Towards the Effect of Examples on In-Context Learning: A Theoretical Case Study

Anonymous Author(s)
Affiliation
Address
email

Abstract

In-context learning (ICL) has emerged as a powerful ability for large language models (LLMs) to adapt to new tasks by leveraging a few (demonstration) examples. Despite its effectiveness, the mechanism behind ICL remains underexplored. This paper uses a Bayesian framework to investigate how ICL integrates pre-training knowledge and examples for binary classification. In particular, we introduce a probabilistic model extending from the Gaussian mixture model to exactly quantify the impact of pre-training knowledge, label frequency, and label noise on the prediction accuracy. Based on our analysis, when the pre-training knowledge contradicts the knowledge in the examples, whether ICL prediction relies more on the pre-training knowledge or the examples depends on the number of examples. In addition, the label frequency and label noise of the examples both affect the accuracy of the ICL prediction, where the minor class has a lower accuracy and how the label error impacts the accuracy is determined by the specific error rate of the two classes. Extensive simulations are conducted to verify the correctness of the theoretical results, and real-data experiments also align with the theoretical insights. Our work reveals the dual role of pre-training knowledge and examples in ICL, offering a deeper understanding of LLMs' behaviors in classification tasks.

1 Introduction

Large language models (LLMs) have revolutionized various fields, such as GitHub Copilot for software development, Microsoft 365 Copilot to embrace productivity, and medical applications such as Med-Palm [1]. A particularly intriguing ability of LLMs is in-context learning, where LLMs can adapt to new tasks only using a few examples at the inference stage without changing the model parameters. As ICL enhances the predictive performance of LLMs, various existing literature attempts to understand and quantify such a superiority [2, 3, 4].

During the ICL process, LLMs typically demonstrate two key abilities [5]: retrieving knowledge from the pre-training data and learning from the examples in the prompt. Understanding how pre-training knowledge and specific examples interact during the inference stage is crucial, especially given the complex dynamics observed in practical applications. For instance, existing literature [6] conducts various empirical evaluations to study ICL regarding the example size, demonstration order, prompt templates, etc. Meanwhile, theoretical studies [7, 8, 9, 10, 11, 5] have explored the underlying mechanisms of ICL from various perspectives, including Bayesian approaches and gradient descent, primarily focusing on linear regression models.

However, the existing literature [1] is insufficient to understand the behavior of ICL, especially for classification tasks. First, previous works cannot draw a consensus on certain behaviors of ICL. For example, in [6], it is empirically observed that injecting random noise to the example labels

[1]Related works are discussed in section G.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

does not hurt the ICL performance. They conjecture that the robustness of ICL against the noise is because the pre-training knowledge dominates ICL. On the other hand, based on [5], when taking a large number of examples (i.e., a large example size), the ICL will favor the knowledge provided by the examples. However, there is no systematic understanding of the role of label quality, the difference between pre-training and example knowledge, as well as the example size. Second, existing theoretical frameworks may fail to explain the observed behaviors in classification tasks. For instance, the balance of the example size in different classes matters in classification, while there is no such concept in regression.

The above gaps drive the need for a theoretical exploration of how LLMs utilize pre-training knowledge and specific examples in ICL in classification scenarios. In particular, we aim to answer: **How do LLMs make predictions in classification tasks using their pre-training knowledge and examples?**

This work aims to explore the above question by conducting an exact theoretical analysis in a binary classification task. Our contributions are summarized as follows:

- We leverage Bayesian analysis to exactly quantify the ICL performance (measured by prediction accuracy). When the example size is small, the pre-training knowledge will dominate ICL, and when the example size is large, the ICL prediction mainly relies on the examples. Built upon this finding, we further study the ICL performance under different scenarios such as label noise and imbalanced examples mentioned above, and contradiction in pre-training and example knowledge. Extensive simulations and real-data analysis are conducted to support our theoretical insights.

  Technically, to perform the analysis, we assume all examples in the pre-training are selected independently, and examine the posterior distribution of the parameters of the data generation model with two distinct types of priors: one from the pre-training data and the other from the examples. A central challenge in the analysis lies in the formulation and integration of these two priors into a single coherent posterior for ICL prediction. Our result successfully accounts for both the label distributions and the conditional output distribution within each class.
- When conducting simulations to verify the above theoretical insights, we surprisingly reveal another counter-intuitive behavior when the examples are not selected independently: We fix exactly 50% positive labels in each prompt in pre-training and provide only positive examples in the test prompt, then the ICL prediction is a firm negative. We provide an intuitive explanation and theoretical justification to explain this behavior. This finding can help understand how LLMs consider dependency among tokens/sequences.

2 Classification Analysis via Bayesian

To analyze the ICL performance, we first introduce the model and data assumptions in Section 2.1, then derive the ICL accuracy under general situations in Section 2.2. We finally examine how examples influence ICL under specific demonstration scenarios (Section 2.3).

2.1 Setups

To perform the exact analysis of the ICL prediction, in this subsection, we introduce the pre-training-inference paradigm and impose some assumptions on the data generation distribution.

Pre-training. For the pre-training data, inspired by [2, 4, 5, 12], we form the prompts in the form of $((x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k), (x_{query}, y_{query}))$ and the target is to predict the label y_{query}, i.e., performing ICL using the examples $\{(x_i, y_i)\}_{i\in[k]}$, where $x_i \in \mathbb{R}^m$ and $y_i \in \{-1, +1\}$. To simplify the notation, we use (x, y) and $\{(x_i, y_i)\}$ to refer to (x_{query}, y_{query}) and $\{(x_i, y_i)\}_{i\in[k]}$ respectively when no confusion arises. All (x_i, y_i)s and x in the same prompt are in general sampled from the same distribution, and an exception considering label noise will be described in detail later in Section 2.3. In different prompts, the sampling distribution may vary. We assume that all examples in the demonstration are independent, typically selected randomly from a prompt set [6]. A discussion on the case where the examples are not independent is provided in Section E for a comprehensive analysis.

Denote the pre-trained LLM as M. Without loss of generality, we assume that M learns the exact distribution of the pre-training data and makes predictions based on the pre-training knowledge, i.e., model output $M(x)$ follows the pre-training distribution. This assumption is supported by the capabilities of LLMs and is commonly used in existing research [7, 13].

Inference. At the inference stage, we perform ICL of a test input x given the examples to predict its corresponding test label y, i.e. $\hat{y}_{ICL} = M((\{(x_i, y_i)\}, x))$. Our goal is to study the effect of the pre-training data and the examples on the distribution of $\hat{y}_{ICL}$. To simplify the notation, we denote $P(\hat{y}_{ICL} = s)$ as $P(y = s|x, \{(x_i, y_i)\}, M)$ for $s \in \{-1, +1\}$. Unless otherwise stated, $P(\cdot|..., M)$ represents the meaning of "conditional on the pre-training knowledge".

Data generation process. We mainly follow the idea of Bayesian inference to form the assumptions. Bayesian inference is a well-established theoretical method that has demonstrated its effectiveness in explaining the behavior of LLMs as shown in existing research [7, 8]. To perform Bayesian inference, we impose a prior distribution on the parameters in the data generation process and then use data and the prior distribution together to derive a posterior distribution of the parameters. The idea of prior distribution is widely used in uncertainty quantification in real applications such as various medical studies [14], and is justified by axioms of decision theory [15].

The following two assumptions are imposed in our main study. We consider the pre-training data, example, and test data to be in the same distribution family but with different parameters. Assumption 1 describes how to generate a pair of (x, y) given a specific set of parameters, and Assumption 2 explains how the parameters differ among datasets.

Assumption 1 (Generate (x, y))**.** *Assume $x \in \mathbb{R}^m$ and $y \in \{-1, +1\}$. Given parameters $(\theta_+, \theta_-, \pi, p_+, p_-)$, to generate (x, y), y is first generated from a Bernoulli distribution with π, i.e. $P(y = +1) = \pi$ and $P(y = -1) = 1 - \pi$, then x is generated from a class-wise input distribution accordingly. Given $y = +1$, x follows a Gaussian distribution $N(\theta_+, \sigma_+^2 I)$ with probability p_+ and sample from $N(\theta_-, \sigma_-^2 I)$ with probability $1 - p_+$; given $y = -1$, x is sampled from a Gaussian distribution $N(\theta_-, \sigma_-^2 I)$ with probability p_- and sample from $N(\theta_+, \sigma_+^2 I)$ with probability $1 - p_-$. In addition, the examples are independent with each other.*

Assumption 1 follows the standard Gaussian mixture design for theoretical analysis in classification, e.g., [16, 17, 18]. We further consider "label noise": When $y = +1$, the corresponding x can be from either of the two clusters. When $p_+ = p_- = 1$, it means that there is no label noise.

Assumption 2 (Parameters)**.** *The parameter distributions for pre-training and the inference stage as as follows:*

- *Pre-train: $\theta_+ \sim N(\theta_M, \sigma_M^2 I_m), \theta_- \sim N(-\theta_M, \sigma_M^2 I_m)$; $p_+ = p_- = 1$; $\pi \sim Beta(1, 1)$.*
- *Examples: $\theta_+ \sim N(\theta_+^e, \sigma_{e+}^2 I_m), \theta_- \sim N(\theta_-^e, \sigma_{e-}^2 I_m)$; $p_+^e, p_-^e \in [0, 1]$, $\pi \in [0, 1]$.*
- *Test data (x, y): $p_+^t = p_-^t = 1$. Examples and the test data in the same prompt share the same realization of (θ_+, θ_-).*

In pre-training, all (x_i, y_i)s and (x, y) in the same prompt are conditionally independent and share the same parameters. At the inference stage, the examples are conditionally independently sampled given the parameters and may incur label noise. For the test data (x, y), while it shares the same (θ_+, θ_-) with the examples in the prompt, we do not further consider label noise in the test data. The proportion $P(y = +1)$ is not considered in the test data because the later accuracy analysis is performed on $y = +1$ and $y = -1$ separately.

Assumption 2 aligns with the common scenarios of ICL, i.e., the pre-training distribution and the example distribution at the inference stage can differ. In pre-training, we take $p_+ = p_- = 1$ to simplify the derivation. In this case, there is no label noise, and the misclassification of the Bayes classifier is only caused by the overlap of the two Gaussian clusters in the distribution. At the inference stage, the examples may have a distribution shift compared to the pre-training data, and we also consider potential label noise in the examples.

2.2 ICL Decision and Prediction Accuracy

To compute the ICL prediction accuracy, we first derive the posterior distribution of the parameters $(\theta_+, \theta_-, \pi)$ given the examples $\{(x_i, y_i)\}$ and the pre-training knowledge of θ_M, and then use $(\theta_+, \theta_-, \pi)$ to figure out the ICL accuracy.

Posterior of parameters. Our goal is to compute the posterior distribution of θ_+, θ_-, π given examples $(x_1, y_1), ..., (x_k, y_k)$. Recall that in Assumption 2, $p_+ = p_- = 1$ in the pre-training stage, and the p_+^e and p_-^e in the inference stage can be some values in $[0, 1]$ if label noise occurs. Denote $\#(y_i = +1)$ and $\#(y_i = -1)$ as the number of examples with positive/negative labels respectively (after possible flips if p_+^e or p_-^e is less than 1). The following lemma presents the posterior distribution of π, θ_+, θ_-.

