Shift is Good: Mismatched Data Mixing Improves Test Performance

Anonymous Author(s)
Affiliation
Address
email

Abstract

We consider training and testing on mixture distributions with different training and test proportions. We show that in many settings, and in some sense generically, distribution shift can be beneficial, and test performance can improve due to mismatched training proportions. In a variety of scenarios, we identify the optimal training proportions and the extent to which such distribution shift can be beneficial.

1 Introduction

Imagine that you are taking a high-stakes exam next week. The exam will be 90% on European history and 10% on Chinese history. Both topics are equally familiar to you and equally difficult, and additional study will help you with each topic similarly. You have unlimited access to study material and practice questions for both. How should you spend your limited studying budget? Should your training match your test distribution, studying 90% European and 10% Chinese? Or would you benefit from a distribution shift? Studying more Chinese history? Less? Only European history? *We encourage the reader to pause and make an intuitive guess.*

The answer depends on the specific learning curve for improvement in test performance within a topic as a function of the number of training examples from that topic. But at least for a generic $1/n$ scaling (as obtained from e.g., both learning VC classes and in parametric regression), the answer, as we will see in Section 3, is that you would benefit from a distribution shift, and should study 75% European History and 25% Chinese history—this would reduce your test error by 20% over the $90/10$ non-shifted training.

We just saw an example of what we term **Positive Distribution Shift**: Even if we have unlimited data from the target test distribution D_{test}, training on a shifted distribution $D_{\text{train}} \neq D_{\text{test}}$ can actually *improve* test performance. This contrasts the typical study of *distribution shift*, i.e. training on one distribution but then applying the predictor, or testing, on another. Typically, it is implicitly assumed that the ideal case would be to train on the test distribution, that training on a different distribution is a compromise, either because we don't know or have access to the true D_{test}, or it's expensive to sample from it, or we have only a limited number of samples and want to supplement them with additional data from related distributions. Distribution shift is usually studied as "how much worse do things get if we train on $D_{\text{train}} \neq D_{\text{test}}$", with answers of the form "if D_{train} is close or related enough to D_{test}, then it's not much worse". In this paper, we investigate one of several ways in which distribution shift can be *positive*.

Specifically, we systematically study the benefit of such distribution shift when training with mismatched mixing proportions relative to the test distribution. We model the test distribution as a mixture of K components, with known mixing proportions $\{p_k\}_{k=1}^K$, and consider training distributions which are mixtures over the same components but with different mixing proportions $\{q_k\}_{k=1}^K$.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

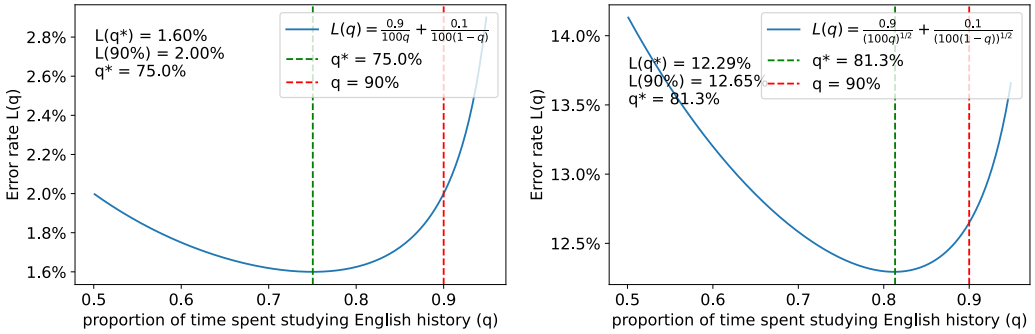

Figure 1: We plot the error rate for a hypothetical scenario modelling the high stakes exam described in Section 1. We model the error rate on each of the test portions as being proportional to $\propto \frac{1}{n_i^\alpha}$, where n_i represents the studying budget spent on that portion of the exam, so $i = 1$ corresponds to European History and $i = 2$ to the Chinese History and set $n_1 + n_2 = N$ to be the total studying budget, with $N = 100$ hours. The exponent α is $\alpha = 1$ on the left plot and $\alpha = 2$ on the right plot. In both cases, we consider $n_1 = qN$ and $n_2 = (1-q)N$, where q is the proportion of time spent studying for the European History portion of the exam. This way, the error rate on the exam can be written as a function of q as $L(q) = 0.9\frac{1}{(100q)^\alpha} + 0.1\frac{1}{(100q)^\alpha}$. We can see on both plots that shifting away from the testing proportion (red line, i.e. $q = 90\%$) can lead to a better error rate with the optimal test proportion (green line, i.e. q^* whose values are displayed accordingly). See also Corollary 3.3.

We can either think of this as providing guidance when we can actively control mixing between different known components, or as helping us understand how and why a mismatched training distribution can actually be beneficial. In Section 5 we discuss how the analysis is also applicable to a setting where we are not testing on a mixture, but rather on compositional tasks, requiring composing multiple skills, and the skills appear with differing frequencies—this compositional setting served as a major motivation for our study.

We consider different per-component learning curves, capturing different error decays, differing hardness among the components, and the possibility of transfer between components. In Section 3 we consider power law error decay, both the $1/n$ decay mentioned earlier and more general power laws, including with differing component hardnesses or error decays. In Section 4 we consider learning curves corresponding to "fact memorization" scenarios (discussed in Section 4), including those applicable to the skill composition setting, and which correspond to coupon-collector type learning curves. In Section 6 we consider the possibility of transfer between components. In all of these, we show that a mismatched training distribution can be beneficial, characterize the optimal training mixture, and the extent to which mismatch can improve test performance and reduce the training complexity.

Beyond all the specific scenarios, we then argue, in Section 7, that benefiting from mismatch is not the exception but rather the rule. We show that only in rare situations (either measure zero or satisfying a conservation property that does not generally hold) is the optimal training distribution equal to the test distribution, while in "most" cases shift is good.

