the unlabeled data provides clearer structure that the transformer can leverage through augmented ICL to estimate class means more accurately.

7 Conclusion

In this work, we introduced augmented ICL, a framework in which models process a mixture of labeled and unlabeled examples within the prompt. We provided theoretical insights showing that transformers equipped with CoT reasoning can implement an EM-style algorithm for augmented ICL in a multi-class linear classification task, with provably decreasing prediction error as the amount of unlabeled data increases. Moreover, we showed that such transformer behavior can emerge through standard teacher forcing training. Our empirical results support the theory.

Acknowledgments

The authors thank Li Fan, Wei Shen and Cong Shen for their helpful discussions during the preparation of this work. RL and JY were partially supported by the U.S. NSF under grants 2318759, 2531023 and 2531789.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Agarwal, R., Singh, A., Zhang, L., Bohnet, B., Rosias, L., Chan, S., Zhang, B., Anand, A., Abbas, Z., Nova, A., et al. (2024). Many-shot in-context learning. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, pages 76930–76966.

Ahn, K., Cheng, X., Daneshmand, H., and Sra, S. (2024). Transformers learn to implement preconditioned gradient descent for in-context learning. *Advances in Neural Information Processing Systems*, 36.

Akyürek, E., Schuurmans, D., Andreas, J., Ma, T., and Zhou, D. (2023). What learning algorithm is in-context learning? investigations with linear models. In *The Eleventh International Conference on Learning Representations*.

Bai, Y., Chen, F., Wang, H., Xiong, C., and Mei, S. (2024). Transformers as statisticians: Provable in-context learning with in-context algorithm selection. *Advances in neural information processing systems*, 36.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., and Mordatch, I. (2021). Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097.

Chen, S. and Li, Y. (2025). Provably learning a multi-head attention layer. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 1744–1754.

Chen, S., Sheen, H., Wang, T., and Yang, Z. (2024). Training dynamics of multi-head softmax attention for in-context learning: Emergence, convergence, and optimality. *CoRR*.

Chen, W.-L., Wu, C.-K., Chen, Y.-N., and Chen, H.-H. (2023). Self-icl: Zero-shot in-context learning with self-generated demonstrations. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Chen, Z., Wang, S., Tan, Z., Li, J., and Shen, C. (2025a). Maple: Many-shot adaptive pseudo-labeling for in-context learning. In *Forty-second International Conference on Machine Learning*.

Chen, Z., Wu, R., and Fang, G. (2025b). Transformers as unsupervised learning algorithms: A study on gaussian mixtures. *arXiv preprint arXiv:2505.11918*.

Cheng, X., Chen, Y., and Sra, S. (2024). Transformers implement functional gradient descent to learn non-linear functions in context. In *Proceedings of the 41st International Conference on Machine Learning*, pages 8002–8037.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. (2023). Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., et al. (2024). Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Cui, Y., Ren, J., He, P., Tang, J., and Xing, Y. (2024). Superiority of multi-head attention in in-context linear regression. *CoRR*.

Deora, P., Ghaderi, R., Taheri, H., and Thrampoulidis, C. (2023). On the optimization and generalization of multi-head attention. *Transactions on Machine Learning Research*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Diakonikolas, I., Kane, D. M., and Stewart, A. (2018). List-decodable robust mean estimation and learning mixtures of spherical gaussians. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1047–1060.

Dosovitskiy, A. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Fu, D., Chen, T., Jia, R., and Sharan, V. (2023). Transformers learn higher-order optimization methods for in-context learning: A study with linear models. In *NeurIPS 2023 Workshop on Mathematics of Modern Machine Learning*.

Garg, S., Tsipras, D., Liang, P. S., and Valiant, G. (2022). What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598.

Gupta, S., Jegelka, S., Lopez-Paz, D., and Ahuja, K. (2024). Context is environment. In *The Twelfth International Conference on Learning Representations*.

He, Y., Chen, H.-Y., Cao, Y., Fan, J., and Liu, H. (2025). Transformers versus the em algorithm in multi-class clustering. *arXiv preprint arXiv:2502.06007*.

Huang, J., Wang, Z., and Lee, J. D. (2025a). Transformers learn to implement multi-step gradient descent with chain of thought. In *The Thirteenth International Conference on Learning Representations*.

Huang, R., Liang, Y., and Yang, J. (2024). Non-asymptotic convergence of training transformers for next-token prediction. *Advances in Neural Information Processing Systems*, 37:80634–80673.

Huang, R., Liang, Y., and Yang, J. (2025b). How transformers learn regular language recognition: A theoretical study on training dynamics and implicit bias. In *Forty-second International Conference on Machine Learning*.

Huang, Y., Cheng, Y., and Liang, Y. (2023). In-context convergence of transformers. *arXiv preprint arXiv:2310.05249*.

Kim, J. and Suzuki, T. (2024). Transformers learn nonlinear features in context: nonconvex mean-field dynamics on the attention landscape. In *Proceedings of the 41st International Conference on Machine Learning*, pages 24527–24561.

Kim, J. and Suzuki, T. (2025). Transformers provably solve parity efficiently with chain of thought. In *The Thirteenth International Conference on Learning Representations*.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Li, H., Wang, M., Lu, S., Cui, X., and Chen, P.-Y. (2024a). Training nonlinear transformers for efficient in-context learning: A theoretical learning and generalization analysis. *arXiv preprint arXiv:2402.15607*.

Li, H., Weng, M., Liu, S., and Chen, P.-Y. (2023). A theoretical understanding of shallow vision transformers: Learning, generalization, and sample complexity. In *International Conference on Learning Representations*.

Li, T., Yi, X., Carmanis, C., and Ravikumar, P. (2017). Minimax gaussian classification & clustering. In *Artificial Intelligence and Statistics*, pages 1–9. PMLR.

Li, Y., Chang, X., Kara, M., Liu, X., Roy-Chowdhury, A., and Oymak, S. (2025). When and how unlabeled data provably improve in-context learning. *arXiv preprint arXiv:2506.15329*.

Li, Y., Huang, Y., Ildiz, M. E., Rawat, A. S., and Oymak, S. (2024b). Mechanics of next token prediction with self-attention. In *International Conference on Artificial Intelligence and Statistics*, pages 685–693. PMLR.

Liu, R., Zhou, R., Shen, C., and Yang, J. (2025). On the learn-to-optimize capabilities of transformers in in-context sparse recovery. In *The Thirteenth International Conference on Learning Representations*.

Mahankali, A. V., Hashimoto, T., and Ma, T. (2023). One step of gradient descent is provably the optimal in-context learner with one layer of linear self-attention. In *The Twelfth International Conference on Learning Representations*.

Nichani, E., Damian, A., and Lee, J. D. (2024). How transformers learn causal structure with gradient descent. In *Proceedings of the 41st International Conference on Machine Learning*, pages 38018–38070.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022). Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Radford, A. (2018). Improving language understanding by generative pre-training.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Raventós, A., Paul, M., Chen, F., and Ganguli, S. (2023). Pretraining task diversity and the emergence of non-bayesian in-context learning for regression. *Advances in neural information processing systems*, 36:14228–14246.

Shen, W., Zhou, R., Yang, J., and Shen, C. (2025). On the training convergence of transformers for in-context classification of gaussian mixtures. In *Forty-second International Conference on Machine Learning*.

Sula, E. and Zheng, L. (2022). On the semi-supervised expectation maximization. *arXiv preprint arXiv:2211.00537*.

Sun, Z., Shen, Y., Zhou, Q., Zhang, H., Chen, Z., Cox, D., Yang, Y., and Gan, C. (2023). Principle-driven self-alignment of language models from scratch with minimal human supervision. *Advances in Neural Information Processing Systems*, 36:2511–2565.

Tarzanagh, D. A., Li, Y., Thrampoulidis, C., and Oymak, S. (2023a). Transformers as support vector machines. In *NeurIPS 2023 Workshop on Mathematics of Modern Machine Learning*.

Tarzanagh, D. A., Li, Y., Zhang, X., and Oymak, S. (2023b). Max-margin token selection in attention mechanism. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Thoppilan, R., De Freitas, D., Hall, J., Shazeer, N., Kulshreshtha, A., Cheng, H.-T., Jin, A., Bos, T., Baker, L., Du, Y., et al. (2022). Lamda: Language models for dialog applications. *CoRR*.

Tian, Y., Wang, Y., Chen, B., and Du, S. S. (2023). Scan and snap: Understanding training dynamics and token composition in 1-layer transformer. *Advances in Neural Information Processing Systems*, 36:71911–71947.

Tian, Y., Wang, Y., Zhang, Z., Chen, B., and Du, S. S. (2024). Joma: Demystifying multilayer transformers via joint dynamics of mlp and attention. In *The Twelfth International Conference on Learning Representations*.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Vasudeva, B., Deora, P., and Thrampoulidis, C. (2024). Implicit bias and fast convergence rates for self-attention. *Transactions on Machine Learning Research*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Von Oswald, J., Niklasson, E., Randazzo, E., Sacramento, J., Mordvintsev, A., Zhmoginov, A., and Vladymyrov, M. (2023a). Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pages 35151–35174. PMLR.

Von Oswald, J., Niklasson, E., Schlegel, M., Kobayashi, S., Zucchet, N., Scherrer, N., Miller, N., Sandler, M., Vladymyrov, M., Pascanu, R., et al. (2023b). Uncovering mesa-optimization algorithms in transformers. *arXiv preprint arXiv:2309.05858*.

Wan, X., Sun, R., Nakhost, H., Dai, H., Eisenschlos, J. M., Arik, S. O., and Pfister, T. (2023). Universal self-adaptive prompting. *arXiv preprint arXiv:2305.14926*.

Wang, Y., Kordi, Y., Mishra, S., Liu, A., Smith, N. A., Khashabi, D., and Hajishirzi, H. (2023). Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Wen, K., Zhang, H., Lin, H., and Zhang, J. (2025). From sparse dependence to sparse attention: Unveiling how chain-of-thought enhances transformer sample efficiency. In *The Thirteenth International Conference on Learning Representations*.

Xie, S. M., Raghunathan, A., Liang, P., and Ma, T. (2022). An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*.

Yang, J., Ma, S., and Wei, F. (2023). Auto-icl: In-context learning without human supervision. *arXiv preprint arXiv:2311.09263*.

Zhang, R., Frei, S., and Bartlett, P. L. (2024a). Trained transformers learn linear models in-context. *Journal of machine learning research*, 25(49).

Zhang, R., Wu, J., and Bartlett, P. (2024b). In-context learning of a linear transformer block: Benefits of the mlp component and one-step gd initialization. *Advances in Neural Information Processing Systems*, 37:18310–18361.

Zhao, R., Li, Y., and Sun, Y. (2020). Statistical convergence of the em algorithm on gaussian mixture models. *Electronic Journal of Statistics*, 14:632–660.

Zhou, C., Liu, P., Xu, P., Iyer, S., Sun, J., Mao, Y., Ma, X., Efrat, A., Yu, P., Yu, L., et al. (2023). Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We mentioned our main contribution and the scope in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitation in the Appendix.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We give the assumption and proof sketch for our theorems in the main paper, and give complete proof in the Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide all experiment details in the Experimental Results section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide the code in the supplemental material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We specify all the training and test details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We provide 2-sigma confidence interval in Section 6.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Provided in the Experimental Results section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have read and fully understood the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The broader impacts of this work are discussed in Appendix A. Due to the theoretical nature of this work, we do not foresee major negative impacts.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All sources are cited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No asset.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Does not involve any human subject.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Does not involve any human subject.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: Core method does not involve LLMs.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

Supplementary Materials

Contents

A Broader Impacts

This work provides theoretical insights into how transformers can leverage unlabeled data to improve in-context learning, a core capability underlying many recent advances in language models. By improving data efficiency and adaptability, our findings could enable more accessible and capable AI systems, particularly in low-resource settings where labeled data is limited. These advances may benefit a range of applications, including next-generation wireless communications and networking, healthcare, and financial services. Given the theoretical nature of this work, we anticipate minimal direct negative societal impact. Nonetheless, we recognize that future practical implementations inspired by this research should adhere to responsible AI principles.