Lemma 1. *Under Assumption 1 and Assumption 2, the posterior distribution of π, θ_+, θ_- satisfies*

$$P(\pi|\{(x_i, y_i)\}_{i\in[k]}, M) \propto \pi^{\#(y_i=+1)}(1-\pi)^{\#(y_i=-1)},$$

$$\theta_+ \sim N\left(\frac{\sigma_+^2\theta_M + \sigma_M^2\sum_{y_i=+1}x_i}{\sigma_+^2 + \#(y_i=+1)\sigma_M^2}, \frac{\sigma_+^2\sigma_M^2}{\sigma_+^2 + \#(y_i=+1)\sigma_M^2}I\right) \triangleq N(\hat{\theta}_+, \sigma_{\theta_+}^2 I),$$

and

$$\theta_- \sim N\left(\frac{\sigma_M^2\sum_{y_i=-1}x_i - \sigma_-^2\theta_M}{\sigma_-^2 + \#(y_i=-1)\sigma_M^2}, \frac{\sigma_M^2\sigma_-^2}{\sigma_-^2 + \#(y_i=-1)\sigma_M^2}I\right) \triangleq N(\hat{\theta}_-, \sigma_{\theta_-}^2 I).$$

The proof of Lemma 1 can be found in Section B.1. In short, since the examples $\{(x_i, y_i)\}$ are given, we can directly write the likelihood for $(\pi, \theta_+, \theta_-)$ to derive the corresponding posterior distributions.

ICL decision. Given Lemma 1, denoting $z_k = (\#(y_i = -1) + 1)/(\#(y_i = +1) + 1)$, the following lemma shows the ICL decision boundary for the test data:

Lemma 2. *Under Assumption 1 and Assumption 2, the probability of $y = +1/-1$ is as follows*

$$\begin{aligned}
P(y=+1|x, \{(x_i,y_i)\}, M) &= \frac{P(x, y=+1|\{(x_i,y_i)\}, M)}{P(x, y+=1|\{(x_i,y_i)\}, M) + P(x, y=-1|\{(x_i,y_i)\}, M)} \\
&= \frac{\frac{\#(y_i=+1)+1}{k+2}N(x)}{\frac{\#(y_i=+1)+1}{k+2}N(x) + \frac{\#(y_i=-1)+1}{k+2}}, \\
P(y=-1|x, \{(x_i,y_i)\}, M) &= \frac{P(x, y=-1|\{(x_i,y_i)\}, M)}{P(x, y+=1|\{(x_i,y_i)\}, M) + P(x, y=-1|\{(x_i,y_i)\}, M)} \\
&= \frac{\frac{\#(y_i=-1)+1}{k+2}}{\frac{\#(y_i=+1)+1}{k+2}N(x) + \frac{\#(y_i=-1)+1}{k+2}},
\end{aligned}$$

where

$$N(x) = \left(\sqrt{\frac{\sigma_-^2 + \sigma_{\theta_-}^2}{\sigma_+^2 + \sigma_{\theta_+}^2}}\right)^m \exp\left[-\frac{(x-\hat{\theta}_+)^T(x-\hat{\theta}_+)}{2(\sigma_+^2 + \sigma_{\theta_+}^2)} + \frac{(x-\hat{\theta}_-)^T(x-\hat{\theta}_-)}{2(\sigma_-^2 + \sigma_{\theta_-}^2)}\right].$$

The decision boundary is $\hat{y}_{ICL} = 1(f_{ICL}(x) > 0)$, where $f_{ICL}(x) = N(x) - z_k$.

The proof of Lemma 2 can be found in Section B.2. When $(\pi, \theta_+, \theta_-)$ are fixed, given y, x follows a Gaussian distribution. When integrating over all possible $(\pi, \theta_+, \theta_-)$, the marginal distribution of x given y still follows a Gaussian distribution. Hence, (x, y) marginally follows a Gaussian mixture distribution, and the decision boundary can be further obtained.

From Lemma 2, we can see how the pre-training distribution (θ_M, σ_M^2) and examples $\{(x_i, y_i)\}$ impact $P(y = +1|x, \{(x_i, y_i)\}, M)$ and $P(y = -1|x, \{(x_i, y_i)\}, M)$, and further change the decision boundary correspondingly. The pre-training knowledge (θ_M, σ_M^2) and examples $\{(x_i, y_i)\}$ first determines $(\hat{\theta}_+, \hat{\theta}_-, \sigma_{\theta_+}^2, \sigma_{\theta_-}^2)$, the latter of which further determines the decision boundary. More details about the interplay of pre-training and examples under different scenarios will be provided in Section 2.3. Besides, a larger π will result in higher weights of positive component in the conditional probability as shown in Lemma 2, and may lead to a higher probability of classifying test input as +1, as formally stated in Proposition 2 in Section 2.3.

ICL Accuracy. After obtaining the decision boundary from Lemma 2, we finally provide the general formula of the ICL prediction accuracy. In the following, we consider a simplified scenario and derive the exact accuracy of ICL in Theorem 1.

Theorem 1. *Under Assumption 1 and Assumption 2, and further assume $\sigma_+^2 = \sigma_-^2 = \sigma^2$ and $k \to \infty$, we have the following probability of correct prediction for each class.*

$$P(correct|y=+1,\{(x_i,y_i)\},M) = 1 - \Phi\left(\frac{(\theta_+^e - \frac{1}{2}(\hat{\theta}_+ + \hat{\theta}_-))^T}{\sqrt{\sigma^2+\sigma_{e+}^2}}\frac{\hat{\theta}_- - \hat{\theta}_+}{\|\hat{\theta}_- - \hat{\theta}_+\|_2} + \frac{\sigma^2 \log z_k}{\sqrt{\sigma^2+\sigma_{e+}^2}\|\hat{\theta}_- - \hat{\theta}_+\|_2}\right),$$

$$P(correct|y=-1,\{(x_i,y_i)\},M) = \Phi\left(\frac{(\theta_-^e - \frac{1}{2}(\hat{\theta}_+ + \hat{\theta}_-))^T}{\sqrt{\sigma^2+\sigma_{e-}^2}}\frac{\hat{\theta}_- - \hat{\theta}_+}{\|\hat{\theta}_- - \hat{\theta}_+\|_2} + \frac{\sigma^2 \log z_k}{\sqrt{\sigma^2+\sigma_{e+}^2}\|\hat{\theta}_- - \hat{\theta}_+\|_2}\right),$$

where $\Phi(\cdot)$ is the cumulative distribution function of a standard Gaussian distribution.

To prove Theorem 1, we first obtain the marginal distribution of $x|y$ given the example distribution in Assumption 2 to remove internal parameters, denoted as $P(x|y=+1)$. Then the ICL performance, i.e., prediction accuracy, can be computed via $P(correct|y=+1,\{(x_i,y_i)\},M) = \int_{f_{ICL}(x)\geq 0} P(x|y=+1)dx$. Similar steps apply for $y=-1$. The detailed proof of Theorem 1 can be found in Section B.3.

Theorem 1 describes how the ICL performance is affected by the interplay between the pre-training knowledge and the examples. For example, in addition to the model parameters $(\theta_+^e, \sigma^2, \sigma_{e+}^2)$, $P(correct|y=+,\{(x_i,y_i)\},M)$ is further determined by $\hat{\theta}_- - \hat{\theta}_+$ and $\hat{\theta}_- + \hat{\theta}_+$. One key insight is that these two terms are mixtures of examples and pre-training knowledge. The example size k, the variance of data σ^2, as well as pre-training distribution σ_M^2 will also affect their exact formulas.

A final note is that, under Theorem 1, the decision boundary is a hyperplane, and one can integrate the above two probabilities. We also provide technical discussions when the decision boundary is not a hyperplane. In such a case, the boundary is a sphere, and the details can be found in Section A.

2.3 Different Demonstration Scenarios

In the following, we extend the above results to investigate how ICL is affected in specific situations. We consider contradicting knowledge, imbalanced examples, and label noise.

To simplify the analysis, we assume that in the inference stage, $\#(y_i=+1)=\pi$. Since $\#(y_i = +1)/k - \pi \to 0$ in k, the additional fluctuation in $\#(y_i=+1)$ does not affect the result.

Contradicting knowledge. In practical applications, it is possible that the examples exhibit different or even contradicting knowledge of the pre-training. To study this case, we compare $\theta_+^e = -\theta_M = -\theta_-^e$ and $\theta_+^e = \theta_M = -\theta_-^e$, i.e., the input distribution in examples is the opposite/same to that of pre-training distribution. The following result is obtained based on Theorem 1 in these scenarios:

Proposition 1 (Contradicting knowledge)**.** *Assume the conditions of Theorem 1 hold, and also assume $\sigma_{e+}^2, \sigma_{e-}^2 \to 0$, and $\pi = 0.5$ at the inference stage. Then when $k\sigma_M^2 \ll \sigma^2$, i.e., insufficient example size,*

$$\begin{aligned} & P(correct|y=+1,\theta_+^e=\theta_M=-\theta_-^e) - P(correct|y=+1,\theta_+^e=-\theta_M=-\theta_-^e) \\ \to\ & \Phi\left(\frac{\|\theta_M\|}{\sqrt{\sigma^2}}\right) - \Phi\left(-\frac{\|\theta_M\|}{\sqrt{\sigma^2}}\right) > 0. \end{aligned}$$

When $k\sigma_M^2 \gg \sigma^2$, i.e., sufficient example size, both $P(correct|y=+1,\theta_+^e=-\theta_M=-\theta_-^e)$ and $P(correct|y=+1,\theta_+^e=\theta_M=-\theta_-^e)$ converges to $1-\Phi\left(-\|\theta_M\|/\sqrt{\sigma^2}\right)$.

The accuracy of $y=-1$ exhibits a similar behavior.

The proof of Proposition 1 can be found in Section B.4. We mainly follow the result in Theorem 1 and calculate the probabilities under the specific scenario.

There are two observations in Proposition 1. First, when there are no enough examples and the pre-training knowledge contradicts to the knowledge in the examples and the test data, there is an obvious drop in ICL performance compared to the case when the knowledge matches. Second, when there are enough examples, the knowledge from the examples will dominate, and ICL performance of contradicting knowledge converges to that of matching knowledge.

Imbalanced examples. In the following, we consider the case where the two classes are imbalanced at the inference stage, i.e. $\pi \neq 0.5$. In this case, the value of π will impact the ICL prediction.

Proposition 2 (Imbalanced examples)**.** *Under Assumption 1 and Assumption 2, assume σ^2 and σ_M^2 are constants, $\pi k \to \infty$ and $(1-\pi)k \to \infty$, then*

$$f_{ICL}(x) \to \exp\left[-\frac{(x-\hat{\theta}_+)^T(x-\hat{\theta}_+)}{2\sigma^2} + \frac{(x-\hat{\theta}_-)^T(x-\hat{\theta}_-)}{2\sigma^2}\right] - \frac{1-\pi}{\pi}.$$

When $\pi \to 0$, $P(correct|y=+1, \{(x_i, y_i)\}, M) \to 0$.

The proof of Proposition 2 is in Section B.5. In short, we follow Lemma 2 to obtain the decision boundary. Then, we repeat the steps of Theorem 1 to obtain the conclusion.

Based on Proposition 2, when the examples for one class are much fewer than the other class, ICL performance for the minor class will significantly drop. In addition, the parameter π learned from pre-training will be overlooked.