2 Setup

Learning Setup and Loss For concreteness, let $\ell(h, \boldsymbol{z})$ be the loss function that describes how well a model h performs on and instance $\boldsymbol{z} \in \mathcal{Z}$. For example, in supervised learning, $\boldsymbol{z}$ can be an input-output pair $(\boldsymbol{x}, y)$, and $\ell(h, \boldsymbol{z})$ can be the prediction error of $h(\boldsymbol{x})$ vs y. Or, in next-word prediction, $\boldsymbol{z}$ can be a document and $\ell(h, \boldsymbol{z})$ can be the average cross-entropy loss when using h to predict each of the next tokens in the document. In any case, for a test distribution D_{test} over $\boldsymbol{z}$, we evaluate the model through the *test loss* $\mathcal{L}_{D_{\text{test}}}(h) := \mathbb{E}_{\boldsymbol{z}\sim D_{\text{test}}}[\ell(h, z)]$.

Test Distribution. We consider test distributions consisting of a mixture of K components $\mathcal{D}_1, \ldots, \mathcal{D}_K$. A mixture $\mathcal{D}_{\boldsymbol{p}} = \sum_k p_k \mathcal{D}_k$ is then specified by mixing proportions $\boldsymbol{p} =$

$(p_1, \ldots, p_K) \in \Delta_K$ on the probability simplex Δ_K. We let $\boldsymbol{p}$ be the mixing proportions in the test distribution, i.e. $D_{\text{test}} = \mathcal{D}_{\boldsymbol{p}}$, and so the test loss is $\mathcal{L}_{\mathcal{D}_{\boldsymbol{p}}}(h) = \mathcal{L}_{\boldsymbol{p}}(h)$, where here and elsewhere we use the subscript $\boldsymbol{p}$ to denote the mixture $\mathcal{D}_{\boldsymbol{p}}$.

Learning Algorithm. We consider abstract "learning algorithm" $\mathcal{A}$, which, given training data (or sequence of training examples) $S \in \mathcal{Z}^N$ of size N, outputs a model $\mathcal{A}(S)$ with test loss $\mathcal{D}_{\boldsymbol{p}}(\mathcal{A}(S))$.

Training Distribution. We consider training on i.i.d. samples $S \sim \mathcal{D}_{\boldsymbol{q}}^N$ from mixtures $\mathcal{D}_{\boldsymbol{q}}$ of the same K components, but with potentially different mixing proportions $\boldsymbol{q} \in \Delta_K$. For training mixing proportions $\boldsymbol{q}$, we denote $L_N(\boldsymbol{p}, \boldsymbol{q}) = \mathbb{E}_{S \sim \mathcal{D}_{\boldsymbol{p}}^N}[\mathcal{L}_{\boldsymbol{p}}(\mathcal{A}(S))]$ the expected test error on $D_{\text{test}} = \mathcal{D}_{\boldsymbol{p}}$ when training with $D_{\text{train}} = \mathcal{D}_{\boldsymbol{q}}$ (we frequently drop the subscript N if its clear from context). The "non-shifted" expected test loss is then denoted $L_N^{\text{same}}(\boldsymbol{p}) = L_N(\boldsymbol{p}, \boldsymbol{p})$. In contrast, we denote $L_N^*(\boldsymbol{p}) = \min_{\boldsymbol{q} \in \Delta_K} L_N(\boldsymbol{p}, \boldsymbol{q})$ the test error with the best mixing ratios, and $\boldsymbol{q}^*$ the minimizing ratios. When $L^* < L^{\text{same}}$ and so $\boldsymbol{q}^* \neq \boldsymbol{p}$, this means we can benefit from mismatched training. **Our main analysis objective is to charactarize $\boldsymbol{q}^*$, L^* and the improvement over L^{same}.**

We can measure the mismatch benefit through the improvement in test error for a fixed training budget $L_N^{\text{ratio}} = L_N^*/L_N^{\text{same}}$. Or, we can consider the training complexity $N_\epsilon(\boldsymbol{p}, \boldsymbol{q}) = \min\ N$ s.t. $L_N(\boldsymbol{p}, \boldsymbol{q}) \leq \epsilon$ and the improvement $N_\epsilon^{\text{ratio}} := \frac{N_\epsilon^*(\boldsymbol{p})}{N_\epsilon^{\text{same}}(\boldsymbol{p})}$.

Specifying the Learning Model The expected test loss $L_N(\boldsymbol{p}, \boldsymbol{q})$, and so $\boldsymbol{q}^*$ and the benefit of mismatch, depend on the data distributions and learning behaviour of the algorithm. We capture these by modeling the *subpoluation error function* $e_k(\boldsymbol{n})$, i.e. the error on each component $\mathcal{D}_k$ when training with n_i examples from each component $\mathcal{D}_i$. That is, for a vector of sample sizes $\boldsymbol{n} = (n_1, \ldots, n_K) \in \mathbb{Z}_{\geq 0}^K$, denote $\boldsymbol{\mathcal{D}^n} = (\mathcal{D}_1)^{n_1} \times \cdots \times (\mathcal{D}_K)^{n_K}$ the distributions over samples with n_i examples from each component $\mathcal{D}_i$. Then $e_k(\boldsymbol{n}) = \mathbb{E}_{S \sim \boldsymbol{\mathcal{D}^n}}[\mathcal{L}_{\mathcal{D}_k}(\mathcal{A}(S))]$. When $e_k(\boldsymbol{n}) = g_k(n_k)$ depends only on the amount of within-component data, we say the components are *orthogonal*, meaning there is no transfer between them (as in our Chinese and European history example). The scalar function $g_k(n_k)$ then captures the *learning curve* for each component. But more generally, there might also be transfer, with data from one component helping learning on another.

In any case, the learnability function $e : \mathbb{Z}_{\geq 0}^K \to \mathbb{R}^K$, captures our "learning model". In each Section, we consider different forms of learning models and characterize $\boldsymbol{q}^*$ and L^* for these models.

Data Sets and Training Sequences In our analysis, we refer to the training budget N and our learning model specifying learning based on n_k examples per component k. We can think of N and $\boldsymbol{n}$ as specifying the number of training examples, in which case the training complexity is a sample complexity. Or, we can think of N as indicating the number of training steps, and n_k as indicating the number of steps in which an example from component k is used. In this case, training complexity is a measure of training time. Either interpretation is valid. But we should emphasize that we only study a dependence on *how many* examples are used from each component, *not* on the *order* (as in curriculum learning).