B Limitations and Future Directions

Our analysis and experiments possess certain limitations. Below, we outline these limitations and propose directions for future research.

First, our analysis tracks parameter updates only in the *first* transformer layer, leaving all other layers frozen. As a result, it remains unclear how weights in non-linear hidden layers evolve under teacher forcing. To the best of our knowledge, the training dynamics of multi-layer transformer with *non-linear* activation is still lacking investigation. A full, multi-layer treatment for end-to-end training remains an open problem.

Second, this paper is the first theoretical investigation of the influence of unlabeled data in in-context learning, therefore, we restricted the experiments to a synthetic data set. However, whether the same behavior emerges in real-world tasks, and how unlabeled examples influence in-context learning for large, fully-trained transformers, is still unknown. Empirically understanding such impact is a promising future direction.

C Proof of Expressiveness

First, we restate the theorem below.

Theorem C.1. *There exists a 4-layer transformer, such that its output sequence at the $(t+1)$-th CoT step satisfies*

$$\widehat{\boldsymbol{\mu}}_i^{(t+1)} = \widehat{\boldsymbol{\mu}}_i^{(t)} - \frac{\eta^{(t)}}{M} \sum_{j=N+1}^{N+M} p_{ij}^{(t)} \left(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j\right) + \mathbf{1}_{\{t=0\}} \cdot \frac{C}{N} \sum_{j=1}^{N} (\mathbf{e}_i^\top \mathbf{y}_j)\mathbf{x}_j, \tag{C.1}$$

for any $i \in [C]$, where $\eta^{(t)} = \alpha/(T'+t)$ for some positive constants α and T', $p_{ij}^{(t)}$ is the normalized weight

$$p_{ij}^{(t)} = \frac{\sum_{\tau=0}^{t} \exp\left(-\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)} - \mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2 + \beta\tau\right)}{\sum_{\tau=0}^{t}\sum_{c=1}^{C} \exp\left(-\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_c^{(\tau)} - \mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2 + \beta\tau\right)}, \tag{C.2}$$

and β is a positive constant.

We start from the proof of Theorem 4.1, which shows the transformer's capability of implementing an EM-style algorithm.

C.1 Proof of Theorem 4.1

Recall that the input sequence at the t-th CoT step is formulated as

$$\widehat{\mathbf{H}}^{(t-1)} = \begin{bmatrix} \mathbf{X}_\ell & \mathbf{X}_u & \mathbf{0} & \star & \cdots & \star \\ \mathbf{Y}_\ell & \mathbf{0} & \mathbf{0} & \star & \cdots & \star \\ \mathbf{P}_\ell & \mathbf{P}_u & \mathbf{Q}^{(0)} & \mathbf{Q}^{(1)} & \cdots & \mathbf{Q}^{(t-1)} \end{bmatrix},$$

where

$$\mathbf{P}_\ell = [\mathbf{p}_1, \quad \mathbf{p}_2, \quad \cdots \quad \mathbf{p}_N], \tag{C.3}$$
$$\mathbf{P}_u = [\mathbf{p}_{N+1}, \quad \mathbf{p}_{N+2}, \quad \cdots \quad \mathbf{p}_{N+M}], \tag{C.4}$$
$$\mathbf{Q}^{(\tau)} = [\mathbf{q}_1^{(\tau)}, \quad \mathbf{q}_2^{(\tau)}, \quad \cdots \quad \mathbf{q}_C^{(\tau)}], \quad \tau \in [0:t-1]. \tag{C.5}$$

We specify $\mathbf{p}_j$ and $\mathbf{q}_i^{(\tau)}$ as follows. For each data sample $j \in [N+M]$, we denote

$$\mathbf{p}_j = \begin{bmatrix} \mathbf{0}_C \\ \mathbf{0}_d \\ \mathbf{0}_C \\ \mathbf{0}_C \\ 0 \\ \mathbf{1}_{j\in[N]} \\ \mathbf{1}_{j\in[N+1:N+M]} \\ 0 \end{bmatrix}, \quad \mathbf{q}_i^{(\tau)} = \begin{bmatrix} \mathbf{e}_i \\ \widehat{\boldsymbol{\mu}}_i^{(\tau)} \\ \mathbf{0}_C \\ \mathbf{0}_C \\ u_i^{(\tau)} \\ 0 \\ 0 \\ \tau \end{bmatrix},$$

where $\widehat{\boldsymbol{\mu}}_i^{(\tau)}$ stores the estimate of the mean vector of class i from the τ-th CoT step, and u_i^τ stores a rescaled norm of $\widehat{\boldsymbol{\mu}}_i^{(\tau)}$, i.e., $u_i^{(\tau)} = -\frac{\sigma}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}\|^2$.

Next, we specify the parameters of each layer of the transformer as follows.

Layer 1: The first layer of the transformer consists of an attention layer with a softmax activation function, and an MLP layer. Let the parameters of the attention layer satisfy

$$\mathbf{Q}_1\mathbf{K}_1 = \begin{bmatrix} \mathbf{0}_{d\times(d+2C)} & \boldsymbol{\Sigma}^{-1} & & \\ & & \mathbf{0}_{(4C+d+2)\times 2C} & \\ & & & \begin{matrix} 1 & \mathbf{0}_{1\times 2} & \beta \\ & & 0 \end{matrix} \end{bmatrix},$$
$$\mathbf{V}_1 = \begin{bmatrix} \mathbf{0}_{(d+2C)\times(d+C)} & & \\ & \mathbf{I}_C & \\ & & \mathbf{0}_{(d+C+4)\times(d+2C+4)} \end{bmatrix}.$$

Denote $\mathrm{attn}_1(\mathbf{p}_j)$ as the output token after passing $\mathbf{p}_j$ through the first attention layer, and let $\boldsymbol{\gamma}_i := \mathrm{attn}_1(\mathbf{p}_j)[d+C+1:d+2C]$. Then, we have

$$\boldsymbol{\gamma}_j = \frac{\sum_{\tau\in[0:t-1]}\sum_{i\in[C]}\exp\Big(-\frac{\sigma}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}\|^2_{\boldsymbol{\Sigma}^{-1}} + (\widehat{\boldsymbol{\mu}}_i^{(\tau)})^\top\mathbf{x}_j + \beta\tau\Big)\mathbf{e}_i}{\sum_{\tau\in[0:t-1]}\sum_{i\in[C]}\exp\Big(-\frac{\sigma}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}\|^2_{\boldsymbol{\Sigma}^{-1}} + (\widehat{\boldsymbol{\mu}}_i^{(\tau)})^\top\mathbf{x}_j + \beta\tau\Big)}.$$

Other entries in $\widehat{\mathbf{H}}^{(t-1)}$ remain unchanged after this attention layer.

Subsequent to the first attention layer, a token-wise MLP is applied. Similar to Kim and Suzuki (2025), in this work, we assume the MLP layer can realize any deterministic token-wise link function with negligible error. The first MLP layer transforms input representations $\mathbf{p}$ such that

$$\mathrm{mlp}_1(\mathrm{attn}_1(\mathbf{p}_j)) = \boldsymbol{\gamma}_j \cdot \mathrm{attn}_1(\mathbf{p}_j)[3C+d+3]$$
$$\mathrm{mlp}_1(u_i^{(\tau)}) = -\frac{\sigma}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}\|^2.$$

Since $\mathbf{p}_j[3C+d+3] = 0$ for $j \in [N]$ and $\mathbf{p}_j[3C+d+3] = 1$ for $j \in [N+1:N+M]$, and the corresponding entries remain unchanged after passing through the first attention layer, this MLP layer only keeps $\boldsymbol{\gamma}_j$ for tokens corresponding to the unlabeled dataset (i.e., $j \in [N]$), and set $\boldsymbol{\gamma}_j$ to zero for all other tokens (i.e., $j \in [N+1:M]$).

Layer 2: The second layer of the transformer consists of an attention layer with a linear activation function, and an MLP layer. The parameters of the attention layer are set to satisfy

$$\mathbf{Q}_2\mathbf{K}_2 = \begin{bmatrix} \mathbf{0}_{(2d+4C+2)\times(2d+4C+2)} & & \\ & 0 & 0 \\ & \alpha_1 & 0 \end{bmatrix},$$
$$\mathbf{V}_2 = \begin{bmatrix} \mathbf{0}_{(2d+3C)\times(d+2C)} & \mathbf{I}_C & \\ & & \mathbf{0}_{4\times(d+C+4)} \end{bmatrix}.$$

We denote $\mathbf{s}_i^{(\tau)} := \mathrm{attn}_2(\mathbf{q}_i^{(\tau)})[d+2C+1 : d+3C]$ as the vector extracted from the output token after passing $\mathbf{q}_i^{(\tau)}$ through the second attention layer. Then, $\mathbf{s}_i^{(\tau)} = \tau\,\alpha_1 \sum_{j=N+1}^{N+M} \boldsymbol{\gamma}_j$, where α_1 is a fixed scalar embedded in $\mathbf{Q}_2\mathbf{K}_2$.

We let the subsequent MLP layer realize the following token-wise Lipschitz function:

$$\mathrm{mlp}_2(\widehat{\boldsymbol{\mu}}_i^{(\tau)}) = \widehat{\boldsymbol{\mu}}_i^{(\tau)} - \frac{1}{\tau(\tau+\alpha_2)}\widehat{\boldsymbol{\mu}}_i^{(\tau)}\mathbf{e}_i^\top \mathbf{s}_i^{(\tau)} = \widehat{\boldsymbol{\mu}}_i^{(\tau)} - \frac{\alpha_1}{\tau+\alpha_2}\widehat{\boldsymbol{\mu}}_i^{(\tau)}\mathbf{e}_i^\top \sum_{j\in[N+1]}^{N+M} \boldsymbol{\gamma}_j,$$
$$\mathrm{mlp}_2(\mathbf{e}_i) = \frac{\alpha_1}{\tau+\alpha_2}\mathbf{e}_i.$$

Layer 3: Similar to the second transformer layer, the third layer also consists of a linear attention layer and an MLP layer. Consider the following parameterization for the attention layer:

$$\mathbf{Q}_3\mathbf{K}_3 = \begin{bmatrix} \mathbf{0}_{(d+C)\times(d+2C)} & \mathbf{I}_C & \\ & & \mathbf{0}_{(d+2C+4)\times(d+C+4)} \end{bmatrix},$$
$$\mathbf{V}_3 = \begin{bmatrix} \mathbf{0}_{(d+3C)\times d} & & \\ \mathbf{I}_d & & \\ & & \mathbf{0}_{C+4;d+4C+4} \end{bmatrix}.$$

Therefore, this attention layer realizes the following updating process:

$$\mathrm{attn}_3(\widehat{\boldsymbol{\mu}}_i^{(\tau)}) = \mathrm{mlp}_2(\widehat{\boldsymbol{\mu}}_i^{(\tau)}) + \frac{\alpha_1}{\tau+\alpha_2}\sum_{j\in[M]} \mathbf{x}_j\mathbf{e}_i^\top\boldsymbol{\gamma}_j^{(\tau)}.$$