Label noise. It is common that there exist label noises in the examples for ICL. For example, an example x_i sampled from $N(\theta_-^e, \sigma_{e-}^2)$ may be labeled as $+1$. Therefore, we change the value of p_+^e and p_-^e in the examples to see how these changes affect the ICL performance, the result of which is summarized as follows:

Proposition 3 (Label noise)**.** *Under the conditions of Theorem 1, assume σ^2 and σ_M^2 are constants, $\theta_M = \theta_+^e$ and $\theta_M = -\theta_-^e$, and $\sigma_{e+}^2, \sigma_{e-}^2 \to 0$. Also assume $\pi = 0.5$ at the inference stage. When $1 - p_+^e - p_-^e < 0$, and $k \to \infty$, $P(correct|y=+1, p_+^e, p_-^e)$ increases in p_+^e, and $P(correct|y=-1, p_+^e, p_-^e)$ increases in p_-^e.*

The proof of Proposition 3 can be found in Section B.6 and is a direct extension from Theorem 1.

In Proposition 3, recall that $p_+^e = p_-^e = 1$ implies no random flip on the example labels. Intuitively, when keeping $p_-^e = 1$ and decreasing p_+^e, the positive class becomes a mixture of two Gaussian distributions. In this case, $\hat{\theta}_+$ is closer to zero, and the decision boundary will shift towards $-\theta_M$. Therefore, it is more likely that ICL predicts a negative label for x, which aligns with the change in $P(correct|y=+1, p_+^e, p_-^e)$ and $P(correct|y=-1, p_+^e, p_-^e)$ in Proposition 3. When $1-p_+^e-p_-^e = 0$, the decision boundary set $\{f_{ICL}(x) > 0\}$ will degenerate to either $\emptyset$ or full space. Therefore, in these special cases, the positive accuracy and negative accuracy will be either (0,1), (1,0), or (0.5,0.5).

Due to the page limit, we postpone all the simulations and real-data experiments to Appendix D. The simulation results for the next section can also be found in Appendix E

2.4 Mean Reversion

When the fraction of positive and negative is fixed in the pre-training, we notice an interesting phenomenon “Mean Reversion".

Theorem 2 (**Mean Reversion**, informal version of Theorem 3)**.** *Let $frac$ denote the fraction of $+1$ among the set of labels in the pre-training set. Under some mild conditions, assume in each prompt in pre-training, $frac$ is always a fixed π, then in the testing prompt: (1) If $\#(y_i = +1)/k < \pi$, then the prediction of x is $+1$. (2) If $\#(y_i = +1)/k > \pi$, then the prediction of x is -1.*

We direct the reader into Appendix B.7 for the formal statement and detailed proof. Theorem 4 indicates that the conditional probability of y is determined by the fraction of labels within the pre-training set and the examples during inference, in addition to the inputs. A direct corollary is that when the fraction of $y_i = +1$ is fixed as 0.5 during the pre-training, and all y_i are negative in the inference stage, the prediction for y is always positive.

3 Conclusion

In this paper, we analyze the behavior of ICL in a binary classification model. We study the ICL performance under different scenarios, including contradicting knowledge, imbalanced examples, and label noise. In addition to the above analysis in which we assume examples are independently chosen in pre-training, we also find out a counter-intuitive phenomenon when the examples are selected in a dependent way. When fixing the number of positive labels and negative labels in the prompt, the ICL prediction behaves in a mean-reversion manner. We believe that our observations and theoretical results can provide deep insights into understanding ICL. A future direction could be to relax the conditions in this paper and consider more general data distributions.

References

[1] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

[2] Ruiqi Zhang, Spencer Frei, and Peter L Bartlett. Trained transformers learn linear models in-context. *arXiv preprint arXiv:2306.09927*, 2023.

[3] Yingqian Cui, Jie Ren, Pengfei He, Jiliang Tang, and Yue Xing. Superiority of multi-head attention in in-context linear regression. *arXiv preprint arXiv:2401.17426*, 2024.

[4] Yu Huang, Yuan Cheng, and Yingbin Liang. In-context convergence of transformers. *arXiv preprint arXiv:2310.05249*, 2023.

[5] Ziqian Lin and Kangwook Lee. Dual operating modes of in-context learning. *arXiv preprint arXiv:2402.18819*, 2024.

[6] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

[7] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*, 2021.

[8] Xinyi Wang, Wanrong Zhu, Michael Saxon, Mark Steyvers, and William Yang Wang. Large language models are implicitly topic models: Explaining and finding good demonstrations for in-context learning. In *Workshop on Efficient Systems for Foundation Models@ ICML2023*, 2023.

[9] Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022.

[10] Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models. *arXiv preprint arXiv:2211.15661*, 2022.

[11] Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pages 35151–35174. PMLR, 2023.

[12] Chi Han, Ziqi Wang, Han Zhao, and Heng Ji. Explaining emergent in-context learning as kernel regression. 2023.

[13] Kabir Ahuja, Madhur Panwar, and Navin Goyal. In-context learning through the bayesian prism. *arXiv preprint arXiv:2306.04891*, 2023.

[14] Wenqi Shi, Ran Xu, Yuchen Zhuang, Yue Yu, Jieyu Zhang, Hang Wu, Yuanda Zhu, Joyce C Ho, Carl Yang, and May Dongmei Wang. Ehragent: Code empowers large language models for few-shot complex tabular reasoning on electronic health records. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*, 2024.

[15] James O Berger. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 2013.

[16] Chen Dan, Yuting Wei, and Pradeep Ravikumar. Sharp statistical guaratees for adversarially robust gaussian classification. In *International Conference on Machine Learning*, pages 2345–2355. PMLR, 2020.

[17] Hossein Taheri, Ramtin Pedarsani, and Christos Thrampoulidis. Asymptotic behavior of adversarial training in binary linear classification. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pages 127–132. IEEE, 2022.

[18] Ke Wang and Christos Thrampoulidis. Benign overfitting in binary classification of gaussian mixtures. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4030–4034. IEEE, 2021.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[20] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[21] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

[22] Jeffrey L Elman. Learning and development in neural networks: The importance of starting small. *Cognition*, 48(1):71–99, 1993.

[23] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR, 2023.

[24] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[25] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[26] Xinyi Wang, Wanrong Zhu, Michael Saxon, Mark Steyvers, and William Yang Wang. Large language models are latent variable models: Explaining and finding good demonstrations for in-context learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[27] Yiming Zhang, Shi Feng, and Chenhao Tan. Active example selection for in-context learning. *arXiv preprint arXiv:2211.04486*, 2022.

[28] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[29] Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. *arXiv preprint arXiv:2310.15916*, 2023.

[30] Eric Todd, Millicent L Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. Function vectors in large language models. *arXiv preprint arXiv:2310.15213*, 2023.

[31] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*, 2021.

[32] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*, 2021.

[33] Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering. *arXiv preprint arXiv:2212.10375*, 2022.

[34] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. PMLR, 2021.

[35] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[36] Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. Why can gpt learn in-context? language models implicitly perform gradient descent as meta-optimizers. *arXiv preprint arXiv:2212.10559*, 2022.

[37] Kwangjun Ahn, Xiang Cheng, Hadi Daneshmand, and Suvrit Sra. Transformers learn to implement preconditioned gradient descent for in-context learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[38] Arvind Mahankali, Tatsunori B Hashimoto, and Tengyu Ma. One step of gradient descent is provably the optimal in-context learner with one layer of linear self-attention. *arXiv preprint arXiv:2307.03576*, 2023.

[39] Hong Jun Jeon, Jason D Lee, Qi Lei, and Benjamin Van Roy. An information-theoretic analysis of in-context learning. *arXiv preprint arXiv:2401.15530*, 2024.

[40] Jane Pan. *What in-context learning "learns" in-context: Disentangling task recognition and task learning*. PhD thesis, Princeton University, 2023.

The structure of the appendix is as follows. In Section A, we provide the discussion when the decision boundary in Lemma 2 is not a hyperplane. Section B collects the proof for all lemmas and theorems in the main content. Section C describes the simulation setups, and Section F includes additional experiment results corresponding to Section D and Section E for both independent and dependent scenarios. Section D provides detailed experiments and results, including simulations and real-data experiments. SectionE provides a detailed discussion on the mean reversion phenomenon both theoretically and empirically. Section G discusses the related works.

A Technical Discussion when $\sigma_+^2 + \sigma_{\theta_+}^2 \neq \sigma_-^2 + \sigma_{\theta_-}^2$

General scenario In Theorem 1 and the propositions in Section 2.3, our assumptions aim to simplify the analysis so that the decision boundary is a hyperplane. When the examples are provided such that $\sigma_{\theta_+}^2 \neq \sigma_{\theta_-}^2$, the general intuition holds, but the decision boundary changes from a hyperplane into a sphere. To be specific, based on Lemma 2, we have

$$N(x) = \left(\sqrt{\frac{\sigma_-^2 + \sigma_{\theta_-}^2}{\sigma_+^2 + \sigma_{\theta_+}^2}}\right)^m \exp\left[-\frac{(x-\hat{\theta}_+)^T(x-\hat{\theta}_+)}{2(\sigma_+^2 + \sigma_{\theta_+}^2)} + \frac{(x-\hat{\theta}_-)^T(x-\hat{\theta}_-)}{2(\sigma_-^2 + \sigma_{\theta_-}^2)}\right].$$

Then for some constant a,

$$\begin{aligned} & N(x) > a \\ \Leftrightarrow \quad & -\frac{(x-\hat{\theta}_+)^T(x-\hat{\theta}_+)}{2(\sigma_+^2 + \sigma_{\theta_+}^2)} + \frac{(x-\hat{\theta}_-)^T(x-\hat{\theta}_-)}{2(\sigma_-^2 + \sigma_{\theta_-}^2)} > \log(a) - m\log\left(\sqrt{\frac{\sigma_-^2 + \sigma_{\theta_-}^2}{\sigma_+^2 + \sigma_{\theta_+}^2}}\right), \end{aligned}$$

where the decision boundary

$$-\frac{(x-\hat{\theta}_+)^T(x-\hat{\theta}_+)}{2(\sigma_+^2 + \sigma_{\theta_+}^2)} + \frac{(x-\hat{\theta}_-)^T(x-\hat{\theta}_-)}{2(\sigma_-^2 + \sigma_{\theta_-}^2)} = \log(a) - m\log\left(\sqrt{\frac{\sigma_-^2 + \sigma_{\theta_-}^2}{\sigma_+^2 + \sigma_{\theta_+}^2}}\right)$$

is a sphere.

In such a case, in Theorem 1, for

$$P(correct|y=+1) = \int_{f_{ICL}(x)\geq 0} P(x|y=+1)dx,$$

instead of directly using Φ to represent the probability, we use the noncentral Chi-square distribution to write the probability. The following is the definition of noncentral Chi-square distribution.

Definition 1 (Noncentral Chi-square distribution[2])**.** *Let $(X_1, X_2, \ldots, X_k)$ be k independent and normally distributed random variables with means μ_i and unit variances. Then the random variable*

$$\sum_{i=1}^k X_i^2$$

is distributed according to the noncentral chi-square distribution. It has two parameters: k which specifies the number of degrees of freedom and λ which is related to the mean of X_is by

$$\lambda = \sum_{i=1}^k \mu_i^2.$$

Case when $\sigma_+^2 = \sigma_-^2$ When $\sigma_+^2 = \sigma_-^2$, asymptotically, when $k \to \infty$, the difference between $\sigma_{\theta_-}^2$ and $\sigma_{\theta_+}^2$ does not hurt the decision boundary. To explain this, based on the formula of $\sigma_{\theta_-}^2$ and $\sigma_{\theta_+}^2$, both of them are in $O(1/k)$, which quickly diminishes to zero in k. On the other hand, for the other terms in the decision boundary, e.g., $\hat{\theta}_+$ and $\hat{\theta}_-$ in $N(x)$, they converge to their expectation in a rate of $O(1/\sqrt{k})$. As a result, the effect of $\sigma_{\theta_-}^2$ and $\sigma_{\theta_+}^2$ are negligible compared to the other quantities in the decision boundary formula.