Learnabilities and Mixing Ratios. We model learning as a function of the *number* of examples from each component, but for our analysis, it will useful to introduce the function $\bar{e}_{N,k}(\boldsymbol{q}) = \mathbb{E}_{S \sim (\mathcal{D}_{\boldsymbol{q}})^n}[\mathcal{L}_k(\mathcal{A}(S))]$, which captures the expected error on component k with mixing proportions $\boldsymbol{q}$. We will refer to $\bar{e}_k(\boldsymbol{q})$ as the subpopulation error function in terms of the mixture $\boldsymbol{q}$. Since the per-component counts $\boldsymbol{n}$ are multinomial, we have $\bar{e}_N(\boldsymbol{q}) = \mathbb{E}_{\boldsymbol{n} \sim \text{Mult}(\boldsymbol{q}, N)}[e(\boldsymbol{n})] \in \mathbb{R}^K$ and $L_N(\boldsymbol{p}, \boldsymbol{q}) = \langle \boldsymbol{p}, \bar{e}_N(\boldsymbol{q}) \rangle$. Frequently for large sample size N, $\bar{e}_N(\boldsymbol{q})$ will concentrate around $e(\boldsymbol{q}N)$, and we will sometimes exploit this in the analysis, or analyze for $\bar{e}(\boldsymbol{q}) \approx e(\boldsymbol{q}N)$.

3 Orthogonal Power Law

Many machine learning tasks can be captured with power law error functions. Some classic examples include linear regression or learning VC classes, both of which have error rate $\propto \frac{1}{n}$, where n is the number of data samples. More recently, there have been many papers studying the loss curves for large language models for various tasks as a function of the compute budget in various scaling laws, such as the Chinchilla Scaling Law [Hoffmann et al., 2022].

To model these situations, we will first consider a setup where each of the K tasks is orthogonal and their subpopulation error functions in terms of the number of samples follow a simple power law.

Model 3.1 (Orthogonal Power Law Error Tasks)**.** There are K orthogonal tasks, each of which takes data from one of the K subpopulations $\mathcal{D}_i$ that appear in the test distribution with probability p_i and whose subpopulation error function $e_k(\boldsymbol{n})$ follows a power law, i.e. $e_k(\boldsymbol{n}) = \frac{A_k}{n_k^{\alpha_k} + B_k}$ for some $A_k > 0, B_k \geq 0$, and $0 < \alpha_k \leq 1$.[1]

In Proposition 3.2, we characterize the test error improvement from the positive distribution shift from optimal data mixing ratios in Model 3.1 when the size of the training data n is large.

Proposition 3.2 (Optimal Data Mixing Ratios For General Power Law)**.** *In Model 3.1, if for the exponents it holds that $\alpha_1 = \alpha_2 = \cdots = \alpha_S < \alpha_{S+1} \leq \alpha_{S+2} \leq \cdots \leq \alpha_K$ for some S then there exist $\varepsilon_1, \varepsilon_2 \geq 0$ that depend on α_i such that for any test data mixing ratio $\mathbf{p}$ and any $n > n_0(A_i, B_i, \alpha_i, p_i)$ we have that the following holds*

$$q_i^* = \frac{1}{N^{\frac{\alpha_i - \alpha_1}{\alpha_i + 1}}} \left(\frac{(\alpha_i p_i A_i)}{\left(\sum_{i=1}^{S} (\alpha_i p_i A_i)^{\frac{1}{\alpha_1 + 1}} \right)^{\alpha_1 + 1}} \right)^{\frac{1}{\alpha_i + 1}} + o\left(\frac{1}{N^{\frac{\alpha_i - \alpha_1}{\alpha_i + 1}}} \right) \tag{1}$$

$$L^{\text{same}}(\boldsymbol{p}) = \frac{1}{N^{\alpha_1}} \sum_{i=1}^{S} p_i^{1-\alpha_1} A_i + o\left(\frac{1}{N^{\alpha_1 + \varepsilon_1}} \right). \tag{2}$$

$$L^*(\boldsymbol{p}) = \frac{1}{N^{\alpha_1}} \left(\sum_{i=1}^{S} (\alpha_i p_i A_i)^{\frac{1}{\alpha_i + 1}} \right)^{\alpha_1} \left(\sum_{i=1}^{S} \frac{(p_i A_i)^{\frac{1}{\alpha_i + 1}}}{\alpha_i^{\frac{\alpha_i}{\alpha_i + 1}}} \right) + o\left(\frac{1}{N^{\alpha_1 + \varepsilon_2}} \right). \tag{3}$$

The $o(\cdot)$ notation hides dependence on A_i, B_i, p_i, K and α_i.

Proposition 3.2 shows that in the power law Model 3.1, positive distribution shift from optimal data mixing ratios improves the prefactor of the test error dependence on the number of data samples N but does not change the decay rate in terms of N. For the proof of Proposition 3.2 and a more precise statement, see Appendix A.1.

To show that this can have significant implications for making training more data efficient, we show the improvement from this positive distribution shift on the sample complexity in the case where we have one majority population and $K - 1$ minority populations that all have the same power exponent α. This will also include the test-taking example from Section 1.

Corollary 3.3 (Sample Complexity Improvement From Optimal Data Mixing For General Power Law)**.** *Consider Model 3.1 with $S = K$, i.e. $\alpha_1 = \cdots = \alpha_K = \alpha$ and $A_1 = \cdots = A_K = A$ with $\boldsymbol{p} = (p, \frac{1-p}{K-1}, \ldots, \frac{1-p}{K-1})$. We have that for any $\epsilon > 0$*

$$N_\epsilon^{ratio}(\boldsymbol{p}) \leq (1 - p) + 2\frac{\alpha + 1}{\alpha} \left(\frac{p}{1 - p} \right)^{\frac{1}{\alpha + 1}} K^{-\frac{\alpha}{\alpha + 1}}.$$

Furthermore, the optimal mixing ratios are given by $q_1^* \propto p^{\frac{1}{\alpha+1}}$ and $q_i^* \propto \left(\frac{1-p}{K-1} \right)^{\frac{1}{\alpha+1}}$ for $i \geq 2$.