After this linear attention layer, we let the MLP layer realize the following function

$$\mathrm{mlp}_3(\mathbf{e}_i) = \frac{\tau+\alpha_2}{\alpha_1}\mathbf{e}_i$$

Layer 4: For the last layer, we introduce a transformer layer with a ReLU-activated attention layer followed by an MLP layer. We parameterize the attention layer as:

$$\mathbf{Q}_4\mathbf{K}_4 = \begin{bmatrix} \mathbf{0}_{(d+C)\times d} & & & \\ & \mathbf{I}_C & & \\ & & \mathbf{0}_{(d+2C+1)\times(d+3C+3)} & \\ & & & \begin{matrix}1\\ \mathbf{0}_2\end{matrix} \end{bmatrix},$$
$$\mathbf{V}_4 = \begin{bmatrix} \mathbf{0}_{(d+2C)\times d} & & \\ \mathbf{I}_d & & \\ & & \mathbf{0}_{(2C+4)\times(4C+d+4)} \end{bmatrix}.$$

The corresponding updating rule of this layer gives

$$\mathrm{attn}_4(\boldsymbol{\mu}_i^{(\tau)}) = \mathrm{attn}_3(\boldsymbol{\mu}_i^{(\tau)}) + \frac{C}{N}\sum_{j\in[N]} \mathbf{x}_j \mathrm{ReLU}(-\tau + \mathbf{e}_i^\top\mathbf{y}_j).$$

Therefore, we can further reformulate it as

$$\mathrm{attn}_4(\boldsymbol{\mu}_i^{(\tau)}) = \begin{cases} \frac{C}{N}\sum_{j\in[N]} \mathbf{x}_j \cdot \big(\mathbf{e}_i^\top \mathbf{y}_j\big), & \text{if } \tau = 0, \\ \mathrm{attn}_3(\boldsymbol{\mu}_\tau^{(\tau)}), & \text{if } \tau > 0. \end{cases}$$

Given the above 4-layer transformer structure, by setting $\alpha_1 = \alpha/M$ and $\alpha_2 = T'$ for fixed $\alpha > 0$, $T' > 0$, the output sequence corresponding to the $\mathbf{Q}^{(t-1)}$ block in the input sequence that satisfies:

$$\widehat{\boldsymbol{\mu}}_i^{(t+1)} = \widehat{\boldsymbol{\mu}}_i^{(t)} - \frac{\eta^{(t)}}{M}\sum_{j=N+1}^{N+M} p_{ij}^{(t)}\big(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j\big) + \mathbf{1}_{\{t=0\}}\cdot\frac{C}{N}\sum_{j=1}^{N}(\mathbf{e}_i^\top \mathbf{y}_j)\mathbf{x}_j, \tag{C.6}$$

for any $i \in [C]$, where $\eta^{(t)} = \alpha/(T'+t)$ for some positive constants α and T', $p_{ij}^{(t)}$ is the normalized weight

$$p_{ij}^{(t)} = \frac{\sum_{\tau=0}^{t}\exp\Big(-\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)} - \mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2 + \beta\tau\Big)}{\sum_{\tau=0}^{t}\sum_{c=1}^{C}\exp\Big(-\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_c^{(\tau)} - \mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2 + \beta\tau\Big)}.$$

The proof is thus complete.

C.2 Proof of Theorem 4.2

In this section, we show the detailed proof of Theorem 4.2. We start by restating the theorem.

Theorem C.2 (Class Mean Estimation Error.)**.** *Given the transformer described in Theorem 4.1, when $N \geq 36\alpha^2 L^2 \log 1/\epsilon$, $M \geq \max\{(T')^4, \log^2 1/\epsilon\}$, and $t \geq \max\{\sqrt[4]{M}, T'\}$, with probability at least $1-\epsilon$, the output of the transformer after t CoT steps satisfies*

$$\|\widehat{\mathbf{M}}^{(t)} - \mathbf{M}\|_F^2 \leq c\frac{\log(1/\epsilon)}{N\sqrt[4]{M}},$$

where c, α, L, T' are positive constants.

Step 1: First, we ensure that the initial estimation of the class mean vectors obtained from the *labeled* data gives a small estimation error.

Lemma 1 (Initial estimation error from labeled data.)**.** *Consider the initial class mean estimates*

$$\boldsymbol{\mu}_i^{(1)} = \frac{C}{N}\sum_{j\in[N]} \mathbf{x}_j \cdot \big(\mathbf{e}_i^\top \mathbf{y}_j\big), \quad \forall i \in [C].$$

Then, for fixed $K \geq 1$ and any positive constant $T' \geq 4K$, we have

$$\mathbb{P}\left[\big\|\boldsymbol{\mu}_i^{(1)} - \boldsymbol{\mu}_i\big\|^2 > \frac{K}{T'}\right] \leq \exp\big(-cNK/T'\big),$$

where c is a positive constant.

Proof. We denote n_i as the number of samples drawn from class i in the N labeled data. Under the assumption that $\mathbf{y}_j \sim \mathrm{Uniform}(\mathcal{Y})$, $\forall j \in [N]$, we have $n_i \sim \mathrm{Binomial}(N, 1/C)$. Then, according to Chernoff's inequality, for any $\epsilon \in (0,1)$, we have

$$\mathbb{P}\Big(\big|n_i - \tfrac{N}{C}\big| > \epsilon\tfrac{N}{C}\Big) \leq 2\exp\Big(-t\frac{\epsilon^2 N}{3C}\Big).$$

For any $u \geq 0$, Let $\epsilon = u\sqrt{K/T'}$, we obtain

$$\mathbb{P}\Big(\big|n_i - \tfrac{N}{C}\big| > u\sqrt{\tfrac{K}{T'}}\tfrac{N}{C}\Big) \leq 2\exp\Big(-\tfrac{u^2 NK}{3CT'}\Big).$$

Therefore,

$$\mathbb{P}\Big(\big|\tfrac{C}{N}n_i - 1\big| > u\sqrt{\tfrac{K}{T'}}\Big) \leq 2\exp\Big(-\tfrac{u^2 NK}{3CT'}\Big). \tag{C.7}$$

Conditional on n_i, we have $\frac{1}{n_i}\sum_{j:\mathbf{y}_i=\mathbf{e}_i}\mathbf{x}_j - \boldsymbol{\mu}_i \sim \mathcal{N}\big(0, \boldsymbol{\Sigma}/n_i\big)$. We assume $\boldsymbol{\Sigma}$ is an isotropic matrix in the form of $\sigma^2\mathbb{1}$. Then, $\|\boldsymbol{\Sigma}\|_2 = \sigma^2$, and we obtain the following inequality based on Hoeffding's inequality.

$$\mathbb{P}\Big(\Big\|\frac{1}{n_i}\sum_{j:\mathbf{y}_j=\mathbf{e}_i}\mathbf{x}_j - \boldsymbol{\mu}_i\Big\| > \sigma\sqrt{\frac{2t}{n_i}}\Big|n_i\Big) \le 2e^{-t}.$$

For any $v \ge 0$, by setting $t = v^2 n_i K/(2\sigma^2 T')$, we have

$$\begin{aligned}
&\mathbb{P}\Big(\Big\|\frac{1}{n_i}\sum_{j:\mathbf{y}_j=\mathbf{e}_i}\mathbf{x}_j - \boldsymbol{\mu}_i\Big\| > v\sqrt{\tfrac{K}{T'}}\Big) \le 2\exp(-v^2 n_i K/(8\sigma^2 T')) \\
&\le 2\exp(-v^2(1-\frac{K}{T'})\frac{NK}{C\sigma^2 T'}) \\
&\le 2\exp(-v^2\frac{NK}{2C\sigma^2 T'}).
\end{aligned} \tag{C.8}$$

Then,

$$\begin{aligned}
\mathbb{P}\Big(\|\widehat{\boldsymbol{\mu}}_i - \boldsymbol{\mu}_i\|^2 > \tfrac{K}{T'}\Big) &= \mathbb{P}\Big(\Big\|\tfrac{C}{N}\sum_{j:\mathbf{y}_j=\mathbf{e}_i}\mathbf{x}_j - \tfrac{Cn_i}{N}\boldsymbol{\mu}_i - (1-\tfrac{Cn_i}{N})\boldsymbol{\mu}_i\Big\| > \sqrt{\tfrac{K}{T'}}\Big) \\
&\le \mathbb{P}\Big(\tfrac{Cn_i}{N}\Big\|\frac{1}{n_i}\sum_{j:\mathbf{y}_j=\mathbf{e}_i}\mathbf{x}_j - \boldsymbol{\mu}_i\Big\| + |1-\tfrac{Cn_i}{N}|\|\boldsymbol{\mu}_i\| \ge \sqrt{\tfrac{K}{T'}}\Big) \\
&\le \mathbb{P}\Big(\tfrac{Cn_i}{N}\Big\|\frac{1}{n_i}\sum_{j:\mathbf{y}_j=\mathbf{e}_i}\mathbf{x}_j - \boldsymbol{\mu}_i\Big\| \ge \sqrt{\tfrac{K}{T'}}, \text{ or } |1-\tfrac{Cn_i}{N}|\|\boldsymbol{\mu}_i\| \ge \sqrt{\tfrac{K}{T'}}\Big) \\
&\overset{(a)}{\le} 4\exp(-c\tfrac{NK}{T'})
\end{aligned}$$

for positive constant c. The inequality (a) holds by setting $u = 1/\|\boldsymbol{\mu}_i\|$ in Equation (C.8) and setting $v = N/Cn_1$ in Equation (C.17). The proof is thus complete. □

Step 2: Next, we bound the discrepancy between the gradient obtained from each CoT step for a given input sequence, and the gradient of the population loss.

We define the population loss for any given set of class mean vectors $\{\boldsymbol{\mu}_i\}_{i\in[C]}$ (i.e., any given $\mathbf{M}$) as:

$$\mathcal{L}(\{\boldsymbol{\mu}_i\}) = \mathbb{E}_{\mathbf{x}}\left[\log\Big(\tfrac{1}{C}\sum_{i=1}^{C}\exp\big(-\tfrac{1}{2}\|\mathbf{x}-\boldsymbol{\mu}_i\|^2_{\boldsymbol{\Sigma}^{-1}}\big)\Big)\right], \tag{C.9}$$

where the expectation is taken over the randomly generated data $\mathbf{x}$ for given $\mathbf{M}$, as specified in Equation (3.1).

We first characterize an important property of $\mathcal{L}(\{\boldsymbol{\mu}_i\})$ as follows.