[2] https://en.wikipedia.org/wiki/Noncentral_chi-squared_distribution

B Proofs

B.1 Proof of Lemma 1

Proof of Lemma 1. Given the prior distribution of θ_-, θ_+, π and the data $\{(x_i, y_i)\}$, the likelihood becomes

$$
\begin{aligned}
& P(\theta_+, \theta_-, \pi | (x_1, y_1), ..., (x_k, y_k), M) \\
& \propto P((x_1, y_1), ..., (x_k, y_k)|\theta_+, \theta_-, \pi, M) P(\theta_+, \theta_-, \pi | M) \\
& = P(\theta_+, \theta_-, \pi) \prod_{i=1}^{k} P((x_i, y_i)|\theta_+, \theta_-, \pi) \text{ (Omit } M \text{ for simplicity)} \\
& = P(\theta_+, \theta_-, \pi) \\
& \qquad \cdot \prod_{y_i=+1} \pi \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x_i - \theta_+)^T (x_i - \theta_+)\right] \\
& \qquad \cdot \prod_{y_i=-1} (1-\pi) \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x_i - \theta_+)^T (x_i - \theta_+)\right] \\
& = [\pi^{\#(y_i=+1)}(1-\pi)^{\#(y_i=-1)}] P(\theta_+, \theta_-) \prod_{y_i=+1} \pi \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x_i - \theta_+)^T (x_i - \theta_+)\right] \\
& \qquad \cdot \prod_{y_i=-1} (1-\pi) \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x_i - \theta_+)^T (x_i - \theta_+)\right]
\end{aligned}
\quad . \tag{1}
$$

Posterior of π. Since all parameters are independent, we can obtain the posterior distribution of π as

$$P(\pi|\{(x_i, y_i)\}, M) \propto P(\pi|M)\pi^{\#(y_i=+1)}(1-\pi)^{\#(y_i=-1)} \propto \pi^{\#(y_i=+1)}(1-\pi)^{\#(y_i=-1)}.$$

Therefore, the posterior of π is $Beta(\#(y_i = +1) + 1, \#(y_i = -1)k + 1)$.

Posterior of θ_+, θ_-. The likelihood of θ_+ satisfies

$$
\begin{aligned}
P(\theta_+|\{(x_i, y_i)\}|M) &\propto P(\theta_+|M) \prod_{y_i=+1} \pi \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x_i - \theta_+)^T (x_i - \theta_+)\right] \\
&\propto \exp\left[-\frac{1}{2\sigma_M^2}(\theta_+ - \theta_M)^T(\theta_+ - \theta_M) - \frac{1}{2\sigma_+^2}\sum_{y_i=+1}(x_i - \theta_+)^T(x_i - \theta_+)\right],
\end{aligned}
$$

which means that the posterior of θ_+ follows a Gaussian distribution, i.e.

$$\theta_+ \sim N\left(\frac{\sigma_+^2\theta_M + \sigma_M^2\sum_{y_i=+1} x_i}{\sigma_+^2 + \#(y_i=+1)\sigma_M^2}, \frac{\sigma_+^2\sigma_M^2}{\sigma_+^2 + \#(y_i=+1)\sigma_M^2} I\right) = N(\hat{\theta}_+, \sigma_{\theta_+}^2 I).$$

Similarly, the posterior of θ_- follows

$$\theta_- \sim N\left(\frac{\sigma_M^2\sum_{y_i=-1} x_i - \sigma_-^2\theta_M}{\sigma_-^2 + \#(y_i=-1)\sigma_M^2}, \frac{\sigma_M^2\sigma_-^2}{\sigma_-^2 + \#(y_i=-1)\sigma_M^2} I\right) = N(\hat{\theta}_-, \sigma_{\theta_-}^2 I).$$

□

B.2 Proof of Lemma 2

Proof of Lemma 2. Given $p_+ = p_- = 1$ and the posterior of θ_+, θ_-, π, we obtain that

$$P(x, y = +1|\{(x_i, y_i)\}, M)$$

$$
\begin{aligned}
&= \int_\pi \int_{\theta_+,\theta_-} \pi \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x-\theta_+)^T(x-\theta_+)\right] \\
&\qquad \cdot P(\pi|\{(x_i,y_i)\},M)P(\theta_+|\{(x_i,y_i)\},M)P(\theta_-|\{(x_i,y_i)\},M)d\pi d\theta_+ d\theta_- \\
&= \int_\pi \int_{\theta_+} \pi \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x-\theta_+)^T(x-\theta_+)\right] \\
&\qquad \cdot P(\pi|\{(x_i,y_i)\},M)P(\theta_+|\{(x_i,y_i)\},M)d\pi d\theta_+ \\
&= \int_\pi \int_{\theta_+} \pi \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m \left(\sqrt{2\pi\sigma_{\theta_+}^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x-\theta_+)^T(x-\theta_+)\right] \\
&\qquad \cdot P(\pi|\{(x_i,y_i)\},M) \exp\left[-\frac{1}{2\sigma_{\theta_+}^2}(\theta_+-\hat\theta_+)^T(\theta_+-\hat\theta_+)\right] d\pi d\theta_+ \\
&= \frac{\#(y_i=+1)+1}{k+2} \int_{\theta_+} \frac{1}{\left(\sqrt{2\pi\sigma_+^2}\right)^m \left(\sqrt{2\pi\sigma_{\theta_+}^2}\right)^m} \exp\left[-\frac{1}{2\sigma_+^2}(x-\theta_+)^T(x-\theta_+)\right] \\
&\qquad \cdot \exp\left[-\frac{1}{2\sigma_{\theta_+}^2}(\theta_+-\hat\theta_+)^T(\theta_+-\hat\theta_+)\right] d\theta_+ \\
&= \frac{\#(y_i=+1)+1}{k+2} \frac{1}{\left(\sqrt{2\pi(\sigma_+^2+\sigma_{\theta_+}^2)}\right)^m} \exp\left[-\frac{1}{2(\sigma_+^2+\sigma_{\theta_+}^2)}(x-\hat\theta_+)^T(x-\hat\theta_+)\right].
\end{aligned}
$$

Similarly, we have

$$
P(x,y=-1|\{(x_i,y_i)\},M) = \frac{\#(y_i=-1)+1}{k+2} \frac{1}{\left(\sqrt{2\pi(\sigma_-^2+\sigma_{\theta_-}^2)}\right)^m} \exp\left[-\frac{(x-\hat\theta_-)^T(x-\hat\theta_-)}{2(\sigma_-^2+\sigma_{\theta_-}^2)}\right].
$$

385 Then we can obtain the predicted probability and decision boundary.

$$
\begin{aligned}
P(y=+1|x,\{(x_i,y_i)\},M) &= \frac{P(x,y=+1|\{(x_i,y_i)\},M)}{P(x,y+=1|\{(x_i,y_i)\},M)+P(x,y=-1|\{(x_i,y_i)\},M)} \\
&= \frac{\frac{\#(y_i=+1)+1}{k+2}N(x)}{\frac{\#(y_i=+1)+1}{k+2}N(x)+\frac{\#(y_i=-1)+1}{k+2}}, \\
P(y=-1|x,\{(x_i,y_i)\},M) &= \frac{P(x,y=-1|\{(x_i,y_i)\},M)}{P(x,y+=1|\{(x_i,y_i)\},M)+P(x,y=-1|\{(x_i,y_i)\},M)} \\
&= \frac{\frac{\#(y_i=-1)+1}{k+2}}{\frac{\#(y_i=+1)+1}{k+2}N(x)+\frac{\#(y_i=-1)+1}{k+2}},
\end{aligned}
$$

where

$$
N(x) = \left(\sqrt{\frac{\sigma_-^2+\sigma_{\theta_-}^2}{\sigma_+^2+\sigma_{\theta_+}^2}}\right)^m \exp\left[-\frac{(x-\hat\theta_+)^T(x-\hat\theta_+)}{2(\sigma_+^2+\sigma_{\theta_+}^2)} + \frac{(x-\hat\theta_-)^T(x-\hat\theta_-)}{2(\sigma_-^2+\sigma_{\theta_-}^2)}\right].
$$

386 Finally, the decision boundary is $\hat y_{ICL} = 1(f_{ICL}(x) > 0)$, where $f_{ICL}(x) = N(x) - \frac{\#(y_i=-1)+1}{\#(y_i=+1)+1}$.
387 □

388 B.3 Proof of Theorem 1

389 In the following, we first provide the posterior of the parameters in a simplified scenario in Lemma 3,
390 and then use this result in Theorem 1 to derive the exact accuracy of ICL.

Lemma 3. *Under the conditions of Theorem 1, when taking $k \to \infty$, we can simplify $\hat{\theta}_+$, $\hat{\theta}_-$, $\sigma^2_{\theta_+}$, $\sigma^2_{\theta_-}$, $N(x)$, $f_{ICL}(x)$ as follows:*

$$\hat{\theta}_+ = \frac{\sigma^2\theta_M + \sigma_M^2 \sum_{y_i=+1} x_i}{\sigma^2 + \#(y_i = +1)\sigma_M^2},\ \hat{\theta}_- = \frac{\sigma_M^2 \sum_{y_i=-1} x_i - \sigma^2\theta_M}{\sigma^2 + \#(y_i = -1)\sigma_M^2},$$
$$\sigma^2_{\theta_+} = \frac{\sigma^2\sigma_M^2}{\sigma^2 + \#(y_i = +1)\sigma_M^2} \to 0,\ \sigma^2_{\theta_-} = \frac{\sigma^2\sigma_M^2}{\sigma^2 + \#(y_i = -1)\sigma_M^2} \to 0,$$
$$N(x) = \exp\left[\frac{1}{\sigma^2}(\hat{\theta}_+ - \hat{\theta}_-)^T x - \frac{1}{2\sigma^2}(\hat{\theta}_+^T\hat{\theta}_+ - \hat{\theta}_-^T\hat{\theta}_-)\right],\ f_{ICL}(x) = N(x) - z_k.$$

As mentioned in Section A, since $\sigma^2_{\theta_+}$ and $\sigma^2_{\theta_-}$ are negligible compared to other terms in $N(x)$, we remove them from f_{ICL}.