Corollary 3.3 demonstrates an example case, that if we have one majority population and a number of minority populations, the positive distribution shift from optimal data mixing ratio significantly improves sample complexity. For fixed p, if K is large enough, $N^{\text{ratio}}(\boldsymbol{p})$ will be close to $N^{\text{ratio}}(\boldsymbol{p}) \approx 1 - p < 1$, i.e. we get sample complexity improvement of up to p. For example, for $p = 0.7$, $\alpha = 0.28$, and $K = 100$, for any $\epsilon > 0$, $N_\epsilon^{\text{ratio}}(\boldsymbol{p}) \approx 0.75$, i.e. we achieve the same error with $\approx 25\%$ less samples. We illustrate this in Figure 2. For the proof of Corollary 3.3, see Appendix A.1.

Furthermore, the test taking example considered in the introduction Section 1 follows from Corollary 3.3, by taking $K = 2$, $\alpha = 1$, and $\boldsymbol{p} = (0.9, 0.1)$. In particular, this shows that the optimal studying budget allocation is $\boldsymbol{q}^* = (0.75, 0.25)$ and the improvement is $N^{\text{ratio}}(\boldsymbol{p}) = 0.8$. This means that if you study for the exam with the right mixing ratio $\boldsymbol{q}^*$, you would need to study 20% less time to achieve the same score as compared to using the test mixing ratio $\boldsymbol{p}$. Further, taking $\alpha = \frac{1}{2}$ we get the second example on Figure 2. This shows that we indeed get $\boldsymbol{q}^* = (0.812\ldots, 0.188\ldots)$ and $N^{\text{ratio}}(\boldsymbol{p}) = 0.944$.

[1] We will also use the convention that if $B_k = 0$ then $e_k(\boldsymbol{n}) = \min\{C_k, \frac{A_k}{n_k^{\alpha_k}}\}$ for some $C_k > 0$. This will prevent $L(\boldsymbol{p}, \boldsymbol{q})$ from blowing up to infinity.

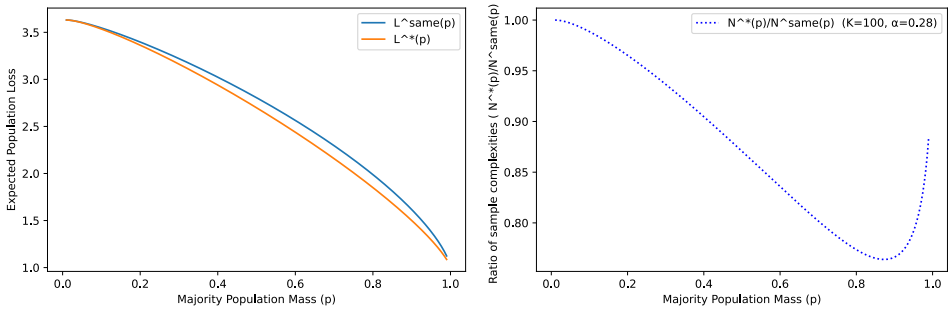

Figure 2: We consider the setup of Corollary 3.3 with $A = 1,\ \alpha = 0.28$, $K = 100$, and some fixed N. On the left plot, we show the "non-shifted" expected population loss $L^{\text{same}}(\boldsymbol{p})$ and the optimally mixed expected population loss $L^*(\boldsymbol{p})$ as a function of majority population mass p. On the right plot, we show the ratio of sample complexities for any fixed $\epsilon > 0$, $N_\epsilon^{\text{ratio}}(\boldsymbol{p})$ as a function of the mass of the majority population, p. We can see significant improvement in the sample complexity from the positive distribution shift from using optimal mixing ratio, even up to $\approx 25\%$.

4 Orthogonal Memorization Tasks

We consider a task of memorizing a number of unique elements from a dataset of fixed size, where the test distribution is a mixture of the tasks we are trying to memorize.

Model 4.1 (Orthogonal Memorization Tasks)**.** Suppose there are K tasks, each of which is a memorization of a unique element. The test distribution is a mixture of these K tasks, where the k-th task appears with probability p_k. In this case the subpopulation error functions in terms of $\boldsymbol{n}$ is given by $e_k(\boldsymbol{n}) = \mathbf{1}_{\{n_k=0\}}$.

The following theorem characterizes the test error improvement from the positive distribution shift from optimal data mixing ratios in the Orthogonal Memorization Task Model 4.1.

Theorem 4.2 (Optimal Data Mixing Test Error Improvement For Orthogonal Memorization Task)**.** *In Model 4.1, for all $\boldsymbol{p} \in \Delta^{K-1}$ with $p_1 \geq p_2 \geq \cdots \geq p_K$, the expected loss when training on n samples is given by*

$$L^{\text{same}}(\boldsymbol{p}) = \sum_{k=1}^{K} p_k(1-p_k)^N \tag{4}$$

$$L^*(\boldsymbol{p}) = (K_N(\boldsymbol{p}) - 1)\delta_N(\boldsymbol{p}) + \sum_{k=K_N(\boldsymbol{p})+1}^{K} p_k, \tag{5}$$

where $\delta_N(\boldsymbol{p}) \in [p_{K_N(\boldsymbol{p})+1},\ p_{K_N(\boldsymbol{p})})$ and $K_N(\boldsymbol{p})$ is defined as follows:

$$K_N(\boldsymbol{p}) := \max\left\{ s \leq K : \sum_{k=1}^{s-1} (1 - (p_s/p_k)^{1/(K-1)}) < 1 \right\}. \tag{6}$$

To understand the magnitute of the test error improvement in Theorem 4.2, we will assume that the test proportions $\boldsymbol{p}$ follow a power law $p_k = \Theta(k^{-\alpha})$ for some $\alpha > 1$ and that the number of tasks to memorize K is larger than the size of the training set N. In this case, we show that the improvement from positive distribution shift Theorem 4.2 improves even the test error scaling in terms of N. For the proof of Theorem 4.2, see Appendix A.2.

Corollary 4.3 (Test Error Improvement For Orthogonal Memorization Taks with Power Law Test Mixing Ratios)**.** *If $p_k = \Theta(k^{-\alpha})$ for some $\alpha > 1$ and $K = \Omega(N)$, then*

$$L^{\text{same}}(\boldsymbol{p}) = \Theta(N^{-1+\frac{1}{\alpha}}), \qquad L^*(\boldsymbol{p}) = \Theta(N^{-\alpha+1}).$$

For example, when $\alpha = 1.5$, we have $L^{\text{same}}(\boldsymbol{p}) = \Theta(N^{-1/3})$ and $L^*(\boldsymbol{p}) = \Theta(N^{-1/2})$. For the proof of Corollary 4.3, see Appendix A.2.