Lemma 2. *The Jacobian of $\nabla_{\boldsymbol{\mu}_i}\mathcal{L}$ at $\boldsymbol{\mu}_i$ for all $i \in [C]$ is negative definite, i.e., $\nabla^2_{\boldsymbol{\mu}_i}\mathcal{L} \prec \mathbf{0}$.*

Proof. Define

$$p_{\mathbf{x}}(\boldsymbol{\mu}_i) = \frac{\exp\Big(-\frac{1}{2}\|\mathbf{x}-\boldsymbol{\mu}_i\|^2_{\boldsymbol{\Sigma}^{-1}}\Big)}{\sum_{c=1}^{C}\exp\Big(-\frac{1}{2}\|\mathbf{x}-\boldsymbol{\mu}_c\|^2_{\boldsymbol{\Sigma}^{-1}}\Big)},$$

so that $p_{\mathbf{x}}(\boldsymbol{\mu}_i)$ is a softmax weight depending on $\mathbf{x}$ and the centers $\{\boldsymbol{\mu}_c\}_{c=1}^{C}$. Note that $\nabla^2_{\boldsymbol{\mu}_i}\mathcal{L}$ is the Hessian of $\nabla\mathcal{L}$ at $\boldsymbol{\mu}_i$, given by

$$\nabla^2_{\boldsymbol{\mu}_i}\mathcal{L} = \mathbb{E}_{\mathbf{x}}\Big[p_{\mathbf{x}}(\boldsymbol{\mu}_i)\big(1-p_{\mathbf{x}}(\boldsymbol{\mu}_i)\big)\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_i-\mathbf{x})(\boldsymbol{\mu}_i-\mathbf{x})^\top\boldsymbol{\Sigma}^{-1} - p_{\mathbf{x}}(\boldsymbol{\mu}_i)\boldsymbol{\Sigma}^{-1}\Big],$$

where and the expectation is taken with respect to the distribution of $\mathbf{x}$. Therefore, there exists a constant $0 \le \alpha < 1$ such that

$$\nabla^2_{\boldsymbol{\mu}_i}\mathcal{L} \preceq \mathbb{E}_{\mathbf{x}}\Big[\alpha\, p_{\mathbf{x}}(\boldsymbol{\mu}_i)\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_i-\mathbf{x})(\boldsymbol{\mu}_i-\mathbf{x})^\top\boldsymbol{\Sigma}^{-1} - p_{\mathbf{x}}(\boldsymbol{\mu}_i)\boldsymbol{\Sigma}^{-1}\Big].$$

Now, for any nonzero vector $\mathbf{u} \in \mathbb{R}^d$, consider the quadratic form $\mathbf{u}^\top \nabla^2_{\boldsymbol{\mu}_i} \mathcal{L} \mathbf{u}$, using the above matrix inequality, we have

$$\mathbf{u}^\top \nabla^2_{\boldsymbol{\mu}_i} \mathcal{L} \mathbf{u} \leq \mathbf{u}^\top \mathbb{E}_{\mathbf{x}} \Big[\alpha\, p_{\mathbf{x}}(\boldsymbol{\mu}_i) \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu}_i - \mathbf{x})(\boldsymbol{\mu}_i - \mathbf{x})^\top \boldsymbol{\Sigma}^{-1} - p_{\mathbf{x}}(\boldsymbol{\mu}_i) \boldsymbol{\Sigma}^{-1} \Big] \mathbf{u}.$$

Therefore, rewriting the expectation as an integral yields

$$\begin{aligned} \mathbf{u}^\top \nabla^2_{\boldsymbol{\mu}_i} \mathcal{L} \mathbf{u} &\leq \frac{1}{C} \int_{\mathbb{R}^d} \alpha\, \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_i, \mathbf{I}) \mathbf{u}^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu}_i - \mathbf{x})(\boldsymbol{\mu}_i - \mathbf{x})^\top \boldsymbol{\Sigma}^{-1} \mathbf{u} \, \mathrm{d}\mathbf{x} - \frac{1}{C} \mathbf{u}^\top \boldsymbol{\Sigma}^{-1} \mathbf{u} \\ &\leq \frac{\alpha}{C} \mathbf{u}^\top \boldsymbol{\Sigma}^{-1} \mathbb{E}_{\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}^{-1})} \Big[(\boldsymbol{\mu}_i - \mathbf{x})(\boldsymbol{\mu}_i - \mathbf{x})^\top \Big] \boldsymbol{\Sigma}^{-1} \mathbf{u} - \frac{1}{C} \mathbf{u}^\top \boldsymbol{\Sigma}^{-1} \mathbf{u} < 0. \end{aligned}$$

Thus, the quadratic form is negative for every nonzero $\mathbf{u}$, and the matrix $\nabla^2_{\boldsymbol{\mu}_i} \mathcal{L}$ is negative definite. This completes the proof. □

We note that for each CoT step $t > 0$, the updating induced by the constructed transformer is

$$\widehat{\boldsymbol{\mu}}_i^{(t+1)} = \widehat{\boldsymbol{\mu}}_i^{(t)} - \frac{\eta^{(t)}}{M} \sum_{j=N+1}^{N+M} p_{ij}^{(t)} \left(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j \right), \tag{C.10}$$

where $p_{ij}^{(t)}$ is defined in Equation (4.2).

To simplify notation, denote

$$\frac{1}{M} \sum_{j=N+1}^{N+M} p_{ij}^{(t)} \big(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j \big) := \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}}. \tag{C.11}$$

We note that $\widehat{\mathcal{L}}$ itself is not an explicit loss function. We use the notation $\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}}$ to represent the equivalent gradient for the updating determined by the t-th CoT step. Lemma 2 states $\nabla^2_{\boldsymbol{\mu}_i} \mathcal{L}$ is negative definite for each $\boldsymbol{\mu}_i$, in the following lemma, we show that $\nabla^2_{\boldsymbol{\mu}} \mathcal{L}$ is negative definite for the concatenate vector $\boldsymbol{\mu}$ when $\{\boldsymbol{\mu}_i\}_i^C$ are well seperated.

Lemma 3. *The Jacobian of $\nabla_{\boldsymbol{\mu}} \mathcal{L}$ at $\boldsymbol{\mu}$ is negative definite, i.e., $\nabla^2_{\boldsymbol{\mu}} \mathcal{L} \prec \mathbf{0}$.*

Proof. Recall that $\boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}$ is a symmetric positive definite. For $\mathbf{x} \in \mathbb{R}^d$ and $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_C) \in (\mathbb{R}^d)^C$ define

$$\ell(\mathbf{x}; \boldsymbol{\mu}) = \log \Big(\tfrac{1}{C} \sum_{i=1}^{C} \exp\big(-\tfrac{1}{2} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2_{\boldsymbol{\Sigma}^{-1}} \big) \Big).$$

Therefore, we have $\mathcal{L}(\boldsymbol{\mu}) = \mathbb{E}_{\mathbf{x}} \big[\ell(\mathbf{x}; \boldsymbol{\mu}) \big]$. Hence the quadratic form of the Hessian of ℓ in direction $\boldsymbol{\Delta} = (\boldsymbol{\Delta}_1, \ldots, \boldsymbol{\Delta}_C)$ can be written as

$$\boldsymbol{\Delta}^\top \nabla^2 \ell(\mathbf{x}; \boldsymbol{\mu})\, \boldsymbol{\Delta} = -\sum_{i=1}^{C} p_{\mathbf{x}}(\boldsymbol{\mu}_i)\, \|\boldsymbol{\Delta}_i\|^2_{\boldsymbol{\Sigma}^{-1}} + \mathrm{Var}_{p_{\mathbf{x}}} \Big(\{ \langle \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}_i),\, \boldsymbol{\Delta}_i \rangle \}_{i=1}^{C} \Big), \tag{C.12}$$

where the variance under $p_{\mathbf{x}}$ is

$$\mathrm{Var}_{p_{\mathbf{x}}}(\{u_i\}_{i=1}^C) = \sum_{i=1}^{C} p_{\mathbf{x}}(\boldsymbol{\mu}_i) u_i^2 - \Big(\sum_{i=1}^{C} p_{\mathbf{x}}(\boldsymbol{\mu}_i) u_i \Big)^2 \geq 0.$$

Define the Mahalanobis separations

$$\rho_{ij}^2 \coloneqq \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|^2_{\boldsymbol{\Sigma}^{-1}}, \qquad \rho_\star \coloneqq \min_{i \neq j} \rho_{ij}.$$

And for $i \neq j$, define

$$\Delta_{ij}(\mathbf{x}) \coloneqq \|\mathbf{x} - \boldsymbol{\mu}_j^\star\|^2_{\boldsymbol{\Sigma}^{-1}} - \|\mathbf{x} - \boldsymbol{\mu}_i^\star\|^2_{\boldsymbol{\Sigma}^{-1}} = \rho_{ij}^2 + 2 Z_{ij},$$

we obtain that

$$\Delta_{ij}(\mathbf{x}) = \rho_{ij}^2 + 2z_{ij},$$

where $z_{ij} \sim \mathcal{N}(0, \rho_{ij}^2)$. From the Gaussian tail bound and takes a union bound over $j \neq i$, we have

$$\Pr\Big\{ \min_{j\neq i} \Delta_{ij}(\mathbf{x}) \geq \tfrac{1}{2}\rho_\star^2 \Big\} \geq 1 - (C-1)e^{-\rho_\star^2/8} =: 1 - \eta_\star. \tag{C.13}$$

Under the event in (C.13) , $p_{\mathbf{x}}(\boldsymbol{\mu}_i)$ is upper bounded by $\beta_\star \in (0,1)$:

$$p_{\mathbf{x}}(\boldsymbol{\mu}_i) \geq \frac{1}{1 + \sum_{j\neq i} e^{-\frac{1}{2}\Delta_{ij}(\mathbf{x})}} \geq \frac{1}{1 + (C-1)e^{-\rho_\star^2/4}} =: \beta_\star. \tag{C.14}$$

Then, use (C.13)–(C.14) we obtain

$$\mathbb{E}_{\mathbf{x}}\big[p_{\mathbf{x}}(\boldsymbol{\mu}_i)\big] \geq \pi_i\,(1-\eta_\star)\,\beta_\star =: c_i > 0. \tag{C.15}$$

Then,

$$\begin{aligned}
\mathrm{Var}_{p_{\mathbf{x}}}(\{u_i\}_{i=1}^C) &= \sum_{i=1}^C p_{\mathbf{x}}(\boldsymbol{\mu}_i)u_i^2 - \Big(\sum_{i=1}^C p_{\mathbf{x}}(\boldsymbol{\mu}_i)u_i\Big)^2 \\
&= \Big(\sum_{i=1}^C p_{\mathbf{x}}(\boldsymbol{\mu}_i)u_i^2\Big)\Big(\sum_{i=1}^C p_{\mathbf{x}}(\boldsymbol{\mu}_i)\Big) - \Big(\sum_{i=1}^C p_{\mathbf{x}}(\boldsymbol{\mu}_i)u_i\Big)^2 \\
&= \frac{1}{2}\Big(\sum_{i,j} p_{\mathbf{x}}(\boldsymbol{\mu}_i)p_{\mathbf{x}}(\boldsymbol{\mu}_j)(u_i^2+u_j^2)\Big) - \Big(\sum_{i=1}^C p_{\mathbf{x}}(\boldsymbol{\mu}_i)u_i\Big)^2 \\
&= \sum_{i<j} p_{\mathbf{x}}(\boldsymbol{\mu}_i)p_{\mathbf{x}}(\boldsymbol{\mu}_j)(u_i-u_j)^2 \\
&\leq \sum_{i<j} p_{\mathbf{x}}(\boldsymbol{\mu}_i)p_{\mathbf{x}}(\boldsymbol{\mu}_j)2(u_i^2+u_j^2) \\
&= 2\sum_i \sum_{j\neq i} p_{\mathbf{x}}(\boldsymbol{\mu}_i)p_{\mathbf{x}}(\boldsymbol{\mu}_j)2u_i^2 \leq \sum_i (1 - \max_i p_{\mathbf{x}}^2(\boldsymbol{\mu}_i))u_i.
\end{aligned} \tag{C.16}$$

Therefore, using Equation (C.15),the first term in Equation (C.12) can be bounded as

$$\mathbb{E}\Big[-\sum_{i=1}^C p_{\mathbf{x}}(\boldsymbol{\mu}_i)\,\|\boldsymbol{\Delta}_i\|_{\boldsymbol{\Sigma}^{-1}}^2\Big] \leq -\sum_i c_i \|\boldsymbol{\Delta}_i\|_{\boldsymbol{\Sigma}^{-1}}.$$

For the variance term, split expectation on the good and bad events of Equation (C.13). On the good event, combining Equation (C.14) and Equation (C.16) gives

$$\mathrm{Var}_{p_{\mathbf{x}}}(\{u_i\}_{i=1}^C) \leq (1-\beta_\star^2)\sum_i u_i^2.$$

On the bad event, we have $\mathrm{Var}_{p_{\mathbf{x}}}(\{u_i\}_{i=1}^C) \leq \frac{1}{C^2}\sum_i u_i^2$ since $\max_i p_{\mathbf{x}}^2(\boldsymbol{\mu}_i) \geq 1/C^2$.