Proof of Theorem 1. We can compute the average ICL accuracy when test samples are sampled from the example distribution. We first derive the marginal distribution for test input x. For any fixed θ_+ and θ_-, following the data generation assumption in Section 2.1, we have

$$\begin{aligned} P(x|y=+1) &= \frac{1}{(\sqrt{2\pi\sigma^2})^m}\exp\left[-\frac{(x-\theta_+)^T(x-\theta_+)}{2\sigma^2}\right] \\ P(x|y=-1) &= \frac{1}{(\sqrt{2\pi\sigma^2})^m}\exp\left[-\frac{(x-\theta_-)^T(x-\theta_-)}{2\sigma^2}\right]. \end{aligned}$$

As a result, when integrating over all possible θ_+ and θ_-, it becomes

$$\begin{aligned} & P(x|y=+1) \\ = & \int_{\theta_+,\theta_-} P(x|y=+1,\theta_+,\theta_-)P_M(\theta_+)P_M(\theta_-)d\theta_+ d\theta_- \\ = & \int_{\theta_+,\theta_-} \frac{1}{(\sqrt{2\pi\sigma^2})^m}\exp\left[-\frac{1}{2\sigma^2}(x-\theta_+)^T(x-\theta_+)\right] \\ & \cdot\frac{1}{\left(\sqrt{2\pi\sigma_{e+}^2}\right)^m}\exp\left[-\frac{1}{2\sigma_{e+}^2}(\theta_+-\theta_+^e)^T(\theta_+-\theta_+^e)\right] \\ & \cdot\frac{1}{\left(\sqrt{2\pi\sigma_{e-}^2}\right)^m}\exp\left[-\frac{1}{2\sigma_{e-}^2}(\theta_--\theta_-^e)^T(\theta_--\theta_-^e)\right] d\theta_+ d\theta_- \\ = & \int_{\theta_+} \frac{1}{(\sqrt{2\pi\sigma^2})^m}\exp\left[-\frac{1}{2\sigma^2}(x-\theta_+)^T(x-\theta_+)\right] \\ & \cdot\frac{1}{\left(\sqrt{2\pi\sigma_{e+}^2}\right)^m}\exp\left[-\frac{1}{2\sigma_{e+}^2}(\theta_+-\theta_+^e)^T(\theta_+-\theta_+^e)\right] d\theta_+ \\ = & \frac{1}{\left(\sqrt{2\pi(\sigma^2+\sigma_{e+}^2)}\right)^m}\exp\left[-\frac{1}{2(\sigma^2+\sigma_{e+}^2)}(x-\theta_+^e)^T(x-\theta_+^e)\right]. \end{aligned}$$

Similarly,

$$P(x|y=-1) = \frac{1}{\left(\sqrt{2\pi(\sigma^2+\sigma_{e-}^2)}\right)^m}\exp\left[-\frac{1}{2(\sigma^2+\sigma_{e-}^2)}(x-\theta_-^e)^T(x-\theta_-^e)\right].$$

Then, we compute the probability of correct prediction for each class respectively.

$$P(correct|y=+1,\{(x_i,y_i)\},M) = \int_{f_{ICL}(x)\geq 0} P(x|y=+1)dx$$

Based on Lemma 2, we know that

$$\{x : f_{ICL}(x) \geq 0\} = \left\{x : (\hat{\theta}_+ - \hat{\theta}_-)^T x - \frac{\hat{\theta}_+^T\hat{\theta}_+ - \hat{\theta}_-^T\hat{\theta}_-}{2} \geq \sigma^2 \log z_k\right\}.$$

Let $z = (\hat{\theta}_- - \hat{\theta}_+)^T x - \frac{\hat{\theta}_+^T\hat{\theta}_+ - \hat{\theta}_-^T\hat{\theta}_-}{2} - \sigma^2 \log z_k$, then we have

$$z|y = +1, \{(x_i, y_i)\}, M \quad \sim \quad N\left((\hat{\theta}_+ - \hat{\theta}_-)^T\theta_+^e - \frac{\hat{\theta}_+^T\hat{\theta}_+ - \hat{\theta}_-^T\hat{\theta}_-}{2} - \sigma^2 \log z_k, \|\hat{\theta}_- - \hat{\theta}_+\|_2^2(\sigma^2 + \sigma_{e+}^2)\right),$$

which is still a Gaussian distribution. Therefore, we have

$$\begin{aligned} P(correct|y = +1, \{(x_i, y_i)\}, M) &= \int_{z\geq 0} P(z|y = +1)dz \\ &= \left(1 - \Phi\left(-\frac{(\hat{\theta}_+ - \hat{\theta}_-)^T\theta_+^e - \frac{\hat{\theta}_+^T\hat{\theta}_+ - \hat{\theta}_-^T\hat{\theta}_-}{2} - \sigma^2 \log z_k}{\|\hat{\theta}_- - \hat{\theta}_+\|_2\sqrt{\sigma^2 + \sigma_{e+}^2}}\right)\right) \\ &= \left(1 - \Phi\left(\frac{\theta_+^e - (\hat{\theta}_+ + \hat{\theta}_-)/2}{\sqrt{\sigma^2 + \sigma_{e+}^2}}\frac{\hat{\theta}_- - \hat{\theta}_+}{\|\hat{\theta}_- - \hat{\theta}_+\|} + \frac{\sigma^2 \log z_k}{\sqrt{\sigma^2 + \sigma_{e+}^2}\|\hat{\theta}_- - \hat{\theta}_+\|}\right)\right), \end{aligned}$$

and

$$\begin{aligned} P(correct|y = -1, \{(x_i, y_i)\}, M) &= \int_{z<0} P(z|y = -1)dz \\ &= \Phi\left(-\frac{(\hat{\theta}_+ - \hat{\theta}_-)^T\theta_-^e - \frac{\hat{\theta}_+^T\hat{\theta}_+ - \hat{\theta}_-^T\hat{\theta}_-}{2} - \sigma^2 \log z_k}{\|\hat{\theta}_- - \hat{\theta}_+\|_2\sqrt{(\sigma^2 + \sigma_{e-}^2)}}\right) \\ &= \Phi\left(\frac{(\theta_-^e - (\hat{\theta}_+ + \hat{\theta}_-)/2)^T}{\sqrt{\sigma^2 + \sigma_{e-}^2}}\frac{\hat{\theta}_- - \hat{\theta}_+}{\|\hat{\theta}_- - \hat{\theta}_+\|_2} + \frac{\sigma^2 \log z_k}{\sqrt{\sigma^2 + \sigma_{e-}^2}\|\hat{\theta}_- - \hat{\theta}_+\|}\right). \end{aligned}$$

□

B.4 Proof of Proposition 1

Proof of Proposition 1. Denote

$$\bar{x}_+ = \frac{1}{\#(y_i = +1)}\sum_{y_i=+1} x_i,\ \bar{x}_- = \frac{1}{\#(y_i = -1)}\sum_{y_i=-1} x_i,\ \bar{x} = \frac{1}{k}\sum_{i=1}^{k} x_i.$$

From Assumption 1 and Assumption 2, we know that

$$\bar{x}_- \sim N\left(\theta_-^e, \frac{\sigma_{e+}^2 + \sigma^2}{\#(y_i = +1)}I\right),\ \bar{x}_+ \sim N\left(\theta_+^e, \frac{\sigma_{e-}^2 + \sigma^2}{\#(y_i = -1)}I\right),$$

which implies that

$$\bar{x} \sim N\left(\frac{\#(y_i = -1)}{k}\theta_-^e + \frac{\#(y_i = +1)}{k}\theta_+^e, \left(\frac{\#(y_i = +1)}{k}\sigma_{e+}^2 + \frac{\#(y_i = -1)}{k}\sigma_{e-}^2\right)I\right).$$

We rewrite $\bar{x}_+$ and $\bar{x}_-$ via introducing zero-mean variables:

$$\bar{z}_+ = \bar{x}_+ - \theta_+^e,\ \bar{z}_- = \bar{x}_- - \theta_-^e.$$

When $\sigma_{e+}^2, \sigma_{e-}^2 \to 0$, the mean of $\bar{z}_+, \bar{z}_-$, and $\bar{x}$ are always zero.

Then, when $\theta^e_+ + \theta^e_- = 0$, we have $\mathbb{E}\bar{x} = 0$, and

$$\begin{aligned} & P(correct|y=+1,\{(x_i,y_i)\},M) \\ = \quad & 1-\Phi\left(\frac{\theta^e_+ - (\hat\theta_+ + \hat\theta_-)/2}{\sqrt{\sigma^2+\sigma^2_{e+}}}\frac{\hat\theta_- - \hat\theta_+}{\|\hat\theta_- - \hat\theta_+\|} + \frac{\sigma^2\log z_k}{\sqrt{\sigma^2+\sigma^2_{e+}}\|\hat\theta_- - \hat\theta_+\|}\right) \\ = \quad & 1-\Phi\left(\left(\frac{\theta^e_+ - \frac{k\sigma^2_M\bar{x}}{2\sigma^2+k\sigma^2_M}}{\sqrt{\sigma^2+\sigma^2_{e+}}}\right)^T \frac{0.5k\sigma^2_M[\theta^e_- - \theta^e_+ + (\bar{z}_- - \bar{z}_+)] - 2\sigma^2\theta_M}{\|0.5k\sigma^2_M[\theta^e_- - \theta^e_+ + (\bar{z}_- - \bar{z}_+)] - 2\sigma^2\theta_M\|_2}\right). \end{aligned}$$

Now we compare the case of contradicted knowledge and matched knowledge.

Contradict knowledge. When $\theta^e_- = \theta_M = -\theta^e_+$, we have

$$\begin{aligned} & P(correct|y=+1,\{(x_i,y_i)\},M) \\ = \quad & 1-\Phi\left(\left(\frac{-\theta_M - \frac{k\sigma^2_M\bar{x}}{2\sigma^2+k\sigma^2_M}}{\sqrt{\sigma^2+\sigma^2_{e+}}}\right)^T \frac{0.5k\sigma^2_M(\bar{z}_- - \bar{z}_+) + (k\sigma^2_M - 2\sigma^2)\theta_M}{\|0.5k\sigma^2_M(\bar{z}_- - \bar{z}_+) + (k\sigma^2_M - 2\sigma^2)\theta_M\|_2}\right). \end{aligned}$$

When $k\sigma^2_M \ll \sigma^2$, we have

$$P(correct|y=+1,\{(x_i,y_i)\},M) \to \left(1-\Phi\left(\frac{\|\theta_M\|_2}{\sqrt{\sigma^2+\sigma^2_{e+}}}\right)\right).$$

When $k\sigma^2_M \gg \sigma^2$, we have

$$P(correct|y=+1,\{(x_i,y_i)\},M) \to \left(1-\Phi\left(\frac{-\|\theta_M\|_2}{\sqrt{\sigma^2+\sigma^2_{e+}}}\right)\right).$$

Matched knowledge. When $\theta^e_- = -\theta_M = -\theta^e_+$, we have

$$\begin{aligned} & P(correct|y=+1,\{(x_i,y_i)\},M) \\ = \quad & 1-\Phi\left(\left(\frac{\theta_M - \frac{k\sigma^2_M\bar{x}}{2\sigma^2+k\sigma^2_M}}{\sqrt{\sigma^2+\sigma^2_{e+}}}\right)^T \frac{0.5k\sigma^2_M(\bar{z}_- - \bar{z}_+) - (k\sigma^2_M + 2\sigma^2)\theta_M}{\|0.5k\sigma^2_M(\bar{z}_- - \bar{z}_+) - (k\sigma^2_M + 2\sigma^2)\theta_M\|_2}\right). \end{aligned}$$

When $k\sigma^2_M \ll \sigma^2$, we have

$$P(correct|y=+1,\{(x_i,y_i)\},M) = \left(1-\Phi\left(\frac{-\|\theta_M\|^2}{\sqrt{\sigma^2+\sigma^2_{e+}}}\right)\right).$$

When $k\sigma^2_M \gg \sigma^2$, we have

$$P(correct|y=+1,\{(x_i,y_i)\},M) \to \left(1-\Phi\left(\frac{-\|\theta_M\|^2}{\sqrt{\sigma^2+\sigma^2_{e+}}}\right)\right).$$

□

B.5 Proof of Proposition 2

Proof of Proposition 2. When $k \to \infty$, we have $\hat\theta_+ \to \theta_M$ and $\hat\theta_- \to -\theta_M$, as well as $\sigma^2_{\theta_+} \to 0, \sigma^2_{\theta_-} \to 0$. In this case, the ICL decision boundary is still a hyperplane, and we can use the result in Lemma 2 and simplify the decision function into

$$f_{ICL}(x) = \exp\left[-\frac{(x-\hat\theta_+)^T(x-\hat\theta_+)}{2\sigma^2} + \frac{(x-\hat\theta_-)^T(x-\hat\theta_-)}{2\sigma^2}\right] - \frac{1-\pi}{\pi}.$$