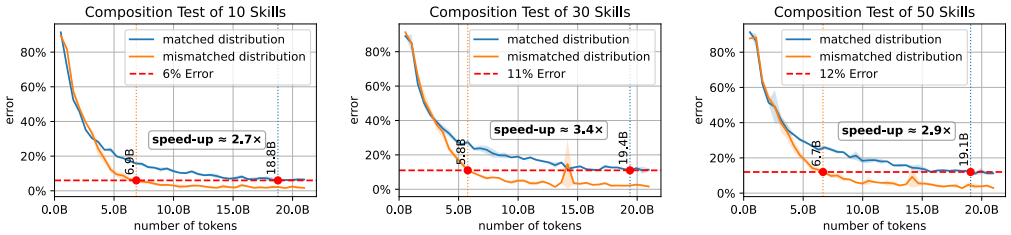

Figure 3: Mismatched distribution improves the test accuracy of a language model in solving a synthetic CoT reasoning task on skill composition (Section 5). During test, the model is asked to compose several functions following a power law. Instead of training directly on this task (blue curve), mixing with another task that uniformly samples the functions improves the final accuracy (orange curve).

5 Connection to Skill Composition

All the above analyses focus on the case where tasks are orthogonal. However, if we already know that the test distribution can be decomposed into K tasks, then maybe we should deal with these K tasks independently. So why do we have test mixing ratios in the first place?

We note here that in some cases, we may need to compose these K tasks later at inference time, and the test mixing ratios can come from the proportions in the composition. Imagine that we are training a language model to do mathematical reasoning. Each problem may involve several math skills, and a language model can acquire a math skill only if it sees the skill enough times during training. This can be conceptually modeled as the orthogonal memorization task discussed above, but at inference time, the language model has to sequentially apply the math skills in its chain of thought (CoT). The natural distribution of math skills then determines the test mixing ratios we care about.

We demonstrate this in a concrete synthetic task on skill composition. There are K skills, where the i-th skill is a function g_i that maps a number from $\{0, \ldots, 9\}$ to $\{0, \ldots, 9\}$. Each skill has a unique English name. Assume that all these skills are randomly sampled: the names are uniformly random from a name set, and each g_i is uniformly random among all possible functions that map from $\{0, \ldots, 9\}$ to $\{0, \ldots, 9\}$. At inference time, a set of k skills $g_{i_1}, \ldots, g_{i_k}$ are sampled IID following a power law with exponent $\alpha = 1.5$. The language model is prompted with the names of these skills and a number $x \in \{0, \ldots, 9\}$: "`[x] -> [skill name 1] -> [skill name 2] -> ⋯ -> [skill name k]`". The model is expected to output the result after function composition: $y = g_{i_k}(g_{i_{k-1}}(\cdots g_{i_1}(x) \cdots))$.

Let D_{test} be the distribution of the above prompt and a CoT calculating the correct answer, with $M = 10^5$, k sampled uniformly from 10 to 50. Is the best strategy just training on the same distribution ($D_{\text{train}} = D_{\text{test}}$)? Inspired by our calculation for the orthogonal memorization task above, properly adjusting the occurrence probability for each skill may lead to better test accuracy. To demonstrate this, we construct another distribution $\mathcal{D}_{\text{uniform}}$ consisting of strings in the form of "`[x] [skill name] = [expected output]`", where the skill and input number are uniformly sampled. In Figure 3, we conduct experiments with a model with GPT-2 architecture and ∼50M parameters. We show that training with $D_{\text{train}} = 30\% \cdot \mathcal{D}_{\text{uniform}} + 70\% \cdot D_{\text{test}}$ significantly outperform training with D_{test} directly. We defer the experiment details to Appendix C.

6 Non-orthogonal Tasks and Transfer Learning

Many transfer learning setups, such as multi-task learning of linear classifiers over linear representation with feature learning Baxter [2011], Maurer [2009], Pontil and Maurer [2013], Aliakbarpour et al. [2024] and multi-task learning with shared sparsityWang et al. [2016, 2017], the subpopulation error functions $e_k(\boldsymbol{n})$ can be written in the form $e_k(\boldsymbol{n}) = \frac{A_{0,k}}{(n_1+\cdots+n_k)^{\alpha_k}} + \frac{A_{1,k}}{n_k^{\alpha_k}}$. For example, in multi-task learning of shared sparsity Wang et al. [2017], the error bound takes this form with $\alpha_1 = \cdots = \alpha_K = 1$.

To model all of these cases, we consider the following model of transfer learning.

Model 6.1 (Standard Transfer Learning Model)**.** There are K subpopulations, each of which appears in the test distribution with proportion p_k. The subpopulation error functions depend on the number of samples $\boldsymbol{n}$ as $e_k(\boldsymbol{n}) = \frac{A_{0,k}}{(n_1+\cdots+n_k)^{\alpha_k}} + \frac{A_{1,k}}{n_k^{\alpha_k}}$, for some $A_{0,k}, A_{1,k} > 0$ and $0 < \alpha_k \leq 1$.