Next, we substitute u_i by $\langle \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_i),\, \boldsymbol{\Delta}_i\rangle$. Note that

$$\mathbb{E}_{\mathbf{x}}\big[u_i^2\big] = \boldsymbol{\Delta}_i^\top \boldsymbol{\Sigma}^{-1}\,\mathbb{E}\big[(\mathbf{x}-\boldsymbol{\mu}_i^\star)(\mathbf{x}-\boldsymbol{\mu}_i^\star)^\top\big]\,\boldsymbol{\Sigma}^{-1}\,\boldsymbol{\Delta}_i \leq k_\star\,\|\boldsymbol{\Delta}_i\|_{\boldsymbol{\Sigma}^{-1}}^2,$$

where

$$k_\star = \max_{1\leq i\leq C} \mathbb{E}_{\mathbf{x}}\big[\|(\mathbf{x}-\boldsymbol{\mu}_i)\|_{\boldsymbol{\Sigma}^{-1}}^2\big].$$

Then, by taking expectation we obtain

$$\mathbb{E}_{\mathbf{x}}\big[\mathrm{Var}_{p_{\mathbf{x}}}\big] \leq \Big((1-\eta_\star)(1-\beta_\star^2) + \tfrac{\eta_\star}{4}\Big)\,k_\star \sum_i \|\boldsymbol{\Delta}_i\|_{\boldsymbol{\Sigma}^{-1}}^2.$$

Combining the two parts,

$$\boldsymbol{\Delta}^\top \nabla^2 \mathcal{L}(\boldsymbol{\mu})\, \boldsymbol{\Delta} \leq -\Big(\min_i c_i - \big((1-\eta_\star)(1-\beta_\star^2) + \tfrac{\eta_\star}{4}\big)\, k_\star \Big) \sum_i \|\boldsymbol{\Delta}_i\|^2_{\boldsymbol{\Sigma}^{-1}}.$$

For sufficiently large $\rho_\star$, we obtain the negative definite $\nabla^2 \mathcal{L}(\boldsymbol{\mu})$.

□

In the following, we characterize $\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}}$ and compare it with $\nabla \mathcal{L}$, i.e., the gradient if GD is performed on the population loss. We have the following lemma.

Lemma 4 (Properties of the CoT gradient descent)**.** *Fix an epoch* t *and a component index* $i \in [C]$*, there exist constants* $c_1, c_2 > 0$ *such that, for every* $M \geq 1$*,*

$$\Pr\Big(\big\| \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}} - \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \mathcal{L} \big\| \leq c_1\, M^{-1/4} \Big) \geq 1 - \exp(-\sqrt{M}),$$

and

$$\Pr\Big(\big\| \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}} \big\|^2 \leq c_2 + c_3 M^{-1/2} \Big) \geq 1 - \exp(-\sqrt{M}).$$

Proof. Recall that

$$\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}} = \frac{1}{M} \sum_{j=N+1}^{N+M} p_{ij}^{(t)} \big(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j \big)$$

where $\widehat{p}_{ij}^{(k,t)}$ is given by

$$p_{ij}^{(t)} = \frac{\sum_{\tau=0}^{t} \exp\Big(-\frac{1}{2} \|\widehat{\boldsymbol{\mu}}_i^{(\tau)} - \mathbf{x}_j\|^2_{\boldsymbol{\Sigma}^{-1}} + \beta\tau \Big)}{\sum_{\tau=0}^{t} \sum_{c=1}^{C} \exp\Big(-\frac{1}{2} \|\widehat{\boldsymbol{\mu}}_c^{(\tau)} - \mathbf{x}_j\|^2_{\boldsymbol{\Sigma}^{-1}} + \beta\tau \Big)}.$$

By choosing $\beta \to \infty$, we further have

$$p_{ij}^{(t)} = \frac{\exp\Big(-\frac{1}{2} \|\widehat{\boldsymbol{\mu}}_i^{(\tau)} - \mathbf{x}_j\|^2_{\boldsymbol{\Sigma}^{-1}} \Big)}{\sum_{c=1}^{C} \exp\Big(-\frac{1}{2} \|\widehat{\boldsymbol{\mu}}_c^{(\tau)} - \mathbf{x}_j\|^2_{\boldsymbol{\Sigma}^{-1}} \Big)}.$$

where the samples $\{\mathbf{x}_j\}_{j \geq N+1}$ are drawn from a Gaussian mixture distribution.

Therefore, given $\widehat{\boldsymbol{\mu}}_{ij}^{(t)}$, the random variable $p_{ij}^{(t)} \big(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j \big)$ admits a sub-Gaussian tail bound since $\mathbf{x}_j$ are Guassian random vectors and $p_{ij}^{(t)} \big(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j \big)$ is Lipschitz continuous over $\mathbf{x}_j$.

Then, by the Bernstein's inequality, for any fixed $\delta \in (0,1)$, with probability at least $1 - \delta$, we have

$$\begin{aligned} \Big\| \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}} - \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \mathcal{L} \Big\| &= \left\| \frac{1}{M} \sum_{j=N+1}^{N+M} p_{ij}^{(t)} \big(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j \big) - \mathbb{E}_{\mathbf{x}_j} \Big[p_{ij}^{(t)} \big(\widehat{\boldsymbol{\mu}}_i^{(t)} - \mathbf{x}_j \big) \Big] \right\| \\ &\leq \frac{c_4}{\sqrt{M}} \sqrt{\log\Big(\tfrac{2}{\delta} \Big)}, \end{aligned}$$

where $c_4 > 0$ is some absolute constant.

By choosing $\delta = \exp(-\sqrt{M})$, we obtain that with probability at least $1 - \exp(-\sqrt{M})$,

$$\Big\| \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}} - \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \mathcal{L} \Big\| \leq c'\, M^{-\frac{1}{4}}.$$

for another constant $c' > 0$. This completes the proof of the first inequality.

Next, we show that $\|\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}} \widehat{\mathcal{L}}\|$ itself is bounded with high probability.

Consequently,

$$\begin{aligned}\left\|\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}}\widehat{\mathcal{L}}\right\| &= \left\|\frac{1}{M}\sum_{j=N+1}^{N+M} p_{ij}^{(t)}\left(\widehat{\boldsymbol{\mu}}_i^{(t)}-\mathbf{x}_j\right)\right\| \\ &\le \frac{1}{M}\sum_{j=N+1}^{N+M}\left\|p_{ij}^{(t)}\left(\widehat{\boldsymbol{\mu}}_i^{(t)}-\mathbf{x}_j\right)\right\| \\ &\overset{(a)}{\le} \frac{1}{M}\sum_{j\ge N+1}\left\|\widehat{\boldsymbol{\mu}}_i^{(t)}-\mathbf{x}_j\right\| \\ &\le \frac{1}{M}\sum_{j\ge N+1}\left(\|\widehat{\boldsymbol{\mu}}_i^{(t)}\|+\|\mathbf{x}_j\|\right) \\ &= \|\widehat{\boldsymbol{\mu}}_i^{(t)}\| + \frac{1}{M}\sum_{j\ge N+1}\|\mathbf{x}_j\|, \end{aligned} \tag{C.17}$$

where inequality (a) holds since $p_{ij}^{(t)} \le 1$. Note that

$$\begin{aligned}\widehat{\boldsymbol{\mu}}_i^{(t)} &= \widehat{\boldsymbol{\mu}}_i^{(t-1)} - \frac{\eta^{(t-1)}}{M}\sum_{j=N+1}^{N+M} p_{ij}^{(t-1)}\left(\widehat{\boldsymbol{\mu}}_i^{(t-1)}-\mathbf{x}_j\right) \\ &= \left(1-\frac{\eta^{(t-1)}}{M}\right)\widehat{\boldsymbol{\mu}}_i^{(t-1)} + \frac{\eta^{(t-1)}}{M}\sum_{j=N+1}^{N+M} p_{i,j}^{(t-1)}\mathbf{x}_j.\end{aligned}$$

Therefore, we have

$$\begin{aligned}\|\widehat{\boldsymbol{\mu}}_i^{(t)}\| &\le \|\widehat{\boldsymbol{\mu}}_i^{(t-1)}\| + \frac{1}{M}\sum_{j\ge N+1}\|\mathbf{x}_j\| \\ &\le \|\widehat{\boldsymbol{\mu}}_i^{(1)}\| + \frac{t-1}{M}\sum_{j\ge N+1}\|\mathbf{x}_j\|\end{aligned}$$

Combining with Equation (C.17), we have

$$\left\|\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}}\widehat{\mathcal{L}}\right\| \le \|\widehat{\boldsymbol{\mu}}_i^{(1)}\| + \frac{t}{M}\sum_{j\ge N+1}\|\mathbf{x}_j\|.$$

Applying the Bernstein's inequality, with probability at least $1-\exp(-\sqrt{M})$, we have

$$\frac{1}{M}\sum_{j\ge N+1}\|\mathbf{x}_j\| \le \frac{1}{C}\sum_{i=1}^{C}\boldsymbol{\mu}_i + c_5 M^{-\frac{1}{4}},$$

where c_5 is a positive constant.

Therefore, for any $t \le T$ where T is total number of CoT steps, we have

$$\|\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}}\widehat{\mathcal{L}}\| \le \|\widehat{\boldsymbol{\mu}}_i^{(1)}\| + \frac{T}{C}\sum_{i=1}^{C}\boldsymbol{\mu}_i + c_5 t M^{-\frac{1}{4}},$$

which implies

$$\|\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}}\widehat{\mathcal{L}}\|^2 \le c_2 + c_3 M^{-\frac{1}{2}},$$

where c_2 and c_3 are positive constants depends on T, $\|\widehat{\boldsymbol{\mu}}_i^{(1)}\|$ and $\frac{T}{C}\sum_{i=1}^{C}\boldsymbol{\mu}_i$. The proof is thus complete. □

Step 3: Finally, we show the convergence of the class mean estimation error.