Then, following the same definition of z as Theorem 1, we can compute the ICL accuracy:

$$P(correct|y=+1,\{(x_i,y_i)\},M)=\int_{z\geq\log\left(\frac{1-\pi}{\pi}\right)}P(z|y=+1)dz.$$

Since $\log\frac{1-\pi}{\pi}$ goes to ∞ when $\pi\to 0$, $P(correct|y=+1)\to 0$. $\square$

B.6 Proof of Proposition 3

Proof of Proposition 3. Following the definition of $\hat\theta_+$ and $\hat\theta_-$, when taking $k\to\infty$ and $\theta_+^e=\theta_M=-\theta_-^e$, we obtain

$$\begin{aligned}\hat\theta_-+\hat\theta_+ &\to (1+p_+^e-p_-^e)\theta_+^e+(1-p_+^e+p_-^e)\theta_-^e=2(p_+^e-p_-^e)\theta_M,\\ \hat\theta_--\hat\theta_+ &\to (1-p_+^e-p_-^e)(\theta_+^e-\theta_-^e)=2(1-p_+^e-p_-^e)\theta_M.\end{aligned}$$

Taking the above into the formula of $P(correct|y=+1)$ in Theorem 1, we obtain

$$\begin{aligned}P(correct|y=+1,\{(x_i,y_i)\},M) &= 1-\Phi\left(\frac{\theta_+^e-(\hat\theta_++\hat\theta_-)/2}{\sqrt{(\sigma^2+\sigma_{e+}^2)}}\frac{\hat\theta_--\hat\theta_+}{\|\hat\theta_--\hat\theta_+\|}+\frac{(\sigma^2+\sigma_\theta^2)\log z_k}{\|\hat\theta_--\hat\theta_+\|\sqrt{(\sigma^2+\sigma_{e+}^2)}}\right)\\ &= 1-\Phi\left(C_1(1-p_+^e+p_-^e)\|\theta_M\|\text{sign}(1-p_+^e-p_-^e)+C_2\frac{\log z_k}{|1-p_+^e-p_-^e|}\right),\end{aligned}$$

and

$$\begin{aligned}P(correct|y=-1,\{(x_i,y_i)\},M) &= \Phi\left(\frac{(\theta_-^e-(\hat\theta_++\hat\theta_-)/2)^T}{\sqrt{\sigma^2+\sigma_{e-}^2}}\frac{\hat\theta_--\hat\theta_+}{\|\hat\theta_--\hat\theta_+\|_2}+\frac{(\sigma^2+\sigma_\theta^2)\log z_k}{\|\hat\theta_--\hat\theta_+\|\sqrt{(\sigma^2+\sigma_{e+}^2)}}\right)\\ &= \Phi\left(-C_1(1+p_+^e-p_-^e)\text{sign}(1-p_+^e-p_-^e)+C_2\frac{\log z_k}{|1-p_+^e-p_-^e|}\right).\end{aligned}$$

where $C_1=2\frac{\|\theta_M\|}{\sqrt{(\sigma^2+\sigma_{e+}^2)}}>0, C_2=\frac{\sigma^2}{\sqrt{(\sigma^2+\sigma_{e+}^2)}\|\theta_M\|}>0.$ $\square$

B.7 Formal statement and proof of Theorem 3

Theorem 3 (Dependent examples)**.** *Assume $\{(x_i,y_i)\}$ are not independent and are considered as a sequence of inputs. Let $frac$ denote the fraction of 1 among the set of labels in the pre-training set. Assume $frac$ approximately[3] follows Beta(α,β) with $\alpha,\beta\to\infty$ and $\alpha/\beta\to\pi/(1-\pi)$ for some constant $\pi\in(0,1)$, i.e., $frac$ is around π with probability tending to 1. Further, assume that all y_is and y have an equal chance of being positive in pre-training. Then when $P(x|y=+1,\{(x_i,y_i)\},M)$ and $P(x|y=-1,\{(x_i,y_i)\},M)$ are both bounded and bounded away from zero, the following holds:*

$$P(y=+1|x,\{(x_i,y_i)\},M)\to\begin{cases}1 & \textit{if } \frac{\#(y_i=1)}{k+1}<\frac{\lfloor\pi(k+1)\rfloor-1}{k+1}\\ 0 & \textit{if } \frac{\#(y_i=1)}{k+1}>\frac{\lceil\pi(k+1)\rceil+1}{k+1}\end{cases}.$$

Proof of Theorem 3. During pre-training, since there are k examples and one test sample in the prompt, the fraction of positive labels can only take values in the form of $i/(k+1)$ for $i=0,\dots,k$, rather than a continuous variable in $[0,1]$. As a result, to connect the distribution of $frac$ with the Beta distribution, we assume $k+1$ is odd and denote B as a random variable following Beta(α,β). Then we set the following:

$$P(frac=i/(k+1)|M)=\begin{cases}P\left(B<\frac{1}{2(k+1)}\right) & i=0\\ P\left(\frac{2i-1}{2(k+1)}\leq B<\frac{2i+1}{(2k+1)}\right) & 0<i<k+1\\ P\left(B\geq\frac{2k+1}{2(k+1)}\right) & i=k+1\end{cases}.$$

[3] The fraction number given a finite number of examples follows a discrete distribution. Here we approximate it to the Beta distribution and focus on the intuition. Details can be found in the proof.

In addition, we assume that all y_is and y have equal chance of being positive.

Based on our assumption, when LLM learns from the pre-training data, it can exactly learn the distribution of $frac$, and use the likelihood to make a decision when receiving the testing data.

In the testing stage, when receiving $\{(x_i, y_i)\}_{i\in[k]}$ and x. From the definition of conditional probability, we know that

$$P(y=+1|x,\{(x_i,y_i)\},M)=\frac{P((x,y=+1)|\{(x_i,y_i)\},M)}{P((x,y=+1)|\{(x_i,y_i)\},M)+P((x,y=-1)|\{(x_i,y_i)\},M)},$$

where

$$P((x,y=+1)|\{(x_i,y_i)\},M)=P(x|y=+1,\{(x_i,y_i)\},M)P(y=+1|\{(x_i,y_i)\},M).$$

From the above, we need to figure out the following quantity:

$$\begin{aligned} & P((x,y=+1)|\{(x_i,y_i)\},M) \\ = & P(x|y=+1,(x_i,y_i),M)P(y=+1|\{(x_i,y_i)\},M) \\ = & P(x|y=+1,\{(x_i,y_i)\},M)P\left(frac=\frac{1+\#(y_i=+1)}{k+1}\middle|\{(x_i,y_i)\},M\right), \end{aligned}$$

where

$$\begin{aligned} & P\left(frac=\frac{1+\#(y_i=+1)}{k+1}\middle|\{(x_i,y_i)\},M\right) \\ = & \frac{P\left(frac=\frac{1+\#(y_i=+1)}{k+1},\{(x_i,y_i)\}\middle|M\right)}{P\left(frac=\frac{1+\#(y_i=+1)}{k+1},\{(x_i,y_i)\}\middle|M\right)+P\left(frac=\frac{\#(y_i=+1)}{k+1},\{(x_i,y_i)\}\middle|M\right)}. \end{aligned} \tag{2}$$

To calculate $P(frac=(1+\#(y_i=+1))/(k+1),\{(x_i,y_i)\}|M)$, when $frac=(1+\#(y_i=+1))/(k+1)$, it means that there are $1+\#(y_i=+1)$ examples (and the query) which have a positive label. Given a total of $k+1$ data, there are $C_{k+1}^{1+\#(y_i=+1)}$ different combinations. As a result, for a fixed $\{(x_i,y_i)\}$, we have

$$P\left(frac=\frac{1+\#(y_i=+1)}{k+1},\{(x_i,y_i)\}\middle|M\right)=\frac{1}{C_{k+1}^{1+\#(y_i=+1)}}P\left(frac=\frac{1+\#(y_i=+1)}{k+1}\middle|M\right).$$

Similarly, we obtain that

$$P\left(frac=\frac{\#(y_i=+1)}{k+1},\{(x_i,y_i)\}\middle|M\right)=\frac{1}{C_{k+1}^{\#(y_i=+1)}}P\left(frac=\frac{\#(y_i=+1)}{k+1}\middle|M\right).$$

Taking the above into (2), it becomes

$$\begin{aligned} & P\left(frac=\frac{1+\#(y_i=+1)}{k+1}\middle|\{(x_i,y_i)\},M\right) \\ = & \frac{P\left(frac=\frac{1+\#(y_i=+1)}{k+1}\middle|M\right)C_{k+1}^{\#(y_i=+1)}}{P\left(frac=\frac{1+\#(y_i=+1)}{k+1}\middle|M\right)C_{k+1}^{\#(y_i=+1)}+P\left(frac=\frac{\#(y_i=+1)}{k+1}\middle|M\right)C_{k+1}^{1+\#(y_i=+1)}} \\ = & \frac{1}{1+\frac{P\left(frac=\frac{\#(y_i=+1)}{k+1}|M\right)}{P\left(frac=\frac{1+\#(y_i=+1)}{k+1}|M\right)}\frac{k-\#(y_i=+1)+1}{1+\#(y_i=+1)}}. \end{aligned}$$

To further look at the exact value of $P\left(frac=\frac{1+\#(y_i=+1)}{k+1}\middle|\{(x_i,y_i)\},M\right)$, we need to figure out $P(frac=i/(k+1)|M)$ using the Beta distribution. Recall that the probability density function f of Beta(α,β) satisfies

$$f(u)=\frac{u^{\alpha-1}(1-u)^{\beta-1}}{B(\alpha,\beta)}$$

for Beta function $B(\alpha, \beta)$. Recall that we assume that $\alpha/\beta = \pi/(1-\pi)$ for some $\pi \in (0, 1)$, and both $\alpha, \beta \to \infty$. When $u < (\alpha-1)/(\alpha+\beta-2) \approx \pi$, we have f is an increasing function in u, otherwise f is decreasing. This implies that the largest probability of $frac$ may be taken from $P(frac = (\lfloor \pi(k+1) \rfloor + 1)/(k+1))$, $P(frac = \lfloor \pi(k+1) \rfloor/(k+1))$ or $P(frac = (\lfloor \pi(k+1) \rfloor - 1)/(k+1))$.