Interestingly, the Standard Transfer Learning Model 6.1 is equivalent to the setup of Orthogonal Power Law Tasks Model 3.1 in the sense that we can understand optimal data mixing ratio $\boldsymbol{q}^*$ and the error improvement of the Standard Transfer Learning model from a specific instance of the Orthogonal Power Law model. Namely, the transfer term in each of the subpopulation loss functions can be decomposed into a transfer error term and a specific task error term $e_k(\boldsymbol{n}) = e_k^{\text{transfer}}(\boldsymbol{n}) + e_k^{\text{spec}}(\boldsymbol{n})$, where $e_k^{\text{transfer}}(\boldsymbol{n}) = \frac{A_{0,k}}{(n_1+\cdots+n_k)^{\alpha_k}}$ is independent of the distribution of samples across different tasks, and $e_k^{\text{spec}}(\boldsymbol{n}) = \frac{A_{1,k}}{n_k^{\alpha_k}}$ only depends on n_k. Therefore, the transfer error term $e_k^{\text{transfer}}(\boldsymbol{n})$ in each of the subpoluation error functions will only offset the final expected loss $L(\boldsymbol{p}, \boldsymbol{q})$ by $\sum_{i=1}^K p_i \frac{A_{0,k}}{N^{\alpha_k}}$, which only depends on the total number of samples N. On the other hand, the specific task error terms $e_k^{\text{spec}}(\boldsymbol{n})$ can be thought of as orthogonal tasks and will behave tha same as in Model 3.1. So, for the Standard Transfer Learning Model 6.1, the optimal data mixing ratio $\boldsymbol{q}^*$ and the expected test losses $L^*(\boldsymbol{p})$ and $L^{\text{same}}(\boldsymbol{p})$ are given by Equation (1) and Equation (2) respectively in Proposition 3.2 with A_k being replaced by $A_{1,k}$.

6.1 Data Mixing Transfer Learning.

Ye et al. [2025] consider the problem of estimating the outcome performance of a large langue model trained on a mixture of domains. In particular, they find that an exponential function over the linear combinations of mixing proportions leads to good prediction. Namely, they fix the training budget N and only vary the mixing ratio $\boldsymbol{q}$ and show that the validation loss on i-th domain can be predicted well by a function of the form $c_i + b_i \exp\left(-\sum_{j=1}^K t_{ij} q_j\right)$, where c_i, b_i, t_{ij} are parameters to fit. Following their work, we propose the following model for the Data Mixing Transfer Learning.

Model 6.2 (Data Mixing Transfer Learning)**.** There are K subpopulations, each of which appears with probability p_k in the test distribution. Each of the subpopulation error functions in terms of the mixing ratio $\boldsymbol{q}$ are $\bar{e}_k(\boldsymbol{q}) = c_k + b_k \exp\left(-\sum_{j=1}^K t_{ij} q_j\right)$ for some constants c_k and $b_k > 0, t_{ij}$.

We note that even though Model 6.2 is indeed not defined by the subpopulation error functions $e_k(\boldsymbol{n})$, it is precisely the setup that Ye et al. [2025] consider. This slightly deviates from our main setup, which focuses on specifying models by their error functions. However, when the number of samples N is large, it is reasonable to make the approximation that $e_k(\boldsymbol{n}) \approx e_k(\boldsymbol{q}N)$, and Model 6.2 can be interpreted as being defined by the subpopulation error functions of the form $e_k(\boldsymbol{n}) = c_k(|\boldsymbol{n}|) + b_k(|\boldsymbol{n}|) \exp\left(-\sum_{j=1}^K t_{ij}(|\boldsymbol{n}|) n_j\right)$, where c_k, b_k, and t_{ij} are functions that depend only on the total compute budget $N = |\boldsymbol{n}|$.

The following proposition characterizes the test error improvement from the positive distribution shift coming from the optimal data mixing ratio in the data mixing transfer model.

Proposition 6.3 (Optimal Train Data Mixing Ratio for Data Mixing Transfer Learning Model)**.** *In Model 6.2, if the coefficients t_{ij} are such that $\boldsymbol{T}$ is invertible and and $(\boldsymbol{T}^T)^{-1}\boldsymbol{1} > 0$, and $p_i \neq 0$ for all i, the following hold*

$$\boldsymbol{q}^* = (\boldsymbol{T})^{-1}\left(\frac{1 + \boldsymbol{1}^\top \boldsymbol{T}^{-1}\tau}{\boldsymbol{1}\boldsymbol{T}^{-1}\boldsymbol{1}}\boldsymbol{1} - \tau\right)$$

$$L^{\text{same}}(\boldsymbol{p}) = \sum_{i=1}^K c_i p_i + \sum_{i=1}^K p_i b_i \exp\left(-\sum_{j=1}^K t_{ij} p_j\right)$$

$$L^*(\boldsymbol{p}) = \sum_{i=1}^K c_i p_i + \exp\left(\frac{-1 - \boldsymbol{1}^\top \boldsymbol{T}^{-1}\tau}{\boldsymbol{1}^T \boldsymbol{T}^{-1}\boldsymbol{1}}\right) \boldsymbol{1}^T (\boldsymbol{T}^\top)^{-1}\boldsymbol{1},$$

where τ is a vector with entreis $\tau_l = \log\left(\frac{[(\boldsymbol{T}^\top)^{-1}\boldsymbol{1}]_l}{p_l b_l}\right)$.

Proposition 6.3 shows the positive distribution from the optimal data mixing for Model 6.2. Note that the additional conditions on $\boldsymbol{T}, p_i$ are technical conditions used in order to simplify presentation. For the complete statement and the proof of Proposition 6.3, see Appendix A.3.

To demonstrate how large the gap can be, we consider the problem of data mixing transfer learning Model 6.2 with $K = 2$ tasks and a one-directional transfer from the second to the first task.

Corollary 6.4 (Optimal Data Mixing Ratio Can Have Significant Improvement in the Transfer Learning Model)**.** *Let $K = 2$, let $\boldsymbol{p} = (\frac{1}{2}, \frac{1}{2})$, and let $b_1 = b_2 = b > 0$. If $\boldsymbol{T} = \begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix}$ then we have that*

$$L^{\text{same}} - L^* = 2be^{-\frac{1}{2}} \left(1 - \frac{1}{4}\alpha + O(a^2)\right).$$

Furthermore, if we let $C = \frac{c_1+c_2}{2}$ and $B = be^{-\frac{1}{2}}$ then we have that

$$L^{ratio} = \frac{L_N}{L^*} = \frac{C-B}{C+B} + \frac{BC}{2(B+C)^2}\alpha + O(\alpha^2)$$

Corollary 6.4 shows that for two tasks with a small of transfer between the second to the first we can have error improvement from the positive distribution shift by mismatching training and test distribution, that is $L^{\text{ratio}} \approx \frac{C-B}{C+B} < 1$ for small α. For the proof of Corollary 6.4, see Appendix A.3.