Expanding the squared error $\|\widehat{\boldsymbol{\mu}}_i^{(t+1)} - \boldsymbol{\mu}_i\|^2$ gives

$$\begin{aligned}\|\widehat{\boldsymbol{\mu}}_i^{(t+1)} - \boldsymbol{\mu}_i\|^2 =& \|\widehat{\boldsymbol{\mu}}_i^{(t)} - \boldsymbol{\mu}_i\|^2 + 2\,\eta^{(t)}\Big\langle \widehat{\boldsymbol{\mu}}_i^{(t)} - \boldsymbol{\mu}_i,\, \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}}\widehat{\mathcal{L}}\Big\rangle + (\eta^{(t)})^2\left\|\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}}\widehat{\mathcal{L}}\right\|^2 \\ \le& \|\widehat{\boldsymbol{\mu}}_i^{(t)} - \boldsymbol{\mu}_i\|^2 + 2\,\eta^{(t)}\Big\langle \widehat{\boldsymbol{\mu}}_i^{(t)} - \boldsymbol{\mu}_i,\, \nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}}\mathcal{L}\Big\rangle + 2\,\eta^{(t)}\left\|\nabla\mathcal{L} - \nabla\widehat{\mathcal{L}}\right\| \\ &+ (\eta^{(t)})^2\left\|\nabla_{\widehat{\boldsymbol{\mu}}_i^{(t)}}\widehat{\mathcal{L}}\right\|^2. \end{aligned} \tag{C.18}$$

Denote $\widehat{\boldsymbol{\mu}}^{(t)}$ and $\boldsymbol{\mu}$ as the vectors obtained by stacking $\{\widehat{\boldsymbol{\mu}}_i^{(t)}\}_{i=1}^C$ and $\{\boldsymbol{\mu}_i\}_{i=1}^C$, respectively. Therefore, we have

$$\|\widehat{\boldsymbol{\mu}}^{(t+1)} - \boldsymbol{\mu}\|^2 \le \|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 + 2\,\eta^{(t)}\big\langle \widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu},\, \nabla_{\widehat{\boldsymbol{\mu}}^{(t)}}\mathcal{L}\big\rangle + 2\,\eta^{(t)}\left\|\nabla\mathcal{L} - \nabla\widehat{\mathcal{L}}\right\| + (\eta^{(t)})^2\left\|\nabla_{\widehat{\boldsymbol{\mu}}^{(t)}}\widehat{\mathcal{L}}\right\|^2.$$

To control the inner product term $\big\langle \widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu},\, \nabla_{\widehat{\boldsymbol{\mu}}^{(t)}}\mathcal{L}\big\rangle$, we perform a first-order Taylor expansion of $\nabla_{\widehat{\boldsymbol{\mu}}^{(t)}}\mathcal{L}$ around $\boldsymbol{\mu}$ as

$$\begin{aligned}\nabla_{\widehat{\boldsymbol{\mu}}^{(t)}}\mathcal{L} &= \nabla_{\boldsymbol{\mu}}\mathcal{L} + (\nabla^2_{\boldsymbol{\mu}}\mathcal{L})\,(\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}) + \mathbf{R}(\widehat{\boldsymbol{\mu}}^{(t)}, \boldsymbol{\mu}) \\ &\overset{(a)}{=} (\nabla^2_{\boldsymbol{\mu}}\mathcal{L})\,(\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}) + \mathbf{R}(\widehat{\boldsymbol{\mu}}^{(t)}, \boldsymbol{\mu}),\end{aligned}$$

where equality (a) holds since $\boldsymbol{\mu}$ is the global minimizer of $\mathcal{L}$ and $\mathcal{L}$ is differentiable on $\mathbb{R}^d$, thus $\nabla_{\boldsymbol{\mu}}\mathcal{L} = 0$, and $\mathbf{R}(\widehat{\boldsymbol{\mu}}^{(t)}, \boldsymbol{\mu})$ is the remainder term.

For the remainder term, we have

$$\begin{aligned}&\langle \mathbf{R}(\widehat{\boldsymbol{\mu}}^{(t)}, \boldsymbol{\mu}), \widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\rangle \\ &= \int_0^1 (\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu})^\top\Big(\nabla^2_{\boldsymbol{\mu}+\xi(\widehat{\boldsymbol{\mu}}^{(t)}-\boldsymbol{\mu})}\mathcal{L} - \nabla^2_{\boldsymbol{\mu}}\mathcal{L}\Big)(\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu})\mathrm{d}\xi \\ &\le \int_0^1 \left\|\nabla^2_{\boldsymbol{\mu}+\xi(\widehat{\boldsymbol{\mu}}^{(t)}-\boldsymbol{\mu})}\mathcal{L} - \nabla^2_{\boldsymbol{\mu}}\mathcal{L}\right\| \|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2\mathrm{d}\xi \\ &\overset{(b)}{\le} \int_0^1 L\xi\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^3\mathrm{d}\xi = L\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^3,\end{aligned}$$

where Inequality (b) follows from the fact that $\nabla^2\mathcal{L}$ is twice continuously differentiable, its Jacobian is Lipchitz continuous in a neighborhood of $\boldsymbol{\mu}$, and L is the Lipchitz constant.

Therefore, there exists a constant $\lambda > 0$ such that

$$\begin{aligned}&\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 + \Big\langle \widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}, 2\eta^{(t)}\nabla_{\boldsymbol{\mu}}\mathcal{L}^{(t)}\Big\rangle \\ &\le \|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 + 2\eta^{(t)}(\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu})^\top\nabla^2_{\boldsymbol{\mu}}\mathcal{L}(\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}) + 2\eta^{(t)}L\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^3 \\ &\overset{(c)}{\le} \Big(1 - 2\eta^{(t)}\lambda\Big)\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 + 2\eta^{(t)}L\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^3,\end{aligned} \tag{C.19}$$

where Inequality (c) follows from Lemma 3 which proves $\nabla^2_{\boldsymbol{\mu}}\mathcal{L}$ is negative definite.

Meanwhile, Lemma 4 ensures with probability at least $1 - \exp(-\sqrt{M})$,

$$\eta^{(t)}\left\|\nabla\mathcal{L} - \nabla\widehat{\mathcal{L}}\right\| \le c_1\,\eta^{(t)}\,M^{-\frac{1}{4}}, \tag{C.20}$$

$$(\eta^{(t)})^2\left\|\nabla_{\widehat{\boldsymbol{\mu}}^{(t)}}\widehat{\mathcal{L}}\right\|^2 \le c_2\,(\eta^{(t)})^2\,M^{-\frac{1}{2}} + c_3\,(\eta^{(t)})^2. \tag{C.21}$$

Substituting (C.19), (C.20), and (C.21) into (C.18) then yields the one-step error recursion

$$\begin{aligned}\|\widehat{\boldsymbol{\mu}}^{(t+1)} - \boldsymbol{\mu}\|^2 \le& \Big(1 - 2\eta^{(t)}\lambda\Big)\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 + 2\eta^{(t)}L\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^3 \\ &+ c_1\eta^{(t)}M^{-\frac{1}{4}} + c_2(\eta^{(t)})^2M^{-\frac{1}{2}} + c_3(\eta^{(t)})^2.\end{aligned} \tag{C.22}$$

Next, we aim prove $\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 \le K/(t + T')$ for a positive constant K by induction.

Let $\eta^{(t)} = \frac{\alpha}{t+T'}$ and $M^{(t)} = (t+T')^p$ for some $p > 4$. First, assume $\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 \leq K/(t+T')$ for a fixed $t \geq 1$. From Equation (C.22), we note that there exists a constant $c_4 > 0$ such that

$$\begin{aligned}\|\widehat{\boldsymbol{\mu}}^{(t+1)} - \boldsymbol{\mu}\|^2 &\leq \left(1 - 2\frac{\alpha\lambda}{t+T'}\right)\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 + c_3\frac{\alpha^2}{(t+T')^2} + 2\frac{\alpha L}{t+T'}\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^3 + c_4\frac{\alpha}{t+T'}t^{-\frac{p}{4}} \\ &\leq \left(1 - 2\frac{\alpha\lambda}{t+T'}\right)\frac{K}{t+T'} + 2\frac{\alpha L}{t+T'}\left(\frac{K}{t+T'}\right)^{\frac{3}{2}} + c_4\alpha(t+T')^{-(1+\frac{p}{4})} + c_3\frac{\alpha^2}{(t+T')^2}.\end{aligned}$$

Therefore, we have

$$\begin{aligned}&\|\widehat{\boldsymbol{\mu}}^{(t+1)} - \boldsymbol{\mu}\|^2 - \frac{K}{t+T'+1} \qquad \text{(C.23)} \\ &\leq \left(1 - 2\frac{\alpha\lambda}{t+T'}\right)\frac{K}{t+T'} + 2\frac{\alpha L}{t+T'}\left(\frac{K}{t+T'}\right)^{\frac{3}{2}} + c_4\alpha(t+T')^{-(1+\frac{p}{4})} + c_3\frac{\alpha^2}{(t+T')^2} - \frac{K}{t+T'} + \frac{K}{(t+T')^2} \\ &= (-2\alpha\lambda+1)\frac{K}{(t+T')^2} + 2\frac{\alpha L}{t+T'}\left(\frac{K}{t+T'}\right)^{\frac{3}{2}} + c_4\alpha(t+T')^{-(1+\frac{p}{4})} + c_3\frac{\alpha^2}{(t+T')^2}. \qquad \text{(C.24)}\end{aligned}$$

By choosing $\alpha \geq 1/\lambda$, $K \geq \max\{3c_3\alpha^2, 3c_4\alpha\}$ and $T' \geq 36\alpha^2L^2K$, we have

$$\begin{aligned}&(-2\alpha\lambda+1)\frac{K}{(t+T')^2} \leq -\frac{K}{(t+T')^2}, \\ &2\frac{\alpha L}{t+T'}\Big(\frac{K}{t+T'}\Big)^{\frac{3}{2}} \leq \frac{K}{3(t+T')^2}, \\ &c_4\alpha(t+T')^{-\left(1+\frac{p}{4}\right)} \leq \frac{K}{3(t+T')^2}, \\ &c_3\frac{\alpha^2}{(t+T')^2} \leq \frac{K}{3(t+T')^2}.\end{aligned} \qquad \text{(C.25)}$$

Therefore, by substituting Equation (C.25) into Equation (C.24), we have

$$\|\widehat{\boldsymbol{\mu}}^{(t+1)} - \boldsymbol{\mu}\|^2 - \frac{K}{t+T'+1} \leq 0, \quad \forall t \geq 1.$$

Recall Lemma 1 indicates that, with probability at least $1 - \exp(-cNK/T' + 1)$, for some constant c, it holds that $\|\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}\| \leq K/(T'+1)$. Therefore, for any fixed $\epsilon \in [0,1]$, if

$$\begin{aligned}&N \geq 36\alpha^2L^2\log 1/\epsilon, \\ &M \geq \max\{(T')^4, \log^2 1/\epsilon\}, \\ &t \geq \sqrt[4]{M},\end{aligned}$$

with probability at least $1-\epsilon$, the estimation error is upper bounded by

$$\|\widehat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{\mu}\|^2 \leq c\frac{\log(1/\epsilon)}{N\sqrt[4]{M}},$$

where c is a positive constant. This completes the proof of Theorem 4.2.

C.3 Proof of Corollary 4.1

First, we restate the corollary below.