When $frac < (\lfloor \pi(k+1) \rfloor - 1)/(k+1)$, when α and β are large enough, one can obtain that

$$\frac{P\left(frac = \frac{\#(y_i=+1)}{k+1} | M\right)}{P\left(frac = \frac{1+\#(y_i=+1)}{k+1} | M\right)} \frac{k - \#(y_i = +1) + 1}{1 + \#(y_i = +1)} \to 0,$$

which implies that

$$P\left(frac = \frac{1+\#(y_i = +1)}{k+1} \middle| \{(x_i, y_i)\}, M\right) = \frac{1}{1 + \frac{P\left(frac = \frac{\#(y_i=+1)}{k+1} | M\right)}{P\left(frac = \frac{1+\#(y_i=+1)}{k+1} | M\right)} \frac{k-\#(y_i=+1)+1}{1+\#(y_i=+1)}} \to 1.$$

Similarly, when $frac > (\lceil \pi(k+1) \rceil + 1)/(k+1)$,

$$P\left(frac = \frac{1+\#(y_i = +1)}{k+1} \middle| \{(x_i, y_i)\}, M\right) = \frac{1}{1 + \frac{P\left(frac = \frac{\#(y_i=+1)}{k+1} | M\right)}{P\left(frac = \frac{1+\#(y_i=+1)}{k+1} | M\right)} \frac{k-\#(y_i=+1)+1}{1+\#(y_i=+1)}} \to 0.$$

Finally, we put $P\left(frac = \frac{1+\#(y_i=+1)}{k+1} \middle| \{(x_i, y_i)\}, M\right)$ into $P(y = +1|x, \{(x_i, y_i), M\})$:

$$\begin{aligned} & P(y = +1|x, \{(x_i, y_i)\}, M) \\ = & \frac{P((x, y = +1)|\{(x_i, y_i)\}, M)}{P((x, y = +1)|\{(x_i, y_i)\}, M) + P((x, y = -1)|\{(x_i, y_i)\}, M)} \\ = & \frac{P(x|y = +1, \{(x_i, y_i)\}|M) P(frac = \frac{1+\#(y_i=+1)}{k+1} |\{(x_i, y_i)\}, M)}{P(x|y = +1, \{(x_i, y_i), M\}) P(frac = \frac{1+\#(y_i=+1)}{k+1} |\{(x_i, y_i)\}, M) + P(x|y = -1, \{(x_i, y_i)\}, M) P(frac = \frac{\#(y_i=+1)}{k+1} |\{(x_i, y_i)\}, M)}. \end{aligned}$$

When $P(x|y = +1, \{(x_i, y_i), M\})$ and $P(x|y = -1, \{(x_i, y_i), M\})$ are both bounded and bounded away from zero, we have

$$P(y = +1|x, \{(x_i, y_i)\}, M) \to \begin{cases} 1 & \text{if } \frac{\#(y_i=+1)}{k+1} < \frac{\lfloor \pi(k+1) \rfloor - 1}{k+1} \\ 0 & \text{if } \frac{\#(y_i=+1)}{k+1} > \frac{\lceil \pi(k+1) \rceil + 1}{k+1} \end{cases},$$

which completes the proof. $\square$

C Simulation setups

In this section, we provide details of experimental setups for simulation.

Model structure. We pre-train a decoder-only Transformer [19] from the GPT-2 [20] family. This model has 12 layers, 8 attention heads, and a 256-dimensional embedding space. The model input takes the form of $(x_1, y_1, x_2, y_2, \ldots)$. In our training, $x_i \in \mathbb{R}^5$ and $y_i \in \{1, -1\}$. We map y_i to the same dimension of x_i by appending zeros. Then the whole prompt will be projected into the latent embedding space of the Transformer through a (learnable) MLP layer. Another (learnable) MLP layer is used to project embeddings back to scalars in the output.

Pre-training. We train the model using a cross-entropy loss function for binary classification. We sample a batch of random prompts at each training step and update the model through a gradient update. We train with a batch size of 64 and for 50k steps. This training is done from scratch, that is, we do not fine-tune a pre-trained language model, nor do we train on actual text. Following previous work [9], we also use curriculum learning [21, 22]. In particular, we start with a shorter length of prompts (10 input-output pairs) and increase the length by 2 every 2000 training steps. For the other hyperparameters, e.g., learning rate, we use the default values as in [9].

Pre-training data. We follow the data generation model in Section 2.1. We first select label $y \in \{+1, -1\}$ with probability π (positive probability). Then for inputs with positive labels, we first sample a mean value θ_+ from a Gaussian distribution $N(\theta_M, \sigma_M^2 I)$, and then sample data x from Gaussian distribution $N(\theta_+, \sigma^2 I)$; similar for the inputs with label -1, we sample θ_- from $N(-\theta_M, \sigma_M^2 I)$, and sample x from $N(\theta_-, \sigma^2 I)$. Specifically, we let $\theta_M = 0.5\mathbf{1}, \sigma_M^2 = \sigma^2 = 1$. During the pre-training, to ensure the transformer can learn the population information rather than overfitting a particular set of data, we sample a new pair of (θ_+, θ_-) for each iteration and generate corresponding sample pair (x_i, y_i).

Computation resources. Both simulations and real-world experiments are running on a server with 8 Nvidia RTX A6000 GPU (48G GPU memory each) and 32 AMD EPYC 7302 16-Core Processors.

D Experiments

In this section, we empirically verify the analysis in Section 2. In summary, both simulation and real-data experiments are consistent Section 2[4].

D.1 Simulation

To set up the experiment, we pre-train a decoder-only Transformer [19] from the GPT-2 [20] family. We follow Section 2.1 to construct the pre-training data and follow [9] to perform next-token prediction to estimate all y_is and y_{query}. During the pre-training, we sample a new pair of (θ_+, θ_-) for each iteration and generate corresponding demonstration examples. A detailed setting for simulation can be found in Appendix C.

We implement the scenarios as in Section 2.3:

Contradicting knowledge We pre-train the model with $\theta_M = 0.5\mathbf{1}_5, \sigma_M^2 = 1, \sigma^2 = 1$. During the inference stage, we generate examples and test data with $\theta_+^e = -0.5\mathbf{1}_5, \theta_-^e = 0.5\mathbf{1}_5$ which contradicts the pre-training distribution. We let $\sigma_{e+}^2 = \sigma_{e-}^2 = 1$, and test on various $\sigma^2 \in \{1, 2, 4\}$. The results for $\sigma^2 = 2, 4$ are postponed to Section F.1. We compute the ICL accuracy for each class when k increases. For comparison, we also generate examples with matched knowledge ($\theta_+^e = 0.5\mathbf{1}_5 = -\theta_-^e$) and examine the ICL accuracy.

The results can be found in Figures 1, where the X-axis represents the number of examples k, and the Y-axis is the ICL accuracy. The red dash denotes σ^2/σ_M^2 based on Proposition 1 and $\sigma^2/\sigma_M^2 = 1$ in our simulation. There are two observations. First, when $k \leq \sigma^2/\sigma_M^2$, the ICL performance of contradicting knowledge is worse than that of matching knowledge, verifying that the transformer heavily relies on the pre-training knowledge when there are limited examples. Second, when k

[4] Code is available in `https://anonymous.4open.science/r/ICL-understanding-classification-DC1C`

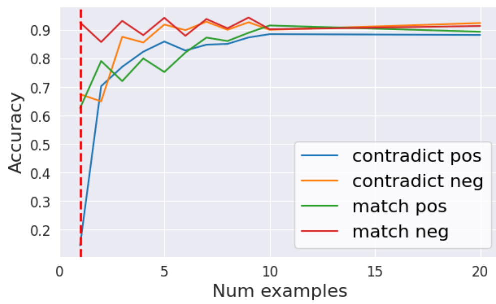

Figure 1: Contradicting knowledge when $\sigma^2 = 1$

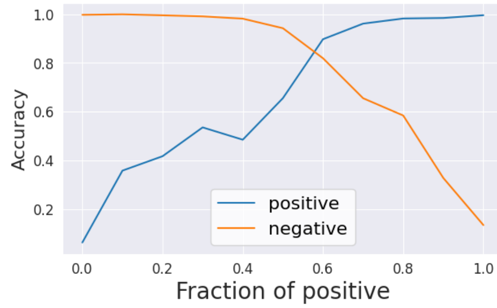

Figure 2: Imbalanced examples

increases, the ICL performance for contradicting knowledge increases to around 87% when $k = 20$, indicating that the knowledge from the examples will dominate when k is large.

Imbalanced examples We pre-train the model with $\theta_M = 0.5\mathbf{1}_5, \sigma_M^2 = 1$, $\sigma^2 = 1$. During the inference stage, we generate examples and the test data with $\theta_+^e = 0.5\mathbf{1}_5, \theta_-^e = -0.5\mathbf{1}_5$ and $\sigma_{e+}^2 = \sigma_{e-}^2 = 1, \sigma^2 = 1$. We test with various fraction $\pi \in \{0, 0.1, 0.2, ..., 0.9, 1.0\}$. In Figure 2, the X-axis represents the fraction of positive examples among all examples in the demonstration, i.e., π. We can observe that when the fraction of positive is increasing, ICL accuracy for positive inputs is increasing and finally reaches 100%, while ICL accuracy for negative inputs is decreasing, which is the same as described in Proposition 2.

Label noise We pre-train the model with $\theta_M = 0.5\mathbf{1}_5, \sigma_M^2 = 1$, $\sigma^2 = 1$. During the inference stage, we generate examples and the test data with $\theta_+^e = 0.5\mathbf{1}_5, \theta_-^e = -0.5\mathbf{1}_5$, and fix $\sigma_{e+}^2 = \sigma_{e-}^2 = 1, \sigma^2 = 0.01$. The example size k is 100, and π is 0.5. For the examples in each class, we randomly flip their label with probability $1 - p_+^e, 1 - p_-^e$ respectively, and test ICL accuracy for each class for $1 - p_+^e, 1 - p_-^e \in \{1, 0.9, 0.8, ..., 0.1, 0\}$. The ICL accuracies for each class and the overall result are summarized in Figures 3.

The phenomenon in these heatmaps is consistent with our conclusion in Proposition 3. Take the ICL accuracy of the positive class as an example (the first figure in Figure 3); we observe that when the flipping probability in the negative class is fixed, smaller flipping probability (higher p_+^e) in the positive class usually leads to higher accuracy in the positive class. Moreover, the diagonal from the bottom left to the upper right represents cases when $p_+^e + p_-^e = 1$ and it is obvious that the positive accuracy is either approximately 0, 0.5, or 1. Similar observations can be found for the negative class as well (the middle panel of Figure3). In terms of the overall accuracy, only when both p_+^e and p_-^e are close to 1, the overall accuracy is greater than 80%.

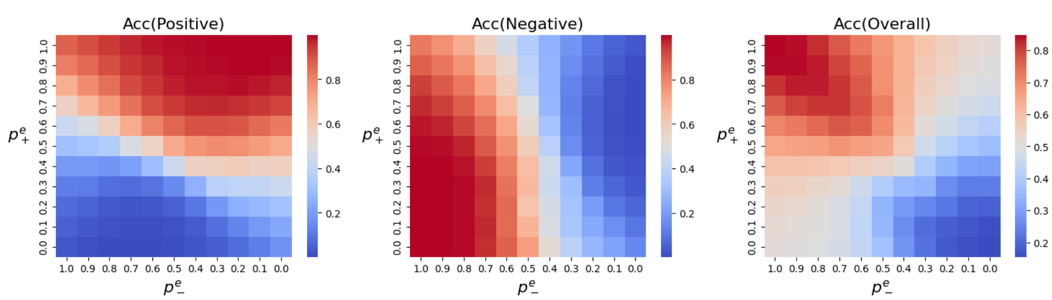

Figure 3: Simulation results on positive and negative accuracy facing label noises.

D.2 Real-Data Experiment

In this subsection, we conduct experiments on real datasets to show that the theoretical insights in Section 2 also align with the practical scenarios.

We consider two popular pre-trained LLMs, Pythia-6.9B [23] and Llama2-7B [24]. We test on a sentiment analysis dataset, SST2 dataset [25], which is also a binary classification task (labeled as "positive" and "negative"). During the inference, we randomly select k samples from the training set as examples and compute the ICL accuracy for each class. We repeat the process 10 times and record the average accuracy. If not specified, $k = 50$.

Label noise Similar to the simulation, we also randomly flip the label of examples from positive and negative classes and the correct probability is p_+^e, p_-^e respectively. We let $p_+^e, p_-^e \in \{0.0, 0.1, ..., 0.9, 1.0\}$ and record ICL accuracy in each class. Results are shown in Figure 4, 5, 6, 7. It can be consistently observed that when p_-^e is fixed, larger p_+^e leads to higher accuracy in the positive class (Figure 4 and 6); when p_+^e is fixed, larger p_-^e leads to higher accuracy in the negative class (Figure 5 and 7). This observation is consistent with our analysis in Proposition 3.