7 It's Almost Always Better to Mismatch

So far, we have shown the existence of and quantified the positive distribution shift coming from mistmatched test and train data mixing ratios for the cases of orthogonal power law tasks in Section 3, orthogonal memorization tasks in Section 4, and standard transfer learning and data mixing transfer learning in Section 6. that positive distribution shift from mismatching test and train mixing ratios exists. In this section, we will provide further mathematical justification that a positive distribution shift coming from the data mixing ratio almost always exists. That is, we show that it's almost always better to mismatch the training and test distributions: $\boldsymbol{q}^* \neq \boldsymbol{p}$ and $L^*(\boldsymbol{p}, \boldsymbol{q}^*) < L^{\text{same}}(\boldsymbol{p})$.

More precisely, we will show that either the test data mixing ratio is on a measure zero set of the simplex or the subpopulation error functions $e_k(\boldsymbol{n})$ have to be very specific functions, which are meaningless. For example, in the case of orthogonal tasks, either the test mixing ratio is on a measure zero subset or the subpopulation error functions $e_k(\boldsymbol{n})$ are all constants, which we show in Corollary 7.4.

We define the probability simplex $\Delta^{K-1} := \left\{\boldsymbol{p} \in \mathbb{R}^K : \boldsymbol{p} \geq 0,\ |\boldsymbol{p}| = 1\right\}$, and its interior $\Delta_+^{K-1} := \left\{\boldsymbol{p} \in \mathbb{R}^K : \boldsymbol{p} > 0,\ |\boldsymbol{p}| = 1\right\}$, where $|\boldsymbol{p}| := \sum_{k=1}^K p_k$. We will define $f_k(\boldsymbol{p})$ by extending the domain of each $\bar{e}_k(\boldsymbol{p})$ to the set of non-zero, non-negative vectors $\mathbb{R}^K_{\geq 0} \setminus \{\boldsymbol{0}\}$ by defining $f_k(\boldsymbol{p}) := \bar{e}_k(\frac{\boldsymbol{p}}{|\boldsymbol{p}|})$. We further define $L^{\text{same}}(\boldsymbol{p}) := \sum_{k=1}^K p_k f_k(\boldsymbol{p})$, which extends the definition of L^{same} to the set of non-zero, non-negative vectors $\mathbb{R}^K_{\geq 0} \setminus \{\boldsymbol{0}\}$.

Condition 7.1 (Conservation Condition)**.** $(f_1(\boldsymbol{p}), \ldots, f_K(\boldsymbol{p})) = \nabla L^{\text{same}}(\boldsymbol{p})$ *for all* $\boldsymbol{p} \in \mathbb{R}^K_{\geq 0} \setminus \{\boldsymbol{0}\}$.

Theorem 7.2 (Positive Distribution Shift Almost Always Exists For Data Mixing)**.** *For any set of subpopulations $\mathcal{D}_1, \ldots, \mathcal{D}_K$ and any learning algorithm $\mathcal{A}$, either Condition 7.1 holds, or there exists a zero-measure set U on Δ^{K-1} such that for all $\boldsymbol{p} \in \Delta^{K-1} \setminus U$, $L_N^*(\boldsymbol{p}) < L^{\text{same}}(\boldsymbol{p})$.*

Theorem 7.2 shows that either $\boldsymbol{p}$ is on a measure zero set U on Δ^{K-1} or the Conservation Condition 7.1 must hold. We will show that Conservation Condition 7.1 happens only for very specific cases of subpopulation error functions.

Conservation Condition Rarely Holds. First, we will show that if the subtasks are orthogonal, the conservation condition Condition 7.1 is only satisfied if all of the subpopulation error functions are constants.

Lemma 7.3 (Orthogonal Tasks)**.** *If $K \geq 3$, and if for all $k \in [K]$, $f_k(\boldsymbol{p}) = g_k(\frac{p_k}{|\boldsymbol{p}|})$ for some function g_k, then Condition 7.1 holds if and only if g_k's are all constant functions.*

Theorem 7.2 and Lemma 7.3 together show that in the case of orthogonal tasks, positive distirbution shift always exists by changing the training data mixing ratio away from the test mixing ratio, unless all the subpopulation error functions are constant.

Corollary 7.4 (Positive Distribution Shift Always Exists for Orthogonal Tasks)**.** *For any set of $K \geq 3$ subpopulations $\mathcal{D}_1, \ldots, \mathcal{D}_K$ and any learning algorithm $\mathcal{A}$, if there exists subpopulation $k \in [K]$ such that its error function e_k is not a constant functions over $[N]$ where N is the number of total samples then there exists a measure zero set U on Δ^{K-1} such that for all $\boldsymbol{p} \in \Delta^{K-1} \setminus U$ positive distribution shift from data mixing exists in the sense that there is $\boldsymbol{q}^* \neq p$ for which $L_N(\boldsymbol{p}, \boldsymbol{q}) = L^*(\boldsymbol{p}) < L^{\text{same}}(\boldsymbol{p})$.*

Further, we show that if the Conservation Condition 7.1 is satisfied, then one function f_i determines the rest up to a constant.

Lemma 7.5. *If both $(f_1, \ldots, f_K, L^{\text{same}})$ and $(\hat{f}_1, \ldots, \hat{f}_K, \hat{L}^{\text{same}})$ satisfy Condition 7.1, and if $f_i = \hat{f}_i$ for some $i \in [m]$, then for all $k \neq i$, $f_k(\boldsymbol{p}) = \hat{f}_k(\boldsymbol{p}) + C_k$ for some constant C_k.*

The above Lemma 7.5 implies that for every k and corresponding error function $e_k(\boldsymbol{n})$, there exists at most one tuple of error functions $\{e_j\}_{j=1, j\neq k}^K$ (up to a individual constant offset for each error function e_j) that positive distribution shift does not happen for $\boldsymbol{p}$ of positive measure. This further implies the following corollary.

Corollary 7.6 (Positive Distribution Shift *Almost* Always Exists for General Tasks)**.** *For any set of $K \geq 3$ subpopulations $\mathcal{D}_1, \ldots, \mathcal{D}_K$ and any learning algorithm $\mathcal{A}$, for all $\boldsymbol{p} \in \Delta_+^{K-1}$, the configuration of $[e_k(\boldsymbol{n})]_{k\in[K], \boldsymbol{n}}$ that positive distribution shift does not happen is zero-measure.*

Corollary 7.6 shows that either the test mixing ratio $\boldsymbol{p}$ is on a set of measure zero on the simplex or the configuration of subpopulation error functions $e_k(\boldsymbol{n})$ is on a set of measure zero. This implies that positive distribution shift exists *almost* always.