Corollary C.1 (Restatement of Corollary 4.1)**.** *Let $\widehat{\mathbf{y}}_j$ be the predicted label for $\mathbf{x}_j$ according to Equation* (3.5)*. Let $\mathcal{R}^*$ be the prediction error under the Bayes-optimal classifier with known class mean vectors $\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_C$. Then, under the same conditions as described in Theorem 4.2, we have*

$$\mathbb{P}[\widehat{\mathbf{y}}_j \neq \mathbf{y}|\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_C] - \mathcal{R}^* \leq \mathcal{O}(1/\sqrt{N\text{poly}(M)}).$$

Proof. First, we define $\Delta = \|\widehat{\mathbf{M}} - \mathbf{M}\|_F$, define $\widehat{g}$ as the Bayes-optimal classifier given estimated class means $\widehat{\mathbf{M}}$ and define g as the Bayes-optimal classifier given ground truth class means $\mathbf{M}$ Suppose $\widehat{g}(\mathbf{x}) \neq g(\mathbf{x})$. Then, there exist indices $i \neq k$ such that $g(\mathbf{x}) = i$ and $\widehat{g}(\mathbf{x}) = k$. Because $g(\mathbf{x}) = i$ is Bayes-optimal, we have

$$\|\mathbf{x} - \boldsymbol{\mu}_i\| \leq \|\mathbf{x} - \boldsymbol{\mu}_k\| \text{ and } \|\mathbf{x} - \widehat{\boldsymbol{\mu}}_k\| \leq \|\mathbf{x} - \widehat{\boldsymbol{\mu}}_i\|.$$

Denote $\zeta = \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_k\|$. Therefore, from the geometric observation, the misclassification only happens when $\mathbf{x}$ is in the dihedral cone with angle θ, where $\tan(\theta) = \Delta/\zeta$ (Diakonikolas et al., 2018).Thus, the probability for misclassification is upper bounded

$$\mathbb{P}[\widehat{g}(\mathbf{x}) \neq g(\mathbf{x})] \leq c'\theta,$$

for a positive constant c'. Since $\mathbb{P}[\widehat{\mathbf{y}}_j \neq \mathbf{y}|\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_C] - \mathcal{R}^* = \mathbb{P}[\widehat{g}(\mathbf{x}) \neq g(\mathbf{x})]$ and from Theorem 4.2 we have $\Delta \leq c'\sqrt{1/N\sqrt[4]{(M)}}$ for positive constant c', the proof is thus complete. □

D Proof of Training Dynamics

First, we restate Theorem 5.1 below.

Theorem D.1 (Restatement of Theorem 5.1)**.** *Let $\{\mathbf{Q}^{(k)}, \mathbf{K}^{(k)}, \mathbf{V}^{(k)}\}_{k\geq 0}$ be the parameters of the first attention layer of the transformer after applying k iterations of gradient descent on the population loss defined in Equation* (5.2)*. Then, with the initialization specified in Assumption 1, we have*

$$\|\mathbf{W}^{(k)} - \boldsymbol{\Sigma}^{-1}\|_F^2 \leq c^k \|\mathbf{W}^{(0)} - \boldsymbol{\Sigma}^{-1}\|_F^2,$$

for some positive constant c, while the other parameters in $\mathbf{Q}^{(0)}$, $\mathbf{K}^{(0)}$ and $\mathbf{V}^{(0)}$ remain unchanged.

We assume ground truth means are IID sampled from standard Gaussian distribution: $\boldsymbol{\mu}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for all i. Then, we introduce the following quantities: 1) the formulation of class mean estimations given by the transformer during teacher forcing training; 2) the reference class mean estimations given by the reference policy; and 3) the formulation of the gradient of the teacher forcing training loss.

At the k-th GD iteration during training, we denote the set of reference class mean estimations as $\boldsymbol{\mu}_{\mathrm{ref},1}^{(k,t)}, \cdots, \boldsymbol{\mu}_{\mathrm{ref},C}^{(k,t)}$ for the CoT steps $t \in [T]$. Given the reference class mean estimations, the estimation given by the transformer throughout teacher forcing satisfies

$$\widehat{\boldsymbol{\mu}}_i^{(k,t+1)} = \boldsymbol{\mu}_{\mathrm{ref},i}^{(k,t)} - \frac{\eta^{(t)}}{M} \sum_{j=N+1}^{N+M} \widehat{p}_{ij}^{(k,t)} \left(\boldsymbol{\mu}_{\mathrm{ref},i}^{(k,t)} - \mathbf{x}_j\right),$$

where $\widehat{p}_{ij}^{(k,t)}$ is given by

$$\widehat{p}_{ij}^{(k,t)} = \frac{\sum_{\tau=0}^{t} \exp\left(-\frac{w}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}\|^2 + \mathbf{x}_j^\top \mathbf{W}^{(k)} \widehat{\boldsymbol{\mu}}_i^{(\tau)} + \beta\tau\right)}{\sum_{\tau=0}^{t}\sum_{c=1}^{C} \exp\left(-\frac{w}{2}\|\widehat{\boldsymbol{\mu}}_c^{(\tau)}\|^2 + \mathbf{x}_j^\top \mathbf{W}^{(k)} \widehat{\boldsymbol{\mu}}_c^{(\tau)} + \beta\tau\right)}.$$

By choosing $\beta \to \infty$, we further have

$$\widehat{p}_{ij}^{(k,t)} = \frac{\exp\left(-\frac{w}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}\|^2 + \mathbf{x}_j^\top \mathbf{W}^{(k)} \widehat{\boldsymbol{\mu}}_i^{(\tau)}\right)}{\sum_{c=1}^{C} \exp\left(-\frac{w}{2}\|\widehat{\boldsymbol{\mu}}_c^{(\tau)}\|^2 + \mathbf{x}_j^\top \mathbf{W}^{(k)} \widehat{\boldsymbol{\mu}}_c^{(\tau)}\right)}.$$

We choose the reference policy under which

$$\boldsymbol{\mu}_{\mathrm{ref},i}^{(k,t+1)} = \boldsymbol{\mu}_{\mathrm{ref},i}^{(k,t)} - \frac{\eta^{(t)}}{M} \sum_{j=N+1}^{N+M} p_{ij}^{(k,t)} \left(\boldsymbol{\mu}_{\mathrm{ref},i}^{(k,t)} - \mathbf{x}_j\right),$$

with

$$p_{ij}^{(k,t)} = \frac{\exp\left(-\frac{1}{2}\|\boldsymbol{\mu}_{\mathrm{ref},i}^{(k,t)} - \mathbf{x}_j\|_{\Sigma^{-1}}^2\right)}{\sum_{c=1}^{C} \exp\left(-\frac{1}{2}\|\boldsymbol{\mu}_{\mathrm{ref},c}^{(k,t)} - \mathbf{x}_j\|_{\Sigma^{-1}}^2\right)}.$$

To simplify the notation, when there is no ambiguity, we drop the superscript (k) for the training iteration. Denote $\widehat{\mathbf{q}}_j^{(t)} = [\widehat{p}_{1j}^{(t)} \cdots \widehat{p}_{Cj}^{(t)}]$ and $\mathbf{q}_j^{(t)} = [p_{1j}^{(t)} \cdots p_{Cj}^{(t)}]$. At the k-th training iteration, the CoT training loss with teacher forcing is

$$\begin{aligned}\widehat{\mathcal{L}}_{\text{CoT-train}}(\mathbf{\Theta}; \mathcal{I}_{\mathbf{M}}) &= \frac{1}{T}\sum_{t=1}^{T}\sum_{j=N+1}^{N+M} \mathrm{CE}\left(\mathbf{q}_j^{(t)}, [\mathrm{TF}_{\mathbf{\Theta}}(\mathbf{H}_{\text{ref}}^{(t-1)})]_{2d+2c+1:2d+3c,N+j}\right) \\ &= \frac{1}{T}\sum_{t=1}^{T}\sum_{j=N+1}^{N+M} \mathrm{CE}\left(\mathbf{q}_j^{(t)}, \widehat{\mathbf{q}}_j^{(t)}\right),\end{aligned}$$

where CE is the cross entropy loss function.

Define $s_{ij}^{(t)} = -\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}\|^2 + \mathbf{x}_j^\top \mathbf{W}^{(k)} \widehat{\boldsymbol{\mu}}_i^{(\tau)}$ and $\mathbf{s}_j^{(t)} = [s_{1j}^{(t)} \cdots s_{Cj}^{(t)}]$. Note that the derivative can be written as

$$\frac{\partial \mathrm{CE}\left(\mathbf{q}_j^{(t)}, \widehat{\mathbf{q}}_j^{(t)}\right)}{\partial s_{ij}^{(t)}} = \frac{\partial \frac{\exp(s_{ij}^{(t)})}{\sum_{k=1}^{C}\exp(s_{kj}^{(t)})}}{\partial s_{ij}^{(t)}} = \widehat{p}_{ij}^{(t)} - p_{ij}^{(t)}.$$

Furthermore, since $\partial s_{ij}^{(t)}/\partial \mathbf{W}_{ab} = \mathbf{M}_{a,i}\mathbf{x}_{jb}$, where $a, b \in [d]$, by the chain rule, we have

$$\frac{\partial \mathrm{CE}\left(\mathbf{q}_j^{(t)}, \widehat{\mathbf{q}}_j^{(t)}\right)}{\partial \mathbf{W}_{ab}} = \frac{\partial \mathrm{CE}\left(\mathbf{q}_j^{(t)}, \widehat{\mathbf{q}}_j^{(t)}\right)}{\partial s_{ij}^{(t)}} \frac{\partial s_{ij}^{(t)}}{\partial \mathbf{W}_{ab}} = \sum_i (\widehat{p}_{ij}^{(t)} - p_{ij}^{(t)})\mathbf{M}_{a,i}\mathbf{x}_{jb}. \tag{D.1}$$

Based on the notations, we will prove Theorem 5.1 as follows.

Step 1: Given the gradient of the cross entropy loss with respect to the learnable parameter matrix $\mathbf{W}$, our first step is to provide a decomposition of the gradient so that it becomes analytically tractable.

In the matrix form, Equation (D.1) can be written as

$$\nabla_{\mathbf{W}} \mathrm{CE}\left(\mathbf{q}_j^{(t)}, \widehat{\mathbf{q}}_j^{(t)}\right) = \mathbf{M}(\widehat{\mathbf{q}}_j^{(t)} - \mathbf{q}_j^{(t)})\mathbf{x}_j^T.$$

By the Stein's lemma, we have

$$\begin{aligned}&\mathbb{E}[\mathbf{M}(\widehat{\mathbf{q}}_j^{(t)} - \mathbf{q}_j^{(t)})\mathbf{x}_j^\top] \\ &= \mathbb{E}_{\mathbf{M}}\left[\mathbf{M}\left[\mathbb{E}_{\mathbf{x}_j}[\widehat{\mathbf{q}}_j^{(t)} - \mathbf{q}_j^{(t)}]\mathbb{E}[\mathbf{x}_j^\top] + \mathbb{E}_{\mathbf{x}_j}[\nabla\widehat{\mathbf{q}}_j^{(t)} - \nabla\mathbf{q}_j^{(t)}]\mathbf{\Sigma}\right]\right] \\ &= \underbrace{\mathbb{E}_{\mathbf{M}}\left[\mathbf{M}\mathbb{E}_{\mathbf{x}_j}[\widehat{\mathbf{q}}_j^{(t)} - \mathbf{q}_j^{(t)}]\mathbb{E}[\mathbf{x}^\top]\right]}_{\mathcal{A}_1} + \underbrace{\mathbb{E}_{\mathbf{M}}\left[\mathbf{M}\mathbb{E}_{\mathbf{x}_j}[\nabla\widehat{\mathbf{q}}_j^{(t)} - \nabla\mathbf{q}_j^{(t)}]\mathbf{\Sigma}\right]}_{\mathcal{A}_2}.\end{aligned}$$

Step 2: Based on the decomposition, we aim to show that $\mathcal{A}_1 = 0$.