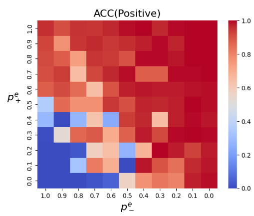

Figure 4: Pythia-6.9B: ICL acc, positive class.

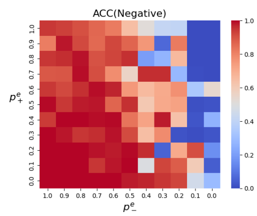

Figure 5: Pythia-6.9B: ICL acc, negative class.

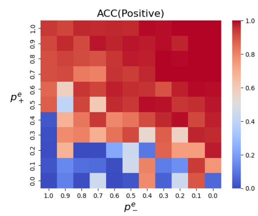

Figure 6: Llama2-7B: ICL acc, positive class.

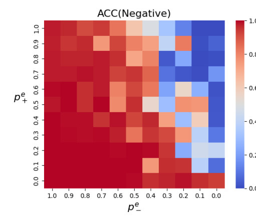

Figure 7: Llama2-7B: ICL acc, negative class.

Imbalanced examples We also conduct experiments when the fraction π of examples from the positive class is not 0.5. Specifically, we test with $\pi \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$. As depicted in Figure 8 and 9, when the number of positive examples increases, the accuracy of the positive class increases as that of the negative class decreases, which also supports our analysis in Proposition 2.

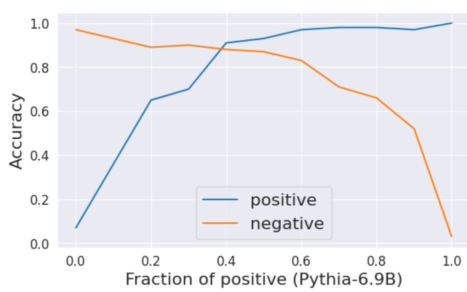

Figure 8: Imbalance examples, Pythia-6.9B.

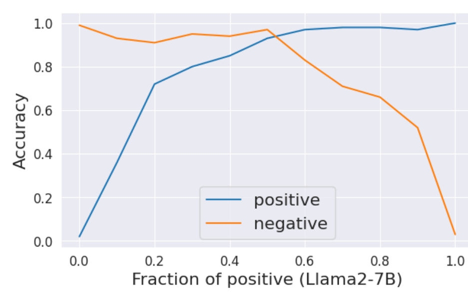

Figure 9: Imbalance examples, Llama2-7B.

E Mean Reversion in ICL with Dependent Examples

The previous analysis and experiments provide a comprehensive understanding of the effect of pre-training and examples under the independent-example scenario. However, it is also common when examples are sampled dependently, especially when examples are strategically selected to serve a specific objective, such as ensuring a balanced representation of 50% positive and 50% negative examples to prevent dataset imbalance [26, 27]. Surprisingly, we discover a counter-intuitive phenomenon, named as "**mean reversion**", under this scenario. We first empirically illustrate this phenomenon and then provide a theoretical analysis.

Empirical illustration of Mean Reversion We follow a similar procedure as introduced in Section D, while the difference is that during the pre-training stage, we fix the fraction of positive labels (examples + test data) to be exactly 0.5. This differs from the independent case since the fraction π

may fluctuate around 0.5 instead of strictly equal to 0.5. During the inference, in-context examples are sampled from both classes but are all labeled as positive or negative. We test with various fractions of positive examples in the prompt to see how the prediction for the test input x is affected. Results are shown in Figure 10 11.

In Figure 10, all examples are labeled as positive and the X-axis reflects the fraction of examples truly from the positive class; while in 11, all examples are labeled as negative and the X-axis reflects the fraction of examples truly from the negative class. There are four lines in the two figures. The "Positive"/"Negative" refers to the probability of the prediction being positive/negative when the correct label is positive/negative. The "Total pos"/"Total neg" represents the marginal probability of the prediction being positive/negative.

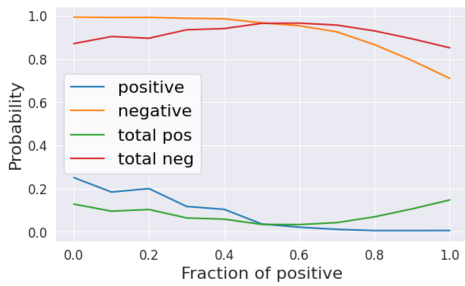

Figure 10: All y_is are positive.

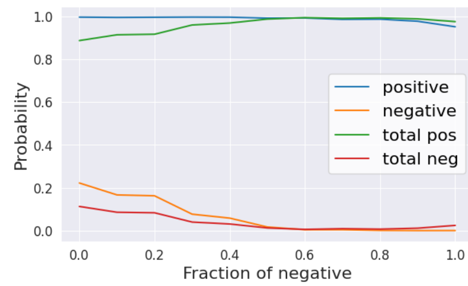

Figure 11: All y_is are negative.

As shown in Figure 10, we can see that regardless of the change in the fraction of examples' true classes (X-axis), we always obtain a low positive rate for the true positive test data, and the overall positive rate is low. This contradicts the independent case in Figure 2. Intuitively, since LLM learns that the fraction of positive labels is exactly 0.5 in the pre-training, at the inference stage, the joint label distribution of examples and test input appears to converge to 0.5-0.5 [5].

In the following, we provide a rigorous theoretical analysis to explain this phenomenon:

Theorem 4 (**Mean Reversion**, informal version of Theorem 3)**.** *Let $frac$ denote the fraction of $+1$ among the set of labels in the pre-training set. Under some mild conditions, assume in each prompt in pre-training, $frac$ is always a fixed π, then in the testing prompt: (1) If $\#(y_i = +1)/k < \pi$, then the prediction of x is $+1$. (2) If $\#(y_i = +1)/k > \pi$, then the prediction of x is -1.*

We direct the reader into Appendix B.7 for the formal statement and detailed proof. In short, when calculating $P((x, y = +1)|\{(x_i, y_i)\}, M) = P(x|y = +1, \{(x_i, y_i)\}, M)P(y = +1|\{(x_i, y_i)\}, M)$, since the examples are not independent, we need to follow the dependency among y_is and y to determine $P(y = +1|\{(x_i, y_i)\}, M)$, which is determined by the relationship between $\#(y_i = +1)/(k + 1)$ in the testing data and $frac$ in pre-training.

Theorem 4 indicates that the conditional probability of y is determined by the fraction of labels within the pre-training set and the examples during inference, in addition to the inputs. A direct corollary is that when the fraction of $y_i = +1$ is fixed as 0.5 during the pre-training, and all y_i are negative in the inference stage, the prediction for y is always positive. This is consistent with the observation in Figure 10, 11.

F Additional Experiment Results

F.1 Simulation for Independent Exampels

Figure 12, 13, 14 represents additional results corresponding to the contradict knowledge setting in Figure 1. The observations are similar to Figure 1.

F.2 Mean Reversion

We further pre-train GPT models with different fractions ($frac = 0.2, 0.5, 0.8$) and test the posterior distribution of labels when the fraction of positive labels within examples varies. We do not add noise

[5] This is similar to the "mean reversion" in certain stochastic differential equations (SDEs) where the variable tends to move toward a long-term average over time, thus we also name our observation as "mean reversion".

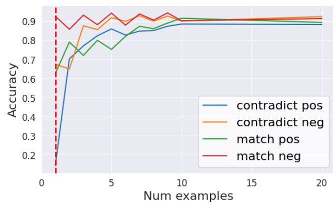

Figure 12: $\sigma^2 = 1$

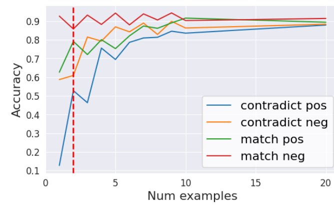

Figure 13: $\sigma^2 = 2$

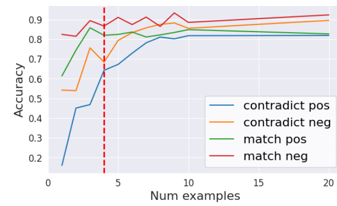

Figure 14: $\sigma^2 = 4$

and keep other settings unchanged. We observe a dramatic change around 0.2,0.5,0.8 respectively in Figure 15, and these figures directly verify our results: in Theorem 3, the cutting point are $0.2, 0.5, 0.8$ respectively in the three settings.

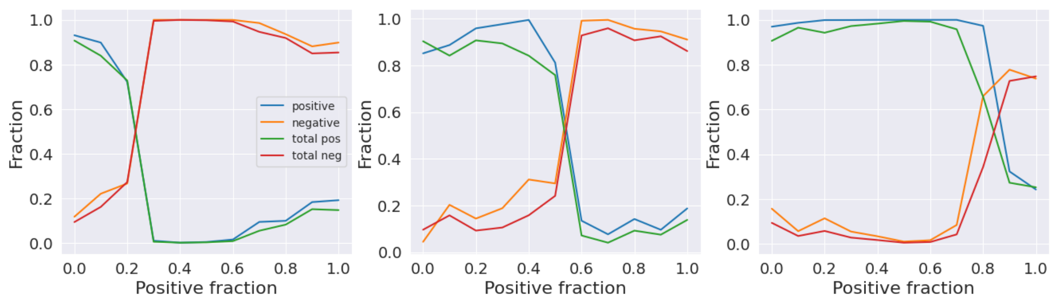

Figure 15: Pre-training with a fraction of positive 0.2.

G Related Works

Empirical findings of ICL. There are many empirical studies working on understanding ICL. [28] first reveals that LLMs can learn from examples, and refers to it as in-context learning (ICL). Later, to investigate the properties of ICL, [9] empirically shows that a transformer-based model can learn linear functions in context. [29, 30] find that transformer models can encode input-output relationships in the hidden space of attention layers. [6] observes that the key respects of the demonstration are label space, distribution of input, and format of the prompt. They also notice that randomly replacing labels barely hurts the performance when the example size is not large. [31, 32, 33] reveals the importance of examples, including orders and templates. More works are proposed to select examples [32, 26, 27] or design prompts [34, 35] to improve the ICL performance.

Theoretical understanding of ICL. To theoretically understand ICL, one popular line of research is to treat the ICL process as an implicit gradient descent procedure on examples. [2] shows that a single linear self-attention layer trained by gradient flow results in a competitive prediction error with the best linear predictor during ICL. [10, 11, 36] shows that one attention layer can be exactly constructed to perform gradient descent. [37, 38] further prove that under some conditions, a transformer with one or more attention layers trained on noisy linear regression task minimizing the pre-training loss will implement gradient descent algorithm on examples. [12] show that ICL can asymptotically converge to kernel regression as the number of examples increases.

Another line of research focuses on Bayesian inference. [7] first leverages a Hidden Markov Model to represent the pre-training data and prove that a transformer trained on such data exhibits the ICL ability. [39] introduces an information-theoretic tool to show how ICL error decays in the number and length of examples. [5] introduces a probabilistic model to understand two modes of ICL, i.e., task learning and task retrieval [40], on the linear regression tasks.

However, these analyses primarily focus on regression tasks with continuous outputs, and lack precise quantification for classification scenarios. Furthermore, they often overlook scenarios where the distribution of in-context examples diverges from pre-training data, such as cases of label noise, imbalanced examples, or contradictory information. Our work addresses these limitations by focusing on classification problems, providing exact quantification of example effects on predictions, and offering insights into the impact of label noise, imbalanced examples, and contradictory knowledge on in-context predictions.