8 Related Works

Distribution Shift That is Not Harmful. The benefits of mismathcing the training and test distribution has already been in studied in some settings. González and Abu-Mostafa [2015] demonstrate in many linear regression problems that mismatched training and test distributions can outperform matched ones. Unlike in our paper, they do not restrict to changing the train distribution only through data mixing, so their results do not fit our framework. On the other hand, we explicitly characterize the positive distribution shift, while González and Abu-Mostafa [2015] only show its existence for linear regression problems and are only able to characterize the distribution explicitly in very special cases. Canatar et al. [2021] show how in high-dimensional kernel regression problems to numerically optimize the training distribution for better test performance. However, they do not characterize the positive distribution shift, but rather only show how to numerically find it for kernel regression. Similarly, they do not restrict the test distribution to one coming from a data mixture, so their results do not fit our framework.

Data Mixing. There a number of recent empiricaly works that consider the same setting of data mixing as we do. Ye et al. [2025] introduce data mixing laws, quantitative empirical predictions of large language model performance based on the data mixture proportions. Furthermore, they show experimental results demonstrating that their approach significantly decreases the number of steps needed to reach certain performance. This paper informed our data mixing transfer model and fits in our framework. Goyal et al. [2024] show that data curation for VLMs cannot be compute agnostic. They introduce neural scaling laws that allow for estimating performance on multiple data pools without jointly training on them. Their work fits our framework. Similarly, we also find that optimal mixing ratios are not compute agnostic, specifically in the orthogonal power law tasks, orthogonal memorization task, and standard transfer learning task. Jiang et al. [2025] introduce an algorithm for online optimization of data distributions, that adjusts mixture based on the estimated per-domain learning potential, achieving comparable or better performance than previous methods while maintaing compuatational efficiency. While all of these works consider the same phenomena of changing the training mixing ratio to improve test performacne, the main difference between our work and theirs is that we consider positive distribution shift from data mixing ratio in a broader context and from the theoretical standpoint as well.

References

Maryam Aliakbarpour, Konstantina Bairaktari, Gavin Brown, Adam Smith, Nathan Srebro, and Jonathan Ullman. Metalearning with very few samples per task. In Shipra Agrawal and Aaron Roth, editors, *Proceedings of Thirty Seventh Conference on Learning Theory*, volume 247 of *Proceedings of Machine Learning Research*, pages 46–93. PMLR, 30 Jun–03 Jul 2024. URL https://proceedings.mlr.press/v247/aliakbarpour24a.html.

Jonathan Baxter. A model of inductive bias learning. *CoRR*, abs/1106.0245, 2011. URL http://arxiv.org/abs/1106.0245.

Abdulkadir Canatar, Blake Bordelon, and Cengiz Pehlevan. Out-of-distribution generalization in kernel regression. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 12600–12612. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/691dcb1d65f31967a874d18383b9da75-Paper.pdf.

Carlos R. González and Yaser S. Abu-Mostafa. Mismatched training and test distributions can outperform matched ones. *Neural Computation*, 27(2):365–387, 2015. doi: 10.1162/NECO_a_00697.

Sachin Goyal, Pratyush Maini, Zachary C. Lipton, Aditi Raghunathan, and J. Zico Kolter. Scaling laws for data filtering—data curation cannot be compute agnostic. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22702–22711, 2024. doi: 10.1109/CVPR52733.2024.02142.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. An empirical analysis of compute-optimal large language model training. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=iBBcRUlOAPR.

Yiding Jiang, Allan Zhou, Zhili Feng, Sadhika Malladi, and J Zico Kolter. Adaptive data optimization: Dynamic sample selection with scaling laws. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=aqok1UX7Z1.

Andreas Maurer. Transfer bounds for linear feature learning. *Machine Learning*, 75:327–350, 2009. URL https://api.semanticscholar.org/CorpusID:14682470.

Massimiliano Pontil and Andreas Maurer. Excess risk bounds for multitask learning with trace norm regularization. In Shai Shalev-Shwartz and Ingo Steinwart, editors, *Proceedings of the 26th Annual Conference on Learning Theory*, volume 30 of *Proceedings of Machine Learning Research*, pages 55–76, Princeton, NJ, USA, 12–14 Jun 2013. PMLR. URL https://proceedings.mlr.press/v30/Pontil13.html.

Jialei Wang, Mladen Kolar, and Nathan Srerbo. Distributed multi-task learning. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 751–760, Cadiz, Spain, 09–11 May 2016. PMLR. URL https://proceedings.mlr.press/v51/wang16d.html.

Jialei Wang, Mladen Kolar, Nathan Srebro, and Tong Zhang. Efficient distributed learning with sparsity. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3636–3645. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/wang17f.html.

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=jjCB27TMK3.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

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

   Answer: [Yes]

   Justification: Yes, the main claim accuretly reflects the paper's contribution and scope.

   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: Yes, we discuss the limitations of our work and clearly define the scope of each of our claims.

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 provide full set of assumptions and complete and corrected proofs in the appendix. For some of the claims, we only state an informal or a limited scope version in the main body for the ease of presentation.

   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: Yes, we disclose the information needed to reproduce the experiments.

   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: Yes, we provide the access in to the code and data in the appendix.

   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: Yes, we specify all the details of the experiment necessary to understand and reproduce the experiments.

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: Yes, we provide information about statistical significance of results where appropriate.

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: Yes, we provide sufficient information on the computer resources needed to reproduce the experiments in the appendix.

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: Yes, our research conforms in every aspect to 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: [NA]

Justification: As this is mainly a theoretical paper, there is no immediate societal impact of the owrk.

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: The paper poses 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: Yes, we properly credit all the original owners of assets where due.

    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: We do not realease new assets.

    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: The paper does not involve crowdourcing nor research with human subjects.

    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: See previous point.

    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: The core methods developed in this research do not involve LLMs as any important, original, or non-standard components.

    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.