We note that when taking the expectation over the labeled dataset, we have

$$\mathbb{E}\left[\frac{C}{N}\sum_{j\in[N]} \mathbf{x}_j \cdot \left(\mathbf{e}_i^\top \mathbf{y}_j\right)\right] = \boldsymbol{\mu}_i.$$

Therefore, $\boldsymbol{\mu}_{\text{ref,i}}^0 = \boldsymbol{\mu}_i$. When the reference class mean estimations are generated by gradient descent over the population loss, we have $\boldsymbol{\mu}_{\text{ref},i}^{(t)} = \boldsymbol{\mu}_i$ for any $i \in [C]$ and $t \in [T]$ the gradient

over the population loss is zero:

$$\begin{aligned}
&\mathbb{E}_{\mathbf{x}}\left[\frac{\exp\left(-\frac{1}{2}\|\boldsymbol{\mu}_{\mathrm{ref},i}^{(t)}-\mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2\right)}{\sum_{c=1}^{C}\exp\left(-\frac{1}{2}\|\boldsymbol{\mu}_{\mathrm{ref},c}^{(t)}-\mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2\right)}\left(\boldsymbol{\mu}_{\mathrm{ref},i}^{(t)}-\mathbf{x}_j\right)\right]\\
&=\int_{\mathbb{R}^d}\frac{\exp(-\frac{1}{2}\|\boldsymbol{\mu}_{\mathrm{ref},i}^{(t)}-\mathbf{x}\|_{\boldsymbol{\Sigma}^{-1}}^2)}{\sum_{c=1}^{C}\exp(-\frac{1}{2}\|\boldsymbol{\mu}_{\mathrm{ref},c}^{(t)}-\mathbf{x}\|_{\boldsymbol{\Sigma}^{-1}}^2)}\left[\sum_{k=1}^{C}\frac{1}{C}\varphi_k(\mathbf{x})\right](\boldsymbol{\mu}_{\mathrm{ref},i}^{(t)}-\mathbf{x})d\mathbf{x},\\
&\overset{(a)}{=}\int_{\mathbb{R}^d}\frac{1}{C}\varphi_i(\mathbf{x})(\boldsymbol{\mu}_i-\mathbf{x})d\mathbf{x}=0,
\end{aligned}$$

where $\varphi_i(\mathbf{x})$ is the pdf of Gaussian distribution with mean $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}$, and equality (a) holds since $\boldsymbol{\mu}_{\mathrm{ref},i}^{(k,t)}=\boldsymbol{\mu}_i$. Given the above-discussed property of the reference class mean estimations, for $\mathbb{E}_{\mathbf{x}_j}[\widehat{\mathbf{q}}_j^{(t)}-\mathbf{q}_j^{(t)}]$ in $\mathcal{A}_1$, its is obvious that $\mathbb{E}_{\mathbf{x}_j}[\mathbf{q}_j^{(t)}]=1/C$. For $\mathbb{E}_{\mathbf{x}_j}[\widehat{\mathbf{q}}_j^{(t)}]$, we let $\mathbf{W}^{(0)}$ initialize form a isotropic matrix $w\mathbf{I}$, and we assume at training iteration step t, it preserve the isotropic as $\mathbf{W}^{(t)}$. Therefore, since the ground truth $\boldsymbol{\Sigma}$ is an isotropic matrix, the temperature acts identically on all classes:

$$\mathbb{E}\left[\frac{\exp\left(-\frac{\alpha}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}-\mathbf{x}_j\|^2\right)}{\sum_{c=1}^{C}\exp\left(-\frac{\alpha}{2}\|\boldsymbol{\mu}_c-\mathbf{x}_j\|^2\right)}\right]=\mathbb{E}\left[\frac{\exp\left(-\frac{1}{2}\|\widehat{\boldsymbol{\mu}}_i^{(\tau)}-\mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2\right)}{\sum_{c=1}^{C}\exp\left(-\frac{1}{2}\|\boldsymbol{\mu}_c-\mathbf{x}_j\|_{\boldsymbol{\Sigma}^{-1}}^2\right)}\right].$$

Therefore, we have $\mathbb{E}_{\mathbf{x}_j}[\widehat{\mathbf{p}}_j^{(t)}-\mathbf{p}_j^{(t)}]=0$, which gives $\mathcal{A}_1=0$.

Step 3: Finally, we analyze the properties of $\mathcal{A}_2$, and obtain the final results afterwards. We will prove that $\mathbf{W}^{(t)}$ preserves isotropic by induction. Note that we assume training iteration step t, $\mathbf{W}^{(t)}$ is isotropic. Besides, we initialize $\mathbf{W}^{(0)}$ as an isotropic matrix.

$\mathcal{A}_2$ can be rewritten as

$$\begin{aligned}
\mathcal{A}_2&=\mathbb{E}_{\mathbf{M}}\left[\mathbf{M}\mathbb{E}_{\mathbf{x}_j}[\nabla\widehat{\mathbf{q}}_j^{(t)}-\nabla\mathbf{q}_j^{(t)}]\boldsymbol{\Sigma}\right]\\
&=\mathbb{E}_{\mathbf{M}}\left[\mathbf{M}\left(\left(\mathrm{diag}(\mathbb{E}[\widehat{\mathbf{q}}_j^{(t)}])-\mathbb{E}_{\mathbf{x}}[\widehat{\mathbf{q}}_j^{(t)}(\widehat{\mathbf{q}}_j^{(t)})^\top]\right)\mathbf{M}^\top\mathbf{W}^{(k)}\boldsymbol{\Sigma}-\left(\mathrm{diag}(\mathbb{E}[\mathbf{q}_j^{(t)}])-\mathbb{E}_{\mathbf{x}}[\mathbf{q}_j^{(t)}(\mathbf{q}_j^{(t)})^\top]\right)\mathbf{M}^\top\right)\right].
\end{aligned}$$

Because the class prior is uniform and the isotropic initialisation, we have

$$\mathbb{E}_{\mathbf{x}_j}[\widehat{\mathbf{q}}_j^{(t)}]=\mathbb{E}_{\mathbf{x}_j}[\mathbf{q}_j^{(t)}]=\frac{1}{C}\mathbf{1}.$$

Since each coordinate of $\widehat{\mathbf{q}}_j^{(t)}$ (or $\mathbf{q}_j^{(t)}$) has the same marginal distribution and any two distinct coordinates have the same joint distribution, we have

$$\mathrm{diag}(\mathbb{E}[\widehat{\mathbf{q}}_j^{(t)}])=\mathrm{diag}(\mathbf{q}_j^{(t)})=\frac{1}{C}\mathbf{I},\qquad\mathbb{E}_{\mathbf{x}}[\widehat{\mathbf{q}}_j^{(t)}(\widehat{\mathbf{q}}_j^{(t)})^\top]=\mathbb{E}_{\mathbf{x}}[\mathbf{q}_j^{(t)}(\mathbf{q}_j^{(t)})^\top]=\frac{1}{C^2}\mathbf{1}\mathbf{1}^\top.$$

Therefore, we have

$$\mathcal{A}_2=\mathbb{E}_{\mathbf{M}}\left[\mathbf{M}\left(\mathrm{diag}(1/C)-\frac{1}{C^2}\mathbf{1}\mathbf{1}^\top\right)\mathbf{M}^\top\left(\mathbf{W}^{(k)}\boldsymbol{\Sigma}-\mathbf{I}\right)\right].$$

Note that $\nabla_{\mathbf{W}}L_{\mathrm{CoT}}(\mathbf{W}^{(k)})=\mathcal{A}_2$, therefore, we obtain

$$\begin{aligned}
\|\nabla_{\mathbf{W}}L_{\mathrm{CoT}}(\mathbf{W}^{(k)})\|_F&=\left\|\mathbb{E}_{\mathbf{M}}\left[\mathbf{M}\left(\mathrm{diag}(1/C)-\frac{1}{C^2}\mathbf{1}\mathbf{1}^\top\right)\mathbf{M}^\top\left(\mathbf{W}^{(k)}\boldsymbol{\Sigma}-\mathbf{I}\right)\right]\right\|_F\\
&=\left\|\mathbb{E}_{\mathbf{M}}\left[\frac{1}{C}\mathbf{M}\mathbf{M}^\top-\frac{1}{C^2}\mathbf{M}\mathbf{1}\mathbf{1}^\top\mathbf{M}^\top\right]\left(\mathbf{W}^{(k)}\boldsymbol{\Sigma}-\mathbf{I}\right)\right\|_F\\
&=\sigma^2\left(1-\frac{1}{C}\right)\left\|\left(\mathbf{W}^{(k)}-\boldsymbol{\Sigma}^{-1}\right)\right\|_F.
\end{aligned}$$

Since all columns in $\mathbf{M}$ are sampled from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\mathbf{W}^{(k)}$ is assumed to be an isotropic matrix, it's obvious that $\mathcal{A}_2$ is also an isotropic matrix. It follows that

$$
\begin{aligned}
&\langle \mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1}, \nabla_{\mathbf{W}} L_{\text{CoT}} \rangle \\
&= \mathbb{E}_{\mathbf{M}} \left[\text{trace} \left(\mathbf{M} \left(\text{diag}(1/C) - \frac{1}{C^2} \mathbf{1}\mathbf{1}^\top \right) \mathbf{M}^\top \left(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \right) \mathbf{\Sigma} \left(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \right)^\top \right) \right] \\
&\overset{(a)}{=} \sigma^2 \text{trace} \Big(\mathbb{E}_{\mathbf{M}} \Big[\frac{1}{C} \mathbf{M}\mathbf{M}^\top \Big] \Big(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \Big) \Big(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \Big)^\top \\
&\quad - \mathbb{E}_{\mathbf{M}} \Big[\frac{1}{C^2} \mathbf{M}\mathbf{1}\mathbf{1}^\top \mathbf{M}^\top \Big] \Big(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \Big) \Big(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \Big)^\top \Big) \\
&= \sigma^2 \text{trace} \left(\left(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \right) \left(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \right)^\top - \frac{1}{C} \left(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \right) \left(\mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \right)^\top \right) \\
&= \sigma^2 \left(1 - \frac{1}{C} \right) \| \mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \|_F^2.
\end{aligned}
$$

where equation (a) follows from the assumption that $\mathbf{\Sigma} = \sigma^2 \mathbf{I}$.

Let $\gamma = \sigma^2(1 - 1/C)$. Then,

$$
\begin{aligned}
\| \mathbf{W}^{(k+1)} - \mathbf{\Sigma}^{-1} \|_F^2 \leq & \| \mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \|_F^2 - 2\gamma\eta \| \mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \|_F^2 + \eta^2 \gamma^2 \| \mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \|_F^2 \\
\leq & (1 - \gamma\eta)^2 \| \mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \|_F^2
\end{aligned}
$$

Select step size such that $(1 - \gamma\eta)^2 \leq 1$ and let $c := (1 - \gamma\eta)^2$, we obtain

$$
\| \mathbf{W}^{(k)} - \mathbf{\Sigma}^{-1} \|_F^2 \leq c^k \| \mathbf{W}^{(0)} - \mathbf{\Sigma}^{-1} \|_F^2.
$$

E Auxiliary Lemmas

Lemma 5 (Stein's Lemma)**.** *Let $X \in \mathbb{R}^d$ be a random vector with*

$$
X \sim \mathcal{N}(\mu, \Sigma),
$$

where $\mu \in \mathbb{R}^d$ and $\Sigma \in \mathbb{R}^{d \times d}$ is a positive definite matrix. Let $f : \mathbb{R}^d \to \mathbb{R}^k$ be a continuously differentiable function such that

$$
\mathbb{E}\big[\| f(X) \| \big] < \infty \quad \textit{and} \quad \mathbb{E}\big[\| \nabla f(X) \|_F \big] < \infty,
$$

where $\| \cdot \|$ denotes the Euclidean norm in $\mathbb{R}^k$ and $\| \cdot \|_F$ is the Frobenius norm. Then, the following identity holds:

$$
\mathbb{E}\Big[(X - \mu)\, f(X)^T \Big] = \Sigma\, \mathbb{E}\Big[\nabla f(X) \Big],
$$

where $\nabla f(X)$ is the $k \times d$ Jacobian matrix of f evaluated at X.